Amazon GuardDuty ユーザーガイド

Amazon GuardDuty



Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon GuardDuty: Amazon GuardDuty ユーザーガイド

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon の商標およびトレードドレスは Amazon 以外の製品およびサービスに使用することはできません。また、お客様に誤解を与える可能性がある形式で、または Amazon の信用を損なう形式で使用することもできません。Amazon が所有していないその他のすべての商標は Amazon との提携、関連、支援関係の有無にかかわらず、それら該当する所有者の資産です。

Table of Contents

GuardDuty とは 1
GuardDuty の機能 2
PCI DSS コンプライアンス 7
GuardDuty の料金 7
GuardDuty の 30 日間無料トライアルの使用 8
無料利用枠での Malware Protection for S3 の使用 9
GuardDuty へのアクセス 10
概念と主要な用語 11
開始方法 16
[開始する前に] 16
ステップ 1: Amazon GuardDuty を有効にする 18
ステップ 2: サンプル検出結果を生成し、ベーシックなオペレーションの詳細を確認する 21
ステップ 3: Amazon S3 バケットへの GuardDuty の検出結果のエクスポートを設定する 22
ステップ 4: SNS を使用して GuardDuty の検出結果アラートを設定する 28
次のステップ 31
基本データソース 33
AWS CloudTrail 管理イベント 33
GuardDuty が AWS CloudTrail グローバルイベントを処理する方法 34
VPC フローログ 35
Route53 Resolver DNS クエリログ 36
Extended Threat Detection 37
攻撃シーケンスの脅威シナリオの例 37
Extended Threat Detection の仕組み 38
脅威検出を最大化するための保護プランの有効化 39
Amazon EKS クラスターでの攻撃シーケンスの検出 40
Amazon S3 バケット内の攻撃シーケンスの検出 41
Amazon ECS クラスターでの攻撃シーケンスの検出 41
Amazon EC2 インスタンスグループの攻撃シーケンスの検出 42
AI ワークロードを含む攻撃シーケンスの検出 43
GuardDuty コンソールの Extended Threat Detection 43
攻撃シーケンスの検出結果の理解と管理 44
その他のリソース 44
保護プラン 45
概要 45

すべての保護プランの設定 45
自動有効化オプション 46
Runtime Monitoring の設定 46
双方向制約 47
組織の統計 48
アカウントページの統合 48
AI 保護 49
AI Protection の仕組み 50
AI Protection の料金 51
AI ワークロードに対する追加の脅威検出 51
マルチアカウント環境で AI Protection を有効にする 52
スタンドアロンアカウントの AI Protection の有効化 56
EKS Protection 58
EKS Protection における EKS 監査ログ 59
マルチアカウント環境での EKS Protection の有効化 59
スタンドアロンアカウントの EKS Protection の有効化 67
S3 Protection 69
AWS CloudTrail S3 のデータイベント 70
GuardDuty が S3 の CloudTrail データイベントを使用する仕組み 70
攻撃シーケンスに S3 の CloudTrail データイベントを使用する GuardDuty 71
マルチアカウント環境での S3 Protection の有効化 71
スタンドアロンアカウントの S3 Protection の有効化 79
ランタイムモニタリング 81
仕組み 82
有効化 83
Amazon ECS へのデプロイ 84
Amazon EC2 にデプロイする 86
Fargate で Amazon ECS にデプロイする 87
エージェント接続 89
Runtime Monitoring カバレッジの追跡 90
30 日間の無料トライアル 91
GuardDuty のトライアル期間を使用している、または EKS Runtime Monitoring を一度も有効にしたことがない 92
Runtime Monitoring を開始する前に EKS Runtime Monitoring を有効にしました 92
前提条件 93
EC2 インスタンスの場合 94

ECS-EC2 Bottlerocket の場合 98
Fargate (ECS のみ) クラスターの場合 102
EKS クラスターの場合 108
サポートされている CPU アーキテクチャ、オペレーティングシステム、カーネルバージョン 111
Runtime Monitoring の有効化 112
マルチアカウント環境の Runtime Monitoring の有効化 113
スタンドアロンアカウントの Runtime Monitoring の有効化 116
GuardDuty セキュリティエージェントの管理 117
Amazon EC2 リソースの自動エージェント 117
Amazon EC2 リソースの手動エージェント管理 130
ECS-EC2 Bottlerocket リソースの自動エージェント 147
Fargate (Amazon ECS のみ) リソースの自動エージェント 148
Amazon EKS リソースの自動エージェント 184
Amazon EKS クラスターの手動エージェント管理 222
EKS アドオンパラメータを設定する 231
VPC エンドポイント設定の検証 234
ランタイムカバレッジの問題とトラブルシューティング 236
Amazon EC2 リソースのカバレッジとトラブルシューティング 237
ECS-EC2 Bottlerocket のカバレッジとトラブルシューティング 253
Amazon ECS クラスターのカバレッジとトラブルシューティング 261
Amazon EKS クラスターのカバレッジとトラブルシューティング 277
CPU とメモリモニタリングの設定 293
共有 VPC の使用 294
仕組み 295
前提条件 296
自動エージェントでの IaC の使用 297
IaC リソース依存関係グラフの概要 298
一般的な問題 - IaC でのリソースの削除 298
収集されたランタイムイベントタイプ 300
イベントを処理する 300
コンテナイベント 302
AWS Fargate (Amazon ECS のみ) タスクイベント 303
Kubernetes ポッドイベント 303
ドメインネームシステム (DNS) イベント 304
オープンイベント 305

ロードモジュールのイベント 305
モプロテクトイベント 305
マウントイベント 306
リンクイベント 306
Symlink イベント 307
Dup イベント 307
メモリマッピングイベント 308
ソケットイベント 308
イベントを接続 309
VM Readv イベントの処理 310
VM Writev イベントの処理 310
プロセストレース (Ptrace) イベント 311
バインドイベント 311
リッスンイベント 312
名前変更イベント 313
ユーザー ID (UID) 設定イベント 313
Chmod イベント 313
Amazon ECR リポジトリホスティング GuardDuty エージェント 314
EKS v1.16.0 - 1.8.1 (eks.build.2) の ECR リポジトリ 314
EKS バージョン 1.8.1.eks.build.1 の ECR リポジトリ 319
のセキュリティエージェントの ECR リポジトリ AWS Fargate (Amazon ECS のみ) 322
ECS-EC2 Bottlerocket のセキュリティエージェントの ECR リポジトリ 326
同じホスト上のセキュリティエージェント 330
概要 330
Impact 331
GuardDuty が複数のエージェントを処理する方法 331
EKS Runtime Monitoring 332
マルチアカウント環境の EKS Runtime Monitoring の設定 (API) 332
スタンドアロンアカウントの EKS Runtime Monitoring の設定 (API) 374
EKS Runtime Monitoring から Runtime Monitoring への移行 380
GuardDuty セキュリティエージェントのリリースバージョン 384
その他のリソース - 次のステップ 426
無効化、アンインストール、リソースクリーンアップ 427
Amazon EC2 リソースのセキュリティエージェントの手動アンインストール 429
ECS-EC2 Bottlerocket リソースのセキュリティエージェントのアンインストール 431
セキュリティエージェントのリソースのクリーンアップ 431

Malware Protection for EC2 434
GuardDuty 実行型マルウェアスキャンとオンデマンドマルウェアスキャンの比較 435
GuardDuty が EBS ボリュームをスキャンしてマルウェアを検出する方法 437
サポートされている EBS ボリューム 439
デフォルト KMS キー ID を変更する 440
スナップショット保持と EC2 スキャンカバレッジを設定する 441
スナップショットの保持 441
ユーザー定義タグ付きのスキャンオプション 443
グローバル GuardDutyExcluded タグ 447
GuardDuty 実行型マルウェアスキャン 447
30 日間の無料トライアル 448
マルチアカウント環境での GuardDuty 実行型マルウェアスキャンの有効化 449
スタンドアロンアカウントでの GuardDuty 実行型マルウェアスキャンの有効化 460
GuardDuty 実行型マルウェアスキャンを起動する検出結果 461
オンデマンドのマルウェアスキャン 463
オンデマンドのマルウェアスキャンの仕組み 464
オンデマンドマルウェアスキャンの開始 465
以前にスキャンした Amazon EC2 インスタンスの再スキャン 468
スキャンステータスと結果の監視 468
GuardDuty サービスアカウント 471
Malware Protection for EC2 のクォータ 474
S3 向けのマルウェア防御 479
料金と使用コスト 481
使用コストの確認 482
仕組み 483
概要 483
IAM ロールのアクセス許可 483
必要に応じてスキャン結果に基づいてオブジェクトにタグを追加する 483
バケットに対して Malware Protection for S3 を有効にした後のプロセス 484
Malware Protection for S3 の機能 486
(オプション) Malware Protection for S3 のみの使用を開始する (コンソール) 488
バケットの Malware Protection for S3 の設定 489
バケットの Malware Protection for S3 脅威検出の有効化 490
IAM ロールのアクセス許可 496
IAM ロールのアクセス許可エラーのトラブルシューティング 502
Malware Protection for S3 を有効にした後の手順 503

オンデマンド S3 マルウェアスキャン 504
前提条件 504
オンデマンドマルウェアスキャンを開始する 505
タグベースのアクセスコントロール (TBAC) の使用 505
S3 バケットリソースへの TBAC の追加 506
保護されたバケットのステータスの表示と理解 508
S3 オブジェクトスキャンのモニタリング 510
取りうる S3 オブジェクトスキャンステータスと結果ステータス 510
Amazon EventBridge の使用 513
S3 オブジェクトタグの使用 523
CloudWatch のアラームとメトリクスの使用 525
トラブルシューティング 528
Malware Protection プランのステータスのトラブルシューティング 528
オンデマンドマルウェアスキャンのトラブルシューティング 532
保護されたバケットの Malware Protection プランの編集 532
保護されたバケットの Malware Protection for S3 の無効化 534
Amazon S3 機能のサポート状況 535
Malware Protection for S3 クォータ 549
Malware Protection for AWS Backup 552
概要 552
仕組み 555
概要 552
スキャンに必要な IAM ロール 557
リソーススキャンのステータスと結果の確認 557
生成された検出結果の確認 557
IAM ロールのアクセス許可 558
マルウェアスキャンのために提供される顧客ロール 558
アクセス許可の詳細 558
ロールの保護 559
GuardDuty Malware Protection が KMS AWS で許可を使用する方法 560
GuardDuty Malware Protection 暗号化コンテキスト 560
ロールのアクセス許可と信頼ポリシー 561
個別に開始する 565
Malware Protection for Backup の使用を開始する手順 565
オンデマンドスキャンの開始 566
コンソール 566

API/CLI 567
スキャンステータスと結果の監視 567
Malware Protection for Backup クォータ 578
RDS Protection 582
サポートされているデータベース 584
RDS ログインアクティビティ 585
RDS データアクティビティ 586
マルチアカウント環境での RDS Protection の有効化 586
スタンドアロンアカウントの RDS Protection の設定 595
RDS Protection のモニタリングに関する問題のトラブルシューティング 596
RDS ストレージがいっぱいです 596
RDS for PostgreSQL のプライマリデータベースでサポートされていないバージョン 597
セキュリティに関するその他の考慮事項 597
Lambda Protection 598
Lambda Network Activity Monitoring 599
マルチアカウント環境での Lambda Protection の有効化 599
スタンドアロンアカウントに対する Lambda Protection の有効化 606
調査 - プレビュー 608
分析タイプ 608
クロスリージョン推論 609
前提条件 610
管理者アカウントとメンバーアカウントのアクセスモデル 611
調査の作成 612
調査結果の表示 614
調査結果の解釈 616
調査の一覧表示 617
カスタム検出ルール 620
カスタム検出ルールの仕組み 621
データソースとシグナル 621
モード 622
結果の集約 622
編集されたイベントフィールド 623
使用可能なカスタム検出ルール 624
ルール検出ロジックの表示 624
検出ロジックで使用される関数 626
カスタム検出ルールの管理 627

使用可能なルールの一覧表示 627
ルールのステータスの表示 628
一括操作 629
ドライラン 629
ライブ 630
ルールの無効化 631
マルチアカウント環境でのカスタム検出ルールの管理 632
組織設定 632
同じルールで両方のモードを実行する 634
例: すべてのメンバーアカウントにドライランモードでルールを適用する 634
組織全体にすべてのルールを適用する 634
複数のアカウントの管理 636
管理者アカウントとメンバーアカウントの関係 637
メンバー関係のステータス 641
AWS Organizationsを使用したアカウントの管理 642
考慮事項とレコメンデーション 643
委任 GuardDuty 管理者アカウントの指定に必要な許可 645
委任 GuardDuty 管理者アカウントの指定 647
組織の自動有効化の詳細設定の指定 648
組織へのメンバーの追加 652
(オプション) 既存のメンバーアカウントの保護プランの有効化 654
GuardDuty 内でのメンバーアカウントの継続的管理 655
メンバーアカウントの GuardDuty の停止 656
管理者アカウントからのメンバーアカウントの関連付け解除 (削除) 658
GuardDuty 組織からのメンバーアカウントの削除 659
委任 GuardDuty 管理者アカウントの変更 661
組織ポリシーを使用したアカウントの管理 663
前提条件 663
組織ポリシーの仕組み 664
コンソールでのポリシーの作成 667
ポリシーを適用する前にポリシーの影響を確認する 668
考慮事項と制限事項 669
招待によるアカウントの管理 671
招待によるアカウントの追加 672
管理者アカウントの単一の組織への統合 677
アカウントの [CSV のエクスポート] オプションの GuardDuty に関する考慮事項 680

検出結果タイプ 681
EC2 の検出結果タイプ 681
Backdoor:EC2/C&CActivity.B 683
Backdoor:EC2/C&CActivity.B!DNS 684
Backdoor:EC2/DenialOfService.Dns 685
Backdoor:EC2/DenialOfService.Tcp 686
Backdoor:EC2/DenialOfService.Udp 686
Backdoor:EC2/DenialOfService.UdpOnTcpPorts 687
Backdoor:EC2/DenialOfService.UnusualProtocol 688
Backdoor:EC2/Spambot 688
Behavior:EC2/NetworkPortUnusual 689
Behavior:EC2/TrafficVolumeUnusual 689
CryptoCurrency:EC2/BitcoinTool.B 690
CryptoCurrency:EC2/BitcoinTool.B!DNS 691
DefenseEvasion:EC2/UnusualDNSResolver 691
DefenseEvasion:EC2/UnusualDoHActivity 692
DefenseEvasion:EC2/UnusualDoTActivity 692
Impact:EC2/AbusedDomainRequest.Reputation 693
Impact:EC2/BitcoinDomainRequest.Reputation 694
Impact:EC2/MaliciousDomainRequest.Reputation 695
Impact:EC2/MaliciousDomainRequest.Custom 695
Impact:EC2/PortSweep 696
Impact:EC2/SuspiciousDomainRequest.Reputation 696
Impact:EC2/WinRMBruteForce 697
Recon:EC2/PortProbeEMRUnprotectedPort 698
Recon:EC2/PortProbeUnprotectedPort 698
Recon:EC2/Portscan 699
Trojan:EC2/BlackholeTraffic 700
Trojan:EC2/BlackholeTraffic!DNS 701
Trojan:EC2/DGADomainRequest.B 701
Trojan:EC2/DGADomainRequest.C!DNS 702
Trojan:EC2/DNSDataExfiltration 703
Trojan:EC2/DriveBySourceTraffic!DNS 704
Trojan:EC2/DropPoint 704
Trojan:EC2/DropPoint!DNS 705
Trojan:EC2/PhishingDomainRequest!DNS 705

UnauthorizedAccess:EC2/MaliciousIPCaller.Custom 706
UnauthorizedAccess:EC2/MetadataDNSRebind 706
UnauthorizedAccess:EC2/RDPBruteForce 707
UnauthorizedAccess:EC2/SSHBruteForce 708
UnauthorizedAccess:EC2/TorClient 709
UnauthorizedAccess:EC2/TorRelay 710
IAM の検出結果タイプ 710
CredentialAccess:IAMUser/AnomalousBehavior 712
CredentialAccess:IAMUser/CompromisedCredentials 712
DefenseEvasion:IAMUser/AnomalousBehavior 713
DefenseEvasion:IAMUser/BedrockLoggingDisabled 714
Discovery:IAMUser/AnomalousBehavior 714
Exfiltration:IAMUser/AnomalousBehavior 715
Impact:IAMUser/AnomalousBehavior 716
InitialAccess:IAMUser/AnomalousBehavior 716
PenTest:IAMUser/KaliLinux 717
PenTest:IAMUser/ParrotLinux 717
PenTest:IAMUser/PentooLinux 718
Persistence:IAMUser/AnomalousBehavior 718
Policy:IAMUser/RootCredentialUsage 719
Policy:IAMUser/ShortTermRootCredentialUsage 720
PrivilegeEscalation:IAMUser/AnomalousBehavior 721
Recon:IAMUser/MaliciousIPCaller 721
Recon:IAMUser/MaliciousIPCaller.Custom 722
Recon:IAMUser/TorIPCaller 722
Stealth:IAMUser/CloudTrailLoggingDisabled 723
Stealth:IAMUser/PasswordPolicyChange 723
UnauthorizedAccess:IAMUser/ConsoleLoginSuccess.B 724
UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.InsideAWS 724
UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.OutsideAWS 726
UnauthorizedAccess:IAMUser/MaliciousIPCaller 728
UnauthorizedAccess:IAMUser/MaliciousIPCaller.Custom 728
UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.InsideAWS 729
UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.OutsideAWS 731
UnauthorizedAccess:IAMUser/TorIPCaller 732
攻撃シーケンスの検出結果タイプ 732

AttackSequence:EKS/CompromisedCluster ... 733
AttackSequence:ECS/CompromisedCluster ... 734
AttackSequence:EC2/CompromisedInstanceGroup ... 735
AttackSequence:IAM/CompromisedCredentials ... 736
AttackSequence:S3/CompromisedData ... 737
カスタム検出ルールの検出結果タイプ ... 738
AWS CloudTrail 管理イベントのカスタム検出ルール ... 738
ランタイムモニタリングのカスタム検出ルール ... 743
S3 Protection の検出結果タイプ ... 748
Discovery:S3/AnomalousBehavior ... 749
Discovery:S3/MaliciousIPCaller ... 750
Discovery:S3/MaliciousIPCaller.Custom ... 751
Discovery:S3/TorIPCaller ... 751
Exfiltration:S3/AnomalousBehavior ... 752
Exfiltration:S3/MaliciousIPCaller ... 752
Impact:S3/AnomalousBehavior.Delete ... 753
Impact:S3/AnomalousBehavior.Permission ... 754
Impact:S3/AnomalousBehavior.Write ... 755
Impact:S3/MaliciousIPCaller ... 755
PenTest:S3/KaliLinux ... 756
PenTest:S3/ParrotLinux ... 756
PenTest:S3/PentooLinux ... 757
Policy:S3/AccountBlockPublicAccessDisabled ... 757
Policy:S3/BucketAnonymousAccessGranted ... 758
Policy:S3/BucketBlockPublicAccessDisabled ... 759
Policy:S3/BucketPublicAccessGranted ... 760
Stealth:S3/ServerAccessLoggingDisabled ... 760
UnauthorizedAccess:S3/MaliciousIPCaller.Custom ... 761
UnauthorizedAccess:S3/TorIPCaller ... 762
EKS Protection の検出結果タイプ ... 762
CredentialAccess:Kubernetes/MaliciousIPCaller ... 764
CredentialAccess:Kubernetes/MaliciousIPCaller.Custom ... 765
CredentialAccess:Kubernetes/SuccessfulAnonymousAccess ... 765
CredentialAccess:Kubernetes/TorIPCaller ... 766
DefenseEvasion:Kubernetes/MaliciousIPCaller ... 767
DefenseEvasion:Kubernetes/MaliciousIPCaller.Custom ... 767

DefenseEvasion:Kubernetes/SuccessfulAnonymousAccess 768
DefenseEvasion:Kubernetes/TorIPCaller 769
Discovery:Kubernetes/MaliciousIPCaller 769
Discovery:Kubernetes/MaliciousIPCaller.Custom 770
Discovery:Kubernetes/SuccessfulAnonymousAccess 771
Discovery:Kubernetes/TorIPCaller 772
Execution:Kubernetes/ExecInKubeSystemPod 772
Impact:Kubernetes/MaliciousIPCaller 773
Impact:Kubernetes/MaliciousIPCaller.Custom 773
Impact:Kubernetes/SuccessfulAnonymousAccess 774
Impact:Kubernetes/TorIPCaller 775
Persistence:Kubernetes/MaliciousIPCaller 775
Persistence:Kubernetes/MaliciousIPCaller.Custom 776
Persistence:Kubernetes/SuccessfulAnonymousAccess 777
Persistence:Kubernetes/TorIPCaller 777
Policy:Kubernetes/AdminAccessToDefaultServiceAccount 778
Policy:Kubernetes/AnonymousAccessGranted 779
Policy:Kubernetes/ExposedDashboard 779
Policy:Kubernetes/KubeflowDashboardExposed 780
CredentialAccess:Kubernetes/AnomalousBehavior.SecretsAccessed 780
PrivilegeEscalation:Kubernetes/AnomalousBehavior.RoleBindingCreated 781
Execution:Kubernetes/AnomalousBehavior.ExecInPod 782
PrivilegeEscalation:Kubernetes/AnomalousBehavior.WorkloadDeployed!PrivilegedContainer 783
Persistence:Kubernetes/AnomalousBehavior.WorkloadDeployed!ContainerWithSensitiveMount 784
Execution:Kubernetes/AnomalousBehavior.WorkloadDeployed 785
PrivilegeEscalation:Kubernetes/AnomalousBehavior.RoleCreated 786
Discovery:Kubernetes/AnomalousBehavior.PermissionChecked 787
Runtime Monitoring の検出結果タイプ 788
Runtime Monitoring がデフォルトで生成する検出結果タイプ 789
カスタム検出ルールを必要とする検出タイプ 826
Malware Protection for EC2 の検出結果のタイプ 828
Execution:EC2/MaliciousFile 829
Execution:ECS/MaliciousFile 830
Execution:Kubernetes/MaliciousFile 830

Execution:Container/MaliciousFile 831
Execution:EC2/SuspiciousFile 831
Execution:ECS/SuspiciousFile 832
Execution:Kubernetes/SuspiciousFile 832
Execution:Container/SuspiciousFile 833
Malware Protection for S3 の検出結果タイプ 834
Object:S3/MaliciousFile 834
Malware Protection for Backup の検出結果タイプ 835
Execution:EC2/MaliciousFile!Snapshot 835
Execution:EC2/MaliciousFile!AMI 836
Execution:EC2/MaliciousFile!RecoveryPoint 836
Execution:S3/MaliciousFile!RecoveryPoint 837
RDS Protection の検出結果タイプ 837
CredentialAccess:RDS/AnomalousBehavior.SuccessfulLogin 838
CredentialAccess:RDS/AnomalousBehavior.FailedLogin 839
CredentialAccess:RDS/AnomalousBehavior.SuccessfulBruteForce 840
CredentialAccess:RDS/MaliciousIPCaller.SuccessfulLogin 841
CredentialAccess:RDS/MaliciousIPCaller.FailedLogin 841
Discovery:RDS/MaliciousIPCaller 842
CredentialAccess:RDS/TorIPCaller.SuccessfulLogin 842
CredentialAccess:RDS/TorIPCaller.FailedLogin 843
Discovery:RDS/TorIPCaller 844
Exfiltration:RDS/AnomalousBehavior 844
Impact:RDS/AnomalousBehavior.Drop 846
Impact:RDS/AnomalousBehavior.Delete 847
Exfiltration:RDS/MaliciousIPCaller 848
Exfiltration:RDS/TorIPCaller 849
Impact:RDS/MaliciousIPCaller 850
Impact:RDS/TorIPCaller 850
Lambda Protection の検出結果タイプ 851
Backdoor:Lambda/C&CActivity.B 852
CryptoCurrency:Lambda/BitcoinTool.B 852
Trojan:Lambda/BlackholeTraffic 853
Trojan:Lambda/DropPoint 854
UnauthorizedAccess:Lambda/MaliciousIPCaller.Custom 854
UnauthorizedAccess:Lambda/TorClient 855

UnauthorizedAccess:Lambda/TorRelay 855
AI Protection の検出結果タイプ 856
Impact:IAMUser/AnomalousModelInvocation 856
Impact:IAMUser/CostHarvesting 858
Impact:IAMUser/PromptInjection.Direct 859
廃止された検出結果タイプ 862
PrivilegeEscalation:Kubernetes/PrivilegedContainer 863
Persistence:Kubernetes/ContainerWithSensitiveMount 863
Exfiltration:S3/ObjectRead.Unusual 864
Impact:S3/PermissionsModification.Unusual 865
Impact:S3/ObjectDelete.Unusual 865
Discovery:S3/BucketEnumeration.Unusual 866
Persistence:IAMUser/NetworkPermissions 867
Persistence:IAMUser/ResourcePermissions 868
Persistence:IAMUser/UserPermissions 868
PrivilegeEscalation:IAMUser/AdministrativePermissions 869
Recon:IAMUser/NetworkPermissions 870
Recon:IAMUser/ResourcePermissions 871
Recon:IAMUser/UserPermissions 872
ResourceConsumption:IAMUser/ComputeResources 872
Stealth:IAMUser/LoggingConfigurationModified 873
UnauthorizedAccess:IAMUser/ConsoleLogin 874
UnauthorizedAccess:EC2/TorIPCaller 875
Backdoor:EC2/XORDDOS 875
Behavior:IAMUser/InstanceLaunchUnusual 876
CryptoCurrency:EC2/BitcoinTool.A 876
UnauthorizedAccess:IAMUser/UnusualASNCaller 876
影響を受ける可能性のあるリソース別の GuardDuty 検出結果タイプ 877
GuardDuty のアクティブな検出結果タイプ 877
検出結果の理解と生成 903
GuardDuty の検出結果の形式 904
脅威の目的 905
GuardDuty マルウェア検出のスキャンエンジン 908
サンプルの検出結果 909
GuardDuty の コンソールや API で使用したサンプル検出結果の生成 909
GuardDuty の検出結果のテスト 911

考慮事項 911
テスタースクリプトが生成できる GuardDuty の検出結果 913
ステップ 1 - 前提条件 915
ステップ 2 - AWS リソースをデプロイする 916
ステップ 3 - テスタースクリプトを実行する 917
ステップ 4 - AWS テストリソースをクリーンアップする 920
一般的な問題のトラブルシューティング 920
GuardDuty コンソールの検出結果ページ 922
検出結果ページの操作 923
検出結果の重大度レベル 924
重大度重大 925
重大度高 925
重大度中 925
重大度低 926
検出結果の詳細 926
検出結果の概要 927
[リソース] 928
攻撃シーケンスの検出結果の詳細 935
カスタム検出ルールの検出結果の詳細 943
RDS データベース (DB) ユーザーの詳細 945
Runtime Monitoring の検出結果の詳細 945
EBS ボリュームのスキャンの詳細 948
Malware Protection for EC2 の検出結果の詳細 948
Malware Protection for S3 の検出結果の詳細 949
アクション 950
アクターまたはターゲット 952
位置情報の詳細 953
追加情報 953
証拠 954
異常な動作 954
GuardDuty の検出結果の集約 960
GuardDuty の検出結果の管理 961
GuardDuty の [要約] ダッシュボード 962
概要 963
検出結果 964
最も一般的な検出結果タイプ 965

重大度別の検出結果 965
検出結果が最も多いアカウント 965
検出結果を含むリソース 965
最も少ない検出結果 966
保護プランカバレッジ 966
GuardDuty の検出結果のフィルタリング 967
GuardDuty コンソールでのフィルターの作成と保存 968
GuardDuty API と CLI を使用したフィルターセットの作成と保存 970
ライフサイクルメタデータ 972
GuardDuty のプロパティフィルター 973
抑制ルール 1007
Extended Threat Detection での抑制ルールの使用 1007
抑制ルールの一般的ユースケースとその例 1008
抑制ルールを作成する 1011
抑制ルールの更新 1016
抑制ルールを削除する 1018
抑制ルールメトリクス 1019
エンティティリストおよび IP アドレスリスト 1021
エンティティリストと IP アドレスリストについて理解する 1021
GuardDuty リストに関する重要な考慮事項 1022
リストフォーマット 1023
リストのステータスについて理解する 1033
エンティティリストと IP アドレスリストの前提条件の設定 1033
エンティティリストまたは IP リストの追加とアクティブ化 1036
エンティティリストまたは IP アドレスリストの更新 1042
エンティティリストまたは IP アドレスリストの非アクティブ化 1047
エンティティリストまたは IP アドレスリストの削除 1049
生成された検出結果を Amazon S3 にエクスポートする 1052
考慮事項 1053
ステップ 1 – 検出結果のエクスポートに必要な許可 1054
ステップ 2 – KMS キーにポリシーをアタッチする 1054
ステップ 3: Amazon S3 バケットにポリシーをアタッチする 1056
ステップ 4 - S3 バケットに検出結果をエクスポートする (コンソール) 1060
ステップ 5 – 検出結果をエクスポートする頻度 1062
EventBridge を使用した検出結果の処理 1062
GuardDuty の EventBridge 通知頻度 1063

Amazon SNS トピックおよびエンドポイントの設定 1064
GuardDuty での EventBridge の使用 1066
EventBridge ルールの作成 1067
マルチアカウント環境向け EventBridge ルール 1075
CloudWatch Logs を監査する方法と、リソースがスキップされる理由について 1076
GuardDuty Malware Protection for EC2 の CloudWatch Logs の監査 1076
GuardDuty Malware Protectionfor EC2 ログの保持 1078
リソースをスキップする理由 1078
EC2 マルウェアスキャン結果における誤検出の報告 1085
偽陽性の S3 オブジェクトスキャンの結果の報告 1086
Backup マルウェアスキャン結果の誤検出の報告 1087
検出結果の修復 1088
侵害された可能性のある Amazon EC2 インスタンスの修復 1088
侵害された可能性のある S3 バケットの修復 1090
特定の S3 バケットへのアクセスニーズに基づく推奨事項 1092
悪意のある可能性がある S3 オブジェクトの修復 1093
侵害された可能性のある EBS スナップショットの修復 1093
侵害された可能性のある EC2 AMI の修復 1094
侵害された可能性のある EC2 復旧ポイントの修復 1095
侵害された可能性のある S3 復旧ポイントの修復 1096
侵害された可能性のある ECS クラスターの修復 1097
侵害された可能性のある AWS 認証情報の修正 1098
侵害された可能性のあるスタンドアロンコンテナの修復 1099
EKS Protection の検出結果の修復 1100
潜在的な設定の問題 1101
侵害された可能性のある Kubernetes ユーザーの修復 1102
侵害された可能性のある Kubernetes ポッドの修復 1105
侵害された可能性のあるコンテナイメージの修復 1106
侵害された可能性のある Kubernetes ノードの修復 1107
Runtime Monitoring 検出結果の修正 1107
侵害されたコンテナイメージの修復 1109
侵害された可能性のあるデータベースの修復 1110
ログインイベントの成功により、侵害された可能性のあるデータベースの修復 1111
ログインイベントの失敗により、侵害された可能性のあるデータベースの修正 1111
異常な流出アクティビティによる侵害された可能性のあるデータベースの修復 1113
異常な破壊的アクティビティによる侵害された可能性のあるデータベースの修復 1114

漏えいした可能性のある認証情報の修正 1116
ネットワークアクセスを制限する 1116
侵害された可能性のある Lambda 関数の修復 1117
使用状況のモニタリングとコストの見積もり 1118
Amazon CloudWatch 使用状況メトリクス 1118
メトリクスの詳細 1118
S3 向けのマルウェア防御 1121
GuardDuty の使用について 1122
GuardDuty イベント処理 1122
GuardDuty Runtime Monitoring によってモニタリングされるインスタンスの GuardDuty VPC フローログ処理料金 1122
GuardDuty コストの見積もり 1123
AWS 料金計算ツールの使用 1123
Security Hub のお客様 1124
API の保護プランの機能名 1125
データソースから機能への変更 1125
GuardDuty API の変更 1125
データソースと比較した機能 1126
機能を使用する API の仕組みを理解する 1126
API への機能の変更の組み込み 1127
マッピングされた GuardDuty 機能 1128
セキュリティ 1130
データ保護 1131
保管中の暗号化 1132
転送中の暗号化 1132
サービス改善のためのデータ使用をオプトアウトする 1132
CloudTrail による口グ記録 1134
CloudTrail での GuardDuty 情報 1134
CloudTrail の GuardDuty コントロールプレーンイベント 1135
CloudTrail の GuardDuty データイベント 1135
例: GuardDuty ログアーカイブエントリ 1136
Identity and Access Management 1139
オーディエンス 1140
アイデンティティを使用した認証 1140
ポリシーを使用したアクセスの管理 1141
Amazon GuardDuty が IAM で機能する方法 1143

アイデンティティベースのポリシーの例 1149
サービスにリンクされたロールの使用 1159
AWS 管理ポリシー 1169
トラブルシューティング 1187
コンプライアンス検証 1189
耐障害性 1190
インフラストラクチャセキュリティ 1190
VPC エンドポイント (AWS PrivateLink) 1191
GuardDuty VPC エンドポイントに関する考慮事項 1191
GuardDuty 用のインターフェイス VPC エンドポイントの作成 1191
GuardDuty 用の VPC エンドポイントポリシーの作成 1192
共有サブネット 1193
AWS セキュリティサービスとの統合 1194
GuardDuty と の統合 AWS Security Hub CSPM 1194
GuardDuty と Amazon Detective の統合 1194
AWS Security Hub CSPM 統合 1194
Amazon GuardDuty が検出結果を に送信する方法 AWS Security Hub CSPM 1195
での GuardDuty の検出結果の表示 AWS Security Hub CSPM 1196
統合の有効化と構成 1217
Security Hub CSPM での GuardDuty コントロールの使用 1217
検出結果の Security Hub CSPM への公開の停止 1217
Amazon Detective の統合 1217
統合の有効化 1218
GuardDuty の検出結果から Amazon Detective へのピボット 1218
GuardDuty マルチアカウント環境との統合を使用します。 1219
停止または無効化 1220
GuardDuty のお知らせ 1222
Amazon SNS メッセージ形式 1228
GuardDuty クォータ 1234
マネージドインスタンスのサポート 1241
トラブルシューティング 1242
Amazon S3 への検出結果のエクスポート - アクセスエラー 1242
Malware Protection for EC2 の問題 1243
GuardDuty 実行型マルウェアスキャンを有効にするときに必要な AWS Organizations 管理アクセス許可がない 1243

オンデマンドのマルウェアスキャンを開始しようとしていますが、必要な許可がないというエラーが表示されます。 1243
Malware Protection for EC2 を使用しているときに iam:GetRole エラーが表示されます。 1243
私は、GuardDuty 実行型マルウェアスキャンを有効にする必要がある GuardDuty 管理者アカウントですが、GuardDuty の管理に AWS マネージドポリシー AmazonGuardDutyFullAccess を使用しません。 1244
Runtime Monitoring の問題 1244
ランタイムカバレッジの問題 1244
メモリ不足の問題のトラブルシューティング 1244
AWS Step Functions ワークフローが予期せず失敗している 1245
その他の問題のトラブルシューティング 1245
リージョンとエンドポイント 1247
リージョン固有機能の可用性 1247
従来のアクションとパラメータ 1250
ドキュメント履歴 1252
以前の更新 1360
..... mccclxi

Amazon GuardDuty とは

Amazon GuardDuty は、 AWS 環境内の AWS データソースとログを継続的にモニタリング、分析、処理する脅威検出サービスです。GuardDuty は、悪意のある IP アドレスとドメインのリスト、ファイルハッシュ、機械学習 (ML) モデルなどの脅威インテリジェンスフィードを使用して、AWS 環境内の疑わしいアクティビティや悪意のある可能性のあるアクティビティを特定します。次に、GuardDuty で検出可能な脅威シナリオの概要を示します。

- 漏洩した AWS 認証情報。
- データを引き出し破棄することでランサムウェアイベントを引き起こす。Amazon Aurora と Amazon RDS データベースに対応したエンジンバージョンで、ログインイベントの異常なパターンが異常な動作を示唆する。
- Amazon Elastic Compute Cloud (Amazon EC2) インスタンスとコンテナワークロードに不正なクリプトマイニングアクティビティが見られる。
- Amazon EC2 インスタンスとコンテナワークロードにマルウェアが存在し、Amazon Simple Storage Service (Amazon S3) バケットにファイルが新規にアップロードされる。
- オペレーティングシステムレベルイベント、ネットワークイベント、ファイルイベントにより、Amazon Elastic Kubernetes Service (Amazon EKS) クラスター、Amazon Elastic Container Service (Amazon ECS)、 AWS Fargate タスク、Amazon EC2 インスタンス、コンテナワークロードでの不正な動作が示唆される。

次の動画では、GuardDuty で AWS 環境内の脅威をどのように検出できるのかを簡単に説明しています。

Amazon GuardDuty とは

内容

- GuardDuty の機能
- PCI DSS コンプライアンス
- GuardDuty の料金
- GuardDuty へのアクセス

GuardDuty の機能

Amazon GuardDuty が AWS 環境内の潜在的な脅威のモニタリング、検出、管理に役立つ主な方法をいくつか紹介します。

特定のデータソースとイベントログを継続してモニタリングする

- 基本的な脅威の検出 – で GuardDuty を有効にすると AWS アカウント、GuardDuty はそのアカウントに関連付けられた基本的なデータソースの取り込みを自動的に開始します。こうしたデータソースには、 AWS CloudTrail 管理イベント、VPC フローログ (Amazon EC2 インスタンスから生成)、DNS ログなどがあります。GuardDuty がこれらのデータソースの分析と処理

を開始して関連付けられたセキュリティ検出結果を生成するのに、他のものを有効にする必要はありません。詳細については、「GuardDuty 基本データソース」を参照してください。

- 拡張脅威検出 – この機能は、 内の基本的なデータソース、複数のタイプの AWS リソース、および時間にわたるマルチステージ攻撃を検出します AWS アカウント。アカウントには、個々に見れば脅威として現れないイベントが複数存在する場合があります。ただし、これらのイベントが疑わしいアクティビティを示すシーケンスで観察された場合、GuardDuty はそれを攻撃シーケンスとして識別します。GuardDuty は、関連する攻撃シーケンスの検出結果タイプを生成して通知し、観測された攻撃シーケンスに関する詳細を提供します。

  追加コストなしで、拡張脅威検出は GuardDuty を有効にする AWS アカウント と、それぞれに対して自動的に有効になります。この機能では、ユースケースに特化した保護プランを有効にする必要はありません。ただし、Amazon S3 リソースに対するセキュリティの幅を広げるために、GuardDuty ではアカウントで S3 Protection を有効にすることをお勧めします。これにより、Extended Threat Detection は、Amazon S3 リソースに影響を与える可能性のあるマルチステージ攻撃を特定するのに役立ちます。

  この機能の仕組みと対象となる脅威シナリオの詳細については、「GuardDuty Extended Threat Detection」を参照してください。

- 調査 (プレビュー) – GuardDuty 調査は、GuardDuty の検出結果の AI を活用したセキュリティ分析を提供します。調査を作成して特定の結果、アカウント、または組織全体を分析し、リスク評価、信頼スコア、MITRE ATT&CK® 分類、および実行可能な次のステップを含む構造化された調査概要を受け取ることができます。詳細については、「GuardDuty 調査 (プレビュー)」を参照してください。

- カスタム検出ルール – カスタム検出ルールは、Amazon GuardDuty の検出ルールの厳選されたライブラリであり、脅威アクターの手法と整合しています。各ルールは、 AWS CloudTrail 管理イベントまたは GuardDuty セキュリティエージェントからの Runtime Monitoring イベントのフィールドを評価することで、特定の手法を検出します。ルールを参照、有効化、管理するには、GuardDuty コンソールまたは GuardDuty API を使用します。詳細については、「GuardDuty のカスタム検出ルール」を参照してください。

- ユースケースに焦点を当てた GuardDuty 保護プラン – AWS 環境のセキュリティに対する脅威検出の可視性を強化するために、GuardDuty は有効にできる専用の保護プランを提供します。保護プランは、他の AWS サービスのログとイベントをモニタリングするのに役立ちます。これらのソースには、EKS 監査ログ、RDS ログインアクティビティ、CloudTrail の Amazon S3 データイベント、EBS ボリューム、Amazon EKS、Amazon EC2、Amazon ECS-Fargate 全体のランタイムモニタリング、Lambda ネットワークアクティビティログ、Amazon Bedrock、Amazon Bedrock AgentCore、Amazon SageMaker AI for AI Protection

AWS CloudTrail のデータイベントが含まれます。GuardDuty では、こうしたログとイベントソースをまとめて「機能」と呼んでいます。サポートされている で 1 つ以上の専用保護プラン AWS リージョン をいつでも有効にできます。どの保護プランを有効にしているかに基づいて、GuardDuty がアクティビティのモニタリング、処理、分析を開始します。各保護プランとその仕組みの詳細については、対応する保護プランのドキュメントを参照してください。

GuardDuty を初めて有効にすると、GuardDuty はアカウントの Runtime Monitoring を除くすべての保護プランを自動的に有効にし、30 日間の無料トライアルに含めます。保護プランはいつでも無効にできます。既存の GuardDuty のお客様の場合、GuardDuty を有効にした後に起動する保護プランは自動的に有効にならないため、有効にすることを選択できます。デフォルトで有効になっている保護プランについては、「」を参照してくださいGuardDuty の 30 日間無料トライアルの使用。

保護プラン	説明
AI 保護	異常なモデル呼び出しやコスト収集攻撃など、Amazon Bedrock、Amazon Bedrock AgentCore、Amazon SageMaker AI を使用する AI ワークロードに対する脅威を検出します。
S3 Protection	Amazon S3 バケット内のデータを引き出して破棄しようとするなど、潜在するセキュリティリスクを識別します。
EKS Protection	EKS 監査ログのモニタリングは、Amazon EKS クラスターで記録された Kubernetes 監査ログを分析して、疑わしいアクティビティや悪意のあるアクティビティがないかチェックします。
ランタイムモニタリング	Amazon EKS、Amazon EC2、Amazon ECS (AWS Fargate を含む) でオペレーティングシステムレベルのイベントをモニタリングし分析して、潜在するランタイム脅威がないか検出します。
Malware Protection for EC2	Amazon EC2 インスタンスに関連付けられている Amazon EBS ボリュームをスキャンして、潜在するマルウェアがないか検出します。この機能をオンデマンドで使用するオプションもあります。

保護プラン	説明
S3 向けのマルウェア防御	Amazon S3 バケット内に新規にアップロードされたオブジェクトに潜在するマルウェアがないか検出します。この機能をオンデマンドで使用するオプションもあります。
Malware Protection for AWS Backup	EBS スナップショット、EC2 AMI の 、バックアップリカバリポイントなどの AWS バックアップリソースをスキャンして、マルウェアの潜在的な存在を検出します。この機能をオンデマンドで使用するオプションもあります。
RDS Protection	RDS ログインアクティビティを分析しプロファイリングして、サポートされている Amazon Aurora と Amazon RDS データベースへのアクセス脅威が潜んでいないか確認します。
Lambda Protection	VPC フローログから Lambda ネットワークアクティビティログのモニタリングを始めて、 AWS Lambda 関数への脅威を検出します。こうした潜在する脅威の例として、クリプトマイニングや悪意あるサーバーとの通信などがあります。

> **Malware Protection for S3 を個別に有効にする**
>
> GuardDuty には、Amazon GuardDuty サービスを有効にすることなく Malware Protection for S3 を個別に使用できるという柔軟性があります。Malware Protection for S3 のみを開始する方法については、「GuardDuty Malware Protection for S3」を参照してください。これ以外の保護プランを使用する場合は、GuardDuty サービスを有効にする必要があります。

マルチアカウント環境を管理する

マルチアカウント AWS 環境を管理するには、 AWS Organizations (推奨) または従来の招待方法を使用します。詳細については、「複数のアカウントの管理」を参照してください。

検出された脅威に関するセキュリティ検出結果を生成する

GuardDuty は、 AWS リソースに関して潜在するセキュリティ脅威を検出すると、セキュリティ検出結果の生成を開始して侵害された可能性のあるリソースを記録します。アカウントで GuardDuty を有効にしたら、サンプルの検出結果を生成して関連付けられた検出結果の詳細を表

示します。セキュリティ検出結果の詳細なリストについては、「GuardDuty 検出結果タイプ」を参照してください。

GuardDuty では、テススクリプトを使用して特定の GuardDuty セキュリティ検出結果を生成することもできます。これにより、GuardDuty の検出結果を確認して対応する方法を理解できます。詳細については、「専用アカウントでの GuardDuty の検出結果のテスト」を参照してください。

セキュリティ検出結果の評価および管理

GuardDuty は、複数のアカウントにわたるセキュリティ検出結果をまとめて、結果を GuardDuty コンソールの [概要] ダッシュボードに表示します。API、 AWS Security Hub CSPM AWS Command Line Interface、または AWS SDK を使用して検出結果を取得することもできます。現在のセキュリティ状態を全体的に把握することで、傾向と潜在する問題を識別し、必要な修復手順を実行できます。詳細については、「GuardDuty の検出結果の管理」を参照してください。

関連する AWS セキュリティサービスとの統合

AWS 環境のセキュリティ傾向の分析と調査をさらに支援するには、以下の AWS セキュリティ関連サービスを GuardDuty と組み合わせて使用することを検討してください。

- AWS Security Hub CSPM – このサービスは、リソースのセキュリティ状態を包括的に AWS 把握し、セキュリティ業界標準とベストプラクティスに照らして AWS 環境をチェックするのに役立ちます。これは、複数の AWS サービス (Amazon Macie を含む) およびサポートされている AWS パートナーネットワーク (APN) 製品からセキュリティ検出結果を消費、集約、整理、優先順位付けすることによって部分的に行われます。Security Hub CSPM は、セキュリティの傾向を分析し、 AWS 環境全体で最も優先度の高いセキュリティ問題を特定するのに役立ちます。

  GuardDuty と Security Hub CSPM を一緒に使用する方法については、「」を参照してくださいGuardDuty と の統合 AWS Security Hub CSPM。Security Hub CSPM の詳細については、AWS Security Hub 「 ユーザーガイド」を参照してください。

- Amazon Detective - このサービスを使用すると、セキュリティに関する検出結果や疑わしいアクティビティを分析し、調査して根本原因を迅速に特定できます。Detective は、 AWS リソースからログデータを自動的に収集します。その後、機械学習、統計分析、グラフ理論を使用して、セキュリティ調査をより迅速かつ効率的に行うのに役立つビジュアライゼーションを生成します。Detective の事前に作成されたデータの集計、要約、およびコンテキストは、考えられるセキュリティ問題の性質と範囲を分析して特定するのに役立ちます。

GuardDuty と Detective を共に使用する方法については、「GuardDuty と Amazon Detective の統合」を参照してください。Detective の詳細については、「Amazon Detective ユーザーガイド」を参照してください。

- Amazon EventBridge - このサービスを使用すると、通知を受け取って GuardDuty のセキュリティ検出結果にほぼリアルタイムに対応できます。検出結果に変化があると、GuardDuty はイベントを作成します。EventBridge から通知を受け取る頻度を選択できます。詳細については、「Amazon EventBridge ユーザーガイド」の「Amazon EventBridge とは」を参照してください。

PCI DSS コンプライアンス

GuardDuty は、加盟店またはサービスプロバイダーによるクレジットカードデータの処理、ストレージ、および送信をサポートし、Payment Card Industry (PCI) データセキュリティスタンダード (DSS) に準拠していることが検証されています。PCI コンプライアンスパッケージのコピーをリクエストする方法など、 AWS PCI DSS の詳細については、「PCI DSS レベル 1」を参照してください。

詳細については、「AWS セキュリティブログ」の「New third-party test compares Amazon GuardDuty to network intrusion detection systems」を参照してください。

GuardDuty の料金

このセクションでは、GuardDuty がさまざまな保護プランに使用する AWS 無料利用枠 モデルと、推定および実際の使用コストを表示する方法について説明します。サポートされているリージョンのすべての保護プランに関連する料金の詳細については、「GuardDuty の料金」を参照してください。

AWS 無料利用枠

AWS 無料利用枠 は、各サービスで指定された制限まで AWS のサービス 無料で探索して試すのに役立ちます。12 か月間無料、常に無料、短期無料トライアルの 3 つのカテゴリがあります。Amazon GuardDuty は短期無料トライアルのカテゴリに属し、30 日間無料トライアルが用意されています。この無料トライアルの終了後も継続して GuardDuty を使用すると、このサービスを使用する方法に基づいてコストが発生します。

[1]GuardDuty の 30 日間の無料トライアルの例外

オンデマンドの (Malware Protection for EC2 の下での) マルウェアスキャンと Malware Protection for S3 は、GuardDuty の 30 日間短期無料トライアルのカテゴリに分類されません。Malware Protection for S3 には無料利用枠が含まれていますが、オンデマンドマルウェアスキャンはpay-as-you-useコストモデルに従います。オンデマンドマルウェアスキャンを使用した 30 日間の無料トライアルや無料利用枠のコストモデルはありません。

GuardDuty の 30 日間無料トライアルの使用

で初めて GuardDuty を使用する場合 AWS リージョン、 AWS アカウント はそのリージョンで 30 日間の無料トライアルに自動的に登録されます。一部の保護プランも自動的に有効になり、30 日間無料トライアルに含まれます。GuardDuty はリージョン別のサービスであるため、別のリージョンで初めて有効にすると、アカウントはそのリージョンで GuardDuty の 30 日間無料トライアルが利用可能になります。GuardDuty 組織で複数のアカウントを使用した場合は、アカウントごとに 30 日間無料トライアルに提供されます。

次の表を使用して、GuardDuty でデフォルトで有効になっている保護プランと、その無料トライアルの有無を確認してください。

保護プラン	GuardDuty ではデフォルトで有効化	無料トライアルの個別提供[2]
EKS Protection	はい	はい
S3 Protection	はい	はい
ランタイムモニタリング	いいえ	はい
Malware Protection for EC2 – GuardDuty 実行型マルウェアスキャン	はい	はい
Malware Protection for EC2 – GuardDuty	いいえ	いいえ[1]

保護プラン	GuardDuty ではデフォルトで有効化	無料トライアルの個別提供[2]	
でのオンデマンドマルウェアスキャン			
GuardDuty Malware Protection for S3	いいえ	いいえ[1]	
RDS Protection	はい	はい	
Lambda Protection	はい	はい	
Malware Protection for AWS Backup	いいえ	いいえ	

[2]GuardDuty を初めて有効にすると、保護プラン (Runtime Monitoring を除く) が自動的に有効になり、最初の 30 日間の無料トライアルに含まれます。既存の GuardDuty アカウントが、最初の GuardDuty 無料トライアルの有効期限が切れた後に新しい保護プランを有効にすると、その保護プランには独自の 30 日間の無料トライアルが付帯します。保護プランの無料トライアルの詳細については、各保護プランの該当するドキュメントを参照してください。

無料トライアルの期間中に推定使用コストを表示する - GuardDuty の 30 日間無料トライアルの期間中に、GuardDuty でアカウントの推定使用コスト (保護プランを使用していればそのコストも含む) を確認できます。委任 GuardDuty 管理者アカウントを利用している場合は、GuardDuty を有効にしているすべてのメンバーアカウントについて合計推定使用コストとアカウントレベルの内訳を確認できます。詳細については、「GuardDuty の使用状況のモニタリングとコストの見積もり」を参照してください。

無料トライアルの終了後の使用コスト - 無料トライアルの終了後に GuardDuty またはそのいずれかの保護プランを継続して使用すると、その分の使用コストが発生します。請求書を表示するには、https://console.aws.amazon.com/costmanagement/ コンソールで Cost Explorer に移動します。AWS アカウントの請求の詳細については、AWS Billing 「 ユーザーガイド」を参照してください。

無料利用枠での Malware Protection for S3 の使用

Malware Protection for S3 には、 に関連付けられた無料利用枠が含まれています AWS アカウント。詳細については、「Malware Protection for S3 の料金と使用コスト」を参照してください。

GuardDuty へのアクセス

Amazon GuardDuty はほとんどの で利用できます AWS リージョン。GuardDuty を現在使用できるリージョンのリストについては、「リージョンとエンドポイント」を参照してください。

GuardDuty は次のいずれかの方法で使用できます。

GuardDuty のコンソール

https://console.aws.amazon.com/guardduty/

コンソールは、GuardDuty にアクセスして使用するためのブラウザベースのインターフェイスです。GuardDuty コンソールでは、GuardDuty アカウント、データ、およびリソースにアクセスできます。

AWS Command Line Interface

AWS Command Line Interface (AWS CLI) を使用すると、システムのコマンドラインでコマンドを発行して、GuardDuty タスクと AWS タスクを実行できます。 AWS CLI コマンドは、タスクを実行するスクリプトを構築する場合に便利です。

のインストールと使用の詳細については AWS CLI、AWS Command Line Interface 「 ユーザーガイド」を参照してください。GuardDuty で使用できる AWS CLI コマンドを表示するには、AWS CLI 「 コマンドリファレンス」を参照してください。

GuardDuty HTTPS API

GuardDuty HTTPS API を使用して、GuardDuty および に AWS プログラムでアクセスできます。これにより、HTTPS リクエストを サービスに直接発行できます。詳細については、「Amazon GuardDuty API Reference」を参照してください。

AWS SDKs

AWS は、さまざまなプログラミング言語とプラットフォーム (Java、Python、Ruby、.NET、iOS、Android など) のライブラリとサンプルコードで構成されるソフトウェア開発キット (SDKs) を提供します。SDK は、GuardDuty へのアクセス権をプログラムによって作成するのに役立ちます。ダウンロードやインストールなどの方法を含む AWS SDK の詳細については、「Tools for Amazon Web Services」(Amazon Web Services 用のツール) を参照してください。

Amazon GuardDuty の概念と主要な用語

Amazon GuardDuty の使用を開始するにあたり、その概念と関連する主要な用語を確認しておくと利点があります。

アカウント

AWS リソースを含む標準の Amazon Web Services (AWS) アカウント。アカウント AWS で にサインインし、GuardDuty を有効にできます。

他のアカウントを招待して、GuardDuty を有効にし、GuardDuty の AWS アカウントに関連付けることもできます。招待が承諾されると、アカウントは GuardDuty アカウントの管理者アカウントとして指定され、これに追加されたアカウントはメンバーアカウントになります。これにより、代わりにそれらのアカウントの GuardDuty の検出結果を表示および管理することができます。

管理者アカウントのユーザーは、自分のアカウントおよびすべてのメンバーアカウントの GuardDuty の検出結果を表示および管理できるだけでなく、GuardDuty の設定もできます。管理者アカウントが管理できるメンバーアカウントの数については、「GuardDuty クォータ」を参照してください。

メンバーアカウントのユーザーは、GuardDuty マネジメントコンソールまたは GuardDuty API を使用して、自分のアカウントの GuardDuty の検出結果を表示および管理するだけでなく、GuardDuty の設定もできます。メンバーアカウントのユーザーは、他のメンバーアカウントの検出結果を表示または管理することはできません。

は、GuardDuty 管理者アカウントとメンバーアカウントを同時に AWS アカウント 使用することはできません。 AWS アカウント で承諾できるメンバーシップの招待は 1 つのみです。メンバーシップの招待の承諾はオプションです。

詳細については、「Amazon GuardDuty での複数のアカウントの管理」を参照してください。

攻撃シーケンス

攻撃シーケンスとは、GuardDuty が観測した複数のイベントが特定の順序で発生し、不審な活動のパターンに一致する相関関係です。GuardDuty は、その Extended Threat Detection 機能を使用して、アカウント内の基本的なデータソース、 AWS リソース、タイムラインにまたがるこれらのマルチステージ攻撃を検出します。

以下のリストでは、攻撃シーケンスに関連する主要な用語を簡単に説明します。

- インジケータ – 一連のイベントが疑わしいアクティビティと一致する理由に関する情報を提供します。
- シグナル – シグナルは、GuardDuty が観測した API アクティビティ、またはアカウントで既に検出された GuardDuty の検出結果です。アカウント内で特定の順序で観測されたイベントを相関分析することで、GuardDuty は攻撃シーケンスを特定します。

  アカウント内のイベントには、潜在的な脅威を示すものではないものがあります。GuardDuty はそれらを弱いシグナルと見なします。ただし、弱いシグナルと GuardDuty の検出結果が特定の順序で観測され、相関分析によって潜在的に不審な活動と一致する場合、GuardDuty は攻撃シーケンス検出を生成します。
- エンドポイント – 脅威アクターが攻撃シーケンスで使用した可能性のあるネットワークエンドポイントに関する情報。

ディテクター

Amazon GuardDuty はリージョンレベルのサービスです。特定の で GuardDuty を有効にすると AWS リージョン、 AWS アカウント はディテクター ID に関連付けられます。この英数字 32 文字の ID は、そのリージョンのアカウントに固有です。例えば、別のリージョンで同じアカウントに対して GuardDuty を有効にすると、アカウントは別のディテクター ID に関連付けられます。detectorId の形式は `12abc34d567e8fa901bc2d34e56789f0` です。

すべての GuardDuty の検出結果、アカウント、検出結果の管理に関するアクションおよび GuardDuty サービスは、ディテクター ID を使用して API オペレーションを実行します。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

> **Note**
>
> 複数アカウント環境では、メンバーアカウントのすべての検出結果が管理者アカウントのディテクターに関連付けられます。

CloudWatch Events の通知頻度の設定や、GuardDuty が処理するオプションの保護プランの有効化または無効化など、GuardDuty の一部の機能は、ディテクターを介して設定されます。

GuardDuty 内での Malware Protection for S3 の使用

GuardDuty が有効になっているアカウントで Malware Protection for S3 を有効にすると、保護されたリソースの有効化、編集、無効化などの Malware Protection for S3 アクションは、ディテクター ID に関連付けられません。

GuardDuty を有効にせず、Malware Protection for S3 の脅威検出オプションを選択した場合、アカウントにディテクター ID は作成されません。

基本データソース

データのセットのオリジンまたは場所です。 AWS 環境内の不正または予期しないアクティビティを検出するには。GuardDuty は、 AWS CloudTrail イベントログ、 AWS CloudTrail 管理イベント、S3 AWS CloudTrail のデータイベント、VPC フローログ、DNS ログのデータを分析して処理します。「」を参照してくださいGuardDuty 基本データソース。

機能

GuardDuty 保護プラン用に設定された機能オブジェクトは、 AWS 環境内の不正または予期しないアクティビティを検出するのに役立ちます。各 GuardDuty 保護プランでは、データを分析および処理するために対応する機能オブジェクトを設定します。機能オブジェクトには、EKS 監査ログ、RDS ログインアクティビティモニタリング、Lambda ネットワークアクティビティログ、EBS ボリュームなどがあります。詳細については、「GuardDuty API の保護プランの機能名」を参照してください。

結果

GuardDuty によって発見された潜在的なセキュリティの問題。詳細については、「Amazon GuardDuty の検出結果の理解と生成」を参照してください。

検出結果は、セキュリティ問題に関する詳細な説明と合わせて、GuardDuty コンソールに表示されます。GetFindings および ListFindings API オペレーションを呼び出して生成された検出結果を取得することもできます。

GuardDuty の検出結果は、Amazon CloudWatch Events を使用して確認することもできます。GuardDuty は、検出結果を HTTPS プロトコルを介して Amazon CloudWatch に送信します。詳細については、「Amazon EventBridge を使用した GuardDuty 検出結果の処理」を参照してください。

IAM ロール

これは、S3 オブジェクトをスキャンするために必要なアクセス許可を持つ IAM ロールのことです。スキャンされたオブジェクトのタグ付けを有効にする場合、IAM PassRole アクセス許可により、スキャンされたオブジェクトに GuardDuty がタグを追加することができます。

Malware Protection プランのリソース

バケットの Malware Protection for S3 を有効にすると、GuardDuty は Malware Protection プランのリソースを作成します。このリソースは、保護されたバケットを一意に識別する Malware Protection プラン ID に関連付けられます。Malware Protection プランのリソースを使用して、保護されたリソースに対して API オペレーションを実行します。

保護されたバケット (保護されたリソース)

Amazon S3 バケットは、このバケットで Malware Protection for S3 を有効にし、その保護ステータスが [アクティブ] に変わると、保護されているとみなされます。

GuardDuty は、保護されたリソースとして S3 バケットのみをサポートします。

保護ステータス

Malware Protection プランのリソースに関連付けられているステータス。バケットで Malware Protection for S3 を有効にすると、このステータスはバケットが正しく設定されているかどうかを表します。

S3 オブジェクトプレフィックス

Amazon Simple Storage Service (Amazon S3) バケットでは、プレフィックスを使用してストレージを整理できます。プレフィックスは、S3 バケット内のオブジェクトの論理グループです。詳細については、「Amazon S3 ユーザーガイド」の「オブジェクトの整理とリスト化」を参照してください。

スキャンオプション

GuardDuty Malware Protection for EC2 が有効になっている場合、スキャンまたはスキップする Amazon EC2 インスタンスと Amazon Elastic Block Store (EBS) ボリュームを指定できます。この機能を使用すると、EC2 インスタンスおよび EBS ボリュームに関連付けられている既存のタグを、包含タグリストまたは除外タグリストのいずれかに追加できます。包含タグリストに追加するタグに関連付けられているリソースは、マルウェアのスキャンが行われ、除外タグリストに追加されたリソースはスキャンされません。詳細については、「ユーザー定義タグ付きのスキャンオプション」を参照してください。

スナップショットの保持

GuardDuty Malware Protection for EC2 を有効にすると、EBS ボリュームのスナップショットを AWS アカウント内に保持するオプションが提供されます。GuardDuty は、EBS ボリュームのスナップショットに基づいてレプリカ EBS ボリュームを生成します。Malware Protection for EC2 スキャンがレプリカ EBS ボリュームでマルウェアを検出した場合にのみ、EBS ボリュームのスナップショットを保持できます。レプリカ EBS ボリュームでマルウェアが検出されない場合、GuardDuty は、スナップショットの保持設定に関係なく、EBS ボリュームのスナップショットを自動的に削除します。詳細については、「スナップショットの保持」を参照してください。

抑制ルール

抑制ルールを使用すると、特定の属性の組み合わせを作成して検出結果数を抑えることができます。例えば、特定の VPC のインスタンス、特定の AMI を実行するインスタンス、特定の EC2 タグがあるインスタンスなどのみで、`Recon:EC2/Portscan` を自動アーカイブするために、GuardDuty フィルターを使ってルールを定義できます。このルールにより、ポートスキャンの検出結果は、条件を満たすインスタンスから自動的にアーカイブされます。ただし、暗号化通貨のマイニングなど、他の悪意のあるアクティビティを行っているインスタンスが GuardDuty によって検出された場合には、アラートが出されます。

GuardDuty 管理者アカウントで定義された抑制ルールは、GuardDuty メンバーアカウントに適用されます。GuardDuty メンバーアカウントは抑制ルールを変更できません。

抑制ルールでは、GuardDuty は依然としてすべての検出結果を生成します。抑制ルールは、すべてのアクティビティの完全で不変な履歴を維持しながら、検出結果の数を抑えます。

通常、抑制ルールの使用目的は、環境に対して誤検知と判断した検出結果を非表示にし、重大度の低い検出結果からのノイズを減らして、より大きな脅威に集中できるようにすることです。詳細については、「GuardDuty の抑制ルール」を参照してください。

信頼できる IP リスト

AWS 環境と高度に安全な通信を行うための、信頼できる IP アドレスのリスト GuardDuty では、信頼できる IP リストに基づく検出結果は生成されません。詳細については、「Customizing threat detection with entity lists and IP address lists」を参照してください。

脅威 IP リスト

悪意のある既知の IP アドレスのリスト GuardDuty は、疑わしいアクティビティの可能性があるために検出結果を生成するだけでなく、これらの脅威リストに基づく検出結果も生成します。詳細については、「Customizing threat detection with entity lists and IP address lists」を参照してください。

GuardDuty の開始方法

このチュートリアルでは、GuardDuty の実践入門を学習します。でスタンドアロンアカウントまたは GuardDuty 管理者として GuardDuty を有効にするための最小要件 AWS Organizations については、ステップ 1 で説明します。ステップ 2 ~ 5 では、検出結果を最大限に活用するために、GuardDuty が推奨する追加機能の使用をカバーします。

トピック

- [開始する前に]
- ステップ 1: Amazon GuardDuty を有効にする
- ステップ 2: サンプル検出結果を生成し、ベーシックなオペレーションの詳細を確認する
- ステップ 3: Amazon S3 バケットへの GuardDuty の検出結果のエクスポートを設定する
- ステップ 4: SNS を使用して GuardDuty の検出結果アラートを設定する
- 次のステップ

[開始する前に]

GuardDuty は、 AWS CloudTrail 管理イベント、Amazon VPC フローログ、DNS Amazon Route 53 Resolver クエリログ基本データソースなどの をモニタリングする脅威検出サービスです。また、GuardDuty は、保護タイプに関連する機能を個別に有効にした場合にのみ分析します。機能には、Kubernetes 監査ログ、RDS ログインアクティビティ、Amazon S3 AWS CloudTrail のデータイベント、Amazon EBS ボリューム、ランタイムモニタリング、Lambda ネットワークアクティビティログなどがあります。 Amazon S3 これらのデータソースおよび機能 (有効である場合) を使用して、GuardDuty はアカウントのセキュリティ結果を生成します。

GuardDuty を有効にすると、基本的なデータソースのアクティビティに基づいて、潜在的な脅威がないかアカウントのモニタリングが開始されます。デフォルトでは、GuardDuty が有効になっている AWS アカウント るすべての に対して Extended Threat Detectionが有効になっています。この機能は、アカウント内の複数の基本的なデータソース、 AWS リソース、および時間にわたるマルチステージ攻撃シーケンスを検出します。特定の AWS リソースに対する潜在的な脅威を検出するには、GuardDuty が提供するユースケースに焦点を当てた保護プランを有効にすることを選択できます。詳細については、「GuardDuty の機能」を参照してください。

基礎データソースのいずれかを明示的に有効にする必要はありません。S3 Protection を有効にした場合は、Amazon S3 データイベントのログ記録を明示的に有効にする必要はありません。同様に、EKS Protection を有効にした場合は、Amazon EKS 監査ログを明示的に有効にする必要はあり

ません。Amazon GuardDuty は、これらのサービスから直接、データの独立したストリームを取得します。

新しい GuardDuty アカウントでは、 でサポートされている利用可能な保護タイプの一部 AWS リージョン が有効になり、デフォルトで 30 日間の無料トライアル期間に含まれます。それらのいずれか、またはすべての設定をオプトアウトできます。GuardDuty を有効に AWS アカウント した既存の がある場合は、リージョンで利用可能な保護プランのいずれかまたはすべてを有効にすることができます。保護プランの概要とデフォルトで有効になる保護プランについては、「GuardDuty の料金」を参照してください。

GuardDuty を有効にするときは、次の項目を考慮してください。

- GuardDuty はリージョンレベルのサービスです。つまり、このページで行う設定手順はすべて、GuardDuty を使用してモニタリングするリージョンごとに繰り返す必要があります。

  サポートされているすべての AWS リージョンで GuardDuty を有効にすることを強くお勧めします。このように設定することで、GuardDuty はアクティブに使用されていないリージョンでも、許可されていないアクティビティや異常なアクティビティに関する検出結果を生成できます。これにより、GuardDuty は IAM などのグローバル AWS サービスの AWS CloudTrail イベントをモニタリングすることもできます。GuardDuty がサポートされているすべてのリージョンで有効になっていない場合、グローバルサービスに関連するアクティビティを検出する機能は低下します。GuardDuty を使用できるリージョンのリストについては、「リージョンとエンドポイント」を参照してください。
- AWS アカウントの管理者権限を持つユーザーは誰でも GuardDuty を有効にできますが、最小特権のセキュリティのベストプラクティスに従って、GuardDuty を特に管理するための IAM ロール、ユーザー、またはグループを作成することをお勧めします。GuardDuty を有効にするために必要なアクセス権限の詳細については、「GuardDuty の有効化に必要な許可」を参照してください。
- GuardDuty を任意の で初めて有効にすると AWS リージョン、デフォルトでは、Malware Protection for EC2 など、そのリージョンでサポートされているすべての利用可能な保護タイプも有効になります。GuardDuty は、サービスにリンクされた `AWSServiceRoleForAmazonGuardDuty` という名前のロールをアカウントに作成します。このロールには、GuardDuty がセキュリティ検出結果を生成するために、GuardDuty 基本データソース から直接イベントを消費および分析することを可能にする許可と信頼ポリシーが含まれます。Malware Protection for EC2 は、別のサービスにリンクされた `AWSServiceRoleForAmazonGuardDutyMalwareProtection` という名前のロールをアカウントに作成します。このロールには、Malware Protection for EC2 がエージェントレススキャンを実行して GuardDuty アカウント内のマルウェアを検出できるようにするアクセス許可と信頼ポリシーが含まれます。これにより、GuardDuty はアカウントに EBS ボリュームスナップショットを

作成し、そのスナップショットを GuardDuty サービスアカウントと共有できます。詳細については、「GuardDuty のためのサービスにリンクされたロールの許可」を参照してください。サービスにリンクされたロールの詳細については、「サービスリンクロールの使用」を参照してください。

- 任意のリージョンで初めて GuardDuty を有効にすると、そのリージョンの 30 日間の GuardDuty 無料トライアルに AWS アカウントが自動的に登録されます。

次の動画では、管理者アカウントで GuardDuty の使用を開始し、その後複数のメンバーアカウントで有効にする方法について説明します。

使用開始: スタンドアロン環境またはマルチアカウント環境での Amazon GuardDuty の有効化

ステップ 1: Amazon GuardDuty を有効にする

GuardDuty を使用する最初のステップは、アカウントでそれを有効にすることです。有効にすると、GuardDuty は、直ちに現在のリージョンでセキュリティ脅威のモニタリングを開始します。

GuardDuty 管理者として、組織内の他のアカウントの GuardDuty の検出結果を管理する場合は、メンバーアカウントを追加し、それらのアカウントに対して GuardDuty を有効にする必要があります。

> **Note**
>
> GuardDuty は有効にしないで GuardDuty Malware Protection for S3 を有効にする場合の手順については、「GuardDuty Malware Protection for S3」を参照してください。

Standalone account environment

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. [Amazon GuardDuty - すべての機能] オプションを選択します。
3. [開始する] を選択します。
4. [GuardDuty にようこそ] ページで、サービス条件を表示します。[GuardDuty を有効にする] を選択します。

Multi-account environment

> **Important**
>
> このプロセスの前提条件として、組織内の GuardDuty の管理者を委任するには、管理するすべてのアカウントと同じ組織に存在し、 AWS Organizations 管理アカウントにアクセスできる必要があります。管理者の委任には追加の許可が必要になる場合があります。詳細については、「委任 GuardDuty 管理者アカウントの指定に必要な許可」を参照してください。

委任 GuardDuty 管理者アカウントを指定するには

1. 管理アカウントを使用して、https://console.aws.amazon.com/organizations/ で AWS Organizations コンソールを開きます。
2. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   アカウント内で GuardDuty が既に有効になっていますか?

   - GuardDuty がまだ有効になっていない場合は、[Get Started] (今すぐ始める) を選択し、[Welcome to GuardDuty] (GuardDuty にようこそ) ページで GuardDuty の委任された管理者を指定できます。

- GuardDuty が有効になっている場合、[Settings] (設定) ページで GuardDuty の委任された管理者を指定することができます。

3. 組織の GuardDuty 委任管理者として指定する AWS アカウントの 12 桁のアカウント ID を入力し、Delegate を選択します。

> **Note**
>
> GuardDuty がまだ有効になっていない場合、委任された管理者を指定すると、そのアカウントに対し現在のリージョンで GuardDuty が有効になります。

メンバーアカウントを追加するには

この手順では、 を介してメンバーアカウントを GuardDuty 委任管理者アカウントに追加する方法について説明します AWS Organizations。招待でメンバーを追加するオプションもあります。GuardDuty でメンバーを関連付ける 2 つの方法については、「Amazon GuardDuty での複数のアカウントの管理」を参照してください。

1. 委任された管理者のアカウントにログイン
2. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
3. ナビゲーションパネルで [設定] を選択してから、[アカウント] を選択します。

   アカウントの一覧に組織内のすべてのアカウントが表示されます。
4. アカウント ID の横にあるチェックボックスを選択し、メンバーとして追加したいアカウントを選択します。その後、[アクション] メニューから、[メンバーの追加] を選びます。

> **Tip**
>
> 新しいアカウントをメンバーとして追加するには、[Auto-enable] (自動有効化) の機能をオンにします。ただし、この機能は、機能を有効にした後に組織に参加したアカウントにのみ適用されます。

ステップ 2: サンプル検出結果を生成し、ベーシックなオペレーションの詳細を確認する

GuardDuty がセキュリティ上の問題を検出すると、検出結果が生成されます。GuardDuty の検出結果は、その固有のセキュリティ問題に関する詳細を含むデータセットです。検出結果の詳細を使用して、問題の調査に役立てることができます。

GuardDuty は、placeholder 値を使用したサンプル検出結果の生成をサポートしています。このサンプルを使用して、GuardDuty の機能をテストし、検出結果を理解してから GuardDuty によって発見された実際のセキュリティ問題に馴染むために使われます。GuardDuty で利用可能な各検出結果タイプのサンプル検出結果を生成するには、以下のガイドに従ってください。アカウント内でシミュレートされたセキュリティイベントを生成するなど、サンプル検出結果を生成するその他の方法については、「サンプルの検出結果」を参照してください。

サンプル検出結果を作成して調査するには

1. ナビゲーションペインで [設定] を選択します。
2. [設定] ページの [結果のサンプル] で、[結果サンプルの生成] を選択します。
3. ナビゲーションペインで、概要 を選択して、 AWS 環境で生成された結果に関するインサイトを表示します。概要ダッシュボードのコンポーネントの詳細については、「Amazon GuardDuty の [要約] ダッシュボード」を参照してください。
4. ナビゲーションペインで [結果] を選択します。[Current findings] (最近の結果) ページに [SAMPLE] とプレフィックスされたサンプル検出結果が表示されます。
5. リストから検出結果を選択して、検出結果の詳細を表示します。

    - 検出結果の詳細ペインで使用可能なさまざまな情報フィールドを確認できます。検出結果のタイプが異なると、フィールドが異なる場合があります。すべての検出結果タイプで使用可能なフィールドの詳細については、「検出結果の詳細」を参照してください。詳細ペインから、次のアクションを実行できます。

        - ペインの一番上の [結果 ID] を選択し、検出結果のための完全な JSON 詳細を開きます。完全な JSON ファイルは、このパネルからダウンロードすることもできます。JSON には、コンソールビューに含まれない追加情報が含まれており、他のツールやサービスで取り込むことができる形式です。
        - [影響を受けるリソース] セクションを表示します。実際の検出結果では、ここに記載されている情報は、調査すべきアカウント内のリソースを特定するのに役立ち、実用的なリソース AWS マネジメントコンソール に適した へのリンクが含まれます。

- 「+」または「-」の虫眼鏡アイコンを選択して、その詳細の包括的または排他的なフィルターを作成します。検出結果フィルターの詳細については、「GuardDuty での検出結果のフィルタリング」を参照してください。

6. サンプル検出結果をすべてアーカイブする

   a. リストの上部にあるチェックボックスを選んで、すべての検出結果を選択します。

   b. 保持したい検出結果を選択解除します。

   c. [アクション] メニューを選択し、[アーカイブ] をクリックして、サンプルの検出結果を非表示にします。

> **Note**
>
> Actions メニューを見るために、[最新] を選択し、検出結果ビューを切り替えるために [アーカイブ] を選択してください。

ステップ 3: Amazon S3 バケットへの GuardDuty の検出結果のエクスポートを設定する

GuardDuty では、検出結果のエクスポートを設定することをお勧めします。これにより、GuardDuty の 90 日間の保持期間を超えた無期限のストレージとして S3 バケットに検出結果をエクスポートできます。これにより、検出結果の記録を保持したり、 AWS 環境内の問題を経時的に追跡したりできます。GuardDuty は、 AWS Key Management Service () を使用して S3 バケットの検出結果データを暗号化しますAWS KMS key。設定を構成するには、GuardDuty に KMS キーへのアクセス許可を付与する必要があります。より詳細なステップについては、「生成された検出結果を Amazon S3 にエクスポートする」を参照してください。

GuardDuty の検出結果を Amazon S3 バケットにエクスポートするには

1. KMS キーにポリシーをアタッチする

   a. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/kms で AWS Key Management Service (AWS KMS) コンソールを開きます。

   b. を変更するには AWS リージョン、ページの右上隅にあるリージョンセレクターを使用します。

   c. ナビゲーションペインで、[カスタマーマネージドキー] を選択します。

d. 既存の KMS キーを選択するか、「 AWS Key Management Service デベロッパーガイド」の「対称暗号化 KMS キーを作成する」の手順を実行します。

KMS キーのリージョンおよび Amazon S3 バケットは同じである必要があります。

後のステップで使用するために、キー ARN をメモ帳にコピーします。

e. KMS キーの [キーポリシー] セクションで、[編集] を選択します。[ポリシービューへの切り替え] が表示された場合はそれを選択して [キーポリシー] を表示してから、[編集] を選択します。

f. 以下のポリシーブロックを KMS キーポリシーにコピーしてください:

```
{
    "Sid": "AllowGuardDutyKey",
    "Effect": "Allow",
    "Principal": {
        "Service": "guardduty.amazonaws.com"
    },
    "Action": "kms:GenerateDataKey",
    "Resource": "KMS key ARN",
    "Condition": {
        "StringEquals": {
            "aws:SourceAccount": "123456789012",
            "aws:SourceArn":
 "arn:aws:guardduty:Region2:123456789012:detector/SourceDetectorID"
        }
    }
}
```

ポリシーの例で#でフォーマットされている次の値を置き換えることにより、ポリシーを編集します。

1. *KMS key ARN* は、KMS キーの Amazon リソースネーム (ARN) に置き換えます。キー ARN を検索するには、「AWS Key Management Service 開発者ガイド」の「Finding the key ID and ARN」を参照してください。
2. *123456789012* は、検出結果をエクスポートする GuardDuty アカウントを所有する AWS アカウント ID に置き換えます。
3. *Region2* を、GuardDuty の結果が生成される AWS リージョン に置き換えます。
4. *SourceDetectorID* は、検出結果が生成された特定のリージョンの GuardDuty アカウントの detectorID に置き換えます。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

2. Amazon S3 バケットにポリシーをアタッチする

これらの検出結果をエクスポートする Amazon S3 バケットがまだない場合は、「Amazon S3 ユーザーガイド」の「バケットの作成」を参照してください。

a. [バケットポリシーを編集] ページが表示されるまで、「Amazon S3 ユーザーガイド」の「バケットポリシーを作成または編集するには」の手順を実行します。

b. 次のポリシー例は、Amazon S3 バケットに検出結果をエクスポートする許可を GuardDuty に付与する方法を示しています。検出結果のエクスポートを設定した後にパスを変更する場合は、新しい場所に対する許可を付与するように、ポリシーを変更する必要があります。

次のポリシー例をコピーし、[バケットポリシーエディタ] に貼り付けます。

最後のステートメントの前にポリシーステートメントを追加した場合は、このステートメントを追加する前にカンマを追加します。KMS キーポリシーの JSON 構文が有効であることを確認します。

S3 バケットポリシーの例

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "Allow GetBucketLocation",
            "Effect": "Allow",
            "Principal": {
                "Service": "guardduty.amazonaws.com"
            },
            "Action": "s3:GetBucketLocation",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "123456789012",
```

```
                    "aws:SourceArn": "arn:aws:guardduty:us-
east-2:123456789012:detector/SourceDetectorID"

                }
            }
        },
        {
            "Sid": "Allow PutObject",
            "Effect": "Allow",
            "Principal": {
                "Service": "guardduty.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket[optional
 prefix]/*",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "123456789012",
                    "aws:SourceArn": "arn:aws:guardduty:us-
east-2:123456789012:detector/SourceDetectorID"

                }
            }
        },
        {
            "Sid": "Deny unencrypted object uploads",
            "Effect": "Deny",
            "Principal": {
                "Service": "guardduty.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket[optional
 prefix]/*",
            "Condition": {
                "StringNotEquals": {
                    "s3:x-amz-server-side-encryption": "aws:kms"
                }
            }
        },
        {
            "Sid": "Deny incorrect encryption header",
            "Effect": "Deny",
            "Principal": {
                "Service": "guardduty.amazonaws.com"
```

```
            },
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket[optional
 prefix]/*",
            "Condition": {
                "StringNotEquals": {
                "s3:x-amz-server-side-encryption-aws-kms-key-id":
 "arn:aws:kms:us-east-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111"
                }
            }
        },
        {
            "Sid": "Deny non-HTTPS access",
            "Effect": "Deny",
            "Principal": "*",
            "Action": "s3:*",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket[optional
 prefix]/*",
            "Condition": {
                "Bool": {
                    "aws:SecureTransport": "false"
                }
            }
        }
    ]
}
```

c. ポリシーの例で#でフォーマットされている次の値を置き換えることにより、ポリシーを編集します。

1. *Amazon S3 bucket ARN* は、Amazon S3 バケットの Amazon リソースネーム (ARN) に置き換えます。[バケット ARN] は、https://console.aws.amazon.com/s3/ コンソールの [バケットポリシーを編集] ページで確認できます。
2. *123456789012* は、検出結果をエクスポートする GuardDuty アカウントを所有する AWS アカウント ID に置き換えます。
3. *Region2* は、GuardDuty の検出結果が生成される AWS リージョン に置き換えます。
4. *SourceDetectorID* は、検出結果が生成された特定のリージョンの GuardDuty アカウントの `detectorID` に置き換えます。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

5. *`S3 bucket ARN/[optional prefix]`* プレースホルダー値の *`[optional prefix]`* 部分は、検出結果をエクスポートするオプションのフォルダの場所に置き換えます。プレフィックスの使用の詳細については、「Amazon S3 ユーザーガイド」の「プレフィックスを使用してオブジェクトを整理する」を参照してください。

   まだ存在していないオプションのフォルダの場所を指定した場合、GuardDuty は、S3 バケットに関連付けられたアカウントが検出結果をエクスポートしているアカウントと同じ場合にのみ、その場所を作成します。別のアカウントに属する S3 バケットに検出結果をエクスポートする場合は、このフォルダの場所が既に存在している必要があります。

6. *`KMS key ARN`* は、S3 バケットにエクスポートされた検出結果の暗号化に関連付けられた KMS キーの Amazon リソースネーム (ARN) に置き換えます。キー ARN を検索するには、「AWS Key Management Service 開発者ガイド」の「Finding the key ID and ARN」を参照してください。

3. GuardDuty コンソールのステップ

   a. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   b. ナビゲーションペインで [設定] を選択します。

   c. [設定] ページの [検出結果のエクスポートオプション] で、[S3 バケット] の [今すぐ設定] (または必要に応じて [編集]) を選択します。

   d. S3 バケット ARN には、検出結果を送信する **`bucket ARN`** を入力します。バケット ARN を表示するには、「Amazon S3 ユーザーガイド」の「S3 バケットのプロパティの表示」を参照してください。

   e. [KMS キー ARN] で、**`key ARN`** を入力します。キー ARN を検索するには、「AWS Key Management Service デベロッパーガイド」の「キー ID とキー ARN を検索する」を参照してください。

   f. [保存] を選択します。

ステップ 4: SNS を使用して GuardDuty の検出結果アラートを設定する

GuardDuty は、Amazon EventBridge と統合されており、これを使用して、検出結果データを他のアプリケーションやサービスに送信して処理することができます。EventBridge では、GuardDuty の検出結果を使用して、検出結果イベントを AWS Lambda 関数、Amazon EC2 Systems Manager オートメーション、Amazon Simple Notification Service (SNS) などのターゲットに接続することで、検出結果への自動応答を開始できます。

この例では、EventBridge ルールのターゲットとなる SNS トピックを作成し、EventBridge を使用して GuardDuty から検出結果データをキャプチャするルールを作成します。結果として得られるルールは、検出結果の詳細をメールアドレスに転送します。検出結果を Slack または Amazon Chime に送信する方法、および送信される検出結果のアラートタイプを変更する方法については、「Amazon SNS トピックおよびエンドポイントの設定」を参照してください。

検出結果アラートの SNS トピックを作成するには

1. Amazon SNS コンソール (https://console.aws.amazon.com/sns/v3/home) を開きます。
2. ナビゲーションペインで、[トピック] を選択してください。
3. [トピックの作成] を選択します。
4. [Type] (タイプ) に、[Standard] (標準) を選択します。
5. [名前] に **`GuardDuty`** と入力します。
6. [トピックの作成] を選択します。新しいトピックのトピック詳細が開きます。
7. [サブスクリプション] セクションで、[サブスクリプションの作成] を選択します。
8. [プロトコル] で [E メール] を選択します。
9. [Endpoint] (エンドポイント) に、通知を送信する先の E メールアドレスを入力します。
10. [Create subscription] (サブスクリプションの作成) を選択します。

    サブスクリプションを作成した後に、E メールを通してサブスクリプションを確認する必要があります。
11. サブスクリプションメッセージを確認するには、E メールの受信ボックスに移動して、サブスクリプションメッセージで [Confirm subscription] (サブスクリプションの確認) を選択します。

> **Note**
>
> メール確認のステータスを確認するには、SNS コンソールに移動して [Subscriptions] (サブスクリプション) を選択します。

GuardDuty の検出結果をキャプチャしてフォーマットする EventBridge ルールを作成するには

1. https://console.aws.amazon.com/events/ で EventBridge コンソールを開きます。
2. ナビゲーションペインの [バス] で、[ルール] を選択します。
3. [ルールの作成] を選択します。
4. ルールの名前と説明を入力します。

   ルールには同じリージョン内および同じイベントバス上の別のルールと同じ名前を付けることはできません。
5. イベントバスを選択 の下で、 デフォルトのイベントバスが選択され、選択したイベントバスのルールを有効にするがオンになっていることを確認します。
6. [ルールタイプ] で、[イベントパターンを持つルール] を選択してください。
7. [次へ] を選択します。
8. [イベントパターンを構築] ページの [イベントソース] で、[AWS イベント] または [EventBridge パートナーイベント] を選択します。
9. 「イベントパターン」セクションの「作成方法」で、「パターンフォームを使用する」を選択します。
10. [イベントパターンフォーム] では、AWS [サービス] を選択します。
11. [AWS service] (サービス) では、[GuardDuty] を選択します。
12. [イベントタイプ] では、[GuardDuty の検出結果] を選択します。
13. [Next] (次へ) を選択します。
14. [ターゲットタイプ] では、[AWS サービス] を選択します。
15. [Select a target] (ターゲットの選択) には、[SNS topic] (SNS トピック) を選択してください。ターゲットの場所で、このアカウントのターゲットを選択します。トピック で、前に作成した SNS トピックの名前を選択します。
16. [追加の設定] を展開します。[ターゲット入力の設定] では、[入力トランスフォーマー] を選択します。

入力トランスフォーマーを追加すると、GuardDuty から送信される JSON 検出結果データが、人が読み取り可能なメッセージに書式設定されます。

17. [Configure input transformer] (入力トランスフォーマーの設定) を選択します。
18. [Target input transformer] (ターゲット入力トランスフォーマー) セクションで、[Input path] (入力パス) に以下のコードを貼り付けます。

```
{
  "severity": "$.detail.severity",
  "Finding_ID": "$.detail.id",
  "Finding_Type": "$.detail.type",
  "region": "$.region",
  "Finding_Description": "$.detail.description"
}
```

19. E メールをフォーマットするには、Template に次のコードを貼り付けます。EventBridge が各検出結果を配信すると、入力トランスフォーマーは入力パス<Finding_ID>から<severity>や などの角括弧で値を入力するため、置き換える必要はありません。

```
{"message":"You have a severity <severity> GuardDuty finding type <Finding_Type> in
 the <region> Region.\n\nFinding Description:\n<Finding_Description>\n\nFor more
 details open the GuardDuty console at https://console.aws.amazon.com/guardduty/
home?region=<region>#/findings?search=id%3D<Finding_ID>"}
```

> **Important**
>
> テンプレートを単一行の JSON オブジェクトとして指定します。前の例messageに示すように、メッセージは などのキーでラップされます。\n を使用して改行を追加します。
> などの GuardDuty 検出結果フィールドには、改行を含むエスケープされていないコントロール文字を含めるdescriptionことができます。次のいずれかのテンプレート形式を使用しないでください。
>
> - 裸引用符で囲まれた文字列
> - 別々の行で別々の引用符で囲まれた文字列を使用する複数行テンプレート

これらの形式では、置換された値が無効な JSON を生成する可能性があります。その後、EventBridge はイベントを拒否し、通知を配信しません。

> **Note**
>
> E メールサブスクリプションはテンプレート出力をそのまま受け取るため、E メールには `message` JSON オブジェクトが表示され、改行は として表示されます\n。フォーマットされた通知の場合は、チャットアプリケーション (Slack または Amazon Chime) で Amazon Q Developer に Amazon SNS トピックを配信するか、Lambda 関数をターゲットとして使用します。

20. [確認] を選択します。
21. [次へ] をクリックします。
22. (オプション) ルールに 1 つ以上のタグを入力します。詳細については、Amazon EventBridge ユーザーガイドのAmazon EventBridge のタグを参照してください。
23. 次へをクリックします。
24. ルールの詳細を確認し、[ルールの作成] を選択します。
25. (オプション) ステップ 2 のプロセスでサンプル検出結果を生成して、新しいルールをテストします。生成されたサンプル検出結果ごとにメールが届きます。

次のステップ

GuardDuty を引き続き使用すると、環境に関連する検出結果の種類を理解できるようになります。新しい検出結果を受け取ったときはいつでも、検出結果の詳細ペイン中の検出結果の記述から [詳細はこちら] を選択することにより、あるいは GuardDuty 検出結果タイプ の検出結果名を探すことにより、その検出結果に関する修復に関するレコメンデーションを含む情報を検索できます。

以下の機能は、GuardDuty を調整して AWS 、環境に最も関連性の高い検出結果を提供できるようにするのに役立ちます。

- インスタンス ID、アカウント ID、S3 バケット名などの特定の条件に基づいて検出結果を簡単に並べ替えるには、GuardDuty 内でフィルターを作成して保存できます。詳細については、「GuardDuty での検出結果のフィルタリング」を参照してください。

- 環境内で予想される動作に関する検出結果を受け取っている場合は、[suppression rules] (抑制ルール) で定義した基準に基づいて検出結果を自動的にアーカイブできます。
- 信頼できる IP のサブセットから検出結果を生成しないようにしたり、GuardDuty が通常のモニタリング範囲外の IP をモニタリングするには、[Trusted IP and threat lists] (信頼されている IP および脅威リスト) をセットアップできます。

GuardDuty 基本データソース

GuardDuty は基本データソースを使用して、既知の悪質なドメインや IP アドレスとの通信を検出し、潜在的に異常な動作と不正なアクティビティを特定します。これらのソースから GuardDuty への通信中、ログデータはすべて暗号化されます。GuardDuty は、プロファイリングや異常の検出用のログソースから様々なフィールドを抽出して、これらのログを破棄します。

リージョンで初めて GuardDuty を有効にするときは、すべての基本データソースの脅威検出を含む 30 日間の無料トライアルがあります。この無料トライアルでは、基本データソースごとに分類された月単位の推定使用量をモニタリングできます。委任 GuardDuty 管理者アカウントは、組織に属し、GuardDuty を有効にしているメンバーアカウントごとに分類された月単位の推定使用コストを表示できます。30 日間のトライアルが終了したら、 を使用して使用コスト AWS Billing に関する情報を得ることができます。

GuardDuty がこれらの基本データソースからのイベントとログにアクセスする場合、追加料金はかかりません。

で GuardDuty を有効にすると AWS アカウント、次のセクションで説明するログソースのモニタリングが自動的に開始されます。GuardDuty がこれらのデータソースの分析と処理を開始して関連付けられたセキュリティ検出結果を生成するのに、他のものを有効にする必要はありません。

トピック

- AWS CloudTrail 管理イベント
- VPC フローログ
- Route53 Resolver DNS クエリログ

AWS CloudTrail 管理イベント

AWS CloudTrail は、、 AWS マネジメントコンソール AWS SDKs、コマンドラインツール、および特定の AWS サービスを使用して行われた AWS API コールなど、アカウントの API コールの履歴を提供します。CloudTrail は、CloudTrail をサポートするサービスの AWS APIs を呼び出したユーザーとアカウント、呼び出し元のソース IP アドレス、および呼び出しが呼び出された時間を特定するのにも役立ちます。詳細については、「AWS CloudTrail ユーザーガイド」の「AWS CloudTrailとは」を参照してください。

GuardDuty は、コントロールプレーンイベントとも呼ばれる CloudTrail 管理イベントをモニタリングします。これらのイベントは、 のリソースで実行される管理オペレーションに関するインサイトを提供します AWS アカウント。

次は、GuardDuty がモニタリングする CloudTrail 管理イベントの例です。

- セキュリティの設定 (IAM `AttachRolePolicy` API オペレーション)
- データをルーティングするルールの設定 (Amazon EC2 `CreateSubnet` API オペレーション)
- ログ記録の設定 (AWS CloudTrail `CreateTrail` API オペレーション)

GuardDuty を有効にすると、イベントの独立した重複ストリームを通して CloudTrail 管理イベントが直接 CloudTrail から読み込まれ、CloudTrail イベントログが分析されます。

GuardDuty は CloudTrail イベントを管理したり、既存の CloudTrail 設定に影響を与えたりすることはありません。同様に、CloudTrail の設定は GuardDuty がイベントログを消費する方法や処理する方法に影響しません。CloudTrail イベントのアクセスと保持を管理するには、CloudTrail サービスコンソールまたは API を使用します。詳細については、「AWS CloudTrail ユーザーガイド」の「Viewing events with CloudTrail event history」を参照してください。

GuardDuty が AWS CloudTrail グローバルイベントを処理する方法

ほとんどの AWS サービスでは、CloudTrail イベントは作成された AWS リージョン に記録されます。 AWS Identity and Access Management (IAM) AWS Security Token Service 、(AWS STS)、Amazon Simple Storage Service (Amazon S3)、Amazon CloudFront、Amazon Route 53 (Route 53) などのグローバルサービスの場合、イベントは発生したリージョンでのみ生成されますが、グローバルに重要です。

GuardDuty がネットワーク設定やユーザーアクセス許可などのセキュリティ値を持つ CloudTrail グローバルサービスイベント (GSE) を消費すると、それらのイベントが複製され、GuardDuty が有効な各リージョンで処理されます。その結果、異常なイベントの検出に不可欠な各リージョンのユーザーとロールのプロファイルを GuardDuty で維持しやすくなります。

> **Note**
>
> これらのグローバルサービスイベントから生成された検出結果について、検出結果内の地域値は、 GuardDuty が検出を作成した地域と異なる場合があります。例えば、GuardDuty が別のリージョンで検出を作成しても、検出結果がリージョン `us-east-1` として表示される場合があります。

で使用できるすべての で GuardDuty AWS リージョン を有効にすることをお勧めします AWS アカウント。特定のリージョンにリソースを展開していなくても、GuardDuty を有効にすることでアカウントを潜在的な脅威から保護できます。脅威アクターは、グローバルサービス (IAM、 AWS STS Amazon CloudFront など) を介して攻撃を仕掛ける可能性があります。存在が制限されているリージョンを悪用するために、不正なリソースを作成しようとする可能性があります。GuardDuty は、デフォルトリージョンとオプトインリージョンの両方で、サービスが有効化されているすべてのリージョンにおけるグローバルサービスイベントを処理します。これにより、GuardDuty は、リソースをアクティブに使用していないリージョンなど、 AWS アカウント全体で疑わしい可能性のあるアクティビティを検出するのに役立ちます。

VPC フローログ

Amazon VPC の VPC フローログ機能は、 AWS 環境内の Amazon Elastic Compute Cloud (Amazon EC2) インスタンスに接続されたネットワークインターフェイスとの間で送受信される IP トラフィックに関する情報をキャプチャします。

GuardDuty を有効にすると、アカウント内の Amazon EC2 インスタンスからの VPC フローログデータの分析がすぐに開始されます。GuardDuty は、フローログの独立した重複ストリームを介してこのデータを消費します。GuardDuty がこのデータにアクセスするために、アカウントで VPC フローログを作成または設定する必要はありません。このプロセスによる既存のフローログ設定への影響はありません。

Lambda Protection

Lambda Protection は Amazon GuardDuty の拡張機能です (オプション)。現在、Lambda Network Activity Monitoring には、VPC ネットワークを使用しないログも含め、アカウントのすべての Lambda 関数からの Amazon VPC フローログが含まれています。Lambda 関数を潜在的なセキュリティ脅威から保護するには、GuardDuty アカウントで Lambda 保護を設定する必要があります。詳細については、「Lambda Protection」を参照してください。

GuardDuty Runtime Monitoring

GuardDuty セキュリティエージェントが Amazon EC2 インスタンスにデプロイされ、そこからランタイムイベントを受信した場合、GuardDuty はそのインスタンスからの VPC フローログの分析 AWS アカウント に対して に課金しません。これにより、2 倍の使用コストを回避できます。

GuardDuty はフローログを管理したり、アカウントへのアクセスを可能にしたりしません。フローログのアクセスと保持期間を管理するには、VPC フローログ機能を設定する必要があります。

Route53 Resolver DNS クエリログ

Amazon EC2 インスタンスに AWS DNS リゾルバーを使用する場合 (デフォルト設定) 、GuardDuty は内部 DNS リゾルバーを介して Route53 Resolver DNS AWS クエリログにアクセスしてリクエストとレスポンスを処理できます。OpenDNS や GoogleDNS などの別の DNS リゾルバーを使用している場合、または独自の DNS リゾルバーを設定している場合、GuardDuty は、このデータソースのデータにアクセスして処理できません。

GuardDuty を有効にすると、独立データストリームから Route53 Resolver DNS クエリログの分析がすぐに開始されます。このデータストリームは、Route 53 リゾルバークエリログ記録機能を通じて提供されるデータとは別のものです。この機能の設定は、GuardDuty の解析には影響しません。

> **Note**
>
> GuardDuty は、 Amazon Route 53 Resolver クエリログ記録機能がその環境で使用できない AWS Outposts ため、 で起動された Amazon EC2 インスタンスの DNS ログのモニタリングをサポートしていません。

GuardDuty Extended Threat Detection

GuardDuty 拡張脅威検出は、 内のデータソース、複数のタイプの AWS リソース、および時間にわたるマルチステージ攻撃を自動的に検出します AWS アカウント。この機能を使用すると、GuardDuty はさまざまなタイプのデータソースをモニタリングして観察する複数のイベントのシーケンスに焦点を当てます。拡張脅威検出は、これらのイベントを関連付けて、 AWS 環境に対する潜在的な脅威として存在するシナリオを特定し、攻撃シーケンスの検出結果を生成します。

トピック

- 攻撃シーケンスの脅威シナリオの例
- 仕組み
- 脅威検出を最大化するための保護プランの有効化
- GuardDuty コンソールの Extended Threat Detection
- 攻撃シーケンスの検出結果の理解と管理
- その他のリソース

攻撃シーケンスの脅威シナリオの例

拡張脅威検出は、 AWS 認証情報の悪用、Amazon S3 バケットでのデータ侵害の試み、Amazon EKS クラスターでのコンテナと Kubernetes リソースの侵害に関連する脅威シナリオを対象としています。1 つの検出結果には、攻撃シーケンス全体を含めることができます。例えば、次のリストは、GuardDuty が検出する可能性のあるシナリオを示しています。

例 1 - AWS 認証情報と Amazon S3 バケットデータの侵害

- コンピューティングワークロードへの不正アクセスを取得する脅威アクター。
- 次に、アクターは権限の昇格や永続化の確立などの一連のアクションを実行します。
- 最後に、アクターは Amazon S3 リソースからデータを抽出します。

例 2 - Amazon EKS クラスターの侵害

- 脅威アクターは、Amazon EKS クラスター内のコンテナアプリケーションを悪用しようとします。
- アクターは、侵害されたコンテナを使用して特権サービスアカウントトークンを取得します。
- 次に、アクターはこれらの昇格された権限を活用して、ポッドアイデンティティを介して機密性の高い Kubernetes シークレットまたは AWS リソースにアクセスします。

関連する脅威シナリオの性質上、GuardDuty はすべての 攻撃シーケンスの検出結果タイプ を重大と見なします。

以下の動画では、Extended Threat Detection の使用方法を示します。

Amazon GuardDuty Extended Threat Detection のデモンストレーション

仕組み

特定の のアカウントで Amazon GuardDuty を有効にすると AWS リージョン、拡張脅威検出もデフォルトで有効になります。Extended Threat Detection の使用に関連する追加コストはありません。デフォルトでは、すべての 基本データソース のイベントを関連付けます。ただし、S3 Protection、EKS Protection、Runtime Monitoring など、より多くの GuardDuty 保護プランを有効にすると、イベントソースの範囲を広げることで、追加のタイプの攻撃シーケンス検出が可能になります。これは、より包括的な脅威分析と攻撃シーケンスの検出の向上に役立つ可能性があります。詳細については、「脅威検出を最大化するための保護プランの有効化」を参照してください。

GuardDuty は、API アクティビティや GuardDuty の検出結果など、複数のイベントを関連付けます。これらのイベントは シグナル と呼ばれます。環境内で、単独では明確な潜在的な脅威とならないイベントが発生する場合があります。GuardDuty はそれらを弱いシグナルと呼んでいます。Extended Threat Detection を使用すると、GuardDuty は複数のアクションのシーケンスが疑

わしいアクティビティと一致するタイミングを特定し、アカウントに攻撃シーケンスの検出結果を生成します。これらの複数のアクションには、弱いシグナルやアカウント内で既に特定された GuardDuty の検出結果が含まれる場合があります。

> **Note**
>
> 攻撃シーケンスのイベントを相関させる場合、Extended Threat Detection は、抑制ルール のために自動的にアーカイブされる検出結果を含め、アーカイブされた検出結果を考慮しません。この動作により、アクティブで関連するシグナルのみが攻撃シーケンスの検出に寄与します。この影響を受けないようにするには、アカウントの既存の抑制ルールを確認してください。詳細については、「Extended Threat Detection での抑制ルールの使用」を参照してください。

GuardDuty は、アカウントで進行中の潜在的な動作または最近の攻撃動作 (24 時間のローリング時間枠内) を特定するように設計されています。例えば、攻撃者がコンピューティングワークロードへの意図しないアクセスを取得すると、攻撃が開始される可能性があります。次に、アクターは列挙、権限のエスカレーション、 AWS 認証情報の流出など、一連のステップを実行します。これらの認証情報は、さらなる侵害やデータへの悪意のあるアクセスに使用される可能性があります。

脅威検出を最大化するための保護プランの有効化

拡張脅威検出は、すべての GuardDuty アカウントで自動的に有効になり、アカウントで有効になっているすべての保護プランからのシグナルがデフォルトで評価されます。GuardDuty Foundational データソースは、いくつかの攻撃シーケンス検出に重要なシグナルを提供します。例えば、基本的な脅威検知機能により、GuardDuty は Amazon S3 API における IAM 特権発見活動から始まる潜在的な攻撃シーケンスを特定し、バケットリソースポリシーをより許可的なものに変更するといった、その後の S3 コントロールプレーンの変更を検知できます。ただし、他の攻撃シーケンスでは、ランタイムモニタリング、S3 保護、EKS 監査ログモニタリングなどの高度な保護プランでのみ利用できる高度なシグナルが必要です。S3

トピック

- Amazon EKS クラスターでの攻撃シーケンスの検出
- Amazon S3 バケット内の攻撃シーケンスの検出
- Amazon ECS クラスターでの攻撃シーケンスの検出
- Amazon EC2 インスタンスグループの攻撃シーケンスの検出

- AI ワークロードを含む攻撃シーケンスの検出

Amazon EKS クラスターでの攻撃シーケンスの検出

GuardDuty は、EKS 監査ログ、プロセスのランタイム動作、 AWS API アクティビティにわたって複数のセキュリティシグナルを相関させて、高度な攻撃パターンを検出しました。EKS の Extended Threat Detection を活用するには、EKS Protection または Runtime Monitoring (EKS アドオンを使用) の少なくとも 1 つの機能を有効にする必要があります。EKS Protection は監査ログを通じてコントロールプレーンのアクティビティをモニタリングし、Runtime Monitoring はコンテナ内の動作を監視します。

最大カバレッジと包括的な脅威検出のために、GuardDuty では両方の保護プランを有効にすることをお勧めします。これらを組み合わせることで、EKS クラスターの完全なビューが作成され、GuardDuty が複雑な攻撃パターンを検出できるようになります。例えば、特権コンテナの異常なデプロイ (EKS Protection で検出) を識別し、その後、そのコンテナ内で永続化の試行、クリプトマイニング、リバースシェルの作成 (Runtime Monitoring で検出) を行うことができます。GuardDuty は、これらの関連イベントを AttackSequence:EKS/CompromisedCluster と呼ばれる単一の「重大」な検出結果として表します。両方の保護プランを有効にすると、攻撃シーケンスの検出結果は以下の脅威シナリオをカバーします。

- 脆弱なウェブアプリケーションを実行しているコンテナの侵害
- 認証情報の設定ミスによる不正アクセス
- 権限昇格の試み
- 疑わしい API リクエスト
- データに悪意を持ってアクセスしようとする

以下のリストは、これらの専用保護プランが個別に有効になっている場合の詳細を示しています。

EKS Protection

EKS Protection を有効にすると、GuardDuty は Amazon EKS クラスターコントロールプレーンアクティビティを含む攻撃シーケンスを検出できます。これにより、GuardDuty は EKS 監査ログと AWS API アクティビティを関連付けることができます。例えば、GuardDuty は、アクターがクラスターシークレットへの不正アクセスを試み、Kubernetes ロールベースのアクセスコントロール (RBAC) アクセス許可を変更し、特権ポッドを作成する攻撃シーケンスを検出できます。この保護プランを有効にする方法については、「EKS Protection」を参照してください。

Amazon EKS の Runtime Monitoring

Amazon EKS クラスターの Runtime Monitoring を有効にすると、GuardDuty はコンテナレベルの可視性で EKS 攻撃シーケンス検出を強化できます。これにより、GuardDuty は潜在的な悪意のあるプロセス、疑わしいランタイム動作、および潜在的なマルウェア実行を検出するのに役立ちます。例えば、GuardDuty は、コンテナがクリプトマイニングプロセや既知の悪意のあるエンドポイントへの接続の確立など、疑わしい動作をし始める攻撃シーケンスを検出できます。この保護プランを有効にする方法については、「ランタイムモニタリング」を参照してください。

EKS Protection または Runtime Monitoring を有効にしない場合、GuardDuty は個々の EKS Protection の検出結果タイプまたは Runtime Monitoring の検出結果タイプを生成できません。したがって、GuardDuty は関連する検出結果を含む複数ステージの攻撃シーケンスを検出できません。

Amazon S3 バケット内の攻撃シーケンスの検出

S3 Protection を有効にすると、GuardDuty は Amazon S3 バケット内のデータ侵害の試みを含む攻撃シーケンスを検出できます。S3 Protection を使用しない場合、GuardDuty は S3 バケットリソースポリシーが過度に寛容になったことを検出できます。S3 Protection を有効にすると、GuardDuty は S3 バケットが過度に寛容になった後に発生する可能性のあるデータ流出アクティビティを検出できるようになります。

S3 Protection が有効になっていない場合、GuardDuty は個々の S3 Protection の検出結果タイプ を生成できません。したがって、GuardDuty は関連する検出結果を含む複数ステージの攻撃シーケンスを検出できません。この保護プランを有効にする方法については、「S3 Protection」を参照してください。

Amazon ECS クラスターでの攻撃シーケンスの検出

GuardDuty は、悪意のあるプロセス、悪意のあるエンドポイントへの接続、ECS コンテナ内の暗号マイニング動作などの攻撃シーケンスを特定できます。GuardDuty は、ネットワークアクティビティ、プロセスランタイム動作、 AWS API アクティビティ間で多様なシグナルを相関させることで、個々の検出で見逃される可能性のある複雑な攻撃パターンを特定し、完全な攻撃ベクトルをマッピングできます。

GuardDuty は、これらの関連イベントを `AttackSequence:ECS/CompromisedCluster` と呼ばれる単一の「重大」な検出結果として表します。攻撃シーケンスの検出結果は、次の脅威シナリオを対象としています。

- 脆弱なサービスを実行しているコンテナの侵害

- 疑わしいツール、マルウェア、または暗号化プロセスの不正な実行
- 権限昇格の試み
- 疑わしいエンドポイントとの通信

> **Important**
>
> ECS の脅威検出を拡張するには、ECS インフラストラクチャタイプに応じて Fargate または EC2 のランタイムモニタリングが必要です。Runtime Monitoring は、Fargate インスタンスと EC2 インスタンスの両方で実行されている ECS コンテナ内の動作を監視します。マネージド EC2 インスタンスの ECS はサポートされていません。

Amazon EC2 インスタンスグループの攻撃シーケンスの検出

GuardDuty は、Auto Scaling グループ、IAM インスタンスプロファイル、起動テンプレート、CloudFormation スタック、AMIs、VPC IDs などの一般的な属性を共有する個々のインスタンスまたはインスタンスのグループに影響を与える攻撃シーケンスを特定できます。これらのインスタンスは、悪意のあるプロセス、悪意のあるエンドポイントへの接続、暗号マイニング、EC2 インスタンス認証情報の異常な使用などの疑わしい動作を示す可能性があります。

攻撃シーケンスの検出結果は、次の脅威シナリオを検出します。

- 脆弱なサービスを実行しているインスタンスの侵害
- 認証情報が発行された AWS アカウントの外部から IMDS 経由で取得した Amazon EC2 インスタンス認証情報の使用
- プロキシ、スキャン、サービス拒否などの攻撃のインフラストラクチャとして使用する
- 疑わしいツール、マルウェア、または暗号化プロセスの不正な実行
- 疑わしいエンドポイントとの通信

拡張脅威検出は、これらの脅威を特定するのに役立ちます。GuardDuty は、これらの関連イベントを `AttackSequence:EC2/CompromisedInstanceGroup` と呼ばれる単一の「重大」な検出結果として表します。

EC2 検出機能の延長脅威検出機能を強化するには、Runtime Monitoring を有効にします。CloudTrail とネットワークアクティビティをモニタリングする基本的な GuardDuty と、インスタンス内のプロセス動作とシステム呼び出しを監視する Runtime Monitoring の組み合わせにより、より包括的な脅

威検出が可能になります。この統合モニタリングにより、GuardDuty は、誤って設定された認証情報による不正アクセス、特権エスカレーションの試行、機密データへの不正アクセスなどの高度な攻撃シーケンスを検出できます。これらの多様なシグナルを相関させることで、GuardDuty は個々の検出で見逃される可能性のある複雑な攻撃パターンを特定し、完全な攻撃ベクトルをマッピングできます。

AI ワークロードを含む攻撃シーケンスの検出

AI 保護 アカウント内の AI ワークロードを脅かす攻撃シーケンスを GuardDuty が検出できるようにを有効にします。AI Protection は、Amazon Bedrock と Amazon SageMaker AI AWS CloudTrail のデータイベントを分析します。異常なモデル呼び出し、コスト収集攻撃、直接的なプロンプトインジェクション試行などのアクティビティを検出できます。

AI ワークロードからのシグナルをAttackSequence:IAM/CompromisedCredentials攻撃シーケンスに追加するには、AI Protection を有効にする必要があります。その後、GuardDuty は次の検出結果タイプを相関シグナルとして使用できます。

- Impact:IAMUser/AnomalousModelInvocation
- Impact:IAMUser/CostHarvesting
- Impact:IAMUser/PromptInjection.Direct

たとえば、脅威アクターは、侵害された可能性のある認証情報と同じ AWS 認証情報を使用して、異常な Amazon Bedrock または Amazon SageMaker AI モデルの呼び出しを行う場合があります。GuardDuty は、その呼び出しを、それらの認証情報で実行された他の疑わしいアクションと関連付けることができます。次に、それらを単一のAttackSequence:IAM/CompromisedCredentials攻撃シーケンスの検出結果として表します。単独では、AI Protection の検出結果は攻撃シーケンスを生成しません。

AI Protection が有効になっていない場合、GuardDuty は個々の を生成できませんAI Protection の検出結果タイプ。その結果、GuardDuty はこれらの検出結果を含む複数ステージの攻撃シーケンスを検出できません。この保護プランを有効にする方法については、「AI 保護」を参照してください。

GuardDuty コンソールの Extended Threat Detection

デフォルトでは、GuardDuty コンソールの Extended Threat Detection ページに ステータス が 有効 と表示されます。基礎的な脅威検出では、このステータスは GuardDuty が Amazon S3 API での IAM 権限検出活動を含む潜在的な攻撃シーケンスを検出し、その後の S3 コントロールプレーンの変更を検出できることを示しています。

GuardDuty コンソールで Extended Threat Detection ページにアクセスするには、次の手順に従います。

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開くことができます。
2. 左のナビゲーションペインで [Extended Threat Detection] を選択します。

   このページでは、Extended Threat Detection の対象となる脅威シナリオについて詳しく説明します。
3. [Extended Threat Detection] ページで、[関連する保護プラン] セクションを表示します。専用保護プランを有効にしてアカウントの脅威検出カバレッジを強化する場合は、その保護プランの設定オプションを選択します。

攻撃シーケンスの検出結果の理解と管理

攻撃シーケンスの検出結果は、アカウント内の他の GuardDuty の検出結果と同じです。[検出結果] ページは GuardDuty コンソールに表示できます。検出結果の表示と対処については、「GuardDuty コンソールの検出結果ページ」を参照してください。

他の GuardDuty の検出結果と同様に、攻撃シーケンスの検出結果も自動的に Amazon EventBridge に送信されます。設定に基づいて、攻撃シーケンスの検出結果は発行先 (Amazon S3 バケット) にもエクスポートされます。新しい発行先を設定するか、既存の発行先を更新するには、「生成された検出結果を Amazon S3 にエクスポートする」を参照してください。

その他のリソース

攻撃シーケンスの詳細については、以下のセクションを参照してください。

- Extended Threat Detection と攻撃シーケンスについて学習したら、「サンプルの検出結果」の手順に従って、サンプル攻撃シーケンスの検出結果タイプを生成できます。
- 攻撃シーケンスの検出結果タイプ についてはこちら。
- 検出結果を確認し、攻撃シーケンスの検出結果の詳細 に関連する検出結果の詳細を調べます。
- 検出結果の修復 の関連する影響を受けるリソースの手順に従って、攻撃シーケンスの検出結果タイプに優先順位を付け、対処します。

保護プランの設定

GuardDuty コンソールの Protection Plans ページは、すべての保護プラン設定を 1 つのページに統合します。保護プランを有効または無効にしたり、 AWS Organizations アカウントの自動有効化設定を設定したり、1 つの場所からメンバーアカウントの統計を表示したりできます。

概要

統合保護プランページは、設定のために個々の保護プランページに移動する必要性を置き換えます。コンソールナビゲーションには、1 つの Protection Plans リンクが表示されます。

このページには、次の順序で保護プランが表示されます。

1. Foundational GuardDuty – GuardDuty を有効または無効にし、 AWS Organizations アカウントの自動有効化を設定します。
2. S3 Protection – Amazon S3 データイベントをモニタリングして、データに対する潜在的な脅威がないか確認します。
3. ランタイムモニタリング – EKS、ECS、EC2 ワークロードのオペレーティングシステムレベルのイベントをモニタリングします。
4. EKS 監査ログ – クラスターに対する潜在的な脅威がないか Amazon EKS 監査ログをモニタリングします。
5. RDS Protection – データベースへの潜在的な脅威がないか、RDS ログインアクティビティをモニタリングします。
6. Lambda Protection – Lambda 関数のネットワークアクティビティをモニタリングして潜在的な脅威がないか調べます。

各保護プランには、次の情報が表示されます。

- 機能ステータス (有効または無効)
- AWS Organizations アカウントのステータスの自動有効化
- AWS Organizations アカウントメンバーの統計

すべての保護プランの設定

統合ページから保護プランを設定するには、次の手順を実行します。

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、保護プランを選択します。
3. すべての有効化の設定 を選択します。すべての保護プランコンテナは編集モードに切り替わります。
4. 一部またはすべての保護プランを変更します。プランごとに、機能をオンまたはオフに切り替え、組織アカウントの自動有効化設定を設定できます。
5. [すべて保存] を選択します。変更された各プランの機能名、ステータス、自動有効化値など、すべての変更をまとめた確認モーダルが表示されます。
6. 変更を確認し、確認して保存します。すべての変更はバッチで保存されます。

自動有効化オプション

各保護プランには、組織アカウントに対して次の自動有効化オプションが用意されています。

すべてのアカウントで を有効にする

組織内のすべてのアクティブな GuardDuty アカウントでこの機能を有効にします。これには、組織に参加する新しいアカウントが含まれます。

新しいアカウントで を有効にする

これは、新しいアカウントが組織に追加された場合に適用されます。

自動有効化しない

この機能は、組織アカウントに対して自動的に有効になるわけではありません。

Foundational GuardDuty コンテナの場合、自動有効化オプションは AWS Organizations特定の文言を使用します。

- すべての AWS Organizations アカウントで を有効にする – 内のすべてのアクティブなアカウントで GuardDuty を有効にします AWS Organizations。
- 新しい AWS Organizations アカウントで を有効にする – これは、新しいアカウントが に追加されるときに適用されます AWS Organizations。

Runtime Monitoring の設定

Runtime Monitoring コンテナには、次のサブ機能を備えた親子構造があります。

- ランタイムモニタリング (EKS)
- ランタイムモニタリング (ECS)
- ランタイムモニタリング (EC2)

Runtime Monitoring Infrastructure 設定は、展開可能なセクションに表示されます。この設定は、EKS、ECS、および EC2 のランタイムモニタリングが機能するために必要ですが、単独では保護を提供しません。

双方向制約

Runtime Monitoring のサブ機能とインフラストラクチャには、次の関係があります。

サブ機能を有効にする

無効になっている場合、ランタイムモニタリングインフラストラクチャを自動的に有効にします。

サブ機能の無効化

Runtime Monitoring インフラストラクチャには影響しません。

ランタイムモニタリングインフラストラクチャを有効にする

サブ機能には影響しません。

ランタイムモニタリングインフラストラクチャを無効にする

すべてのサブ機能を無効にします。

自動有効化設定では、次の制約が適用されます。

サブ機能の Auto-Enable を上げる

Runtime Monitoring Infrastructure は常にすべてのサブ機能の最大値に等しくなります。

サブ機能の自動有効化を下げる

Runtime Monitoring Infrastructure は、すべてのサブ機能の新しい最大値に再計算されます。

ランタイムモニタリングインフラストラクチャの自動有効化

サブ機能には影響しません。

Lower Runtime Monitoring インフラストラクチャの自動有効化

一致する新しい親値を超えるサブ機能をすべて下げます。

> **Note**
>
> いずれかのサブ機能に Runtime Monitoring インフラストラクチャとは異なる値 (トグルまたは自動有効化の不一致) がある場合、警告が表示されます。Runtime Monitoring Infrastructure のみを有効にすると、セキュリティエージェントの自動管理なしで手動モードになります。EKS、ECS、または EC2 のランタイムモニタリングを有効にして、ワークロードのセキュリティエージェントを自動的に管理します。

組織の統計

各保護プランコンテナには、機能が有効になっているアカウントの数を示すメンバーアカウントの統計が表示されます。組織統計がまだ利用できない場合 (たとえば、有効にしてから最初の 24 時間以内) 、ページには、統計が収集されていることを説明するメッセージが表示された保留中のステータスインジケータが表示されます。

アカウントページの統合

アカウントページの自動有効化セクションの編集ボタンは、レガシー設定モーダルを開く代わりに、保護プランページにリダイレクトします。

GuardDuty AI 保護

GuardDuty AI Protection は、Amazon GuardDuty の脅威検出を 上に構築された AI ワークロードに拡張します AWS。AI Protection を有効にすると、GuardDuty は Amazon Bedrock、Amazon Bedrock AgentCore、Amazon SageMaker AI AWS CloudTrail のデータイベント AWS CloudTrail を管理イベントとともに分析します。GuardDuty はこのデータを使用して、基盤モデルとそのモデルを呼び出すアプリケーションをターゲットとする疑わしいアクティビティを検出します。GuardDuty が潜在的な脅威を特定すると、1 つ以上の が生成されますAI Protection の検出結果タイプ。

AI Protection を有効にすると、GuardDuty は次のタイプの脅威を検出できます。

- 異常な IP アドレス、異常な API、異常なモデルからの呼び出しなど、ID またはアカウントの確立されたベースラインから逸脱する Amazon Bedrock または Amazon SageMaker AI モデルの異常な呼び出し。
- コスト収集攻撃。脅威アクターが計算コストの高い入力を Amazon Bedrock または Amazon SageMaker AI モデルに送信して、アカウントのトークン消費と運用コストを増大させます。
- 直接プロンプト挿入の試行。脅威アクターが悪意のあるプロンプトを作成して、基盤モデルが元の指示を無視します。この検出には Amazon Bedrock ガードレールが必要で、Amazon Bedrock ワークロードにのみ適用されます。

> **Note**
>
> GuardDuty が で生成できる検出結果タイプは、そのリージョンで利用可能な AI サービス AWS リージョン によって異なります。
>
> - Impact:IAMUser/AnomalousModelInvocation および Impact:IAMUser/CostHarvesting検出結果タイプには、Amazon Bedrock または Amazon SageMaker AI が必要です。Amazon Bedrock が利用できないリージョンでは、GuardDuty は Amazon SageMaker AI モデルの呼び出しからのみこれらの検出結果タイプを生成します。
> - Impact:IAMUser/PromptInjection.Direct 検出結果タイプには Amazon Bedrock ガードレールが必要です。Amazon Bedrock ガードレールがサポートされていないリージョンでは利用できません。
>
> 各検出結果タイプがサポートされているリージョンのリストについては、「」を参照してくださいリージョン固有機能の可用性。

> **Tip**
>
> AI Protection から最大限のセキュリティ価値を得るには、 AWS 組織内のすべてのアカウントでプロンプト攻撃に Amazon Bedrock ガードレールを適用します。AWS Organizations Amazon Bedrock ポリシーを使用して、これらのガードレール設定を一元管理します。これにより、一元的な制御と管理により、すべてのアカウントで統一された保護が可能になります。プロンプト攻撃ガードレールが適用されると、GuardDuty はプロンプトインジェクションの検出結果を作成します。詳細については、「Impact:IAMUser/PromptInjection.Direct」を参照してください。

AI ワークロードへの脅威を検出するには、GuardDuty アカウントで AI Protection を有効にする必要があります。AI Protection を有効にする方法については、マルチアカウント環境マルチアカウント環境で AI Protection を有効にするの場合は「」、または「」を参照してくださいスタンドアロンアカウントの AI Protection の有効化。

トピック

- AI Protection の仕組み
- AI Protection の料金
- AI ワークロードに対する追加の脅威検出
- マルチアカウント環境で AI Protection を有効にする
- スタンドアロンアカウントの AI Protection の有効化

AI Protection の仕組み

アカウントまたは組織全体で AI Protection を有効にすると、GuardDuty はモニタリング対象アカウントの AI サービスから AWS CloudTrail データイベントの収集を自動的に開始します。証跡の作成、データイベントのログ記録の有効化、AI アプリケーションの変更を行う必要はありません。

データイベントを収集するために、GuardDuty はモニタリング対象アカウントごとに AWS CloudTrail サービスにリンクされたチャネルを作成します。サービスにリンクされたチャネルは、Amazon Bedrock、Amazon Bedrock AgentCore、Amazon SageMaker AI、および関連する AI リソース AWS CloudTrail のデータイベントを GuardDuty にストリーミングして分析します。GuardDuty はこのチャネルを作成および管理するため:

- GuardDuty はチャネルのデータイベント設定を構成するため、アカウント所有者は変更できません。つまり、脅威の検出は、アカウント所有者が証跡を設定または維持することに依存しません。
- モニタリング対象の各アカウントは、チャネルが CloudTrail コンソール (設定、サービスにリンクされたチャネルの下) でアクティブであることを確認するか、 API オペレーションを AWS CloudTrail ListChannels呼び出します。

GuardDuty は、検出結果のタイプに応じて、2 つの検出方法を使用して AI Protection の検出結果を生成します。

- Impact:IAMUser/AnomalousModelInvocation および Impact:IAMUser/CostHarvesting検出結果タイプの場合、GuardDuty は異常検出機械学習 (ML) モデルを使用して、収集されたイベントを分析し、各 IAM ID および の通常のモデル呼び出しアクティビティのベースラインを確立します AWS アカウント。GuardDuty は、このベースラインから大幅に逸脱したアクティビティを検出すると、検出結果を生成します。
- Impact:IAMUser/PromptInjection.Direct 検出結果タイプの場合、GuardDuty は Amazon Bedrock ガードレールが結果 AWS CloudTrail のデータイベントに基づいてプロンプトを評価し、プロンプト攻撃を検出すると検出結果を生成します。

AWS CloudTrail データイベントの詳細については、「 AWS CloudTrail ユーザーガイド」の「データイベントのログ記録」を参照してください。

AI Protection の料金

GuardDuty AI Protection を有効にすると、GuardDuty は分析する AWS CloudTrail データイベントの量に対して課金し、GB 単位で測定されます。GuardDuty は、追加の CloudTrail 料金なしでサービスにリンクされたチャネルを作成します。これらのデータイベントを収集するコストは、AI Protection の使用コストに含まれます。また、GuardDuty およびその他の有効な保護プランの標準使用コストも引き続き発生します。現在の料金と例については、Amazon GuardDutyの料金」を参照してください。

AI ワークロードに対する追加の脅威検出

AI Protection プランに加えて、Amazon GuardDuty Foundational および Lambda Protection プランは、構築された AI ワークロードに対する脅威を保護および検出するのに役立つ検出を提供します AWS。

GuardDuty Foundational プランは AWS CloudTrail 、管理イベントをモニタリングして、Amazon Bedrock や Amazon SageMaker AI などの AWS サービスを使用して作成された AI ワークロードの疑わしいアクティビティや悪意のあるアクティビティを検出します。 Amazon SageMaker 例えば、GuardDuty は次のようなアクティビティを特定できます。

- Amazon Bedrock セキュリティガードレールの異常な削除
- データポイズニング攻撃につながる可能性のあるモデルトレーニングデータソースの変更
- 検出を回避する試みを示す可能性のある Amazon Bedrock モデル呼び出しのログ記録を無効にしました
- Amazon SageMaker AI での異常なノートブックインスタンスまたはトレーニングジョブの作成
- Amazon Bedrock、Amazon SageMaker AI、または Amazon EC2 インスタンス、Amazon EKS クラスター、または Amazon ECS タスクのセルフマネージド AI ワークロードで APIs を呼び出すために使用された可能性のある Amazon Elastic Compute Cloud 認証情報を抽出しました。

GuardDuty Lambda Protection は、Amazon Bedrock エージェントに関連する潜在的な脅威を検出するのに役立ちます。これには、暗号化などの疑わしいネットワークアクティビティや、悪意のあるコマンドサーバーやコントロールサーバーとの通信が含まれる場合があります。これらの脅威は、サプライチェーン攻撃や複雑なプロンプトによって引き起こされる可能性があります。

マルチアカウント環境で AI Protection を有効にする

マルチアカウント環境では、委任 GuardDuty 管理者アカウントのみが、組織内のメンバーアカウントの AI Protection を有効または無効にできます。メンバーアカウントは、自分のアカウントからこの設定を変更することはできません。委任 GuardDuty 管理者アカウントは、 を使用してメンバーアカウントを管理し AWS Organizations、組織に参加するすべての新しいアカウントの AI Protection を自動的に有効にすることを選択できます。マルチアカウント環境の詳細については、「Amazon GuardDuty での複数のアカウントの管理」を参照してください。

委任 GuardDuty 管理者アカウントの AI Protection の有効化

任意のアクセス方法を選択して、委任 GuardDuty 管理者アカウントの AI Protection を有効にします。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、保護プランを選択します。

3. すべての有効化の設定 を選択します。AI Protection では、AI Protection の現在の設定ステータスを表示できます。
4. 次のいずれかを行います:

   [すべてのアカウントについて有効にする] の使用

   - [すべてのアカウントについて有効にする] を選択します。これにより、 AWS 組織に参加する新しいアカウントを含め、組織内のすべてのアクティブな GuardDuty アカウントの保護プランが有効になります。
   - [保存] を選択します。

   [アカウントを手動で設定] の使用

   - 委任 GuardDuty 管理者アカウントでのみ保護プランを有効にするには、[アカウントを手動で設定] を選択します。
   - [委任 GuardDuty 管理者 (このアカウント)] セクションで [有効にする] を選択します。
   - [保存] を選択します。

API/CLI

ユーザー独自のリージョンレベルのディテクター ID を使用し、`features` オブジェクト `name` を `AI_PROTECTION` として、`status` を `ENABLED` または `DISABLED` として渡して、UpdateDetector API オペレーションを実行します。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

次の例では、委任 GuardDuty 管理者アカウントの AI Protection を有効にします。*12abc34d567e8fa901bc2d34e56789f0* をアカウントのディテクター ID に置き換え、*us-east-1* を AI Protection を有効にするリージョンに置き換えます。

```
aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --
region us-east-1 --features '[{"Name": "AI_PROTECTION", "Status": "ENABLED"}]'
```

AI Protection を無効にするには、 `ENABLED`を に置き換えます`DISABLED`。

既存のすべてのアクティブなメンバーアカウントの AI Protection の有効化

任意のアクセス方法を選択して、組織内のすべての既存のアクティブなメンバーアカウントの AI Protection を有効にします。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   必ず委任 GuardDuty 管理者アカウントの認証情報を使用してください。
2. ナビゲーションペインで、保護プランを選択します。
3. すべての有効化の設定 を選択します。AI Protection の「アクティブメンバーアカウント」セクションで「アクション」を選択します。
4. [アクション] ドロップダウンメニューから、[すべての既存のアクティブなメンバーアカウントについて有効にする] を選択します。
5. [保存] を選択します。

API/CLI

メンバーアカウントの AI Protection を選択的に有効または無効にするには、独自の`###### ID` を使用して UpdateMemberDetectors API オペレーションを実行します。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

次の の例では、単一のメンバーアカウントの AI Protection を有効にします。無効にするには、`ENABLED` を `DISABLED` に置き換えます。スペースで区切られたアカウント ID のリストを渡すこともできます。

```
aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0
 --account-ids 111122223333 --region us-east-1 --features '[{"Name":
 "AI_PROTECTION", "Status": "ENABLED"}]'
```

コマンドが正常に実行されると、空の のリストが返されますUnprocessedAccounts。アカウントのディテクター設定を変更する際に問題が発生した場合、レスポンスにはそのアカウント ID と問題の概要が一覧表示されます。

新しいメンバーアカウントの AI Protection の自動有効化

任意のアクセス方法を選択して、組織に参加する新しいアカウントの AI Protection を有効にします。

コンソールでは、委任 GuardDuty 管理者アカウントは、保護プランページまたはアカウントページを使用して、新しいメンバーアカウントの AI Protection を自動的に有効にできます。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   必ず委任 GuardDuty 管理者アカウントの認証情報を使用してください。

2. 次のいずれかを行います:

   - Protection Plans ページの使用:

     1. ナビゲーションペインで、保護プランを選択します。
     2. すべての有効化の設定 を選択します。
     3. AI Protection で、アカウントを手動で設定を選択し、新しいメンバーアカウントに対して自動有効化を選択します。このステップにより、新しいアカウントが組織に加わるたびに、GuardDuty によってそのアカウントの AI Protection が自動的に有効になります。この設定を変更できるのは、組織の委任 GuardDuty 管理者アカウントだけです。
     4. [保存] を選択します。

   - [Accounts] (アカウント) ページを使用する場合:

     1. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
     2. [アカウント] ページで、[自動有効化] 設定を選択します。
     3. 自動有効化設定の管理ウィンドウで、AI Protection で新しいアカウントの有効化を選択します。
     4. [保存] を選択します。

API/CLI

新しいメンバーアカウントの AI Protection を有効にするには、独自の`###### ID` を使用して UpdateOrganizationConfiguration API オペレーションを実行します。

アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

次の例では、組織に参加する新しいアカウントの AI Protection を有効にします。すべての新しいアカウントで有効にしない場合は、 AutoEnable を に設定しますNONE。

```
aws guardduty update-organization-configuration --detector-
id 12abc34d567e8fa901bc2d34e56789f0 --region us-east-1 --auto-enable --features
 '[{"Name": "AI_PROTECTION", "AutoEnable": "NEW"}]'
```

スタンドアロンアカウントの AI Protection の有効化

スタンドアロンアカウントは、特定のリージョンの AWS アカウントで保護プランを有効または無効にする決定を所有します。

アカウントが を通じて、 AWS Organizations または招待によって GuardDuty 管理者アカウントに関連付けられている場合、このセクションは適用されません。複数のアカウントの管理の詳細については、「マルチアカウント環境で AI Protection を有効にする」を参照してください。

AI Protection を有効にすると、GuardDuty はアカウントの CloudTrail から収集する Amazon Bedrock および Amazon SageMaker AI モデルの呼び出しアクティビティのモニタリングを開始します。

任意のアクセス方法を選択して、スタンドアロンアカウントで AI Protection を有効にします。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. 右上隅のリージョンセレクターから、AI Protection を有効にするリージョンを選択します。
3. ナビゲーションペインで、保護プランを選択します。
4. すべての有効化の設定 を選択します。AI Protection で、Enable を選択して AI Protection を有効にします。
5. すべて保存を選択し、確認して保存を選択します。

API/CLI

ユーザー独自のリージョンレベルのディテクター ID を使用し、`features` オブジェクト `name` を `AI_PROTECTION` として、`status` を `ENABLED` として渡して、UpdateDetector API オペレーションを実行します。

または、 AWS CLI を使用して AI Protection を有効にすることもできます。次のコマンドを実行し、*`12abc34d567e8fa901bc2d34e56789f0`* をアカウントのディテクター ID に置き換え、*`us-east-1`* を AI Protection を有効にするリージョンに置き換えます。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --
region us-east-1 --features '[{"Name": "AI_PROTECTION", "Status": "ENABLED"}]'
```

AI Protection を無効にするには、 `ENABLED`を に置き換えます`DISABLED`。

GuardDuty EKS Protection

> **Note**
>
> EKS Protection と他のすべての保護プランは、GuardDuty コンソールの 1 ページから設定できます。詳細については、「保護プランの設定」を参照してください。

EKS Protection は、 AWS 環境内の Amazon Elastic Kubernetes Service (Amazon EKS) クラスターで潜在的なセキュリティリスクを検出するのに役立ちます。例えば、クラスターからシークレットまたは AWS 認証情報を収集しようとする認証されていないアクターによって、誤って設定された EKS クラスターがいつアクセスされているかを検出するのに役立ちます。EKS Protection は、EKS 監査ログを使用して、ユーザーとアプリケーションのアクティビティを分析します。

EKS Protection を有効にすると、GuardDuty は自動的にAmazon EKS クラスターの潜在的なセキュリティ脅威のモニタリングを開始します。GuardDuty は独自の独立したストリームを使用して EKS Protection における EKS 監査ログ の収集と分析を行います。追加の設定は必要ありません。

EKS 監査ログのモニタリングに基づいて潜在する脅威が検出されると、セキュリティの検出結果が生成されます。EKS Protection を有効にしているときに GuardDuty で生成されうる検出結果タイプについては、「EKS Protection の検出結果タイプ」を参照してください。

> **Note**
>
> アカウント内の EKS 監査ログを表示するには (オプション)、Amazon EKS コントロールプレーンのロギングを設定して監査ログを CloudWatch Logs に送信できます。この設定は EKS Protection とは別のものであり、GuardDuty のセキュリティモニタリング機能には必要ありません。

30 日間の無料トライアル

- の AWS アカウント で AWS リージョン GuardDuty を初めて有効にすると、30 日間の無料トライアルが受けられます。この場合、EKS Protection も 30 日間無料トライアルに含まれているため一緒に有効になります。
- 既に GuardDuty を使用している場合に、初めて EKS Protection を有効にすると、このリージョンのアカウントに EKS Protection の 30 日間無料トライアルが適用されます。

- EKS Protection はいつでもどのリージョンでも無効にできます。
- 30 日間の無料トライアルでは、そのアカウントとリージョンの使用コストの見積もりを取得できます。30 日間無料トライアルが終了しても、EKS Protection が自動的に無効になることはありません。このリージョンのアカウントで使用コストが発生し始めます。詳細については、「使用状況のモニタリングとコストの見積もり」を参照してください。

EKS Protection を無効にすると、Amazon EKS リソースの EKS 監査ログのモニタリングと分析が直ちに停止します。

EKS Protection は、GuardDuty AWS リージョン が利用可能なすべての で使用できない場合があります。詳細については、「リージョン固有機能の可用性」を参照してください。

> **Note**
>
> EKS Runtime Monitoring は、Runtime Monitoring の一部として管理されます。詳細については、「GuardDuty Runtime Monitoring」を参照してください。

EKS Protection における EKS 監査ログ

EKS 監査ログは、ユーザー、Kubernetes API を使用するアプリケーション、コントロールプレーンからのアクティビティなど、Amazon EKS クラスター内のシーケンシャルアクションをキャプチャします。監査ログは、すべての Kubernetes クラスターのコンポーネントです。

詳細については、Kubernetes ドキュメントの「監査」を参照してください。

Amazon EKS を使用すると、EKS コントロールプレーンのログ記録機能を通じて、EKS 監査ログを Amazon CloudWatch Logs として取り込むことができます。GuardDuty は、Amazon EKS コントロールプレーンのログ記録を管理したり、アカウントの EKS 監査ログをアクセス可能な状態にしたりしません (Amazon EKS 向けに有効にしていない場合)。EKS 監査ログへのアクセスと保持期間を管理するには、Amazon EKS コントロールプレーンのログ機能を設定する必要があります。詳細については、「Amazon EKS ユーザーガイド」の「コントロールプレーンログの有効化と無効化」を参照してください。

マルチアカウント環境での EKS Protection の有効化

マルチアカウント環境で組織内のメンバーアカウントの EKS Protection 機能を有効または無効にできるのは、委任 GuardDuty 管理者アカウントのみです。GuardDuty メンバーアカウントからは、こ

の設定を変更できません。委任 GuardDuty 管理者アカウントでは、 AWS Organizationsを使用してメンバーアカウントを管理します。この委任 GuardDuty 管理者アカウントは、組織に加わるすべての新規アカウントに対して、EKS Protection の自動有効化を選択できます。マルチアカウント環境の詳細については、「Amazon GuardDuty での複数のアカウントの管理」を参照してください。

委任 GuardDuty 管理者アカウントでの EKS 監査ログのモニタリングの設定

任意のアクセス方法を選択して、委任 GuardDuty 管理者アカウントのために EKS 監査ログのモニタリングを設定します。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、保護プランを選択します。
3. すべての有効化の設定 を選択します。EKS 監査ログでは、EKS 監査ログモニタリングの現在の設定ステータスを表示できます。
4. 次のいずれかを行います。

   [すべてのアカウントについて有効にする] の使用

   - [すべてのアカウントについて有効にする] を選択します。これにより、 AWS 組織に参加する新しいアカウントを含め、組織内のすべてのアクティブな GuardDuty アカウントの保護プランが有効になります。
   - [保存] を選択します。

   [アカウントを手動で設定] の使用

   - 委任 GuardDuty 管理者アカウントでのみ保護プランを有効にするには、[アカウントを手動で設定] を選択します。
   - [委任 GuardDuty 管理者 (このアカウント)] セクションで [有効にする] を選択します。
   - [保存] を選択します。

API/CLI

ユーザー独自のリージョンレベルのディテクター ID を使用し、`features` オブジェクト `name` を `EKS_AUDIT_LOGS` として、`status` を `ENABLED` または `DISABLED` として渡して、updateDetector API オペレーションを実行します。

アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

次の AWS CLI コマンドを実行して、EKS 監査ログのモニタリングを有効または無効にできます。必ず委任 GuardDuty 管理者アカウントの有効な###### *ID* を使用してください。

> **Note**
>
> 次のサンプルコードは EKS 監査ログのモニタリングを有効にします。*12abc34d567e8fa901bc2d34e56789f0* をdetector-id委任 GuardDuty 管理者アカウントの に、*555555555555* AWS アカウント を委任 GuardDuty 管理者アカウントの に置き換えてください。

アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --
features '[{"Name": "EKS_AUDIT_LOGS", "Status": "ENABLED"}]'
```

EKS 監査ログモニタリングを無効にするには、ENABLED を DISABLED に置き換えます。

すべてのメンバーアカウントの EKS 監査ログのモニタリングを自動で有効にする

任意のアクセス方法を選択して、組織内の既存のメンバーアカウントの EKS 監査ログのモニタリングを有効にします。

Console

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   必ず委任 GuardDuty 管理者アカウントの認証情報を使用してください。

2. 次のいずれかを行います。

Protection Plans ページの使用

1. ナビゲーションペインで、保護プランを選択します。
2. すべての有効化の設定 を選択します。EKS 監査ログでは、組織内のアクティブなメンバーアカウントの EKS 監査ログモニタリングの現在のステータスを表示できます。
3. [すべてのアカウントについて有効にする] を選択します。このアクションにより、組織内の既存のアカウントと新しいアカウントの両方について EKS 監査ログモニタリングが自動的に有効になります。
4. すべて保存を選択し、確認して保存を選択します。

> **Note**
>
> メンバーアカウントの設定を更新するには、最大 24 時間かかる場合があります。

[アカウント] ページの使用

1. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
2. [アカウント] ページで、[招待によるアカウントの追加] の前に [自動有効化] の詳細設定を選択します。
3. [自動有効化の詳細設定を管理] ウィンドウで、[EKS 監査ログモニタリング] の下の [すべてのアカウントについて有効にする] を選択します。
4. [保存] を選択します。

[すべてのアカウントについて有効にする] オプションを使用できず、組織内の特定のアカウントのために EKS 監査ログモニタリング設定をカスタマイズする場合は、「メンバーアカウントの EKS 監査ログのモニタリングを選択的に有効または無効にする」を参照してください。

API/CLI

- メンバーアカウントの EKS 監査ログのモニタリングを選択的に有効または無効にするには、自分の`###### ID` を使用し、updateMemberDetectors API オペレーションを呼び出します。
- 次の例では、単一のメンバーアカウントで EKS 監査ログのモニタリングを有効にする方法を示しています。無効にするには、`ENABLED` を `DISABLED` に置き換えます。

アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-member-detectors --detector-
id 12abc34d567e8fa901bc2d34e56789f0 --account-ids 111122223333 --features
 '[{"name": "EKS_AUDIT_LOGS", "status": "ENABLED"}]'
```

> **Note**
>
> スペースで区切られたアカウント ID のリストを渡すこともできます。

- コードが正常に実行されると、UnprocessedAccounts の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

すべての既存のアクティブなメンバーアカウントのために EKS 監査ログモニタリングを有効にする

任意のアクセス方法を選択して、組織内のすべての既存のアクティブなメンバーアカウントの EKS 監査ログのモニタリングを有効にします。

Console

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   委任 GuardDuty 管理者アカウントの認証情報を使用してサインインします。
2. ナビゲーションペインで、保護プランを選択します。
3. すべての有効化の設定 を選択します。EKS 監査ログの「アクティブメンバーアカウント」セクションで、「アクション」を選択します。
4. [アクション] ドロップダウンメニューから、[すべての既存のアクティブなメンバーアカウントについて有効にする] を選択します。
5. [保存] を選択します。

API/CLI

- メンバーアカウントの EKS 監査ログのモニタリングを選択的に有効または無効にするには、自分の*###### ID* を使用し、updateMemberDetectors API オペレーションを呼び出します。
- 次の例では、単一のメンバーアカウントで EKS 監査ログのモニタリングを有効にする方法を示しています。無効にするには、ENABLED を DISABLED に置き換えます。

  アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-member-detectors --detector-
id 12abc34d567e8fa901bc2d34e56789f0 --account-ids 111122223333 --features
 '[{"name": "EKS_AUDIT_LOGS", "status": "ENABLED"}]'
```

  > **Note**
  >
  > スペースで区切られたアカウント ID のリストを渡すこともできます。

- コードが正常に実行されると、UnprocessedAccounts の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

新しいメンバーアカウントの EKS 監査ログのモニタリングを自動で有効にする

新しく追加されたメンバーアカウントは、GuardDuty が開始するマルウェアスキャンの設定を選択する前に GuardDuty を [有効にする] 必要があります。招待によって管理されるメンバーアカウントは、各自でアカウントに GuardDuty 実行型マルウェアスキャンを手動で設定できます。詳細については、「Step 3 - Accept an invitation」を参照してください。

任意のアクセス方法を選択して、組織に参加する新規アカウントの EKS 監査ログのモニタリングを有効にします。

Console

委任 GuardDuty 管理者アカウントは、[EKS 監査ログのモニタリング] または [アカウント] のいずれかのページを使用して、組織内の新しいメンバーアカウントのために EKS 監査ログのモニタリングを有効にすることができます。

新しいメンバーアカウントの EKS 監査ログのモニタリングを自動で有効にするには

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   必ず委任 GuardDuty 管理者アカウントの認証情報を使用してください。

2. 次のいずれかを行います。

   - Protection Plans ページの使用:

     1. ナビゲーションペインで、保護プランを選択します。
     2. すべての有効化の設定 を選択します。EKS 監査ログで、自動有効化の設定を行います。
     3. [アカウントを手動で設定] を選択します。
     4. [新しいメンバーアカウントについて自動的に有効にする] を選択します。このステップにより、新しいアカウントが組織に参加するたびに、そのアカウントのために EKS 監査ログモニタリングが自動的に有効になります。この設定を変更できるのは、組織の委任 GuardDuty 管理者アカウントだけです。
     5. [保存] を選択します。

   - [Accounts] (アカウント) ページを使用する場合:

     1. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
     2. [アカウント] ページで、[自動有効化] 設定を選択します。
     3. [自動有効化の詳細設定を管理] ウィンドウで、[EKS 監査ログモニタリング] の下の [新しいアカウントについて有効にする] を選択します。
     4. [保存] を選択します。

API/CLI

- 新しいアカウントの EKS 監査ログのモニタリングを選択的に有効または無効にするには、自分の`###### ID` を使用し、UpdateOrganizationConfiguration API オペレーションを実行します。
- 次の例では、組織に参加する新規メンバーの EKS 監査ログのモニタリングを有効にする方法を示しています。スペースで区切られたアカウント ID のリストを渡すこともできます。

  アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-organization-configuration --detector-
id 12abc34d567e8fa901bc2d34e56789f0 --auto-enable --features '[{"Name":
 "EKS_AUDIT_LOGS", "AutoEnable": "NEW"}]'
```

メンバーアカウントの EKS 監査ログのモニタリングを選択的に有効または無効にする

任意のアクセス方法を選択して、組織内の選択的メンバーアカウントの EKS 監査ログのモニタリングを有効または無効にします。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   必ず委任 GuardDuty 管理者アカウントの認証情報を使用してください。
2. ナビゲーションペインで、[Accounts] (アカウント) を選択します。

   [アカウント] ページで、[EKS 監査ログのモニタリング] 列でメンバーアカウントのステータスを確認します。
3. EKS 監査ログのモニタリングを有効または無効にするには

   EKS 監査ログのモニタリング用に設定するアカウントを選択します。一度に複数のアカウントを選択できます。[保護プランの編集] ドロップダウンで [EKS 監査ログのモニタリング] を選択し、適切なオプションを選択します。

API/CLI

メンバーアカウントの EKS 監査ログのモニタリングを選択的に有効または無効にするには、自身の`###### ID` を使用し、updateMemberDetectors API オペレーションを呼び出します。

次の例では、単一のメンバーアカウントで EKS 監査ログのモニタリングを有効にする方法を示しています。無効にするには、`ENABLED` を `DISABLED` に置き換えます。スペースで区切られたアカウント ID のリストを渡すこともできます。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0
 --accountids 111122223333 --features '[{"Name": "EKS_AUDIT_LOGS", "Status":
 "ENABLED"}]'
```

スタンドアロンアカウントの EKS Protection の有効化

スタンドアロンアカウントは、特定のリージョンの AWS アカウントで保護プランを有効または無効にする決定を所有します。

アカウントが を通じて AWS Organizations、または招待方法によって GuardDuty 管理者アカウントに関連付けられている場合、このセクションは適用されません。複数のアカウントの管理の詳細については、「マルチアカウント環境での EKS Protection の有効化」を参照してください。

EKS Protection を有効にすると、GuardDuty はアカウント内の Amazon EKS クラスターの EKS 監査ログのモニタリングを開始します。

任意のアクセス方法を選択して、スタンドアロンアカウントの EKS Protection を有効にします。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. 右上隅のリージョンセレクターから、EKS Protection を有効にするリージョンを選択します。
3. ナビゲーションペインで、保護プランを選択します。
4. すべての有効化の設定 を選択します。EKS 監査ログで、Enable を選択して EKS Protection を有効にします。
5. すべて保存を選択し、確認して保存を選択します。

API/CLI

- 委任 GuardDuty 管理者アカウントのリージョンレベルのディテクター ID を使用し、`features` オブジェクト名を `EKS_AUDIT_LOGS` として、ステータスを `ENABLED` として渡して、updateDetector API オペレーションを実行します。

  または、 AWS CLI コマンドを実行して EKS Protection を有効にすることもできます。次のコマンドを実行し、*`12abc34d567e8fa901bc2d34e56789f0`* をアカウントのディテクター ID に置き換え、*`us-east-1`* を EKS Protection を有効にするリージョンに置き換えます。

アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --
region us-east-1 --features [{"Name" : "EKS_AUDIT_LOGS", "Status" : "ENABLED"}]'
```

GuardDuty S3 Protection

> **Note**
>
> GuardDuty コンソールの 1 つのページから、S3 Protection と他のすべての保護プランを設定できます。詳細については、「保護プランの設定」を参照してください。

S3 Protection を使用すると、Amazon Simple Storage Service (Amazon S3) バケットでデータの引き出しや破棄などデータに潜むセキュリティリスクを検出できます。GuardDuty は Amazon S3 AWS CloudTrail のデータイベントをモニタリングします。これには、アカウント内のすべての Amazon S3 バケットでこれらのリスクを特定するためのオブジェクトレベルの API オペレーションが含まれます。

GuardDuty が S3 データイベントモニタリングに基づいて潜在する脅威を検出すると、セキュリティの検出結果が生成されます。S3 Protection を有効にしているときに GuardDuty で生成されうる検出結果タイプについては、「GuardDuty S3 Protection の検出結果タイプ」を参照してください。

基本的な脅威検出では、デフォルトで AWS CloudTrail 管理イベントをモニタリングして Amazon S3 リソースに潜む脅威を識別します。このデータソースは、環境でモニタリングするアクティビティの種類が違うため、S3 の AWS CloudTrail データイベントとは異なります。

GuardDuty がこの機能をサポートしているリージョンのアカウントで S3 Protection を有効にできます。これにより、そのアカウントとリージョンで S3 の CloudTrail データイベントをモニタリングできるようになります。S3 Protection を有効にすると、GuardDuty は Amazon S3 バケットを完全にモニタリングしたり、S3 バケットに保存されているデータへの疑わしいアクセスに関する検出結果を生成したりできます。

S3 Protection を使用するために、 AWS CloudTrailで S3 データイベントのログ記録を明示的に有効にしたり設定したりする必要はありません。

30 日間の無料トライアル

次に、アカウントに適用される 30 日間無料トライアルの仕組みについて説明します。

- AWS アカウント 新しいリージョンの で GuardDuty を初めて有効にすると、30 日間の無料トライアルが受けられます。この場合、S3 Protection も無料トライアルに含まれているため一緒に有効になります。

- 既に GuardDuty を使用している場合に、初めて S3 Protection を有効にすると、このリージョンのアカウントに S3 Protection の 30 日間無料トライアルが適用されます。
- S3 Protection はいつでもどのリージョンでも無効にできます。
- 30 日間の無料トライアルでは、そのアカウントとリージョンの使用コストの見積もりを取得できます。30 日間無料トライアルが終了しても、S3 Protection が自動的に無効になることはありません。このリージョンのアカウントで使用コストが発生し始めます。詳細については、「GuardDuty の使用状況のモニタリングとコストの見積もり」を参照してください。

AWS CloudTrail S3 のデータイベント

データプレーンオペレーションとして知られるデータイベントは、リソース上またはリソース内で実行したリソースオペレーションに関するインサイトを提供します。それらは、多くの場合、高ボリュームのアクティビティです。

次は、GuardDuty がモニタリングできる S3 の CloudTrail データイベントの例です。

- `GetObject` API オペレーション
- `PutObject` API オペレーション
- `ListObjects` API オペレーション
- `DeleteObject` API オペレーション

こうした API の詳細については、「Amazon Simple Storage Service API リファレンス」を参照してください。

GuardDuty が S3 の CloudTrail データイベントを使用する仕組み

S3 Protection を有効にすると、GuardDuty はいずれかの S3 バケットからの S3 の CloudTrail データイベントの分析を開始し、悪意のあるアクティビティや疑わしいアクティビティがないかモニタリングします。詳細については、「AWS CloudTrail 管理イベント」を参照してください。

未認証のユーザーから S3 オブジェクトへのアクセスがあった場合、その S3 オブジェクトは一般公開されていることになります。このため、GuardDuty はそのようなリクエストを処理しません。GuardDuty は、有効な IAM (AWS Identity and Access Management) または AWS STS (AWS Security Token Service) 認証情報を使用して S3 オブジェクトに対して行われたリクエストを処理します。

> **メモ**
>
> S3 Protection を有効にすると、GuardDuty は自身が有効にされたのと同じリージョンにある Amazon S3 バケットでデータイベントをモニタリングします。

特定のリージョンのアカウントで S3 Protection を無効にすると、GuardDuty は S3 バケットに保存されているデータに関して S3 データイベントをモニタリングしなくなります。GuardDuty は、そのリージョンのアカウントに対して S3 Protection 検出結果タイプを生成しなくなります。

攻撃シーケンスに S3 の CloudTrail データイベントを使用する GuardDuty

GuardDuty Extended Threat Detection は、アカウントの基本的なデータソース、 AWS リソース、タイムラインにまたがるマルチステージ攻撃シーケンスを検出します。GuardDuty は、アカウント内で最近発生した、または進行中の疑わしいアクティビティを示す一連のイベントを観察すると、関連する攻撃シーケンスの検出結果を生成します。

デフォルトでは、GuardDuty を有効にすると、Extended Threat Detection もアカウントで有効になります。この機能は、追加料金なしで CloudTrail 管理イベントに関連する脅威シナリオをカバーします。ただし、Extended Threat Detection を最大限に活用するために、GuardDuty で S3 Protection を有効にして、S3 の CloudTrail データイベントに関連する脅威シナリオをカバーすることをお勧めします。

S3 Protection を有効にすると、GuardDuty は、Amazon S3 リソースが関与する可能性のあるデータ侵害や破壊などの攻撃シーケンスの脅威シナリオを自動的にカバーします。

マルチアカウント環境での S3 Protection の有効化

マルチアカウント環境では、委任 GuardDuty 管理者アカウントのみが、 AWS 組織内のメンバーアカウントの S3 Protection を設定 (有効化または無効化) するオプションがあります。GuardDuty メンバーアカウントからは、この設定を変更できません。委任 GuardDuty 管理者アカウントは、 を使用してメンバーアカウントを管理します AWS Organizations。委任 GuardDuty 管理者アカウントは、組織内のすべてのアカウントまたは新しいアカウントのみで S3 Protection を自動的に有効にするか、あるいはどのアカウントでも S3 Protection を自動的に有効にしないかを選択できます。詳細については、「 AWS Organizationsを使用したアカウントの管理」を参照してください。

委任 GuardDuty 管理者アカウントでの S3 Protection の有効化

任意のアクセス方法を選択して、委任 GuardDuty 管理者アカウントに対して S3 Protection を有効にします。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[S3 Protection] を選択します。
3. [S3 Protection] ページで、[編集] を選択します。
4. 次のいずれかを行います。

   [すべてのアカウントについて有効にする] の使用

   - [すべてのアカウントについて有効にする] を選択します。これにより、 AWS 組織に参加する新しいアカウントを含め、組織内のすべてのアクティブな GuardDuty アカウントの保護プランが有効になります。
   - [保存] を選択します。

   [アカウントを手動で設定] の使用

   - 委任 GuardDuty 管理者アカウントでのみ保護プランを有効にするには、[アカウントを手動で設定] を選択します。
   - [委任 GuardDuty 管理者 (このアカウント)] セクションで [有効にする] を選択します。
   - [保存] を選択します。

API/CLI

現在のリージョンの委任 GuardDuty 管理者アカウントのディテクター ID を使用し、`features` オブジェクト `name` を `S3_DATA_EVENTS` として、`status` を `ENABLED` として渡すことで updateDetector を実行します。

または、 を使用して S3 Protection を設定することもできます AWS Command Line Interface。次のコマンドを実行し、*`12abc34d567e8fa901bc2d34e56789f0`* を現在のリージョンの委任 GuardDuty 理者アカウントのディテクター ID に置き換えます。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --
features '[{"Name": "S3_DATA_EVENTS", "Status": "ENABLED"}]'
```

組織内のすべてのメンバーアカウントのために S3 Protection を自動的に有効にする

任意のアクセス方法を選択して、委任 GuardDuty 管理者アカウントに対して S3 Protection を有効にします。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   管理者アカウントを使用してサインインします。

2. 次のいずれかを行います。

   [S3 Protection] ページの使用

   1. ナビゲーションペインで、[S3 Protection] を選択します。
   2. [すべてのアカウントについて有効にする] を選択します。このアクションにより、組織内の既存のアカウントと新しいアカウントの両方について S3 Protection が自動的に有効になります。
   3. [保存] を選択します。

      > **Note**
      > メンバーアカウントの設定を更新するには、最大 24 時間かかる場合があります。

   [アカウント] ページの使用

   1. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
   2. [アカウント] ページで、[招待によるアカウントの追加] の前に [自動有効化] の詳細設定を選択します。

3. [自動有効化の詳細設定を管理] ウィンドウで、[S3 Protection] の下の [すべてのアカウントについて有効にする] を選択します。
4. [保存] を選択します。

[すべてのアカウントについて有効にする] オプションを使用できない場合は、「メンバーアカウントで S3 Protection を選択的に有効にする」を参照してください。

API/CLI

- メンバーアカウントの S3 Protection を選択的に有効にするには、ユーザー独自の*###### ID* を使用して updateMemberDetectors API オペレーションを呼び出します。
- 次の例では、単一のメンバーアカウントで S3 Protection を有効にする方法を示しています。必ず *12abc34d567e8fa901bc2d34e56789f0* を委任 GuardDuty 管理者アカウントの `detector-id`、および *111122223333* に置き換えてください。

  アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-member-detectors --detector-
id 12abc34d567e8fa901bc2d34e56789f0 --account-ids 111122223333 --features
 '[{"name": "S3_DATA_EVENTS", "status": "ENABLED"}]'
```

> **Note**
>
> スペースで区切られたアカウント ID のリストを渡すこともできます。

- コードが正常に実行されると、`UnprocessedAccounts` の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

すべての既存のアクティブなメンバーアカウントのために S3 Protection を有効にする

任意のアクセス方法を選択して、組織内のすべての既存のアクティブなメンバーアカウントのために S3 Protection を有効にします。

Console

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   委任 GuardDuty 管理者アカウントの認証情報を使用してサインインします。
2. ナビゲーションペインで、[S3 Protection] を選択します。
3. [S3 Protection] ページでは、設定の現在のステータスを表示できます。[アクティブなメンバーアカウント] セクションで、[アクション] を選択します。
4. [アクション] ドロップダウンメニューから、[すべての既存のアクティブなメンバーアカウントについて有効にする] を選択します。
5. [確認] を選択します。

API/CLI

- メンバーアカウントの S3 Protection を選択的に有効にするには、ユーザー独自の*###### ID* を使用して updateMemberDetectors API オペレーションを呼び出します。
- 次の例では、単一のメンバーアカウントで S3 Protection を有効にする方法を示しています。必ず *12abc34d567e8fa901bc2d34e56789f0* を委任 GuardDuty 管理者アカウントの `detector-id`、および *111122223333* に置き換えてください。

  アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

  ```
  aws guardduty update-member-detectors --detector-
  id 12abc34d567e8fa901bc2d34e56789f0 --account-ids 111122223333 --features
   '[{"name": "S3_DATA_EVENTS", "status": "ENABLED"}]'
  ```

  > **Note**
  >
  > スペースで区切られたアカウント ID のリストを渡すこともできます。

- コードが正常に実行されると、`UnprocessedAccounts` の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

新しいメンバーアカウントの S3 Protection を自動で有効にする

任意のアクセス方法を選択して、組織に参加する新しいアカウントのために S3 Protection を有効にします。

Console

委任 GuardDuty 管理者アカウントは、[S3 Protection] または [アカウント] ページのいずれかを使用して、コンソールで組織の新しいメンバーアカウントに対して有効にできます。

新しいメンバーアカウントの S3 Protection を自動で有効にするには

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   必ず委任 GuardDuty 管理者アカウントの認証情報を使用してください。
2. 次のいずれかを行います。
   - [S3 Protection] ページの使用:
     1. ナビゲーションペインで、[S3 Protection] を選択します。
     2. [S3 Protection] ページで、[編集] を選択します。
     3. [アカウントを手動で設定] を選択します。
     4. [新しいメンバーアカウントについて自動的に有効にする] を選択します。このステップにより、新しいアカウントが組織に参加するたびに、そのアカウントのために S3 Protection が自動的に有効になります。この設定を変更できるのは、組織の委任 GuardDuty 管理者アカウントだけです。
     5. [保存] を選択します。
   - [Accounts] (アカウント) ページを使用する場合:
     1. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
     2. [アカウント] ページで、[自動有効化] 設定を選択します。
     3. [自動有効化の詳細設定を管理] ウィンドウで、[S3 Protection] の下の [新しいアカウントについて有効にする] を選択します。
     4. [保存] を選択します。

API/CLI

- メンバーアカウントの S3 Protection を選択的に有効にするには、ユーザー独自の`###### ID` を使用して UpdateOrganizationConfiguration API オペレーションを呼び出します。
- 次の例では、単一のメンバーアカウントで S3 Protection を有効にする方法を示しています。組織に参加する新しいアカウント (NEW) もしくはすべてのアカウント (ALL) のために、そのリージョンで保護プランを自動的に有効または無効にするか、または組織内のどのアカウントのためにも自動的に有効または無効にしない (NONE) 詳細設定を行います。詳細については、「autoEnableOrganizationMembers」を参照してください。必要に応じて、NEW を ALL または NONE に置き換える必要がある場合があります。

  アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

  ```
  aws guardduty update-organization-configuration --detector-
  id 12abc34d567e8fa901bc2d34e56789f0 --auto-enable --features '[{"Name":
   "S3_DATA_EVENTS", "autoEnable": "NEW"}]'
  ```

- コードが正常に実行されると、`UnprocessedAccounts` の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

メンバーアカウントで S3 Protection を選択的に有効にする

任意のアクセス方法を選択して、メンバーアカウントに対して S3 Protection を選択的に有効にします。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   必ず委任 GuardDuty 管理者アカウントの認証情報を使用してください。

2. ナビゲーションペインで、[Accounts] (アカウント) を選択します。

   [アカウント] ページで、[S3 Protection] 列でメンバーアカウントのステータスを確認します。

3. S3 Protection を選択的に有効にするには

   S3 Protection を有効にするアカウントを選択します。一度に複数のアカウントを選択できます。[保護プランの編集] ドロップダウンメニューで、[S3Pro] を選択し、適切なオプションを選択します。

API/CLI

メンバーアカウントの S3 Protection を選択的に有効にするには、自身のディテクター ID を使用し、updateMemberDetectors API オペレーションを実行します。次の例では、単一のメンバーアカウントで S3 Protection を有効にする方法を示しています。無効にするには、`true` を `false` に置き換えます。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
 aws guardduty update-member-detectors --detector-
id 12abc34d567e8fa901bc2d34e56789f0 --account-ids 123456789012 --features
 '[{"Name" : "S3_DATA_EVENTS", "Status" : "ENABLED"}]'
```

> **Note**
>
> スペースで区切られたアカウント ID のリストを渡すこともできます。

コードが正常に実行されると、`UnprocessedAccounts` の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

> **Note**
>
> スクリプトを使用して新しいアカウントをオンボーディングし、新しいアカウントで S3 Protection を無効にする場合は、このトピックで説明されているように、オプションの `dataSources` オブジェクトで createDetector API オペレーションを変更できます。

スタンドアロンアカウントの S3 Protection の有効化

スタンドアロンアカウントは、特定の の で保護プランを有効または無効にする決定を所有 AWS アカウント します AWS リージョン。

アカウントが を通じて AWS Organizations、または招待方法によって GuardDuty 管理者アカウントに関連付けられている場合、このセクションはアカウントには適用されません。詳細については、「マルチアカウント環境での S3 Protection の有効化」を参照してください。

S3 Protection を有効にすると、GuardDuty はアカウント内の S3 バケット AWS CloudTrail のデータイベントのモニタリングを開始します。

任意のアクセス方法を選択して、スタンドアロンアカウントのために S3 Protection を設定します。

Console

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. 右上隅のリージョンセレクターから、S3 Protection を有効にするリージョンを選択します。
3. ナビゲーションペインで、[S3 Protection] を選択します。
4. [S3 Protection] ページには、アカウントの S3 Protection の現在のステータスが表示されます。[有効にする] または [無効にする] を選択すると、いつでも S3 Protection を有効または無効にできます。
5. [確認] を選択して、選択を確定します。

API/CLI

現在のリージョンの有効なディテクター ID を使用し、S3 Protection を有効にするように `ENABLED` に設定された `features` オブジェクト `name` を `S3_DATA_EVENTS` として渡して updateDetector を実行します。

> **Note**
>
> アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

または、 を使用することもできます AWS Command Line Interface。S3 Protection を有効にするには、次のコマンドを実行し、*12abc34d567e8fa901bc2d34e56789f0* をアカウントのディテクター ID に、*us-east-1* を S3 Protection を有効にするリージョンにそれぞれ置き換えます。

```
aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --
region us-east-1 --features '[{"Name" : "S3_DATA_EVENTS", "Status" : "ENABLED"}]'
```

GuardDuty Runtime Monitoring

> **Note**
>
> Runtime Monitoring は、GuardDuty コンソールの 1 つのページから他のすべての保護プランとともに設定できます。詳細については、「保護プランの設定」を参照してください。

Runtime Monitoring は、オペレーティングシステムレベル、ネットワーク、ファイルイベントを監視および分析し、環境内の特定の AWS ワークロードで潜在的な脅威を検出するのに役立ちます。

Runtime Monitoring でサポートされている AWS リソース – Runtime Monitoring は、Amazon Elastic Kubernetes Service (Amazon EKS)、Amazon Elastic Container Service (Amazon ECS) on AWS Fargate、Amazon Elastic Compute Cloud (Amazon EC2) リソースの脅威検出カバレッジを提供します。

「 AWS Fargate」で実行されている Amazon EKS クラスターはサポートされていません。

このドキュメントおよび Runtime Monitoring に関連する他のセクションでは、GuardDuty はリソースタイプという用語を使用して、Amazon EKS、Fargate Amazon ECS、Amazon EC2 リソースを指します。

Runtime Monitoring では、GuardDuty セキュリティエージェントを使用して、ファイルアクセス、プロセス実行、コマンドライン引数、ネットワーク接続などのランタイムアクティビティを可視化します。潜在的な脅威をモニタリングするリソースタイプごとに、その特定のリソースタイプのセキュリティエージェントを自動または手動で管理できます (Fargate (Amazon ECS のみ) を除く)。セキュリティエージェントの自動管理は、GuardDuty がユーザーに代わってセキュリティエージェントをインストールおよび更新することを許可することを意味します。一方、リソースのセキュリティエージェントを手動で管理する場合、必要に応じてセキュリティエージェントをインストールして更新する責任がユーザーにあります。

この拡張機能により、GuardDutyは、個々のワークロードやインスタンスで実行されているアプリケーションやデータを標的にする可能性のある潜在的な脅威を特定して対応するのに役立ちます。例えば、脅威は、脆弱なウェブアプリケーションを実行している 1 つのコンテナを危険にさらすことから始まる可能性があります。このウェブアプリケーションには、基盤となるコンテナとワークロードへのアクセス権限が存在する可能性があります。この場合、認証情報が正しく設定されていないと、アカウントとその中に保存されているデータへのアクセスを制御できない可能性があります。

個々のコンテナとワークロードのランタイムイベントを分析することで、GuardDuty はコンテナおよび関連する AWS 認証情報の侵害を初期段階で特定し、特権、疑わしい API リクエスト、環境内のデータへの悪意のあるアクセスをエスカレートしようとする試みを検出できます。

内容

- 仕組み
- Runtime Monitoring の 30 日間の無料トライアルの仕組み
- Runtime Monitoring を有効にする前提条件
- GuardDuty Runtime Monitoring の有効化
- GuardDuty セキュリティエージェントの管理
- ランタイムカバレッジ統計の確認と問題のトラブルシューティング
- CPU とメモリモニタリングの設定
- Runtime Monitoring で共有 VPC を使用する
- GuardDuty 自動セキュリティエージェントでの Infrastructure as Code (IaC) の使用
- GuardDuty が使用する収集されたランタイムイベントタイプ
- Amazon ECR リポジトリホスティング GuardDuty エージェント
- 同じ基盤となるホスト上の 2 つのセキュリティエージェント
- GuardDuty での EKS Runtime Monitoring
- GuardDuty セキュリティエージェントのリリースバージョン
- Runtime Monitoring でのリソースの無効化、アンインストール、クリーンアップ

仕組み

Runtime Monitoring は、軽量のセキュリティエージェントを使用して、Fargate ワークロード上の Amazon EC2、Amazon EKS、Amazon ECS のランタイム動作を監視し、脅威を検出します。エンドツーエンドで次のように動作します。

1. エージェントは リソースにデプロイされます。GuardDuty はエージェントをデプロイして更新することも (自動エージェント設定) 、自分でデプロイして管理することもできます (手動エージェント設定) 。ステップはリソースタイプによって異なります。
2. エージェントはランタイムイベントを収集します。モニタリング対象のリソースごとに、エージェントはプロセスの実行、ファイルアクセス、ネットワーク接続などのオペレーティングシステムレベルのアクティビティを監視します。

3. エージェントは GuardDuty にイベントを送信します。エージェントはイベントを GuardDuty バックエンドに配信します。データは AWS ネットワーク内に残り、GuardDuty はコンピューティングリソースから GuardDuty に送信されたイベントに対して課金しません。詳細については、「セキュリティエージェントが GuardDuty に接続する方法」を参照してください。
4. GuardDuty はイベントを分析し、結果を生成します。特権エスカレーションや既知の悪意のあるホストとの通信などの疑わしい動作を検出すると、GuardDuty は検出結果を生成します。GuardDuty が生成するup-to-date検出結果のリストを確認するには、「」を参照してくださいGuardDuty Runtime Monitoring の検出結果タイプ。

エージェントは定義された CPU とメモリの制限内で実行されるため、リソースの消費量は予測可能で、同じホストで実行されているワークロードのパフォーマンスや可用性を低下させません。これらの制限については、「」を参照してくださいRuntime Monitoring を有効にする前提条件。

内容

- Runtime Monitoring の有効化
- セキュリティエージェントの Amazon EKS へのデプロイ
- セキュリティエージェントの Amazon EC2 へのデプロイ
- Fargate での Amazon ECS へのセキュリティエージェントのデプロイ
- セキュリティエージェントが GuardDuty に接続する方法
- Runtime Monitoring カバレッジの追跡

Runtime Monitoring の有効化

Runtime Monitoring の有効化は 2 ステップのプロセスです。

1. アカウントの Runtime Monitoring を有効にします。その後、GuardDuty は Amazon EC2 インスタンス、Amazon ECS クラスター、Amazon EKS ワークロードからのランタイムイベントを受け入れます。
2. モニタリングするリソースの GuardDuty セキュリティエージェントを管理します。リソースタイプに基づいて、次のことができます。
   - 自動エージェント設定を使用する – GuardDuty はエージェントをデプロイし、その接続を確立します。
   - エージェントを手動で管理する – VPC エンドポイントを作成し、エージェントのインストールと更新を管理します。

エージェントの管理は、リソースタイプによって異なります。詳細については、以下を参照してください。

- セキュリティエージェントの Amazon EKS へのデプロイ
- セキュリティエージェントの Amazon EC2 へのデプロイ
- Fargate での Amazon ECS へのセキュリティエージェントのデプロイ

GuardDuty セキュリティエージェントが Amazon EC2 インスタンスにデプロイされ、そこからランタイムイベントを受信した場合、GuardDuty はそのインスタンスからの VPC フローログの分析 AWS アカウント に対して に課金しません。これにより、2 倍の使用コストを回避できます。

セキュリティエージェントの Amazon EKS へのデプロイ

GuardDuty Runtime Monitoring は、GuardDuty セキュリティエージェントをデプロイすることで、Amazon EKS クラスターの脅威をモニタリングします。Amazon EKS では、GuardDuty はエージェントを EKS アドオン `aws-guardduty-agent`としてデプロイします。

> **注意事項**
>
> Runtime Monitoring は、Amazon EC2 インスタンスと Amazon EKS Auto Mode で実行されている Amazon EKS クラスターをサポートします。
> Runtime Monitoring は、Amazon EKS Hybrid Nodes を使用する Amazon EKS クラスター、および AWS Fargateで実行されているクラスターをサポートしていません。
> これらの Amazon EKS の詳細については、「Amazon EKS ユーザーガイド」の「Amazon EKS とは」を参照してください。

GuardDuty でエージェントを自動的に管理することも (推奨) 、手動で管理することもできます。

自動エージェント設定 (推奨)

GuardDuty は、EKS クラスターのセキュリティエージェントへのデプロイと更新を管理します。GuardDuty コンソールでは、この設定は Runtime Monitoring (EKS) として表示されます。

セキュリティエージェントが GuardDuty に接続する方法については、「」を参照してくださいセキュリティエージェントが GuardDuty に接続する方法。

セキュリティエージェントを有効にした場合の影響

GuardDuty はセキュリティエージェントを EKS クラスターにデプロイし、Amazon Elastic Kubernetes Service (Amazon EKS) はノードへのデプロイを調整します。

モニタリングスコープ

デフォルトでは、GuardDuty は、作成した新しいクラスターを含め、アカウント内のすべての EKS クラスターのセキュリティエージェントを管理します。GuardDuty が特定のクラスターでエージェントを管理できないようにするには、タグ `GuardDutyManaged:false` をそのクラスターに追加します。

EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「Amazon EKS リソースのタグ付け」を参照してください。

考慮事項

- 自動エージェント設定がオフの場合、GuardDuty は でタグ付けした EKS クラスターでのみセキュリティエージェントを管理しますGuardDutyManagedtrue。
- `GuardDutyManaged:false` タグを追加すると、GuardDuty によるセキュリティエージェントの管理は停止しますが、インストール済みのエージェントはアンインストールされません。
- サービスコントロールポリシーまたは IAM ポリシーを使用して、タグを変更できるユーザーを制御してGuardDutyManagedタグを保護します。詳細については、「AWS Organizations ユーザーガイド」の「サービスコントロールポリシー (SCP)」または「IAM ユーザーガイド」の「AWS のリソースへのアクセスの制御」を参照してください。

手動エージェント設定

このアプローチは、GuardDuty セキュリティエージェントをすべての EKS クラスターに手動でデプロイおよび管理する場合に使用します。アカウントで Runtime Monitoring が有効になっていることを確認します。Runtime Monitoring が有効になっていない場合、GuardDuty セキュリティエージェントが期待どおりに動作しない可能性があります。

このアプローチを使用した場合の影響

EKS クラスター内の GuardDuty セキュリティエージェントのデプロイは、すべてのアカウントで調整する必要があります。最新のセキュリティ機能を取得するには、GuardDuty が新しいバージョンをリリースしたときにエージェントも更新する必要があります。EKS のエージェントバージョンの詳細については、「Amazon EKS リソースの GuardDuty セキュリティエージェントバージョン」を参照してください。

考慮事項

新しいクラスターやワークロードが継続的にデプロイされるにつれて、カバレッジのギャップをモニタリングして対処しながら、安全なデータフローをサポートする必要があります。

関連トピック

- セキュリティエージェントが GuardDuty に接続する方法
- Runtime Monitoring カバレッジの追跡

セキュリティエージェントの Amazon EC2 へのデプロイ

GuardDuty Runtime Monitoring は、GuardDuty セキュリティエージェントをデプロイすることで、Amazon EC2 インスタンスの脅威をモニタリングします。

> **Note**
>
> Runtime Monitoring は、Amazon EC2 インスタンスで実行されている Amazon ECS タスクからランタイムイベントも収集します。

GuardDuty でエージェントを自動的に管理することも (推奨) 、手動で管理することもできます。

自動エージェント設定 (推奨)

GuardDuty は、Amazon EC2 インスタンスのセキュリティエージェントのデプロイと更新を管理します。GuardDuty コンソールでは、この設定は Runtime Monitoring (EC2) として表示されます。

セキュリティエージェントが GuardDuty に接続する方法については、「」を参照してくださいセキュリティエージェントが GuardDuty に接続する方法。

セキュリティエージェントを有効にした場合の影響

自動エージェント設定を使用すると、GuardDuty は SSM 管理の Amazon EC2 インスタンス (https://console.aws.amazon.com/systems-manager/ コンソールの Fleet Manager の下を参照) に 1 つの SSM 関連付けを作成し、それを使用してエージェントをインストールおよび更新します。

モニタリングスコープ

デフォルトでは、GuardDuty はアカウント内のすべての Amazon EC2 インスタンスのセキュリティエージェントを管理します。GuardDuty が特定のインスタンスでエージェントを管理できないようにするには、そのインスタンスにタグ `GuardDutyManaged:false` を追加します。

考慮事項

- 自動エージェント設定がオフの場合、GuardDuty は でタグ付けした Amazon EC2 インスタンスでのみセキュリティエージェントを管理しますGuardDutyManagedtrue。
- GuardDutyManaged:false タグを追加すると、GuardDuty によるセキュリティエージェントの管理は停止しますが、インストール済みのエージェントはアンインストールされません。
- サービスコントロールポリシーまたは IAM ポリシーを使用して、タグを変更できるユーザーを制御してGuardDutyManagedタグを保護します。詳細については、「AWS Organizations ユーザーガイド」の「サービスコントロールポリシー (SCP)」または「IAM ユーザーガイド」の「AWS のリソースへのアクセスの制御」を参照してください。

手動エージェント設定

Amazon EC2 のセキュリティエージェントを手動で管理する方法は 2 つあります。

- SSM マネージド Amazon EC2 インスタンスの場合、 で GuardDuty マネージドドキュメントを使用してセキュリティエージェント AWS Systems Manager をインストールできます。

> **Note**
>
> このオプションを使用するには、起動する新しい Amazon EC2 インスタンスごとに SSM が有効になっていることを確認します。

- Amazon EC2 インスタンスが SSM マネージドであるかどうかに関係なく、RPM パッケージマネージャー (RPM) スクリプトを使用して、Amazon EC2 インスタンスにセキュリティエージェントをインストールします。

関連トピック

- セキュリティエージェントが GuardDuty に接続する方法
- Runtime Monitoring カバレッジの追跡

Fargate での Amazon ECS へのセキュリティエージェントのデプロイ

GuardDuty Runtime Monitoring は、GuardDuty セキュリティエージェントをデプロイすることで、Fargate 上の Amazon ECS クラスターの脅威をモニタリングします。

> **Note**
>
> Runtime Monitoring は、Amazon ECS マネージドインスタンスで実行されているアプリケーションをサポートしていません。

GuardDuty はエージェントを自動的に管理します。手動管理は Fargate では使用できません。GuardDuty コンソールでは、この設定は Runtime Monitoring (ECS) として表示されます。

自動エージェント設定 (推奨)

GuardDuty は、Amazon ECS クラスター内の新しい Fargate タスクに対するセキュリティエージェントのデプロイと更新を管理します。

セキュリティエージェントが GuardDuty に接続する方法については、「」を参照してくださいセキュリティエージェントが GuardDuty に接続する方法。

セキュリティエージェントを有効にした場合の影響

新しい Fargate タスクごとに、GuardDuty はセキュリティエージェントを実行するサイドカーコンテナをアタッチし、タスク内のコンテナのランタイムイベントを収集します。

サイドカーイメージは Amazon S3 のレイヤーとともに Amazon Elastic Container Registry (Amazon ECR) に保存され、タスクの開始時にプルされます。ネットワーク設定によっては、ECR と S3 へのアクセスを許可する必要がある場合があります。たとえば、制限付きセキュリティグループを使用する場合は、S3 マネージドプレフィックスリストを許可します。詳細については、「コンテナイメージアクセスの前提条件」を参照してください。

サイドカーを起動できない場合、Runtime Monitoring はタスクの実行を妨げません。Fargate タスクは起動後に変更できないため、GuardDuty はタスクの起動時にサイドカーをアタッチします。

モニタリングスコープ

デフォルトでは、GuardDuty はアカウント内のすべての Amazon ECS クラスターのセキュリティエージェントを管理します。GuardDuty が特定のクラスターでエージェントを管理できないようにするには、タグ `GuardDutyManaged:false` をそのクラスターに追加します。

Fargate タスクのエージェントの管理の詳細については、「」を参照してくださいFargate の自動セキュリティエージェントの管理 (Amazon ECS のみ)。

考慮事項

- 自動エージェント設定がオフの場合、GuardDuty は でタグ付けした Amazon ECS クラスターでのみセキュリティエージェントを管理しますGuardDutyManagedtrue。
- GuardDutyManaged:false タグを追加すると、GuardDuty によるセキュリティエージェントの管理は停止しますが、インストール済みのエージェントはアンインストールされません。
- サービスコントロールポリシーまたは IAM ポリシーを使用して、タグを変更できるユーザーを制御してGuardDutyManagedタグを保護します。詳細については、「AWS Organizations ユーザーガイド」の「サービスコントロールポリシー (SCP)」または「IAM ユーザーガイド」の「AWS のリソースへのアクセスの制御」を参照してください。

関連トピック

- セキュリティエージェントが GuardDuty に接続する方法
- Runtime Monitoring カバレッジの追跡

セキュリティエージェントが GuardDuty に接続する方法

Runtime Monitoring セキュリティエージェントは、ランタイムイベントを GuardDuty に配信します。データは AWS ネットワーク内に残ります。GuardDuty が設定する接続には追加料金はかかりません。エージェントを認証するために、GuardDuty はインスタンス ID ロールを使用します。Fargate 上の Amazon ECS の場合、GuardDuty はタスク実行ロールを使用します。「」を参照してくださいアクセス許可の要件。

GuardDuty は、link-local と Amazon VPC エンドポイントの 2 つの接続方法をサポートしています。セキュリティエージェントは、これらの方法を使用して接続を試み、ランタイムイベントを GuardDuty に配信します。

自動エージェント設定を使用すると、GuardDuty は自動的に接続を確立します。エージェントを手動で管理する場合は、Amazon VPC エンドポイントを作成する必要があります。

リンクローカル接続

エージェントは、 のリンクローカルアドレスを介して GuardDuty バックエンドに接続しますguardduty-data-v2.*us-east-1*.amazonaws.com。リンクローカル接続はプライベート DNS を使用しません。(*us-east-1*) は AWS リージョン 、リージョンに基づいて変わります。

Amazon VPC エンドポイント接続

エージェントは Amazon VPC エンドポイントを介して GuardDuty バックエンドに接続します。エンドポイントのプライベート DNS 名を解決して接続します。FIPS 以外のエンドポイントの場合、これは ですguardduty-data.*us-east-1*.amazonaws.com。(*us-east-1*) は AWS リージョン 、リージョンに応じて変わります。

リソースには、アクティブな guardduty-data Amazon VPC エンドポイントへの有効なネットワークパスが必要です。GuardDuty は IP の可用性に基づいてサブネットを選択するため、高度なネットワークトポロジーでは、接続が可能であることを確認します。

前提条件

前提条件については、「Runtime Monitoring を有効にする前提条件」を参照してください。

考慮事項

- すべての VPCs – エージェントの自動設定により、GuardDuty は集中型 VPCs を含むすべての VPCs間の接続を設定します。一元化された VPCs「インターフェイス VPC エンドポイント AWS - スケーラブルで安全なマルチ VPC AWS ネットワークインフラストラクチャの構築」を参照してください。
- 共有 VPC – 共有 VPC を使用する場合、GuardDuty はその共有 VPC を使用して、前提条件が満たされた後にリソースからランタイムイベントを受信します。「Runtime Monitoring で共有 VPC を使用する」を参照してください。

関連トピック

- セキュリティエージェントの Amazon EC2 へのデプロイ
- セキュリティエージェントの Amazon EKS へのデプロイ
- Fargate での Amazon ECS へのセキュリティエージェントのデプロイ
- ランタイムカバレッジ統計の確認と問題のトラブルシューティング

Runtime Monitoring カバレッジの追跡

Runtime Monitoring を有効にし、スタンドアロンアカウントまたは複数のメンバーアカウントに GuardDuty セキュリティエージェントをインストールした後、次のステップを実行して保護プラン

設定が期待どおりに機能することを確認し、GuardDuty セキュリティエージェントが使用するメモリと CPU の量をモニタリングできます。

「ランタイムカバレッジの評価」

GuardDuty では、セキュリティエージェントをデプロイしたリソースのカバレッジステータスを継続的に評価することをお勧めします。カバレッジステータスは「正常」または「異常」の場合があります。「正常」のカバレッジステータスは、オペレーティングシステムレベルのアクティビティがあるときに、GuardDuty が対応するリソースからランタイムイベントを受信していることを示します。

リソースのカバレッジステータスが「正常」になると、GuardDuty はランタイムイベントを受信し、脅威を検出するために分析できます。GuardDuty が、コンテナワークロードとインスタンスで実行されているタスクまたはアプリケーションで潜在的なセキュリティ脅威を検出すると、GuardDuty はGuardDuty Runtime Monitoring の検出結果タイプを生成します。

カバレッジステータスが [異常] から [正常] に変わったときなどに通知を受け取るように Amazon EventBridge (EventBridge) を設定することもできます。詳細については、「ランタイムカバレッジ統計の確認と問題のトラブルシューティング」を参照してください。

GuardDuty セキュリティエージェントの CPU とメモリのモニタリングを設定する

カバレッジステータスが [正常] と表示されていることを評価した後、リソースタイプのセキュリティエージェントのパフォーマンスを評価できます。セキュリティエージェントリリース v1.5 以降の Amazon EKS クラスターの場合、GuardDuty は (アドオン) セキュリティエージェントのパラメータの設定をサポートしています。詳細については、「CPU とメモリモニタリングの設定」を参照してください。

Runtime Monitoring の 30 日間の無料トライアルの仕組み

30 日間の無料トライアル期間は、新しい GuardDuty アカウントと、Runtime Monitoring 機能が Amazon EC2 インスタンスおよび AWS Fargate (Amazon ECS のみ) に拡張される前に EKS Runtime Monitoring を既に有効にしている既存のアカウントでは動作が異なります。

GuardDuty のトライアル期間を使用している、または EKS Runtime Monitoring を一度も有効にしたことがない

以下のリストでは、GuardDuty の 30 日間の試用期間を使用している場合や、EKS Runtime Monitoring を有効にしたことがない場合に、30 日間の無料試用期間がどのように機能するかを説明しています。

- GuardDuty を初めて有効にする場合、Runtime Monitoring と EKS Runtime Monitoring はデフォルトで有効になっていません。

  アカウントまたは組織の Runtime Monitoring を有効にする場合は、脅威検出を監視したいリソースの GuardDuty セキュリティエージェントも必ず設定してください。例えば、Amazon EC2 インスタンスに Runtime Monitoring を使用する場合は、Runtime Monitoring を有効にした後、Amazon EC2 のセキュリティエージェントも設定する必要があります。これは、手動で行うか、GuardDuty を使用して自動的に行うかを選択できます。
- Runtime Monitoring 保護プランはアカウントレベルで有効になっています。30 日間の無料トライアル期間のリソースレベルでの仕組み。GuardDuty セキュリティエージェントが特定のリソースタイプにデプロイされた後、30 日間の無料トライアルは、GuardDuty がこのリソースタイプに関連付けられた最初のランタイムイベントを受信した時点で開始されます。例えば、GuardDuty エージェントをリソースレベル (Amazon EC2 インスタンス、Amazon ECS クラスター、Amazon EKS クラスターに対して) でデプロイしたとします。GuardDuty が Amazon EC2 インスタンスの最初のランタイムイベントを受信すると、Amazon EC2 に対してのみ 30 日間の無料トライアルが開始されます。
- EKS Runtime Monitoring のみを有効にする場合 - GuardDuty を初めて有効にすると、EKS Runtime Monitoring はデフォルトで有効になっていません (Runtime Monitoring のリリース以降)。EKS Runtime Monitoring を有効にする必要があります。これを最適に使用するには、GuardDuty セキュリティエージェントを手動で管理するか、自動エージェント設定を有効にして GuardDuty がユーザーに代わってエージェントを管理するようにしてください。EKS Runtime Monitoring の 30 日間の無料トライアル期間は、GuardDuty が Amazon EKS リソースの最初のランタイムイベントを受信した時点で開始されます。

Runtime Monitoring を開始する前に EKS Runtime Monitoring を有効にしました

このセクションは、 で EKS Runtime Monitoring が有効になっていて AWS アカウント、Runtime Monitoring に移行する場合にのみ使用します。

次のリストには、Runtime Monitoring を有効にするユースケースに適用されるシナリオが含まれています。

- EKS Runtime Monitoring 保護プランが有効で、GuardDuty コンソールエクスペリエンスを使用してこの保護プランを使用する既存の GuardDuty アカウントの場合 — Runtime Monitoring の発表により、EKS Runtime Monitoring コンソールエクスペリエンスは Runtime Monitoring に統合されました。EKS Runtime Monitoring の既存の設定は変わりません。引き続き API/CLI サポートを使用して、EKS Runtime Monitoring に関連する操作を実行できます。
- EKS Runtime Monitoring を Runtime Monitoring の一部として使用するには、アカウントまたは組織の Runtime Monitoring を設定する必要があります。Runtime Monitoring で同じ設定を維持するには、「EKS Runtime Monitoring から Runtime Monitoring への移行」を参照してください。ただし、Amazon EKS リソースの 30 日間の無料トライアルには影響しません。
- Runtime Monitoring 保護プランは、リージョンごとのアカウントレベルで有効になります。GuardDuty セキュリティエージェントが指定されたリソースタイプ (Amazon EC2 インスタンスと Amazon ECS クラスター) のいずれかにデプロイされた後、30 日間の無料トライアルは、GuardDuty がリソースタイプに関連付けられた最初のランタイムイベントを受信した時点で開始されます。各リソースタイプには、30 日間の無料トライアルがあります。

  例えば、Runtime Monitoring を有効にした後、GuardDuty エージェントを Amazon EC2 インスタンスにのみデプロイすることを選択すると、このリソースの 30 日間の無料トライアルは、GuardDuty が Amazon EC2 インスタンスの最初のランタイムイベントを受信したときにのみ開始されます。その後、Fargate (Amazon ECS のみ) の GuardDuty エージェントをデプロイすると、GuardDuty が Amazon ECS クラスターの最初のランタイムイベントを受信したときにのみ、このリソースの 30 日間の無料トライアルが開始されます。アカウントで EKS Runtime Monitoring が既に有効になっている場合、GuardDuty は Amazon EKS リソースの 30 日間の無料トライアルをリセットしません。

Runtime Monitoring を有効にする前提条件

Runtime Monitoring を有効にして GuardDuty セキュリティエージェントを管理するには、脅威検出のためにモニタリングする各リソースタイプの前提条件を満たす必要があります。リソースタイプごとに前提条件が異なります。例えば、GuardDuty はリソースタイプに基づいて異なる OS ディストリビューションをサポートしています。

Amazon EC2 リソースのみをモニタリングする場合は、Amazon EC2 インスタンスの前提条件に従います。後で Amazon EKS リソースをモニタリングする場合は、Amazon EKS クラスターに固有の前提条件に従う必要があります。

以下のセクションには、リソースタイプに基づく前提条件が含まれています。

内容

- Amazon EC2 インスタンスサポートの前提条件
- ECS-EC2 Bottlerocket サポートの前提条件
- AWS Fargate (Amazon ECS のみ) サポートの前提条件
- Amazon EKS クラスターサポートの前提条件
- サポートされている CPU アーキテクチャ、オペレーティングシステム、カーネルバージョン

Amazon EC2 インスタンスサポートの前提条件

このセクションでは、Amazon EC2 インスタンスのランタイム動作をモニタリングするための前提条件について説明します。これらの前提条件が満たされた後、「GuardDuty Runtime Monitoring の有効化」を参照してください。

トピック

- EC2 インスタンスを SSM 管理にする (自動エージェント設定のみ)
- アーキテクチャ要件を検証する
- マルチアカウント環境での組織サービスコントロールポリシーの検証
- 自動エージェント設定を使用する場合
- GuardDuty エージェントにおける CPU とメモリの制限
- 次のステップ

EC2 インスタンスを SSM 管理にする (自動エージェント設定のみ)

GuardDuty は AWS Systems Manager (SSM) を使用して、インスタンスにセキュリティエージェントを自動的にデプロイ、インストール、管理します。GuardDuty エージェントを手動でインストールして管理する場合は、SSM は必要ありません。

Amazon EC2 インスタンスを Systems Manager で管理するには、「AWS Systems Manager ユーザーガイド」の「Amazon EC2 インスタンス用のシステムマネージャーのセットアップ」を参照してください。

アーキテクチャ要件を検証する

OS ディストリビューションのアーキテクチャは、GuardDuty セキュリティエージェントの動作に影響を与える可能性があります。Amazon EC2 インスタンスに Runtime Monitoring を使用する前に、次の要件を満たす必要があります。

検証済みの CPU アーキテクチャ、OS ディストリビューション、カーネルのバージョンについては、「」を参照してくださいサポートされている CPU アーキテクチャ、オペレーティングシステム、カーネルバージョン。

- 追加要件 - Amazon ECS/Amazon EC2 をお持ちの場合のみ

  Amazon ECS/Amazon EC2 については、Amazon ECS に最適化された最新の AMI (2023 年 9 月 29 日以降) を使用するか、Amazon ECS エージェントバージョン v1.77.0 を使用することをお勧めします。

RLIMIT_MEMLOCK 値のユーザーの表示と更新

システムの RLIMIT_MEMLOCK 制限が低すぎると、GuardDuty セキュリティエージェントが設計どおりに動作しない可能性があります。GuardDuty では、ハードリミットとソフトリミットの両方が 32 MB 以上であることを推奨しています。制限を更新しない場合、GuardDuty はリソースのランタイムイベントをモニタリングできません。RLIMIT_MEMLOCK が規定の最小制限を上回った場合、これらの制限を更新することはオプションになります。

デフォルトの RLIMIT_MEMLOCK 値は、GuardDuty セキュリティエージェントをインストールする前またはインストール後に変更できます。

RLIMIT_MEMLOCK 値を表示するには

1. `ps aux | grep guardduty` を実行します。これにより、プロセス ID (pid) が出力されます。
2. 前のコマンドの出力からプロセス ID (pid) をコピーします。
3. pid を前のステップでコピーしたプロセス ID に置き換えてから `grep "Max locked memory" /proc/pid/limits` を実行します。

   これにより、GuardDuty セキュリティエージェントの実行時に使用される最大ロックメモリが表示されます。

RLIMIT_MEMLOCK 値を更新するには

1. /etc/systemd/system.conf.d/*NUMBER*-limits.conf ファイルが存在する場合は、このファイルから DefaultLimitMEMLOCK の行をコメントアウトします。このファイルは、優先度の高い デフォルト RLIMIT_MEMLOCK を設定し、/etc/systemd/system.conf ファイルの設定を上書きします。
2. /etc/systemd/system.conf ファイルを開き、#DefaultLimitMEMLOCK= がある行のコメントを解除します。
3. ハード制限とソフト RLIMIT_MEMLOCK 制限の両方を 32 MB 以上に指定して、デフォルト値を更新します。更新は次のようになります: DefaultLimitMEMLOCK=32M:32M。形式は soft-limit:hard-limit です。
4. sudo reboot を実行します。

マルチアカウント環境での組織サービスコントロールポリシーの検証

組織内のアクセス許可を管理するようにサービスコントロールポリシー (SCP) を設定している場合は、アクセス許可の境界が guardduty:SendSecurityTelemetry アクションを許可していることを確認します。GuardDuty では、さまざまなリソースタイプでランタイムモニタリングをサポートするためにこのアクセス許可が必要です。

アカウントがメンバーアカウントの場合は、関連する委任管理者にお問い合わせください。組織の SCP の管理については、「Service control policies (SCPs)」を参照してください。

自動エージェント設定を使用する場合

を使用するには自動エージェント設定 (推奨)、 が次の前提条件を満たす AWS アカウント 必要があります。

- エージェントの自動設定で包含タグを使用する場合、GuardDuty が新しいインスタンスの SSM 関連付けを作成するには、新しいインスタンスが SSM 管理され、https://console.aws.amazon.com/systems-manager/ コンソールの Fleet Manager の下に表示されることを確認します。
- 自動エージェント設定で除外タグを使用する場合:
  - アカウントの GuardDuty 自動エージェントを設定する前に、GuardDutyManaged:false タグを追加します。

    Amazon EC2 インスタンスを起動する前に、必ず除外タグを追加してください。Amazon EC2 の自動エージェント設定を有効にすると、除外タグなしで起動する EC2 インスタンスは、GuardDuty 自動エージェント設定の対象となります。

- インスタンスの [メタデータ内でタグを許可する] 設定を有効にします。この設定が必要なのは、GuardDuty がインスタンスメタデータサービス (IMDS) から除外タグを読み取って、インスタンスをエージェントインストールから除外するべきかどうかを判断しなければならないからです。詳細については、「Amazon EC2 ユーザーガイド」の「インスタンスメタデータ内のタグへのアクセスを有効にする」を参照してください。

GuardDuty エージェントにおける CPU とメモリの制限

CPU 制限

GuardDuty は、セキュリティエージェントをインスタンスの合計 vCPU 容量の 10% に制限します。たとえば、4 つの vCPU コアを持つインスタンスでは、エージェントは最大 0.4 個の vCPU を使用できます。

メモリ制限

Amazon EC2 インスタンスに関連付けられているメモリから、GuardDuty セキュリティエージェントが使用できるメモリは限られています。

次の表は、メモリ制限を示しています。

Amazon EC2 インスタンスのメモリ	GuardDuty エージェントの最大メモリ
8 GB 未満	128 MB
8 GB ~ 32 GB 未満	256 MB
32 GB 以上	1 GB

次のステップ

次のステップでは、Runtime Monitoring を設定し、セキュリティエージェントを (自動または手動で) 管理します。

ECS-EC2 Bottlerocket サポートの前提条件

> **Important**
>
> このセクションは、以下のバリアントを含む Bottlerocket ECS 最適化 AMIs バージョン `v1.62.1` 以上にのみ適用されます。
>
> - `bottlerocket-aws-ecs-2-x86_64` / `bottlerocket-aws-ecs-2-aarch64`
> - `bottlerocket-aws-ecs-3-x86_64` / `bottlerocket-aws-ecs-3-aarch64`
> - `bottlerocket-aws-ecs-2-nvidia` / `bottlerocket-aws-ecs-2-nvidia-fips`
> - `bottlerocket-aws-ecs-3-nvidia` / `bottlerocket-aws-ecs-3-nvidia-fips`

Runtime Monitoring は、Bottlerocket のコンテナ最適化設計に合わせてホストコンテナとして実行されるセキュリティエージェントを使用して、GuardDuty の脅威検出を Amazon ECS クラスターの Bottlerocket Amazon EC2 インスタンスに拡張します。有効にするには、このセクションの前提条件を完了します。これらの前提条件が満たされた後、「GuardDuty Runtime Monitoring の有効化」を参照してください。

要件と手順の詳細については、「」を参照してくださいBottlerocket での GuardDuty セキュリティエージェントの管理 (Amazon EC2 での Amazon ECS)。

トピック

- EC2 インスタンスを SSM で管理し、インスタンスのアクセス許可を設定する
- アーキテクチャ要件の検証
- ネットワーク接続要件
- マルチアカウント環境での組織サービスコントロールポリシーの検証
- 自動エージェント設定を使用する場合
- GuardDuty エージェントにおける CPU とメモリの制限
- 次のステップ

EC2 インスタンスを SSM で管理し、インスタンスのアクセス許可を設定する

GuardDuty は AWS Systems Manager (SSM) を使用して Bottlerocket インスタンスにセキュリティエージェントをデプロイ、インストール、管理し、Bottlerocket コントロールコンテナの SSM エージェントを介して実行します。

ほとんどの場合、Bottlerocket に対するインスタンスのアクセス許可は GuardDuty も対象としています。追加のポリシーは必要ありません。Bottlerocket がコントロールコンテナイメージと管理コンテナイメージをプルするために使用するAmazonEC2ContainerRegistryReadOnlyポリシーは、エージェントコンテナイメージをプルする GuardDuty の要件を満たします。詳細については、Bottlerocket README を参照してください。

インスタンスプロファイルには、次の管理ポリシーを含める必要があります。

- AmazonSSMManagedInstanceCore – Bottlerocket コントロールコンテナが SSM エージェントを実行するために必要です。Amazon EC2 インスタンスを Systems Manager で管理するには、「AWS Systems Manager ユーザーガイド」の「Amazon EC2 インスタンス用のシステムマネージャーのセットアップ」を参照してください。
- AmazonEC2ContainerRegistryReadOnly – Amazon ECR から GuardDuty エージェントコンテナイメージをプルするために必要です。少なくとも、次のアクセス許可が必要です。

```
...
      "ecr:GetAuthorizationToken",
      "ecr:BatchCheckLayerAvailability",
      "ecr:GetDownloadUrlForLayer",
      "ecr:BatchGetImage",
...
```

Amazon ECR アクセス許可をさらに制限するには、Amazon ECS-Amazon EC2 Bottlerocket の GuardDuty セキュリティエージェントをホストする Amazon ECR リポジトリ URI を追加できます。詳細については、「ECS-EC2 Bottlerocket 上の GuardDuty エージェント用の ECR リポジトリ」を参照してください。

アーキテクチャ要件の検証

インスタンスのアーキテクチャによって、GuardDuty セキュリティエージェントの動作が決まります。Bottlerocket ECS-2 および ECS-3 バリアントは、すぐにこれらの要件を満たします。Runtime Monitoring を有効にする前に、以下を確認してください。

- カーネルサポートには、eBPF、Tracepoints、および が含まれますKprobe。Bottlerocket ECS-2 バリアントはカーネルを使用し6.1、ECS-3 バリアントはカーネル を使用します6.12。どちらのバージョンも Runtime Monitoring のカーネル要件を満たしています。CONFIG_DEBUG_INFO_BTF=y カーネルオプションは、Bottlerocket ECS-2 および ECS-3 カーネルでデフォルトで有効になっています。

- CPU アーキテクチャの場合、Runtime Monitoring は AMD64 (x86_64) と ARM64 () をサポートしていますaarch64。

ネットワーク接続要件

GuardDuty はセキュリティエージェントを Amazon ECR のコンテナイメージとしてホストするため、インスタンスをプルするには Amazon ECR へのネットワーク接続が必要です。ネットワーク設定に応じて、次のいずれかのオプションを選択します。

オプション 1 - パブリックネットワークアクセスの使用 (利用可能な場合)

Bottlerocket インスタンスがアウトバウンドインターネットアクセスを持つサブネットで実行されている場合、追加のネットワーク設定は必要ありません。

オプション 2 - Amazon VPC エンドポイントを使用する (プライベートサブネット用)

プライベートサブネットの Bottlerocket インスタンスの場合、VPC エンドポイントはエージェントが必要とする接続を提供します。インスタンスがエージェントイメージをプルし、デプロイのために Systems Manager と通信できるように、Amazon ECR、Amazon S3、および Systems Manager のエンドポイントを設定します。

VPC エンドポイントの設定の詳細については、「Amazon Elastic Container Registry ユーザーガイド」の「Amazon ECR の VPC エンドポイントの作成」および「 ユーザーガイド」の「VPC エンドポイントの作成」を参照してください。 AWS Systems Manager

マルチアカウント環境での組織サービスコントロールポリシーの検証

組織内のアクセス許可を管理するようにサービスコントロールポリシー (SCP) を設定している場合は、アクセス許可の境界で guardduty:SendSecurityTelemetryアクションが許可されていることを確認します。GuardDuty では、さまざまなリソースタイプでランタイムモニタリングをサポートするためにこのアクセス許可が必要です。

アカウントがメンバーアカウントの場合は、関連する委任管理者にお問い合わせください。組織の SCP の管理については、「Service control policies (SCPs)」を参照してください。

自動エージェント設定を使用する場合

自動エージェント設定を使用するには、 が以下の前提条件を満たす AWS アカウント 必要があります。

- エージェントの自動設定で包含タグを使用する場合、GuardDuty が新しいインスタンスの SSM 関連付けを作成するには、新しいインスタンスが SSM 管理されており、https://console.aws.amazon.com/systems-manager/ コンソールの Fleet Manager の下に表示されることを確認します。
- 自動エージェント設定で除外タグを使用する場合:
  - `GuardDutyManaged:false` タグを Amazon EC2 インスタンスに追加してから起動し、アカウントの GuardDuty 自動エージェントを設定します。自動エージェント設定を有効にすると、除外タグなしで起動するすべてのインスタンスが自動的にカバーされます。
  - インスタンスの [メタデータ内でタグを許可する] 設定を有効にします。この設定が必要なのは、GuardDuty がインスタンスメタデータサービス (IMDS) から除外タグを読み取って、インスタンスをエージェントインストールから除外するべきかどうかを判断しなければならないからです。詳細については、「Amazon EC2 ユーザーガイド」の「インスタンスメタデータ内のタグへのアクセスを有効にする」を参照してください。

GuardDuty エージェントにおける CPU とメモリの制限

CPU 制限

GuardDuty は、セキュリティエージェントをインスタンスの合計 vCPU 容量の 10% に制限します。たとえば、4 つの vCPU コアを持つインスタンスでは、エージェントは最大 0.4 個の vCPU を使用できます。

メモリ制限

Amazon EC2 インスタンスに関連付けられたメモリから、GuardDuty セキュリティエージェントが使用できるメモリに制限があります。

次の表は、メモリ制限を示しています。

Amazon EC2 インスタンスのメモリ	GuardDuty エージェントの最大メモリ
8 GB 未満	128 MB
8 GB ~ 32 GB 未満	256 MB
32 GB 以上	1 GB

次のステップ

次のステップでは、Runtime Monitoring を設定し、セキュリティエージェントを自動的に管理します。

AWS Fargate (Amazon ECS のみ) サポートの前提条件

このセクションでは、Fargate-Amazon ECS リソースのランタイム動作をモニタリングするための前提条件について説明します。これらの前提条件が満たされた後、「GuardDuty Runtime Monitoring の有効化」を参照してください。

トピック

- アーキテクチャ要件の検証
- コンテナイメージアクセスの前提条件
- マルチアカウント環境での組織サービスコントロールポリシーの検証
- ロールのアクセス許可とポリシーのアクセス許可の境界の検証
- CPU とメモリの制限

アーキテクチャ要件の検証

使用するプラットフォームは、Amazon ECS クラスターからランタイムイベントを受信する際に GuardDuty セキュリティエージェントが GuardDuty をサポートする方法に影響を与える可能性があります。検証済みのプラットフォームのいずれかを使用していることを検証する必要があります。

最初の検討事項:

> Amazon ECS クラスターの AWS Fargate プラットフォームは Linux である必要があります。対応するプラットフォームバージョンは少なくとも `1.4.0` または `LATEST` である必要があります。有効なプラットフォームバージョンの詳細については、「Amazon Elastic Container Service デベロッパーガイド」の「Linux プラットフォームのバージョン」を参照してください。
>
> Windows プラットフォームバージョンはまだサポートされていません。

検証済みの CPU アーキテクチャ、OS ディストリビューション、カーネルのバージョンについては、「」を参照してくださいサポートされている CPU アーキテクチャ、オペレーティングシステム、カーネルバージョン。

コンテナイメージアクセスの前提条件

以下の前提条件は、Amazon ECR リポジトリから GuardDuty サイドカーコンテナイメージにアクセスするのに役立ちます。

アクセス許可の要件

タスク実行ロールには、GuardDuty セキュリティエージェントコンテナイメージをダウンロードするための特定の Amazon Elastic Container Registry (Amazon ECR) アクセス許可が必要です。

```
...
      "ecr:GetAuthorizationToken",
      "ecr:BatchCheckLayerAvailability",
      "ecr:GetDownloadUrlForLayer",
      "ecr:BatchGetImage",
...
```

Amazon ECR アクセス許可をさらに制限するには、GuardDuty セキュリティエージェントをホストする Amazon ECR リポジトリ URI を追加します AWS Fargate (Amazon ECS のみ) 。詳細については、「Amazon ECR リポジトリホスティング GuardDuty エージェント」を参照してください。

AmazonECSTaskExecutionRolePolicy マネージドポリシーを使用するか、`TaskExecutionRole` ポリシーに上記のアクセス許可を追加できます。

タスク定義の設定

Amazon ECS サービスを作成または更新する際には、タスク定義でサブネット情報を提供する必要があります。

「Amazon Elastic Container Service API リファレンス] の [CreateService] API と [UpdateService] API を実行するには、サブネット情報を渡す必要があります。詳細については、「Amazon Elastic Container Service デベロッパーガイド」の「Amazon ECS のタスク定義」を参照してください。

ネットワーク接続要件

Amazon ECR から GuardDuty コンテナイメージをダウンロードするには、ネットワーク接続を確保する必要があります。この要件は、Amazon ECR を使用してセキュリティエージェントをホストするため、GuardDuty に固有のものです。ネットワーク設定に応じて、以下のいずれかのオプションを実装する必要があります。

オプション 1 - パブリックネットワークアクセスの使用 (利用可能な場合)

Fargate タスクがアウトバウンドインターネットアクセスのあるサブネットで実行されている場合、追加のネットワーク設定は必要ありません。

オプション 2 - Amazon VPC エンドポイントを使用する (プライベートサブネット用)

Fargate タスクがインターネットアクセスできないプライベートサブネットで実行される場合は、GuardDuty セキュリティエージェントをホストする ECR リポジトリ URI がネットワークにアクセスできるようにするために、ECR の VPC エンドポイントを設定する必要があります。これらのエンドポイントがないと、プライベートサブネットのタスクは GuardDuty コンテナイメージをダウンロードできません。

VPC エンドポイントのセットアップ手順については、「Amazon Elastic コンテナレジストリユーザーガイド」の「Amazon ECR 用の VPC エンドポイントを作成する」を参照してください。

Fargate で GuardDuty コンテナをダウンロードできるようにする方法については、「Amazon Elastic Container Registry ユーザーガイド」の「Using Amazon ECR images with Amazon ECS」を参照してください。

セキュリティグループの設定

GuardDuty コンテナイメージは Amazon ECR にあり、Amazon S3 アクセスが必要です。この要件は、Amazon ECR からのコンテナイメージのダウンロードに固有です。ネットワークアクセスが制限されたタスクでは、S3 へのアクセスを許可するようにセキュリティグループを設定する必要があります。

セキュリティグループに、ポート 443 の S3 マネージドプレフィックスリスト (pl-xxxxxxxx) へのトラフィックを許可するアウトバウンドルールを追加します。アウトバウンドルールを追加するには、「Amazon VPC ユーザーガイド」の「セキュリティグループの設定」を参照してください。

AWSマネージドプレフィックスリストをコンソールで表示したり、 AWS Command Line Interface (AWS CLI) を使用して説明したりするには、「Amazon VPC ユーザーガイド」のAWS「 マネージドプレフィックスリスト」を参照してください。

マルチアカウント環境での組織サービスコントロールポリシーの検証

このセクションでは、サービスコントロールポリシー (SCP) の設定を検証して、組織全体で Runtime Monitoring が期待どおりに動作することを確認する方法について説明します。

組織内のアクセス許可を管理するために 1 つ以上のサービスコントロールポリシーを設定している場合は、guardduty:SendSecurityTelemetry アクションを拒否しないことを検証する必要があります。SCP の仕組みについては、「AWS Organizations ユーザーガイド」の「SCP の評価」を参照してください。

アカウントがメンバーアカウントの場合は、関連する委任管理者にお問い合わせください。組織の SCP の管理については、「AWS Organizations ユーザーガイド」の「サービスコントロールポリシー(SCP)」を参照してください。

マルチアカウント環境で設定したすべての SCP に対して次の手順を実行します。

SCP で **guardduty:SendSecurityTelemetry** が否定されていないことを検証するには

1. https://console.aws.amazon.com/organizations/ で Organizations コンソールを開きます。組織の管理アカウントで、IAM ロールとしてサインインするか、ルートユーザーとしてサインインする (推奨されません) 必要があります。
2. 左のナビゲーションペインで [ポリシー] を選択します。次に、[サポートされているポリシータイプ] で、[サービスコントロールポリシー] を選択します。
3. [サービスコントロールポリシー] ページで、検証するポリシーの名前を選択します。
4. ポリシーの詳細ページで、このポリシーのコンテンツを表示します。guardduty:SendSecurityTelemetry アクションを拒否しないことを確認してください。

   次の SCP ポリシーは、 guardduty:SendSecurityTelemetry アクションを拒否しない例です。

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
    "Effect": "Allow",
            "Action": [
                "guardduty:SendSecurityTelemetry"
            ],
            "Resource": "*"
        }
    ]
```

```
}
```

ポリシーがこのアクションを拒否する場合は、ポリシーを更新する必要があります。詳細については、「AWS Organizations ユーザーガイド」の「サービスコントロールポリシーの更新」を参照してください。

ロールのアクセス許可とポリシーのアクセス許可の境界の検証

次のステップを使用して、ロールとそのポリシーに関連付けられたアクセス許可の境界が`guardduty:SendSecurityTelemetry`アクションを制限していないことを確認します。

ロールとそのポリシーのアクセス許可の境界を表示するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/iam/ で IAM コンソールを開きます。
2. 左のナビゲーションペインの [アクセス管理] で、[ロール] を選択します。
3. [ロール] ページで、作成したロール *`TaskExecutionRole`* を選択します。
4. 選択したロールのページで、[アクセス許可] タブで、このロールに関連付けられたポリシー名を展開します。次に、このポリシーが `guardduty:SendSecurityTelemetry` を制限していないことを確認します。
5. アクセス許可の境界が設定されている場合は、このセクションを展開します。次に、各ポリシーを展開して、`guardduty:SendSecurityTelemetry` アクションが制限されていないことを確認します。ポリシーは次の Example SCP policy のようになります。

   必要に応じて、次のいずれかのアクションを実行します。

   - ポリシーを変更するには、[編集] を選択します。このポリシーの[アクセス許可の変更] ページで、ポリシーエディタでポリシーを更新します。JSON スキーマが有効であることを確認してください。その後、[Next] を選択します。その後、変更を確認して保存することができます。
   - このアクセス許可の境界を変更して別の境界を選択するには、[境界の変更] を選択します。
   - このアクセス許可の境界を削除するには、[境界の削除] を選択します。

   IAM ポリシーの管理の詳細については、「IAM ユーザーガイド」の「AWS Identity and Access Managementでのポリシーとアクセス管理」を参照してください。

CPU とメモリの制限

Fargate タスク定義では、CPU 値とメモリの値をタスクレベルで指定する必要があります。次の表は、タスクレベルの CPU 値とメモリの値の有効な組み合わせ、および、対応する GuardDuty コンテナの最大メモリ制限を示しています。

CPU の値	メモリの値	GuardDuty エージェントの最大メモリ制限
256 (.25 vCPU)	512 MiB、1 GB、2 GB	128 MB
512 (.5 vCPU)	1 GB、2 GB、3 GB、4 GB	
1,024 (1 vCPU)	2 GB、3 GB、4 GB	
	5 GB、6 GB、7 GB、8 GB	
2,048 (2 vCPU)	4 GB ~ 16 GB (1 GB のインクリメント)	
4,096 (4 vCPU)	8 GB ~ 20 GB (1 GB のインクリメント)	
8192 (8 vCPU)	16 GB ~ 28 GB (4 GB のインクリメント)	256 MB
	32 GB ~ 60 GB (4 GB のインクリメント)	512 MB
16384 (16 vCPU)	32 GB ~ 120 GB (8 GB のインクリメント)	1 GB

Runtime Monitoring を有効にして、クラスターのカバレッジステータスが [正常] と評価されると、コンテナインサイトメトリクスを設定および表示できます。詳細については、「Amazon ECS クラスターでの監視設定」を参照してください。

次のステップでは、Runtime Monitoring を設定し、セキュリティエージェントも設定します。

Amazon EKS クラスターサポートの前提条件

このセクションでは、Amazon EKS リソースのランタイム動作をモニタリングするための前提条件について説明します。これらの前提条件は、GuardDuty エージェントが期待どおりに機能するために不可欠です。これらの前提条件が満たされたら、「GuardDuty Runtime Monitoring の有効化」を参照してリソースのモニタリングを開始します。

Amazon EKS 機能のサポート

Runtime Monitoring は、Amazon EC2 インスタンスと Amazon EKS Auto Mode で実行されている Amazon EKS クラスターをサポートします。

Runtime Monitoring は、Amazon EKS Hybrid Nodes を使用する Amazon EKS クラスター、および AWS Fargateで実行されているクラスターをサポートしていません。

これらの Amazon EKS の詳細については、「Amazon EKS ユーザーガイド」の「Amazon EKS とは」を参照してください。

アーキテクチャ要件の検証

使用するプラットフォームは、EKS クラスターからランタイムイベントを受信する際に GuardDuty セキュリティエージェントが GuardDuty をサポートする方法に影響を与える可能性があります。検証済みのプラットフォームのいずれかを使用していることを検証する必要があります。GuardDuty エージェントを手動で管理している場合は、現在使用している GuardDuty エージェントのバージョンが、Kubernetes のバージョンでサポートされていることを確認します。

検証済みプラットフォーム

検証済みの CPU アーキテクチャ、OS ディストリビューション、カーネルのバージョンについては、「」を参照してくださいサポートされている CPU アーキテクチャ、オペレーティングシステム、カーネルバージョン。

現在、ARM64(`aarch64`) `5.4`のカーネルバージョンでは、GuardDuty は に関連する GuardDuty Runtime Monitoring の検出結果タイプ を生成できませんドメインネームシステム (DNS) イベント。

GuardDuty セキュリティエージェントでサポートされている Kubernetes のバージョン

次の表は、GuardDuty セキュリティエージェントでサポートされている EKS クラスターの Kubernetes のバージョンを示しています。

Amazon EKS アドオン GuardDuty セキュリティエージェントバージョン	Kubernetes バージョン
v1.17.1 (最新 - v1.17.1-eksbuild.2)	1.31 ~ 1.36
v1.16.0 (最新 - v1.16.0-eksbuild.2)	1.28 ~ 1.36
v1.15.0 (最新 - v1.15.0-eksbuild.2)	1.28 ~ 1.36
v1.12.2 (最新 - v1.12.2-eksbuild.2)	1.28 ~ 1.36
v1.12.1 (最新 - v1.12.1-eksbuild.4)	1.28 ~ 1.35
v1.11.0 (最新 - v1.11.0-eksbuild.4)	1.28 ~ 1.34
v1.10.0 (最新 - v1.10.0-eksbuild.2)	1.21 ~ 1.33
v1.9.0 (最新 - v1.9.0-eksbuild.2) v1.8.1 (最新 - v1.8.1-eksbuild.2)	1.21 ~ 1.32
v1.7.1 v1.7.0 v1.6.1	1.21 ~ 1.31
v1.6.0 v1.5.0 v1.4.1 v1.4.0 v1.3.1	1.21 ~ 1.29
v1.3.0 v1.2.0	1.21 ~ 1.28
v1.1.0	1.21 ~ 1.26

Amazon EKS アドオン GuardDuty セキュリティエージェントバージョン	Kubernetes バージョン
v1.0.0	1.21 - 1.25

一部の GuardDuty セキュリティエージェントバージョンでは、標準サポートが終了します。

エージェントリリースバージョンの詳細については、「Amazon EKS リソースの GuardDuty セキュリティエージェントバージョン」を参照してください。

CPU とメモリの制限

次の表は、GuardDuty 向け Amazon EKS アドオンの CPU とメモリの制限を示しています (`aws-guardduty-agent`)。

パラメータ	最小限度	最大限度
CPU	200m	1000m
メモリ	256 Mi	1024 Mi

Amazon EKS アドオンバージョン 1.5.0 以降を使用する場合、GuardDuty は CPU とメモリ値にアドオンスキーマを設定する機能を提供します。設定可能な範囲については、「設定可能なパラメータと値」を参照してください。

EKS Runtime Monitoring を有効にして、EKS クラスターのカバレッジステータスを評価すると、コンテナインサイトメトリクスを設定および表示できます。詳細については、「CPU とメモリモニタリングの設定」を参照してください。

組織サービスコントロールポリシーの検証

組織内のアクセス許可を管理するようにサービスコントロールポリシー (SCP) を設定している場合は、アクセス許可の境界が `guardduty:SendSecurityTelemetry` を制限していないことを確認します。GuardDuty では、さまざまなリソースタイプでランタイムモニタリングをサポートするためにこのアクセス許可が必要です。

アカウントがメンバーアカウントの場合は、関連する委任管理者にお問い合わせください。組織の SCP の管理については、「Service control policies (SCPs)」を参照してください。

サポートされている CPU アーキテクチャ、オペレーティングシステム、カーネルバージョン

このセクションでは、GuardDuty セキュリティエージェントをサポートすることが検証された CPU アーキテクチャ、OS ディストリビューション、カーネルバージョンを統合します。リソースタイプが個別に呼び出されない限り、これらの要件は Amazon EC2 インスタンスのランタイムモニタリング AWS Fargate (Amazon ECS のみ) 、および Amazon EKS クラスターに適用されます。

カーネルサポートには、eBPF、Tracepoints、および が含まれますKprobe。CPU アーキテクチャの場合、Runtime Monitoring は AMD64 (x64) と ARM64 (Graviton2 以降) をサポートしています[1]。

次の表は、GuardDuty セキュリティエージェントをサポートすることが検証された OS ディストリビューションとカーネルバージョンを示しています。

OS ディストリビューション[2]	カーネルバージョン[3]
Amazon Linux 2	5.4[4]、5.10[4]、5.15
Amazon Linux 2023	5.4[4]、 5.10[4]、 5.15、 6.1、 6.5、 6.8、 6.12
Ubuntu 20.04、22.04、24.04、26.04	5.4[4]、5.10[4]、5.15、6.1、6.5、6.8、6.13、6.14、6.15、6.16、6.17、6.18、7.0
Debian 11、12、13	5.4[4]、 5.10[4]、 5.15、 6.1、 6.5、 6.8、 6.12
RedHat 9.4、10.2	5.14、6.12
Fedora 34、40、41、43、44	5.11、5.17、6.8、6.12、7.1
CentOS ストリーム 9、10	5.14、6.12
Oracle Linux 8.9、9.3	5.15
Rocky Linux 9.5、10.1	5.14、6.12
Alma Linux 9、10	5.14、6.12

OS ディストリビューション[2]	カーネルバージョン[3]
SUSE Linux Enterprise Server 16	6.12
Bottlerocket	5.4[4]、5.10[4]、5.15、6.1、6.18

1. Runtime Monitoring は、A1 インスタンスタイプなどの第 1 世代 Graviton インスタンスをサポートしていません。
2. さまざまなオペレーティングシステムのサポート - GuardDuty は、前の表に記載されたオペレーティングディストリビューションでの Runtime Monitoring のサポートを検証しました。GuardDuty セキュリティエージェントは、前の表に記載されていないオペレーティングシステムでも実行できますが、GuardDuty チームは予想されるセキュリティ値を保証できません。
3. カーネルバージョンについては、 `CONFIG_DEBUG_INFO_BTF`フラグを y (true を意味する) に設定する必要があります。これは、GuardDuty セキュリティエージェントが想定どおりに実行できるようにするために必要です。

   AWS Fargate (Amazon ECS のみ) の場合、このカーネルオプションは Fargate によって管理され、ユーザーによるアクションは必要ありません。
4. カーネルバージョン 5.10 以前の場合、期待どおりに機能するために GuardDuty セキュリティエージェントは RAM (`RLIMIT_MEMLOCK`) のロックされたメモリを使用します。システムの `RLIMIT_MEMLOCK` 値が低すぎる場合、GuardDuty ではハード制限とソフト制限の両方を少なくとも 32 MB に設定することをお勧めします。Amazon EC2 インスタンスの`RLIMIT_MEMLOCK`デフォルト値の検証と変更については、「」を参照してください`RLIMIT_MEMLOCK` 値のユーザーの表示と更新。

GuardDuty Runtime Monitoring の有効化

アカウントで Runtime Monitoring を有効にする前に、ランタイムイベントを監視したいリソースタイプがプラットフォームの要件を満たしていることを確認してください。詳細については、「前提条件」を参照してください。

Runtime Monitoring の開始前に EKS Runtime Monitoring を使用していた場合は、API を使用して EKS Runtime Monitoring の既存の設定を確認して更新できます。既存の設定を EKS Runtime Monitoring から Runtime Monitoring に移行することもできます。詳細については、「EKS Runtime Monitoring から Runtime Monitoring への移行」を参照してください。

> **Note**
>
> 現在、このドキュメントでは、アカウントと組織の Runtime Monitoring をコンソールでのみ有効にする手順を説明しています。「API アクション」または「GuardDuty のAWS CLI」を使用して、Runtime Monitoring を有効にすることもできます。

Runtime Monitoring は、以下のトピックの手順を使用して設定できます。

内容

- マルチアカウント環境の Runtime Monitoring の有効化
- スタンドアロンアカウントの Runtime Monitoring の有効化

マルチアカウント環境の Runtime Monitoring の有効化

マルチアカウント環境では、委任 GuardDuty 管理者アカウントのみが、メンバーアカウントの Runtime Monitoring を有効または無効にすることや、組織内のメンバーアカウントに属するリソースタイプの自動エージェント設定を管理できます。GuardDuty メンバーアカウントからは、この設定を変更できません。委任 GuardDuty 管理者アカウントは、 を使用してメンバーアカウントを管理します AWS Organizations。マルチアカウント環境の詳細については、「複数のアカウントの管理」を参照してください。

委任 GuardDuty 管理者アカウントで有効にする

委任 GuardDuty 管理者アカウントで Runtime Monitoring を有効にするには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[Runtime Monitoring] を選択します。
3. [設定] タブの [Runtime Monitoring 設定] セクションで [編集] を選択します。

4. [すべてのアカウントについて有効にする] の使用

   委任 GuardDuty 管理者アカウントを含めて、組織に属するすべてのアカウントで Runtime Monitoring を有効にする場合は、[すべてのアカウントについて有効にする] を選択します。

5. [アカウントを手動で設定] の使用

   メンバーアカウントごとに個別に Runtime Monitoring を有効にする場合は、[アカウントを手動で設定] を選択します。

   - [委任された管理者 (このアカウント)] セクションで [有効にする] を選択します。

6. リソースタイプのセキュリティエージェントを管理します。詳細については、「GuardDuty セキュリティエージェントの管理」を参照してください。

すべてのメンバーアカウントで有効にする

組織内のすべてのメンバーアカウントの Runtime Monitoring を有効にするには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   委任 GuardDuty 管理者アカウントを使用してサインインします。

2. ナビゲーションペインで、[Runtime Monitoring] を選択します。
3. [Runtime Monitoring] ページで、[設定] タブの [Runtime Monitoring 設定] セクションの [編集] を選択します。
4. [すべてのアカウントについて有効にする] を選択します。
5. リソースタイプのセキュリティエージェントを管理します。詳細については、「GuardDuty セキュリティエージェントの管理」を参照してください。

すべての既存のアクティブなメンバーアカウントで有効にする

組織内の既存のメンバーアカウントの Runtime Monitoring を有効にするには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   組織の委任 GuardDuty 管理者アカウントを使用してサインインします。

2. ナビゲーションペインで、[Runtime Monitoring] を選択します。

3. [Runtime Monitoring] ページの [設定] タブで、Runtime Monitoring 設定の現在のステータスを表示できます。
4. [Runtime Monitoring] ペイン内の [アクティブなメンバーアカウント] セクションで、[アクション] を選択します。
5. [アクション] ドロップダウンメニューから、[すべての既存のアクティブなメンバーアカウントについて有効にする] を選択します。
6. [確認] を選択します。
7. リソースタイプのセキュリティエージェントを管理します。詳細については、「GuardDuty セキュリティエージェントの管理」を参照してください。

> **Note**
>
> メンバーアカウントの設定を更新するには、最大 24 時間かかる場合があります。

新しいメンバーアカウントでのみ Runtime Monitoring を自動的に有効にする

組織内の新しいメンバーアカウントの Runtime Monitoring を有効にするには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   組織の指定された委任 GuardDuty 管理者アカウントを使用してサインインします。
2. ナビゲーションペインで、[Runtime Monitoring] を選択します
3. [設定] タブの [Runtime Monitoring 設定] セクションで [編集] を選択します。
4. [アカウントを手動で設定] を選択します。
5. [新しいメンバーアカウントについて自動的に有効にする] を選択します。
6. リソースタイプのセキュリティエージェントを管理します。詳細については、「GuardDuty セキュリティエージェントの管理」を参照してください。

選択したアクティブなメンバーアカウントでのみ有効にする

個々のアクティブなメンバーアカウントの Runtime Monitoring を有効にするには

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

委任 GuardDuty 管理者アカウントの認証情報を使用してサインインします。

2. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
3. [アカウント] ページで、[Runtime Monitoring] および [エージェントの自動管理] の列の値を確認します。これらの値は、Runtime Monitoring と GuardDuty エージェント管理が、対応するアカウントで [有効] または [無効] のいずれであるのかを示します。
4. [アカウント] テーブルから、Runtime Monitoring を有効にするアカウントを選択します。一度に複数のアカウントを選択できます。
5. [確認] を選択します。
6. [保護プランを編集] を選択します。適切なアクションを選択します。
7. [確認] を選択します。
8. リソースタイプのセキュリティエージェントを管理します。詳細については、「GuardDuty セキュリティエージェントの管理」を参照してください。

スタンドアロンアカウントの Runtime Monitoring の有効化

スタンドアロンアカウントは、特定の の で保護プランを有効または無効にする決定を所有 AWS アカウント します AWS リージョン。

アカウントが を通じて AWS Organizations、または招待方法によって GuardDuty 管理者アカウントに関連付けられている場合、このセクションはアカウントには適用されません。詳細については、「マルチアカウント環境の Runtime Monitoring の有効化」を参照してください。

Runtime Monitoring を有効にした後、自動設定または手動デプロイを使用して GuardDuty セキュリティエージェントをインストールしてください。次の手順に記載されたすべてのステップを完了する一環として、セキュリティエージェントをインストールしてください。

スタンドアロンアカウントで Runtime Monitoring を有効にするには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[Runtime Monitoring] を選択します。
3. [設定] タブで [有効にする] を選択し、アカウントの EKS Runtime Monitoring を有効にします。
4. リソースタイプのセキュリティエージェントを管理します。詳細については、「GuardDuty セキュリティエージェントの管理」を参照してください。

GuardDuty セキュリティエージェントの管理

監視したいリソースの GuardDuty セキュリティエージェントを管理できます。複数のリソースタイプを監視したい場合は、必ずそのリソースの GuardDuty エージェントを管理してください。

以下のトピックは、セキュリティエージェントを管理する次の手順に役立ちます。

内容

- Amazon EC2 インスタンスの自動セキュリティエージェントの有効化
- Amazon EC2 リソースのセキュリティエージェントの手動管理
- Bottlerocket での GuardDuty セキュリティエージェントの管理 (Amazon EC2 での Amazon ECS)
- Fargate の自動セキュリティエージェントの管理 (Amazon ECS のみ)
- Amazon EKS リソースのセキュリティエージェントの自動管理
- Amazon EKS クラスターのセキュリティエージェントの手動管理
- Amazon EKS の GuardDuty セキュリティエージェント (アドオン) パラメータを設定する
- VPC エンドポイント設定の検証

Amazon EC2 インスタンスの自動セキュリティエージェントの有効化

このセクションでは、スタンドアロンアカウントまたはマルチアカウント環境で Amazon EC2 リソースの GuardDuty 自動エージェントを有効にするステップについて説明します。

続行する前に、必ずすべての「Amazon EC2 インスタンスサポートの前提条件」に従ってください。

トピック

- マルチアカウント環境で Amazon EC2 リソースの GuardDuty エージェントを有効にする
- スタンドアロンアカウントで Amazon EC2 リソースの GuardDuty 自動エージェントを有効にする
- Amazon EC2 手動エージェントから自動エージェントへの移行

GuardDuty エージェントの手動管理から GuardDuty 自動エージェントの有効化に移行する場合は、GuardDuty 自動エージェントを有効にするステップを実行する前に、「Amazon EC2 手動エージェントから自動エージェントへの移行」を参照してください。

マルチアカウント環境で Amazon EC2 リソースの GuardDuty エージェントを有効にする

マルチアカウント環境では、委任 GuardDuty 管理者アカウントのみが、組織内のメンバーアカウントに属するリソースタイプの自動エージェント設定を有効または無効にできます。GuardDuty メンバーアカウントからは、この設定を変更できません。委任 GuardDuty 管理者アカウントは、 を使用してメンバーアカウントを管理します AWS Organizations。マルチアカウント環境の詳細については、「複数のアカウントの管理」を参照してください。

委任 GuardDuty 管理者アカウントで有効にする

Configure for all instances

Runtime Monitoring で [すべてのアカウントで有効にする] を選択した場合、委任 GuardDuty 管理者アカウントで次のいずれかのオプションを選択します。

- オプション 1

  [自動エージェント設定] の [EC2] セクションで、[すべてのアカウントで有効にする] を選択します。

- オプション 2
  - [自動エージェント設定] の [EC2] セクションで、[アカウントを手動で設定する] を選択します。
  - [委任管理者 (このアカウント)] で、[有効にする] を選択します。
- [保存] を選択します。

Runtime Monitoring で [アカウントを手動で設定する] を選択した場合、次のステップを実行します。

- [自動エージェント設定] の [EC2] セクションで、[アカウントを手動で設定する] を選択します。
- [委任管理者 (このアカウント)] で、[有効にする] を選択します。
- [保存] を選択します。

委任 GuardDuty 管理者アカウントの自動エージェント設定を有効にするためにどのオプションを選択したかに関係なく、GuardDuty が作成する SSM 関連付けがこのアカウントに属するすべて

の EC2 リソースにセキュリティエージェントをインストールおよび管理することを確認できます。

1. https://console.aws.amazon.com/systems-manager/ で AWS Systems Manager コンソールを開きます。
2. SSM 関連付け (`GuardDutyRuntimeMonitoring-do-not-delete`) の [ターゲット] タブを開きます。[タグキー] が [InstanceIds] として表示されることを確認します。

Using inclusion tag in selected instances

選択した Amazon EC2 インスタンスに GuardDuty エージェントを設定するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/ec2/ で Amazon EC2 コンソールを開きます。
2. GuardDuty に潜在的な脅威をモニタリングおよび検出させたいインスタンスに `GuardDutyManaged:true` タグを追加します。このタグの追加については、タグを個々のリソースに追加する方法に関するページを参照してください。

   このタグを追加すると、GuardDuty は選択した EC2 インスタンスに対してセキュリティエージェントをインストールおよび管理できます。自動エージェント設定を明示的に有効にする必要はありません。
3. GuardDuty が作成する SSM 関連付けが、包含タグでタグ付けされた EC2 リソースにのみセキュリティエージェントをインストールおよび管理することを確認できます。

   https://console.aws.amazon.com/systems-manager/ で AWS Systems Manager コンソールを開きます。

   - 作成される SSM 関連付け (`GuardDutyRuntimeMonitoring-do-not-delete`) の [ターゲット] タブを開きます。[タグキー] は [tag:GuardDutyManaged] として表示されます。

Using exclusion tag in selected instances

> **Note**
>
> Amazon EC2 インスタンスを起動する前に、必ず除外タグを追加してください。Amazon EC2 の自動エージェント設定を有効にすると、除外タグなしで起動する EC2 インスタンスは、GuardDuty 自動エージェント設定の対象となります。

選択した Amazon EC2 インスタンスに GuardDuty エージェントを設定するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/ec2/ で Amazon EC2 コンソールを開きます。
2. GuardDuty に潜在的な脅威をモニタリングおよび検出させたくないインスタンスに `GuardDutyManaged:false` タグを追加します。このタグの追加については、タグを個々のリソースに追加する方法に関するページを参照してください。
3. インスタンスメタデータで除外タグを使用するには、次のステップを実行します。
   a. インスタンスの [詳細] タブで、[インスタンスメタデータ内のタグを許可する] のステータスを表示します。

      現在 [無効] になっている場合は、次のステップを使用してステータスを [有効] に変更します。それ以外の場合は、この手順をスキップしてください。
   b. [アクション] メニューで、[インスタンス設定] を選択します。
   c. [インスタンスメタデータ内のタグを許可する] を選択します。
4. 除外タグを追加した後、すべてのインスタンスの設定タブに記載されているのと同じステップを実行します。

これでランタイム「Amazon EC2 インスタンスのランタイムカバレッジとトラブルシューティング」を評価できるようになりました。

すべてのメンバーアカウントの自動有効化

> **Note**
>
> メンバーアカウントの設定を更新するには、最大 24 時間かかる場合があります。

Configure for all instances

以下のステップは、Runtime Monitoring セクションで [すべてのアカウントで有効にする] を選択したことを前提としています。

1. [Amazon EC2] の [自動エージェント設定] セクションで、[すべてのアカウントで有効にする] を選択します。
2. GuardDuty が作成する SSM 関連付け (`GuardDutyRuntimeMonitoring-do-not-delete`) が、このアカウントに属するすべての EC2 リソースにセキュリティエージェントをインストールおよび管理することを確認できます。

   a. https://console.aws.amazon.com/systems-manager/ で AWS Systems Manager コンソールを開きます。

   b. SSM 関連付けの [ターゲット] タブを開きます。[タグキー] が [InstanceIds] として表示されることを確認します。

Using inclusion tag in selected instances

選択した Amazon EC2 インスタンスに GuardDuty エージェントを設定するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/ec2/ で Amazon EC2 コンソールを開きます。
2. GuardDuty に潜在的な脅威をモニタリングおよび検出させたい EC2 インスタンスに `GuardDutyManaged:true` タグを追加します。このタグの追加については、タグを個々のリソースに追加する方法に関するページを参照してください。

   このタグを追加すると、GuardDuty は選択した EC2 インスタンスに対してセキュリティエージェントをインストールおよび管理できます。自動エージェント設定を明示的に有効にする必要はありません。
3. GuardDuty が作成する SSM 関連付けが、アカウントに属するすべての EC2 リソースにセキュリティエージェントをインストールおよび管理することを確認できます。

   a. https://console.aws.amazon.com/systems-manager/ で AWS Systems Manager コンソールを開きます。

   b. SSM 関連付け (`GuardDutyRuntimeMonitoring-do-not-delete`) の [ターゲット] タブを開きます。[タグキー] が [InstanceIds] として表示されることを確認します。

Using exclusion tag in selected instances

> **Note**
>
> Amazon EC2 インスタンスを起動する前に、必ず除外タグを追加してください。Amazon EC2 の自動エージェント設定を有効にすると、除外タグなしで起動する EC2 インスタンスは、GuardDuty 自動エージェント設定の対象となります。

選択した Amazon EC2 インスタンスに GuardDuty セキュリティエージェントを設定するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/ec2/ で Amazon EC2 コンソールを開きます。
2. GuardDuty に潜在的な脅威をモニタリングおよび検出させたくないインスタンスに `GuardDutyManaged:false` タグを追加します。このタグの追加については、タグを個々のリソースに追加する方法に関するページを参照してください。
3. インスタンスメタデータで除外タグを使用するには、次のステップを実行します。
   a. インスタンスの [詳細] タブで、[インスタンスメタデータ内のタグを許可する] のステータスを表示します。

      現在 [無効] になっている場合は、次のステップを使用してステータスを [有効] に変更します。それ以外の場合は、この手順をスキップしてください。
   b. [アクション] メニューで、[インスタンス設定] を選択します。
   c. [インスタンスメタデータ内のタグを許可する] を選択します。
4. 除外タグを追加した後、すべてのインスタンスの設定タブに記載されているのと同じステップを実行します。

これでランタイム「Amazon EC2 インスタンスのランタイムカバレッジとトラブルシューティング」を評価できるようになりました。

新しいメンバーアカウントでのみ自動的に有効にする

委任 GuardDuty 管理者アカウントは、Amazon EC2 リソースの自動エージェント設定を設定して、新しいメンバーアカウントが組織に加わるときに自動的に有効にできます。

Configure for all instances

次のステップでは、[Runtime Monitoring] セクションで、[新しいメンバーアカウントで自動的に有効にする] を選択した場合を前提としています。

1. ナビゲーションペインで、[Runtime Monitoring] を選択します。
2. [Runtime Monitoring] ページで、[編集] を選択します。
3. [新しいメンバーアカウントについて自動的に有効にする] を選択します。このステップにより、新しいアカウントが組織に参加するたびに、そのアカウントに対して Amazon EC2 の自動エージェント設定が自動的に有効になります。この選択を変更できるのは、組織の委任 GuardDuty 管理者アカウントのみです。
4. [保存] を選択します。

新しいメンバーアカウントが組織に加わると、この設定が自動的に有効になります。GuardDuty がこの新しいメンバーアカウントに属する Amazon EC2 インスタンスのセキュリティエージェントを管理するには、「EC2 インスタンスの場合」に記載されたすべての前提条件が満たされていることを確認してください。

SSM 関連付けが作成されると (`GuardDutyRuntimeMonitoring-do-not-delete`)、SSM 関連付けが新しいメンバーアカウントに属するすべての EC2 インスタンスにセキュリティエージェントをインストールおよび管理することを確認できます。

- https://console.aws.amazon.com/systems-manager/ で AWS Systems Manager コンソールを開きます。
- SSM 関連付けの [ターゲット] タブを開きます。[タグキー] が [InstanceIds] として表示されることを確認します。

Using inclusion tag in selected instances

アカウント内の選択したインスタンスに GuardDuty セキュリティエージェントを設定するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/ec2/ で Amazon EC2 コンソールを開きます。
2. GuardDuty に潜在的な脅威をモニタリングおよび検出させたいインスタンスに `GuardDutyManaged:true` タグを追加します。このタグの追加については、タグを個々のリソースに追加する方法に関するページを参照してください。

このタグを追加すると、GuardDuty は選択したインスタンスに対してセキュリティエージェントをインストールおよび管理できます。自動エージェント設定を明示的に有効にする必要はありません。

3. GuardDuty が作成する SSM 関連付けが、包含タグでタグ付けされた EC2 リソースにのみセキュリティエージェントをインストールおよび管理することを確認できます。

   a. https://console.aws.amazon.com/systems-manager/ で AWS Systems Manager コンソールを開きます。

   b. 作成される SSM 関連付けの [ターゲット] タブを開きます。[タグキー] は [tag:GuardDutyManaged] として表示されます。

Using exclusion tag in selected instances

> **Note**
>
> Amazon EC2 インスタンスを起動する前に、必ず除外タグを追加してください。Amazon EC2 の自動エージェント設定を有効にすると、除外タグなしで起動する EC2 インスタンスは、GuardDuty 自動エージェント設定の対象となります。

スタンドアロンアカウントの特定のインスタンスに GuardDuty セキュリティエージェントを設定するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/ec2/ で Amazon EC2 コンソールを開きます。
2. GuardDuty に潜在的な脅威をモニタリングおよび検出させたくないインスタンスに `GuardDutyManaged:false` タグを追加します。このタグの追加については、タグを個々のリソースに追加する方法に関するページを参照してください。
3. インスタンスメタデータで除外タグを使用するには、次のステップを実行します。

   a. インスタンスの [詳細] タブで、[インスタンスメタデータ内のタグを許可する] のステータスを表示します。

      現在 [無効] になっている場合は、次のステップを使用してステータスを [有効] に変更します。それ以外の場合は、この手順をスキップしてください。

   b. [アクション] メニューで、[インスタンス設定] を選択します。

c. [インスタンスメタデータ内のタグを許可する] を選択します。

4. 除外タグを追加した後、すべてのインスタンスの設定タブに記載されているのと同じステップを実行します。

これでランタイム「Amazon EC2 インスタンスのランタイムカバレッジとトラブルシューティング」を評価できるようになりました。

選択したメンバーアカウントのみ

Configure for all instances

1. [アカウント] ページで、[Runtime Monitoring 自動エージェント設定 (Amazon EC2)] を有効にする 1 つ以上のアカウントを選択します。このステップで選択したアカウントで Runtime Monitoring が既に有効になっていることを確認してください。
2. [保護プランの編集] から、適切なオプションを選択して [Runtime Monitoring 自動エージェント設定 (Amazon EC2)] を有効にします。
3. [確認] を選択します。

Using inclusion tag in selected instances

選択したインスタンスに GuardDuty セキュリティエージェントを設定するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/ec2/ で Amazon EC2 コンソールを開きます。
2. GuardDuty に潜在的な脅威をモニタリングおよび検出させたいインスタンスに `GuardDutyManaged:true` タグを追加します。このタグの追加については、タグを個々のリソースに追加する方法に関するページを参照してください。

   このタグを追加すると、GuardDuty がタグ付けされた Amazon EC2 インスタンスのセキュリティエージェントを管理できるようになります。自動エージェント設定 [Runtime Monitoring - 自動エージェント設定 (EC2)] を明示的に有効にする必要はありません。

Using exclusion tag in selected instances

> **Note**
>
> Amazon EC2 インスタンスを起動する前に、必ず除外タグを追加してください。Amazon EC2 の自動エージェント設定を有効にすると、除外タグなしで起動する EC2 インスタンスは、GuardDuty 自動エージェント設定の対象となります。

選択したインスタンスに GuardDuty セキュリティエージェントを設定するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/ec2/ で Amazon EC2 コンソールを開きます。
2. GuardDuty に潜在的な脅威をモニタリングまたは検出させたくない EC2 インスタンスに `GuardDutyManaged:false` タグを追加します。このタグの追加については、タグを個々のリソースに追加する方法に関するページを参照してください。
3. インスタンスメタデータで除外タグを使用するには、次のステップを実行します。
   a. インスタンスの [詳細] タブで、[インスタンスメタデータ内のタグを許可する] のステータスを表示します。

      現在 [無効] になっている場合は、次のステップを使用してステータスを [有効] に変更します。それ以外の場合は、この手順をスキップしてください。
   b. [アクション] メニューで、[インスタンス設定] を選択します。
   c. [インスタンスメタデータ内のタグを許可する] を選択します。
4. 除外タグを追加した後、すべてのインスタンスの設定タブに記載されているのと同じステップを実行します。

これで、「Amazon EC2 インスタンスのランタイムカバレッジとトラブルシューティング」を評価できるようになりました。

スタンドアロンアカウントで Amazon EC2 リソースの GuardDuty 自動エージェントを有効にする

スタンドアロンアカウントは、特定の の で保護プランを有効または無効にする決定を所有 AWS アカウント します AWS リージョン。

アカウントが を通じて AWS Organizations、または招待方法によって GuardDuty 管理者アカウントに関連付けられている場合、このセクションはアカウントには適用されません。詳細については、「マルチアカウント環境の Runtime Monitoring の有効化」を参照してください。

Runtime Monitoring を有効にした後、自動設定または手動デプロイを使用して GuardDuty セキュリティエージェントをインストールしてください。次の手順に記載されたすべてのステップを完了する一環として、セキュリティエージェントをインストールしてください。

すべての Amazon EC2 リソースをモニタリングするか、選択した Amazon EC2 リソースをモニタリングするかに応じて、任意の方法を選択し、次の表のステップに従います。

Configure for all instances

スタンドアロンアカウント内のすべてのインスタンスで Runtime Monitoring を設定するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[Runtime Monitoring] を選択します。
3. [設定] タブで、[編集] を選択します。
4. [EC2] セクションで [有効にする] を選択します。
5. [保存] を選択します。
6. GuardDuty が作成する SSM 関連付けが、アカウントに属するすべての EC2 リソースにセキュリティエージェントをインストールおよび管理することを確認できます。

    a. https://console.aws.amazon.com/systems-manager/ で AWS Systems Manager コンソールを開きます。

    b. SSM 関連付け (`GuardDutyRuntimeMonitoring-do-not-delete`) の [ターゲット] タブを開きます。[タグキー] が [InstanceIds] として表示されることを確認します。

Using inclusion tag in selected instances

選択した Amazon EC2 インスタンスに GuardDuty セキュリティエージェントを設定するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/ec2/ で Amazon EC2 コンソールを開きます。
2. GuardDuty に潜在的な脅威をモニタリングおよび検出させたいインスタンスに `GuardDutyManaged:true` タグを追加します。このタグの追加については、タグを個々のリソースに追加する方法に関するページを参照してください。

3. GuardDuty が作成する SSM 関連付けが、包含タグでタグ付けされた EC2 リソースにのみセキュリティエージェントをインストールおよび管理することを確認できます。

   https://console.aws.amazon.com/systems-manager/ で AWS Systems Manager コンソールを開きます。

   - 作成される SSM 関連付け (`GuardDutyRuntimeMonitoring-do-not-delete`) の [ターゲット] タブを開きます。[タグキー] は [tag:GuardDutyManaged] として表示されます。

Using exclusion tag in selected instances

> **Note**
>
> Amazon EC2 インスタンスを起動する前に、必ず除外タグを追加してください。Amazon EC2 の自動エージェント設定を有効にすると、除外タグなしで起動する EC2 インスタンスは、GuardDuty 自動エージェント設定の対象となります。

選択した Amazon EC2 インスタンスに GuardDuty セキュリティエージェントを設定するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/ec2/ で Amazon EC2 コンソールを開きます。
2. GuardDuty に潜在的な脅威をモニタリングおよび検出させたくないインスタンスに `GuardDutyManaged:false` タグを追加します。このタグの追加については、タグを個々のリソースに追加する方法に関するページを参照してください。
3. インスタンスメタデータで除外タグを使用するには、次のステップを実行します。

   a. インスタンスの [詳細] タブで、[インスタンスメタデータ内のタグを許可する] のステータスを表示します。

      現在 [無効] になっている場合は、次のステップを使用してステータスを [有効] に変更します。それ以外の場合は、この手順をスキップしてください。

   b. タグを許可するインスタンスを選択します。

   c. [アクション] メニューで、[インスタンス設定] を選択します。

   d. [インスタンスメタデータ内のタグを許可する] を選択します。

   e. [インスタンスメタデータのタグへのアクセス] で [許可する] を選択します。

f. [保存] を選択します。

4. 除外タグを追加した後、すべてのインスタンスの設定タブに記載されているのと同じステップを実行します。

これでランタイム「Amazon EC2 インスタンスのランタイムカバレッジとトラブルシューティング」を評価できるようになりました。

Amazon EC2 手動エージェントから自動エージェントへの移行

このセクションは、以前にセキュリティエージェントを手動で管理 AWS アカウント していて、GuardDuty 自動エージェント設定を使用する場合に適用されます。これに該当しない場合は、アカウントのセキュリティエージェントの設定を続行します。

GuardDuty 自動エージェントを有効にすると、GuardDuty はユーザーに代わってセキュリティエージェントを管理します。GuardDuty が実行するステップについては、「自動エージェント設定 (推奨)」を参照してください。

リソースをクリーンアップする

SSM 関連付けの削除

- Amazon EC2 のセキュリティエージェントを手動で管理するときに作成した SSM 関連付けを削除します。詳細については、「関連付けを削除する」を参照してください。
- これは、アカウントレベルまたはインスタンスレベルで (包含タグまたは除外タグを使用して) 自動エージェントを使用するかどうかにかかわらず、GuardDuty が SSM アクションの管理を引き継ぐことができるようにするために行われます。GuardDuty が実行できる SSM アクションの詳細については、「GuardDuty のためのサービスにリンクされたロールの許可」を参照してください。
- セキュリティエージェントを手動で管理するために以前作成された SSM 関連付けを削除すると、GuardDuty がセキュリティエージェントを自動的に管理するための SSM 関連付けを作成するときに、短時間の重複が発生する場合があります。この期間中、SSM スケジューリングに基づいて競合が発生する可能性があります。詳細については、Amazon EC2 SSM のスケジューリングに関するページを参照してください。

Amazon EC2 インスタンスの包含タグと除外タグを管理する

- 包含タグ – GuardDuty 自動エージェント設定を有効にせず、Amazon EC2 インスタンスに包含タグ (`GuardDutyManaged:true`) をタグ付けすると、GuardDuty は、選択した EC2 インスタンスにセキュリティエージェントをインストールおよび管理する SSM 関連付けを作成しま

す。これは、選択した EC2 インスタンスでのみセキュリティエージェントを管理するのに役立つ正常な動作です。詳細については、「セキュリティエージェントの Amazon EC2 へのデプロイ」を参照してください。

GuardDuty がセキュリティエージェントをインストールおよび管理できないようにするには、これらの EC2 インスタンスから包含タグを削除します。詳細については、「Amazon EC2 ユーザーガイド」のタグの追加と削除に関するページを参照してください。

- 除外タグ – アカウント内のすべての EC2 インスタンスで GuardDuty 自動エージェント設定を有効にする場合は、EC2 インスタンスに除外タグ (`GuardDutyManaged:false`) がタグ付けされていないことを確認してください。

Amazon EC2 リソースのセキュリティエージェントの手動管理

このセクションでは、Amazon EC2 リソースのセキュリティエージェントを手動でインストールおよび更新するステップについて説明します。

Runtime Monitoring を有効にしたら、GuardDuty セキュリティエージェントを手動でインストールする必要があります。GuardDuty セキュリティエージェントを手動で管理するには、まず Amazon VPC エンドポイントを手動で作成する必要があります。その後、GuardDuty が Amazon EC2 インスタンスからランタイムイベントを受信し始めるように、セキュリティエージェントをインストールできます。GuardDuty がこのリソースの新しいエージェントバージョンをリリースすると、アカウントでエージェントバージョンを更新できます。

以下のトピックでは、Amazon EC2 リソースのセキュリティエージェントを継続的に管理するステップについて説明します。

トピック

- 前提条件 - Amazon VPC エンドポイントの手動作成
- セキュリティエージェントの手動インストール
- Amazon EC2 インスタンスの GuardDuty セキュリティエージェントの手動更新

前提条件 - Amazon VPC エンドポイントの手動作成

GuardDuty セキュリティエージェントをインストールする前に、Amazon Virtual Private Cloud (Amazon VPC) エンドポイントを作成する必要があります。これにより、GuardDuty は Amazon EC2 インスタンスのランタイムイベントを受信できるようになります。

> **Note**
>
> VPC エンドポイントの使用に追加コストはかかりません。

Amazon VPC エンドポイントを作成するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/vpc/ で Amazon VPC コンソールを開きます。
2. ナビゲーションペインの [VPC プライベートクラウド] で、[エンドポイント] を選択します。
3. [エンドポイントの作成] を選択します。
4. [エンドポイントの作成] ページの [サービスカテゴリ] で [その他のエンドポイントサービス] を選択します。
5. [サービス名] に **com.amazonaws.*us-east-1*.guardduty-data** と入力します。

   必ず「*us-east-1*」をあなたの AWS リージョンに置き換えてください。これは、 AWS アカウント ID に属する Amazon EC2 インスタンスと同じリージョンである必要があります。
6. [サービスの確認] を選択します。
7. サービス名が正常に確認されたら、インスタンスが置かれている [VPC] を選択します。次のポリシーを追加して、Amazon VPC エンドポイントの使用を指定されたアカウントのみに制限します。このポリシー下で提供されている組織 `Condition` を使用して、次のポリシーを更新してエンドポイントへのアクセスを制限できます。組織内の特定のアカウント ID に Amazon VPC エンドポイントサポートを提供するには、「Organization condition to restrict access to your endpoint」を参照してください。

   JSON

```
{
 "Version":"2012-10-17",
 "Statement": [
  {
   "Action": "*",
   "Resource": "*",
   "Effect": "Allow",
   "Principal": "*"
  },
  {
   "Condition": {
```

```
    "StringNotEquals": {
     "aws:PrincipalAccount": "111122223333"
    }
   },
   "Action": "*",
   "Resource": "*",
   "Effect": "Deny",
   "Principal": "*"
  }
 ]
}
```

aws:PrincipalAccount アカウント ID は、VPC と VPC エンドポイントを含むアカウントと一致する必要があります。次のリストは、VPC エンドポイントを他の AWS アカウント IDs と共有する方法を示しています。

- VPC エンドポイントにアクセスする複数のアカウントを指定するには、"aws:PrincipalAccount: "*111122223333*"を以下のブロックに置き換えます。

  ```
  "aws:PrincipalAccount": [
            "666666666666",
            "555555555555"
       ]
  ```

  AWS アカウント IDs は、VPC エンドポイントにアクセスする必要があるアカウントのアカウント IDs に置き換えてください。

- 組織のすべてのメンバーが VPC エンドポイントにアクセスできるようにするには、"aws:PrincipalAccount: "*111122223333*" を以下のラインに置き換えます。

  ```
  "aws:PrincipalOrgID": "o-abcdef0123"
  ```

  組織「*o-abcdef0123*」は必ず自分の組織 ID に置き換えてください。

- リソースへのアクセスを組織 ID で制限するには、ResourceOrgID をポリシーに追加します。詳細については、「IAM ユーザーガイド」の「*aws:ResourceOrgID*」を参照してください。

  ```
  "aws:ResourceOrgID": "o-abcdef0123"
  ```

8. [追加設定] で [DNS 名を有効にする] を選択します。
9. [サブネット] で、インスタンスが存在するサブネットを選択します。
10. [セキュリティグループ] で、VPC (または Amazon EC2 インスタンス) からのインバウンドポート 443 が有効になっているセキュリティグループを選択します。インバウンドポート 443 が有効になっているセキュリティグループがまだない場合は、「Amazon VPC ユーザーガイド」の「VPC 用のセキュリティグループを作成するには」を参照してください。

    VPC (またはインスタンス) へのインバウンド許可を制限する際に問題が発生した場合は、任意の IP アドレス (`0.0.0.0/0`) からのインバウンド 443 ポートをサポートできます。ただし、GuardDuty では、VPC の CIDR ブロックに一致する IP アドレスを使用することをお勧めします。詳細については、「Amazon VPC ユーザーガイド」の「VPC CIDR ブロック」を参照してください。

ステップに従った後、「VPC エンドポイント設定の検証」を参照して、VPC エンドポイントが正しく設定されていることを確認します。

セキュリティエージェントの手動インストール

GuardDuty には、Amazon EC2 インスタンスに GuardDuty セキュリティエージェントをインストールするための次の 2 つの方法が用意されています。続行する前に、「前提条件 - Amazon VPC エンドポイントの手動作成」のステップに従ってください。

Amazon EC2 リソースにセキュリティエージェントをインストールする優先アクセス方法を選択します。

- 方法 1 - の使用 AWS Systems Manager – この方法では、Amazon EC2 インスタンス AWS Systems Manager を管理する必要があります。
- 方法 2 - Linux パッケージマネージャーを使用する – Amazon EC2 インスタンスが AWS Systems Manager 管理されているかどうかにかかわらず、この方法を使用できます。OS ディストリビューションに応じて、RPM スクリプトまたは Debian スクリプトのいずれかをインストールする適切な方法を選択できます。Fedora プラットフォームを使用する場合は、この方法を使用してエージェントをインストールする必要があります。

方法 1 - の使用 AWS Systems Manager

この方法を使用するには、Amazon EC2 インスタンスが AWS Systems Manager 管理されていることを確認し、エージェントをインストールします。

AWS Systems Manager マネージド Amazon EC2 インスタンス

Amazon EC2 インスタンスを AWS Systems Manager マネージドにするには、次のステップを実行します。

- AWS Systems Manager は、 AWS アプリケーションとリソースをend-to-endで管理し、安全なオペレーションを大規模に有効にするのに役立ちます。

  で Amazon EC2 インスタンスを管理するには AWS Systems Manager、 AWS Systems Manager ユーザーガイドの「Amazon EC2 インスタンス用の Systems Manager のセットアップ」を参照してください。
- 次の表に、新しい GuardDuty 管理 AWS Systems Manager ドキュメントを示します。

ドキュメント名	[Document type (ドキュメントタイプ)]	目的
`AmazonGuardDuty-RuntimeMonitoringSsmPlugin`	ディストリビューター	GuardDuty セキュリティエージェントのパッケージ化
`AmazonGuardDuty-ConfigureRuntimeMonitoringSsmPlugin`	コマンド	インストール/アンインストールスクリプトを実行して GuardDuty セキュリティエージェントをインストールします。

詳細については AWS Systems Manager、「 AWS Systems Manager ユーザーガイド」のAmazon EC2 Systems Manager ドキュメント」を参照してください。

> **Debian サーバーの場合**
>
> が提供する AWS Debian サーバー用の Amazon マシンイメージ (AMIs) では、 AWS Systems Manager エージェント (SSM エージェント) をインストールする必要があります。Amazon EC2 Debian サーバーインスタンスを SSM マネージドにするには、SSM エージェントをインストールするための追加のステップを実行する必要があります。必要なステップについては、「AWS Systems Manager ユーザーガイド」の「Debian サーバーインスタンスに SSM Agent を手動でインストールする」を参照してください。

を使用して Amazon EC2 インスタンス用の GuardDuty エージェントをインストールするには AWS Systems Manager

1. https://console.aws.amazon.com/systems-manager/ で AWS Systems Manager コンソールを開きます。
2. ナビゲーションペインで、[ドキュメント] を選択します。
3. [Amazon が所有] で、`AmazonGuardDuty-ConfigureRuntimeMonitoringSsmPlugin` を選択します。
4. [Run Command] を選択します。
5. 次の Run Command パラメータを入力します。
   - アクション: [インストール] を選択します。
   - インストールのタイプ: [インストール] または [アンインストール] を選択します。
   - 名前: `AmazonGuardDuty-RuntimeMonitoringSsmPlugin`
   - バージョン:これが空のままの場合は、最新バージョンの GuardDuty セキュリティエージェントが取得されます。リリースバージョンの詳細については、「Amazon EC2 インスタンス用の GuardDuty セキュリティエージェントバージョン」を参照してください。
6. 対象の Amazon EC2 インスタンス を選択します。複数の Amazon EC2 インスタンスを選択できます。詳細については、「AWS Systems Manager ユーザーガイド」の「AWS Systems Manager コンソールからコマンドを実行する」を参照してください
7. GuardDuty エージェントが正常にインストールされているかどうかを検証します。詳細については、「GuardDuty セキュリティエージェントのインストールステータスの検証」を参照してください。

方法 2 - Linux パッケージマネージャーを使用する

この方法では、RPM スクリプトまたは Debian スクリプトを実行して、GuardDuty セキュリティエージェントをインストールできます。オペレーティングシステムに応じて、任意の方法を選択できます。

- RPM スクリプトを使用して、OS ディストリビューション AL2、AL2023、RedHat、CentOS,または Fedora にセキュリティエージェントをインストールします。
- Debian スクリプトを使用して、OS ディストリビューション Ubuntu または Debian にセキュリティエージェントをインストールします。サポートされている Ubuntu および Debian OS ディストリビューションの詳細については、「アーキテクチャ要件を検証する」を参照してください。

RPM installation

> **Important**
>
> GuardDuty セキュリティエージェントの RPM 署名をマシンにインストールする前に検証することをお勧めします。

1. GuardDuty セキュリティエージェントの RPM 署名の検証

   a. テンプレートを準備する

   適切なパブリックキー、x86_64 RPM の署名、arm64 RPM の署名、および Amazon S3 バケットでホストされている RPM スクリプトへの対応するアクセスリンクを使用してコマンドを準備します。RPM スクリプトにアクセスするには AWS リージョン、、AWS アカウント ID、および GuardDuty エージェントバージョンの値を置き換えます。

   - パブリックキー :

     ```
     s3://694911143906-eu-west-1-guardduty-agent-rpm-artifacts/1.17.1/
     publickey.pem
     ```

   - GuardDuty セキュリティエージェントの RPM 署名 :

     x86_64 RPM の署名

     ```
     s3://694911143906-eu-west-1-guardduty-agent-rpm-artifacts/1.17.1/
     x86_64/amazon-guardduty-agent-1.17.1.x86_64.sig
     ```

     arm64 RPM の署名

     ```
     s3://694911143906-eu-west-1-guardduty-agent-rpm-artifacts/1.17.1/arm64/
     amazon-guardduty-agent-1.17.1.arm64.sig
     ```

   - Amazon S3 バケット内の RPM スクリプトへのアクセスリンク:

     x86_64 RPM 用アクセスリンク

     ```
     s3://694911143906-eu-west-1-guardduty-agent-rpm-artifacts/1.17.1/
     x86_64/amazon-guardduty-agent-1.17.1.x86_64.rpm
     ```

arm64 RPM 用アクセスリンク

```
s3://694911143906-eu-west-1-guardduty-agent-rpm-artifacts/1.17.1/arm64/
amazon-guardduty-agent-1.17.1.arm64.rpm
```

AWS リージョン	リージョン名	AWS アカウント ID
eu-west-1	欧州 (アイルランド)	694911143906
us-east-1	米国東部 (バージニア北部)	593207742271
us-west-2	米国西部 (オレゴン)	733349766148
eu-west-3	欧州 (パリ)	665651866788
us-east-2	米国東部 (オハイオ)	307168627858
eu-central-1	欧州 (フランクフルト)	323658145986
ap-northeast-2	アジアパシフィック (ソウル)	914738172881
eu-north-1	欧州 (ストックホルム)	591436053604
ap-east-1	アジアパシフィック (香港)	258348409381
me-south-1	中東 (バーレーン)	536382113932
eu-west-2	欧州 (ロンドン)	892757235363
ap-northeast-1	アジアパシフィック (東京)	533107202818
ap-southeast-1	アジアパシフィック (シンガポール)	174946120834

ap-south-1	アジアパシフィック (ムンバイ)	251508486986
ap-southeast-3	アジアパシフィック (ジャカルタ)	510637619217
sa-east-1	南米 (サンパウロ)	758426053663
ap-northeast-3	アジアパシフィック (大阪)	273192626886
eu-south-1	欧州 (ミラノ)	266869475730
af-south-1	アフリカ (ケープタウン)	197869348890
ap-southeast-2	アジアパシフィック (シドニー)	005257825471
me-central-1	中東 (アラブ首長国連邦)	000014521398
us-west-1	米国西部 (北カリフォルニア)	684579721401
ca-central-1	カナダ (中部)	354763396469
ca-west-1	カナダ西部 (カルガリー)	339712888787
ap-south-2	アジアパシフィック (ハイデラバード)	950823858135
eu-south-2	欧州 (スペイン)	919611009337
eu-central-2	欧州 (チューリッヒ)	529164026651
ap-southeast-4	アジアパシフィック (メルボルン)	251357961535
ap-southeast-7	アジアパシフィック (タイ)	054037130133
il-central-1	イスラエル (テルアビブ)	870907303882

mx-central-1	メキシコ (中部)	982081086614
ap-east-2	アジアパシフィック (台北)	259886477082

b. テンプレートをダウンロードする

次のコマンドで、適切なパブリックキー、x86_64 RPM の署名、arm64 RPM の署名、Amazon S3 バケットでホストされている RPM スクリプトへの対応するアクセスリンクをダウンロードするには、アカウント ID を適切な AWS アカウント ID に、リージョンを現在のリージョンに置き換えてください。

```
aws s3 cp s3://694911143906-eu-west-1-guardduty-agent-rpm-artifacts/1.17.1/
x86_64/amazon-guardduty-agent-1.17.1.x86_64.rpm ./amazon-guardduty-
agent-1.17.1.x86_64.rpm
aws s3 cp s3://694911143906-eu-west-1-guardduty-agent-rpm-artifacts/1.17.1/
x86_64/amazon-guardduty-agent-1.17.1.x86_64.sig ./amazon-guardduty-
agent-1.17.1.x86_64.sig
aws s3 cp s3://694911143906-eu-west-1-guardduty-agent-rpm-artifacts/1.17.1/
publickey.pem ./publickey.pem
```

c. パブリックキーをインポートする

次のコマンドを使用して、パブリックキーをデータベースにインポートします。

```
gpg --import publickey.pem
```

gpg はインポートの成功を示しています。

```
gpg: key 093FF49D: public key "AwsGuardDuty" imported
gpg: Total number processed: 1
gpg:               imported: 1  (RSA: 1)
```

d. 署名を検証する

次のコマンドを使用して、署名を確認します。

```
gpg --verify amazon-guardduty-agent-1.17.1.x86_64.sig amazon-guardduty-
agent-1.17.1.x86_64.rpm
```

検証に成功すると、次の結果のようなメッセージが表示されます。これで、RPM を使用して GuardDuty セキュリティエージェントをインストールできます。

出力の例:

```
gpg: Signature made Fri 17 Nov 2023 07:58:11 PM UTC using ? key ID 093FF49D
gpg: Good signature from "AwsGuardDuty"
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the
 owner.
Primary key fingerprint: 7478 91EF 5378 1334 4456  7603 06C9 06A7 093F F49D
```

検証に失敗した場合は、RPM の署名が改ざんされている可能性があることを意味します。パブリックキーをデータベースから削除して、検証プロセスを再試行する必要があります。

例:

```
gpg: Signature made Fri 17 Nov 2023 07:58:11 PM UTC using ? key ID 093FF49D
gpg: BAD signature from "AwsGuardDuty"
```

次のコマンドを使用して、データベースからパブリックキーを削除します。

```
gpg --delete-keys AwsGuardDuty
```

次に、検証プロセスを再試行します。

2. 「Linux または macOS から SSH で接続する」。
3. 次のコマンドを使用して GuardDuty セキュリティエージェントをインストールします。

```
sudo rpm -ivh amazon-guardduty-agent-1.17.1.x86_64.rpm
```

4. GuardDuty エージェントが正常にインストールされているかどうかを検証します。これらの手順の詳細については、「GuardDuty セキュリティエージェントのインストールステータスの検証」を参照してください。

Debian installation

> **Important**
>
> GuardDuty セキュリティエージェントの Debian 署名をマシンにインストールする前に検証することをお勧めします。

1. GuardDuty セキュリティエージェントの Debian 署名の検証

   a. 適切なパブリックキー、amd64 Debian パッケージの署名、arm64 Debian パッケージの署名、および Amazon S3 バケットでホストされている Debian スクリプトへの対応するアクセスリンク用のテンプレートを準備する

   次のテンプレートで、 AWS リージョン、 AWS アカウント ID、および GuardDuty エージェントバージョンの値を置き換えて、Debian パッケージスクリプトにアクセスします。

   - パブリックキー :

     ```
     s3://694911143906-eu-west-1-guardduty-agent-deb-artifacts/1.17.1/
     publickey.pem
     ```

   - GuardDuty セキュリティエージェントの Debian 署名:

     amd64 の署名

     ```
     s3://694911143906-eu-west-1-guardduty-agent-deb-artifacts/1.17.1/amd64/
     amazon-guardduty-agent-1.17.1.amd64.sig
     ```

     arm64 の署名

     ```
     s3://694911143906-eu-west-1-guardduty-agent-deb-artifacts/1.17.1/arm64/
     amazon-guardduty-agent-1.17.1.arm64.sig
     ```

   - Amazon S3 バケット内の Debian スクリプトへのアクセスリンク:

     amd64 のアクセスリンク

     ```
     s3://694911143906-eu-west-1-guardduty-agent-deb-artifacts/1.17.1/amd64/
     amazon-guardduty-agent-1.17.1.amd64.deb
     ```

arm64 のアクセスリンク

```
s3://694911143906-eu-west-1-guardduty-agent-deb-artifacts/1.17.1/arm64/
amazon-guardduty-agent-1.17.1.arm64.deb
```

AWS リージョン	リージョン名	AWS アカウント ID
eu-west-1	欧州 (アイルランド)	694911143906
us-east-1	米国東部 (バージニア北部)	593207742271
us-west-2	米国西部 (オレゴン)	733349766148
eu-west-3	欧州 (パリ)	665651866788
us-east-2	米国東部 (オハイオ)	307168627858
eu-central-1	欧州 (フランクフルト)	323658145986
ap-northeast-2	アジアパシフィック (ソウル)	914738172881
eu-north-1	欧州 (ストックホルム)	591436053604
ap-east-1	アジアパシフィック (香港)	258348409381
me-south-1	中東 (バーレーン)	536382113932
eu-west-2	欧州 (ロンドン)	892757235363
ap-northeast-1	アジアパシフィック (東京)	533107202818
ap-southeast-1	アジアパシフィック (シンガポール)	174946120834

ap-south-1	アジアパシフィック (ムンバイ)	251508486986
ap-southeast-3	アジアパシフィック (ジャカルタ)	510637619217
sa-east-1	南米 (サンパウロ)	758426053663
ap-northeast-3	アジアパシフィック (大阪)	273192626886
eu-south-1	欧州 (ミラノ)	266869475730
af-south-1	アフリカ (ケープタウン)	197869348890
ap-southeast-2	アジアパシフィック (シドニー)	005257825471
me-central-1	中東 (アラブ首長国連邦)	000014521398
us-west-1	米国西部 (北カリフォルニア)	684579721401
ca-central-1	カナダ (中部)	354763396469
ca-west-1	カナダ西部 (カルガリー)	339712888787
ap-south-2	アジアパシフィック (ハイデラバード)	950823858135
eu-south-2	欧州 (スペイン)	919611009337
eu-central-2	欧州 (チューリッヒ)	529164026651
ap-southeast-4	アジアパシフィック (メルボルン)	251357961535
il-central-1	イスラエル (テルアビブ)	870907303882
mx-central-1	メキシコ (中部)	982081086614

ap-east-2	アジアパシフィック (台北)	259886477082

b. 適切なパブリックキー、amd64 の署名、arm64 の署名、および Amazon S3 バケットでホストされている Debian スクリプトへの対応するアクセスリンクをダウンロードしてください

次のコマンドで、アカウント ID を適切な AWS アカウント ID に置き換え、リージョンを現在のリージョンに置き換えます。

```
aws s3 cp s3://694911143906-eu-west-1-guardduty-agent-deb-artifacts/1.17.1/
amd64/amazon-guardduty-agent-1.17.1.amd64.deb ./amazon-guardduty-
agent-1.17.1.amd64.deb
aws s3 cp s3://694911143906-eu-west-1-guardduty-agent-deb-artifacts/1.17.1/
amd64/amazon-guardduty-agent-1.17.1.amd64.sig ./amazon-guardduty-
agent-1.17.1.amd64.sig
aws s3 cp s3://694911143906-eu-west-1-guardduty-agent-deb-artifacts/1.17.1/
publickey.pem ./publickey.pem
```

c. プライベートキーをデータベースにインポートします。

```
gpg --import publickey.pem
```

gpg はインポートの成功を示しています。

```
gpg: key 093FF49D: public key "AwsGuardDuty" imported
gpg: Total number processed: 1
gpg:               imported: 1  (RSA: 1)
```

d. 署名を検証する

```
gpg --verify amazon-guardduty-agent-1.17.1.amd64.sig amazon-guardduty-
agent-1.17.1.amd64.deb
```

検証が成功すると、次の結果のようなメッセージが表示されます。

出力の例:

```
gpg: Signature made Fri 17 Nov 2023 07:58:11 PM UTC using ? key ID 093FF49D
gpg: Good signature from "AwsGuardDuty"
```

```
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the
 owner.
Primary key fingerprint: 7478 91EF 5378 1334 4456  7603 06C9 06A7 093F F49D
```

これで、Debian を使用して GuardDuty セキュリティエージェントをインストールできます。

ただし、検証が失敗した場合は、Debian パッケージの署名が改ざんされている可能性があることを意味します。

例:

```
gpg: Signature made Fri 17 Nov 2023 07:58:11 PM UTC using ? key ID 093FF49D
gpg: BAD signature from "AwsGuardDuty"
```

次のコマンドを使用して、データベースからパブリックキーを削除します。

```
gpg --delete-keys AwsGuardDuty
```

次に、検証プロセスを再試行します。

2. 「Linux または macOS から SSH で接続する」。
3. 次のコマンドを使用して GuardDuty セキュリティエージェントをインストールします。

```
sudo dpkg -i amazon-guardduty-agent-1.17.1.amd64.deb
```

4. GuardDuty エージェントが正常にインストールされているかどうかを検証します。これらの手順の詳細については、「GuardDuty セキュリティエージェントのインストールステータスの検証」を参照してください。

メモリ不足エラー

Amazon EC2 の GuardDuty セキュリティエージェントを手動でインストールまたは更新するときに `out-of-memory` エラーが発生した場合は、「メモリ不足の問題のトラブルシューティング」を参照してください。

GuardDuty セキュリティエージェントのインストールステータスの検証

GuardDuty セキュリティエージェントをインストールするステップを実行した後、次のステップを使用してエージェントのステータスを検証します。

GuardDuty セキュリティエージェントが正常であるかを検証するには

1. 「Linux または macOS から SSH で接続する」。
2. 次のコマンドを実行して、GuardDuty セキュリティエージェントのステータスを確認します。

```
sudo systemctl status amazon-guardduty-agent
```

セキュリティエージェントのインストールログを表示する場合は、`/var/log/amzn-guardduty-agent/` で確認できます。

ログを表示するには、`sudo journalctl -u amazon-guardduty-agent` を実行します。

Amazon EC2 インスタンスの GuardDuty セキュリティエージェントの手動更新

GuardDuty は、セキュリティエージェントバージョンの更新をリリースします。セキュリティエージェントを手動で管理する場合、Amazon EC2 インスタンスのエージェントを更新する責任がユーザーにあります。新しいエージェントバージョンについては、Amazon EC2 インスタンスの「GuardDuty セキュリティエージェントのリリースバージョン」を参照してください。新しいエージェントバージョンリリースに関する通知を受け取るには、「Amazon SNS GuardDuty のお知らせへのサブスクライブ」を参照してください。

Amazon EC2 インスタンスのセキュリティエージェントを手動で更新するには

セキュリティエージェントを更新するプロセスは、セキュリティエージェントをインストールするプロセスと同じです。エージェントのインストールに使用した方法に応じて、Amazon EC2 インスタンスの「セキュリティエージェントの手動インストール」のステップを実行できます。

方法 1 を使用する場合 - を使用すると AWS Systems Manager、Run コマンドを使用してセキュリティエージェントを更新できます。更新するエージェントバージョンを使用します。

「メソッド 2 - Linux パッケージマネージャーを使用する」を使用する場合は、「セキュリティエージェントの手動インストール」セクションに記載されたスクリプトを使用できます。スクリプトには、既に最新のエージェントリリースバージョンが含まれています。最近リリースされたエージェントバージョンについては、「Amazon EC2 インスタンス用の GuardDuty セキュリティエージェントバージョン」を参照してください。

セキュリティエージェントを更新した後、ログを確認してインストールステータスを確認できます。詳細については、「GuardDuty セキュリティエージェントのインストールステータスの検証」を参照してください。

Bottlerocket での GuardDuty セキュリティエージェントの管理 (Amazon EC2 での Amazon ECS)

> **Important**
>
> このページは、以下のバリアントを含む Bottlerocket ECS 最適化 AMIsバージョン v1.62.1以降にのみ適用されます。
>
> - `bottlerocket-aws-ecs-2-x86_64` / `bottlerocket-aws-ecs-2-aarch64`
> - `bottlerocket-aws-ecs-3-x86_64` / `bottlerocket-aws-ecs-3-aarch64`
> - `bottlerocket-aws-ecs-2-nvidia` / `bottlerocket-aws-ecs-2-nvidia-fips`
> - `bottlerocket-aws-ecs-3-nvidia` / `bottlerocket-aws-ecs-3-nvidia-fips`
>
> Amazon EKS クラスターの Bottlerocket インスタンスの場合、Runtime Monitoring は別のセットアップを使用します。「Amazon EKS クラスターサポートの前提条件」を参照してください。

このページでは、GuardDuty セキュリティエージェントが Bottlerocket ECS 最適化 AMIs でどのように実行されるか、および GuardDuty がそれをデプロイする方法について説明します。Bottlerocket では、エージェントはインストールされた RPM または Debian パッケージとしてではなくホストコンテナとして実行されます。Bottlerocket は、変更不可能なルートファイルシステムを備えたコンテナ最適化 Linux OS であるため、従来のパッケージマネージャーを使用しません。GuardDuty は、管理する Amazon Elastic Container Registry (Amazon ECR) リポジトリでエージェントコンテナイメージをホストします。自動エージェント設定で Runtime Monitoring を有効にすると、GuardDuty は AWS Systems Manager Distributor を使用してエージェントホストコンテナをデプロイします。

前提条件

続行する前に、必ずすべての「ECS-EC2 Bottlerocket サポートの前提条件」に従ってください。

自動エージェント設定の有効化

Bottlerocket ECS-EC2 インスタンスは、標準の Amazon EC2 インスタンスと同じ SSM ドキュメントと自動エージェント設定を使用します。Bottlerocket インスタンスの GuardDuty 自動エージェントを有効にするには、「」の手順に従いますAmazon EC2 リソースの自動エージェント。これには、以下が含まれます。

- マルチアカウント環境で GuardDuty エージェントを有効にする。
- スタンドアロンアカウントで GuardDuty 自動エージェントを有効にする。

包含タグまたは除外タグを使用する際のその他の前提条件については、「」を参照してください自動エージェント設定を使用する場合。

エージェントの検証

自動エージェント設定を有効にしたら、Bottlerocket インスタンスでエージェントが実行されていることを確認します。

1. SSM Session Manager を使用してインスタンスに接続します。
2. 管理者コンテナ を入力しますenter-admin-container。
3. 次のいずれかを使用してエージェントログを確認します。
   - `sheltie journalctl -u host-containers@amazon-guardduty-agent`
   - シェルティセッション (`sudo sheltie`): `ls /var/log/amzn-guardduty-agent/`。

Amazon EC2 インスタンスのカバレッジ統計を確認する方法については、「」を参照してくださいカバレッジ統計の確認。

カバレッジステータスが異常と表示される場合は、「」を参照してくださいBottlerocket ECS-EC2 ランタイムカバレッジの問題のトラブルシューティング。

Fargate の自動セキュリティエージェントの管理 (Amazon ECS のみ)

Runtime Monitoring は、GuardDuty を介してのみ Amazon ECS クラスター (AWS Fargate) のセキュリティエージェントの管理をサポートします。Amazon ECS クラスターでのセキュリティエージェントの手動管理はサポートされていません。

このセクションのステップに進む前に、必ず「AWS Fargate (Amazon ECS のみ) サポートの前提条件」に従ってください。

のアプローチに基づいてFargate での Amazon ECS へのセキュリティエージェントのデプロイ、リソースに対して GuardDuty 自動エージェントを有効にする任意の方法を選択します。

マルチアカウント環境用の GuardDuty エージェントの設定

マルチアカウント環境では、委任 GuardDuty 管理者アカウントのみが、メンバーアカウントの自動エージェント設定を有効または無効にすることや、組織内のメンバーアカウントに属する Amazon ECS クラスターの自動エージェント設定を管理することができます。GuardDuty メンバーアカウントからは、この設定を変更できません。委任 GuardDuty 管理者アカウントは、 を使用してメンバーアカウントを管理します AWS Organizations。マルチアカウント環境の詳細については、「GuardDuty での複数のアカウントの管理」を参照してください。

委任 GuardDuty 管理者アカウントの自動エージェント設定の有効化

Manage for all Amazon ECS clusters (account level)

Runtime Monitoring の [すべてのアカウントについて有効にする] を選択した場合、次のオプションがあります。

- [自動エージェント設定] セクションで [すべてのアカウントについて有効にする] を選択します。GuardDuty は、起動されるすべての Amazon ECS タスクについてセキュリティエージェントをデプロイして管理します。
- [アカウントを手動で設定] を選択します。

[Runtime Monitoring] セクションで [アカウントを手動で設定] を選択した場合、次の操作を行います。

1. [自動エージェント設定] セクションで [アカウントを手動で設定する] を選択します。
2. [委任 GuardDuty 管理者 (このアカウント)] セクションで [有効にする] を選択します。

[保存] を選択します。

GuardDuty でサービスの一部であるタスクをモニタリングする場合は、Runtime Monitoring を有効にした後に新しいサービスデプロイが必要です。Runtime Monitoring を有効にする前に特定の ECS サービスの最後のデプロイが開始された場合は、サービスを再起動するか、`forceNewDeployment` を使用してサービスを更新できます。

サービスを更新するステップについては、以下のリソースを参照してください。

- 「Amazon Elastic Container Service デベロッパーガイド」の「コンソールを使用した Amazon ECS サービスの更新」。
- 「Amazon Elastic Container Service API リファレンス」の「UpdateService」。
- 「AWS CLI コマンドリファレンス」の「update-service」。

Manage for all Amazon ECS clusters but exclude some of the clusters (cluster level)

1. この Amazon ECS クラスターに、キーと値のペアを`GuardDutyManaged-false` というタグを追加します。
2. 信頼できるエンティティ以外は、タグを変更しないようにしてください。「AWS Organizations ユーザーガイド」の「認められている原則による場合を除き、タグが変更されないようにする」に記載されているポリシーが、ここに適用できるように変更されました。

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "DenyModifyTagsIfResAuthzTagAndPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "ecs:ResourceTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "ecs:ResourceTag/GuardDutyManaged": false
                }
            }
        },
```

```
        {
            "Sid": "DenyModifyResAuthzTagIfPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:RequestTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "ForAnyValue:StringEquals": {
                    "aws:TagKeys": [
                        "GuardDutyManaged"
                    ]
                }
            }
        },
        {
            "Sid": "DenyModifyTagsIfPrinTagNotExists",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "aws:PrincipalTag/GuardDutyManaged": true
                }
            }
        }
```

```
    ]
}
```

3. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
4. ナビゲーションペインで、[Runtime Monitoring] を選択します。
5.

> **Note**
>
> アカウントの自動エージェントの設定を有効にする前に、必ず Amazon ECS クラスターに除外タグを追加してください。追加しないと、GuardDuty サイドカーコンテナが起動される Amazon ECS タスクのすべてのコンテナに接続されます。

   [設定] タブの [自動エージェント設定] で [有効化] を選択します。

   除外されていない Amazon ECS クラスターの場合、GuardDuty がサイドカーコンテナ内のセキュリティエージェントのデプロイを管理します。
6. [保存] を選択します。
7. GuardDuty でサービスの一部であるタスクをモニタリングする場合は、Runtime Monitoring を有効にした後に新しいサービスデプロイが必要です。Runtime Monitoring を有効にする前に特定の ECS サービスの最後のデプロイが開始された場合は、サービスを再起動するか、`forceNewDeployment` を使用してサービスを更新できます。

   サービスを更新するステップについては、以下のリソースを参照してください。

   - 「Amazon Elastic Container Service デベロッパーガイド」の「コンソールを使用した Amazon ECS サービスの更新」。
   - 「Amazon Elastic Container Service API リファレンス」の「UpdateService」。
   - 「AWS CLI コマンドリファレンス」の「update-service」。

Manage for selective (inclusion only) Amazon ECS clusters (cluster level)

1. すべてのタスクを含む Amazon ECS クラスターにタグを追加します。キーと値のペアは `GuardDutyManaged–true` である必要があります。
2. 信頼できるエンティティ以外は、これらのタグを変更しないようにしてください。「AWS Organizations ユーザーガイド」の「認められている原則による場合を除き、タグが変更されないようにする」に記載されているポリシーが、ここに適用できるように変更されました。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "DenyModifyTagsIfResAuthzTagAndPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "ecs:ResourceTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "ecs:ResourceTag/GuardDutyManaged": false
                }
            }
        },
        {
            "Sid": "DenyModifyResAuthzTagIfPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:RequestTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
```

```
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "ForAnyValue:StringEquals": {
                    "aws:TagKeys": [
                        "GuardDutyManaged"
                    ]
                }
            }
        },
        {
            "Sid": "DenyModifyTagsIfPrinTagNotExists",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "aws:PrincipalTag/GuardDutyManaged": true
                }
            }
        }
    ]
}
```

> **Note**
>
> Amazon ECS クラスターに包含タグを使用する場合、自動エージェント設定によって GuardDuty エージェントを明示的に有効にする必要はありません。

3. GuardDuty でサービスの一部であるタスクをモニタリングする場合は、Runtime Monitoring を有効にした後に新しいサービスデプロイが必要です。Runtime Monitoring を有効にする

前に特定の ECS サービスの最後のデプロイが開始された場合は、サービスを再起動するか、`forceNewDeployment` を使用してサービスを更新できます。

サービスを更新するステップについては、以下のリソースを参照してください。

- 「Amazon Elastic Container Service デベロッパーガイド」の「コンソールを使用した Amazon ECS サービスの更新」。
- 「Amazon Elastic Container Service API リファレンス」の「UpdateService」。
- 「AWS CLI コマンドリファレンス」の「update-service」。

すべてのメンバーアカウントの自動有効化

Manage for all Amazon ECS clusters (account level)

以下の手順は、[Runtime Monitoring] セクションで [すべてのアカウントで有効化] を選択したことを前提としています。

1. [自動エージェント設定] セクションで [すべてのアカウントについて有効にする] を選択します。GuardDuty は、起動されるすべての Amazon ECS タスクについてセキュリティエージェントをデプロイして管理します。
2. [保存] を選択します。
3. GuardDuty でサービスの一部であるタスクをモニタリングする場合は、Runtime Monitoring を有効にした後に新しいサービスデプロイが必要です。Runtime Monitoring を有効にする前に特定の ECS サービスの最後のデプロイが開始された場合は、サービスを再起動するか、`forceNewDeployment` を使用してサービスを更新できます。

   サービスを更新するステップについては、以下のリソースを参照してください。

   - 「Amazon Elastic Container Service デベロッパーガイド」の「コンソールを使用した Amazon ECS サービスの更新」。
   - 「Amazon Elastic Container Service API リファレンス」の「UpdateService」。
   - 「AWS CLI コマンドリファレンス」の「update-service」。

Manage for all Amazon ECS clusters but exclude some of the clusters (cluster level)

1. この Amazon ECS クラスターに、キーと値のペアを`GuardDutyManaged-false` というタグを追加します。

2. 信頼できるエンティティ以外は、タグを変更しないようにしてください。「AWS Organizations ユーザーガイド」の「認められている原則による場合を除き、タグが変更されないようにする」に記載されているポリシーが、ここに適用できるように変更されました。

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "DenyModifyTagsIfResAuthzTagAndPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "ecs:ResourceTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "ecs:ResourceTag/GuardDutyManaged": false
                }
            }
        },
        {
            "Sid": "DenyModifyResAuthzTagIfPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
```

```
                "StringNotEquals": {
                    "aws:RequestTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "ForAnyValue:StringEquals": {
                    "aws:TagKeys": [
                        "GuardDutyManaged"
                    ]
                }
            }
        },
        {
            "Sid": "DenyModifyTagsIfPrinTagNotExists",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "aws:PrincipalTag/GuardDutyManaged": true
                }
            }
        }
    ]
}
```

3. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
4. ナビゲーションペインで、[Runtime Monitoring] を選択します。

5.

> **Note**
>
> アカウントの自動エージェントの設定を有効にする前に、必ず Amazon ECS クラスターに除外タグを追加してください。追加しないと、GuardDuty サイドカーコンテナが起動される Amazon ECS タスクのすべてのコンテナに接続されます。

   [設定] タブで、[編集] を選択します。
6. [自動エージェント設定] セクションで [すべてのアカウントについて有効にする] を選択します。

   除外されていない Amazon ECS クラスターの場合、GuardDuty がサイドカーコンテナ内のセキュリティエージェントのデプロイを管理します。
7. [保存] を選択します。
8. GuardDuty でサービスの一部であるタスクをモニタリングする場合は、Runtime Monitoring を有効にした後に新しいサービスデプロイが必要です。Runtime Monitoring を有効にする前に特定の ECS サービスの最後のデプロイが開始された場合は、サービスを再起動するか、`forceNewDeployment` を使用してサービスを更新できます。

   サービスを更新するステップについては、以下のリソースを参照してください。

   - 「Amazon Elastic Container Service デベロッパーガイド」の「コンソールを使用した Amazon ECS サービスの更新」。
   - 「Amazon Elastic Container Service API リファレンス」の「UpdateService」。
   - 「AWS CLI コマンドリファレンス」の「update-service」。

Manage for selective (inclusion-only) Amazon ECS clusters (cluster level)

Runtime Monitoring をどのように有効にするかにかかわらず、以下の手順は組織内のすべてのメンバーアカウントで選択する Amazon ECS Fargate タスクを監視するのに役立ちます。

1. [自動エージェント設定] セクションの設定はどれも有効にしないでください。Runtime Monitoring の設定は、前の手順で選択したものと同じにします。
2. [保存] を選択します。
3. 信頼できるエンティティ以外は、これらのタグを変更しないようにしてください。「AWS Organizations ユーザーガイド」の「認められている原則による場合を除き、タグが変更されないようにする」に記載されているポリシーが、ここに適用できるように変更されました。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "DenyModifyTagsIfResAuthzTagAndPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "ecs:ResourceTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "ecs:ResourceTag/GuardDutyManaged": false
                }
            }
        },
        {
            "Sid": "DenyModifyResAuthzTagIfPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:RequestTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
```

```
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "ForAnyValue:StringEquals": {
                    "aws:TagKeys": [
                        "GuardDutyManaged"
                    ]
                }
            }
        },
        {
            "Sid": "DenyModifyTagsIfPrinTagNotExists",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "aws:PrincipalTag/GuardDutyManaged": true
                }
            }
        }
    ]
}
```

> **Note**
>
> Amazon ECS クラスターにインクルージョンタグを使用する場合、[GuardDuty エージェントの自動管理] を明示的に有効にする必要はありません。

4. GuardDuty でサービスの一部であるタスクをモニタリングする場合は、Runtime Monitoring を有効にした後に新しいサービスデプロイが必要です。Runtime Monitoring を有効にする

前に特定の ECS サービスの最後のデプロイが開始された場合は、サービスを再起動するか、`forceNewDeployment` を使用してサービスを更新できます。

サービスを更新するステップについては、以下のリソースを参照してください。

- 「Amazon Elastic Container Service デベロッパーガイド」の「コンソールを使用した Amazon ECS サービスの更新」。
- 「Amazon Elastic Container Service API リファレンス」の「UpdateService」。
- 「AWS CLI コマンドリファレンス」の「update-service」。

既存のアクティブなメンバーアカウントでエージェントの自動設定を有効にする

Manage for all Amazon ECS clusters (account level)

1. [Runtime Monitoring] ページの [設定] タブで、自動エージェント 設定の現在のステータスを表示できます。
2. [自動エージェント設定] ペイン内の [アクティブメンバーアカウント] セクションで、[アクション] を選択します。
3. [アクション] から、[すべての既存のアクティブなメンバーアカウントについて有効にする] を選択します。
4. [確認] を選択します。
5. GuardDuty でサービスの一部であるタスクをモニタリングする場合は、Runtime Monitoring を有効にした後に新しいサービスデプロイが必要です。Runtime Monitoring を有効にする前に特定の ECS サービスの最後のデプロイが開始された場合は、サービスを再起動するか、`forceNewDeployment` を使用してサービスを更新できます。

   サービスを更新するステップについては、以下のリソースを参照してください。

   - 「Amazon Elastic Container Service デベロッパーガイド」の「コンソールを使用した Amazon ECS サービスの更新」。
   - 「Amazon Elastic Container Service API リファレンス」の「UpdateService」。
   - 「AWS CLI コマンドリファレンス」の「update-service」。

Manage for all Amazon ECS clusters but exclude some of the clusters (cluster level)

1. この Amazon ECS クラスターに、キーと値のペアを`GuardDutyManaged-false` というタグを追加します。

2. 信頼できるエンティティ以外は、タグを変更しないようにしてください。「AWS Organizations ユーザーガイド」の「認められている原則による場合を除き、タグが変更されないようにする」に記載されているポリシーが、ここに適用できるように変更されました。

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "DenyModifyTagsIfResAuthzTagAndPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "ecs:ResourceTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "ecs:ResourceTag/GuardDutyManaged": false
                }
            }
        },
        {
            "Sid": "DenyModifyResAuthzTagIfPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
```

```
                "StringNotEquals": {
                    "aws:RequestTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "ForAnyValue:StringEquals": {
                    "aws:TagKeys": [
                        "GuardDutyManaged"
                    ]
                }
            }
        },
        {
            "Sid": "DenyModifyTagsIfPrinTagNotExists",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "aws:PrincipalTag/GuardDutyManaged": true
                }
            }
        }
    ]
}
```

3. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
4. ナビゲーションペインで、[Runtime Monitoring] を選択します。

5.

> **Note**
>
> アカウントの自動エージェントの設定を有効にする前に、必ず Amazon ECS クラスターに除外タグを追加してください。追加しないと、GuardDuty サイドカーコンテナが起動される Amazon ECS タスクのすべてのコンテナに接続されます。

[自動エージェント設定] セクションの [設定] タブから [アクティブなメンバーアカウント] で [アクション] を選択します。

6. [アクション] から、[すべてのアクティブなメンバーアカウントについて有効にする] を選択します。

除外されていない Amazon ECS クラスターの場合、GuardDuty がサイドカーコンテナ内のセキュリティエージェントのデプロイを管理します。

7. [確認] を選択します。
8. GuardDuty でサービスの一部であるタスクをモニタリングする場合は、Runtime Monitoring を有効にした後に新しいサービスデプロイが必要です。Runtime Monitoring を有効にする前に特定の ECS サービスの最後のデプロイが開始された場合は、サービスを再起動するか、`forceNewDeployment` を使用してサービスを更新できます。

サービスを更新するステップについては、以下のリソースを参照してください。

- 「Amazon Elastic Container Service デベロッパーガイド」の「コンソールを使用した Amazon ECS サービスの更新」。
- 「Amazon Elastic Container Service API リファレンス」の「UpdateService」。
- 「AWS CLI コマンドリファレンス」の「update-service」。

Manage for selective (inclusion only) Amazon ECS clusters (cluster level)

1. すべてのタスクを含む Amazon ECS クラスターにタグを追加します。キーと値のペアは `GuardDutyManaged–true` である必要があります。
2. 信頼できるエンティティ以外は、これらのタグを変更しないようにしてください。「AWS Organizations ユーザーガイド」の「認められている原則による場合を除き、タグが変更されないようにする」に記載されているポリシーが、ここに適用できるように変更されました。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "DenyModifyTagsIfResAuthzTagAndPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "ecs:ResourceTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "ecs:ResourceTag/GuardDutyManaged": false
                }
            }
        },
        {
            "Sid": "DenyModifyResAuthzTagIfPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:RequestTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
```

```
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "ForAnyValue:StringEquals": {
                    "aws:TagKeys": [
                        "GuardDutyManaged"
                    ]
                }
            }
        },
        {
            "Sid": "DenyModifyTagsIfPrinTagNotExists",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "aws:PrincipalTag/GuardDutyManaged": true
                }
            }
        }
    ]
}
```

> **Note**
>
> Amazon ECS クラスターに包含タグを使用する場合、[自動エージェント設定] を明示的に有効にする必要はありません。

3. GuardDuty でサービスの一部であるタスクをモニタリングする場合は、Runtime Monitoring を有効にした後に新しいサービスデプロイが必要です。Runtime Monitoring を有効にする

前に特定の ECS サービスの最後のデプロイが開始された場合は、サービスを再起動するか、`forceNewDeployment` を使用してサービスを更新できます。

サービスを更新するステップについては、以下のリソースを参照してください。

- 「Amazon Elastic Container Service デベロッパーガイド」の「コンソールを使用した Amazon ECS サービスの更新」。
- 「Amazon Elastic Container Service API リファレンス」の「UpdateService」。
- 「AWS CLI コマンドリファレンス」の「update-service」。

新規メンバー用の自動エージェント設定を自動有効化

Manage for all Amazon ECS clusters (account level)

1. [Runtime Monitoring] ページで、[編集] を選択して既存の設定を更新します。
2. [自動エージェント設定] セクションで [すべてのアカウントについて有効にする] を選択します。
3. [保存] を選択します。
4. GuardDuty でサービスの一部であるタスクをモニタリングする場合は、Runtime Monitoring を有効にした後に新しいサービスデプロイが必要です。Runtime Monitoring を有効にする前に特定の ECS サービスの最後のデプロイが開始された場合は、サービスを再起動するか、`forceNewDeployment` を使用してサービスを更新できます。

   サービスを更新するステップについては、以下のリソースを参照してください。

   - 「Amazon Elastic Container Service デベロッパーガイド」の「コンソールを使用した Amazon ECS サービスの更新」。
   - 「Amazon Elastic Container Service API リファレンス」の「UpdateService」。
   - 「AWS CLI コマンドリファレンス」の「update-service」。

Manage for all Amazon ECS clusters but exclude some of the clusters (cluster level)

1. この Amazon ECS クラスターに、キーと値のペアを`GuardDutyManaged-false` というタグを追加します。
2. 信頼できるエンティティ以外は、タグを変更しないようにしてください。「AWS Organizations ユーザーガイド」の「認められている原則による場合を除き、タグが変更されないようにする」に記載されているポリシーが、ここに適用できるように変更されました。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "DenyModifyTagsIfResAuthzTagAndPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "ecs:ResourceTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "ecs:ResourceTag/GuardDutyManaged": false
                }
            }
        },
        {
            "Sid": "DenyModifyResAuthzTagIfPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:RequestTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
```

```
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "ForAnyValue:StringEquals": {
                    "aws:TagKeys": [
                        "GuardDutyManaged"
                    ]
                }
            }
        },
        {
            "Sid": "DenyModifyTagsIfPrinTagNotExists",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "aws:PrincipalTag/GuardDutyManaged": true
                }
            }
        }
    ]
}
```

3. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
4. ナビゲーションペインで、[Runtime Monitoring] を選択します。
5.

> **Note**
>
> アカウントの自動エージェントの設定を有効にする前に、必ず Amazon ECS クラスターに除外タグを追加してください。追加しないと、GuardDuty サイドカーコンテナが起動される Amazon ECS タスクのすべてのコンテナに接続されます。

[設定] タブの [自動エージェント設定] セクションで、[新しいメンバーアカウントについて自動的に有効にする] を選択します。

除外されていない Amazon ECS クラスターの場合、GuardDuty がサイドカーコンテナ内のセキュリティエージェントのデプロイを管理します。

6. [保存] を選択します。
7. GuardDuty でサービスの一部であるタスクをモニタリングする場合は、Runtime Monitoring を有効にした後に新しいサービスデプロイが必要です。Runtime Monitoring を有効にする前に特定の ECS サービスの最後のデプロイが開始された場合は、サービスを再起動するか、`forceNewDeployment` を使用してサービスを更新できます。

   サービスを更新するステップについては、以下のリソースを参照してください。

   - 「Amazon Elastic Container Service デベロッパーガイド」の「コンソールを使用した Amazon ECS サービスの更新」。
   - 「Amazon Elastic Container Service API リファレンス」の「UpdateService」。
   - 「AWS CLI コマンドリファレンス」の「update-service」。

Manage for selective (inclusion only) Amazon ECS clusters (cluster level)

1. すべてのタスクを含む Amazon ECS クラスターにタグを追加します。キーと値のペアは `GuardDutyManaged–true` である必要があります。
2. 信頼できるエンティティ以外は、これらのタグを変更しないようにしてください。「AWS Organizations ユーザーガイド」の「認められている原則による場合を除き、タグが変更されないようにする」に記載されているポリシーが、ここに適用できるように変更されました。

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "DenyModifyTagsIfResAuthzTagAndPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
```

```
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "ecs:ResourceTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "ecs:ResourceTag/GuardDutyManaged": false
                }
            }
        },
        {
            "Sid": "DenyModifyResAuthzTagIfPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:RequestTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "ForAnyValue:StringEquals": {
                    "aws:TagKeys": [
                        "GuardDutyManaged"
                    ]
                }
            }
        },
        {
            "Sid": "DenyModifyTagsIfPrinTagNotExists",
            "Effect": "Deny",
```

```
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "aws:PrincipalTag/GuardDutyManaged": true
                }
            }
        }
    ]
}
```

> **Note**
>
> Amazon ECS クラスターに包含タグを使用する場合、[自動エージェント設定] を明示的に有効にする必要はありません。

3. GuardDuty でサービスの一部であるタスクをモニタリングする場合は、Runtime Monitoring を有効にした後に新しいサービスデプロイが必要です。Runtime Monitoring を有効にする前に特定の ECS サービスの最後のデプロイが開始された場合は、サービスを再起動するか、`forceNewDeployment` を使用してサービスを更新できます。

   サービスを更新するステップについては、以下のリソースを参照してください。

   - 「Amazon Elastic Container Service デベロッパーガイド」の「コンソールを使用した Amazon ECS サービスの更新」。
   - 「Amazon Elastic Container Service API リファレンス」の「UpdateService」。
   - 「AWS CLI コマンドリファレンス」の「update-service」。

アクティブなメンバーアカウントの自動エージェント設定を選択的に有効にする

Manage for all Amazon ECS (account level)

1. [アカウント] ページで、Runtime Monitoring 自動エージェント設定 (ECS-Fargate) を有効にするアカウントを選択します。複数のアカウントを選択できます。この手順で選択したアカウントで Runtime Monitoring が既に有効になっていることを確認してください。
2. [保護プランの編集] から、適切なオプションを選択して [Runtime Monitoring 自動エージェントを設定 (ECS-Fargate)] を有効にします。
3. [確認] を選択します。
4. GuardDuty でサービスの一部であるタスクをモニタリングする場合は、Runtime Monitoring を有効にした後に新しいサービスデプロイが必要です。Runtime Monitoring を有効にする前に特定の ECS サービスの最後のデプロイが開始された場合は、サービスを再起動するか、`forceNewDeployment` を使用してサービスを更新できます。

   サービスを更新するステップについては、以下のリソースを参照してください。

   - 「Amazon Elastic Container Service デベロッパーガイド」の「コンソールを使用した Amazon ECS サービスの更新」。
   - 「Amazon Elastic Container Service API リファレンス」の「UpdateService」。
   - 「AWS CLI コマンドリファレンス」の「update-service」。

Manage for all Amazon ECS clusters but exclude some of the clusters (cluster level)

1. この Amazon ECS クラスターに、キーと値のペアを`GuardDutyManaged-false` というタグを追加します。
2. 信頼できるエンティティ以外は、タグを変更しないようにしてください。「AWS Organizations ユーザーガイド」の「認められている原則による場合を除き、タグが変更されないようにする」に記載されているポリシーが、ここに適用できるように変更されました。

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "DenyModifyTagsIfResAuthzTagAndPrinTagDontMatch",
```

```
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "ecs:ResourceTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "ecs:ResourceTag/GuardDutyManaged": false
                }
            }
        },
        {
            "Sid": "DenyModifyResAuthzTagIfPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:RequestTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "ForAnyValue:StringEquals": {
                    "aws:TagKeys": [
                        "GuardDutyManaged"
                    ]
                }
            }
        },
```

```
        {
            "Sid": "DenyModifyTagsIfPrinTagNotExists",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "aws:PrincipalTag/GuardDutyManaged": true
                }
            }
        }
    ]
}
```

3. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
4. ナビゲーションペインで、[Runtime Monitoring] を選択します。
5.

> **Note**
>
> アカウントの GuardDuty エージェントの自動管理を有効にする前に、必ず Amazon ECS クラスターに除外タグを追加してください。追加しないと、GuardDuty サイドカーコンテナが起動される Amazon ECS タスク内のコンテナすべてにアタッチされます。

[アカウント] ページで、Runtime Monitoring 自動エージェント設定 (ECS-Fargate) を有効にするアカウントを選択します。複数のアカウントを選択できます。この手順で選択したアカウントで Runtime Monitoring が既に有効になっていることを確認してください。

除外されていない Amazon ECS クラスターの場合、GuardDuty がサイドカーコンテナ内のセキュリティエージェントのデプロイを管理します。

6. [保護プランの編集] から、適切なオプションを選択して [Runtime Monitoring 自動エージェントを設定 (ECS-Fargate)] を有効にします。
7. [保存] を選択します。
8. GuardDuty でサービスの一部であるタスクをモニタリングする場合は、Runtime Monitoring を有効にした後に新しいサービスデプロイが必要です。Runtime Monitoring を有効にする前に特定の ECS サービスの最後のデプロイが開始された場合は、サービスを再起動するか、`forceNewDeployment` を使用してサービスを更新できます。

   サービスを更新するステップについては、以下のリソースを参照してください。

   - 「Amazon Elastic Container Service デベロッパーガイド」の「コンソールを使用した Amazon ECS サービスの更新」。
   - 「Amazon Elastic Container Service API リファレンス」の「UpdateService」。
   - 「AWS CLI コマンドリファレンス」の「update-service」。

Manage for selective (inclusion only) Amazon ECS clusters (cluster level)

1. 監視する Amazon ECS クラスターを含む選択したアカウントで、[自動エージェント設定] (または [Runtime Monitoring 自動エージェント設定 (ECS-Fargate)]) を有効にしていないことを確認してください。
2. すべてのタスクを含む Amazon ECS クラスターにタグを追加します。キーと値のペアは `GuardDutyManaged–true` である必要があります。
3. 信頼できるエンティティ以外は、これらのタグを変更しないようにしてください。「AWS Organizations ユーザーガイド」の「認められている原則による場合を除き、タグが変更されないようにする」に記載されているポリシーが、ここに適用できるように変更されました。

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "DenyModifyTagsIfResAuthzTagAndPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
```

```
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "ecs:ResourceTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "ecs:ResourceTag/GuardDutyManaged": false
                }
            }
        },
        {
            "Sid": "DenyModifyResAuthzTagIfPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:RequestTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "ForAnyValue:StringEquals": {
                    "aws:TagKeys": [
                        "GuardDutyManaged"
                    ]
                }
            }
        },
        {
            "Sid": "DenyModifyTagsIfPrinTagNotExists",
            "Effect": "Deny",
            "Action": [
```

```
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:PrincipalArn": "arn:aws:iam::123456789012:role/
org-admins/iam-admin"
                },
                "Null": {
                    "aws:PrincipalTag/GuardDutyManaged": true
                }
            }
        }
    ]
}
```

> **Note**
>
> Amazon ECS クラスターに包含タグを使用する場合、[自動エージェント設定] を明示的に有効にする必要はありません。

4. GuardDuty でサービスの一部であるタスクをモニタリングする場合は、Runtime Monitoring を有効にした後に新しいサービスデプロイが必要です。Runtime Monitoring を有効にする前に特定の ECS サービスの最後のデプロイが開始された場合は、サービスを再起動するか、`forceNewDeployment` を使用してサービスを更新できます。

   サービスを更新するステップについては、以下のリソースを参照してください。

   - 「Amazon Elastic Container Service デベロッパーガイド」の「コンソールを使用した Amazon ECS サービスの更新」。
   - 「Amazon Elastic Container Service API リファレンス」の「UpdateService」。
   - 「AWS CLI コマンドリファレンス」の「update-service」。

スタンドアロンアカウントの GuardDuty エージェント の設定

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[Runtime Monitoring] を選択します。
3. [設定] タブ:

   a. すべての Amazon ECS クラスターの自動エージェント設定を管理するには (アカウントレベル)

      [AWS Fargate (ECS のみ)] の [自動エージェント設定] セクションで [有効にする] を選択します。新しい Fargate Amazon ECS タスクが起動すると、GuardDuty がセキュリティエージェントのデプロイを管理します。

      - [保存] を選択します。

   b. Amazon ECS クラスターの一部を除外して自動エージェント設定を管理するには (クラスターレベル)

      i. すべてのタスクを除外する Amazon ECS クラスターにタグを追加します。キーと値のペアは `GuardDutyManaged–false` である必要があります。

      ii. 信頼できるエンティティ以外は、これらのタグを変更しないようにしてください。「AWS Organizations ユーザーガイド」の「認められている原則による場合を除き、タグが変更されないようにする」に記載されているポリシーが、ここに適用できるように変更されました。

      JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "DenyModifyTagsIfResAuthzTagAndPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
```

```
            ],
            "Condition": {
                "StringNotEquals": {
                    "ecs:ResourceTag/GuardDutyManaged":
"${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn":
"arn:aws:iam::123456789012:role/org-admins/iam-admin"
                },
                "Null": {
                    "ecs:ResourceTag/GuardDutyManaged": false
                }
            }
        },
        {
            "Sid": "DenyModifyResAuthzTagIfPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:RequestTag/GuardDutyManaged":
"${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn":
"arn:aws:iam::123456789012:role/org-admins/iam-admin"
                },
                "ForAnyValue:StringEquals": {
                    "aws:TagKeys": [
                        "GuardDutyManaged"
                    ]
                }
            }
        },
        {
            "Sid": "DenyModifyTagsIfPrinTagNotExists",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
```

```
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:PrincipalArn":
 "arn:aws:iam::123456789012:role/org-admins/iam-admin"
                },
                "Null": {
                    "aws:PrincipalTag/GuardDutyManaged": true
                }
            }
        }
    ]
}
```

iii. [設定] タブの [自動エージェント設定] セクションで [有効化] を選択します。

> **Note**
>
> アカウントの GuardDuty エージェントの自動管理を有効にする前に、必ず Amazon ECS クラスターに除外タグを追加してください。追加しないと、セキュリティエージェントが対応する Amazon ECS クラスター内で開始されるすべてのタスクにデプロイされます。

除外されていない Amazon ECS クラスターの場合、GuardDuty がサイドカーコンテナ内のセキュリティエージェントのデプロイを管理します。

iv. [保存] を選択します。

c. Amazon ECS クラスターの一部を含めて自動エージェント設定を管理するには (クラスターレベル)

i. すべてのタスクを含む Amazon ECS クラスターにタグを追加します。キーと値のペアは `GuardDutyManaged–true` である必要があります。

ii. 信頼できるエンティティ以外は、これらのタグを変更しないようにしてください。「AWS Organizations ユーザーガイド」の「認められている原則による場合を除き、タグが変更されないようにする」に記載されているポリシーが、ここに適用できるように変更されました。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "DenyModifyTagsIfResAuthzTagAndPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "ecs:ResourceTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
                    "aws:PrincipalArn":
 "arn:aws:iam::123456789012:role/org-admins/iam-admin"
                },
                "Null": {
                    "ecs:ResourceTag/GuardDutyManaged": false
                }
            }
        },
        {
            "Sid": "DenyModifyResAuthzTagIfPrinTagDontMatch",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:RequestTag/GuardDutyManaged":
 "${aws:PrincipalTag/GuardDutyManaged}",
```

```
                    "aws:PrincipalArn":
 "arn:aws:iam::123456789012:role/org-admins/iam-admin"
                },
                "ForAnyValue:StringEquals": {
                    "aws:TagKeys": [
                        "GuardDutyManaged"
                    ]
                }
            }
        },
        {
            "Sid": "DenyModifyTagsIfPrinTagNotExists",
            "Effect": "Deny",
            "Action": [
                "ecs:TagResource",
                "ecs:UntagResource"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringNotEquals": {
                    "aws:PrincipalArn":
 "arn:aws:iam::123456789012:role/org-admins/iam-admin"
                },
                "Null": {
                    "aws:PrincipalTag/GuardDutyManaged": true
                }
            }
        }
    ]
}
```

4. GuardDuty でサービスの一部であるタスクをモニタリングする場合は、Runtime Monitoring を有効にした後に新しいサービスデプロイが必要です。Runtime Monitoring を有効にする前に特定の ECS サービスの最後のデプロイが開始された場合は、サービスを再起動するか、`forceNewDeployment` を使用してサービスを更新できます。

   サービスを更新するステップについては、以下のリソースを参照してください。

   - 「Amazon Elastic Container Service デベロッパーガイド」の「コンソールを使用した Amazon ECS サービスの更新」。
   - 「Amazon Elastic Container Service API リファレンス」の「UpdateService」。

- 「AWS CLI コマンドリファレンス」の「update-service」。

Amazon EKS リソースのセキュリティエージェントの自動管理

Runtime Monitoring では、GuardDuty 自動設定と手動によるセキュリティエージェントの有効化をサポートしています。このセクションでは、Amazon EKS クラスターの自動エージェント設定を有効にするステップについて説明します。

続行する前に、「Amazon EKS クラスターサポートの前提条件」に従っていることを確認してください。

「自動エージェント設定 (推奨)」の方法に関する任意のアプローチに応じて、以下のセクションのステップを適宜選択します。

マルチアカウント環境の自動エージェントの設定

マルチアカウント環境では、委任 GuardDuty 管理者アカウントのみが、メンバーアカウントの自動エージェント設定を有効または無効にすることや、組織内のメンバーアカウントに属する EKS クラスターの自動エージェントを管理することができます。GuardDuty メンバーアカウントからは、この設定を変更できません。委任 GuardDuty 管理者アカウントは、 を使用してメンバーアカウントを管理します AWS Organizations。マルチアカウント環境の詳細については、「複数のアカウントの管理」を参照してください。

委任 GuardDuty 管理者アカウントの自動エージェント設定

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
GuardDuty によるセキュリティエージェントの管理 (すべての EKS クラスターのモニタリング)	[Runtime Monitoring] セクションで [すべてのアカウントについて有効にする] を選択した場合、次のオプションがあります。 • [自動エージェント設定] セクションで [すべてのアカウントについて有効にする] を選択します。GuardDuty は、委任 GuardDuty 管理者アカウントに属するすべての EKS クラスターと、組織内のすべての既存のメンバーアカウントと潜在的な新しいメンバーアカウントに属するすべての EKS クラスターについて、セキュリティエージェントをデプロイおよび管理します。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	• [アカウントを手動で設定] を選択します。 [Runtime Monitoring] セクションで [アカウントを手動で設定] を選択した場合、次の操作を行います。 1. [自動エージェント設定] セクションで [アカウントを手動で設定する] を選択します。 2. [委任 GuardDuty 管理者 (このアカウント)] セクションで [有効にする] を選択します。 [保存] を選択します。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
一部を除外したすべての EKS クラスターのモニタリング (除外タグの使用)	次の手順から、該当するシナリオを 1 つ選択してください。 GuardDuty セキュリティエージェントが EKS クラスターにデプロイされていない場合に EKS クラスターをモニタリングから除外するには 1. キーを `GuardDutyManaged` 、値を `false` として、この EKS クラスターにタグを追加します。 Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *`ec2:CreateTags`* を `eks:TagResource` で置き換えます。 • *`ec2:DeleteTags`* を `eks:UntagResource` で置き換えます。 • *`access-project`* を `GuardDutyManaged` に置き換えます。 • *`123456789012`* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の `PrincipalArn` を追加します。 `"aws:PrincipalArn":["arn:aws:iam::12345678901 2:role/org-admins/iam-admin", "arn:aws:iam::1234`

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ

```
56789012:role/org-admins/iam-admin", "arn:aws:
iam::123456789012:role/org-admins/iam-admin"]
```

3. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
4. ナビゲーションペインで、[Runtime Monitoring] を選択します。

> **Note**
>
> アカウントの GuardDuty エージェントの自動管理を有効にする前に、必ず EKS クラスターに除外タグを追加してください。追加しないと、GuardDuty セキュリティエージェントがアカウント内のすべての EKS クラスターにデプロイされます。

5. [設定] タブの [GuardDuty エージェント管理] セクションで [有効にする] を選択します。

   モニタリングから除外されていない EKS クラスターについて、GuardDuty が GuardDuty セキュリティエージェントのデプロイと更新を管理します。
6. [保存] を選択します。

GuardDuty セキュリティエージェントが EKS クラスターにデプロイされている場合に EKS クラスターをモニタリングから除外するには

1. キーを `GuardDutyManaged` 、値を `false` として、この EKS クラスターにタグを追加します。

   Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。
2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリン

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ

シパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。

- *ec2:CreateTags* を eks:TagResource で置き換えます。
- *ec2:DeleteTags* を eks:UntagResource で置き換えます。
- *access-project* を GuardDutyManaged に置き換えます。
- *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。

  信頼できるエンティティが複数ある場合は、次の例を使用して複数の PrincipalArn を追加します。

```
"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]
```

3. この EKS クラスターで自動エージェントを有効にしている場合、このステップを完了すると、GuardDuty はこのクラスターのセキュリティエージェントを更新しなくなります。ただし、セキュリティエージェントはデプロイされたままになり、GuardDuty はこの EKS クラスターからランタイムイベントを受信し続けます。これにより、使用統計に影響を与える可能性があります。

   このクラスターからのランタイムイベントの受信を停止するには、デプロイされたセキュリティエージェントをこの EKS クラスターから削除する必要があります。デプロイされたセキュリティエージェントを削除する方法の詳細については、「Runtime

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	Monitoring でのリソースの無効化、アンインストール、クリーンアップ」を参照してください。 4. この EKS クラスターの GuardDuty セキュリティエージェントを手動で管理していた場合は、「Runtime Monitoring でのリソースの無効化、アンインストール、クリーンアップ」を参照してください。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
包含タグを使用した選択的な EKS クラスターのモニタリング	Runtime Monitoring をどのように有効にするかにかかわらず、以下のステップはアカウント内の選択的な EKS クラスターをモニタリングするために役立ちます。 1. [自動エージェント設定] セクションで、必ず [委任 GuardDuty 管理者アカウント (このアカウント)] の [無効にする] を選択してください。Runtime Monitoring の設定は、前のステップで設定したものと同じにします。 2. [保存] を選択します。 3. キーを `GuardDutyManaged` 、値を `true` として、EKS クラスターにタグを追加します。 Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。 GuardDuty は、モニタリングする選択的な EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 4. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *`ec2:CreateTags`* を `eks:TagResource` で置き換えます。 • *`ec2:DeleteTags`* を `eks:UntagResource` で置き換えます。 • *`access-project`* を `GuardDutyManaged` に置き換えます。 • *`123456789012`* を信頼されたエンティティの AWS アカウント ID に置き換えます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	信頼できるエンティティが複数ある場合は、次の例を使用して複数の `PrincipalArn` を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]`
GuardDuty セキュリティエージェントの手動管理	Runtime Monitoring をどのように有効にするかにかかわらず、EKS クラスターのセキュリティエージェントを手動で管理できます。 1. [自動エージェント設定] セクションで、必ず [委任 GuardDuty 管理者アカウント (このアカウント)] の [無効にする] を選択してください。Runtime Monitoring の設定は、前のステップで設定したものと同じにします。 2. [保存] を選択します。 3. セキュリティエージェントを管理するには、「Amazon EKS クラスターのセキュリティエージェントの手動管理」を参照してください。

すべてのメンバーアカウントの自動エージェントを自動で有効にする

> **Note**
>
> メンバーアカウントの設定を更新するには、最大 24 時間かかる場合があります。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
GuardDuty によるセキュリティエージェントの管理 (すべての EKS クラスターのモニタリング)	このトピックは、すべてのメンバーアカウントの Runtime Monitoring を有効にするためのものです。そのため、以下のステップでは、[Runtime Monitoring] セクションで [すべてのアカウントについて有効にする] を選択していることを前提としています。 1. [自動エージェント設定] セクションで [すべてのアカウントについて有効にする] を選択します。GuardDuty は、委任 GuardDuty 管理者アカウントに属するすべての EKS クラスターと、組織内のすべての既存のメンバーアカウントと潜在的な新しいメンバーアカウントに属するすべての EKS クラスターについて、セキュリティエージェントをデプロイおよび管理します。 2. [保存] を選択します。
一部を除外したすべての EKS クラスターのモニタリング (除外タグの使用)	次の手順から、該当するシナリオを 1 つ選択してください。 GuardDuty セキュリティエージェントが EKS クラスターにデプロイされていない場合に EKS クラスターをモニタリングから除外するには 1. キーを `GuardDutyManaged` 、値を `false` として、この EKS クラスターにタグを追加します。 Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *`ec2:CreateTags`* を `eks:TagResource` で置き換えます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ

- *ec2:DeleteTags* を eks:UntagResource で置き換えます。
- *access-project* を GuardDutyManaged に置き換えます。
- *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。

  信頼できるエンティティが複数ある場合は、次の例を使用して複数の PrincipalArn を追加します。

```
"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]
```

3. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
4. ナビゲーションペインで、[Runtime Monitoring] を選択します。

> **Note**
>
> アカウントの自動エージェントを有効にする前に、必ず EKS クラスターに除外タグを追加してください。追加しないと、GuardDuty セキュリティエージェントがアカウント内のすべての EKS クラスターにデプロイされます。

5. [設定] タブの [Runtime Monitoring 設定] セクションで [編集] を選択します。
6. [自動エージェント設定] セクションで [すべてのアカウントについて有効にする] を選択します。モニタリングから除外されていない EKS クラスターについて、GuardDuty が GuardDuty セキュリティエージェントのデプロイと更新を管理します。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	7. [保存] を選択します。 GuardDuty セキュリティエージェントが EKS クラスターにデプロイされている場合に EKS クラスターをモニタリングから除外するには 1. キーを `GuardDutyManaged` 、値を `false` として、この EKS クラスターにタグを追加します。 Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。 2. この EKS クラスターで自動エージェント設定を有効にしている場合、このステップを完了すると、GuardDuty はこのクラスターのセキュリティエージェントを更新しなくなります。ただし、セキュリティエージェントはデプロイされたままになり、GuardDuty はこの EKS クラスターからランタイムイベントを受信し続けます。これにより、使用統計に影響を与える可能性があります。 このクラスターからのランタイムイベントの受信を停止するには、デプロイされたセキュリティエージェントをこの EKS クラスターから削除する必要があります。デプロイされたセキュリティエージェントを削除する方法の詳細については、「Runtime Monitoring でのリソースの無効化、アンインストール、クリーンアップ」を参照してください。 3. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *`ec2:CreateTags`* を `eks:TagResource` で置き換えます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ

- *ec2:DeleteTags* を eks:UntagResource で置き換えます。
- *access-project* を GuardDutyManaged に置き換えます。
- *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。

  信頼できるエンティティが複数ある場合は、次の例を使用して複数の PrincipalArn を追加します。

```
"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]
```

4. この EKS クラスターの GuardDuty セキュリティエージェントを手動で管理していた場合は、「Runtime Monitoring でのリソースの無効化、アンインストール、クリーンアップ」を参照してください。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
包含タグを使用した選択的な EKS クラスターのモニタリング	Runtime Monitoring をどのように有効にするかにかかわらず、以下のステップは組織内のすべてのメンバーアカウントの選択的な EKS クラスターをモニタリングするために役立ちます。 1. [自動エージェント設定] セクションの設定はどれも有効にしないでください。Runtime Monitoring の設定は、前のステップで設定したものと同じにします。 2. [保存] を選択します。 3. キーを `GuardDutyManaged` 、値を `true` として、EKS クラスターにタグを追加します。 Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。 GuardDuty は、モニタリングする選択的な EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 4. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *`ec2:CreateTags`* を `eks:TagResource` で置き換えます。 • *`ec2:DeleteTags`* を `eks:UntagResource` で置き換えます。 • *`access-project`* を `GuardDutyManaged` に置き換えます。 • *`123456789012`* を信頼されたエンティティの AWS アカウント ID に置き換えます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	信頼できるエンティティが複数ある場合は、次の例を使用して複数の `PrincipalArn` を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]`
GuardDuty セキュリティエージェントの手動管理	Runtime Monitoring をどのように有効にするかにかかわらず、EKS クラスターのセキュリティエージェントを手動で管理できます。 1. [自動エージェント設定] セクションの設定はどれも有効にしないでください。Runtime Monitoring の設定は、前のステップで設定したものと同じにします。 2. [保存] を選択します。 3. セキュリティエージェントを管理するには、「Amazon EKS クラスターのセキュリティエージェントの手動管理」を参照してください。

すべての既存のアクティブなメンバーアカウントの自動エージェントを有効にする

> **Note**
>
> メンバーアカウントの設定を更新するには、最大 24 時間かかる場合があります。

組織内の既存のアクティブなメンバーアカウントの GuardDuty セキュリティエージェントを管理するには

- GuardDuty が組織内の既存のアクティブなメンバーアカウントに属する EKS クラスターからランタイムイベントを受信できるようにするには、これらの EKS クラスターの GuardDuty セキュリティエージェントを管理するための推奨アプローチを選択する必要があります。上記の各アプ

ローチの詳細については、「セキュリティエージェントの Amazon EKS へのデプロイ」を参照してください。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
GuardDuty によるセキュリティエージェントの管理 (すべての EKS クラスターのモニタリング)	既存のすべてのアクティブなメンバーアカウントのすべての EKS クラスターをモニタリングするには 1. [Runtime Monitoring] ページの [設定] タブで、自動エージェント 設定の現在のステータスを表示できます。 2. [自動エージェント設定] ペイン内の [アクティブなメンバーアカウント] セクションで、[アクション] を選択します。 3. [アクション] から、[すべての既存のアクティブなメンバーアカウントについて有効にする] を選択します。 4. [確認] を選択します。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
一部を除外したすべての EKS クラスターのモニタリング (除外タグの使用)	次の手順から、該当するシナリオを 1 つ選択してください。 GuardDuty セキュリティエージェントが EKS クラスターにデプロイされていない場合に EKS クラスターをモニタリングから除外するには 1. キーを `GuardDutyManaged` 、値を `false` として、この EKS クラスターにタグを追加します。 Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *`ec2:CreateTags`* を `eks:TagResource` で置き換えます。 • *`ec2:DeleteTags`* を `eks:UntagResource` で置き換えます。 • *`access-project`* を `GuardDutyManaged` に置き換えます。 • *`123456789012`* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の `PrincipalArn` を追加します。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ

```
"aws:PrincipalArn":["arn:aws:iam::12
3456789012:role/org-admins/iam-
admin", "arn:aws:iam::123456789012:
role/org-admins/iam-admin", "arn:aws:
iam::123456789012:role/org-admins/ia
m-admin"]
```

3. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
4. ナビゲーションペインで、[Runtime Monitoring] を選択します。

> **Note**
>
> アカウントの自動エージェント設定を有効にする前に、必ず EKS クラスターに除外タグを追加してください。追加しないと、GuardDuty セキュリティエージェントがアカウント内のすべての EKS クラスターにデプロイされます。

5. [自動エージェント設定] ペイン内の [設定] タブから、[アクティブなメンバーアカウント] で、[アクション] を選択します。
6. [アクション] から、[すべてのアクティブなメンバーアカウントについて有効にする] を選択します。
7. [確認] を選択します。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	GuardDuty セキュリティエージェントが既に EKS クラスターにデプロイされた後に EKS クラスターをモニタリングから除外するには 1. キーを `GuardDutyManaged` 、値を `false` として、この EKS クラスターにタグを追加します。 Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。 このステップが完了すると、GuardDuty はこのクラスターのセキュリティエージェントを更新しなくなります。ただし、セキュリティエージェントはデプロイされたままになり、GuardDuty はこの EKS クラスターからランタイムイベントを受信し続けます。これにより、使用統計に影響を与える可能性があります。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *ec2:CreateTags* を `eks:TagResource` で置き換えます。 • *ec2:DeleteTags* を `eks:UntagResource` で置き換えます。 • *access-project* を `GuardDutyManaged` に置き換えます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	• *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の PrincipalArn を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]` 3. セキュリティエージェントの管理方法 (GuardDuty の使用または手動) にかかわらず、このクラスターからのランタイムイベントの受信を停止するには、この EKS クラスターからデプロイされたセキュリティエージェントを削除する必要があります。デプロイされたセキュリティエージェントを削除する方法の詳細については、「Runtime Monitoring でのリソースの無効化、アンインストール、クリーンアップ」を参照してください。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
包含タグを使用した選択的な EKS クラスターのモニタリング	1. [アカウント] ページで、Runtime Monitoring を有効にした後は、[Runtime Monitoring 自動エージェント設定] を有効にしないでください。 2. モニタリングする選択したアカウントに属する EKS クラスターにタグを追加します。タグのキーと値のペアは `GuardDutyManaged -true` である必要があります。 Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。 GuardDuty は、モニタリングする選択的な EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 3. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *ec2:CreateTags* を `eks:TagResource` で置き換えます。 • *ec2:DeleteTags* を `eks:UntagResource` で置き換えます。 • *access-project* を `GuardDutyManaged` に置き換えます。 • *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	信頼できるエンティティが複数ある場合は、次の例を使用して複数の `PrincipalArn` を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]`
GuardDuty セキュリティエージェントの手動管理	1. [自動エージェント設定] セクションで [有効化] を選択しないようにしてください。Runtime Monitoring は有効のままにします。 2. [保存] を選択します。 3. セキュリティエージェントを管理するには、「Amazon EKS クラスターのセキュリティエージェントの手動管理」を参照してください。

新規メンバー用の自動エージェント設定を自動有効化

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
GuardDuty によるセキュリティエージェントの管理 (すべての EKS クラスターのモニタリング)	1. [Runtime Monitoring] ページで、[編集] を選択して既存の設定を更新します。 2. [自動エージェント設定] セクションで [すべてのアカウントについて有効にする] を選択します。 3. [保存] を選択します。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
一部を除外したすべての EKS クラスターのモニタリング (除外タグの使用)	次の手順から、該当するシナリオを 1 つ選択してください。 GuardDuty セキュリティエージェントが EKS クラスターにデプロイされていない場合に EKS クラスターをモニタリングから除外するには 1. キーを `GuardDutyManaged` 、値を `false` として、この EKS クラスターにタグを追加します。 Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *`ec2:CreateTags`* を `eks:TagResource` で置き換えます。 • *`ec2:DeleteTags`* を `eks:UntagResource` で置き換えます。 • *`access-project`* を `GuardDutyManaged` に置き換えます。 • *`123456789012`* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の `PrincipalArn` を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-`

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]
	3. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。 4. ナビゲーションペインで、[Runtime Monitoring] を選択します。 **Note** アカウントの自動エージェント設定を有効にする前に、必ず EKS クラスターに除外タグを追加してください。追加しないと、GuardDuty セキュリティエージェントがアカウント内のすべての EKS クラスターにデプロイされます。 5. [設定] タブの [GuardDuty エージェント管理] セクションで、[新しいメンバーアカウントについて自動的に有効にする] を選択します。 モニタリングから除外されていない EKS クラスターについて、GuardDuty が GuardDuty セキュリティエージェントのデプロイと更新を管理します。 6. [保存] を選択します。 GuardDuty セキュリティエージェントが EKS クラスターにデプロイされている場合に EKS クラスターをモニタリングから除外するには 1. GuardDuty セキュリティエージェントを GuardDuty を使用して管理するか、手動で管理するかにかかわらず、キーとして `GuardDutyManaged` 、値として `false` を使用したタグをこの EKS クラスターに追加します。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。 この EKS クラスターで自動エージェントを有効にしている場合、このステップを完了すると、GuardDuty はこのクラスターのセキュリティエージェントを更新しなくなります。ただし、セキュリティエージェントはデプロイされたままになり、GuardDuty はこの EKS クラスターからランタイムイベントを受信し続けます。これにより、使用統計に影響を与える可能性があります。 このクラスターからのランタイムイベントの受信を停止するには、デプロイされたセキュリティエージェントをこの EKS クラスターから削除する必要があります。デプロイされたセキュリティエージェントを削除する方法の詳細については、「Runtime Monitoring でのリソースの無効化、アンインストール、クリーンアップ」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *ec2:CreateTags* を eks:TagResource で置き換えます。 • *ec2:DeleteTags* を eks:UntagResource で置き換えます。 • *access-project* を GuardDutyManaged に置き換えます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	• *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の `PrincipalArn` を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]` 3. この EKS クラスターの GuardDuty セキュリティエージェントを手動で管理していた場合は、「Runtime Monitoring でのリソースの無効化、アンインストール、クリーンアップ」を参照してください。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
包含タグを使用した選択的な EKS クラスターのモニタリング	Runtime Monitoring をどのように有効にするかにかかわらず、以下のステップは組織内の新しいメンバーアカウントの選択的な EKS クラスターをモニタリングするために役立ちます。 1. [自動エージェント設定] セクションの [新しいメンバーアカウントについて自動的に有効にする] を必ずオフにしてください。Runtime Monitoring の設定は、前のステップで設定したものと同じにします。 2. [保存] を選択します。 3. キーを `GuardDutyManaged` 、値を `true` として、EKS クラスターにタグを追加します。 Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。 GuardDuty は、モニタリングする選択的な EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 4. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *`ec2:CreateTags`* を `eks:TagResource` で置き換えます。 • *`ec2:DeleteTags`* を `eks:UntagResource` で置き換えます。 • *`access-project`* を `GuardDutyManaged` に置き換えます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	• *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の `PrincipalArn` を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]`
GuardDuty セキュリティエージェントの手動管理	Runtime Monitoring をどのように有効にするかにかかわらず、EKS クラスターのセキュリティエージェントを手動で管理できます。 1. [GuardDuty エージェント管理] セクションの [新しいメンバーアカウントについて自動的に有効にする] のチェックボックスを必ずオフにしてください。Runtime Monitoring の設定は、前のステップで設定したものと同じにします。 2. [保存] を選択します。 3. セキュリティエージェントを管理するには、「Amazon EKS クラスターのセキュリティエージェントの手動管理」を参照してください。

アクティブなメンバーアカウントの自動エージェントを選択的に設定する

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
GuardDuty によるセキュリティエージェントの管理 (すべての EKS クラスターのモニタリング)	1. [アカウント] ページで、[自動エージェント設定] を有効にするアカウントを選択します。一度に複数のアカウントを選択することもできます。このステップで選択したアカウントで EKS Runtime Monitoring が既に有効になっていることを確認してください。 2. [保護プランの編集] から、適切なオプションを選択して [Runtime Monitoring 自動エージェント設定] を有効にします。 3. [確認] を選択します。
一部を除外したすべての EKS クラスターのモニタリング (除外タグの使用)	次の手順から、該当するシナリオを 1 つ選択してください。 GuardDuty セキュリティエージェントが EKS クラスターにデプロイされていない場合に EKS クラスターをモニタリングから除外するには 1. キーを `GuardDutyManaged` 、値を `false` として、この EKS クラスターにタグを追加します。 Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *ec2:CreateTags* を `eks:TagResource` で置き換えます。 • *ec2:DeleteTags* を `eks:UntagResource` で置き換えます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ

- *access-project* を GuardDutyManaged に置き換えます。
- *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。

  信頼できるエンティティが複数ある場合は、次の例を使用して複数の PrincipalArn を追加します。

```
"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]
```

3. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

> **Note**
>
> アカウントの自動エージェント設定を有効にする前に、必ず EKS クラスターに除外タグを追加してください。追加しないと、GuardDuty セキュリティエージェントがアカウント内のすべての EKS クラスターにデプロイされます。

4. [アカウント] ページで、[エージェントの自動管理] を有効にするアカウントを選択します。一度に複数のアカウントを選択することもできます。
5. [保護プランの編集] から、適切なオプションを選択して選択したアカウントの [Runtime Monitoring 自動エージェント設定] を有効にします。

   モニタリングから除外されていない EKS クラスターについて、GuardDuty が GuardDuty セキュリティエージェントのデプロイと更新を管理します。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	6. [保存] を選択します。 GuardDuty セキュリティエージェントが EKS クラスターにデプロイされている場合に EKS クラスターをモニタリングから除外するには 1. キーを `GuardDutyManaged` 、値を `false` として、この EKS クラスターにタグを追加します。 Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。 この EKS クラスターで GuardDuty エージェント管理を以前に有効にしている場合、このステップを完了すると、GuardDuty はこのクラスターのセキュリティエージェントを更新しなくなります。ただし、セキュリティエージェントはデプロイされたままになり、GuardDuty はこの EKS クラスターからランタイムイベントを受信し続けます。これにより、使用統計に影響を与える可能性があります。 このクラスターからのランタイムイベントの受信を停止するには、デプロイされたセキュリティエージェントをこの EKS クラスターから削除する必要があります。デプロイされたセキュリティエージェントを削除する方法の詳細については、「Runtime Monitoring でのリソースの無効化、アンインストール、クリーンアップ」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *`ec2:CreateTags`* を `eks:TagResource` で置き換えます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ

- *ec2:DeleteTags* を eks:UntagResource で置き換えます。
- *access-project* を GuardDutyManaged に置き換えます。
- *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。

  信頼できるエンティティが複数ある場合は、次の例を使用して複数の PrincipalArn を追加します。

```
"aws:PrincipalArn":["arn:aws:iam::12345678901
2:role/org-admins/iam-admin", "arn:aws:iam::1234
56789012:role/org-admins/iam-admin", "arn:aws:
iam::123456789012:role/org-admins/iam-admin"]
```

3. この EKS クラスターの GuardDuty セキュリティエージェントを手動で管理していた場合は、それを削除する必要があります。詳細については、「Runtime Monitoring でのリソースの無効化、アンインストール、クリーンアップ」を参照してください。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
包含タグを使用した選択的な EKS クラスターのモニタリング	Runtime Monitoring をどのように有効にするかにかかわらず、以下のステップは選択したアカウントに属する選択的な EKS クラスターをモニタリングするために役立ちます。 1. モニタリングする EKS クラスターを含む選択したアカウントについて、[Runtime Monitoring 自動エージェント設定] を有効にしていないことを確認してください。 2. キーを `GuardDutyManaged` 、値を `true` として、EKS クラスターにタグを追加します。 Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。 タグを追加すると、GuardDuty は、モニタリングする選択的な EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 3. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *`ec2:CreateTags`* を `eks:TagResource` で置き換えます。 • *`ec2:DeleteTags`* を `eks:UntagResource` で置き換えます。 • *`access-project`* を `GuardDutyManaged` に置き換えます。 • *`123456789012`* を信頼されたエンティティの AWS アカウント ID に置き換えます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	信頼できるエンティティが複数ある場合は、次の例を使用して複数の `PrincipalArn` を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]`
GuardDuty セキュリティエージェントの手動管理	1. Runtime Monitoring の設定は、前のステップで設定したものと同じにします。選択したどのアカウントでも、[Runtime Monitoring 自動エージェント設定] を有効にしていないことを確認してください。 2. [確認] を選択します。 3. セキュリティエージェントを管理するには、「Amazon EKS クラスターのセキュリティエージェントの手動管理」を参照してください。

スタンドアロンアカウント用の自動エージェントの設定

スタンドアロンアカウントは、特定の の で保護プランを有効または無効にする決定を所有 AWS アカウント します AWS リージョン。

アカウントが を通じて AWS Organizations、または招待方法によって GuardDuty 管理者アカウントに関連付けられている場合、このセクションはアカウントには適用されません。詳細については、「マルチアカウント環境の Runtime Monitoring の有効化」を参照してください。

Runtime Monitoring を有効にした後、自動設定または手動デプロイを使用して GuardDuty セキュリティエージェントをインストールしてください。次の手順に記載されたすべてのステップを完了する一環として、セキュリティエージェントをインストールしてください。

すべての Amazon EKS リソースをモニタリングするか、選択した Amazon EKS リソースをモニタリングするかに応じて、任意の方法を選択し、次の表のステップに従います。

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[Runtime Monitoring] を選択します。
3. [設定]タブで[有効にする]を選択し、アカウントの自動エージェント設定を有効にします。

GuardDuty セキュリティエージェントをデプロイするための推奨アプローチ	ステップ
GuardDuty によるセキュリティエージェントの管理 (すべての EKS クラスターのモニタリング)	1. [自動エージェント設定] セクションで [有効化] を選択します。GuardDuty は、アカウント内のすべての既存の EKS クラスターと潜在的な新しい EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 2. [保存] を選択します。
一部を除外したすべての EKS クラスターのモニタリング (除外タグの使用)	次の手順から、該当するシナリオを 1 つ選択してください。 GuardDuty セキュリティエージェントが EKS クラスターにデプロイされていない場合に EKS クラスターをモニタリングから除外するには 1. キーを `GuardDutyManaged` 、値を `false` として、この EKS クラスターにタグを追加します。 Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。

GuardDuty セキュリティエージェントをデプロイするための推奨アプローチ	ステップ
	• *ec2:CreateTags* を eks:TagResource で置き換えます。 • *ec2:DeleteTags* を eks:UntagResource で置き換えます。 • *access-project* を GuardDutyManaged に置き換えます。 • *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の PrincipalArn を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]` 3. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。 4. ナビゲーションペインで、[Runtime Monitoring] を選択します。 **Note** アカウントの GuardDuty エージェントの自動管理を有効にする前に、必ず EKS クラスターに除外タグを追加してください。追加しないと、GuardDuty セキュリティエージェントがアカウント内のすべての EKS クラスターにデプロイされます。

GuardDuty セキュリティエージェントをデプロイするための推奨アプローチ	ステップ
	5. [設定] タブの [GuardDuty エージェント管理] セクションで [有効にする] を選択します。 モニタリングから除外されていない EKS クラスターについて、GuardDuty が GuardDuty セキュリティエージェントのデプロイと更新を管理します。 6. [保存] を選択します。 GuardDuty セキュリティエージェントが既に EKS クラスターにデプロイされた後に EKS クラスターをモニタリングから除外するには 1. キーを `GuardDutyManaged` 、値を `false` として、この EKS クラスターにタグを追加します。 Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。 このステップが完了すると、GuardDuty はこのクラスターのセキュリティエージェントを更新しなくなります。ただし、セキュリティエージェントはデプロイされたままになり、GuardDuty はこの EKS クラスターからランタイムイベントを受信し続けます。これにより、使用統計に影響を与える可能性があります。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。

GuardDuty セキュリティエージェントをデプロイするための推奨アプローチ	ステップ
	• *ec2:CreateTags* を eks:TagResource で置き換えます。 • *ec2:DeleteTags* を eks:UntagResource で置き換えます。 • *access-project* を GuardDutyManaged に置き換えます。 • *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の PrincipalArn を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]` 3. このクラスターからのランタイムイベントの受信を停止するには、デプロイされたセキュリティエージェントをこの EKS クラスターから削除する必要があります。デプロイされたセキュリティエージェントを削除する方法の詳細については、「Runtime Monitoring でのリソースの無効化、アンインストール、クリーンアップ」を参照してください。

GuardDuty セキュリティエージェントをデプロイするための推奨アプローチ	ステップ
包含タグを使用した選択的な EKS クラスターのモニタリング	1. 必ず [自動エージェント設定] セクションで [無効化] を選択してください。Runtime Monitoring は有効のままにします。 2. [保存] を選択します。 3. キーを `GuardDutyManaged` 、値を `true` として、この EKS クラスターにタグを追加します。 Amazon EKS クラスターのタグ付けの詳細については、「Amazon EKS ユーザーガイド」の「コンソールでのタグの処理」を参照してください。 GuardDuty は、モニタリングする選択的な EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 4. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *`ec2:CreateTags`* を `eks:TagResource` で置き換えます。 • *`ec2:DeleteTags`* を `eks:UntagResource` で置き換えます。 • *`access-project`* を `GuardDutyManaged` に置き換えます。 • *`123456789012`* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の `PrincipalArn` を追加します。

GuardDuty セキュリティエージェントをデプロイするための推奨アプローチ	ステップ
	<pre>"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]</pre>
エージェントの手動管理	1. 必ず [自動エージェント設定] セクションで [無効化] を選択してください。Runtime Monitoring は有効のままにします。 2. [保存] を選択します。 3. セキュリティエージェントを管理するには、「Amazon EKS クラスターのセキュリティエージェントの手動管理」を参照してください。

Amazon EKS クラスターのセキュリティエージェントの手動管理

このセクションでは、Runtime Monitoring (または EKS Runtime Monitoring) を有効にした後に Amazon EKS アドオンエージェント (GuardDuty エージェント) を管理する方法について説明します。Runtime Monitoring を使用するには、Runtime Monitoring を有効にして Amazon EKS アドオン `aws-guardduty-agent` を設定する必要があります。GuardDuty が潜在的な脅威を検出し、「GuardDuty Runtime Monitoring の検出結果タイプ」を生成するには、両方のステップを実行する必要があります。

エージェントを手動で管理するには、前提条件として VPC エンドポイントを作成する必要があります。これは、GuardDuty がランタイムイベントを受信するのに役立ちます。その後、GuardDuty が Amazon EKS リソースからランタイムイベントを受信し始めるように、セキュリティエージェントをインストールできます。GuardDuty がこのリソースの新しいエージェントバージョンをリリースすると、アカウントでエージェントバージョンを更新できます。

トピック

- 前提条件 – Amazon VPC エンドポイントの作成

- Amazon EKS リソースへの GuardDuty セキュリティエージェントの手動インストール
- Amazon EKS リソースのセキュリティエージェントの手動更新

前提条件 – Amazon VPC エンドポイントの作成

GuardDuty セキュリティエージェントをインストールする前に、Amazon Virtual Private Cloud (Amazon VPC) エンドポイントを作成する必要があります。これにより、GuardDuty は Amazon EKS リソースのランタイムイベントを受信できるようになります。

> **Note**
>
> VPC エンドポイントの使用に追加コストはかかりません。

任意のアクセス方法を選択して、Amazon VPC エンドポイントを作成します。

Console

VPC エンドポイントを作成するには

1. Amazon VPC コンソールの https://console.aws.amazon.com/vpc/ を開いてください。
2. ナビゲーションペインの [仮想プライベートクラウド] で、[VPC] を選択します。
3. [エンドポイントの作成] を選択します。
4. [エンドポイントの作成] ページの [サービスカテゴリ] で [その他のエンドポイントサービス] を選択します。
5. [サービス名] に **`com.amazonaws.`*`us-east-1`*`.guardduty-data`** と入力します。

   必ず *`us-east-1`* を正しいリージョンに置き換えてください。これは、 AWS アカウント ID に属する EKS クラスターと同じリージョンである必要があります。
6. [サービスの確認] を選択します。
7. サービス名が正常に確認されたら、クラスターが置かれている [VPC] を選択します。次のポリシーを追加して、VPC エンドポイントの使用を指定されたアカウントのみに制限します。このポリシー下で提供されている組織 `Condition` を使用して、次のポリシーを更新してエンドポイントへのアクセスを制限できます。組織内の特定のアカウント ID に VPC エンドポイントサポートを提供するには、「Organization condition to restrict access to your endpoint」を参照してください。

JSON

```
{
 "Version":"2012-10-17",
 "Statement": [
  {
   "Action": "*",
   "Resource": "*",
   "Effect": "Allow",
   "Principal": "*"
  },
  {
   "Condition": {
    "StringNotEquals": {
     "aws:PrincipalAccount": "111122223333"
    }
   },
   "Action": "*",
   "Resource": "*",
   "Effect": "Deny",
   "Principal": "*"
  }
 ]
}
```

aws:PrincipalAccount アカウント ID は、VPC と VPC エンドポイントを含むアカウントと一致する必要があります。次のリストは、VPC エンドポイントを他の AWS アカウント ID と共有する方法を示しています。

エンドポイントへのアクセスを制限する組織の条件

- VPC エンドポイントにアクセスする複数のアカウントを指定するには、"aws:PrincipalAccount": "111122223333"を以下に置き換えます。

```
"aws:PrincipalAccount": [
          "666666666666",
          "555555555555"
    ]
```

- 組織のすべてのメンバーが VPC エンドポイントにアクセスできるようにするには、`"aws:PrincipalAccount": "111122223333"` を以下に置き換えます。

```
"aws:PrincipalOrgID": "o-abcdef0123"
```

- リソースへのアクセスを組織 ID に制限するには、`ResourceOrgID` をポリシーに追加します。

  詳細については、「ResourceOrgID」を参照してください。

```
"aws:ResourceOrgID": "o-abcdef0123"
```

8. [追加設定] で [DNS 名を有効にする] を選択します。
9. [サブネット] で、クラスターが存在するサブネットを選択します。
10. [セキュリティグループ] で、VPC (または EKS クラスター) からのインバウンドポート 443 が有効になっているセキュリティグループを選択します。インバウンドポート 443 が有効になっているセキュリティグループがまだない場合は、セキュリティグループを作成します。

    VPC (またはインスタンス) へのインバウンド許可を制限する際に問題が発生した場合は、任意の IP アドレス (`0.0.0.0/0`) からのインバウンド 443 ポートをサポートできます。ただし、GuardDuty では、VPC の CIDR ブロックに一致する IP アドレスを使用することをお勧めします。詳細については、「Amazon VPC ユーザーガイド」の「VPC CIDR ブロック」を参照してください。

API/CLI

VPC エンドポイントを作成するには

- CreateVpcEndpoint を呼び出します。
- パラメータで以下の値を使用します。
  - [サービス名] に **`com.amazonaws.`*`us-east-1`*`.guardduty-data`** と入力します。

    必ず *`us-east-1`* を正しいリージョンに置き換えてください。これは、 AWS アカウント ID に属する EKS クラスターと同じリージョンである必要があります。
  - DNSOptions には、プライベート DNS オプションを `true` に設定して有効にします。
- 詳細については AWS Command Line Interface、「create-vpc-endpoint」を参照してください。

ステップに従った後、「VPC エンドポイント設定の検証」を参照して、VPC エンドポイントが正しく設定されていることを確認します。

Amazon EKS リソースへの GuardDuty セキュリティエージェントの手動インストール

このセクションでは、特定の EKS クラスターに GuardDuty セキュリティエージェントを初めてデプロイする方法について説明します。このセクションに進む前に、アカウントの前提条件をセットアップし、Runtime Monitoring を有効にしていることを確認してください。Runtime Monitoring を有効にしないと、GuardDuty セキュリティエージェント (EKS アドオン) は動作しません。

任意のアクセス方法を選択して、GuardDuty セキュリティエージェントを初めてデプロイします。

Console

1. https://console.aws.amazon.com/eks/home#/clusters で Amazon EKS コンソールを開きます。
2. [クラスター名] を選択します。
3. [アドオン] タブを選択してください。
4. [その他のアドオンを入手] を選択します。
5. [アドオンの選択] ページで、[Amazon GuardDuty EKS Runtime Monitoring] を選択します。
6. GuardDuty では、最新およびデフォルトのエージェントバージョンを選択することをお勧めします。
7. [選択したアドオン設定の設定] ページで、デフォルトの設定を使用します。EKS アドオンの [ステータス] は、[アクティベーションが必要] の場合は、[GuardDuty を有効化] を選択します。このアクションにより、GuardDuty コンソールが開き、アカウントの Runtime Monitoring を設定できます。
8. アカウントに Runtime Monitoring を設定したら、Amazon EKS コンソールに戻ってください。EKS アドオンの [ステータス] は、[インストール準備完了] に変わっているはずです。
9. (オプション) EKS アドオン設定スキーマの提供

   アドオン [バージョン] で [v1.5.0] 以上を選択した場合、Runtime Monitoring は GuardDuty エージェントの特定のパラメータの設定をサポートします。パラメータ範囲の詳細については、「EKS アドオンパラメータを設定する」を参照してください。

   a. [オプションの構成設定] を展開して、設定可能なパラメータとその期待値と形式を表示します。

b. パラメータを設定します。値は、「EKS アドオンパラメータを設定する」に記載されたされた範囲内である必要があります。

c. 詳細設定に基づいてアドオンを作成するには、[変更の保存] を選択します。

d. [競合解決方法] で選択したオプションが、パラメータの値をデフォルト値以外の値に更新する際の競合を解決するために使用されます。記載されたオプションの詳細については、「Amazon EKS API リファレンス」の「resolveConflicts」を参照してください。

10. [次へ] を選択します。
11. [確認と作成] ページで、すべての詳細を確認し、[作成] を選択します。
12. クラスターの詳細に戻り、[リソース] タブを選択します。
13. [aws-guardduty-agent] というプレフィックスが付いた新しいポッドを表示できます。

API/CLI

Amazon EKS アドオンエージェント (aws-guardduty-agent) は、次のオプションのいずれかを使用して設定できます。

- アカウントの [CreateAddon] を実行します。
-

> **Note**
>
> アドオン version で [v1.5.0 以上] を選択した場合、Runtime Monitoring は GuardDuty エージェントの特定のパラメータの設定をサポートします。詳細については、「EKS アドオンパラメータを設定する」を参照してください。

リクエストパラメータに以下の値を使用します。

- addonName に「aws-guardduty-agent」と入力します。

アドオンバージョン v1.5.0以降でサポートされている設定可能な値を使用する場合は、次の AWS CLI 例を使用できます。赤で強調表示されたプレースホルダー値と、設定済みの値に関連付けられた Example.json を必ず置き換えてください。

```
aws eks create-addon --region us-east-1 --cluster-name myClusterName --addon-
name aws-guardduty-agent --addon-version v1.17.1-eksbuild.2 --configuration-
values 'file://example.json'
```

Example.json

```
{
 "priorityClassName": "aws-guardduty-agent.priorityclass-high",
 "dnsPolicy": "Default",
 "resources": {
  "requests": {
   "cpu": "237m",
   "memory": "512Mi"
  },
  "limits": {
   "cpu": "2000m",
   "memory": "2048Mi"
  }
 }
}
```

- サポートされている addonVersion については、「GuardDuty セキュリティエージェントでサポートされている Kubernetes のバージョン」を参照してください。

- または、 を使用することもできます AWS CLI。詳細については、「create-addon」を参照してください。

VPC エンドポイントのプライベート DNS 名

デフォルトでは、セキュリティエージェントは VPC エンドポイントのプライベート DNS 名を解決して接続します。FIPS 以外のエンドポイントの場合、プライベート DNS は以下の形式で表示されます。

FIPS 以外のエンドポイント – guardduty-data.*us-east-1*.amazonaws.com

us AWS リージョン*-east-1* の は、リージョンに応じて変わります。

Amazon EKS リソースのセキュリティエージェントの手動更新

GuardDuty セキュリティエージェントを手動で管理する場合、アカウントのセキュリティエージェントを更新する責任がユーザーにあります。新しいエージェントバージョンに関する通知については、「GuardDuty セキュリティエージェントのリリースバージョン」の RSS フィードにサブスクライブできます。

セキュリティエージェントを最新バージョンに更新して、追加されたサポートと改善を活用できます。現在のエージェントバージョンが標準サポートを終了していて、引き続き Runtime Monitoring (または EKS Runtime Monitoring) を使用する場合は、利用可能な次のバージョンまたは現在のエージェントバージョンに更新する必要があります。

前提条件

セキュリティエージェントのバージョンを更新する前に、現在使用する予定のエージェントのバージョンが Kubernetes バージョンと互換性があることを確認してください。詳細については、「GuardDuty セキュリティエージェントでサポートされている Kubernetes のバージョン」を参照してください。

Console

1. https://console.aws.amazon.com/eks/home#/clusters で Amazon EKS コンソールを開きます。
2. [クラスター名] を選択します。
3. [クラスター情報] で、[アドオン] タブを選択します。
4. [アドオン] タブで、[GuardDuty Runtime Monitoring] を選択します。
5. [編集] を選択してエージェントの詳細を更新します。
6. [GuardDuty EKS Runtime Monitoring の設定] ページで、詳細を更新します。
7. (オプション) オプションの設定の更新

   EKS アドオン [バージョン] が 1.5.0 以降の場合は、アドオン設定スキーマを更新することもできます。

   a. [オプションの構成設定] を展開して、設定スキーマを表示します。
   b. 「EKS アドオンパラメータを設定する」に記載された範囲に基づいてパラメータ値を更新します。
   c. [変更を保存] を選択して更新を開始します。
   d. [競合解決方法] で選択したオプションが、パラメータの値をデフォルト値以外の値に更新する際の競合を解決するために使用されます。記載されたオプションの詳細については、「Amazon EKS API リファレンス」の「resolveConflicts」を参照してください。

API/CLI

Amazon EKS クラスターの GuardDuty セキュリティエージェントを更新するには、「アドオンの更新」を参照してください。

> **Note**
>
> アドオン `version` で [1.5.0] 以降を選択した場合、Runtime Monitoring は GuardDuty エージェントの特定のパラメータの設定をサポートします。パラメータ範囲の詳細については、「EKS アドオンパラメータを設定する」を参照してください。

アドオンバージョン 1.5.0 以降でサポートされている設定可能な値を使用する場合は、次の AWS CLI 例を使用できます。赤で強調表示されたプレースホルダー値と、設定済みの値に関連付けられた `Example.json` を必ず置き換えてください。

```
aws eks update-addon --region us-east-1 --cluster-name myClusterName --addon-
name aws-guardduty-agent --addon-version v1.17.1-eksbuild.2 --configuration-
values 'file://example.json'
```

Example.json

```
{
 "priorityClassName": "aws-guardduty-agent.priorityclass-high",
 "dnsPolicy": "Default",
 "resources": {
  "requests": {
   "cpu": "237m",
   "memory": "512Mi"
  },
  "limits": {
   "cpu": "2000m",
   "memory": "2048Mi"
  }
 }
}
```

Amazon EKS アドオンバージョンが 1.5.0 以降で、アドオンスキーマを設定している場合は、クラスターに値が正しく表示されるかどうかを確認できます。詳細については、「設定スキーマの更新の検証」を参照してください。

Amazon EKS の GuardDuty セキュリティエージェント (アドオン) パラメータを設定する

Amazon EKS の GuardDuty セキュリティエージェントの特定のパラメータを設定できます。このサポートは、GuardDuty セキュリティエージェントバージョン 1.5.0 以降で利用できます。最新のアドオンバージョンについては、「Amazon EKS リソースの GuardDuty セキュリティエージェントバージョン」を参照してください。

セキュリティエージェント設定スキーマを更新する理由

> GuardDuty セキュリティエージェントの設定スキーマは、Amazon EKS クラスター内のすべてのコンテナで同じです。デフォルト値が関連するワークロードおよびインスタンスサイズと一致しない場合は、CPU 設定、メモリ設定、`PriorityClass`、および `dnsPolicy` 設定の設定を検討してください。Amazon EKS クラスターの GuardDuty エージェントを管理する方法に関係なく、これらのパラメータの既存の設定を設定または更新できます。

設定されたパラメータを使用した自動エージェント設定動作

GuardDuty がユーザーに代わってセキュリティエージェント (EKS アドオン) を管理する場合、必要に応じてアドオンを更新します。GuardDuty は、設定可能なパラメータの値をデフォルト値に設定します。ただし、パラメータを目的の値に更新することはできます。これにより競合が発生する場合、resolveConflicts のデフォルトオプションは `None` です。

設定可能なパラメータと値

アドオンパラメータを設定するステップについては、以下を参照してください。

- Amazon EKS リソースへの GuardDuty セキュリティエージェントの手動インストール または
- Amazon EKS リソースのセキュリティエージェントの手動更新

次の表は、Amazon EKS アドオンを手動でデプロイしたり、既存のアドオン設定を更新したりするために使用できる範囲と値を示しています。

CPU 設定

パラメータ	デフォルトの値	設定可能な範囲
リクエスト	200m	200m から 10000m の間、両方を含む
制限	1000m	200m から 10000m の間、両方を含む
`disableCpuLimits`	`false`	`true`、または `false`

`disableCpuLimits` パラメータは、GuardDuty セキュリティエージェントバージョン 1.12.1-eksbuild.3 以降で使用できます。以前のバージョンでは、アドオンはこのパラメータをサポートしておらず、Amazon EKS アドオン APIs (`CreateAddon`、`UpdateAddon`) を指定すると検証エラーが返されます。

`disableCpuLimits` を に設定すると`true`、セキュリティエージェントポッドは CPU 制限を適用しません。その他のリソース設定は影響を受けません。

CPU 制限を無効にするには、次の設定を使用します。

```
{"resources":{"disableCpuLimits":true}}
```

[メモリの設定]

パラメータ	デフォルトの値	設定可能な範囲
リクエスト	256Mi	256Mi から 20000Mi の間、両方を含む
制限	1024Mi	

`PriorityClass` 設定

GuardDuty が Amazon EKS アドオンを作成する場合、割り当てられる `PriorityClass` は `aws-guardduty-agent.priorityclass` です。つまり、エージェントポッドの優先

度に基づいてアクションは実行されません。このアドオンパラメータは、次のいずれかの PriorityClass オプションを選択して設定できます。

設定可能な **PriorityClass**	**preemptionPolicy** 値	**preemptionPolicy** の説明	ポッド値
aws-guardduty-agent.priorityclass	Never	アクションなし	1000000
aws-guardduty-agent.priorityclass-high	PreemptLowerPriority	この値を割り当てると、優先度値がエージェントポッド値よりも低いポッドの実行が中断されます。	100000000
system-cluster-critical [1]	PreemptLowerPriority		2000000000
system-node-critical [1]	PreemptLowerPriority		2000001000

[1] Kubernetes には、system-cluster-critical と system-node-critical の 2 つの PriorityClass オプションがあります。詳細については、「Kubernetes ドキュメント」の「PriorityClass」を参照してください。

dnsPolicy 設定

Kubernetes がサポートする次の DNS ポリシーオプションのいずれかを選択します。設定が指定されていない場合は、ClusterFirst がデフォルト値として使用されます。

- ClusterFirst
- ClusterFirstWithHostNet
- Default

これらのポリシーの詳細については、「Kubernetes ドキュメント」の「Pod の DNS ポリシー」を参照してください。

設定スキーマの更新の検証

パラメータを設定した後、次のステップを実行して設定スキーマが更新されていることを確認します。

1. https://console.aws.amazon.com/eks/home#/clusters で Amazon EKS コンソールを開きます。
2. ナビゲーションペインで [Clusters] (クラスター) を選択します。
3. [クラスター] ページで、更新を検証する [クラスター名] を選択します。
4. [リソース] タブを選択してください。
5. [リソースタイプ] ペインの [ワークロード] で、[DaemonSets] を選択します。
6. [aws-guardduty-agent] を選択します。
7. [aws-guardduty-agent] ページで [Raw ビュー] を選択して、フォーマットされていない JSON レスポンスを表示します。設定可能なパラメータに、指定した値が表示されていることを確認します。

確認した後、GuardDuty コンソールに切り替えます。対応する AWS リージョン を選択し、Amazon EKS クラスターのカバレッジステータスを表示します。詳細については、「Amazon EKS クラスターのランタイムカバレッジとトラブルシューティング」を参照してください。

VPC エンドポイント設定の検証

セキュリティエージェントを手動でインストールするか、GuardDuty 自動設定を使用してインストールした後、このドキュメントを使用して VPC エンドポイント設定を検証できます。リソースタイプのランタイムカバレッジの問題をトラブルシューティングした後、これらのステップを使用することもできます。ステップが期待どおりに機能し、カバレッジステータスが [正常] として表示される可能性があることを確認できます。

次のステップを使用して、リソースタイプの VPC エンドポイント設定が VPC 所有者アカウントで正しく設定されていることを確認します。

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/vpc/ で Amazon VPC コンソールを開きます。
2. ナビゲーションペインの [仮想プライベートクラウド] で、[VPC] を選択します。
3. [VPC] ページで、VPC ID に関連付けられた IPv4 CIDR を選択します。
4. ナビゲーションペインの [仮想プライベートクラウド] で、[VPC] を選択します。

5. [エンドポイント] テーブルで、[com.amazonaws.***us-east-1***.guardduty-data] のような [サービス名] を持つ行を選択します。リージョン (us-east-1) はエンドポイントによって異なる場合があります。
6. エンドポイントの詳細のパネルが表示されます。[セキュリティグループ] タブで、関連する [グループ ID] リンクを選択して詳細を確認します。
7. セキュリティグループテーブルで、関連するセキュリティグループ ID を持つ行を選択して詳細を表示します。
8. [インバウンドルール] タブで、[ポート範囲] が [443]、[ソース] が [IPv4 CIDR] からコピーされた値であるイングレスポリシーがあることを確認します。インバウンドルールは、インスタンスに到達できる受信トラフィックを制御します。次の図は、GuardDuty セキュリティエージェントが使用する VPC に関連付けられているセキュリティグループのインバウンドルールを示しています。

   インバウンドポート 443 が有効になっているセキュリティグループがまだない場合は、Amazon EC2 ユーザーガイド」の「セキュリティグループを作成する」を参照してください。

   VPC (またはクラスター) へのインバウンドアクセス許可の制限中に問題が発生した場合は、任意の IP アドレス (0.0.0.0/0) からのインバウンドトラフィックをポート 443 で許可します。

次のリストには、セキュリティエージェントをインストールまたは更新した後に知っておくべき項目が含まれています。

「ランタイムカバレッジの評価」

セキュリティエージェントをインストールまたは更新した後の次のステップは、リソースのランタイムカバレッジを評価することです。ランタイムカバレッジステータスが [異常] の場合、問題をトラブルシューティングする必要があります。詳細については、「ランタイムカバレッジの問題とトラブルシューティング」を参照してください。

ランタイムカバレッジのステータスが [正常] と表示されている場合は、Runtime Monitoring がランタイムイベントを収集して受信できることを示します。これらのイベントのリストについては、「収集されたランタイムイベントタイプ」を参照してください。

エンドポイントのプライベート DNS 名

リソースに GuardDuty セキュリティエージェントをインストールすると、デフォルトで VPC エンドポイントのプライベート DNS 名を解決して接続します。FIPS 以外のエンドポイントの場合、プライベート DNS は以下の形式で表示されます。

guardduty-data.*us-east-1*.amazonaws.com

us AWS リージョン*-east-1* の は、リージョンに応じて変わります。

ホストには 2 つのセキュリティエージェントがインストールされる場合があります

Amazon EC2 インスタンスの GuardDuty セキュリティエージェントを使用する場合、Amazon EKS クラスター内の基盤となるホストにエージェントをインストールして使用することができます。その EKS クラスターにセキュリティエージェントを既にデプロイしている場合、同じホストで 2 つのセキュリティエージェントが同時に実行される可能性があります。このシナリオでの GuardDuty の仕組みについては、「同じホスト上のセキュリティエージェント」を参照してください。

ランタイムカバレッジ統計の確認と問題のトラブルシューティング

Runtime Monitoring を有効にし、GuardDuty セキュリティエージェントがリソースにデプロイされると、GuardDuty は対応するリソースタイプのカバレッジ統計と、アカウントに属するリソースの個別のカバレッジステータスを提供します。カバレッジステータスは、Runtime Monitoring が有効になっていること、Amazon VPC エンドポイントが作成されていること、対応するリソースの GuardDuty セキュリティエージェントがデプロイされていることを確認することによって決定されます。[正常] カバレッジステータスは、リソースに関連するランタイムイベントが発生すると、GuardDuty が Amazon VPC エンドポイントを通じてそのランタイムイベントを受信し、動作を監視できることを示します。Runtime Monitoring の設定、Amazon VPC エンドポイントの作成または GuardDuty セキュリティエージェントのデプロイ時に問題が発生した場合、カバレッジステータスは [異常] と表示されます。カバレッジステータスが異常の場合、GuardDuty は対応するリソースのランタイム動作を受信または監視できず、Runtime Monitoring の結果を生成することもできません。

以下のトピックは、カバレッジ統計の確認、EventBridge 通知の設定、特定のリソースタイプのカバレッジ問題のトラブルシューティングに役立ちます。

内容

- Amazon EC2 インスタンスのランタイムカバレッジとトラブルシューティング
- ECS-EC2 Bottlerocket のランタイムカバレッジとトラブルシューティング
- Amazon ECS クラスターのランタイムカバレッジとトラブルシューティング
- Amazon EKS クラスターのランタイムカバレッジとトラブルシューティング

Amazon EC2 インスタンスのランタイムカバレッジとトラブルシューティング

Amazon EC2 リソースの場合、ランタイムカバレッジはインスタンスレベルで評価されます。Amazon EC2 インスタンスは、 AWS 環境内のさまざまなタイプのアプリケーションやワークロードを実行できます。この機能は Amazon ECS によって管理されている Amazon EC2 インスタンスもサポートしており、Amazon EC2 インスタンスで Amazon ECS クラスターを実行している場合、インスタンスレベルでのカバレッジの問題は Amazon EC2 ランタイムカバレッジに表示されます。

トピック

- カバレッジ統計の確認
- カバレッジステータス変更時の EventBridge 通知
- Amazon EC2 ランタイムカバレッジの問題のトラブルシューティング

カバレッジ統計の確認

自分のアカウントまたはメンバーアカウントに関連付けられた Amazon EC2 インスタンスのカバレッジ統計は、選択した AWS リージョンのすべての EC2 インスタンスに対する正常な EC2 インスタンスの割合です。次の式はこれを次のように表します。

(正常なインスタンス/すべてのインスタンス)*100

Amazon ECS クラスターに GuardDuty セキュリティエージェントもデプロイしている場合、Amazon EC2 インスタンスで実行されている Amazon ECS クラスターに関連するインスタンスレベルのカバレッジの問題は、Amazon EC2 インスタンスのランタイムカバレッジ問題として表示されます。

いずれかのアクセス方法を選択して、アカウントのカバレッジ統計を確認してください。

Console

- にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
- ナビゲーションペインで、[Runtime Monitoring] を選択します。
- [ランタイムカバレッジ] タブを選択します。

- [EC2 インスタンスのランタイムカバレッジ] タブでは、[クラスターリスト] テーブルにある各 Amazon EC2 インスタンスのカバレッジステータス別に集計されたカバレッジ統計を表示できます。
  - [インスタンスリスト] テーブルは次の列でフィルタリングできます。
    - アカウント ID
    - [エージェント管理タイプ]
    - [エージェントバージョン]
    - [カバレッジステータス]
    - [インスタンス ID]
    - クラスター ARN
- EC2 インスタンスのいずれかの [カバレッジステータス] が [異常] の場合、[問題] 列には、[異常] ステータスである理由に関する追加情報が含まれています。

API/CLI

- 独自の有効なディテクター ID、現在のリージョン、サービスエンドポイントを使用して ListCoverage API を実行します。この API を使用して、インスタンスリストをフィルタリングしたり、ソートしたりできます。
  - 以下の `CriterionKey` のオプションのいずれかを使用して例 `filter-criteria` を変更できます。
    - `ACCOUNT_ID`
    - `RESOURCE_TYPE`
    - `COVERAGE_STATUS`
    - `AGENT_VERSION`
    - `MANAGEMENT_TYPE`
    - `INSTANCE_ID`
    - `CLUSTER_ARN`
  - `filter-criteria` に [EC2] として `RESOURCE_TYPE` が含まれている場合、Runtime Monitoring は `AttributeName` としての [ISSUE] の使用をサポートしていません。これを使用すると、API レスポンスは `InvalidInputException` になります。

    以下のオプションで `sort-criteria` の例 `AttributeName` を変更できます。
    - `ACCOUNT_ID`

    - COVERAGE_STATUS
    - INSTANCE_ID
    - UPDATED_AT
  - *max-results* (最大 50) を変更できます。
  - アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty --region us-east-1 list-coverage --detector-
id 12abc34d567e8fa901bc2d34e56789f0 --sort-criteria '{"AttributeName":
 "EKS_CLUSTER_NAME", "OrderBy": "DESC"}' --filter-criteria
 '{"FilterCriterion":[{"CriterionKey":"ACCOUNT_ID", "FilterCondition":
{"EqualsValue":"111122223333"}}] }'  --max-results 5
```

- GetCoverageStatistics API を実行すると、statisticsType に基づいてカバレッジ集計された統計情報を取得できます。
  - 例 statisticsType を次のオプションのいずれかに変更できます。
    - COUNT_BY_COVERAGE_STATUS - カバレッジステータス別に集計された EKS クラスターのカバレッジ統計を表します。
    - COUNT_BY_RESOURCE_TYPE – リスト内の AWS リソースのタイプに基づいて集計されたカバレッジ統計。
    - コマンドで例 filter-criteria を変更できます。CriterionKey に対して次のオプションを設定できます。
      - ACCOUNT_ID
      - RESOURCE_TYPE
      - COVERAGE_STATUS
      - AGENT_VERSION
      - MANAGEMENT_TYPE
      - INSTANCE_ID
      - CLUSTER_ARN
  - アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty --region us-east-1 get-coverage-statistics --detector-
id 12abc34d567e8fa901bc2d34e56789f0 --statistics-type COUNT_BY_COVERAGE_STATUS
 --filter-criteria '{"FilterCriterion":[{"CriterionKey":"ACCOUNT_ID",
 "FilterCondition":{"EqualsValue":"123456789012"}}] }'
```

EC2 インスタンスのカバレッジステータスが [異常] である場合は、「Amazon EC2 ランタイムカバレッジの問題のトラブルシューティング」を参照してください。

カバレッジステータス変更時の EventBridge 通知

Amazon EC2 インスタンスのカバレッジステータスが [異常] と表示される場合があります。カバレッジステータスの変化を検出するためにカバレッジステータスを定期的に監視し、ステータスが [異常] の場合はトラブルシューティングすることをお勧めします。または、Amazon EventBridge ルールを作成して、カバレッジステータスが [異常] から [正常] に変更された場合やそれ以外の変更があった場合に通知を受け取ることもできます。デフォルトでは、GuardDuty はこれをアカウントの EventBridge バスに公開します。

通知スキーマの例

EventBridge のルールでは、あらかじめ定義されたサンプルイベントとイベントパターンを使用して、カバレッジステータスの通知を受け取ることができます。EventBridge ルールの作成の詳細については、「Amazon EventBridge ユーザーガイド」の「ルールの作成」を参照してください。

さらに、次の通知スキーマの例を使用して、カスタムイベントパターンを作成します。アカウントの値を必ず置き換えてください。Amazon EC2 インスタンスのカバレッジステータスが `Healthy` から `Unhealthy` に変更されたときに通知を受け取るには、`detail-type` が *`GuardDuty Runtime Protection Unhealthy`* である必要があります。カバレッジステータスが `Unhealthy` から `Healthy` に変更されたときに通知を受け取るには、`detail-type` の値を *`GuardDuty Runtime Protection Healthy`* に置き換えます。

```
{
  "version": "0",
  "id": "event ID",
  "detail-type": "GuardDuty Runtime Protection Unhealthy",
  "source": "aws.guardduty",
  "account": "AWS ##### ID",
  "time": "event timestamp (string)",
  "region": "AWS #####",
```

```
  "resources": [
      ],
  "detail": {
    "schemaVersion": "1.0",
    "resourceAccountId": "string",
    "currentStatus": "string",
    "previousStatus": "string",
    "resourceDetails": {
        "resourceType": "EC2",
        "ec2InstanceDetails": {
          "instanceId":"",
          "instanceType":"",
          "clusterArn": "",
          "agentDetails": {
            "version":""
          },
          "managementType":""
        }
    },
    "issue": "string",
    "lastUpdatedAt": "timestamp"
  }
}
```

Amazon EC2 ランタイムカバレッジの問題のトラブルシューティング

Amazon EC2 インスタンスのカバレッジステータスが [異常] の場合、その理由を [問題] 列で確認できます。

EC2 インスタンスが EKS クラスターに関連付けられていて、EKS のセキュリティエージェントが手動または自動エージェント設定でインストールされている場合は、「Amazon EKS クラスターのランタイムカバレッジとトラブルシューティング」を参照してカバレッジの問題をトラブルシューティングします。

次の表に、問題のタイプと対応するトラブルシューティングステップを示します。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
エージェントからの報告なし	SSM 通知を待機中	SSM 通知の受信には数分かかる場合があります。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
		Amazon EC2 インスタンスが SSM マネージドであることを確認してください。詳細については、「メソッド 1 - で AWS Systems Manager を使用する」の手順を参照してくださいセキュリティエージェントの手動インストール。
	(意図的に空にしてあります)	GuardDuty セキュリティエージェントを手動で管理している場合は、「Amazon EC2 リソースのセキュリティエージェントの手動管理」のステップに従っていることを確認してください。
		自動エージェント設定を有効にしている場合: • EC2 インスタンスは SSM マネージドです。 • セキュリティエージェントのステータスを定期的に確認します。詳細については、「GuardDuty セキュリティエージェントのインストールステータスの検証」を参照してください。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
		Amazon EC2 インスタンスの VPC エンドポイントが正しく設定されていることを確認します。詳細については、「VPC エンドポイント設定の検証」を参照してください。
		組織にサービスコントロールポリシー (SCP) がある場合は、アクセス許可の境界が `guardduty:SendSecurityTelemetry` アクセス許可を制限していないことを確認します。詳細については、「マルチアカウント環境での組織サービスコントロールポリシーの検証」を参照してください。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
	エージェントが切断されました	• セキュリティエージェントのステータスを確認します。詳細については、「GuardDuty セキュリティエージェントのインストールステータスの検証」を参照してください。 • セキュリティエージェントのログを表示して、潜在的な根本原因を特定します。ログには、問題を自分でトラブルシューティングするために使用できる詳細なエラーが表示されます。ログファイルは `/var/log/amzn-guardduty-agent/` で入手できます。 `sudo journalctl -u amazon-guardduty-agent` を行います。
エージェントがプロビジョニングされていない	除外タグを持つインスタンスは Runtime Monitoring から除外されます。	GuardDuty は、除外タグ `GuardDutyManaged :false` を使用して起動された Amazon EC2 インスタンスからランタイムイベントを受信しません。 この Amazon EC2 インスタンスからランタイムイベントを受信するには、除外タグを削除します。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
	カーネルバージョンがサポートされているバージョンより低いです。	OS ディストリビューション全体でサポートされているカーネルバージョンについては、Amazon EC2 インスタンスの「アーキテクチャ要件を検証する」を参照してください。
	カーネルバージョンがサポートされているバージョンを超えています。	OS ディストリビューション全体でサポートされているカーネルバージョンについては、Amazon EC2 インスタンスの「アーキテクチャ要件を検証する」を参照してください。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
	インスタンスアイデンティティドキュメントを取得できません。	以下の手順に従ってください。 1. リソースが Amazon EC2 インスタンスであり、ハイブリッド non-EC2 インスタンスではないことを確認します。 2. インスタンスメタデータサービス (IMDS) が有効になっていることを確認します。これを行うには、「Amazon EC2 ユーザーガイド」の「インスタンスメタデータサービスのオプションを設定する」を参照してください。 3. インスタンスアイデンティティドキュメントが存在することを確認します。これを行うには、「Amazon EC2 ユーザーガイド」の「インスタンス ID ドキュメントの取得」を参照してください。 4. インスタンス ID ドキュメントがまだ存在しない場合は、インスタンスを再起動します。インスタンスアイデンティティドキュメントは、インスタンスが停止して起動、再起動、起動するときに生成されます。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
SSM 関連付けの作成に失敗	GuardDuty SSM 関連付けがアカウント内に既に存在します	1. 既存の関連付けを手動で削除します。詳細については、「AWS Systems Manager ユーザーガイド」の「関連付けを削除する」を参照してください。 2. 関連付けを削除した後、Amazon EC2 の GuardDuty 自動エージェント設定を無効にしてから再度有効にします。
	アカウントに SSM 関連付けが多すぎます	次の 2 つのオプションのいずれかを選択します。 • 未使用の SSM 関連付けを削除します。詳細については、「AWS Systems Manager ユーザーガイド」の「関連付けを削除する」を参照してください。 • アカウントがクォータ引き上げの対象かどうかを確認します。詳細については、「AWS 全般のリファレンス」の「System Manager Service Quotas」を参照してください。
SSM 関連付けの更新に失敗	GuardDuty SSM 関連付けがアカウントに存在しません	GuardDuty SSM 関連付けがアカウントに存在しません。Runtime Monitoring を無効にしてから再度有効にします。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
SSM 関連付けの削除に失敗	GuardDuty SSM 関連付けがアカウントに存在しません	SSM 関連付けがアカウントに存在しません。SSM 関連付けが意図的に削除された場合、アクションは必要ありません。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
SSM インスタンス関連付けの実行に失敗	アーキテクチャ要件またはその他の前提条件が満たされていません。	検証済みのオペレーティングシステムディストリビューションの詳細については、「Amazon EC2 インスタンスサポートの前提条件」を参照してください。 それでもこの問題が解決しない場合は、次のステップが問題の特定と解決に役立ちます。 1. https://console.aws.amazon.com/systems-manager/ で AWS Systems Manager コンソールを開きます。 2. ナビゲーションペインの [ノード管理] で、[ステートマネージャー] を選択します。 3. [ドキュメント名] プロパティでフィルタリングし、「AmazonGuardDuty-ConfigureRuntimeMonitoringSsmPlugin」と入力します。 4. 対応する関連付け ID を選択し、その [実行履歴] を表示します。 5. 実行履歴を使用して、障害を表示し、潜在的な根

問題タイプ	問題メッセージ	トラブルシューティングのステップ
		本原因を特定し、解決を試みます。
VPC エンドポイントの作成に失敗	VPC エンドポイントの作成は、共有 VPC *vpcId* ではサポートされていません	Runtime Monitoring は、組織内の共有 VPC の使用をサポートしています。詳細については、「Runtime Monitoring で共有 VPC を使用する」を参照してください。
	自動エージェント設定で共有 VPC を使用する場合のみ 共有 VPC *vpcId* の所有者アカウント ID *111122223333* では、Runtime Monitoring、自動エージェント設定、またはその両方が有効になっていません	共有 VPC 所有者アカウントは、少なくとも 1 つのリソースタイプ (Amazon EKS または Amazon ECS (AWS Fargate)) の Runtime Monitoring および自動エージェント設定を有効にする必要があります。詳細については、「GuardDuty Runtime Monitoring に固有の前提条件」を参照してください。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
	プライベート DNS を有効にするには、enableDnsSupport とenableDnsHostnames VPC 属性の両方が *vpcId* に対して true に設定されている必要があります (サービス: Ec2、ステータス コード: 400、リクエスト ID: *a1b2c3d4-5678-90ab-cdef-EXAMPLE11111*)。	次の VPC 属性が true - enableDnsSupport および enableDnsHostnames に設定されていることを確認する必要があります。詳細については、「VPC の DNS 属性」を参照してください。 https://console.aws.amazon.com/vpc/ にある Amazon VPC コンソールを使用して Amazon VPC を作成する場合は、[DNS ホスト名を有効にする] および [DNS 解決を有効にする] の両方を選択してください。詳細については、「VPC 設定オプション」を参照してください。 VPC 属性を更新した後、以下のいずれかの変更を行うことで、VPC エンドポイントの作成を再試行する必要があります。 • Amazon EC2 インスタンスの GuardDuty Managed タグを追加または変更できます (推奨)。例えば、GuardDuty Managed :true を追加して、Runtime Monitoring のインスタンスを含めることができます。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
		• Amazon EC2 インスタンスの状態は変更できます (停止または再起動)。
共有 VPC エンドポイントの削除に失敗	共有 VPC エンドポイントの削除は、アカウント ID *111122223333* 、共有 VPC *vpcId*、所有者アカウント ID *555555555555* では許可されません。	考えられるステップ: • 共有 VPC 参加者アカウントの Runtime Monitoring ステータスを無効にしても、共有 VPC エンドポイントポリシーと所有者アカウントに存在するセキュリティグループには影響しません。 共有 VPC エンドポイントとセキュリティグループを削除するには、共有 VPC 所有者アカウントで Runtime Monitoring または自動エージェント設定ステータスを無効にする必要があります。 • 共有 VPC 参加者アカウントは、共有 VPC 所有者アカウントでホストされている共有 VPC エンドポイントとセキュリティグループを削除することはできません。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
エージェントが報告しない	(意図的に空にしてあります)	問題のタイプがサポート終了になりました。この問題が引き続き発生し、まだ行っていない場合は、Amazon EC2 の GuardDuty 自動エージェントを有効にします。 それでも問題が解決しない場合は、Runtime Monitoring を数分間無効にしてから、再度有効にすることを検討してください。

ECS-EC2 Bottlerocket のランタイムカバレッジとトラブルシューティング

Amazon ECS クラスターの Bottlerocket Amazon EC2 インスタンスの場合、GuardDuty はインスタンスレベルでランタイムカバレッジを評価します。GuardDuty セキュリティエージェントは、インストールされた RPM または Debian パッケージとしてではなく、Bottlerocket のホストコンテナとして実行されます。

Bottlerocket Amazon EC2 インスタンスは、EC2 インスタンスランタイムカバレッジタブの下に表示されます。Bottlerocket インスタンスのカバレッジ統計を確認するには、GuardDuty コンソールを開き、ナビゲーションペインで Runtime Monitoring を選択し、Runtime coverage タブを選択します。EC2 インスタンスランタイムカバレッジでは、各インスタンスのカバレッジステータスを表示し、アカウント ID、エージェントバージョン、またはカバレッジステータスでフィルタリングできます。詳細については、「カバレッジ統計の確認」を参照してください。

Bottlerocket ECS-EC2 ランタイムカバレッジの問題のトラブルシューティング

Bottlerocket Amazon EC2 インスタンスのカバレッジステータスが異常である場合は、問題列で理由を表示できます。

次の表に、問題のタイプと対応するトラブルシューティングステップを示します。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
エージェントからの報告なし	SSM 通知を待機中	SSM 通知の受信には数分かかる場合があります。 Bottlerocket インスタンスが SSM マネージドであることを確認します。SSM エージェントは、デフォルトで Bottlerocket コントロールコンテナで実行されます。インスタンスプロファイルに `AmazonSSMManagedInstanceCore` および `AmazonEC2ContainerRegistryReadOnly` 管理ポリシーが含まれていることを確認します。
エージェントからの報告なし	該当しない	エージェントホストコンテナが有効で実行されていることを確認します。 1. SSM Session Manager を使用してインスタンスに接続します。 2. 管理者コンテナ を入力します`enter-admin-container` 。 3. 設定の確認: `apiclient get settings.host-containers.amazon-guardduty-agent` 。`enabled` が `true`で、 `superpowered`

問題タイプ	問題メッセージ	トラブルシューティングのステップ
		が であることを確認しますtrue。 4. エージェントプロセスが実行されているかどうかを確認します: `sheltie ps aux | grep guardduty` 。 5. 次のいずれかを使用してエージェントログを確認します。 • `sheltie journalctl -u host-containers@amazon-guardduty-agent` • シェルティセッション (`sudo sheltie`): `ls /var/log/amzn-guardduty-agent/` 。
エージェントからの報告なし	該当しない	インスタンスが Amazon ECR エンドポイントと GuardDuty VPC エンドポイントにネットワーク接続されていることを確認します。インスタンスがプライベートサブネットで実行されている場合は、Amazon ECR と GuardDuty の両方に VPC エンドポイントが設定されていることを確認してください。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
エージェントからの報告なし	該当しない	組織にサービスコントロールポリシー (SCP) がある場合は、アクセス許可の境界で`guardduty:SendSecurityTelemetry` アクセス許可が許可されていることを確認します。
SSM 関連付けの作成に失敗	GuardDuty SSM の関連付けはアカウントに既に存在します。	1. 既存の関連付けを手動で削除します。詳細については、「AWS Systems Manager ユーザーガイド」の「関連付けを削除する」を参照してください。 2. 関連付けを削除した後、Amazon EC2 の GuardDuty 自動エージェント設定を無効にしてから再度有効にします。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
SSM 関連付けの作成に失敗	アカウントに SSM の関連付けが多すぎます。	次の 2 つのオプションのいずれかを選択します。 • 未使用の SSM 関連付けを削除します。詳細については、「AWS Systems Manager ユーザーガイド」の「関連付けを削除する」を参照してください。 • アカウントがクォータ引き上げの対象かどうかを確認します。詳細については、「AWS 全般のリファレンス」の「System Manager Service Quotas」を参照してください。
SSM 関連付けの更新に失敗しました	GuardDuty SSM の関連付けがアカウントに存在しません。	GuardDuty SSM 関連付けがアカウントに存在しません。Runtime Monitoring を無効にしてから再度有効にします。
SSM 関連付けの削除に失敗	GuardDuty SSM の関連付けがアカウントに存在しません。	SSM 関連付けがアカウントに存在しません。SSM 関連付けが意図的に削除された場合、アクションは必要ありません。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
SSM インスタンス関連付けの実行に失敗	アーキテクチャ要件またはその他の前提条件が満たされていません。	検証済み要件の詳細については、「」を参照してくださいECS-EC2 Bottlerocket サポートの前提条件。 この問題が解決しない場合は、SSM ステートマネージャーの実行履歴を確認してください。 1. https://console.aws.amazon.com/systems-manager/ で AWS Systems Manager コンソールを開きます。 2. ナビゲーションペインの [ノード管理] で、[ステートマネージャー] を選択します。 3. [ドキュメント名] プロパティでフィルタリングし、「AmazonGuardDuty-ConfigureRuntimeMonitoringSsmPlugin」と入力します。 4. 対応する関連付け ID を選択し、その [実行履歴] を表示します。 5. 実行履歴を使用して、障害を表示し、潜在的な根本原因を特定し、解決を試みます。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
エージェントが切断されました	該当しない	• 管理コンテナからエージェントログを確認します。 `enter-admin-container sheltie journalctl -u host-containers@amazon-guardduty-agent` • ホストコンテナ設定の `source` フィールドが有効な Amazon ECR URI を指していることを確認することで、エージェントコンテナイメージが正常にプルされたことを確認します。 • インスタンスプロファイルに Amazon ECR プルアクセス許可があることを確認します。詳細については、「EC2 インスタンスを SSM で管理し、インスタンスのアクセス許可を設定する」を参照してください。

問題タイプ	問題メッセージ	トラブルシューティングのステップ
エージェントがプロビジョニングされていない	除外タグを持つインスタンスは Runtime Monitoring から除外されます。	GuardDuty は、除外タグ `GuardDutyManaged :false` を使用して起動された Amazon EC2 インスタンスからランタイムイベントを受信しません。 このインスタンスからランタイムイベントを受信するには、除外タグを削除します。
エージェントがプロビジョニングされていない	カーネルバージョンがサポートされているバージョンより低いです。	自動エージェント設定のために、最新の Bottlerocket ECS-2 または ECS-3 最適化 AMI `v1.62.1`以降に更新します。
ECR イメージのプルに失敗しました	エージェントコンテナイメージを Amazon ECR からプルできませんでした。	• インスタンスプロファイルに`AmazonEC2ContainerRegistryReadOnly` 管理ポリシーまたは必要な個々の Amazon ECR アクセス許可が含まれていることを確認します。 • リージョンの Amazon ECR エンドポイントへのネットワーク接続を確認します。 • プライベートサブネットを使用する場合は、VPC エンドポイントが設定されていることを確認してください。詳細については、「ネットワーク接続要件」を参照してください。

Amazon ECS クラスターのランタイムカバレッジとトラブルシューティング

Amazon ECS クラスターのランタイムカバレッジには、 AWS Fargate および Amazon ECS コンテナインスタンスで実行されるタスクが含まれます[1]。

Fargate で実行される Amazon ECS クラスターの場合、ランタイムカバレッジはタスクレベルで評価されます。ECS クラスターのランタイムカバレッジには、Fargate (ECS のみ) の Runtime Monitoring と自動エージェント設定を有効にした後に実行を開始した Fargate タスクが含まれます。デフォルトでは、Fargate タスクは変更できません。GuardDuty は、セキュリティエージェントをインストールして、既に実行されているタスクのコンテナをモニタリングすることはできません。このような Fargate タスクを含めるには、タスクを停止して再度開始する必要があります。関連付けられたサービスがサポートされていることを確認します。

Amazon ECS コンテナの詳細については、キャパシティの作成に関するページを参照してください。

内容

- カバレッジ統計の確認
- カバレッジステータス変更時の EventBridge 通知
- Amazon ECS-Fargate ランタイムカバレッジの問題のトラブルシューティング

カバレッジ統計の確認

自分のアカウントまたはメンバーアカウントに関連付けられた Amazon ECS リソースのカバレッジ統計は、選択した AWS リージョン内のすべての ECS クラスターに対する正常な ECS クラスターの割合です。これには、Fargate インスタンスと Amazon EC2 インスタンスの両方に関連付けられた Amazon ECS クラスターのカバレッジが含まれます。次の式はこれを次のように表します。

(正常なクラスター/すべてのクラスター)*100

考慮事項

- ECS クラスターのカバレッジ統計には、その ECS クラスターに関連付けられた Fargate タスクまたは ECS コンテナインスタンスのカバレッジステータスが含まれます。Fargate タスクのカバレッジステータスには、実行中のタスク、または最近実行を終了したタスクが含まれます。

- ECS クラスターのランタイムカバレッジタブの Container instances covered フィールドには、Amazon ECS クラスターに関連付けられた Amazon EC2 コンテナインスタンスのカバレッジステータスが表示されます。

  Amazon ECS クラスターが Fargate 起動タイプのみを実行する場合、このフィールドにはダッシュ (-) が表示され、カウントが適用されないことを示します。Fargate 専用クラスターには、対象とする Amazon EC2 コンテナインスタンスはありません。
- Amazon ECS クラスターがセキュリティエージェントを持たない Amazon EC2 インスタンスに関連付けられている場合、Amazon ECS クラスターのカバレッジステータスも [異常] になります。

  関連付けられた Amazon EC2 インスタンスのカバレッジの問題を特定してトラブルシューティングするには、Amazon EC2 インスタンスの「Amazon EC2 ランタイムカバレッジの問題のトラブルシューティング」を参照してください。

いずれかのアクセス方法を選択して、アカウントのカバレッジ統計を確認してください。

Console

- にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
- ナビゲーションペインで、[Runtime Monitoring] を選択します。
- [ランタイムカバレッジ] タブを選択します。
- [ECS クラスターのランタイムカバレッジ] タブでは、[クラスターリスト] テーブルにある各 Amazon ECS クラスターのカバレッジステータス別に集計されたカバレッジ統計を表示できます。
  - [クラスターリスト] テーブルは次の列でフィルタリングできます。
    - アカウント ID
    - [クラスター名]
    - [エージェント管理タイプ]
    - [カバレッジステータス]
  - Amazon ECS クラスターのいずれかの [カバレッジステータス] が [異常] の場合、[問題] 列には、[異常] ステータスである理由に関する追加情報が含まれています。

    Amazon ECS クラスターが Amazon EC2 インスタンスに関連付けられている場合は、[EC2 インスタンスランタイムカバレッジ] タブに移動し、[クラスター名] フィールドでフィルタリングして、関連する [問題] を表示します。

API/CLI

- 独自の有効なディテクター ID、現在のリージョン、サービスエンドポイントを使用して ListCoverage API を実行します。この API を使用して、インスタンスリストをフィルタリングしたり、ソートしたりできます。
  - 以下の CriterionKey のオプションのいずれかを使用して例 filter-criteria を変更できます。
    - ACCOUNT_ID
    - ECS_CLUSTER_NAME
    - COVERAGE_STATUS
    - MANAGEMENT_TYPE
  - 以下のオプションで sort-criteria の例 AttributeName を変更できます。
    - ACCOUNT_ID
    - COVERAGE_STATUS
    - ISSUE
    - ECS_CLUSTER_NAME
    - UPDATED_AT

      このフィールドは、関連する Amazon ECS クラスターで新しいタスクが作成されたとき、または対応するカバレッジステータスに変更があったときにのみ更新されます。
  - *max-results* (最大 50) を変更できます。
  - アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty --region us-east-1 list-coverage --detector-
id 12abc34d567e8fa901bc2d34e56789f0 --sort-criteria '{"AttributeName":
 "ECS_CLUSTER_NAME", "OrderBy": "DESC"}' --filter-criteria
 '{"FilterCriterion":[{"CriterionKey":"ACCOUNT_ID", "FilterCondition":
{"EqualsValue":"111122223333"}}] }'  --max-results 5
```

- GetCoverageStatistics API を実行すると、statisticsType に基づいてカバレッジ集計された統計情報を取得できます。
  - 例 statisticsType を次のオプションのいずれかに変更できます。

- COUNT_BY_COVERAGE_STATUS - カバレッジステータス別に集計された ECS クラスターのカバレッジステータスを表します。
- COUNT_BY_RESOURCE_TYPE – リスト内の AWS リソースのタイプに基づいて集計されたカバレッジ統計。
- コマンドで例 filter-criteria を変更できます。CriterionKey に対して次のオプションを設定できます。
  - ACCOUNT_ID
  - ECS_CLUSTER_NAME
  - COVERAGE_STATUS
  - MANAGEMENT_TYPE
  - INSTANCE_ID
- アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty --region us-east-1 get-coverage-statistics --detector-
id 12abc34d567e8fa901bc2d34e56789f0 --statistics-type COUNT_BY_COVERAGE_STATUS
 --filter-criteria '{"FilterCriterion":[{"CriterionKey":"ACCOUNT_ID",
 "FilterCondition":{"EqualsValue":"123456789012"}}] }'
```

カバレッジ問題を解決する方法の詳細については、「Amazon ECS-Fargate ランタイムカバレッジの問題のトラブルシューティング」を参照してください。

カバレッジステータス変更時の EventBridge 通知

Amazon ECS クラスターのカバレッジステータスは [異常] と表示される場合があります。カバレッジステータスの変化を検出するためにカバレッジステータスを定期的に監視し、ステータスが [異常] の場合はトラブルシューティングすることをお勧めします。または、Amazon EventBridge ルールを作成して、カバレッジステータスが [異常] から [正常] に変更された場合やそれ以外の変更があった場合に通知を受け取ることもできます。デフォルトでは、GuardDuty はこれをアカウントの EventBridge バスに公開します。

通知スキーマの例

EventBridge のルールでは、あらかじめ定義されたサンプルイベントとイベントパターンを使用して、カバレッジステータスの通知を受け取ることができます。EventBridge ルールの作成の詳細については、「Amazon EventBridge ユーザーガイド」の「ルールの作成」を参照してください。

さらに、次の通知スキーマの例を使用して、カスタムイベントパターンを作成します。アカウントの値を必ず置き換えてください。Amazon ECS クラスターのカバレッジステータスが Healthy から Unhealthy に変更されたときに通知を受け取るには、detail-type が *GuardDuty Runtime Protection Unhealthy* である必要があります。カバレッジステータスが Unhealthy から Healthy に変更されたときに通知を受け取るには、detail-type の値を *GuardDuty Runtime Protection Healthy* に置き換えます。

```
{
  "version": "0",
  "id": "event ID",
  "detail-type": "GuardDuty Runtime Protection Unhealthy",
  "source": "aws.guardduty",
  "account": "AWS ##### ID",
  "time": "event timestamp (string)",
  "region": "AWS #####",
  "resources": [
       ],
  "detail": {
    "schemaVersion": "1.0",
    "resourceAccountId": "string",
    "currentStatus": "string",
    "previousStatus": "string",
    "resourceDetails": {
        "resourceType": "ECS",
        "ecsClusterDetails": {
          "clusterName":"",
          "fargateDetails":{
            "issues":[],
            "managementType":""
          },
          "containerInstanceDetails":{
            "coveredContainerInstances":int,
            "compatibleContainerInstances":int
          }
        }
    },
```

```
    "issue": "string",
    "lastUpdatedAt": "timestamp"
  }
}
```

Amazon ECS-Fargate ランタイムカバレッジの問題のトラブルシューティング

Amazon ECS クラスターのカバレッジステータスが [異常] の場合、その理由を [問題] 列で確認できます。

次のテーブルは、Fargate (Amazon ECS のみ) の問題に推奨されるトラブルシューティング手順を示しています。Amazon EC2 インスタンスカバレッジの問題の詳細については、Amazon EC2 インスタンスの「Amazon EC2 ランタイムカバレッジの問題のトラブルシューティング」を参照してください。

問題タイプ	追加情報	推奨されるトラブルシューティングの手順
エージェントが報告しない	エージェントが TaskDefinition - ' *TASK_DEFINITION* ' のタスクを報告しない	Amazon ECS クラスターのタスクの VPC エンドポイントが正しく設定されていることを確認します。詳細については、「VPC エンドポイント設定の検証」を参照してください。 組織にサービスコントロールポリシー (SCP) がある場合は、アクセス許可の境界が guardduty:SendSecurityTelemetry アクセス許可を制限していないことを確認します。詳細については、「マルチアカウント環境での組織サービスコントロールポリシーの検証」を参照してください。

問題タイプ	追加情報	推奨されるトラブルシューティングの手順
	VPC_ISSUE ; for task in TaskDefinition - '*TASK_DEFINITION* '	追加情報の VPC の問題の詳細を確認してください。
エージェントが終了した	終了コード: TaskDefinition - ' *TASK_DEFINITION* ' のタスク用 EXIT_CODE	追加情報で問題の詳細を確認してください。
	理由: TaskDefinition - '*TASK_DEFINITION* ' のタスクの *[##]*	
	終了コード: 理由付き EXIT_CODE : TaskDefinition - ' *TASK_DEFINITION* ' のタスクの「*EXIT_CODE* 」	

問題タイプ	追加情報	推奨されるトラブルシューティングの手順
	エージェントが終了しました: 理由: CannotPullContainerError : プルイメージマニフェストが再試行されました...	このシナリオでは、GuardDuty がサイドカーコンテナイメージをプルできない可能性があります。タスクは引き続き実行されますが、GuardDuty は潜在的な脅威を検出できません。以下のトラブルシューティング手順を 1 つずつ実行して、カバレッジの問題の解決に役立つかどうかを確認します。 • アクセス許可: タスク実行ロールに必要な ECR アクセス許可が アクセス許可の要件 にリストされていることを確認します。 • ネットワーク接続: 「ネットワーク接続要件」で説明されているように、Fargate タスクがパブリックインターネットアクセスまたは適切に設定された VPC エンドポイントを介して ECR に到達できることを確認します。 • セキュリティグループ設定:セキュリティグループの設定 で説明されているように、セキュリティグループがポート 443 の S3 マネージドプレフィックスリストへのアウトバウンドアクセ

問題タイプ	追加情報	推奨されるトラブルシューティングの手順
		スを許可していることを確認します。 • クラスターの リージョンで AWSSupport-TroubleshootECSTaskFailedToStart ランブックを実行して、具体的な問題を特定します。 • エラーメッセージのタスクログを表示します。これを実行する方法の詳細については、「Amazon Elastic Container Service デベロッパーガイド」の「Amazon ECS コンテナエージェントのログの表示」を参照してください。 一般的なエラーとトラブルシューティングについては、「Amazon Elastic Container Service デベロッパーガイド」の「Amazon ECS のトラブルシューティング」を参照してください。 これら 3 つのコンポーネント (アクセス許可、ネットワーク接続、セキュリティグループ設定) は独立していますが、すべて Amazon ECR から GuardDuty コンテナイメージ

問題タイプ	追加情報	推奨されるトラブルシューティングの手順
		を正常にダウンロードするために必要です。 問題が解決しない場合は、「AWS Step Functions ワークフローが予期せず失敗している」を参照してください。
VPC エンドポイントの作成に失敗	プライベート DNS を有効にするには、enableDnsSupport とenableDnsHostnames VPC 属性の両方が *vpcId* に対して true に設定されている必要があります (サービス: EC2、ステータス コード: 400、リクエスト ID: *a1b2c3d4-5678-90ab-cdef-EXAMPLE11111*)。	次の VPC 属性が true - enableDnsSupport および enableDnsHostnames に設定されていることを確認する必要があります。詳細については、「VPC の DNS 属性」を参照してください。 https://console.aws.amazon.com/vpc/ にある Amazon VPC コンソールを使用して Amazon VPC を作成する場合は、必ず [DNS ホスト名を有効にする] および [解決を有効にする] の両方を選択してください。詳細については、「VPC 設定オプション」を参照してください。
エージェントがプロビジョニングされていない	TaskDefinition - '*TASK_DEFINITION* ' のタスクに対する *SERVICE* によるサポートされていない呼び出し	このタスクは、サポートされていない *SERVICE* によって呼び出されました。

問題タイプ	追加情報	推奨されるトラブルシューティングの手順
	TaskDefinition - '*TASK_DEFINITION* ' のタスクでサポートされていない CPU アーキテクチャ '*TYPE*'	このタスクは、サポートされていない CPU アーキテクチャで実行されています。サポートされる CPU アーキテクチャの詳細については、「アーキテクチャ要件の検証」を参照してください。
	TaskDefinition - '*TASK_DEFINITION* ' に TaskExecutionRole がありません	ECS タスク実行ロールがありません。タスク実行ロールと必要なアクセス許可の提供については、「コンテナイメージアクセスの前提条件」を参照してください。
	TaskDefinition - '*TASK_DEFINITION* ' のタスクのネットワーク設定 '*CONFIGURATION_DETAILS* ' がありません	VPC 設定が欠落しているか、サブネットが欠落または空であるため、ネットワーク設定の問題が表示される場合があります。 ネットワーク設定が正しいことを確認します。詳細については、「コンテナイメージアクセスの前提条件」を参照してください。 詳細については、「Amazon Elastic Container Service デベロッパーガイド」の「Amazon ECS タスク定義パラメータ」を参照してください。

問題タイプ	追加情報	推奨されるトラブルシューティングの手順
	クラスターに除外タグがあったときに開始されたタスクは、Runtime Monitoring から除外されます。影響を受けるタスク ID (複数可): 「*TASK_ID*	定義済みの GuardDuty タグを `GuardDutyManaged -true` から `GuardDutyManaged -false` に変更すると、GuardDuty はこの Amazon ECS クラスターのランタイムイベントを受信しなくなります。 タグを `GuardDutyManaged -true` に更新し、タスクを再起動します。
	クラスターに除外タグがある場合にデプロイされたサービスは、Runtime Monitoring から除外されます。影響を受けるサービス名 (複数可): 「*SERVICE_NAME* 」	除外タグ `GuardDutyManaged -false` でデプロイされたサービスの場合、GuardDuty はこの Amazon ECS クラスターのランタイムイベントを受信しません。 タグを `GuardDutyManaged -true` に更新し、サービスを再デプロイします。
	自動エージェント設定を有効にする前に開始されたタスクは対象外です。影響を受けるタスク ID (複数可): 「*TASK_ID*」	Amazon ECS の自動エージェント設定を有効にする前に起動されたタスクがクラスターに含まれている場合、GuardDuty はこれを保護することができません。GuardDuty によってモニタリングされるようにタスクを再起動します。

問題タイプ	追加情報	推奨されるトラブルシューティングの手順
	自動エージェント設定を有効にする前にデプロイされたサービスは対象外です。影響を受けるサービス名 (複数可): 「*SERVICE_NAME* 」	Amazon ECS の自動エージェント設定を有効にする前にサービスがデプロイされると、GuardDuty は ECS クラスターのランタイムイベントを受信しません。
	サービス「*SERVICE_NAME* 」を修正/トラブルシューティングするには、新しいデプロイが必要です。ドキュメントを参照、影響を受けるサービス名 (複数可): 「*SERVICE_NAME* 」	Runtime Monitoring を有効にする前に開始されたサービスはサポートされていません。 「Amazon Elastic Container Service デベロッパーガイド」の「コンソールを使用して Amazon ECS サービスを更新する」の手順に従って、サービスを再起動するか、`forceNewDeployment` オプションでサービスを更新できます。または、「Amazon Elastic Container Service API リファレンス」の「UpdateService」 のステップを使用することもできます。

問題タイプ	追加情報	推奨されるトラブルシューティングの手順
	Runtime Monitoring を有効にする前に開始されたタスクには、再起動が必要です。影響を受けるタスク ID (複数可): 「*TASK_ID_1* 」	Amazon ECS では、タスクは変更できません。実行中の AWS Fargate タスクのランタイムの動作を評価するには、Runtime Monitoring がすでに有効になっていることを確認し、GuardDuty のタスクを再起動してコンテナサイドカーを追加します。

問題タイプ	追加情報	推奨されるトラブルシューティングの手順
その他	TaskDefinition - '*TASK_DEFINITION* ' のタスクにおける未確認の問題	次の質問を使用して、問題の根本原因を特定します。 • タスクは Runtime Monitoring を有効にする前に開始されましたか? Amazon ECS では、タスクは変更できません。実行中の Fargate タスクのランタイムの動作を評価するには、Runtime Monitoring がすでに有効になっていることを確認し、GuardDuty のタスクを再起動してコンテナサイドカーを追加します。 • このタスクは、Runtime Monitoring を有効にする前に開始されたサービスデプロイの一部ですか? 「はい」の場合は、サービスを再起動するか、「サービスの更新」の手順に従い、forceNewDeployment によってサービスを更新できます。 [UpdateService] または [AWS CLI] を使用することもできます。

問題タイプ	追加情報	推奨されるトラブルシューティングの手順
		• タスクは ECS クラスターを Runtime Monitoring から除外してから起動しましたか? 定義済みの GuardDuty タグを `GuardDuty Managed -true` から `GuardDuty Managed -false` に変更すると、GuardDuty は ECS クラスターのランタイムイベントを受信しなくなります。 • サービスには、古い形式の `taskArn` のタスクが含まれていますか ? GuardDuty Runtime Monitoring は、古い形式の `taskArn` を持つタスクのカバレッジをサポートしていません。 Amazon ECS リソースの Amazon リソースネーム (ARN) の詳細については、「Amazon リソースネーム (ARN) と ID」を参照してください。

Amazon EKS クラスターのランタイムカバレッジとトラブルシューティング

Runtime Monitoring を有効にして EKS の GuardDuty セキュリティエージェント (アドオン) をインストールした後、EKS クラスターのカバレッジの評価を開始できます。

内容

- カバレッジ統計の確認
- カバレッジステータス変更時の EventBridge 通知
- Amazon EKS ランタイムカバレッジの問題のトラブルシューティング

カバレッジ統計の確認

自分のアカウントまたはメンバーアカウントに関連付けられた EKS クラスターのカバレッジ統計は、選択した AWS リージョンのすべての EKS クラスターに対する正常な EKS クラスターの割合です。次の式はこれを次のように表します。

(正常なクラスター/すべてのクラスター)*100

いずれかのアクセス方法を選択して、アカウントのカバレッジ統計を確認してください。

Console

- にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
- ナビゲーションペインで、[Runtime Monitoring] を選択します。
- [EKS クラスターのランタイムカバレッジ] タブを選択します。
- [EKS クラスターのランタイムカバレッジ] タブでは、[クラスターリスト] テーブルにあるカバレッジステータス別に集計されたカバレッジ統計を表示できます。
  - [クラスターリスト] テーブルは次の列でフィルタリングできます。
    - クラスター名
    - アカウント ID
    - [エージェント管理タイプ]
    - [カバレッジステータス]
    - [アドオンバージョン]

- EKS クラスターのいずれかの [カバレッジステータス] が [異常] の場合、[問題] 列には、[異常] ステータスである理由に関する追加情報が含まれている場合があります。

API/CLI

- 独自の有効なディテクター ID、リージョン、およびサービスエンドポイントを使用して ListCoverage API を実行します。この API を使用して、クラスターリストをフィルタリングしたり、ソートしたりできます。
  - 以下の `CriterionKey` のオプションのいずれかを使用して例 `filter-criteria` を変更できます。
    - `ACCOUNT_ID`
    - `CLUSTER_NAME`
    - `RESOURCE_TYPE`
    - `COVERAGE_STATUS`
    - `ADDON_VERSION`
    - `MANAGEMENT_TYPE`
  - 以下のオプションで `sort-criteria` の例 `AttributeName` を変更できます。
    - `ACCOUNT_ID`
    - `CLUSTER_NAME`
    - `COVERAGE_STATUS`
    - `ISSUE`
    - `ADDON_VERSION`
    - `UPDATED_AT`
  - *`max-results`* (最大 50) を変更できます。
  - アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty --region us-east-1 list-coverage --detector-
id 12abc34d567e8fa901bc2d34e56789f0 --sort-criteria '{"AttributeName":
 "CLUSTER_NAME", "OrderBy": "DESC"}' --filter-criteria
 '{"FilterCriterion":[{"CriterionKey":"ACCOUNT_ID", "FilterCondition":
{"EqualsValue":"111122223333"}}] }'  --max-results 5
```

- GetCoverageStatistics API を実行すると、statisticsType に基づいてカバレッジ集計された統計情報を取得できます。
  - 例 statisticsType を次のオプションのいずれかに変更できます。
    - COUNT_BY_COVERAGE_STATUS - カバレッジステータス別に集計された EKS クラスターのカバレッジ統計を表します。
    - COUNT_BY_RESOURCE_TYPE – リスト内の AWS リソースのタイプに基づいて集計されたカバレッジ統計。
    - コマンドで例 filter-criteria を変更できます。CriterionKey に対して次のオプションを設定できます。
      - ACCOUNT_ID
      - CLUSTER_NAME
      - RESOURCE_TYPE
      - COVERAGE_STATUS
      - ADDON_VERSION
      - MANAGEMENT_TYPE
  - アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty --region us-east-1 get-coverage-statistics --detector-
id 12abc34d567e8fa901bc2d34e56789f0 --statistics-type COUNT_BY_COVERAGE_STATUS
 --filter-criteria '{"FilterCriterion":[{"CriterionKey":"ACCOUNT_ID",
 "FilterCondition":{"EqualsValue":"123456789012"}}] }'
```

EKS クラスターのカバレッジステータスが [異常] である場合は、「Amazon EKS ランタイムカバレッジの問題のトラブルシューティング」を参照してください。

カバレッジステータス変更時の EventBridge 通知

アカウントの EKS クラスターのカバレッジステータスが [異常] と表示されることがあります。カバレッジステータスが [異常] になったことを検出するためにカバレッジステータスを定期的にモニタリングし、ステータスが [異常] の場合はトラブルシューティングすることをお勧めします。または、Amazon EventBridge ルールを作成して、カバレッジステータスが Unhealthy から Healthy に変更された場合やそれ以外の変更があった場合に通知を受け取ることもできます。デフォルトでは、GuardDuty はこれをアカウントの EventBridge バスに公開します。

通知スキーマの例

EventBridge のルールでは、あらかじめ定義されたサンプルイベントとイベントパターンを使用して、カバレッジステータスの通知を受け取ることができます。EventBridge ルールの作成の詳細については、「Amazon EventBridge ユーザーガイド」の「ルールの作成」を参照してください。

さらに、次の通知スキーマの例を使用して、カスタムイベントパターンを作成します。アカウントの値を必ず置き換えてください。Amazon EKS クラスターのカバレッジステータスが Healthy から Unhealthy に変更されたときに通知を受け取るには、detail-type が *GuardDuty Runtime Protection Unhealthy* である必要があります。カバレッジステータスが Unhealthy から Healthy に変更されたときに通知を受け取るには、detail-type の値を *GuardDuty Runtime Protection Healthy* に置き換えます。

```
{
  "version": "0",
  "id": "event ID",
  "detail-type": "GuardDuty Runtime Protection Unhealthy",
  "source": "aws.guardduty",
  "account": "AWS ##### ID",
  "time": "event timestamp (string)",
  "region": "AWS #####",
  "resources": [
       ],
  "detail": {
    "schemaVersion": "1.0",
    "resourceAccountId": "string",
    "currentStatus": "string",
    "previousStatus": "string",
    "resourceDetails": {
        "resourceType": "EKS",
        "eksClusterDetails": {
            "clusterName": "string",
            "availableNodes": "string",
             "desiredNodes": "string",
             "addonVersion": "string"
         }
    },
    "issue": "string",
    "lastUpdatedAt": "timestamp"
  }
}
```

Amazon EKS ランタイムカバレッジの問題のトラブルシューティング

EKS クラスターのカバレッジステータスが Unhealthy の場合、GuardDuty コンソールの [問題] 列または CoverageResource データ型を使用して、対応するエラーを確認できます。

包含タグまたは除外タグを使用して EKS クラスターを選択的にモニタリングする場合、タグの同期に時間がかかることがあります。これにより、関連する EKS クラスターのカバレッジステータスに影響が及ぶ可能性があります。対応するタグ (包含または除外) を削除してから、もう一度追加してみることができます。詳細については、「Amazon EKS ユーザーガイド」の「Amazon EKS リソースのタグ付け」を参照してください。

カバレッジ問題の構造は `Issue type:Extra information` です。通常、問題にはオプションの追加情報があり、特定のクライアント側の例外や問題に関する説明が含まれる場合があります。追加情報に基づき、EKS クラスターのカバレッジの問題をトラブルシューティングするための推奨ステップを以下の表に示します。

問題のタイプ (プレフィックス)	追加情報	推奨されるトラブルシューティングの手順
アドオンの作成に失敗	アドオン `aws-guardduty-agent` は、クラスター *`ClusterName`* の現在のクラスターバージョンと互換性がありません。指定されたアドオンはサポートされていません。	`aws-guardduty-agent` EKS アドオンのデプロイをサポートする Kubernetes バージョンのいずれかを使用していることを確認してください。詳細については、「GuardDuty セキュリティエージェントでサポートされている Kubernetes のバージョン」を参照してください。Kubernetes バージョンの更新については、「Amazon EKS クラスターの Kubernetes バージョンの更新」を参照してください。
アドオンの作成に失敗	EKS アドオンの問題 - `AddonIssueCode` : `AddonIssueMessage`	特定のアドオン問題コードの推奨ステップについては、「Troubleshooting steps for

問題のタイプ (プレフィックス)	追加情報	推奨されるトラブルシューティングの手順
アドオンの更新に失敗しました アドオンのステータスが異常		Addon creation/update error with Addon issue code」を参照してください。 この問題で発生する可能性のあるアドオンの問題コードのリストについては、「AddonIssue」を参照してください。
VPC エンドポイントの作成に失敗	VPC エンドポイントの作成は、共有 VPC *vpcId* ではサポートされていません	Runtime Monitoring では、組織内の共有 VPC の使用がサポートされるようになりました。アカウントがすべての前提条件を満たしていることを確認します。詳細については、「共有 VPC を使用するための前提条件」を参照してください。
	自動エージェント設定で共有 VPC を使用する場合のみ 共有 VPC *vpcId* の所有者アカウント ID *111122223333* では、Runtime Monitoring、自動エージェント設定、またはその両方が有効になっていません。	共有 VPC 所有者アカウントは、少なくとも 1 つのリソースタイプ (Amazon EKS または Amazon ECS (AWS Fargate)) の Runtime Monitoring および自動エージェント設定を有効にする必要があります。詳細については、「GuardDuty Runtime Monitoring に固有の前提条件」を参照してください。

問題のタイプ (プレフィックス)	追加情報	推奨されるトラブルシューティングの手順
	プライベート DNS を有効にするには、`enableDnsSupport` と `enableDnsHostnames` VPC 属性の両方が *`vpcId`* に対して `true` に設定されている必要があります (サービス: Ec2、ステータス コード: 400、リクエスト ID: *`a1b2c3d4-5678-90ab-cdef-EXAMPLE11111`*)。	次の VPC 属性が `true` - `enableDnsSupport` および `enableDnsHostnames` に設定されていることを確認する必要があります。詳細については、「VPC の DNS 属性」を参照してください。 https://console.aws.amazon.com/vpc/ にある Amazon VPC コンソールを使用して Amazon VPC を作成する場合は、必ず [DNS ホスト名を有効にする] および [解決を有効にする] の両方を選択してください。詳細については、「VPC 設定オプション」を参照してください。

問題のタイプ (プレフィックス)	追加情報	推奨されるトラブルシューティングの手順
共有 VPC エンドポイントの削除に失敗	共有 VPC エンドポイントの削除は、アカウント ID *111122223333* 、共有 VPC *vpcId*、所有者アカウント ID *555555555555* では許可されません。	考えられるステップ: • 共有 VPC 参加者アカウントの Runtime Monitoring ステータスを無効にしても、共有 VPC エンドポイントポリシーと所有者アカウントに存在するセキュリティグループには影響しません。 共有 VPC エンドポイントとセキュリティグループを削除するには、共有 VPC 所有者アカウントで Runtime Monitoring または自動エージェント設定ステータスを無効にする必要があります。 • 共有 VPC 参加者アカウントは、共有 VPC 所有者アカウントでホストされている共有 VPC エンドポイントとセキュリティグループを削除することはできません。
ローカル EKS クラスター	EKS アドオンはローカルのアウトポストクラスターではサポートされていません。	実用的ではありません。 詳細については、AWS「Outposts の Amazon EKS」を参照してください。

問題のタイプ (プレフィックス)	追加情報	推奨されるトラブルシューティングの手順
EKS Runtime Monitoring の有効化許可が付与されていません	(追加情報が表示される場合と表示されない場合があります)	1. この問題に関する追加情報がある場合は、根本原因を修正して次の手順に従ってください。 2. EKS Runtime Monitoring を切り替えて、オフにしてから再度オンにします。GuardDuty エージェントも必ず、GuardDuty を使用して自動的にデプロイするか、手動でデプロイしてください。
進行中の EKS Runtime Monitoring 有効化リソースのプロビジョニング	(追加情報が表示される場合と表示されない場合があります)	実用的ではありません。 EKS Runtime Monitoring を有効にしても、リソースプロビジョニングステップが完了するまでカバレッジステータスが `Unhealthy` として残る場合があります。カバレッジステータスは定期的に監視および更新されます。
その他 (その他の問題)	承認が失敗したためにエラーが発生しました	EKS Runtime Monitoring を切り替えて、オフにしてから再度オンにします。GuardDuty エージェントも、GuardDuty を使用して自動的にデプロイするか、手動でデプロイするようにしてください。

アドオン問題コードによるアドオンの作成/更新エラーのトラブルシューティング手順

アドオンの作成または更新エラー	トラブルシューティングのステップ
EKS アドオンの問題 - InsufficientNumberOfReplicas : 必要な数の複製がないため、アドオンは異常です。	• 問題メッセージを使用して、根本原因を特定して修正できます。クラスターの説明から開始できます。例えば、kubectl describe pods を使用してポッド障害の根本原因を特定します。 根本原因を修正した後、ステップ (アドオンの作成または更新) を再試行します。 • 問題が解決しない場合は、Amazon EKS クラスターの VPC エンドポイントが正しく設定されていることを確認します。詳細については、「VPC エンドポイント設定の検証」を参照してください。
EKS アドオンの問題 - InsufficientNumberOfReplicas : 1 つ以上のポッドがスケジュールされた 0/x ノードを使用できないため、アドオンが異常です: x Insufficient cpu. preemption: not eligible due to preemptionPolicy=Never 。	この問題を解決するには、次のいずれかを実行します。 • GuardDuty エージェントのポッドの優先度を更新する: 設定可能なパラメータと値 の場合は、preemptionPolicy 値を PreemptLowerPriority としてサポートするオプションのいずれかに PriorityClass を設定します。ポッドの優先度の詳細については、「Kubernetes ドキュメント」の「Pod Priority と Preemption」を参照してください。 • インスタンスをスケールアップする: リソースを管理し、最適なインスタンスを選択するには、「Amazon EKS ユーザーガイド」の「ノードを使用してコンピューティングリソースを管理する」および「最適な Amazon
EKS アドオンの問題 - InsufficientNumberOfReplicas : 1 つ以上のポッドがスケジュールされた 0/x ノードを使用できないため、アドオンが異常です: x Too many pods. preemption: not eligible due to preemptionPolicy=Never 。	
EKS アドオンの問題 - InsufficientNumberOfReplicas : 1 つ以上の	

アドオンの作成または更新エラー	トラブルシューティングのステップ
ポッドがスケジュールされた `0/x` ノードを使用できないため、アドオンが異常です: `1 Insufficient memory. preemption: not eligible due to preemptionPolicy=Never` 。	EC2 ノードインスタンスタイプを選択する」を参照してください。 **Note** GuardDuty が最初に検出されたエラーのみをレポートするため、メッセージ `0/x` が表示されます。GuardDuty デーモンセットで実行中のポッドの実際の数は 0 を超える可能性があります。

アドオンの作成または更新エラー	トラブルシューティングのステップ
EKS アドオンの問題 - `InsufficientNumberOfReplicas` : 1 つ以上のポッドに待機コンテナ `CrashLoopBackOff: Completed` があるため、アドオンが異常です	ポッドに関連付けられたログを表示し、問題を特定できます。これを行う方法については、「Kubernetes ドキュメント」の「実行中のポッドのデバッグ」を参照してください。 以下のチェックリストを使用して、このアドオンの問題をトラブルシューティングします。 • Runtime Monitoring が有効になっていることを確認します。 • 検証済み OS ディストリビューションやサポートされている Kubernetes バージョンなどの Amazon EKS クラスターサポートの前提条件 が満たされていることを確認します。 • セキュリティエージェントを手動で管理する場合は、すべての VPC の VPC エンドポイントを作成したことを確認します。GuardDuty 自動設定を有効にしても、VPC エンドポイントが作成されていることを検証する必要があります。例えば、自動設定で共有 VPC を使用する場合などです。 これを検証するには、「VPC エンドポイント設定の検証」を参照してください。 • GuardDuty セキュリティエージェントが GuardDuty VPC エンドポイントのプライベート DNS を解決できることを確認します。エンドポイントを確認するには、「GuardDuty セキュリティエージェントの管理」の「エンドポイントのプライベート DNS 名」を参照してください。

アドオンの作成または更新エラー	トラブルシューティングのステップ
	これを行うには、Windows または Mac の nslookup ツール、または Linux の dig ツールのいずれかを使用できます。nslookup を使用する場合は、*us-west-2* リージョンを自分のリージョンに置き換えた後、以下のコマンドを使用できます。 `nslookup guardduty-data. us-west-2 .amazonaws.com` • GuardDuty VPC エンドポイントポリシーまたはサービスコントロールポリシーが guardduty:SendSecurityTelemetry アクションに影響を与えていないことを確認します。

アドオンの作成または更新エラー	トラブルシューティングのステップ
EKS アドオンの問題 - `InsufficientNumberOfReplicas` : 1 つ以上のポッドに待機コンテナ `CrashLoopBackOff: Error` があるため、アドオンが異常です	ポッドに関連付けられたログを表示し、問題を特定できます。これを行う方法については、「Kubernetes ドキュメント」の「実行中のポッドのデバッグ」を参照してください。 問題を特定したら、以下のチェックリストを使用してトラブルシューティングを行います。 • Runtime Monitoring が有効になっていることを確認します。 • 検証済み OS ディストリビューションやサポートされている Kubernetes バージョンなどの Amazon EKS クラスターサポートの前提条件 が満たされていることを確認します。 • GuardDuty セキュリティエージェントは、GuardDuty VPC エンドポイントのプライベート DNS を解決できます。エンドポイントを確認するには、「GuardDuty セキュリティエージェントの管理」の「エンドポイントのプライベート DNS 名」を参照してください。
EKS アドオンの問題 - `Admission RequestDenied` : 許可ウェブフック `"validate.kyverno.svc-fail"` がリクエストを拒否しました: リソース違反のポリシー `DaemonSet/amazon-guardduty/aws-guardduty-agent` : restrict-image-registries: `autogen-validate-registries` :...	1. Amazon EKS クラスターまたはセキュリティ管理者は、アドオンの更新をブロックしているセキュリティポリシーを確認する必要があります。 2. コントローラを無効にする (webhook) か、コントローラが Amazon EKS からのリクエストを受け付けるようにする必要があります。

アドオンの作成または更新エラー	トラブルシューティングのステップ
EKS アドオンの問題 - ConfigurationConflict ：適用しようとしたときにコンフリクトが見つかりました。コンフリクト解決モードのため、続行できません。Conflicts: DaemonSet.apps aws-guardduty-agent - .spec.template.spec.containers[name="aws-guardduty-agent"].image	アドオンを作成または更新するときは、OVERWRITE コンフリクト解決フラグを指定してください。これにより、Kubernetes API を使用して Kubernetes の関連リソースに直接加えられた変更が上書きされる可能性があります。 最初にクラスターから Amazon EKS アドオンを削除してから再インストールできます。

アドオンの作成または更新エラー	トラブルシューティングのステップ

EKS アドオンの問題 - AccessDenied: priorityclasses.scheduling.k8s.io "aws-guardduty-agent.priorityclass" is forbidden: User "eks:addon-manager" cannot patch resource "priorityclasses" in API group "scheduling.k8s.io" at the cluster scope	不足している権限を手動で eks:addon-cluster-admin ClusterRoleBinding に追加する必要があります。以下を yaml から eks:addon-cluster-admin に追加します。
	`---`
	`kind: ClusterRoleBinding`
	`apiVersion: rbac.authorization.k8s.io/v1`
	`metadata:`
	`  name: eks:addon-cluster-admin`
	`subjects:`
	`- kind: User`
	`  name: eks:addon-manager`
	`  apiGroup: rbac.authorization.k8s.io`
	`roleRef:`
	`  kind: ClusterRole`
	`  name: cluster-admin`
	`  apiGroup: rbac.authorization.k8s.io`
	`---`
	次のコマンドを使用して、この yaml を Amazon EKS クラスターに適用できるようになります。
	`kubectl apply -f eks-addon-cluster-admin.yaml`

AddonUpdationFailed: EKSAddonIssue - AccessDenied: namespaces\"amazon-guardduty\"isforbidden:User\"eks:addon-manager\"cannotpatchresource\"namespaces\"inAPIgroup\"\"inthenamespace\"amazon-guardduty\"	

アドオンの作成または更新エラー	トラブルシューティングのステップ
EKS アドオンの問題 - `AccessDenied: admission webhook "validation.gatekeeper.sh" denied the request: [all-namespace-must-have-label-owner] All namespaces must have an `owner` label`	コントローラを無効にするか、コントローラが Amazon EKS クラスターからのリクエストを受け付けるようにする必要があります。 アドオンを作成または更新する前に、GuardDuty 名前空間を作成して owner というラベルを付けることもできます。
EKS アドオンの問題 - `AccessDenied: admission webhook "validation.gatekeeper.sh" denied the request: [allowed-container-registries] container <aws-guardduty-agent> has an invalid image registry`	GuardDuty のイメージレジストリをアドミッションコントローラーの allowed-container-registries に追加します。詳細については、「Amazon ECR リポジトリホスティング GuardDuty エージェント」の「EKS v1.8.1-eks-build.2 の ECR リポジトリ」を参照してください。

CPU とメモリモニタリングの設定

Runtime Monitoring を有効にして、クラスターのカバレッジステータスが [正常] と評価されると、インサイトメトリクスを設定および表示できます。

以下のトピックは、デプロイされたエージェントが GuardDuty エージェントの CPU とメモリの制限に対してどのように動作するかを評価するのに役立ちます。

Amazon ECS クラスターでの監視設定

「Amazon CloudWatch ユーザーガイド」に記載されている以下の手順は、デプロイされたエージェントが GuardDuty エージェントの CPU とメモリの制限に対してどのように動作するかを評価するのに役立ちます。

1. 「クラスターおよびサービスレベルの Amazon ECS でメトリクスの Container Insights の設定」
2. 「Amazon ECS Container Insights メトリクス」

Amazon EKS クラスターでの監視設定

GuardDuty セキュリティエージェントがデプロイされ、クラスターのカバレッジステータスが正常であると評価したら、Container Insights メトリクスを設定して表示できます。

セキュリティエージェントのパフォーマンスを評価する

1. 「Amazon CloudWatch ユーザーガイド」の「Amazon EKS と Kubernetes での Container Insights のセットアップ」
2. 「Amazon CloudWatch ユーザーガイド」の「Amazon EKS および Kubernetes Container Insights のメトリクス」

セキュリティエージェント v1.5.0 以降でパフォーマンスを管理する

セキュリティエージェント v1.5.0 以降では、関連する GuardDuty エージェントが割り当てられた制限に達していることがインサイトで示されると、特定のパラメータを設定できます。詳細については、「EKS アドオンパラメータを設定する」を参照してください。

Runtime Monitoring で共有 VPC を使用する

GuardDuty Runtime Monitoring は、 内の同じ組織 AWS アカウント に属する の共有 Amazon Virtual Private Cloud (Amazon VPC) の使用をサポートしています AWS Organizations。共有 VPC は、以下の 2 つの方法で使用できます。

- 自動エージェント設定 (推奨) – GuardDuty がセキュリティエージェントを自動的に管理する場合、Amazon VPC エンドポイントポリシーも設定されます。このポリシーは、組織の共有 VPC 設定に基づいています。

  共有 VPC 所有者アカウントと、この VPC を共有するすべての参加アカウントで、自動エージェント設定を有効にする必要があります。
- 手動マネージドエージェント – 共有 VPC でセキュリティエージェントを手動で管理する場合は、対応するアカウントが共有 VPC にアクセスできるようにVPC エンドポイントポリシーを更新する必要があります。これを行うには、次の 仕組み セクションで共有されているポリシーの例を使用します。

  共有 VPC の参加アカウントを含む手動管理シナリオでは、カバレッジステータスが正確ではない可能性があります。リソースの up-to-date 保護とカバレッジステータスを確保するために、GuardDuty では、共有 VPC を使用するすべてのアカウントで自動エージェント設定を有効にすることをお勧めします。

トピック

- 仕組み
- 共有 VPC を使用するための前提条件

仕組み

共有 Amazon VPC 所有者アカウントと同じ組織 AWS アカウント に属する は、同じ Amazon VPC エンドポイントを共有することもできます。同じ Amazon VPC を共有する各アカウントは、関連付けられた共有 Amazon VPC の参加者 AWS アカウントと呼ばれます。

次の例は、共有 VPC 所有者アカウントと参加者アカウントのデフォルトの VPC エンドポイントポリシーを示しています。aws:PrincipalOrgID は、共有 VPC リソースに関連付けられた組織 ID を表示します。このポリシーの使用は、所有者アカウントの組織に存在する参加者アカウントに限定されます。

Example共有 VPC エンドポイントポリシーの例

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
            "Action": "*",
            "Resource": "*",
            "Effect": "Allow",
            "Principal": "*"
        },
        {
            "Condition": {
                "StringNotEquals": {
                    "aws:PrincipalOrgID": "o-abcdef0123"
                }
            },
            "Action": "*",
            "Resource": "*",
            "Effect": "Deny",
            "Principal": "*"
        }
    ]
}
```

GuardDuty 自動エージェント設定を使用する

共有 VPC の所有者アカウントがいずれかのリソース (Amazon EKS または AWS Fargate (Amazon ECS のみ)) の Runtime Monitoring および自動エージェント設定を有効にすると、すべての共有 VPCs は、共有 VPC 所有者アカウント内の共有 Amazon VPC エンドポイントおよび関連するセキュリティグループの自動インストールの対象となります。GuardDuty は、共有 Amazon VPC に関連付けられている組織 ID を取得します。

GuardDuty は、共有 VPC 所有者アカウントまたは参加アカウントのいずれかで必要な場合に Amazon VPC エンドポイントを作成します。Amazon VPC エンドポイントを必要とする例には、GuardDuty、Runtime Monitoring および EKS Runtime Monitoring の有効化、新しい Amazon ECS-Fargate タスクの起動などがあります。これらのアカウントが任意のリソースタイプの Runtime Monitoring と自動エージェント設定を有効にすると、GuardDuty は Amazon VPC エンドポイントを作成し、共有 VPC 所有者アカウントのものと同じ組織 ID でエンドポイントポリシーを設定します。GuardDuty は、GuardDuty が作成する Amazon VPC エンドポイントに `GuardDutyManaged` タグを追加し、その値を `true` に設定します。共有 Amazon VPC 所有者アカウントがいずれかのリソースの Runtime Monitoring または自動エージェント設定を有効にしていない場合、GuardDuty は Amazon VPC エンドポイントポリシーを設定しません。Runtime Monitoring の設定と、共有 VPC 所有者アカウントでのセキュリティエージェントの自動管理については、「GuardDuty Runtime Monitoring の有効化」を参照してください。

手動マネージドエージェントを使用する

手動マネージドエージェントで共有 VPC を使用する場合は、共有 VPC を使用する必要があるアカウントをブロックする明示的な `Deny` エンドポイントポリシーがないことを確認します。これにより、セキュリティエージェントがテレメトリを GuardDuty に送信できなくなり、`Unhealthy`カバレッジステータスになります。エンドポイントポリシーの設定については、「Example shared VPC endpoint policy」を参照してください。

共有 VPC に対するアクセス許可がない場合など、ランタイムカバレッジが正確ではない場合があります。ランタイムカバレッジ統計の確認と問題のトラブルシューティング のリソースタイプの手順に従って、リソースカバレッジを継続的にモニタリングできます。

コンピューティングリソースの継続的な Runtime Monitoring 保護を確実にするために、GuardDuty では、共有 VPC 所有者アカウントとリソースのすべての参加アカウントの自動エージェント設定を有効にすることをお勧めします。

共有 VPC を使用するための前提条件

初期設定の一環として、共有 VPC の所有者 AWS アカウント にする で次の手順を実行します。

1. 組織の作成 –「AWS Organizations ユーザーガイド」の「Creating and managing an organization」のステップに従って組織を作成します。

   メンバーアカウントの追加または削除の詳細については、「組織 AWS アカウント での の管理」を参照してください。

2. 共有 VPC リソースの作成 – 所有者アカウントから共有 VPC リソースを作成できます。詳細については、「Amazon VPC ユーザーガイド」の「VPC サブネットを他のアカウントと共有する」を参照してください。

GuardDuty Runtime Monitoring に固有の前提条件

次のリストは、GuardDuty に固有の前提条件を示しています。

- 共有 VPC の所有者アカウントと参加アカウントは、GuardDuty の異なる組織に所属することができます。ただし、 AWS Organizations内の同じ組織に属している必要があります。これは、GuardDuty が Amazon VPC エンドポイントと共有 VPC のセキュリティグループを作成する場合に必須です。共有 VPC の仕組みについては、「Amazon VPC ユーザーガイド」の「VPC を他のアカウントと共有する」を参照してください。
- 共有 VPC 所有者アカウントと参加者アカウントのリソースに対して、Runtime Monitoring または EKS Runtime Monitoring、および GuardDuty 自動エージェント設定を有効にします。詳細については、「Runtime Monitoring の有効化」を参照してください。

  これらの設定を既に完了している場合は、次のステップに進みます。
- Amazon EKS または Amazon ECS (AWS Fargate のみ) タスクを使用する場合は、所有者アカウントに関連付けられた共有 VPC リソースを選択し、そのサブネットを選択してください。

GuardDuty 自動セキュリティエージェントでの Infrastructure as Code (IaC) の使用

このセクションは、次のリストがユースケースに適用される場合にのみ使用します。

- AWS リソースを管理するには、 AWS Cloud Development Kit (AWS CDK) や Terraform などの Infrastructure as Code (IaC) ツールを使用します。
- Amazon EKS、Amazon EC2、または Amazon ECS-Fargate の 1 つ以上のリソースタイプに対して GuardDuty 自動エージェント設定を有効にする必要があります。

IaC リソース依存関係グラフの概要

リソースタイプの GuardDuty 自動エージェント設定を有効にすると、GuardDuty は VPC エンドポイントと、この VPC エンドポイントに関連付けられたセキュリティグループを自動的に作成し、このリソースタイプのセキュリティエージェントをインストールします。デフォルトでは、GuardDuty は、Runtime Monitoring を無効にした後にのみ VPC エンドポイントと関連するセキュリティグループを削除します。詳細については、「Runtime Monitoring でのリソースの無効化、アンインストール、クリーンアップ」を参照してください。

IaC ツールを使用すると、リソースの依存関係グラフが維持されます。IaC ツールを使用してリソースを削除すると、リソースの依存関係グラフの一部として追跡できるリソースのみが削除されます。IaC ツールは、指定された設定外で作成されたリソースを認識できない場合があります。たとえば、IaC ツールを使用して VPC を作成し、 AWS コンソールまたは API オペレーションを使用してこの VPC にセキュリティグループを追加します。リソース依存関係グラフでは、作成する VPC リソースは、関連するセキュリティグループによって異なります。IaC ツールを使用してこの VPC リソースを削除すると、エラーが発生します。このエラーを回避するには、関連付けられたセキュリティグループを手動で削除するか、この追加されたリソースを含めるように IaC 設定を更新します。

一般的な問題 - IaC でのリソースの削除

GuardDuty 自動エージェント設定を使用する場合、IaC ツールを使用して作成したリソース (Amazon EKS、Amazon EC2、または Amazon ECS-Fargate) を削除できます。ただし、このリソースは、GuardDuty が作成した VPC エンドポイントに依存します。これにより、IaC ツールがリソース自体を削除するのを防ぎ、VPC エンドポイントを自動的に削除する Runtime Monitoring を無効にする必要があります。

例えば、GuardDuty がユーザーに代わって作成した VPC エンドポイントを削除しようとすると、次の例のようなエラーが発生します。

Example

CDK を使用する場合のエラー例

```
The following resource(s) failed to delete:
 [mycdkvpcapplicationpublicsubnet1Subnet1SubnetEXAMPLE1, mycdkvpcapplicationprivatesubnet1Subne
Resource handler returned message: "The subnet 'subnet-APKAEIVFHP46CEXAMPLE' has
 dependencies and cannot be deleted. (Service: Ec2, Status Code: 400, Request
 ID: e071c3c5-7442-4489-838c-0dfc6EXAMPLE)" (RequestToken: 4381cff8-6240-208a-8357-5557b7EXAMPL
 HandlerErrorCode: InvalidRequest)
```

Example

Terraform を使用する場合のエラー例

```
module.vpc.aws_subnet.private[1]: Still destroying... [id=subnet-APKAEIVFHP46CEXAMPLE,
 19m50s elapsed]
module.vpc.aws_subnet.private[1]: Still destroying... [id=subnet-APKAEIVFHP46CEXAMPLE,
 20m0s elapsed]

Error: deleting EC2 Subnet (subnet-APKAEIBAERJR2EXAMPLE): DependencyViolation: The
 subnet 'subnet-APKAEIBAERJR2EXAMPLE' has dependencies and cannot be deleted.
       status code: 400, request id: e071c3c5-7442-4489-838c-0dfc6EXAMPLE
```

> **Note**
>
> Terraform AWS プロバイダーバージョン 6.43.0 以降では、この問題は自動的に解決されます。 AWS プロバイダーは、 中に GuardDuty が管理する VPC エンドポイントとセキュリティグループを検出して削除しますterraform destroy。この修正は、すべてのリソースタイプ (Amazon EKS、Amazon EC2、Amazon ECS-Fargate) に適用されます。

解決策 - リソース削除の問題を防ぐ

このセクションでは、GuardDuty とは独立して VPC エンドポイントとセキュリティグループを管理するのに役立ちます。

IaC ツールを使用して設定されたリソースの完全な所有権を取得するには、記載されている順序で次のステップを実行します。

1. VPC を作成します。イングレスアクセス許可を許可するには、GuardDuty VPC エンドポイントをセキュリティグループにこの VPC に関連付けます。
2. リソースタイプの GuardDuty 自動エージェント設定を有効にする

上記のステップを完了すると、GuardDuty は独自の VPC エンドポイントを作成せず、IaC ツールを使用して作成したエンドポイントを再利用します。

ユーザー独自の VPC の作成については、「Amazon VPC Transit Gateways」で「VPC のみを作成する」を参照してください。VPC エンドポイントの作成については、リソースタイプの次のセクションを参照してください。

- Amazon EC2 については、「前提条件 - Amazon VPC エンドポイントの手動作成」を参照してください。
- Amazon EKS については、「前提条件 – Amazon VPC エンドポイントの作成」を参照してください。

GuardDuty が使用する収集されたランタイムイベントタイプ

GuardDuty セキュリティエージェントは、以下のイベントタイプを収集し、脅威の検出と分析のために GuardDuty バックエンドに送信します。GuardDuty では、これらのイベントにはアクセスできません。GuardDuty が潜在的な脅威を検出し、「Runtime Monitoring の検出結果タイプ」を生成した場合は、対応する検出結果の詳細を表示できます。

GuardDuty が Runtime Monitoring で収集したイベントタイプをどのように使用するかについては、「サービス改善のためのデータ使用をオプトアウトする」を参照してください。

イベントを処理する

プロセスイベントは、Amazon EC2 インスタンスとコンテナワークロードで実行されているプロセスに関連する情報を表します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集するプロセスイベントの名前と説明を示します。

フィールド名	説明
プロセス名	監視されたプロセスの名前。
プロセスパス	プロセスの実行可能ファイルの絶対パス。
プロセス ID	オペレーティングシステムによってプロセスに割り当てられた ID。
名前空間 PID	ホストレベルの PID 名前空間以外のセカンダリ PID 名前空間内のプロセスのプロセス ID。コンテナ内のプロセスの場合、コンテナ内で確認されるプロセス ID です。
プロセスユーザー ID	プロセスを実行したユーザーの固有 ID。

フィールド名	説明
プロセス UUID。	GuardDuty によってプロセスに割り当てられた一意の ID。
プロセス GID。	プロセスグループのプロセス ID。
プロセス EGID。	プロセスグループの実効グループ ID。
プロセス EUID。	プロセスの実効ユーザー ID。
プロセスユーザー名	プロセスを実行したユーザー名。
プロセス開始時間	プロセスが作成された時間。このフィールドは、UTC 日付文字列形式 (`2023-03-22T19:37:20.168Z`)です。
プロセスの実行可能ファイル SHA-256	プロセスの実行可能ファイルの `SHA256` ハッシュ。
プロセススクリプトパス	実行されたスクリプトファイルのパス。
プロセス環境変数	プロセスで利用可能になった環境変数。`LD_PRELOAD` および `LD_LIBRARY_PATH` のみ収集可能です。
プロセス現在の作業ディレクトリ (PWD)	プロセスの現在の作業ディレクトリ。
親プロセス	親プロセスのプロセス詳細。親プロセスとは、監視対象のプロセスを作成したプロセスです。

フィールド名	説明
コマンドライン引数 現在、このフィールドはリソースタイプに対応する特定のエージェントバージョンに制限されています。 • GuardDuty セキュリティエージェント v1.0.0 以降の Fargate (Amazon ECS のみ)。 • GuardDuty セキュリティエージェント v1.0.0 以降の Amazon EC2 インスタンス。 • セキュリティエージェント v1.4.0 以降の Amazon EKS クラスター。 詳細については、「GuardDuty セキュリティエージェントのリリースバージョン」を参照してください。	プロセスの実行時に提供されるコマンドライン引数。このフィールドには機密の顧客データが含まれている可能性があります。

コンテナイベント

コンテナイベントは、コンテナワークロードのアクティビティに関連する情報を表します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集するコンテナワークロードイベントのフィールド名と説明を示します。

フィールド名	説明
コンテナ名	コンテナの名前。 使用可能な場合、このフィールドには `io.kubenetes.container.name` ラベルの値が表示されます。
コンテナ UID	コンテナランタイムによって割り当てられたコンテナの固有の ID。
コンテナランタイム	コンテナの実行に使用されたコンテナランタイム (`docker` または `containerd` など)。

フィールド名	説明
コンテナイメージ ID	コンテナのイメージの ID。
コンテナイメージ名	コンテナイメージの名前。

AWS Fargate (Amazon ECS のみ) タスクイベント

Fargate-Amazon ECS タスクイベントは、Fargate コンピューティングで実行されている Amazon ECS タスクに関連付けられたアクティビティを表します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集する Amazon ECS-Fargate タスクイベントのフィールド名と説明を示します。

フィールド名	説明
タスク Amazon リソースネーム (ARN)	タスクの ARN。
クラスター名	Amazon ECS クラスターの名前。
[Family Name] (姓)	タスク定義のファミリー名。`family` は、タスクを起動するために使用されるタスク定義の名前として使用されます。
サービス名	タスクがサービスの一部として起動された場合の Amazon ECS サービスの名前。
起動タイプ	タスクが実行されるインフラストラクチャ。`ECSCluster` のリソースタイプの Runtime Monitoring では、起動タイプは `EC2` または `FARGATE` のどちらかになります。
CPU	タスク定義に示されている、タスクで使用される CPU ユニットの数。

Kubernetes ポッドイベント

次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集する Kubernetes ポッドイベントのフィールド名と説明を示します。

フィールド名	説明
ポッド ID	Kubernetes ポッドの ID。
ポッド名	Kubernetes ポッドの名前。
ポッド名前空間	Kubernetes ワークロードが属する Kubernetes 名前空間の名前。
Kubernetes クラスター名	Kubernetes クラスターの名前。

ドメインネームシステム (DNS) イベント

ドメインネームシステム (DNS) イベントには、リソースタイプと対応するレスポンスによって行われた DNS クエリの詳細が含まれます。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集する DNS イベントの名前と説明を示します。

フィールド名	説明
ソケットタイプ	通信セマンティクスを示すソケットのタイプ。例えば、`SOCK_RAW`。
アドレスファミリー	アドレスに関連付けられた通信プロトコルを示します。例えば、 アドレスファミリー`AF_INET`は IPv4 プロトコルに使用されます。
方向 ID	接続方向の ID。
プロトコル番号	レイヤー 4 プロトコル番号。例えば UDP の場合は 17、TCP の場合は 6 です。
DNS リモートエンドポイント IP	接続のリモート IP。
DNS リモートエンドポイントポート	接続のポート番号。

フィールド名	説明
DNS ローカルエンドポイント IP	接続のローカル IP。
DNS ローカルエンドポイントポート	接続のポート番号。
DNS ペイロード	DNS クエリと応答を含む DNS パケットのペイロード。

オープンイベント

オープンイベントは、ファイルアクセスと変更に関連付けられます。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集するオープンイベントの名前と説明を示します。

フィールド名	説明
Filepath	このイベントで開かれるファイルのパス。
フラグ	読み込み専用、書き込み専用、読み込み/書き込みなどのファイルアクセスモードについて説明します。

ロードモジュールのイベント

次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集するロードモジュールイベントの名前と説明を示します。

フィールド名	説明
モジュール名	カーネルにロードされたモジュールの名前。

モプロテクトイベント

Mprotect イベントは、モニタリング対象のシステムで実行されているプロセスのメモリ保護設定の変更に関する情報を提供します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集する Mprotect イベントの名前と説明を示します。

フィールド名	説明
アドレス範囲	アクセス保護が変更されたアドレス範囲。
メモリ領域	スタックやヒープなど、プロセスのアドレス空間のリージョンを指定します。
フラグ	このイベントの動作をコントロールするオプションを示します。

マウントイベント

マウントイベントは、モニタリング対象のリソースでのファイルシステムのマウントとアンマウントに関連する情報を提供します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集するマウントイベントの名前と説明を示します。

フィールド名	説明
マウントターゲット	マウントソースがマウントされているパス。
マウントソース	マウントターゲットにマウントされているホスト上のパス。
ファイルシステムタイプ	マウントされているファイルシステムのタイプを示します。
フラグ	このイベントの動作をコントロールするオプションを示します。

リンクイベント

リンクイベントは、モニタリング対象のリソース内のファイルシステムリンク管理アクティビティを可視化します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集するリンクイベントのフィールド名と説明を示します。

フィールド名	説明
リンクパス	ハードリンクが作成されるパス。

フィールド名	説明
ターゲットパス	ハードリンクが指すファイルのパス。

Symlink イベント

Symlink イベントは、モニタリング対象のリソース内のファイルシステムのシンボリックリンク管理アクティビティを可視化します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集するシンボリックリンクイベントの名前と説明を示します。

フィールド名	説明
リンクパス	Symlink リンクが作成されるパス。
ターゲットパス	Symlink リンクが指すファイルのパス。

Dup イベント

Dup イベントは、モニタリング対象のリソースで実行されているプロセスによるファイル記述子の重複を可視化します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集する dup イベントの名前と説明を示します。

フィールド名	説明
古いファイルディスクリプタ	開いているファイルオブジェクトを表すファイル記述子。
新規ファイルディスクリプタ	古いファイル記述子の複製である新しいファイル記述子。古いファイル記述子と新しいファイル記述子はどちらも同じ開いているファイルオブジェクトを示します。
Dup リモートエンドポイント IP	古いファイル記述子で表されるネットワークソケットのリモート IP アドレス。古いファイル記述子がネットワークソケットを示す場合にのみ適用されます。
Dup リモートエンドポイントポート	古いファイル記述子で表されるネットワークソケットのリモートポート。古いファイル記述子がネットワークソケットを示す場合にのみ適用されます。

フィールド名	説明
Dup ローカルエンドポイント IP	古いファイルディスクリプタで表されるネットワークソケットのローカル IP アドレス。古いファイル記述子がネットワークソケットを示す場合にのみ適用されます。
Dup ローカルエンドポイントポート	古いファイル記述子で表されるネットワークソケットのローカルポート。古いファイル記述子がネットワークソケットを示す場合にのみ適用されます。

メモリマッピングイベント

次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集するメモリマップイベントの名前と説明を示します。

フィールド名	説明
Filepath	メモリがマッピングされているファイルのパス。

ソケットイベント

ソケットイベントは、モニタリング対象のリソースのアクティビティで使用されるネットワークソケット接続に関する情報を提供します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集するソケットイベントのフィールド名と説明を示します。

フィールド名	説明
アドレスファミリー	アドレスに関連付けられた通信プロトコルを示します。例えば、 アドレスファミリー`AF_INET`は IPv4 プロトコルに使用されます。
ソケットタイプ	通信セマンティクスを示すソケットのタイプ。例えば、`SOCK_RAW`。
プロトコル番号	アドレスファミリー内の特定のプロトコルを指定します。通常、アドレスファミリーには単一のプロトコルがあります。例

フィールド名	説明
	えば、アドレスファミリー AF_INET には IP プロトコルしかありません。

イベントを接続

接続イベントは、モニタリング対象のリソース上のプロセスによって確立されたネットワーク接続を可視化します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集する接続イベントの名前と説明を示します。

フィールド名	説明
アドレスファミリー	アドレスに関連付けられた通信プロトコルを示します。例えば、 アドレスファミリーAF_INETは IPv4 プロトコルに使用されます。
ソケットタイプ	通信セマンティクスを示すソケットのタイプ。例えば、SOCK_RAW。
プロトコル番号	アドレスファミリー内の特定のプロトコルを指定します。通常、アドレスファミリーには単一のプロトコルがあります。例えば、アドレスファミリー AF_INET には IP プロトコルしかありません。
Filepath	アドレスファミリーが AF_UNIX の場合のソケットファイルのパス。
リモートエンドポイント IP	接続のリモート IP。
リモートエンドポイントポート	接続のポート番号。
ローカルエンドポイント IP	接続のローカル IP。
ローカルエンドポイントポート	接続のポート番号。

VM Readv イベントの処理

プロセス VM readv イベントは、プロセスによって独自の仮想メモリ領域で実行される読み取り操作を可視化します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集するプロセス VM readv イベントのフィールド名と説明を示します。

フィールド名	説明
フラグ	このイベントの動作をコントロールするオプションを示します。
ターゲット PID	メモリを読み込んでいるプロセスのプロセス ID。
ターゲットプロセスの UUID	ターゲットプロセスの一意の ID。
ターゲットの実行可能ファイルのパス	ターゲットプロセスの実行可能ファイルの絶対パス。

VM Writev イベントの処理

プロセス VM writev イベントは、プロセスによって独自の仮想メモリ領域で実行される書き込み操作を可視化します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集するプロセス VM writev イベントのフィールド名と説明を示します。

フィールド名	説明
フラグ	このイベントの動作をコントロールするオプションを示します。
ターゲット PID	メモリが書き込まれているプロセスのプロセス ID。
ターゲットプロセスの UUID	ターゲットプロセスの一意の ID。
ターゲットの実行可能ファイルのパス	ターゲットプロセスの実行可能ファイルの絶対パス。

プロセストレース (Ptrace) イベント

プロセストレース (Ptrace) システムコールは、あるプロセス (トレーサー) が別のプロセス (トレース) の実行を監視および制御できるようにするデバッグおよびトレースメカニズムです。これにより、トレーサーはターゲットプロセスのメモリ、レジスタ、実行フローを検査および変更できます。

Ptrace イベントは、モニタリング対象のリソースで実行されているプロセスによる ptrace システムコールの使用を可視化します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集する ptrace イベントのフィールド名と説明を示します。

フィールド名	説明
ターゲット PID	ターゲットプロセスのプロセス ID。
ターゲットプロセスの UUID	ターゲットプロセスの一意の ID。
ターゲットの実行可能ファイルのパス	ターゲットプロセスの実行可能ファイルの絶対パス。
フラグ	このイベントの動作をコントロールするオプションを示します。

バインドイベント

バインドイベントは、モニタリング対象のリソースで実行されているプロセスによるネットワークソケットのバインドを可視化します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集するバインドイベントの名前と説明を示します。

フィールド名	説明
アドレスファミリー	アドレスに関連付けられた通信プロトコルを示します。例えば、 アドレスファミリー`AF_INET`は IPv4 プロトコルに使用されます。
ソケットタイプ	通信セマンティクスを示すソケットのタイプ。例えば、`SOCK_RAW`。

フィールド名	説明
プロトコル番号	レイヤー 4 プロトコル番号。例えば UDP の場合は 17、TCP の場合は 6 です。
ローカルエンドポイント IP	接続のローカル IP。
ローカルエンドポイントポート	接続のポート番号。

リッスンイベント

リッスンイベントは、ネットワークソケットのリッスン状態を可視化し、ネットワークソケットが受信接続を受け入れる準備ができているかどうかを示します。モニタリング対象のリソースで実行されるプロセスは、ネットワークソケットをリッスン状態に設定します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集するリスンイベントのフィールド名と説明を示します。

フィールド名	説明
アドレスファミリー	アドレスに関連付けられた通信プロトコルを示します。例えば、 アドレスファミリー`AF_INET`は IPv4 プロトコルに使用されます。
ソケットタイプ	通信セマンティクスを示すソケットのタイプ。例えば、`SOCK_RAW`。
プロトコル番号	レイヤー 4 プロトコル番号。例えば UDP の場合は 17、TCP の場合は 6 です。
ローカルエンドポイント IP	接続のローカル IP。
ローカルエンドポイントポート	接続のポート番号。

名前変更イベント

名前変更イベントは、モニタリング対象のリソースで実行されているプロセスによるファイルとディレクトリの名前変更に関する情報を提供します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集する名前変更イベントの名前と説明を示します。

フィールド名	説明
Filepath	名前が変更されたファイルのパス。
ターゲット	ファイルの新しいパス。

ユーザー ID (UID) 設定イベント

ユーザー ID (UID) 設定イベントは、モニタリング対象のリソースで実行中のプロセスに関連付けられたユーザー ID (UID) に加えられた変更を可視化します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集する UID 設定イベントの名前と説明を示します。

フィールド名	説明
新しい EUID	プロセスの新しい実効ユーザー ID。
新しい UID	プロセスの新しいユーザー ID。

Chmod イベント

Chmod イベントは、モニタリング対象のリソース上のファイルとディレクトリのアクセス許可 (モード) の変更を可視化します。次の表に、Runtime Monitoring が潜在的な脅威を検出するために収集する chmod イベントの名前と説明を示します。

フィールド名	説明
Filepath	このイベントを呼び出すファイルのパス。
Filemode	関連付けられたファイルの更新されたアクセス許可。

Amazon ECR リポジトリホスティング GuardDuty エージェント

以下のセクションでは、GuardDuty が Amazon EKS、Amazon ECS、Amazon ECS-Amazon EC2 Bottlerocket リソースにデプロイされるセキュリティエージェントをホストする Amazon Elastic Container Registry (Amazon ECR) リポジトリを一覧表示します。

コンテナイメージアクセスの前提条件 の前提条件では、特定の Amazon Elastic Container Registry (Amazon ECR) アクセス許可を持つタスク実行ロールを指定する必要があります。これらのアクセス許可をさらに制限するには、Fargate-Amazon ECS リソースの GuardDuty エージェントをホストする Amazon ECR リポジトリ URI を追加できます。

内容

- EKS エージェントバージョン 1.16.0 ~ 1.8.1 (eks.build.2) の ECR リポジトリ
- EKS エージェントバージョン 1.8.1 (eks.build.1) のリポジトリ
- 上の GuardDuty エージェント用の ECR リポジトリ AWS Fargate (Amazon ECS のみ)
- ECS-EC2 Bottlerocket 上の GuardDuty エージェント用の ECR リポジトリ

EKS エージェントバージョン 1.16.0 ~ 1.8.1 (eks.build.2) の ECR リポジトリ

Runtime Monitoring for EKS で GuardDuty 自動設定を有効にすると、GuardDuty はこのエージェントバージョンを Amazon EKS クラスターにデプロイします。自動エージェントを有効にする方法については、「Amazon EKS リソースのセキュリティエージェントの自動管理」を参照してください。

次の表は、Amazon EKS `1.8.0.eks.build.2`の GuardDuty セキュリティエージェントバージョン `1.16.0.eks.build.2`、`1.15.0.eks.build.2`、、`1.10.0.eks.build.2`、`1.9.0.eks.build.2`、および `1.11.0.eks.build.2`がホストされている Amazon ECR リポジトリ URIs を示しています。

AWS リージョン	Amazon ECR リポジトリ URI
米国西部 (オレゴン)	`602401143452.dkr.ecr.us-west-2.amazonaws.com`

AWS リージョン	Amazon ECR リポジトリ URI
	039403964562.dkr.ecr.us-west-2.amazonaws.com
欧州 (パリ)	602401143452.dkr.ecr.eu-west-3.amazonaws.com
	113643092156.dkr.ecr.eu-west-3.amazonaws.com
アジアパシフィック (ムンバイ)	602401143452.dkr.ecr.ap-south-1.amazonaws.com
	610108029387.dkr.ecr.ap-south-1.amazonaws.com
アジアパシフィック (ハイデラバード)	900889452093.dkr.ecr.ap-south-2.amazonaws.com
	618745550137.dkr.ecr.ap-south-2.amazonaws.com
カナダ (中部)	602401143452.dkr.ecr.ca-central-1.amazonaws.com
	001188825231.dkr.ecr.ca-central-1.amazonaws.com
カナダ西部 (カルガリー)	761377655185.dkr.ecr.ca-west-1.amazonaws.com
	–
中東 (アラブ首長国連邦)	759879836304.dkr.ecr.me-central-1.amazonaws.com
	601769779514.dkr.ecr.me-central-1.amazonaws.com

AWS リージョン	Amazon ECR リポジトリ URI
欧州 (ロンドン)	602401143452.dkr.ecr.eu-west-2.amazonaws.com
	109118265657.dkr.ecr.eu-west-2.amazonaws.com
米国西部 (北カリフォルニア)	602401143452.dkr.ecr.us-west-1.amazonaws.com
	373421517865.dkr.ecr.us-west-1.amazonaws.com
米国東部 (バージニア北部)	602401143452.dkr.ecr.us-east-1.amazonaws.com
	031903291036.dkr.ecr.us-east-1.amazonaws.com
米国東部 (オハイオ)	602401143452.dkr.ecr.us-east-2.amazonaws.com
	591382732059.dkr.ecr.us-east-2.amazonaws.com
欧州 (アイルランド)	602401143452.dkr.ecr.eu-west-1.amazonaws.com
	673884943994.dkr.ecr.eu-west-1.amazonaws.com
南米 (サンパウロ)	602401143452.dkr.ecr.sa-east-1.amazonaws.com
	941219317354.dkr.ecr.sa-east-1.amazonaws.com
欧州 (ストックホルム)	602401143452.dkr.ecr.eu-north-1.amazonaws.com

AWS リージョン	Amazon ECR リポジトリ URI
	366771026645.dkr.ecr.eu-north-1.amazonaws.com
欧州 (フランクフルト)	602401143452.dkr.ecr.eu-central-1.amazonaws.com
	409493279830.dkr.ecr.eu-central-1.amazonaws.com
欧州 (チューリッヒ)	900612956339.dkr.ecr.eu-central-2.amazonaws.com
	718440343717.dkr.ecr.eu-central-2.amazonaws.com
アジアパシフィック (シンガポール)	602401143452.dkr.ecr.ap-southeast-1.amazonaws.com
	584580519942.dkr.ecr.ap-southeast-1.amazonaws.com
アジアパシフィック (シドニー)	602401143452.dkr.ecr.ap-southeast-2.amazonaws.com
	011662287384.dkr.ecr.ap-southeast-2.amazonaws.com
アジアパシフィック (ジャカルタ)	296578399912.dkr.ecr.ap-southeast-3.amazonaws.com
	617474730032.dkr.ecr.ap-southeast-3.amazonaws.com
アジアパシフィック (東京)	602401143452.dkr.ecr.ap-northeast-1.amazonaws.com
	781592569369.dkr.ecr.ap-northeast-1.amazonaws.com

AWS リージョン	Amazon ECR リポジトリ URI
アジアパシフィック (ソウル)	602401143452.dkr.ecr.ap-northeast-2.amazonaws.com
	732248494576.dkr.ecr.ap-northeast-2.amazonaws.com
アジアパシフィック (大阪)	602401143452.dkr.ecr.ap-northeast-3.amazonaws.com
	810724417379.dkr.ecr.ap-northeast-3.amazonaws.com
アジアパシフィック (香港)	800184023465.dkr.ecr.ap-east-1.amazonaws.com
	790429075973.dkr.ecr.ap-east-1.amazonaws.com
中東 (バーレーン)	558608220178.dkr.ecr.me-south-1.amazonaws.com
	541829937850.dkr.ecr.me-south-1.amazonaws.com
欧州 (ミラノ)	590381155156.dkr.ecr.eu-south-1.amazonaws.com
	528450769569.dkr.ecr.eu-south-1.amazonaws.com
欧州 (スペイン)	455263428931.dkr.ecr.eu-south-2.amazonaws.com
	531047660167.dkr.ecr.eu-south-2.amazonaws.com
アフリカ (ケープタウン)	877085696533.dkr.ecr.af-south-1.amazonaws.com

AWS リージョン	Amazon ECR リポジトリ URI
	379032919888.dkr.ecr.af-south-1.amazonaws.com
アジアパシフィック (メルボルン)	491585149902.dkr.ecr.ap-southeast-4.amazonaws.com
	750462861327.dkr.ecr.ap-southeast-4.amazonaws.com
イスラエル (テルアビブ)	066635153087.dkr.ecr.il-central-1.amazonaws.com
	292660727137.dkr.ecr.il-central-1.amazonaws.com
アジアパシフィック (マレーシア)	151610086707.dkr.ecr.ap-southeast-5.amazonaws.com
アジアパシフィック (タイ)	121268973566.dkr.ecr.ap-southeast-7.amazonaws.com
メキシコ (中部)	730335286997.dkr.ecr.mx-central-1.amazonaws.com
アジアパシフィック (台北)	533267051163.dkr.ecr.ap-east-2.amazonaws.com

EKS エージェントバージョン 1.8.1 (**eks.build.1**) のリポジトリ

このセクションでは、Amazon EKS エージェントバージョン 1.8.1 (v1.8.1-eks-build.1) の Amazon ECR リポジトリについて説明します。v1.8.1-eks-build.1 を使用している場合、GuardDuty は通常最新のエージェントバージョンであるデフォルトのエージェントバージョンへの切り替えを推奨します。そのためには、Amazon EKS リソース用にリリースされたエージェントバージョン から最新のエージェントを特定し、Amazon EKS リソースのセキュリティエージェントの手動更新 の手順を実行します。

次の表は、Amazon EKS の GuardDuty セキュリティエージェントバージョン 1.8.1-eks-build.1 がホストされている Amazon ECR リポジトリ URI を示しています。

AWS リージョン	Amazon ECR リポジトリ URI
米国西部 (オレゴン)	039403964562.dkr.ecr.us-west-2.amazonaws.com
欧州 (パリ)	113643092156.dkr.ecr.eu-west-3.amazonaws.com
アジアパシフィック (ムンバイ)	610108029387.dkr.ecr.ap-south-1.amazonaws.com
アジアパシフィック (ハイデラバード)	618745550137.dkr.ecr.ap-south-2.amazonaws.com
カナダ (中部)	001188825231.dkr.ecr.ca-central-1.amazonaws.com
中東 (アラブ首長国連邦)	601769779514.dkr.ecr.me-central-1.amazonaws.com
欧州 (ロンドン)	109118265657.dkr.ecr.eu-west-2.amazonaws.com
米国西部 (北カリフォルニア)	373421517865.dkr.ecr.us-west-1.amazonaws.com
米国東部 (バージニア北部)	031903291036.dkr.ecr.us-east-1.amazonaws.com
米国東部 (オハイオ)	591382732059.dkr.ecr.us-east-2.amazonaws.com
欧州 (アイルランド)	673884943994.dkr.ecr.eu-west-1.amazonaws.com

AWS リージョン	Amazon ECR リポジトリ URI
南米 (サンパウロ)	941219317354.dkr.ecr.sa-east-1.amazonaws.com
欧州 (ストックホルム)	366771026645.dkr.ecr.eu-north-1.amazonaws.com
欧州 (フランクフルト)	409493279830.dkr.ecr.eu-central-1.amazonaws.com
欧州 (チューリッヒ)	718440343717.dkr.ecr.eu-central-2.amazonaws.com
アジアパシフィック (シンガポール)	584580519942.dkr.ecr.ap-southeast-1.amazonaws.com
アジアパシフィック (シドニー)	011662287384.dkr.ecr.ap-southeast-2.amazonaws.com
アジアパシフィック (ジャカルタ)	617474730032.dkr.ecr.ap-southeast-3.amazonaws.com
アジアパシフィック (東京)	781592569369.dkr.ecr.ap-northeast-1.amazonaws.com
アジアパシフィック (ソウル)	732248494576.dkr.ecr.ap-northeast-2.amazonaws.com
アジアパシフィック (大阪)	810724417379.dkr.ecr.ap-northeast-3.amazonaws.com
アジアパシフィック (香港)	790429075973.dkr.ecr.ap-east-1.amazonaws.com
中東 (バーレーン)	541829937850.dkr.ecr.me-south-1.amazonaws.com
欧州 (ミラノ)	528450769569.dkr.ecr.eu-south-1.amazonaws.com

AWS リージョン	Amazon ECR リポジトリ URI
欧州 (スペイン)	531047660167.dkr.ecr.eu-south-2.amazonaws.com
アフリカ (ケープタウン)	379032919888.dkr.ecr.af-south-1.amazonaws.com
アジアパシフィック (メルボルン)	750462861327.dkr.ecr.ap-southeast-4.amazonaws.com
イスラエル (テルアビブ)	292660727137.dkr.ecr.il-central-1.amazonaws.com

上の GuardDuty エージェント用の ECR リポジトリ AWS Fargate (Amazon ECS のみ)

Amazon ECS-Fargate の Runtime Monitoring を使用するための前提条件として、コンテナイメージアクセスの前提条件 が必要です。GuardDuty エージェントのサイドカーコンテナイメージは Amazon ECR に保存され、そのイメージレイヤーは Amazon S3 に保存されます。詳細については、「Fargate での Amazon ECS へのセキュリティエージェントのデプロイ」を参照してください。

次の表は、それぞれ AWS Fargate (Amazon ECS のみ) の GuardDuty エージェントをホストする Amazon ECR リポジトリを示しています AWS リージョン。

AWS リージョン	Amazon ECR リポジトリ URI
米国西部 (オレゴン)	733349766148.dkr.ecr.us-west-2.amazonaws.com/aws-guardduty-agent-fargate
欧州 (パリ)	665651866788.dkr.ecr.eu-west-3.amazonaws.com/aws-guardduty-agent-fargate

AWS リージョン	Amazon ECR リポジトリ URI
アジアパシフィック (ムンバイ)	251508486986.dkr.ecr.ap-south-1.amazonaws.com/aws-guardduty-agent-fargate
アジアパシフィック (ハイデラバード)	950823858135.dkr.ecr.ap-south-2.amazonaws.com/aws-guardduty-agent-fargate
カナダ (中部)	354763396469.dkr.ecr.ca-central-1.amazonaws.com/aws-guardduty-agent-fargate
中東 (アラブ首長国連邦)	000014521398.dkr.ecr.me-central-1.amazonaws.com/aws-guardduty-agent-fargate
欧州 (ロンドン)	892757235363.dkr.ecr.eu-west-2.amazonaws.com/aws-guardduty-agent-fargate
米国西部 (北カリフォルニア)	684579721401.dkr.ecr.us-west-1.amazonaws.com/aws-guardduty-agent-fargate
米国東部 (バージニア北部)	593207742271.dkr.ecr.us-east-1.amazonaws.com/aws-guardduty-agent-fargate
米国東部 (オハイオ)	307168627858.dkr.ecr.us-east-2.amazonaws.com/aws-guardduty-agent-fargate
欧州 (アイルランド)	694911143906.dkr.ecr.eu-west-1.amazonaws.com/aws-guardduty-agent-fargate

AWS リージョン	Amazon ECR リポジトリ URI
南米 (サンパウロ)	758426053663.dkr.ecr.sa-east-1.amazonaws.com/aws-guardduty-agent-fargate
欧州 (ストックホルム)	591436053604.dkr.ecr.eu-north-1.amazonaws.com/aws-guardduty-agent-fargate
欧州 (フランクフルト)	323658145986.dkr.ecr.eu-central-1.amazonaws.com/aws-guardduty-agent-fargate
欧州 (チューリッヒ)	529164026651.dkr.ecr.eu-central-2.amazonaws.com/aws-guardduty-agent-fargate
アジアパシフィック (シンガポール)	174946120834.dkr.ecr.ap-southeast-1.amazonaws.com/aws-guardduty-agent-fargate
アジアパシフィック (シドニー)	005257825471.dkr.ecr.ap-southeast-2.amazonaws.com/aws-guardduty-agent-fargate
アジアパシフィック (ジャカルタ)	510637619217.dkr.ecr.ap-southeast-3.amazonaws.com/aws-guardduty-agent-fargate
アジアパシフィック (東京)	533107202818.dkr.ecr.ap-northeast-1.amazonaws.com/aws-guardduty-agent-fargate
アジアパシフィック (ソウル)	914738172881.dkr.ecr.ap-northeast-2.amazonaws.com/aws-guardduty-agent-fargate

AWS リージョン	Amazon ECR リポジトリ URI
アジアパシフィック (大阪)	273192626886.dkr.ecr.ap-northeast-3.amazonaws.com/aws-guardduty-agent-fargate
アジアパシフィック (香港)	258348409381.dkr.ecr.ap-east-1.amazonaws.com/aws-guardduty-agent-fargate
中東 (バーレーン)	536382113932.dkr.ecr.me-south-1.amazonaws.com/aws-guardduty-agent-fargate
欧州 (ミラノ)	266869475730.dkr.ecr.eu-south-1.amazonaws.com/aws-guardduty-agent-fargate
欧州 (スペイン)	919611009337.dkr.ecr.eu-south-2.amazonaws.com/aws-guardduty-agent-fargate
アフリカ (ケープタウン)	197869348890.dkr.ecr.af-south-1.amazonaws.com/aws-guardduty-agent-fargate
アジアパシフィック (メルボルン)	251357961535.dkr.ecr.ap-southeast-4.amazonaws.com/aws-guardduty-agent-fargate
イスラエル (テルアビブ)	870907303882.dkr.ecr.il-central-1.amazonaws.com/aws-guardduty-agent-fargate
アジアパシフィック (マレーシア)	156041399949.dkr.ecr.ap-southeast-5.amazonaws.com/aws-guardduty-agent-fargate

AWS リージョン	Amazon ECR リポジトリ URI
アジアパシフィック (タイ)	054037130133.dkr.ecr.ap-southeast-7.amazonaws.com/aws-guardduty-agent-fargate
カナダ西部 (カルガリー)	339712888787.dkr.ecr.ca-west-1.amazonaws.com/aws-guardduty-agent-fargate
メキシコ (中部)	311141559934.dkr.ecr.mx-central-1.amazonaws.com/aws-guardduty-agent-fargate
アジアパシフィック (台北)	259886477082.dkr.ecr.ap-east-2.amazonaws.com/aws-guardduty-agent-fargate

ECS-EC2 Bottlerocket 上の GuardDuty エージェント用の ECR リポジトリ

Bottlerocket Amazon ECS-Amazon EC2 インスタンスでは、GuardDuty セキュリティエージェントはホストコンテナとして実行されます。エージェントコンテナイメージは Amazon ECR でホストされます。ネットワーク接続や必要な IAM アクセス許可などの前提条件については、「」を参照してくださいECS-EC2 Bottlerocket サポートの前提条件。インスタンスプロファイルには、`AmazonSSMManagedInstanceCore`および `AmazonEC2ContainerRegistryReadOnly`管理ポリシーが含まれている必要があります。

次の表は、それぞれの Amazon ECS-Amazon EC2 Bottlerocket インスタンスの GuardDuty エージェントをホストする Amazon ECR リポジトリを示しています AWS リージョン。

AWS リージョン	Amazon ECR リポジトリ URI
米国東部 (バージニア北部)	593207742271.dkr.ecr.us-east-1.amazonaws.com/aws-guardduty-agent-ecs-ec2

AWS リージョン	Amazon ECR リポジトリ URI
米国東部 (オハイオ)	307168627858.dkr.ecr.us-east-2.amazonaws.com/aws-guardduty-agent-ecs-ec2
米国西部 (北カリフォルニア)	684579721401.dkr.ecr.us-west-1.amazonaws.com/aws-guardduty-agent-ecs-ec2
米国西部 (オレゴン)	733349766148.dkr.ecr.us-west-2.amazonaws.com/aws-guardduty-agent-ecs-ec2
欧州 (アイルランド)	694911143906.dkr.ecr.eu-west-1.amazonaws.com/aws-guardduty-agent-ecs-ec2
欧州 (ロンドン)	892757235363.dkr.ecr.eu-west-2.amazonaws.com/aws-guardduty-agent-ecs-ec2
欧州 (パリ)	665651866788.dkr.ecr.eu-west-3.amazonaws.com/aws-guardduty-agent-ecs-ec2
欧州 (フランクフルト)	323658145986.dkr.ecr.eu-central-1.amazonaws.com/aws-guardduty-agent-ecs-ec2
欧州 (チューリッヒ)	529164026651.dkr.ecr.eu-central-2.amazonaws.com/aws-guardduty-agent-ecs-ec2
欧州 (ストックホルム)	591436053604.dkr.ecr.eu-north-1.amazonaws.com/aws-guardduty-agent-ecs-ec2

AWS リージョン	Amazon ECR リポジトリ URI
欧州 (ミラノ)	266869475730.dkr.ecr.eu-south-1.amazonaws.com/aws-guardduty-agent-ecs-ec2
欧州 (スペイン)	919611009337.dkr.ecr.eu-south-2.amazonaws.com/aws-guardduty-agent-ecs-ec2
アジアパシフィック (東京)	533107202818.dkr.ecr.ap-northeast-1.amazonaws.com/aws-guardduty-agent-ecs-ec2
アジアパシフィック (ソウル)	914738172881.dkr.ecr.ap-northeast-2.amazonaws.com/aws-guardduty-agent-ecs-ec2
アジアパシフィック (大阪)	273192626886.dkr.ecr.ap-northeast-3.amazonaws.com/aws-guardduty-agent-ecs-ec2
アジアパシフィック (シンガポール)	174946120834.dkr.ecr.ap-southeast-1.amazonaws.com/aws-guardduty-agent-ecs-ec2
アジアパシフィック (シドニー)	005257825471.dkr.ecr.ap-southeast-2.amazonaws.com/aws-guardduty-agent-ecs-ec2
アジアパシフィック (ジャカルタ)	510637619217.dkr.ecr.ap-southeast-3.amazonaws.com/aws-guardduty-agent-ecs-ec2
アジアパシフィック (メルボルン)	251357961535.dkr.ecr.ap-southeast-4.amazonaws.com/aws-guardduty-agent-ecs-ec2

AWS リージョン	Amazon ECR リポジトリ URI
アジアパシフィック (ムンバイ)	251508486986.dkr.ecr.ap-south-1.amazonaws.com/aws-guardduty-agent-ecs-ec2
アジアパシフィック (ハイデラバード)	950823858135.dkr.ecr.ap-south-2.amazonaws.com/aws-guardduty-agent-ecs-ec2
アジアパシフィック (香港)	258348409381.dkr.ecr.ap-east-1.amazonaws.com/aws-guardduty-agent-ecs-ec2
カナダ (中部)	354763396469.dkr.ecr.ca-central-1.amazonaws.com/aws-guardduty-agent-ecs-ec2
カナダ西部 (カルガリー)	339712888787.dkr.ecr.ca-west-1.amazonaws.com/aws-guardduty-agent-ecs-ec2
南米 (サンパウロ)	758426053663.dkr.ecr.sa-east-1.amazonaws.com/aws-guardduty-agent-ecs-ec2
中東 (バーレーン)	536382113932.dkr.ecr.me-south-1.amazonaws.com/aws-guardduty-agent-ecs-ec2
中東 (アラブ首長国連邦)	000014521398.dkr.ecr.me-central-1.amazonaws.com/aws-guardduty-agent-ecs-ec2
アフリカ (ケープタウン)	197869348890.dkr.ecr.af-south-1.amazonaws.com/aws-guardduty-agent-ecs-ec2

AWS リージョン	Amazon ECR リポジトリ URI
イスラエル (テルアビブ)	870907303882.dkr.ecr.il-central-1.amazonaws.com/aws-guardduty-agent-ecs-ec2
アジアパシフィック (マレーシア)	156041399949.dkr.ecr.ap-southeast-5.amazonaws.com/aws-guardduty-agent-ecs-ec2
アジアパシフィック (タイ)	054037130133.dkr.ecr.ap-southeast-7.amazonaws.com/aws-guardduty-agent-ecs-ec2
メキシコ (中部)	311141559934.dkr.ecr.mx-central-1.amazonaws.com/aws-guardduty-agent-ecs-ec2
アジアパシフィック (台北)	259886477082.dkr.ecr.ap-east-2.amazonaws.com/aws-guardduty-agent-ecs-ec2

同じ基盤となるホスト上の 2 つのセキュリティエージェント

Amazon EC2 インスタンスは、複数のタイプのワークロードをサポートできます。Amazon EC2 インスタンスで自動セキュリティエージェントを設定すると、同じ EC2 インスタンスに EKS 経由で別のセキュリティエージェントがある可能性があります。

概要

Runtime Monitoring を有効にしたシナリオを考えてみましょう。次に、GuardDuty を使用して Amazon EKS の自動エージェントを有効にします。また、Amazon EC2 の自動エージェントも有効にしました。同じ基盤となるホストに、Amazon EKS 用と Amazon EC2 用の 2 つのセキュリティエージェントがインストールされる場合があります。これにより、同じホスト内で 2 つのセキュリティエージェントが実行され、ランタイムイベントが収集および GuardDuty に送信され、重複する検出結果が生成される可能性があります。

Impact

- 同じホストで複数のセキュリティエージェントが実行されている場合、アカウントで CPU とメモリの処理ニーズが 2 倍になる可能性があります。各リソースタイプの CPU とメモリの制限については、そのリソースの「前提条件」を参照してください。
- GuardDuty は、同じ基盤となるホストからランタイムイベントを収集する 2 つのセキュリティエージェントが重複しても、アカウントがランタイムイベントの 1 つのストリームに対してのみ課金されるように Runtime Monitoring 機能を設計しました。

GuardDuty が複数のエージェントを処理する方法

GuardDuty は、2 つのセキュリティエージェントが同じホストで実行されているかどうかを検出し、そのうちの 1 つだけをランタイムイベントをアクティブに収集するセキュリティエージェントに指定します。2 番目のエージェントは、アプリケーションのパフォーマンスへの影響を防ぐために、最小限のシステムリソースを消費します。

GuardDuty では、次のシナリオを考慮します。

- EC2 インスタンスが Amazon EKS と Amazon EC2 セキュリティエージェントの両方のスコープに含まれる場合、EKS セキュリティエージェントが優先されます。これは、Amazon EC2 でセキュリティエージェント v1.1.0 以降を使用する場合にのみ適用されます。古いエージェントバージョンは優先順位付けの影響を受けないため、古いエージェントバージョンは引き続き実行され、ランタイムイベントを収集します。
- Amazon EKS と Amazon EC2 の両方に GuardDuty マネージドセキュリティエージェントがあり、Amazon EC2 インスタンスも SSM マネージドである場合、両方のセキュリティエージェントがホストレベルでインストールされます。エージェントがインストールされると、GuardDuty はどちらのセキュリティエージェントが引き続き実行されるかを決定します。両方のセキュリティエージェントが実行されている場合、最終的にはそのうちの 1 つだけがランタイムイベントを収集します。
- EC2 と EKS の両方に関連付けられたセキュリティエージェントが同時に実行されると、GuardDuty は重複期間にのみ重複する検出結果を生成する可能性があります。

  これは、次の場合に発生する可能性があります。

  - EC2 と EKS の両方のセキュリティエージェントが GuardDuty を通じて (自動的に) 設定されている。または
  - Amazon EKS リソースに自動セキュリティエージェントがある。

- EKS セキュリティエージェントが既に実行されている場合、EC2 セキュリティエージェントを同じ基盤となるホストに手動でデプロイし、すべての前提条件を満たすと、GuardDuty は 2 番目のセキュリティエージェントをインストールしない可能性があります。

GuardDuty での EKS Runtime Monitoring

EKS Runtime Monitoring は、 AWS 環境内の Amazon Elastic Kubernetes Service (Amazon EKS) ノードとコンテナのランタイム脅威検出カバレッジを提供します。EKS Runtime Monitoring では、GuardDuty セキュリティエージェントを使用して、ファイルアクセス、プロセス実行、ネットワーク接続などの個々の EKS ワークロードをランタイムで可視化します。GuardDuty セキュリティエージェントは、GuardDuty が EKS クラスター内の危険にさらされている可能性のある特定のコンテナを識別するのに役立ちます。また、個々のコンテナから基盤となる EC2 ホスト、およびより広範な AWS 環境に権限をエスカレートしようとする試みを検出することもできます。

Runtime Monitoring が利用可能になったことで、GuardDuty は EKS Runtime Monitoring のコンソールエクスペリエンスを Runtime Monitoring に統合しました。GuardDuty は、ユーザーに代わって EKS Runtime Monitoring 設定を自動的に移行しません。これには、ユーザーによるアクションが必要です。EKS Runtime Monitoring のみを引き続き使用する場合は、 APIsまたは AWS CLI を使用して EKS Runtime Monitoring の既存の設定ステータスを確認および更新できます。ただし、GuardDuty では、「EKS Runtime Monitoring から Runtime Monitoring への移行」を行い、Amazon EKS クラスターをモニタリングするために Runtime Monitoring を使用することをお勧めします。

トピック

- マルチアカウント環境の EKS Runtime Monitoring の設定 (API)
- スタンドアロンアカウントの EKS Runtime Monitoring の設定 (API)
- EKS Runtime Monitoring から Runtime Monitoring への移行

マルチアカウント環境の EKS Runtime Monitoring の設定 (API)

マルチアカウント環境では、委任 GuardDuty 管理者アカウントのみが、メンバーアカウントの EKS Runtime Monitoring を有効または無効にすることや、組織内のメンバーアカウントに属する EKS クラスターの GuardDuty エージェント管理を管理することができます。GuardDuty メンバーアカウントからは、この設定を変更できません。委任 GuardDuty 管理者アカウントは、 を使用してメンバーアカウントを管理します AWS Organizations。マルチアカウント環境の詳細については、「複数のアカウントの管理」を参照してください。

委任 GuardDuty 管理者アカウントでの EKS Runtime Monitoring の設定

このセクションでは、EKS Runtime Monitoring を設定し、委任 GuardDuty 管理者アカウントに属する EKS クラスターの GuardDuty セキュリティエージェントを管理するステップを説明します。

のアプローチに基づいてセキュリティエージェントの Amazon EKS へのデプロイ、任意のアプローチを選択し、次の表で説明されているステップに従います。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
GuardDuty によるセキュリティエージェントの管理 (すべての EKS クラスターのモニタリング)	ユーザー独自のリージョンレベルのディテクター ID を使用し、`features` オブジェクト名を `EKS_RUNTIME_MONITORING` として、ステータスを `ENABLED` として渡して、updateDetector API を実行します。 `EKS_ADDON_MANAGEMENT` のステータスを `ENABLED` に設定します。 GuardDuty は、アカウント内のすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、`EKS_RUNTIME_MONITORING` と `EKS_ADDON_MANAGEMENT` の両方を有効にします。 `aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : "ENABLED", "AdditionalConfiguration" :` `aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : "ENABLED", "AdditionalConfiguration" :`

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	[{"Name" : "EKS_ADDON_MANAGEMENT", "Status" : "*ENABLED*"}] }]'

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
一部を除外したすべての EKS クラスターのモニタリング (除外タグの使用)	1. モニタリングから除外する EKS クラスターにタグを追加します。キーと値のペアは GuardDuty Managed -false です。タグの追加の詳細については、「Amazon EKS ユーザーガイド」の「CLI、API または eksctl でのタグの操作」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *ec2:CreateTags* を eks:TagResource で置き換えます。 • *ec2:DeleteTags* を eks:UntagResource で置き換えます。 • *access-project* を GuardDutyManaged に置き換えます。 • *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の PrincipalArn を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]` 3. Note EKS_RUNTIME_MONITORING の STATUS を ENABLED に設定する前に、必ず EKS クラス

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	ターに除外タグを追加してください。追加しないと、GuardDuty セキュリティエージェントがアカウント内のすべての EKS クラスターにデプロイされます。 ユーザー独自のリージョンレベルのディテクター ID を使用し、`features` オブジェクト名を `EKS_RUNTIME_MONITORING` として、ステータスを `ENABLED` として渡して、updateDetector API を実行します。 `EKS_ADDON_MANAGEMENT` のステータスを `ENABLED` に設定します。 GuardDuty は、モニタリングから除外されていないすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、`EKS_RUNTIME_MONITORING` と `EKS_ADDON_MANAGEMENT` の両方を有効にします。 `aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : " ENABLED", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "Status" : " ENABLED"}] }]'`

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
選択的な EKS クラスターのモニタリング (包含タグの使用)	1. モニタリングに含める EKS クラスターにタグを追加します。キーと値のペアは `GuardDutyManaged -true` です。タグの追加の詳細については、「Amazon EKS ユーザーガイド」の「CLI、API または eksctl でのタグの操作」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *ec2:CreateTags* を `eks:TagResource` で置き換えます。 • *ec2:DeleteTags* を `eks:UntagResource` で置き換えます。 • *access-project* を `GuardDutyManaged` に置き換えます。 • *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の `PrincipalArn` を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]` 3. ユーザー独自のリージョンレベルのディテクター ID を使用し、`features` オブジェクト名を `EKS_RUNTIME_MONITORING` として、ステータスを `ENABLED` として渡して、updateDetector API を実行します。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	EKS_ADDON_MANAGEMENT のステータスを DISABLED に設定します。 GuardDuty は、GuardDutyManaged -true ペアでタグ付けされたすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、EKS_RUNTIME_MONITORING を有効にして、EKS_ADDON_MANAGEMENT を無効にしています。 `aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : " ENABLED", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "Status" : " DISABLED"}] }]'`

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
セキュリティエージェントの手動管理	1. ユーザー独自のリージョンレベルのディテクター ID を使用し、features オブジェクト名を EKS_RUNTIME_MONITORING として、ステータスを ENABLED として渡して、updateDetector API を実行します。 EKS_ADDON_MANAGEMENT のステータスを DISABLED に設定します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、EKS_RUNTIME_MONITORING を有効にして、EKS_ADDON_MANAGEMENT を無効にしています。 `aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : " ENABLED", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "Status" : " DISABLED"}] }]'` 2. セキュリティエージェントを管理するには、「Amazon EKS クラスターのセキュリティエージェントの手動管理」を参照してください。

すべてのメンバーアカウントのために EKS Runtime Monitoring を自動的に有効にする

このセクションでは、すべてのメンバーアカウントで EKS Runtime Monitoring を有効にし、セキュリティエージェントを管理するステップについて説明します。これには、委任 GuardDuty 管理者アカウント、既存のメンバーアカウント、および組織に参加する新しいアカウントが含まれます。

のアプローチに基づいてセキュリティエージェントの Amazon EKS へのデプロイ、任意のアプローチを選択し、次の表で説明されているステップに従います。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
GuardDuty によるセキュリティエージェントの管理 (すべての EKS クラスターのモニタリング)	メンバーアカウントの EKS Runtime Monitoring を選択的に有効にするには、自分の`###### ID` を使用し、updateMemberDetectors API オペレーションを実行します。 `EKS_ADDON_MANAGEMENT` のステータスを `ENABLED` に設定します。 GuardDuty は、アカウント内のすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、`EKS_RUNTIME_MONITORING` と `EKS_ADDON_MANAGEMENT` の両方を有効にします。 `aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0 --account-ids 111122223333 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : " ENABLED", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "Status" : "ENABLED"}] }]'`

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	Note スペースで区切られたアカウント ID のリストを渡すこともできます。 コードが正常に実行されると、UnprocessedAccounts の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
一部を除外したすべての EKS クラスターのモニタリング (除外タグの使用)	1. モニタリングから除外する EKS クラスターにタグを追加します。キーと値のペアは GuardDutyManaged -false です。タグの追加の詳細については、「Amazon EKS ユーザーガイド」の「CLI、API または eksctl でのタグの操作」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *ec2:CreateTags* を eks:TagResource で置き換えます。 • *ec2:DeleteTags* を eks:UntagResource で置き換えます。 • *access-project* を GuardDutyManaged に置き換えます。 • *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の PrincipalArn を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]` 3. **Note** EKS_RUNTIME_MONITORING の STATUS を ENABLED に設定する前に、必ず EKS クラスターに除外タグを追加してください。追加しないと、GuardDuty セキュリ

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	ティエージェントがアカウント内のすべての EKS クラスターにデプロイされます。 ユーザー独自のリージョンレベルのディテクター ID を使用し、`features` オブジェクト名を `EKS_RUNTIME_MONITORING` として、ステータスを `ENABLED` として渡して、updateDetector API を実行します。 `EKS_ADDON_MANAGEMENT` のステータスを `ENABLED` に設定します。 GuardDuty は、モニタリングから除外されていないすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、`EKS_RUNTIME_MONITORING` と `EKS_ADDON_MANAGEMENT` の両方を有効にします。 `aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0 --account-ids 111122223333 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : " ENABLED", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "Status" : " ENABLED"}] }]'`

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	> **Note** スペースで区切られたアカウント ID のリストを渡すこともできます。 コードが正常に実行されると、`UnprocessedAccounts` の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
選択的な EKS クラスターのモニタリング (包含タグの使用)	1. モニタリングに含める EKS クラスターにタグを追加します。キーと値のペアは GuardDutyManaged -true です。タグの追加の詳細については、「Amazon EKS ユーザーガイド」の「CLI、API または eksctl でのタグの操作」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *ec2:CreateTags* を eks:TagResource で置き換えます。 • *ec2:DeleteTags* を eks:UntagResource で置き換えます。 • *access-project* を GuardDutyManaged に置き換えます。 • *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の PrincipalArn を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]` 3. ユーザー独自のリージョンレベルのディテクター ID を使用し、features オブジェクト名を EKS_RUNTIME_MONITORING として、ステータスを ENABLED として渡して、updateDetector API を実行します。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ

EKS_ADDON_MANAGEMENT のステータスを DISABLED に設定します。

GuardDuty は、GuardDutyManaged -true ペアでタグ付けされたすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。

または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

次の例では、EKS_RUNTIME_MONITORING を有効にして、EKS_ADDON_MANAGEMENT を無効にしています。

```
aws guardduty update-member-detectors --detector-
id 12abc34d567e8fa901bc2d34e56789f0    --account-
ids 111122223333  --features '[{"Name" : "EKS_RUNT
IME_MONITORING", "Status" : "  ENABLED", "Addition
alConfiguration" : [{"Name" : "EKS_ADDON_MANAGEM
ENT", "Status" : " DISABLED"}] }]'
```

> **Note**
>
> スペースで区切られたアカウント ID のリストを渡すこともできます。

コードが正常に実行されると、UnprocessedAccounts の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
セキュリティエージェントの手動管理	1. ユーザー独自のリージョンレベルのディテクター ID を使用し、`features` オブジェクト名を `EKS_RUNTIME_MONITORING` として、ステータスを `ENABLED` として渡して、updateDetector API を実行します。 `EKS_ADDON_MANAGEMENT` のステータスを `DISABLED` に設定します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、`EKS_RUNTIME_MONITORING` を有効にして、`EKS_ADDON_MANAGEMENT` を無効にしています。 `aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0 --account-ids 555555555555 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : " ENABLED", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "Status" : " DISABLED"}] } ]'` 2. セキュリティエージェントを管理するには、「Amazon EKS クラスターのセキュリティエージェントの手動管理」を参照してください。

すべての既存のアクティブなメンバーアカウントの EKS Runtime Monitoring の設定

このセクションでは、EKS Runtime Monitoring を有効にし、組織内の既存のアクティブなメンバーアカウントの GuardDuty セキュリティエージェントを管理するステップについて説明します。

のアプローチに基づいてセキュリティエージェントの Amazon EKS へのデプロイ、任意のアプローチを選択し、次の表で説明されているステップに従います。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
GuardDuty によるセキュリティエージェントの管理 (すべての EKS クラスターのモニタリング)	メンバーアカウントの EKS Runtime Monitoring を選択的に有効にするには、自分の`###### ID` を使用し、updateMemberDetectors API オペレーションを実行します。 `EKS_ADDON_MANAGEMENT` のステータスを `ENABLED` に設定します。 GuardDuty は、アカウント内のすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、`EKS_RUNTIME_MONITORING` と `EKS_ADDON_MANAGEMENT` の両方を有効にします。 `aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0 --account-ids 111122223333 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : " ENABLED", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "Status" : "ENABLED"}] }]'` **Note** スペースで区切られたアカウント ID のリストを渡すこともできます。 コードが正常に実行されると、`UnprocessedAccounts` の空のリストが返されます。アカウントのディテクター設定を変更する際

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
一部を除外したすべての EKS クラスターのモニタリング (除外タグの使用)	1. モニタリングから除外する EKS クラスターにタグを追加します。キーと値のペアは GuardDutyManaged -false です。タグの追加の詳細については、「Amazon EKS ユーザーガイド」の「CLI、API または eksctl でのタグの操作」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *ec2:CreateTags* を eks:TagResource で置き換えます。 • *ec2:DeleteTags* を eks:UntagResource で置き換えます。 • *access-project* を GuardDutyManaged に置き換えます。 • *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の PrincipalArn を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]` 3. **Note** EKS_RUNTIME_MONITORING の STATUS を ENABLED に設定する前に、必ず EKS クラスターに除外タグを追加してください。追加しないと、GuardDuty セキュリ

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ

ティエージェントがアカウント内のすべての EKS クラスターにデプロイされます。

メンバーアカウントの EKS Runtime Monitoring を選択的に有効にするには、自分の`###### ID` を使用し、updateMemberDetectors API オペレーションを実行します。

`EKS_ADDON_MANAGEMENT` のステータスを `ENABLED` に設定します。

GuardDuty は、モニタリングから除外されていないすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。

または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

次の例では、`EKS_RUNTIME_MONITORING` と `EKS_ADDON_MANAGEMENT` の両方を有効にします。

```
aws guardduty update-member-detectors --detector-
id 12abc34d567e8fa901bc2d34e56789f0    --account-
ids 111122223333  --features '[{"Name" : "EKS_RUNT
IME_MONITORING", "Status" : "  ENABLED", "Addition
alConfiguration" : [{"Name" : "EKS_ADDON_MANAGEM
ENT", "Status" : " ENABLED"}] }]'
```

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	> **Note** スペースで区切られたアカウント ID のリストを渡すこともできます。 コードが正常に実行されると、`UnprocessedAccounts` の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
選択的な EKS クラスターのモニタリング (包含タグの使用)	1. モニタリングに含める EKS クラスターにタグを追加します。キーと値のペアは GuardDutyManaged -true です。タグの追加の詳細については、「Amazon EKS ユーザーガイド」の「CLI、API または eksctl でのタグの操作」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *ec2:CreateTags* を eks:TagResource で置き換えます。 • *ec2:DeleteTags* を eks:UntagResource で置き換えます。 • *access-project* を GuardDutyManaged に置き換えます。 • *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の PrincipalArn を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]` 3. メンバーアカウントの EKS Runtime Monitoring を選択的に有効にするには、自分の*###### ID* を使用し、updateMemberDetectors API オペレーションを実行します。 EKS_ADDON_MANAGEMENT のステータスを DISABLED に設定します。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ

GuardDuty は、GuardDutyManaged -true ペアでタグ付けされたすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。

または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

次の例では、EKS_RUNTIME_MONITORING を有効にして、EKS_ADDON_MANAGEMENT を無効にしています。

```
aws guardduty update-member-detectors --detector-
id 12abc34d567e8fa901bc2d34e56789f0    --account-
ids 111122223333  --features '[{"Name" : "EKS_RUNT
IME_MONITORING", "Status" : "  ENABLED", "Addition
alConfiguration" : [{"Name" : "EKS_ADDON_MANAGEM
ENT", "Status" : " DISABLED"}] }]'
```

> **Note**
>
> スペースで区切られたアカウント ID のリストを渡すこともできます。

コードが正常に実行されると、UnprocessedAccounts の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
セキュリティエージェントの手動管理	1. メンバーアカウントの EKS Runtime Monitoring を選択的に有効にするには、自分の`###### ID` を使用し、updateMemberDetectors API オペレーションを実行します。 `EKS_ADDON_MANAGEMENT` のステータスを `DISABLED` に設定します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、`EKS_RUNTIME_MONITORING` を有効にして、`EKS_ADDON_MANAGEMENT` を無効にしています。 `aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0 --account-ids 555555555555 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : " ENABLED", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "Status" : " DISABLED"}] } ]'` 2. セキュリティエージェントを管理するには、「Amazon EKS クラスターのセキュリティエージェントの手動管理」を参照してください。

新規メンバーの EKS Runtime Monitoring を自動有効化

委任 GuardDuty 管理者アカウントは、組織に参加する新しいアカウントの EKS Runtime Monitoring を自動的に有効にし、GuardDuty セキュリティエージェントを管理する方法のアプローチを選択することができます。

のアプローチに基づいてセキュリティエージェントの Amazon EKS へのデプロイ、任意のアプローチを選択し、次の表で説明されているステップに従います。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
GuardDuty によるセキュリティエージェントの管理 (すべての EKS クラスターのモニタリング)	新しいアカウントの EKS Runtime Monitoring を選択的に有効にするには、自身の`###### ID` を使用し、UpdateOrganizationConfiguration API オペレーションを実行します。 `EKS_ADDON_MANAGEMENT` のステータスを `ENABLED` に設定します。 GuardDuty は、アカウント内のすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、1 つのアカウントで `EKS_RUNTIME_MONITORING` と `EKS_ADDON_MANAGEMENT` の両方を有効にします。スペースで区切られたアカウント ID のリストを渡すこともできます。 アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 `aws guardduty update-organization-configuration --detector-id 12abc34d567e8fa901bc2d34e56789f0 --autoEnable --features '[{"Name" : "EKS_RUNTIME_MONITORING", "AutoEnable": "NEW", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "AutoEnable": "NEW"}] }]'`

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	コードが正常に実行されると、UnprocessedAccounts の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
一部を除外したすべての EKS クラスターのモニタリング (除外タグの使用)	1. モニタリングから除外する EKS クラスターにタグを追加します。キーと値のペアは GuardDuty Managed -false です。タグの追加の詳細については、「Amazon EKS ユーザーガイド」の「CLI、API または eksctl でのタグの操作」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *ec2:CreateTags* を eks:TagResource で置き換えます。 • *ec2:DeleteTags* を eks:UntagResource で置き換えます。 • *access-project* を GuardDutyManaged に置き換えます。 • *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の PrincipalArn を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]` 3. Note EKS_RUNTIME_MONITORING の STATUS を ENABLED に設定する前に、必ず EKS クラス

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	ターに除外タグを追加してください。追加しないと、GuardDuty セキュリティエージェントがアカウント内のすべての EKS クラスターにデプロイされます。 新しいアカウントの EKS Runtime Monitoring を選択的に有効にするには、自身の`###### ID` を使用し、UpdateOrganizationConfiguration API オペレーションを実行します。 `EKS_ADDON_MANAGEMENT` のステータスを `ENABLED` に設定します。 GuardDuty は、モニタリングから除外されていないすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、1 つのアカウントで `EKS_RUNTIME_MONITORING` と `EKS_ADDON_MANAGEMENT` の両方を有効にします。スペースで区切られたアカウント ID のリストを渡すこともできます。 アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	``` aws guardduty update-organization-configuration --detector-id 12abc34d567e8fa901bc2d34e56789f0 --autoEnable --features '[{"Name" : "EKS_RUNTIME_MONITORING", "AutoEnable": "NEW", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "AutoEnable": "NEW"}] }]' ``` コードが正常に実行されると、UnprocessedAccounts の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
選択的な EKS クラスターのモニタリング (包含タグの使用)	1. モニタリングに含める EKS クラスターにタグを追加します。キーと値のペアは `GuardDutyManaged -true` です。タグの追加の詳細については、「Amazon EKS ユーザーガイド」の「CLI、API または eksctl でのタグの操作」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *`ec2:CreateTags`* を `eks:TagResource` で置き換えます。 • *`ec2:DeleteTags`* を `eks:UntagResource` で置き換えます。 • *`access-project`* を `GuardDutyManaged` に置き換えます。 • *`123456789012`* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の `PrincipalArn` を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]` 3. 新しいアカウントの EKS Runtime Monitoring を選択的に有効にするには、自身の*`###### ID`* を使用し、UpdateOrganizationConfiguration API オペレーションを実行します。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	EKS_ADDON_MANAGEMENT のステータスを DISABLED に設定します。 GuardDuty は、GuardDutyManaged -true ペアでタグ付けされたすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、1 つのアカウントで EKS_RUNTIME_MONITORING を有効にし、EKS_ADDON_MANAGEMENT を無効にします。スペースで区切られたアカウント ID のリストを渡すこともできます。 アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 `aws guardduty update-organization-configuration --detector-id 12abc34d567e8fa901bc2d34e56789f0 --autoEnable --features '[{"Name" : "EKS_RUNTIME_MONITORING", "AutoEnable": "NEW", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "AutoEnable": "NEW"}] }]'`

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	コードが正常に実行されると、UnprocessedAccounts の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
セキュリティエージェントの手動管理	1. 新しいアカウントの EKS Runtime Monitoring を選択的に有効にするには、自身の`###### ID` を使用し、UpdateOrganizationConfiguration API オペレーションを実行します。 `EKS_ADDON_MANAGEMENT` のステータスを `DISABLED` に設定します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、1 つのアカウントで `EKS_RUNTIME_MONITORING` を有効にし、`EKS_ADDON_MANAGEMENT` を無効にします。スペースで区切られたアカウント ID のリストを渡すこともできます。 アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 `aws guardduty update-organization-configuration --detector-id 12abc34d567e8fa901bc2d34e56789f0 --autoEnable --features '[{"Name" : "EKS_RUNTIME_MONITORING", "AutoEnable": "NEW", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "AutoEnable": "NEW"}] }]'` コードが正常に実行されると、`UnprocessedAccounts` の空のリストが返されます。アカウン

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	トのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。 2. セキュリティエージェントを管理するには、「Amazon EKS クラスターのセキュリティエージェントの手動管理」を参照してください。

個々のアクティブなメンバーアカウントの EKS Runtime Monitoring の有効化

このセクションでは、個々のアクティブメンバーアカウントの EKS Runtime Monitoring を設定し、セキュリティエージェントを管理するステップについて説明します。

のアプローチに基づいてセキュリティエージェントの Amazon EKS へのデプロイ、任意のアプローチを選択し、次の表で説明されているステップに従います。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
GuardDuty によるセキュリティエージェントの管理 (すべての EKS クラスターのモニタリング)	メンバーアカウントの EKS Runtime Monitoring を選択的に有効にするには、自分の`###### ID` を使用し、updateMemberDetectors API オペレーションを実行します。 `EKS_ADDON_MANAGEMENT` のステータスを `ENABLED` に設定します。 GuardDuty は、アカウント内のすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの `detectorId` を検索するに

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	は、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、EKS_RUNTIME_MONITORING と EKS_ADDON_MANAGEMENT の両方を有効にします。 `aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0 --account-ids 111122223333 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : "ENABLED", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "Status" : "ENABLED"}] }]'` **Note** スペースで区切られたアカウント ID のリストを渡すこともできます。 コードが正常に実行されると、UnprocessedAccounts の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
一部を除外したすべての EKS クラスターのモニタリング (除外タグの使用)	1. モニタリングから除外する EKS クラスターにタグを追加します。キーと値のペアは GuardDuty Managed -false です。タグの追加の詳細については、「Amazon EKS ユーザーガイド」の「CLI、API または eksctl でのタグの操作」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *ec2:CreateTags* を eks:TagResource で置き換えます。 • *ec2:DeleteTags* を eks:UntagResource で置き換えます。 • *access-project* を GuardDutyManaged に置き換えます。 • *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の PrincipalArn を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]` 3. Note EKS_RUNTIME_MONITORING の STATUS を ENABLED に設定する前に、必ず EKS クラス

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	ターに除外タグを追加してください。追加しないと、GuardDuty セキュリティエージェントがアカウント内のすべての EKS クラスターにデプロイされます。 メンバーアカウントの EKS Runtime Monitoring を選択的に有効にするには、自分の`###### ID` を使用し、updateMemberDetectors API オペレーションを実行します。 `EKS_ADDON_MANAGEMENT` のステータスを `ENABLED` に設定します。 GuardDuty は、モニタリングから除外されていないすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、`EKS_RUNTIME_MONITORING` と `EKS_ADDON_MANAGEMENT` の両方を有効にします。 `aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0 --account-ids 111122223333 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : "ENABLED", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "Status" : "ENABLED"}] }]'`

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	> **Note** スペースで区切られたアカウント ID のリストを渡すこともできます。 コードが正常に実行されると、`UnprocessedAccounts` の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
選択的な EKS クラスターのモニタリング (包含タグの使用)	1. モニタリングに含める EKS クラスターにタグを追加します。キーと値のペアは `GuardDutyManaged -true` です。タグの追加の詳細については、「Amazon EKS ユーザーガイド」の「CLI、API または eksctl でのタグの操作」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *`ec2:CreateTags`* を `eks:TagResource` で置き換えます。 • *`ec2:DeleteTags`* を `eks:UntagResource` で置き換えます。 • *`access-project`* を `GuardDutyManaged` に置き換えます。 • *`123456789012`* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の `PrincipalArn` を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]` 3. メンバーアカウントの EKS Runtime Monitoring を選択的に有効にするには、自分の*`###### ID`* を使用し、updateMemberDetectors API オペレーションを実行します。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	EKS_ADDON_MANAGEMENT のステータスを DISABLED に設定します。 GuardDuty は、GuardDutyManaged -true ペアでタグ付けされたすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、EKS_RUNTIME_MONITORING を有効にして、EKS_ADDON_MANAGEMENT を無効にしています。 `aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0 --account-ids 111122223333 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : "ENABLED", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "Status" : "DISABLED"}] }]'` **Note** スペースで区切られたアカウント ID のリストを渡すこともできます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	コードが正常に実行されると、UnprocessedAccounts の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
セキュリティエージェントの手動管理	1. メンバーアカウントの EKS Runtime Monitoring を選択的に有効にするには、自分の`###### ID` を使用し、updateMemberDetectors API オペレーションを実行します。 `EKS_ADDON_MANAGEMENT` のステータスを `DISABLED` に設定します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、`EKS_RUNTIME_MONITORING` を有効にして、`EKS_ADDON_MANAGEMENT` を無効にしています。 `aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0 --account-ids 555555555555 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : "ENABLED", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "Status" : "DISABLED"}] } ]'` 2. セキュリティエージェントを管理するには、「Amazon EKS クラスターのセキュリティエージェントの手動管理」を参照してください。

スタンドアロンアカウントの EKS Runtime Monitoring の設定 (API)

スタンドアロンアカウントは、特定の の で保護プランを有効または無効にする決定を所有 AWS アカウント します AWS リージョン。

アカウントが を通じて AWS Organizations、または招待方法によって GuardDuty 管理者アカウントに関連付けられている場合、このセクションはアカウントには適用されません。詳細については、「マルチアカウント環境の EKS Runtime Monitoring の設定 (API)」を参照してください。

Runtime Monitoring を有効にした後、自動設定または手動デプロイを使用して GuardDuty セキュリティエージェントをインストールしてください。次の手順に記載されたすべてのステップを完了する一環として、セキュリティエージェントをインストールしてください。

のアプローチに基づいてセキュリティエージェントの Amazon EKS へのデプロイ、任意のアプローチを選択し、次の表で説明されているステップに従います。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
GuardDuty によるセキュリティエージェントの管理 (すべての EKS クラスターのモニタリング)	1. ユーザー独自のリージョンレベルのディテクター ID を使用し、`features` オブジェクト名を `EKS_RUNTIME_MONITORING` として、ステータスを `ENABLED` として渡して、updateDetector API を実行します。 `EKS_ADDON_MANAGEMENT` のステータスを `ENABLED` に設定します。 GuardDuty は、アカウント内のすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 2. または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	次の例では、EKS_RUNTIME_MONITORING と EKS_ADDON_MANAGEMENT の両方を有効にします。 `aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : " ENABLED", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "Status" : " ENABLED"}] }]'`

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
一部を除外したすべての EKS クラスターのモニタリング (除外タグの使用)	1. モニタリングから除外する EKS クラスターにタグを追加します。キーと値のペアは GuardDuty Managed -false です。タグの追加の詳細については、「Amazon EKS ユーザーガイド」の「CLI、API または eksctl でのタグの操作」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *ec2:CreateTags* を eks:TagResource で置き換えます。 • *ec2:DeleteTags* を eks:UntagResource で置き換えます。 • *access-project* を GuardDutyManaged に置き換えます。 • *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の PrincipalArn を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]` 3. Note EKS_RUNTIME_MONITORING の STATUS を ENABLED に設定する前に、必ず EKS クラス

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	ターに除外タグを追加してください。追加しないと、GuardDuty セキュリティエージェントがアカウント内のすべての EKS クラスターにデプロイされます。 ユーザー独自のリージョンレベルのディテクター ID を使用し、`features` オブジェクト名を `EKS_RUNTIME_MONITORING` として、ステータスを `ENABLED` として渡して、updateDetector API を実行します。 `EKS_ADDON_MANAGEMENT` のステータスを `ENABLED` に設定します。 GuardDuty は、モニタリングから除外されていないすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、`EKS_RUNTIME_MONITORING` と `EKS_ADDON_MANAGEMENT` の両方を有効にします。 `aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : " ENABLED", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "Status" : " ENABLED"}] }]'`

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
選択的な EKS クラスターのモニタリング (包含タグの使用)	1. モニタリングに含める EKS クラスターにタグを追加します。キーと値のペアは GuardDutyManaged -true です。タグの追加の詳細については、「Amazon EKS ユーザーガイド」の「CLI、API または eksctl でのタグの操作」を参照してください。 2. 信頼できるエンティティ以外によるタグの変更を防ぐには、「AWS Organizations ユーザーガイド」の「許可されたプリンシパル以外のタグが変更されないようにする」に記載されているポリシーを使用してください。このポリシーで、以下の詳細を置き換えます。 • *ec2:CreateTags* を eks:TagResource で置き換えます。 • *ec2:DeleteTags* を eks:UntagResource で置き換えます。 • *access-project* を GuardDutyManaged に置き換えます。 • *123456789012* を信頼されたエンティティの AWS アカウント ID に置き換えます。 信頼できるエンティティが複数ある場合は、次の例を使用して複数の PrincipalArn を追加します。 `"aws:PrincipalArn":["arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin", "arn:aws:iam::123456789012:role/org-admins/iam-admin"]` 3. ユーザー独自のリージョンレベルのディテクター ID を使用し、features オブジェクト名を EKS_RUNTIME_MONITORING として、ステータスを ENABLED として渡して、updateDetector API を実行します。

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
	EKS_ADDON_MANAGEMENT のステータスを DISABLED に設定します。 GuardDuty は、GuardDutyManaged -true ペアでタグ付けされたすべての Amazon EKS クラスターのセキュリティエージェントのデプロイと更新を管理します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、EKS_RUNTIME_MONITORING を有効にして、EKS_ADDON_MANAGEMENT を無効にしています。 `aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : " ENABLED", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "Status" : " DISABLED"}] }]'`

GuardDuty セキュリティエージェントを管理するための推奨アプローチ	ステップ
セキュリティエージェントの手動管理	1. ユーザー独自のリージョンレベルのディテクター ID を使用し、`features` オブジェクト名を `EKS_RUNTIME_MONITORING` として、ステータスを `ENABLED` として渡して、updateDetector API を実行します。 `EKS_ADDON_MANAGEMENT` のステータスを `DISABLED` に設定します。 または、独自のリージョンディテクター ID を使用して AWS CLI コマンドを使用することもできます。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。 次の例では、`EKS_RUNTIME_MONITORING` を有効にして、`EKS_ADDON_MANAGEMENT` を無効にしています。 `aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : " ENABLED", "AdditionalConfiguration" : [{"Name" : "EKS_ADDON_MANAGEMENT", "Status" : " DISABLED"}] }]'` 2. セキュリティエージェントを管理するには、「Amazon EKS クラスターのセキュリティエージェントの手動管理」を参照してください。

EKS Runtime Monitoring から Runtime Monitoring への移行

GuardDuty Runtime Monitoring が開始されたことで、脅威検出の対象範囲が Amazon ECS コンテナと Amazon EC2 インスタンスにまで拡大されました。EKS Runtime Monitoring のエクスペリエンス

は Runtime Monitoring に統合されました。Runtime Monitoring を有効にして、ランタイムの動作を監視したいリソースタイプ (Amazon EC2 インスタンス、Amazon ECS クラスター、Amazon EKS クラスター) ごとに個別の GuardDuty セキュリティエージェントを管理できます。

GuardDuty は、EKS Runtime Monitoring のコンソールエクスペリエンスを Runtime Monitoring に統合しました。GuardDuty は「EKS Runtime Monitoring 設定ステータスの確認」および「EKS Runtime Monitoring から Runtime Monitoring への移行」を推奨します。

Runtime Monitoring への移行の一環として、「EKS Runtime Monitoring を無効にする」を必ず行ってください。これは重要です。後で Runtime Monitoring を無効にし、EKS Runtime Monitoring を無効にしない場合、EKS Runtime Monitoring の使用コストが引き続き発生するためです。

EKS Runtime Monitoring から Runtime Monitoring に移行するには

1. GuardDuty コンソールは、Runtime Monitoring の一部として EKS Runtime Monitoring をサポートしています。

   組織やアカウントの EKS Runtime Monitoring 設定ステータスの確認 ごとに Runtime Monitoring の使用を開始できます。

   Runtime Monitoring を有効にする前に、EKS Runtime Monitoring を無効にしないでください。EKS Runtime Monitoring を無効にすると、Amazon EKS アドオン管理も無効になります。記載された順序で、次のステップを続行します。
2. すべての Runtime Monitoring を有効にする前提条件 を満たしていることを確認してください。
3. EKS Runtime Monitoring と同じ組織構成設定を Runtime Monitoring に対して複製することで、Runtime Monitoring を有効にします。詳細については、「Runtime Monitoring の有効化」を参照してください。
   - スタンドアロンアカウントをお持ちの場合は、Runtime Monitoring を有効にする必要があります。

     GuardDuty セキュリティエージェントが既にデプロイされている場合は、対応する設定が自動的に複製されるため、設定を再設定する必要はありません。
   - 自動有効化設定を行っている組織がある場合は、必ず同じ自動有効化設定を Runtime Monitoring に複製してください。
   - 既存のアクティブメンバーアカウントを個別に設定が構成されている組織がある場合は、これらのメンバーに対して必ず Runtime Monitoring を有効にし、GuardDuty セキュリティエージェントを個別に設定してください。

4. Runtime Monitoring と GuardDuty セキュリティエージェントの設定が正しいことを確認したら、API または AWS CLI コマンドを使用して EKS Runtime Monitoring を無効にします。
5. (オプション) GuardDuty セキュリティエージェントに関連付けられているリソースをクリーンアップする場合は、「Runtime Monitoring でのリソースの無効化、アンインストール、クリーンアップ」を参照してください。

Runtime Monitoring を有効にせずに EKS Runtime Monitoring を引き続き使用する場合は、「GuardDuty での EKS Runtime Monitoring」を参照してください。ユースケースに基づいて、スタンドアロンアカウントまたは複数のメンバーアカウントに対して EKS Runtime Monitoring を設定するステップを選択します。

EKS Runtime Monitoring 設定ステータスの確認

次の APIs または AWS CLI コマンドを使用して、EKS Runtime Monitoring の既存の設定ステータスを確認します。

アカウント内の既存の EKS Runtime Monitoring 設定ステータスを確認するには

- [GetDetector] を実行して、自分のアカウントの設定ステータスを確認します。
- または、 AWS CLIを使用して以下のコマンドを実行できます。

```
aws guardduty get-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --
region us-east-1
```

AWS アカウント と現在のリージョンのディテクター ID を必ず置き換えてください。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

組織の既存の EKS Runtime Monitoring 設定ステータスを確認するには (委任 GuardDuty 管理者アカウントのみ)

- [DescribeOrganizationConfiguration] を実行して、組織の設定ステータスを確認します。

  または、 AWS CLIを使用して以下のコマンドを実行できます。

```
aws guardduty describe-organization-configuration --detector-
id 12abc34d567e8fa901bc2d34e56789f0 --region us-east-1
```

ディテクター ID は必ず委任 GuardDuty 管理者アカウントのディテクター ID に、リージョンは現在のリージョンに置き換えてください。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

Runtime Monitoring への移行後に EKS Runtime Monitoring を無効にする

アカウントまたは組織の既存の設定が Runtime Monitoring に複製されたことを確認したら、EKS Runtime Monitoring を無効にできます。

EKS Runtime Monitoring を無効にするには

- 自分のアカウントで EKS Runtime Monitoring を無効にするには

  自身のリージョンの *detector-id* を使用して UpdateDetector API を実行します。

  または、次の AWS CLI コマンドを使用できます。*12abc34d567e8fa901bc2d34e56789f0* を自分のリージョン *detector-id* に置き換えます。

  ```
  aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --
  features '[{"Name" : "EKS_RUNTIME_MONITORING", "Status" : "DISABLED"}]'
  ```

- 組織内のメンバーアカウントの EKS Runtime Monitoring を無効にするには

  組織の委任 GuardDuty 管理者アカウントのリージョンの *detector-id* を使用して UpdateMemberDetectors API を実行します。

  または、次の AWS CLI コマンドを使用できます。*12abc34d567e8fa901bc2d34e56789f0* を組織の委任 GuardDuty 管理者アカウントのリージョン別 *detector-id* に、*111122223333* をこの機能を無効にするメンバーアカウントの AWS アカウント ID に置き換えます。

  ```
  aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0
   --account-ids 111122223333 --features '[{"Name" : "EKS_RUNTIME_MONITORING",
   "Status" : "DISABLED"}]'
  ```

- 組織の EKS Runtime Monitoring 自動有効化設定を更新するには

  EKS Runtime Monitoring 自動有効化設定を組織内の新しい (NEW) メンバーアカウントまたはすべての (ALL) メンバーアカウントに設定した場合にのみ、次のステップを実行します。既に NONE として設定している場合は、このステップをスキップできます。

> **Note**
>
> EKS Runtime Monitoring の自動有効化設定を NONE に設定すると、既存のメンバーアカウントまたは新しいメンバーアカウントが組織に加わったときに EKS Runtime Monitoring が自動的に有効にならないことを意味します。

組織の委任 GuardDuty 管理者アカウントのリージョン *detector-id* を使用して UpdateOrganizationConfiguration API を実行します。

または、次の AWS CLI コマンドを使用できます。*12abc34d567e8fa901bc2d34e56789f0* を、組織の委任 GuardDuty 管理者アカウントのリージョン *detector-id* に置き換えます。*EXISTING_VALUE* を、GuardDuty を自動有効化するための現在の設定に置き換えます。

```
aws guardduty update-organization-configuration --detector-
id 12abc34d567e8fa901bc2d34e56789f0 --auto-enable-organization-members EXISTING_VALUE
 --features '[{"Name" : "EKS_RUNTIME_MONITORING", "AutoEnable": "NONE"}]'
```

GuardDuty セキュリティエージェントのリリースバージョン

GuardDuty は、更新されたエージェントバージョンを随時リリースします。GuardDuty がエージェントを自動的に管理する場合、GuardDuty はユーザーに代わってエージェントを更新するように設計されています。エージェントを手動で管理する場合、Amazon EC2 インスタンス、Amazon ECS クラスター、Amazon EKS クラスターなどのリソースタイプのエージェントバージョンを更新する責任がユーザーにあります。

以下のセクションでは、GuardDuty セキュリティエージェントのリリースバージョンと、サポートされているすべてのリソースタイプに関連するリリースノートについて説明します。

トピック

- Amazon EC2 インスタンス用の GuardDuty セキュリティエージェントバージョン
- Amazon ECS-EC2 Bottlerocket の GuardDuty セキュリティエージェントバージョン
- の GuardDuty セキュリティエージェントバージョン AWS Fargate (Amazon ECS のみ)
- Amazon EKS リソースの GuardDuty セキュリティエージェントバージョン
- その他のリソース - 次のステップ

Amazon EC2 インスタンス用の GuardDuty セキュリティエージェントバージョン

次の表は、Amazon EC2 の GuardDuty セキュリティエージェントのリリースバージョン履歴を示しています。

SSM ディストリビューターバージョン	[エージェントバージョン]	リリースノート	利用可能日
v1.17.1	v1.17.1	カーネル 7.2 のサポートが追加されました。 起動時に、エージェントは GuardDuty 接続タイプを検出し、link-local 経由で guardduty-data-v2 エンドポイントへの接続を試みます。 セキュリティ機能の一般的な更新。	2026 年 9 月 9 日
v1.16.0	v1.16.0	カーネル 6.19、7.0、7.1 のサポートが追加されました。 Ubuntu 26.04、Debian 13、RedHat 10.2、CentOS Stream 10、Rocky Linux 10.1、Fedora 43、および Fedora 44 のサポートが追加されました。Amazon EC2 リソースのすべての検証済み OS	2026 年 7 月 17 日

SSM ディストリビューターバージョン	[エージェントバージョン]	リリースノート	利用可能日
		ディストリビューションのリストについては、「アーキテクチャ要件を検証する」を参照してください。 セキュリティ機能の一般的な更新。	
v1.15.0	v1.15.0	最適化されたイベント収集をサポートする新しいエージェント機能により、フィルターのデプロイが高速化され、検出速度が向上します。 セキュリティ機能の一般的な更新。	2026 年 5 月 14 日

SSM ディストリビューターバージョン	[エージェントバージョン]	リリースノート	利用可能日
v1.9.2	v1.9.2	カーネル 6.15、6.16、6.17、6.18 のサポートが追加されました。 Alma Linux 9、Alma Linux 10、SUSE Linux Enterprise Server 16 のサポートが追加されました。Amazon EC2 リソースのすべての検証済み OS ディストリビューションのリストについては、「アーキテクチャ要件を検証する」を参照してください。	2026 年 2 月 24 日
v1.9.1	v1.9.1	一般的なパフォーマンス調整とセキュリティ機能の更新。	2025 年 11 月 10 日
v1.9.0	v1.9.0	一般的なパフォーマンスの調整と強化。	2025 年 10 月 23 日
v1.8.0	v1.8.0	一般的なパフォーマンスの調整と強化。	2025 年 8 月 12 日

SSM ディストリビューターバージョン	[エージェントバージョン]	リリースノート	利用可能日
v1.7.2	v1.7.1	RPM ベースの Linux ディストリビューション向けローカルエージェントインストール機能のサポート強化しました。	2025 年 7 月 23 日
v1.7.1	1.7.1	Fedora 40 と Fedora 41 のサポートが追加されました。Amazon EC2 リソースのすべての検証済み OS ディストリビューションのリストについては、「アーキテクチャ要件を検証する」を参照してください。 一般的なパフォーマンスの調整と強化。	2025 年 6 月 3 日

SSM ディストリビューターバージョン	[エージェントバージョン]	リリースノート	利用可能日
v1.7.0	v1.7.0	Oracle Linux バージョン 8.9 および 9.3、Rocky Linux バージョン 9.5 のサポートが追加されました。Amazon EC2 リソースのすべての検証済み OS ディストリビューションのリストについては、「アーキテクチャ要件を検証する」を参照してください。 改善されたコンテナ ID の解決。 一般的なパフォーマンスの調整と強化。	2025 年 4 月 3 日
v1.6.0	v1.6.0	一般的なパフォーマンスの調整と強化。	2025 年 2 月 6 日

SSM ディストリビューターバージョン	[エージェントバージョン]	リリースノート	利用可能日
v1.5.0	v1.5.0	CentOS Stream 9.0、RedHat 9.4、Fedora 34.0、および Ubuntu 24.04 のサポートが追加されました。 `.../MetadataDNSRebind` 検出結果の ARM インスタンスのサポート。 一般的なパフォーマンスの調整と強化。	2024 年 11 月 20 日
v1.3.1	v1.3.1	カスタム DNS リゾルバーのサポート。	2024 年 9 月 12 日
v1.3.0	v1.3.0	一般的なパフォーマンスの調整と強化。 将来の「GuardDuty Runtime Monitoring の検出結果タイプ」の追加セキュリティシグナルをキャプチャするためのサポートが含まれています。	2024 年 8 月 19 日

SSM ディストリビューターバージョン	[エージェントバージョン]	リリースノート	利用可能日
v1.2.0	v1.2.0	OS ディストリビューション Ubuntu 20.04、Ubuntu 22.04、Debian 11、および Debian 12 をサポートします。 カーネル 6.5 と 6.8 をサポートします。 一般的なパフォーマンスの調整と強化。	2024 年 6 月 13 日
v1.1.0	v1.1.0	Amazon EC2 インスタンスの Runtime Monitoring で GuardDuty 自動エージェント設定をサポートします。 EC2 インスタンスの Runtime Monitoring の一般提供の発表に伴ってリリースされた新しいセキュリティシグナルと検出結果をサポートします。 一般的なパフォーマンスの調整と強化。	2024 年 3 月 26 日

SSM ディストリビューターバージョン	[エージェントバージョン]	リリースノート	利用可能日
v1.0.2	v1.0.2	最新の Amazon ECS AMI をサポートします。	2024 年 2 月 2 日
v1.0.1	v1.0.1	v1.0.2 より前にリリースされたエージェントバージョンは、2024 年 1 月 31 日以降に起動された Amazon ECS AMI と互換性がありません。 一般的なパフォーマンスの調整と強化。	2024 年 1 月 23 日
v1.0.0	v1.0.0	RPM インストールの初期リリース。 v1.0.2 より前にリリースされたエージェントバージョンは、2024 年 1 月 31 日以降に起動された Amazon ECS AMI と互換性がありません。	2023 年 11 月 26 日

Amazon ECS-EC2 Bottlerocket の GuardDuty セキュリティエージェントバージョン

次の表は、Amazon ECS-EC2 Bottlerocket インスタンスの GuardDuty セキュリティエージェントのリリースバージョン履歴を示しています。エージェントは、Bottlerocket ECS-2 および ECS-3 最適化 AMI バージョン `v1.62.1` 以降でホストコンテナとして実行されます。

SSM ディストリビューターバージョン	[エージェントバージョン]	リリースノート	利用可能日
v1.17.1	v1.17.1	カーネル 7.2 のサポートが追加されました。 起動時に、エージェントは GuardDuty 接続タイプを検出し、link-local 経由で guardduty-data-v2 エンドポイントへの接続を試みます。 セキュリティ機能の一般的な更新。	2026 年 9 月 9 日
v1.16.0	v1.16.0	Amazon ECS-EC2 Bottlerocket インスタンス用の GuardDuty セキュリティエージェントの初回リリース。 Bottlerocket ECS-2 および ECS-3 バリアント (x86_64 および aarch64) をサポートします。	2026 年 7 月 17 日

の GuardDuty セキュリティエージェントバージョン AWS Fargate (Amazon ECS のみ)

次の表は、Fargate 用 GuardDuty セキュリティエージェントのリリースバージョン履歴を示しています (Amazon ECS のみ)。

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日
v1.17.1	x86_64 (AMD64): sha256:27b27c489113ae83a272b5c98112e043f66faf8379838bb38e2d815dd6dfbf3a Graviton (ARM64): sha256:195078e48505cb05b06e61df255fc85a6d7917b28678956c75b11e69855bdff3	カーネル 7.2 のサポートが追加されました。 起動時に、エージェントは GuardDuty 接続タイプを検出し、link-local 経由で guardduty-data-v2 エンドポイントへの接続を試みます。 セキュリティ機能の一般的な更新。	2026 年 9 月 9 日
v1.16.0	x86_64 (AMD64): sha256:711fa37e2cce0417ba3310d81439d6b11170e446f685146b36916e4e38c1560d Graviton (ARM64): sha256:6851e00893eee958085a	セキュリティ機能の一般的な更新。	2026 年 7 月 17 日

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日
	daccddf91 bdc009ebd dcd46b4da 6fb1962b1 32d47454		
v1.15.0	x86_64 (AMD64): sha256:66446df341b ee1a1e218 aa46b7807 d0d047026 6c8f374c5 dbd97efcd b8356348 Graviton (ARM64): sha256:b0 def2aa8ca b6b2060c7 03f386fdb 29dc41660 54e0509a9 362b21a6e d7f9def7	最適化されたイベント収集をサポートする新しいエージェント機能により、フィルターのデプロイが高速化され、検出速度が向上します。 セキュリティ機能の一般的な更新。	2026 年 5 月 28 日

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日
v1.9.0	x86_64 (AMD64): sha256:fd9acaa2326f180f0ed0c5a25d8ff0a2d0498a6e900c34c68d4e973ea7fb26a8 Graviton (ARM64): sha256:0b04f8c28956684e752677bfd83dbc45afc8a43f7791fa44667b8078ba48a295	一般的なパフォーマンスの調整とセキュリティの強化。	2025 年 10 月 28 日

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日
v1.8.0	x86_64 (AMD64): sha256:4425417f39e38b24c1be428bacad1dad53e0645530dcf4422436353bfe358e3a Graviton (ARM64): sha256:aff069418fd6825846f8f575c49906a67c8a446d12d9ed0d21ab95bd0d05497b	一般的なパフォーマンスの調整と強化。	2025 年 8 月 12 日

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日
v1.7.0	x86_64 (AMD64): sha256:bf9197abdf853607e5fa392b4f97ccdd6ca56dd179be3ce8849e552d96582ac8 Graviton (ARM64): sha256:56c8683c948bcd82c0dbcebf755204365ac7285994693c11717bd45f86e279c2	改善されたコンテナ ID の解決。 一般的なパフォーマンスの調整と強化。	2025 年 4 月 4 日

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日
v1.6.0	x86_64 (AMD64): sha256:c8dea71d372bc47b2f236f7a091b9a9b06bc8193c1cfe4c9346eb50f89258897 Graviton (ARM64): sha256:f4032a566b90537646c2a987bef42eca1b498078ccc58a848603f877971a8dbe	一般的なパフォーマンスの調整と強化。	2025 年 2 月 6 日

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日
v1.5.0	x86_64 (AMD64): sha256:5e6fdc41f9eb748219d0498cd6c1dba6a19d875daec50167a0ac80e5028eac54 Graviton (ARM64): sha256:d56801ff6864d601474010 3b70b1c38431851358d182613bede20fe21090e734	.../MetadataDNSRebind 検出結果の ARM タスクのサポート。 一般的なパフォーマンスの調整と強化。	2024 年 11 月 14 日

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日
v1.4.1	x86_64 (AMD64): sha256:ef36a11151ec2d3d7db22273bfb954750dee76f0ac7bec37a7ba7e74c3de1c78 Graviton (ARM64): sha256:a8844544a59d6b4cba98f8e528b513ac2d97432f208e3ad497cc16b331aa9faa	コンテナイメージの強化。 一般的なパフォーマンスの調整と強化。	2024 年 10 月 24 日

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日
v1.3.1	x86_64 (AMD64): sha256:a6e2307d796e2875907bc4c1c69622c906f3192ddc42ef27b99e0a8f0979f3e0 Graviton (ARM64): sha256:ad1b6539d806edb504f17e6bcfb8b4026c5e822300afc31c0d23c6a08f9b99e9	カスタム DNS リゾルバーのサポート。	2024 年 9 月 11 日

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日
v1.3.0	x86_64 (AMD64): sha256:f1ad3fb2dc55a1110c60eecf4453b9f9c02f29acb261df39814e7d29296bf831 Graviton (ARM64): sha256:ff81a755d46681e409f55a95beedae9ebbcf5336e1c0b1e6348af7c6518bdbb1	一般的なパフォーマンスの調整と強化。 将来の GuardDuty「GuardDuty Runtime Monitoring の検出結果タイプ」の追加セキュリティシグナルをキャプチャするためのサポートが含まれています。	2024 年 8 月 9 日

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日
v1.2.0	x86_64 (AMD64): sha256:1dbad20ac2dc66d52d00bb28dde4281fe0d3c5f261b1649b247c2369d9e26b93 Graviton (ARM64): sha256:91930f8446f5f95b93b8ccb18773992affa401eb3f42da89d68077a56bafa6cd	一般的なパフォーマンスの調整と強化。	2024 年 5 月 31 日

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日
v1.1.0	x86_64 (AMD64): sha256:83ce3cf2ef85a349ed1797a8cf30a008ac5d8c9f673f2835823957e9dcf71657 Graviton (ARM64): sha256:0d4b61648d7bdeab8ab8d94684f805498927c7d437d318204dcccfe8c9383dc7	新しいセキュリティシグナルと検出結果をサポートします。 一般的なパフォーマンスの調整と強化。	2024 年 5 月 1 日

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日
v1.0.1	x86_64 (AMD64): sha256:9f8cd438fb66f62d09bfc641286439f7ed5177988a314a6021ef4ff880642e68 Graviton (ARM64): sha256:82c66bb615bd0d1e96db77b1f1fb51dc03220caa593b1962249571bf7147d1b7	一般的なパフォーマンスの調整と強化。	2024 年 1 月 26 日

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日
v1.0.0	x86_64 (AMD64): sha256:359b8b014e5076c625daa1056090e522631587a7afa3b2e055edda6bd1141017 Graviton (ARM64): sha256:b9438690fa8a86067180a11658bec0f4f838ae3fbd225d04b9306250648b3984	(AWS Fargate Amazon ECS のみ) の GuardDuty セキュリティエージェントの初回リリース。	2023 年 11 月 26 日

Amazon EKS リソースの GuardDuty セキュリティエージェントバージョン

GuardDuty は、更新されたエージェントバージョンを随時リリースします。GuardDuty がエージェントを自動的に管理する場合、GuardDuty はユーザーに代わってエージェントの更新を管理するように設計されています。エージェントを手動で管理する場合、Amazon EKS クラスターのエージェントバージョンを更新する責任がユーザーにあります。

エージェントを特定のバージョンに更新する前に、GuardDuty のイメージレジストリをアドミッションコントローラーの `allowed-container-registries` に追加します。詳細については、「Amazon ECR リポジトリホスティング GuardDuty エージェント」を参照してください。

次の表は、Amazon EKS アドオン GuardDuty エージェントのリリースバージョン履歴を示しています。

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.17.1	x86_64 (AMD64): sha256:772f754e9af50a63ab6af74ea24c12a5a3fb8a803c7cbc35ffaca3746c0540 Graviton (ARM64): sha256:e5531367e9ffa539e409e99bbf49e804f01fdd894ee9322858b9bca703cd4016	カーネル 6.19、7.0、7.1 のサポートが追加されました。 起動時に、エージェントは GuardDuty 接続タイプを検出し、link-local 経由で guardduty-data-v2 エンドポイントへの接続を試みます。 セキュリティ機能の一般的な更新。	2026 年 9 月 9 日	–
v1.16.0	x86_64 (AMD64): sha256:4ef4b5b5e6fce0a4b5a13922e65179293ee15a8812a2c7678b1c4b31e1f305 Graviton (ARM64): sha256:0a	カーネル 6.19、7.0、7.1 のサポートが追加されました。 セキュリティ機能の一般的な更新。	2026 年 7 月 17 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
	73c1d905a39db94f6ea37d98784dac84961e9199935fae20345de7aa3a57b7			
v1.15.0	x86_64 (AMD64): sha256:2642a5f9d9a316aa9100368acfdbae2481be7a5cfe9929571c1bb6e19c66fe Graviton (ARM64): sha256:f017e755615ae999fe5f163886f681508e8f53feea7e64e7ada733cdedb368d8	最適化されたイベント収集をサポートする新しいエージェント機能により、フィルターのデプロイが高速化され、検出速度が向上します。 セキュリティ機能の一般的な更新。	2026 年 5 月 14 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.12.2	x86_64 (AMD64): sha256:c25ab9fff7ee93048e5f34148c9627366323593719 9b4bd2e28b5ee2e5c080 Graviton (ARM64): sha256:f89962d971350ff6ab408cf6ec7bca9dd06152f0172b7f9c68ae770cc0fc4598	セキュリティ機能の一般的な更新。	2026 年 3 月 26 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.12.1	x86_64 (AMD64): sha256:9a597745e5acff48809bebac7d8091bbc38229453dd26710c549b817362491 Graviton (ARM64): sha256:089cebdc7e891177e107a099c9a4e362d9d74b5362cd374b448a16078040f6c6	一般的なパフォーマンス調整とセキュリティ機能の更新。	2025 年 11 月 17 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.11.0	x86_64 (AMD64): sha256:398fd50dee5fe493c361aa7fe191e094ddfa000c65b449a9ed6a5cd53610e7 Graviton (ARM64): sha256:98a547b47d4770f45e75eb9730c807d9265722ca2f391e0aa5cb19e487ab455f	Fedora 40 および Fedora 41 OS ディストリビューションのサポートが追加されました。詳細については「アーキテクチャ要件の検証 (EKS の場合)」を参照してください。 一般的なパフォーマンスの調整とセキュリティの強化。	2025 年 8 月 29 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.10.0	x86_64 (AMD64): sha256:cbe5b055e1ef0af903071ede0b08f755ad5b7e9774a67df5399efdaa1f3d7d Graviton (ARM64): sha256:f0536882268 9610a4bab543abf93d3e070b1b559e62a2e67d82dfa9837600f72	改善されたコンテナ ID の解決。 一般的なパフォーマンスの調整と強化。	2025 年 4 月 4 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.9.0	x86_64 (AMD64): sha256:c5789ef6570f9bec879ac48a8f4769718cbc31e45430032569917e219af63f Graviton (ARM64): sha256:9c2f74e7ea0827b7e422ae4c91fffc6c2bc41a1cdb96c7191d05259d337154e1	一般的なパフォーマンスの調整と強化。	2025 年 3 月 2 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.8.1	x86_64 (AMD64): sha256:ce8cf89dbe17e3388cecb35053544dadf21af7770545f8d4b50384076aff47 Graviton (ARM64): sha256:30f586e4b694e704bcafadfa9081ab0aeff3cfbcde39743a0f1e24f77d79627f	CentOS Stream 9.0、RedHat 9.4、Fedora 34.0、および Ubuntu 24.04 のサポートが追加されました。 `.../MetadataDNSRebind` 検出結果の ARM インスタンスのサポート。 一般的なパフォーマンスの調整と強化。	2024 年 11 月 23 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.7.1	x86_64 (AMD64): sha256:b86b5d0872c8b67fecf64ec3d17266636054 5435a1752447d510951a7fd749 Graviton (ARM64): sha256:40ac4cfc354fd430ba7897ca1632e 9a500ed13eeb0c315c5bcad38680e76b6e9	一般的なパフォーマンスの調整と強化。 将来の「GuardDuty Runtime Monitoring の検出結果タイプ」の追加セキュリティシグナルをキャプチャするためのサポートが含まれています。 カスタム DNS リゾルバーのサポート。	2024 年 9 月 13 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.7.0	x86_64 (AMD64): sha256:a2a8806e6c2a7fd63a91cccf6f7dffcd7e68554a423d610cea8c7e8f2185ec Graviton (ARM64): sha256:b1a6db35a072c0de3c695e5e909a03e6c4e1fdbe47ecfaeb2784435cf67ebe0a	一般的なパフォーマンスの調整と強化。 将来の「GuardDuty Runtime Monitoring の検出結果タイプ」の追加セキュリティシグナルをキャプチャするためのサポートが含まれています。	2024 年 8 月 17 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.6.1	x86_64 (AMD64): sha256:650708a6601f6d6b9046f54b30f5fd65af296b1e40b8c24426b9bdb07c3ab1 Graviton (ARM64): sha256:5f637c42ffb306b20f776d9d83e1e0b4be40ce245be44afcf43a8902b4d71019	一般的なパフォーマンスの調整と強化。	2024 年 5 月 14 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.6.0	x86_64 (AMD64): sha256:abcbee30d8b053676752fbc19e89f77272d9a6a53cc93731f5872180ef9010 Graviton (ARM64): sha256:9710f53afccdf4f22b265a1a6fc27f1469403af1f7d5d08c4869a7269cdd2650	• EKS/EC2 リソースの GuardDuty 自動エージェント設定をサポートします。 • 新しいセキュリティシグナルと検出結果をサポートします。詳細については、「GuardDuty が使用する収集されたランタイムイベントタイプ」および「GuardDuty Runtime Monitoring の検出結果タイプ」を参照してください。 • 一般的なパフォーマンスの調整と強化。	2024 年 4 月 29 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.5.0	x86_64 (AMD64): sha256:9a4e70af4058a212f172cc8eb3fc23ad9bed547ed609faa2bb82cf7cc5532d Graviton (ARM64): sha256:afc9a3f8f17ae12499d76069efcf1b46271a5a4b2b3f6ba5de54637b8f55d5c6	• 一般的なパフォーマンスの調整と強化。 • 「収集されたランタイムイベントタイプ」に記載された新しいイベントタイプを含むセキュリティの強化。 • CPU 使用率に関するパフォーマンスの向上。	2024 年 3 月 7 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.4.1	x86_64 (AMD64): sha256:d49192776 3742660fa a87cc2c39 bb97b7873 039157ae8 b90bc999c b73d0b9c Graviton (ARM64): sha256:53 7a330b2dd 82357024f b6daeb876 1034b7def d43b10dff e0792c9e6 d0778b40	一般的なパフォーマンスの調整と強化。	2024 年 1 月 16 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.4.0	x86_64 (AMD64): sha256:8ce13d9430bad554ac23d4699551505326ada2a88e1a721fe9f86b56b52c0f Graviton (ARM64): sha256:0c650aeafeeb5f2bcb8b989ac849bedc1fae1a4de1cf6306ffdd9c6aebe67f8e	マニフェストマウントポイントによるより優れたデータ収集のサポート マニフェストでの AppArmor 設定 コマンドライン引数の収集 一般的なパフォーマンスの調整と強化	2023 年 12 月 21 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.3.1	x86_64 (AMD64): sha256:578fcb7b73097ade5c8404390ef16cf76a7b568490abaae01ac75992b3ea29 Graviton (ARM64): sha256:e3ce8d66ac2121f8d476eb58f8bc50ab51336647615eb7cf514c21421cb818fd	重要なセキュリティパッチと更新。	2023 年 10 月 23 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.3.0	x86_64 (AMD64): sha256:ace2337dfbb7609811be89fb4b23ae0b865f1027ad78fbe69530fbd46c694 Graviton (ARM64): sha256:4928a7c6ef40e77c8ec95841323bb9a110db31f12c0ee7ab965e08b43efd01bb	Ubuntu プラットフォームをサポートしています Kubernetes バージョン 1.28 をサポートしています 一般的なパフォーマンスの向上と安定性の向上。	2023 年 10 月 5 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.2.0	x86_64 (AMD64): sha256:10413d662ec042057f05d6942496d7f2c08e9f5a077ea307ffdb5d3f11bcc3 Graviton (ARM64): sha256:174d7ab28b2f95e5309da80d95b88ad26f602dfe72c2b351a0ef9297a1412bfa	AMD64 ベースのインスタンスに加えて、v1.2.0 では ARM64 ベースのインスタンスもサポートされるようになりました。Bottlerocket のサポートが追加、検証されました Kubernetes バージョン 1.27 をサポートしています 一般的なパフォーマンスの向上と安定性の向上。	2023 年 6 月 16 日	–

エージェントのバージョン	コンテナイメージ	リリースノート	利用可能日	標準サポート終了日[1]
v1.1.0	sha256:b19ba3a3c1a508d153263ae2fda891a7928b5ca9b3a5692db6c101829303281c	GuardDuty セキュリティエージェントでサポートされている Kubernetes のバージョン に加え、このエージェントリリースは Kubernetes バージョン 1.26 もサポートしています 一般的なパフォーマンスの向上と安定性の向上。	2023 年 5 月 2 日	2024 年 5 月 14 日
v1.0.0	sha256:e38bdd2b1323e89113f1a31bd4bc8e5a8098525dd98e6981a28b9906b1e4411e	Amazon EKS アドオンエージェントの初回リリース。	2023 年 3 月 30 日	2024 年 5 月 14 日

[1] 標準サポートの終了が近づいている現在のエージェントバージョンを更新する方法については、「」を参照してくださいAmazon EKS リソースのセキュリティエージェントの手動更新。

その他のリソース - 次のステップ

次のステップの詳細については、次のトピックを参照してください。

- Runtime Monitoring を有効にする前提条件 - 新しいエージェントバージョンでは、前提条件セクションが更新される場合があります。リソースが最新の前提条件を満たしていることを確認します。
- GuardDuty セキュリティエージェントの管理 - エージェントを手動で管理する場合、リソース上で実行されているエージェントバージョンの更新管理はお客様の責任となります。リソースタイプ (Amazon EKS または Amazon EC2-Amazon ECS)に基づいて、セキュリティ エージェントを更新する手順を実行してください。また、VPC エンドポイント設定も必ず検証してください。
- ランタイムカバレッジ統計の確認と問題のトラブルシューティング - セキュリティエージェントを更新したら、リソースのランタイムカバレッジを評価できます。接続に関する問題が発生した場合は、関連するトラブルシューティング手順を参照してください。

Runtime Monitoring でのリソースの無効化、アンインストール、クリーンアップ

このセクションでは、Runtime Monitoring を無効にする AWS アカウント か、リソースタイプの GuardDuty 自動エージェント設定のみを無効にする場合は、 に適用されます。

GuardDuty 自動エージェント設定の無効化

GuardDuty は、リソースにデプロイされているセキュリティエージェントを削除しません。ただし、GuardDuty はセキュリティエージェントに対する更新の管理を停止します。

GuardDuty は、引き続きリソースタイプからランタイムイベントを受信します。使用統計に影響が出ないように、リソースから GuardDuty セキュリティエージェントを必ず削除してください。

が共有 VPC エンドポイント AWS アカウント を使用するかどうかにかかわらず、GuardDuty は VPC エンドポイントを削除しません。必要に応じて、VPC エンドポイントを手動で削除する必要があります。

Runtime Monitoring および EKS Runtime Monitoring の無効化

このセクションは、次のシナリオに適用されます。

- EKS Runtime Monitoring を個別に有効にしたことはないが、Runtime Monitoring を無効にした。
- Runtime Monitoring と EKS Runtime Monitoring の両方を無効にしている。EKS Runtime Monitoring の設定ステータスが不明な場合は、「EKS Runtime Monitoring 設定ステータスの確認」を参照してください。

> **EKS Runtime Monitoring を無効にせずに Runtime Monitoring を無効にする**
>
> このシナリオでは、ある時点で EKS Runtime Monitoring を有効にし、後で EKS Runtime Monitoring を無効にすることなく Runtime Monitoring も有効にしました。ここで Runtime Monitoring を無効にした場合、EKS Runtime Monitoring も無効にする必要があります。無効にしないと、EKS Runtime Monitoring の使用コストが引き続き発生します。

前述のシナリオが当てはまる場合、GuardDuty はアカウントで次のアクションを実行します。

- GuardDuty は、`GuardDutyManaged:true` タグを持つ VPC エンドポイントを削除します。これは、GuardDuty が自動セキュリティエージェントを管理するために作成した VPC です。
- GuardDuty は、`GuardDutyManaged:true` としてタグ付けされたセキュリティグループを削除します。
- 少なくとも 1 つの参加者アカウントで使用されている共有 VPC の場合、GuardDuty は VPC エンドポイントも共有 VPC リソースに関連付けられたセキュリティグループも削除しません。
- Amazon EKS リソースの場合、GuardDuty はセキュリティエージェントを削除します。これは、手動で管理されたか、GuardDuty を通じて管理されたかとは関係ありません。

  Amazon ECS リソースの場合、ECS タスクはイミュータブルであるため、GuardDuty はそのリソースからセキュリティエージェントをアンインストールできません。これは、セキュリティエージェントの管理方法 (手動管理または GuardDuty による自動管理) とは関係ありません。Runtime Monitoring を無効にすると、GuardDuty は新しい ECS タスクの実行開始時にサイドカーコンテナをアタッチしません。Fargate-ECS タスクの操作については、「Fargate での Amazon ECS へのセキュリティエージェントのデプロイ」を参照してください。

  Amazon EC2 リソースの場合、GuardDuty は、次の条件を満たす場合にのみ、すべての Systems Manager (SSM) マネージド Amazon EC2 インスタンスからセキュリティエージェントをアンインストールします。

  - リソースに `GuardDutyManaged:false` 除外タグが付けられていません。
  - GuardDuty には、インスタンスメタデータ内のタグにアクセスするアクセス許可が必要です。この EC2 リソースでは、[インスタンスメタデータのタグへのアクセス] は [許可] に設定されます。

セキュリティエージェントの管理を手動で停止する場合

GuardDuty セキュリティエージェントのデプロイと管理にどのアプローチを使用するかに関係なく、リソース内のランタイムイベントのモニタリングを停止するには、GuardDuty セキュリティエージェントを削除する必要があります。アカウント内のリソースタイプからのランタイムイベントのモニタリングを停止する場合は、Amazon VPC エンドポイントを削除することもできます。

Amazon EC2 リソースのセキュリティエージェントの手動アンインストール

このセクションでは、Amazon EC2 リソースから GuardDuty セキュリティエージェントをアンインストールする方法について説明します。セキュリティエージェントを手動で管理する場合は、リソースからエージェントを削除する責任があります。GuardDuty は、管理するリソースに対してアクションを実行しません。

Amazon VPC エンドポイントを手動で作成した場合、アカウント内のすべてのモニタリング対象のリソースタイプでセキュリティエージェントをアンインストールした後、VPC エンドポイントを削除できます。これは別のステップです。詳細については、「To delete a VPC endpoint」を参照してください。

リソースにセキュリティエージェントをインストールした方法に応じて、次のいずれかの方法を選択してアンインストールします。

トピック

- 方法 1 - Run Command を使用する
- 方法 2 - Linux パッケージマネージャーを使用する

方法 1 - Run Command を使用する

「方法 1 - の使用 AWS Systems Manager」を使用してセキュリティエージェントをインストールした場合、次のステップを実行してエージェントをアンインストールします。

GuardDuty セキュリティエージェントをアンインストールするには

1. GuardDuty セキュリティエージェントは、「AWS Systems Manager ユーザーガイド」の「AWS Systems Manager Run Command」に記載されている手順に従ってアンインストールで

きます。パラメータの [Uninstall] アクションを使用して、GuardDuty セキュリティエージェントをアンインストールします。

[ターゲット] セクションで、セキュリティエージェントをアンインストールする Amazon EC2 インスタンスにのみ影響があることを確認してください。

次の GuardDuty ドキュメントとディストリビュータを使用してください。

- ドキュメント名: `AmazonGuardDuty-ConfigureRuntimeMonitoringSsmPlugin`
- ディストリビューター: `AmazonGuardDuty-RuntimeMonitoringSsmPlugin`

2. すべての詳細を入力した後、[実行する] を選択すると、対象の Amazon EC2 インスタンスにデプロイされたセキュリティエージェントが削除されます。

   Amazon VPC エンドポイント設定を削除するには、Runtime Monitoring と Amazon EKS Runtime Monitoring の両方を無効にする必要があります。

3. このセキュリティエージェントに関連付けられている VPC エンドポイントも削除する場合は、「To delete a VPC endpoint」を参照してください。

方法 2 - Linux パッケージマネージャーを使用する

「方法 2 - Linux パッケージマネージャーを使用する」を使用してセキュリティエージェントをインストールした場合、次のステップを実行してエージェントをアンインストールします。

GuardDuty セキュリティエージェントをアンインストールするには

1. インスタンスに接続します。これを行うステップについては、「Amazon EC2 ユーザーガイド」の「SSH クライアントを使用して Linux インスタンスに接続する」を参照してください。
2. アンインストールするためのコマンド

   次のコマンドは、接続先の Amazon EC2 インスタンスから GuardDuty セキュリティエージェントをアンインストールします。

   - RPM の場合:

     ```
     sudo rpm -e amazon-guardduty-agent
     ```

   - Debian の場合:

```
sudo dpkg --purge amazon-guardduty-agent
```

コマンドを実行した後、コマンドに関連付けられたログを確認することもできます。

3. このセキュリティエージェントに関連付けられている VPC エンドポイントも削除する場合は、「To delete a VPC endpoint」を参照してください。

ECS-EC2 Bottlerocket リソースのセキュリティエージェントのアンインストール

Bottlerocket ECS-EC2 インスタンスから GuardDuty セキュリティエージェントをアンインストールするには、次の手順を実行します。

GuardDuty セキュリティエージェントをアンインストールするには

1. GuardDuty セキュリティエージェントをアンインストールするには、AWS Systems Manager Run Command (AWS Systems Manager ユーザーガイド) を使用し、 アクションパラメータを Uninstall に設定します。

   ターゲットセクションで、影響がセキュリティエージェントをアンインストールする Bottlerocket インスタンスにのみ影響することを確認します。

   次の GuardDuty ドキュメントとディストリビュータを使用してください。

   - ドキュメント名: `AmazonGuardDuty-ConfigureRuntimeMonitoringSsmPlugin`。
   - ディストリビューター: `AmazonGuardDuty-RuntimeMonitoringSsmPlugin`。
2. すべての詳細を指定して Run を選択すると、GuardDuty はターゲットの Bottlerocket インスタンスからセキュリティエージェントを削除します。
3. このセキュリティエージェントに関連付けられている VPC エンドポイントも削除するには、「」を参照してくださいTo delete a VPC endpoint。

セキュリティエージェントのリソースのクリーンアップ

このセクションでは、セキュリティエージェントに関連付けられた AWS リソースをクリーンアップする方法について説明します。「無効化、アンインストール、リソースクリーンアップ」に記載され

ているように、GuardDuty はすべてのセキュリティエージェントリソースを削除または削除しません。次のセクションでは、セキュリティエージェントリソースを削除する方法について説明します。

Amazon VPC エンドポイントを削除するには

セキュリティエージェントを手動で管理する場合、Amazon VPC エンドポイントを手動で作成した可能性があります。アカウント内のすべてのモニタリング対象リソースのセキュリティエージェントをアンインストールした後、この VPC エンドポイントを削除できます。

次のリストは、共有 VPC を使用する場合と、共有 VPC を使用しない場合のシナリオを示しています。

- 共有 VPC を使用しない場合 – アカウントのリソースをモニタリングする必要がなくなった場合は、Amazon VPC エンドポイントの削除を検討してください。
- 共有 VPC を使用する場合 – 共有 VPC 所有者アカウントが、まだ使用されている共有 VPC リソースを削除すると、共有 VPC 所有者アカウントと参加アカウントのリソースの Runtime Monitoring (および該当する場合は EKS Runtime Monitoring) カバレッジステータスが異常になる可能性があります。カバレッジステータスについては、「ランタイムカバレッジ統計の確認と問題のトラブルシューティング」を参照してください。

VPC エンドポイントを削除するには、「AWS PrivateLink ガイド」の「インターフェースエンドポイントを削除する」を参照してください。

セキュリティグループを削除するには

- 共有 VPC を使用しない場合 – アカウントのリソースタイプをモニタリングする必要がなくなった場合は、Amazon VPC に関連付けられたセキュリティグループの削除を検討してください。
- 共有 VPC を使用する場合 – 共有 VPC 所有者アカウントがセキュリティグループを削除すると、共有 VPC に関連付けられたセキュリティグループを現在使用している参加者アカウント、共有 VPC 所有者アカウントおよび参加アカウントのリソースの Runtime Monitoring カバレッジステータスが異常になる可能性があります。詳細については、「ランタイムカバレッジ統計の確認と問題のトラブルシューティング」を参照してください。

ステップの詳細については、「Amazon EC2 ユーザーガイド」の「Amazon EC2 セキュリティグループの削除」を参照してください。

EKS クラスターから GuardDuty セキュリティエージェントを削除するには

モニタリングする必要がなくなった EKS クラスターからセキュリティエージェントを削除するには、「Amazon EKS ユーザーガイド」の「クラスターからの Amazon EKS アドオンの削除」を参照してください。

EKS アドオンエージェントを削除しても、EKS クラスターから amazon-guardduty 名前空間は削除されません。amazon-guardduty 名前空間を削除するには、「名前空間の削除」を参照します。

amazon-guardduty 名前空間を削除するには (EKS クラスター)

自動エージェント設定を無効にしても、EKS クラスターから amazon-guardduty 名前空間は自動的に削除されません。amazon-guardduty 名前空間を削除するには、「名前空間の削除」を参照します。

GuardDuty Malware Protection for EC2

Malware Protection for EC2 は、Amazon EC2 で実行されている Amazon Elastic Compute Cloud (Amazon EC2) インスタンスおよびコンテナワークロードにアタッチされた Amazon Elastic Block Store (Amazon EBS) ボリュームをスキャンすることで、マルウェアの潜在的な存在を検出することに役立ちます。Malware Protection for EC2 は、スキャン時に特定の Amazon EC2 インスタンスを含めるか除外するかを決定できるスキャンオプションを提供します。Amazon EC2 インスタンスまたはコンテナワークロードにアタッチされた Amazon EBS ボリュームのスナップショットを GuardDuty アカウントに保持するオプションも提供します。スナップショットは、マルウェアが検出されて Malware Protection for EC2 の検出結果が生成された場合にのみ保持されます。

Malware Protection for EC2 リソースのパフォーマンスに影響を与えないように設計されています。GuardDuty 内で Malware Protection for EC2 が機能する仕組みについては、「GuardDuty が EBS ボリュームをスキャンしてマルウェアを検出する方法」を参照してください。Malware Protection for EC2 のさまざまな での可用性については AWS リージョン、「」を参照してくださいリージョンとエンドポイント。

> **注意事項**
>
> Malware Protection for EC2 は、Amazon EKS Auto Mode のマネージドインスタンスでのマルウェアスキャンをサポートしています。詳細については、「Guardduty でのマネージドインスタンスのサポート」を参照してください。
> Malware Protection for EC2 は、Amazon EKS または Amazon ECS のいずれかで で実行されているワークロードのマルウェアスキャンをサポートしていません。 AWS Fargate
> これらの Amazon EKS の詳細については、「Amazon EKS ユーザーガイド」の「Amazon EKS とは」を参照してください。

トピック

- GuardDuty 実行型マルウェアスキャンとオンデマンドマルウェアスキャンの比較
- GuardDuty が EBS ボリュームをスキャンしてマルウェアを検出する方法
- マルウェアスキャンでサポートされている Amazon EBS ボリューム
- スナップショット保持と EC2 スキャンカバレッジを設定する
- GuardDuty 実行型マルウェアスキャン
- GuardDuty でのオンデマンドマルウェアスキャン

- Malware Protection for EC2 のスキャンステータスと結果のモニタリング
- による GuardDuty サービスアカウント AWS リージョン
- Malware Protection for EC2 のクォータ

GuardDuty 実行型マルウェアスキャンとオンデマンドマルウェアスキャンの比較

Malware Protection for EC2 は、Amazon EC2 インスタンスおよびコンテナワークロード内の潜在的に悪意のあるアクティビティを検出するために、GuardDuty 実行型マルウェアスキャンおよびオンデマンドマルウェアスキャンという 2 つのタイプのスキャンを提供します。次の表は、両方のスキャンタイプの比較を示しています。

Factor	GuardDuty 実行型マルウェアスキャン	オンデマンドのマルウェアスキャン
スキャンの起動方法	GuardDuty 実行型マルウェアスキャンを有効にすると、GuardDuty が Amazon EC2 インスタンスまたはコンテナワークロードにマルウェアの潜在的な存在を示す検出結果を生成するたびに、GuardDuty が影響を受けた可能性のあるリソースにアタッチされた Amazon EBS ボリュームにエージェントレスマルウェアスキャンを自動的に開始します。詳細については、「GuardDuty 実行型マルウェアスキャン」を参照してください。	Amazon EC2 インスタンスの Amazon リソースネーム (ARN) を指定すると、オンデマンドマルウェアスキャンを開始できます。リソースに GuardDuty の検出結果が生成されない場合でも、オンデマンドのマルウェアスキャンを開始できます。詳細については、「GuardDuty でのオンデマンドマルウェアスキャン」を参照してください。
設定が必要です	GuardDuty 実行型マルウェアスキャンを使用するには、アカウントで有効にする必要があります。 AWS Organizat	アカウントで GuardDuty が有効になっている必要があります。オンデマンドマルウェアスキャンを使用するには、機

Factor	GuardDuty 実行型マルウェアスキャン	オンデマンドのマルウェアスキャン
	ions または招待ベースのメソッドを使用して複数のアカウントを管理するには、「」を参照してくださいマルチアカウント環境での GuardDuty 実行型マルウェアスキャンの有効化。自分のアカウントで GuardDuty 実行型マルウェアスキャンを有効にするには、「スタンドアロンアカウントでの GuardDuty 実行型マルウェアスキャンの有効化」を参照してください。	能レベルでの設定は必要ありません。
新しいスキャンを開始するまでの待ち時間	GuardDuty がいずれかの GuardDuty 実行型マルウェアスキャンを起動する検出結果を生成するたびに、24 時間に 1 回だけマルウェアスキャンが自動的に開始されます。	前回のスキャンの開始時刻から 1 時間後であれば、いつでも同じリソースに対してオンデマンドマルウェアスキャンを開始できます。
30 日間の無料トライアル期間の利用可能性[1]	アカウントで初めて GuardDuty 実行型マルウェアスキャンを有効にする場合、30 日間の無料トライアル期間を使用できます。 詳細については、「GuardDuty 実行型マルウェアスキャンの 30 日間の無料トライアル」を参照してください。	新規または既存の GuardDuty アカウントのオンデマンドマルウェアスキャンには、無料トライアル期間はありません。

Factor	GuardDuty 実行型マルウェアスキャン	オンデマンドのマルウェアスキャン
スキャンオプション[2]	GuardDuty 実行型マルウェアスキャンを設定した後、Malware Protection for EC2 では、タグを使用して特定の Amazon EC2 リソースをスキャンまたはスキップするオプションが提供されます。Malware Protection for EC2 は、スキャンから除外するように選択したリソースに対して自動スキャンを開始しません。詳細については、「ユーザー定義タグ付きのスキャンオプション」を参照してください。	オンデマンドマルウェアスキャンを手動で開始するためのリソース ARN を指定するため、「ユーザー定義タグ付きのスキャンオプション」の使用は適用できません。

[1]EBS ボリュームスナップショットの作成およびスナップショットの保持には使用料が発生します。スナップショットを保持するようにアカウントを設定する方法の詳細については、「スナップショットの保持」を参照してください。

[2]GuardDuty 実行型マルウェアスキャンと、グローバルタグを使用したオンデマンドマルウェアスキャンのサポートの両方により、Amazon EC2 リソースがマルウェアスキャンから除外されます。詳細については、「グローバル `GuardDutyExcluded` タグ」を参照してください。

GuardDuty が EBS ボリュームをスキャンしてマルウェアを検出する方法

このセクションでは、GuardDuty 実行型マルウェアスキャンおよびオンデマンドマルウェアスキャンの両方を含め、Malware Protection for EC2 が Amazon EC2 インスタンスおよびコンテナワークロードに関連付けされた Amazon EBS ボリュームをスキャンする方法について説明します。先に進む前に、次のカスタマイズを検討してください。

- スキャンオプション - Malware Protection for EC2 では、Amazon EC2 インスタンスおよび Amazon EBS ボリュームをスキャンプロセスに含めるまたは除外するタグを指定できます。GuardDuty 実行型マルウェアスキャンのみが、ユーザー定義タグ付けされたスキャンオプションをサポートしています。GuardDuty 実行型マルウェアスキャンおよびオンデマンドのマルウェアスキャンの両方はグローバル `GuardDutyExcluded` タグをサポートしています。詳細については、「ユーザー定義タグ付きのスキャンオプション」を参照してください。
- スナップショットの保持 – Malware Protection for EC2 には、Amazon EBS ボリュームのスナップショットを AWS アカウント内に保持するオプションが用意されています。この設定は、デフォルトでオフになっています。GuardDuty によるマルウェアスキャンとオンデマンドのマルウェアスキャンの両方でスナップショット保持を選択できます。詳細については、「スナップショットの保持」を参照してください。

GuardDuty が 1 つ以上の「GuardDuty 実行型マルウェアスキャンを起動する検出結果」を生成すると、このアクティビティが GuardDuty がマルウェアスキャンを開始する理由になります。スキャンオプションがこのインスタンスを除外しない場合、GuardDuty はスキャンを開始します。

Amazon EC2 インスタンスに関連付けされた Amazon EBS ボリュームにオンデマンドのマルウェアスキャンを開始するには、Amazon EC2 インスタンスの Amazon リソースネーム (ARN) を指定します。

オンデマンドマルウェアスキャンまたは GuardDuty 実行型自動マルウェアスキャンが開始された場合の応答として、GuardDuty は影響を受けた可能性のあるリソースにアタッチされた関連する EBS ボリュームのスナップショットを作成し、「GuardDuty サービスアカウント」と共有します。GuardDuty が EBS ボリュームのスナップショットを作成すると、`GuardDutyScanId` というデフォルトのタグが追加されます。このタグは、GuardDuty がスナップショットにアクセスするのに役立ちます。このタグは削除しないでください。GuardDuty は、それらのスナップショットを元に、サービスアカウントで暗号化された EBS ボリュームのレプリカを作成します。

スキャンが完了すると、GuardDuty は暗号化された EBS ボリュームのレプリカと EBS ボリュームのスナップショットを削除します。デフォルトでは、スナップショットの保持設定はオフになっています。ただし、スキャン結果や設定に関係なく、Amazon EBS スナップショットのロックが有効になっている場合、スナップショットは保持されます。GuardDuty は Amazon EBS スナップショットのロック設定を変更できません。

次のリストでは、EBS スナップショットのロックに関係なく、スナップショットの保持動作について説明します。

スナップショットの保持がオンになっています。

- マルウェアが検出されると、GuardDuty はスナップショットを に保持します AWS アカウント。
- マルウェアが見つからない場合、GuardDuty はスナップショットをロックしない限り保持しません。

スナップショットの保持設定はオフになっています (デフォルト設定)。

- マルウェアが検出されたか否かにかかわらず、スナップショットは保持されません。
- GuardDuty はロックされた Amazon EBS スナップショットを削除できません。

GuardDuty は、各レプリカ EBS ボリュームをサービスアカウントに最大 55 時間保持します。サービスの停止、またはレプリカ EBS ボリュームとそのマルウェアスキャンでの障害が発生した場合、GuardDuty はそのような EBS ボリュームを 7 日間以下の期間にわたって保持します。ボリューム保持期間が延長されている間に、サービスの停止または障害のトリアージと対処をしてください。GuardDuty Malware Protection for EC2 は、停止または障害に対処した後、または延長された保持期間が経過すると、サービスアカウントから EBS ボリュームのレプリカを削除します。

GuardDuty マルウェア検出方法と使用するスキャンエンジンの詳細については、「GuardDuty マルウェア検出のスキャンエンジン」を参照してください。

マルウェアスキャンでサポートされている Amazon EBS ボリューム

GuardDuty AWS リージョン が Malware Protection for EC2 機能をサポートしているすべての で、暗号化されていない、または暗号化されていない Amazon EBS ボリュームをスキャンできます。Amazon EBS ボリュームは、AWS マネージドキー または カスタマーマネージドキーで暗号化できます。現在、Malware Protection for EC2 が利用可能なリージョンの中には、Amazon EBS ボリュームを暗号化する方法の両方をサポートするものもあれば、カスタマーマネージドキーのみをサポートするものもあります。サポートされるリージョンの詳細については、「による GuardDuty サービスアカウント AWS リージョン」を参照してください。GuardDuty は使用できるが Malware Protection for EC2 は使用できないリージョンについては、「リージョン固有機能の可用性」を参照してください。

次のリストは、Amazon EBS ボリュームが暗号化されているかどうかに関係なく、GuardDuty が使用するキーを示しています。

- 暗号化されていないか、 AWS マネージドキーで暗号化されている Amazon EBS ボリューム – GuardDuty は独自のキーを使用してレプリカ Amazon EBS ボリュームを暗号化します。

  デフォルトで Amazon EBS 暗号化を使用して暗号化された Amazon EBS ボリュームのスキャンをリージョンがポートしていない場合は、デフォルトキーをカスタマーマネージドキーに変更する必要があります。これにより、GuardDuty はこれらの EBS ボリュームにアクセスできます。キーを変更することで、将来の EBS ボリュームでも更新されたキーを使用して作成され、GuardDuty がマルウェアスキャンをサポートできるようになります。デフォルトキーを変更するステップについては、次のセクションの「Amazon EBS ボリュームのデフォルト AWS KMS キー ID を変更する」を参照してください。
- カスタマーマネージドキーで暗号化された Amazon EBS ボリューム – GuardDuty は同じキーを使用してレプリカ EBS ボリュームを暗号化します。サポートされている AWS KMS 暗号化関連ポリシーについては、「」を参照してくださいMalware Protection for EC2 のためのサービスにリンクされたロールの許可。

Amazon EBS ボリュームのデフォルト AWS KMS キー ID を変更する

Amazon EBS 暗号化を使用して Amazon EBS ボリュームを作成し、 AWS KMS キー ID を指定しない場合、Amazon EBS ボリュームは暗号化用のデフォルトキーで暗号化されます。デフォルトで暗号化を有効にすると、Amazon EBS は Amazon EBS 暗号化のデフォルト KMS キーを使用して、新しいボリュームとスナップショットを自動的に暗号化します。

デフォルト暗号化キーを変更し、Amazon EBS 暗号化にカスタマーマネージドキーを使用できます。これにより、GuardDuty はこれらの Amazon EBS ボリュームにアクセスできます。EBS デフォルトキー ID を変更するには、次の必要な許可を IAM ポリシーに追加します - `ec2:modifyEbsDefaultKmsKeyId`。暗号化することを選択し、関連付けられた KMS キー ID を指定しない新しく作成された Amazon EBS ボリュームは、デフォルトキー ID を使用します。次のいずれかの方法を使用して、EBS デフォルトキー ID を更新します。

Amazon EBS ボリュームのデフォルト KMS キー ID を変更するには

次のいずれかを行います:

- API の使用 - ModifyEbsDefaultKmsKeyId API を使用できます。ボリュームの暗号化ステータスを表示する方法については、「Amazon EBS ボリュームの作成」を参照してください。
- AWS CLI コマンドの使用 – 次の例では、KMS キー ID を指定しない場合に Amazon EBS ボリュームを暗号化するデフォルトの KMS キー ID を変更します。リージョンを KM キー ID AWS リージョン の に置き換えてください。

```
aws ec2 modify-ebs-default-kms-key-id --region us-west-2 --kms-key-
id AKIAIOSFODNN7EXAMPLE
```

上記のコマンドは、次の出力と同様な出力を生成します。

```
{
  "KmsKeyId": "arn:aws:kms:us-west-2:444455556666:key/AKIAIOSFODNN7EXAMPLE"
}
```

詳細については、「modify-ebs-default-kms-key-id」を参照してください。

スナップショット保持と EC2 スキャンカバレッジを設定する

このセクションでは、Amazon EC2 インスタンスのマルウェアスキャンオプションをカスタマイズする方法について説明します。これらのカスタマイズは、オンデマンドマルウェアスキャンと GuardDuty によって開始されたマルウェアスキャンの両方に適用されます。以下を行うことができます。

- スナップショット保持を有効にする – スキャン前に有効にすると、GuardDuty は悪意があると検出した Amazon EBS スナップショットを保持します。
- スキャン対象の Amazon EC2 インスタンスを選択する – タグを使用して、マルウェアスキャンから特定の Amazon EC2 インスタンスを含めるか除外します。

スナップショットの保持

GuardDuty は、 AWS アカウントで EBS ボリュームのスナップショットを保持するオプションを提供します。デフォルトでは、スナップショットの保持設定はオフになっています。スナップショットは、スキャン開始前にこの設定をオンにしている場合にのみ保持されます。

スキャンが開始されると、GuardDuty は EBS ボリュームのスナップショットに基づいて EBS ボリュームのレプリカを生成します。スキャンが完了して、アカウントのスナップショットの保持設定が既にオンになっている場合、マルウェアが見つかって Malware Protection for EC2 の検出結果のタイプ が生成された場合に限り、EBS ボリュームのスナップショットが保持されます。マルウェアが見つからない場合、スナップショットの設定に関係なく、GuardDuty は作成されたスナップショットで Amazon EBS スナップショットのロックが有効になっていない限り、EBS ボリュームのスナップショットを自動的に削除します。

スナップショットの使用コスト

マルウェアスキャン中に GuardDuty が Amazon EBS ボリュームのスナップショットを作成するため、このステップに関連した使用コストが発生します。アカウントのスナップショット保持設定を有効にした場合、マルウェアが検出されてスナップショットが保持されたとき、同様な内容で使用コストが発生します。スナップショットおよびその保持にかかるコストについては、「Amazon EBS 料金」を参照してください。

組織メンバーアカウントに代わってこの更新を行うことができるのは、委任 GuardDuty 管理者アカウントのみです。ただし、メンバーアカウントが招待方法によって管理されている場合、メンバーアカウントは自分でこの変更を行うことができます。詳細については、「管理者アカウントとメンバーアカウントの関係」を参照してください。

任意のアクセス方法を選択して、スナップショットの保存設定を有効にします。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインの [保護プラン] で、[Malware Protection for EC2] を選択します。
3. コンソールの下部のセクションで [General settings] (一般設定) を選択します。スナップショットを保持するには、[Snapshots retention] (スナップショットの保持) をオンにします。

API/CLI

UpdateMalwareScanSettings を実行して、スナップショット保持設定の現在の設定を更新します。

または、次の AWS CLI コマンドを実行して、GuardDuty Malware Protection for EC2 が検出結果を生成するときにスナップショットを自動的に保持することもできます。

必ず *detector-id* をご自身の有効な detectorId に置き換えてください。

アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-malware-scan-settings --detector-
id 60b87779336485625 54d637e0e4bb3b2 --ebs-snapshot-preservation
 "RETENTION_WITH_FINDING"
```

スナップショットの保持を無効にしたい場合は、RETENTION_WITH_FINDING を NO_RETENTION に置き換えます。

ユーザー定義タグ付きのスキャンオプション

GuardDuty 実行型マルウェアスキャンを使用することで、スキャンおよび脅威検出プロセスから Amazon EC2 インスタンスおよび Amazon EBS ボリュームを含めるまたは除外するタグも指定できます。包含タグリストまたは除外タグリストのタグを編集することで、GuardDuty 実行型マルウェアスキャンをそれぞれカスタマイズできます。各リストには、最大 50 個のタグを含めることができます。

EC2 リソースにまだユーザー定義タグが関連付けされていない場合、「Amazon EC2 ユーザーガイド」の「Amazon EC2 リソースのタグ付け」を参照してください。

> **Note**
>
> オンデマンドのマルウェアスキャンは、ユーザー定義タグ付けされたスキャンオプションをサポートしません。グローバル GuardDutyExcluded タグ をサポートします。

EC2 インスタンスをマルウェアウェアスキャンから除外する方法

スキャンプロセス中に Amazon EC2 インスタンスまたは Amazon EBS ボリュームを除外する場合、任意の Amazon EC2 インスタンスまたは Amazon EBS ボリュームに GuardDutyExcluded タグを true に設定すると、GuardDuty はスキャンを実行しません。GuardDutyExcluded タグの詳細については、「Malware Protection for EC2 のためのサービスにリンクされたロールの許可」を参照してください。Amazon EC2 インスタンスタグを除外リストに追加することもできます。複数のタグを除外タグリストに追加する場合、これらのタグを少なくとも 1 つ含む Amazon EC2 インスタンスがマルウェアスキャンのプロセスから除外されます。

組織メンバーアカウントに代わってこの更新を行うことができるのは、委任 GuardDuty 管理者アカウントのみです。ただし、メンバーアカウントが招待方法によって管理されている場合、メンバーアカウントは自分でこの変更を行うことができます。詳細については、「管理者アカウントとメンバーアカウントの関係」を参照してください。

任意のアクセス方法を選択して、Amazon EC2 インスタンスに関連付けされたタグを除外リストに追加します。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインの [保護プラン] で、[Malware Protection for EC2] を選択します。
3. [包含/除外タグ] セクションを展開します。[Add tags (タグの追加)] を選択します。
4. [除外タグ] を選択したら [確認] を選択します。
5. 除外するタグの **Key** および **Value** ペアを指定します。**Value** を指定するかどうかは任意です。すべてのタグを追加したら、[保存] を選択します。

> **Important**
>
> タグキーと値は大文字と小文字が区別されます。詳細については、「Amazon EC2 ユーザーガイド」の「タグ付けの制限」を参照してください。

キーの値が指定されずに EC2 インスタンスが指定されたキーでタグ付けされている場合、タグに割り当てられた値とは関係なく、この EC2 インスタンスは GuardDuty 実行型マルウェアスキャンのスキャンプロセスから除外されます。

API/CLI

EC2 インスタンスまたはコンテナワークロードをスキャンプロセスから除外することによって、UpdateMalwareScanSettings を実行します。

次のコマンド AWS CLI 例では、除外タグリストに新しいタグを追加します。*detector-id* の例を、自分の有効な `detectorId` に置き換えてください。

`MapEquals` は `Key/Value` ペアの一覧です。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-malware-scan-settings --detector-
id 60b87779336485625554d637e0e4bb3b2 --scan-resource-criteria '{"Exclude":
 {"EC2_INSTANCE_TAG" : {"MapEquals": [{ "Key": "TestKeyWithValue", "Value":
 "TestValue" }, {"Key":"TestKeyWithoutValue"} ]}}}' --ebs-snapshot-preservation
 "RETENTION_WITH_FINDING"
```

> **Important**
>
> タグキーと値は大文字と小文字が区別されます。詳細については、「Amazon EC2 ユーザーガイド」の「タグ付けの制限」を参照してください。

EC2 インスタンスをマルウェアスキャンに含めるには

EC2 インスタンスをスキャンする場合は、そのタグを包含リストに追加します。包含タグのリストにタグを追加すると、追加されたタグを含まない EC2 インスタンスは、マルウェアスキャン時にスキップされます。複数のタグを包含タグのリストに追加した場合、これらのタグを 1 つでも含む EC2 インスタンスには、マルウェアスキャンが実施されます。場合によっては、他の理由により、スキャンプロセス中に EC2 インスタンスがスキップされることがあります。詳細については、「マルウェアスキャン中にリソースをスキップする理由」を参照してください。

組織メンバーアカウントに代わってこの更新を行うことができるのは、委任 GuardDuty 管理者アカウントのみです。ただし、メンバーアカウントが招待方法によって管理されている場合、メンバーアカウントは自分でこの変更を行うことができます。詳細については、「管理者アカウントとメンバーアカウントの関係」を参照してください。

任意のアクセス方法を選択して、EC2 インスタンスに関連付けられているタグを包含リストに追加します。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインの [保護プラン] で、[Malware Protection for EC2] を選択します。
3. [包含/除外タグ] セクションを展開します。[Add tags (タグの追加)] を選択します。
4. [包含タグ] を選択したら [確認] を選択します。
5. [Add new inclusion tag] (新しい包含タグを追加) を選択し、除外するタグの **`Key`** と **`Value`** ペアを指定します。**`Value`** を指定するかどうかは任意です。

   すべての包含タグを追加したら、[保存] を選択します。

   キーの値が指定されずに EC2 インスタンスが指定されたキーでタグ付けされている場合、タグに割り当てられた値とは関係なく、EC2 インスタンスは Malware Protection for EC2 のスキャンプロセスに含まれます。

API/CLI

- UpdateMalwareScanSettings を実行して、スキャンプロセスに EC2 インスタンスまたはコンテナワークロードを含めます。

  次のコマンド AWS CLI 例では、包含タグリストに新しいタグを追加します。必ず *detector-id* の例をお使いの有効な detectorId に置き換えてください。*TestKey* および *TestValue* の例を EC2 リソースに関連付けされたタグの Key と Value ペアに置き換えます。

  MapEquals は Key/Value ペアの一覧です。

  アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-malware-scan-settings --detector-
id 60b87779336485625 54d637e0e4bb3b2 --scan-resource-criteria '{"Include":
 {"EC2_INSTANCE_TAG" : {"MapEquals": [{ "Key": "TestKeyWithValue", "Value":
 "TestValue" }, {"Key":"TestKeyWithoutValue"} ]}}}' --ebs-snapshot-preservation
 "RETENTION_WITH_FINDING"
```

  Important

  タグキーと値は大文字と小文字が区別されます。詳細については、「Amazon EC2 ユーザーガイド」の「タグ付けの制限」を参照してください。

Note

GuardDuty が新しいタグを検出するまでに、最長で 5 分かかることがあります。

ユーザーはいつでも [Inclusion tags] (包含タグ) または [Exclusion tags] (除外タグ) のどちらかを選択できますが、両方を選択することはできません。タグを切り替える場合、新しいタグを追加する際にドロップダウンメニューからそのタグを選択し、選択したものを [確認] します。このアクションにより、現在のタグがすべてクリアされます。

グローバル GuardDutyExcluded タグ

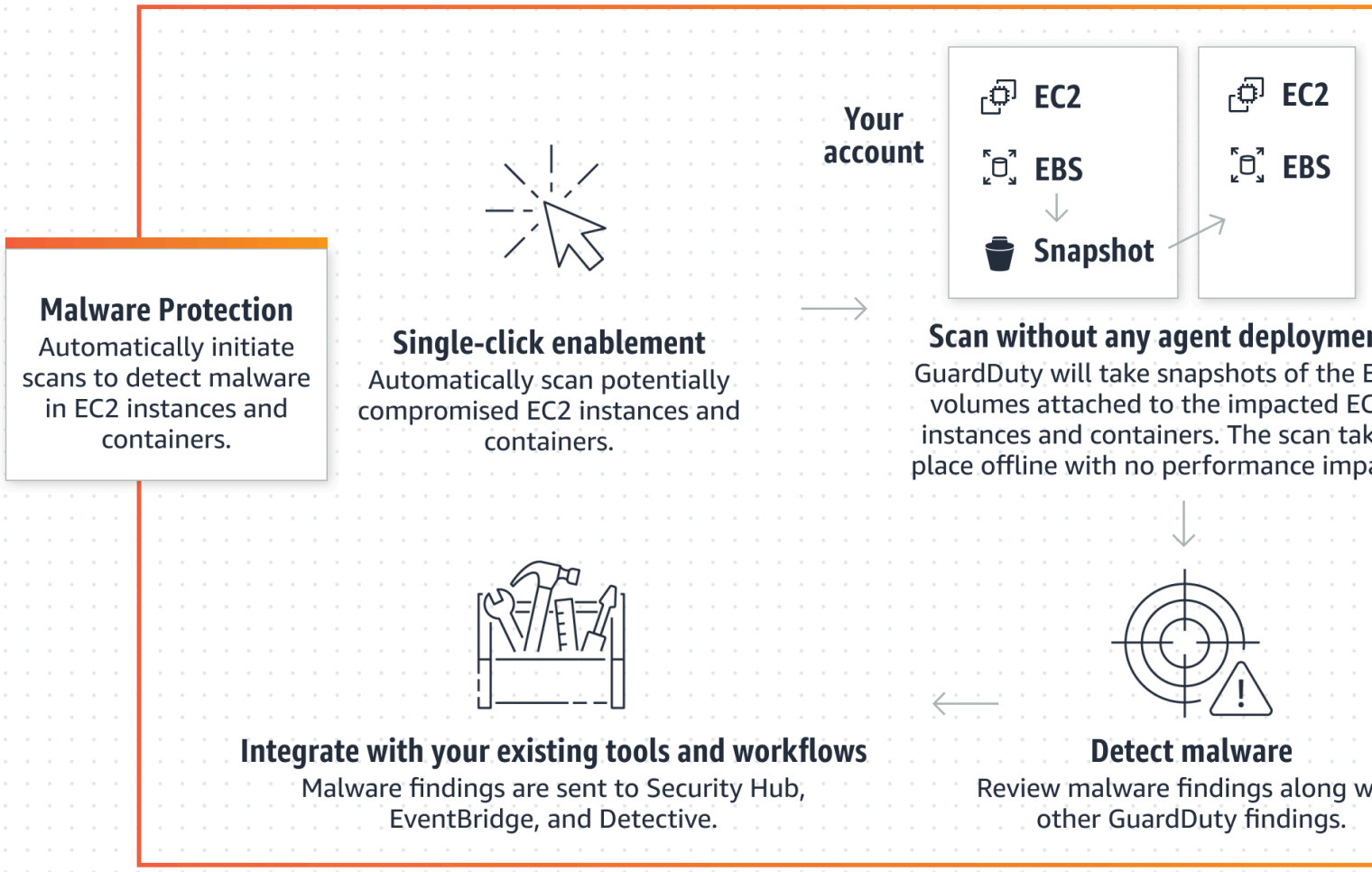
GuardDuty は、Amazon EC2 リソースに追加し、タグ値を `true` に設定できるグローバルタグキー `GuardDutyExcluded` を使用します。このタグキーと値のペアを持つこの Amazon EC2 リソースは、マルウェアスキャンから除外されます。どちらのスキャンタイプ (GuardDuty 実行型マルウェアスキャンおよびオンデマンドマルウェアスキャン) もグローバルタグをサポートしています。Amazon EC2 でオンデマンドマルウェアスキャンを開始すると、スキャン ID が生成されます。ただし、スキャンは `EXCLUDED_BY_SCAN_SETTINGS` によりスキップされます。詳細については、「マルウェアスキャン中にリソースをスキップする理由」を参照してください。

GuardDuty 実行型マルウェアスキャン

GuardDuty 実行型マルウェアスキャンを有効にすると、GuardDuty が GuardDuty 実行型マルウェアスキャンを起動する検出結果 を生成するたびに、影響を受ける可能性のある Amazon EC2 リソースにアタッチされた Amazon Elastic Block Store (Amazon EBS) ボリュームに対するエージェントレスマルウェアスキャンが開始されます。スキャンを開始する前に、カスタマイズのためにアカウントを準備する必要があります。スキャンオプションを使用すると、スキャンするリソースに関連付けされた包含タグを追加したり、スキャンプロセスでスキップするリソースに関連付けされた除外タグを追加したりできます。自動スキャンの開始は、常にスキャンオプションに基づいて実行されます。GuardDuty は、グローバル `GuardDutyExcluded:true` タグのキーと値のペアもサポートしています。このグローバルタグを Amazon EC2 リソースに追加すると、GuardDuty はスキャンを開始した後、このリソースをスキップします。また、スナップショット保持設定をオンにして、マルウェアが検出された可能性のある EBS ボリュームのスナップショットを保持することもできます。スキャンオプション、グローバル除外タグ、スナップショット設定の詳細については、「スナップショット保持と EC2 スキャンカバレッジを設定する」を参照してください。

GuardDuty が同じ Amazon EC2 リソースに対して複数の検出結果を生成した場合、GuardDuty は、最後の GuardDuty 実行型マルウェアスキャンから 24 時間が経過した後にのみスキャンを開始できます。Amazon EC2 インスタンスまたはコンテナワークロードにアタッチされた Amazon EBS ボリュームのスキャン方法については、「GuardDuty が EBS ボリュームをスキャンしてマルウェアを検出する方法」を参照してください。

以下の画像は、GuardDuty 実行型マルウェアスキャンがどのように機能するかについて説明しています。



GuardDuty マルウェア検出方法と使用するスキャンエンジンの詳細については、「GuardDuty マルウェア検出のスキャンエンジン」を参照してください。

マルウェアが見つかると、GuardDuty が Malware Protection for EC2 の検出結果のタイプ を生成します。GuardDuty が同じリソースでマルウェアを示す検出結果が生成しない場合、GuardDuty 実行型マルウェアスキャンは起動されません。同じリソースでオンデマンドのマルウェアスキャンを開始することもできます。詳細については、「GuardDuty でのオンデマンドマルウェアスキャン」を参照してください。

GuardDuty 実行型マルウェアスキャンの 30 日間の無料トライアル

サポートされている AWS アカウント で、GuardDuty 実行型マルウェアスキャンを AWS リージョン いつでも有効または無効にできます。組織がある場合、各メンバーアカウントにはそれぞれ 30 日間の無料トライアルがあります。

30 日間の無料トライアルの仕組みを理解するために、次のシナリオを考慮してください。

- GuardDuty を初めて有効にすると (新しい GuardDuty アカウント)、GuardDuty 実行型マルウェアスキャンも有効になり、GuardDuty サービスに関連付けられた 30 日間の無料トライアルに含まれます。
- 既存の GuardDuty アカウントは、初めて GuardDuty 実行型マルウェアスキャンを有効にする際、30 日間の無料トライアルを利用できます。この機能を別のリージョンで初めて有効にすると、そのリージョンで 30 日間の無料トライアルが適用されます。
- この保護プランが GuardDuty 実行型マルウェアスキャンとオンデマンドマルウェアスキャンの 2 つのスキャンタイプに分割される AWS リージョン 前に で Malware Protection for EC2 を使用していた場合は、同じ の同じ料金モデルで GuardDuty 実行型マルウェアスキャンを引き続き使用できます AWS リージョン。新しいリージョンで初めて GuardDuty 実行型マルウェアスキャンを有効にした場合、アカウントに対して 30 日間の無料トライアルが適用されます。

> **Note**
>
> 30 日間の無料トライアル期間中の場合でも、Amazon EBS ボリュームのスナップショットの作成とその保持には標準の使用コストが適用されます。詳細については、Amazon EBS の料金表を参照してください。

マルチアカウント環境での GuardDuty 実行型マルウェアスキャンの有効化

マルチアカウント環境では、メンバーアカウントに代わって GuardDuty 実行型マルウェアスキャンを有効にできるのは、GuardDuty 管理者アカウントのみです。さらに、 AWS Organizations サポートでメンバーアカウントを管理する管理者アカウントは、組織内のすべての既存および新しいアカウントで GuardDuty 実行型マルウェアスキャンを自動的に有効にすることを選択できます。詳細については、「を使用した GuardDuty アカウントの管理 AWS Organizations」を参照してください。

GuardDuty 実行型マルウェアスキャンを有効にするための信頼されたアクセスの確立

GuardDuty 委任管理者アカウントが組織の管理アカウントと同じではない場合、管理アカウントは組織の GuardDuty 実行型マルウェアスキャンを有効にする必要があります。これにより、委任管理者アカウントは、管理対象のMalware Protection for EC2 のためのサービスにリンクされたロールの許可メンバーアカウントに を作成できます AWS Organizations。

> **Note**
>
> 委任 GuardDuty 管理者アカウントを指定する前に、「考慮事項とレコメンデーション」を参照してください。

任意のアクセス方法を選択して、委任 GuardDuty 管理者アカウントが、組織内のメンバーアカウントで GuardDuty 実行型マルウェアスキャンを有効にできるようにします。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   ログインするには、 AWS Organizations 組織の管理アカウントを使用します。

2. a. 委任 GuardDuty 管理者アカウントを指定していない場合は、次の操作を実行します。

      [設定] ページの [委任 GuardDuty 管理者アカウント] で、組織で GuardDuty ポリシーを管理するために指定する 12 桁の **`account ID`** を入力します。[委任] を選択します。

   b. i. 管理アカウントとは異なる委任 GuardDuty 管理者アカウントを既に指定している場合は、次の操作を実行します。

      [Settings] (設定) ページの [Delegated Administrator] (委任された管理者) で、[Permissions] (許可) 設定をオンにします。このアクションにより、委任 GuardDuty 管理者アカウントが関連するアクセス許可をメンバーアカウントにアタッチし、これらのメンバーアカウントで GuardDuty 実行型マルウェアスキャンを有効にすることができます。

   ii. 管理アカウントと同じ委任 GuardDuty 管理者アカウントを既に指定している場合、メンバーアカウントで GuardDuty 実行型マルウェアスキャンを直接有効にできます。詳細については、「すべてのメンバーアカウントのために GuardDuty が開始するマルウェアスキャンを自動的に有効にする」を参照してください。

> **Tip**
>
> 委任 GuardDuty 管理者アカウントが管理アカウントと異なる場合、メンバーアカウントで GuardDuty 実行型マルウェアスキャンを有効にできるようにするため、アクセス許可を委任 GuardDuty 管理者アカウントに付与する必要があります。

3. 委任 GuardDuty 管理者アカウントが他のリージョンのメンバーアカウントに対して GuardDuty 実行型マルウェアスキャンを有効にすることを許可する場合は、 を変更し AWS リージョン、上記の手順を繰り返します。

API/CLI

1. 管理アカウントの認証情報を使用して、次のコマンドを実行します。

```
aws organizations enable-aws-service-access --service-principal malware-
protection.guardduty.amazonaws.com
```

2. (オプション) 委任管理者アカウントではない管理アカウントで GuardDuty 実行型マルウェアスキャンを有効にするには、管理アカウントで最初に「Malware Protection for EC2 のためのサービスにリンクされたロールの許可」を明示的に作成し、他のメンバーアカウントと同様に、委任管理者アカウントから GuardDuty 実行型マルウェアスキャンを有効にします。

```
aws iam create-service-linked-role --aws-service-name malware-
protection.guardduty.amazonaws.com
```

3. 現在選択されている AWS リージョンで委任 GuardDuty 管理者アカウントを指定しました。あるリージョンでアカウントを委任 GuardDuty 管理者アカウントに指定した場合、そのアカウントは他のすべてのリージョンで委任 GuardDuty 管理者アカウントである必要があります。上記のステップを他のすべてのリージョンについて繰り返します。

委任 GuardDuty 管理者アカウントでの GuardDuty 実行型マルウェアスキャンの設定

任意のアクセス方法を選択して、委任 GuardDuty 管理者アカウントで GuardDuty 実行型マルウェアスキャンを有効または無効にします。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[Malware Protection for EC2] を選択します。
3. [Malware Protection for EC2] ページで、[GuardDuty 実行型マルウェアスキャン] の横にある [編集] を選択します。
4. 次のいずれかを行います:

[すべてのアカウントについて有効にする] の使用

- [すべてのアカウントについて有効にする] を選択します。これにより、 AWS 組織に参加する新しいアカウントを含め、組織内のすべてのアクティブな GuardDuty アカウントの保護プランが有効になります。
- [保存] を選択します。

[アカウントを手動で設定] の使用

- 委任 GuardDuty 管理者アカウントでのみ保護プランを有効にするには、[アカウントを手動で設定] を選択します。
- [委任 GuardDuty 管理者 (このアカウント)] セクションで [有効にする] を選択します。
- [保存] を選択します。

API/CLI

ユーザー独自のリージョンレベルのディテクター ID を使用し、`features` オブジェクト `name` を `EBS_MALWARE_PROTECTION` として、`status` を `ENABLED` として渡して、updateDetector API オペレーションを実行します。

次の AWS CLI コマンドを実行して、GuardDuty 実行型マルウェアスキャンを有効にできます。必ず委任 GuardDuty 管理者アカウントの有効な*###### ID* を使用してください。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 /
              --account-ids 555555555555 /
              --features '[{"Name": "EBS_MALWARE_PROTECTION", "Status": "ENABLED"}]'
```

すべてのメンバーアカウントのために GuardDuty が開始するマルウェアスキャンを自動的に有効にする

任意のアクセス方法を選択して、すべてのメンバーアカウントのために GuardDuty が開始するマルウェアスキャン機能を有効にします。これには、既存のメンバーアカウントと、組織に参加する新しいアカウントが含まれます。

Console

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   必ず委任 GuardDuty 管理者アカウントの認証情報を使用してください。

2. 次のいずれかを行います:

   [Malware Protection for EC2] ページの使用

   1. ナビゲーションペインで、[Malware Protection for EC2] を選択します。
   2. [Malware Protection for EC2] ページの [GuardDuty 実行型マルウェアスキャン] セクションで [編集] を選択します。
   3. [すべてのアカウントについて有効にする] を選択します。このアクションにより、組織内の既存のアカウントと新しいアカウントの両方について GuardDuty が開始するマルウェアスキャンが自動的に有効になります。
   4. [保存] を選択します。

   > **Note**
   >
   > メンバーアカウントの設定を更新するには、最大 24 時間かかる場合があります。

   [アカウント] ページの使用

   1. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
   2. [アカウント] ページで、[招待によるアカウントの追加] の前に [自動有効化] の詳細設定を選択します。
   3. [自動有効化の詳細設定を管理] ウィンドウで、[GuardDuty が開始したマルウェアスキャン] の下の [すべてのアカウントについて有効にする] を選択します。
   4. [Malware Protection for EC2] ページの [GuardDuty 実行型マルウェアスキャン] セクションで [編集] を選択します。
   5. [すべてのアカウントについて有効にする] を選択します。このアクションにより、組織内の既存のアカウントと新しいアカウントの両方について GuardDuty が開始するマルウェアスキャンが自動的に有効になります。
   6. [保存] を選択します。

> **Note**
>
> メンバーアカウントの設定を更新するには、最大 24 時間かかる場合があります。

[アカウント] ページの使用

1. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
2. [アカウント] ページで、[招待によるアカウントの追加] の前に [自動有効化] の詳細設定を選択します。
3. [自動有効化の詳細設定を管理] ウィンドウで、[GuardDuty が開始したマルウェアスキャン] の下の [すべてのアカウントについて有効にする] を選択します。
4. [保存] を選択します。

[すべてのアカウントについて有効にする] オプションを使用できない場合は、「メンバーアカウントで GuardDuty 実行型マルウェアスキャンを選択的に有効にする」を参照してください。

API/CLI

- メンバーアカウントで GuardDuty 実行型マルウェアスキャンを選択的に有効にするには、ユーザー独自の*###### ID* を使用して updateMemberDetectors API オペレーションを呼び出します。
- 次の例では、単一のメンバーアカウントに GuardDuty 実行型マルウェアスキャンを有効にする方法を示しています。メンバーアカウントを無効にするには、DISABLED を ENABLED に置き換えてください。

  アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-member-detectors --detector-
id 12abc34d567e8fa901bc2d34e56789f0 --account-ids 111122223333 --features
 '[{"Name": "EBS_MALWARE_PROTECTION", "Status": "ENABLED"}]'
```

  スペースで区切られたアカウント ID のリストを渡すこともできます。

- コードが正常に実行されると、UnprocessedAccounts の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

すべての既存のアクティブなメンバーアカウントのために GuardDuty が開始するマルウェアスキャンを有効にする

任意のアクセス方法を選択して、組織内のすべての既存のアクティブなメンバーアカウントのために GuardDuty が開始するマルウェアスキャンを有効にします。

すべての既存のアクティブなメンバーアカウントのために GuardDuty が開始するマルウェアスキャンを設定するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   委任 GuardDuty 管理者アカウントの認証情報を使用してサインインします。
2. ナビゲーションペインで、[Malware Protection for EC2] を選択します。
3. [Malware Protection for EC2] では、[GuardDuty 実行型マルウェアスキャン] 設定の現在のステータスを表示できます。[アクティブなメンバーアカウント] セクションで、[アクション] を選択します。
4. [アクション] ドロップダウンメニューから、[すべての既存のアクティブなメンバーアカウントについて有効にする] を選択します。
5. [保存] を選択します。

新しいメンバーアカウントのために GuardDuty が開始するマルウェアスキャンを自動的に有効にする

新しく追加されたメンバーアカウントは、GuardDuty が開始するマルウェアスキャンの設定を選択する前に GuardDuty を [有効にする] 必要があります。招待によって管理されるメンバーアカウントは、各自でアカウントに GuardDuty 実行型マルウェアスキャンを手動で設定できます。詳細については、「Step 3 - Accept an invitation」を参照してください。

任意のアクセス方法を選択して、組織に参加する新しいアカウントのために GuardDuty が開始するマルウェアスキャンを有効にします。

Console

委任 GuardDuty 管理者アカウントは、[Malware Protection for EC2] または [アカウント] のいずれかのページを使用して、組織内の新しいメンバーアカウントで GuardDuty 実行型マルウェアスキャンを有効にすることができます。

新しいメンバーアカウントのために GuardDuty が開始するマルウェアスキャンを自動的に有効にするには

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   必ず委任 GuardDuty 管理者アカウントの認証情報を使用してください。
2. 次のいずれかを行います:
   - [Malware Protection for EC2] ページの使用:
     1. ナビゲーションペインで、[Malware Protection for EC2] を選択します。
     2. [Malware Protection for EC2] ページの [GuardDuty 実行型マルウェアスキャン] で [編集] を選択します。
     3. [アカウントを手動で設定] を選択します。
     4. [新しいメンバーアカウントについて自動的に有効にする] を選択します。このステップにより、新しいアカウントが組織に参加するたびに、そのアカウントのために GuardDuty が開始するマルウェアスキャンが自動的に有効になります。この設定を変更できるのは、組織の委任 GuardDuty 管理者アカウントだけです。
     5. [保存] を選択します。
   - [Accounts] (アカウント) ページを使用する場合:
     1. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
     2. [アカウント] ページで、[自動有効化] 設定を選択します。
     3. [自動有効化の詳細設定を管理] ウィンドウで、[GuardDuty が開始するマルウェアスキャン] の下の [新しいアカウントについて有効にする] を選択します。
     4. [保存] を選択します。

API/CLI

- 新しいメンバーアカウントに GuardDuty が開始するマルウェアスキャンを有効または無効にするには、自分の`###### ID` を使用して UpdateOrganizationConfiguration API オペレーションを起動します。

- 次の例では、単一のメンバーアカウントに GuardDuty 実行型マルウェアスキャンを有効にする方法を示しています。無効にするには、「メンバーアカウントで GuardDuty 実行型マルウェアスキャンを選択的に有効にする」を参照してください。組織に参加する新規アカウントすべてに対して有効にしたくない場合は、`AutoEnable` を `NONE` に設定します。

  アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

  ```
  aws guardduty update-organization-configuration --detector-
  id 12abc34d567e8fa901bc2d34e56789f0 --AutoEnable --features '[{"Name":
   "EBS_MALWARE_PROTECTION", "AutoEnable": NEW}]'
  ```

  スペースで区切られたアカウント ID のリストを渡すこともできます。

- コードが正常に実行されると、`UnprocessedAccounts` の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

メンバーアカウントで GuardDuty 実行型マルウェアスキャンを選択的に有効にする

任意のアクセス方法を選択して、GuardDuty が開始するマルウェアスキャンをメンバーアカウントのために選択的に設定します。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
3. [アカウント] ページで、[GuardDuty が開始するマルウェアスキャン] の列でメンバーアカウントのステータスを確認します。
4. GuardDuty が開始するマルウェアスキャンを設定するアカウントを選択します。一度に複数のアカウントを選択できます。
5. [保護プランを編集] メニューから、[GuardDuty が開始するマルウェアスキャン] に適切なオプションを選択します。

API/CLI

メンバーアカウントに GuardDuty 実行型マルウェアスキャンを選択的に有効または無効にするには、お持ちの*###### ID* を使用して updateMemberDetectors API オペレーションを起動します。

次の例では、単一のメンバーアカウントに GuardDuty 実行型マルウェアスキャンを有効にする方法を示しています。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0
 --account-ids 111122223333 --features '[{"Name": "EBS_MALWARE_PROTECTION",
 "Status": "ENABLED"}]'
```

スペースで区切られたアカウント ID のリストを渡すこともできます。

コードが正常に実行されると、`UnprocessedAccounts` の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

メンバーアカウントに GuardDuty 実行型マルウェアスキャンを選択的に有効にするには、ユーザー独自の*###### ID* を使用して updateMemberDetectors API オペレーションを実行します。次の例では、単一のメンバーアカウントに GuardDuty 実行型マルウェアスキャンを有効にする方法を示しています。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0
 --account-ids 111122223333 --data-sources '{"MalwareProtection":
{"ScanEc2InstanceWithFindings":{"EbsVolumes":true}}}'
```

スペースで区切られたアカウント ID のリストを渡すこともできます。

コードが正常に実行されると、`UnprocessedAccounts` の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

招待によって管理される組織の既存アカウントに GuardDuty 実行型マルウェアスキャンを有効にする

メンバーアカウントで GuardDuty Malware Protection for EC2 のサービスにリンクされたロール (SLR) が作成されている必要があります。 AWS Organizationsによって管理されていないメンバーアカウントで、管理者アカウントが GuardDuty 実行型マルウェアスキャン機能を有効にすることはできません。

現在、https://console.aws.amazon.com/guardduty/ の GuardDuty コンソールから次の手順を実行し、既存のメンバーアカウントに GuardDuty 実行型マルウェアスキャンを有効にできます。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   管理者アカウントの認証情報を使用してサインインします。
2. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
3. GuardDuty が開始するマルウェアスキャンを有効にするメンバーアカウントを選択します。一度に複数のアカウントを選択できます。
4. [アクション] を選択します。
5. [Disassociate member] (メンバーの関連付けを解除する) を選択します。
6. メンバーアカウントのナビゲーションペインで、[保護プラン] から [Malware Protection] を選択します。
7. [GuardDuty が開始するマルウェアスキャンを有効にする] を選択します。GuardDuty でメンバーアカウントの SLR が作成されます。SLR の詳細については、「Malware Protection for EC2 のためのサービスにリンクされたロールの許可」を参照してください。
8. 管理者アカウントのナビゲーションペインで [アカウント] を選択します。
9. 組織に追加し直す必要があるメンバーアカウントを選択します。
10. [Actions] (アクション)、[Add member] (メンバーの追加) の順に選択します。

API/CLI

1. 管理者アカウントを使用し、GuardDuty 実行型マルウェアスキャンを有効にするメンバーアカウントで DisassociateMembers API を実行します。
2. メンバーアカウントを使用して UpdateDetector を起動し、GuardDuty 実行型マルウェアスキャンを有効にします。

アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0
 --data-sources '{"MalwareProtection":{"ScanEc2InstanceWithFindings":
{"EbsVolumes":true}}}'
```

3. 管理者アカウントを使用して CreateMembers API を実行して、メンバーを組織に追加し直します。

スタンドアロンアカウントでの GuardDuty 実行型マルウェアスキャンの有効化

スタンドアロンアカウントは、特定の の で保護プランを有効または無効にする決定を所有 AWS アカウント します AWS リージョン。

アカウントが を通じて AWS Organizations、または招待方法によって GuardDuty 管理者アカウントに関連付けられている場合、このセクションはアカウントには適用されません。詳細については、「マルチアカウント環境での GuardDuty 実行型マルウェアスキャンの有効化」を参照してください。

GuardDuty 実行型マルウェアスキャンを有効にすると、GuardDuty は、GuardDuty に関与していた Amazon EC2 インスタンスにアタッチされている Amazon EBS ボリュームのマルウェアスキャンを開始します。マルウェアスキャンを開始する検出結果のリストについては、「GuardDuty 実行型マルウェアスキャンを起動する検出結果」を参照してください。

任意のアクセス方法を選択して、スタンドアロンアカウントのために GuardDuty が開始するマルウェアスキャンを設定します。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインの [保護プラン] で、[Malware Protection for EC2] を選択します。
3. Malware Protection for EC2 ペインには、アカウントの GuardDuty 実行型マルウェアスキャンにおける現在のステータスが一覧表示されます。[有効にする] を選択し、このアカウントで GuardDuty 実行型マルウェアスキャンを有効にします。
4. [保存] を選択して、選択を確定します。

API/CLI

ユーザー独自のリージョンレベルのディテクター ID を使用し、EbsVolumes を true に設定した dataSources オブジェクトを渡して、updateDetector API オペレーションを実行します。

次の AWS CLI コマンド AWS CLI を実行して、 を使用して GuardDuty 実行型マルウェアスキャンを有効にすることもできます。必ずご自身の有効な *###### ID* を使用してください。

アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
 aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --
features [{"Name" : "EBS_MALWARE_PROTECTION", "Status" : "ENABLED"}]'
```

GuardDuty 実行型マルウェアスキャンを起動する検出結果

GuardDuty が Amazon EC2 インスタンスまたは Amazon EC2 インスタンスで実行されているコンテナワークロードでマルウェアを示す疑わしい動作を検出すると、GuardDuty は検出結果を生成します。この生成された検出結果が次の GuardDuty 検出結果のリストに属する場合、GuardDuty は検出結果に関連する Amazon EC2 インスタンスにアタッチされた Amazon EBS ボリュームに対してマルウェアスキャンを自動的に開始します。スキャン後、GuardDuty がマルウェアを検出すると、1 つ以上の「Malware Protection for EC2 の検出結果のタイプ」も生成されます。

アカウントで次の GuardDuty 検出結果のいずれかが生成された場合、GuardDuty は侵害された可能性のある Amazon EC2 インスタンスの Amazon EBS ボリュームでマルウェアスキャンを自動的に開始します。

- Backdoor:EC2/C&CActivity.B
- Backdoor:EC2/C&CActivity.B!DNS
- Backdoor:EC2/DenialOfService.Dns
- Backdoor:EC2/DenialOfService.Tcp
- Backdoor:EC2/DenialOfService.Udp
- Backdoor:EC2/DenialOfService.UdpOnTcpPorts
- Backdoor:EC2/DenialOfService.UnusualProtocol
- Backdoor:EC2/Spambot
- CryptoCurrency:EC2/BitcoinTool.B

- CryptoCurrency:EC2/BitcoinTool.B!DNS
- DefenseEvasion:Runtime/PtraceAntiDebugging
- DefenseEvasion:Runtime/SuspiciousCommand
- Execution:Runtime/MaliciousFileExecuted
- Execution:Runtime/MaliciousFileExecuted.Custom
- Execution:Runtime/SuspiciousCommand
- Execution:Runtime/SuspiciousShellCreated
- Execution:Runtime/SuspiciousTool
- Impact:EC2/AbusedDomainRequest.Reputation
- Impact:EC2/BitcoinDomainRequest.Reputation
- Impact:EC2/MaliciousDomainRequest.Reputation
- Impact:EC2/PortSweep
- Impact:EC2/SuspiciousDomainRequest.Reputation
- Impact:EC2/WinRMBruteForce (アウトバウンドのみ)
- PrivilegeEscalation:Runtime/ElevationToRoot
- Recon:EC2/Portscan
- Trojan:EC2/BlackholeTraffic
- Trojan:EC2/BlackholeTraffic!DNS
- Trojan:EC2/DGADomainRequest.B
- Trojan:EC2/DGADomainRequest.C!DNS
- Trojan:EC2/DNSDataExfiltration
- Trojan:EC2/DriveBySourceTraffic!DNS
- Trojan:EC2/DropPoint
- Trojan:EC2/DropPoint!DNS
- Trojan:EC2/PhishingDomainRequest!DNS
- UnauthorizedAccess:EC2/RDPBruteForce (アウトバウンドのみ)
- UnauthorizedAccess:EC2/SSHBruteForce (アウトバウンドのみ)
- UnauthorizedAccess:EC2/TorClient
- UnauthorizedAccess:EC2/TorRelay
- Backdoor:Runtime/C&CActivity.B

- Backdoor:Runtime/C&CActivity.B!DNS
- CryptoCurrency:Runtime/BitcoinTool.B
- CryptoCurrency:Runtime/BitcoinTool.B!DNS
- Execution:Runtime/NewBinaryExecuted
- Execution:Runtime/NewLibraryLoaded
- Execution:Runtime/ReverseShell
- Impact:Runtime/AbusedDomainRequest.Reputation
- Impact:Runtime/BitcoinDomainRequest.Reputation
- Impact:Runtime/CryptoMinerExecuted
- Impact:Runtime/MaliciousDomainRequest.Reputation
- Impact:Runtime/SuspiciousDomainRequest.Reputation
- PrivilegeEscalation:Runtime/CGroupsReleaseAgentModified
- PrivilegeEscalation:Runtime/ContainerMountsHostDirectory
- PrivilegeEscalation:Runtime/DockerSocketAccessed
- PrivilegeEscalation:Runtime/RuncContainerEscape
- PrivilegeEscalation:Runtime/UserfaultfdUsage
- Trojan:Runtime/BlackholeTraffic
- Trojan:Runtime/BlackholeTraffic!DNS
- Trojan:Runtime/DropPoint
- Trojan:Runtime/DropPoint!DNS
- Trojan:Runtime/DGADomainRequest.C!DNS
- Trojan:Runtime/DriveBySourceTraffic!DNS
- Trojan:Runtime/PhishingDomainRequest!DNS
- UnauthorizedAccess:Runtime/MetadataDNSRebind

GuardDuty でのオンデマンドマルウェアスキャン

オンデマンドのマルウェアスキャンは、Amazon EC2 インスタンスにアタッチされた Amazon Elastic Block Store (Amazon EBS) ボリュームでマルウェアの存在を検出するのに役立ちます。設定は不要で、スキャンする Amazon EC2 インスタンスの Amazon リソースネーム (ARN) を指定することで、オンデマンドマルウェアスキャンを開始できます。GuardDuty コンソールまたは API を使用

してオンデマンドマルウェアスキャンを開始できます。オンデマンドのマルウェアスキャンを開始する前に、希望する スナップショットの保持 設定をセットできます。以下のシナリオは、GuardDuty でオンデマンドのマルウェアスキャンのスキャンタイプを使用するタイミングを判断するのに役立ちます。

- GuardDuty 実行型マルウェアスキャンを有効にせず、Amazon EC2 インスタンス内のマルウェアの存在を検出したいとします。
- GuardDuty 実行型マルウェアスキャンを有効にし、スキャンが自動的に起動されました。生成された Malware Protection for EC2 の検出結果タイプに応じた推奨修復方法を実行した後、同じリソースにスキャンを開始する場合、前回のスキャン開始時刻から 1 時間経過した後にオンデマンドマルウェアスキャンを開始できます。

  オンデマンドマルウェアスキャンは、前回のマルウェアスキャンが開始された時刻から 24 時間経過する必要はありません。同じリソースでオンデマンドのマルウェアスキャンを開始する前に 1 時間経過している必要があります。同じ EC2 インスタンスでマルウェアスキャンの重複を避けるには、「以前にスキャンした Amazon EC2 インスタンスの再スキャン」を参照してください。

> **Note**
>
> オンデマンドのマルウェアスキャンは、GuardDuty の 30 日間の無料トライアル期間には含まれていません。使用コストは、マルウェアスキャンごとにスキャンされた Amazon EBS ボリュームの合計に適用されます。詳細については、「Amazon GuardDuty 料金設定」を参照してください。Amazon EBS ボリュームのスナップショットおよびその保持にかかるコストについては、「Amazon EBS 料金設定」を参照してください。

オンデマンドのマルウェアスキャンの仕組み

オンデマンドマルウェアスキャンでは、Amazon EC2 インスタンスが現在使用中であっても、そのインスタンスに対するマルウェアスキャンリクエストを開始できます。オンデマンドマルウェアスキャンを開始すると、GuardDuty はスキャン用に Amazon リソースネーム (ARN) が提供された Amazon EC2 インスタンスにアタッチされた Amazon EBS ボリュームのスナップショットを作成します。次に、GuardDuty はこれらのスナップショットを GuardDuty サービスアカウント と共有します。GuardDuty は、GuardDuty サービスアカウントでそれらのスナップショットで暗号化された EBS ボリュームのレプリカを作成します。Amazon EBS ボリュームがスキャンされる方法の詳細については、「GuardDuty が EBS ボリュームをスキャンしてマルウェアを検出する方法」を参照してください。

> **Note**
>
> GuardDuty は、オンデマンドマルウェアスキャンを開始した時点で Amazon EBS ボリュームに既に書き込まれているデータのスナップショットを作成します。

マルウェアが見つかってスナップショットの保持設定を有効にしている場合、EBS ボリュームのスナップショットは自動的に AWS アカウントで保持されます。オンデマンドのマルウェアスキャンは Malware Protection for EC2 の検出結果のタイプ を生成します。マルウェアが見つからなければ、スナップショットの保持設定とは関係なく、EBS ボリュームのスナップショットは削除されます。

GuardDuty は、Amazon EC2 リソースに追加し、タグ値を `true` に設定できるグローバルタグキー `GuardDutyExcluded` を使用します。このタグキーと値のペアを持つこの Amazon EC2 リソースは、マルウェアスキャンから除外されます。どちらのスキャンタイプ (GuardDuty 実行型マルウェアスキャンおよびオンデマンドマルウェアスキャン) もグローバルタグをサポートしています。Amazon EC2 でオンデマンドマルウェアスキャンを開始すると、スキャン ID が生成されます。ただし、スキャンは `EXCLUDED_BY_SCAN_SETTINGS` によりスキップされます。詳細については、「マルウェアスキャン中にリソースをスキップする理由」を参照してください。

GuardDuty でのオンデマンドマルウェアスキャンの開始

このセクションでは、オンデマンドマルウェアスキャンを開始する前の前提条件と、リソースのスキャンを初めて開始するステップを示します。

GuardDuty 管理者アカウントは、アカウントに次の前提条件がセットアップされているアクティブなメンバーアカウントに代わり、オンデマンドマルウェアスキャンを開始できます。GuardDuty のスタンドアロンアカウントおよびアクティブなメンバーアカウントは、各自の Amazon EC2 インスタンスに対してオンデマンドマルウェアスキャンを開始することもできます。

前提条件

オンデマンドマルウェアスキャンを開始する前に、アカウントが以下の前提条件を満たす必要があります。

- GuardDuty は、オンデマンドマルウェアスキャンを開始する AWS リージョン で有効にする必要があります。
- AWS 管理ポリシー: AmazonGuardDutyFullAccess_v2 (推奨) が IAM ユーザーまたは IAM ロールにアタッチされていることを確認します。IAM ユーザーまたは IAM ロールに関連付けされたアクセスキーおよびシークレットキーが必要になります。

- 委任 GuardDuty 管理者アカウントは、アクティブなメンバーアカウントに代わってオンデマンドマルウェアスキャンを開始することができます。
- オンデマンドマルウェアスキャンを開始する前に、過去 1 時間以内に同じリソースに対してスキャンが開始されていないことを確認してください。そうしないと、重複除外されます。詳細については、「以前にスキャンした Amazon EC2 インスタンスの再スキャン」を参照してください。
- メンバーアカウントに「Malware Protection for EC2 のためのサービスにリンクされたロールの許可」がない場合、アカウントに属する Amazon EC2 インスタンスのオンデマンドマルウェアスキャンを開始すると、Malware Protection for EC2 に SLR が自動的に作成されます。

> **Important**
>
> マルウェアスキャンの進行中に、誰も Malware Protection for EC2 に対する SLR アクセス許可を削除しないようにしてください。このマルウェアスキャンは、GuardDuty によって開始することも、オンデマンドで開始することもできます。SLR を削除すると、スキャンが正常に完了できなくなり、明確なスキャン結果が得られなくなります。

オンデマンドマルウェアスキャンを開始する

アカウントでオンデマンドマルウェアスキャンを開始するには、GuardDuty コンソールまたは を使用します AWS CLI。スキャンを開始する Amazon EC2 Amazon リソースネーム (ARN) を指定する必要があります。詳細な手順は、次のセクションのコンソールと API/AWS CLI 手順の両方に記載されています。

任意のアクセス方法を選択して、オンデマンドマルウェアスキャンを開始します。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. 次のオプションのいずれかを使用してスキャンを開始します。

    a. [Malware Protection for EC2] ページの使用:

        i. ナビゲーションペインの [保護プラン] で、[Malware Protection for EC2] を選択します。

        ii. [Malware Protection for EC2] ページで、スキャンを開始する [Amazon EC2 インスタンス ARN][1] を指定します。

b. [マルウェアスキャン] ページの使用:

i. ナビゲーションペインで、[マルウェアスキャン] を選択します。

ii. [オンデマンドスキャンを開始] を選択し、スキャンを開始する [Amazon EC2 インスタンス ARN][1] を指定します。

iii. 再スキャンの場合、[マルウェアスキャン] ページで [Amazon EC2] インスタンス ID を選択します。

[オンデマンドスキャンの開始] ドロップダウンを展開し、[選択したインスタンスを再スキャン] を選択します。

3. いずれかの方法でスキャンを正常に開始すると、スキャン ID が生成されます。このスキャン ID を使用して、スキャンの進行状況を追跡できます。詳細については、「スキャンステータスと結果の監視」を参照してください。

API/CLI

オンデマンドマルウェアスキャンを開始する Amazon EC2 インスタンス[1] の `resourceArn` を受け入れる StartMalwareScan を呼び出します。

```
aws guardduty start-malware-scan --resource-arn "arn:aws:ec2:us-
east-1:555555555555:instance/i-b188560f"
```

スキャンを正常に開始すると、`StartMalwareScan` は `scanId` を返します。DescribeMalwareScans を呼び出すと、開始されたスキャンの進行状況をモニタリングできます。

[1]Amazon EC2 インスタンス ARN の形式については、「Amazon リソースネーム (ARN)」を参照してください。Amazon EC2 インスタンスの場合、パーティション、リージョン、 AWS アカウント ID、Amazon EC2 インスタンス ID の値を置き換えることで、次の ARN 形式の例を使用できます。インスタンス ID の長さについては、「リソース ID」を参照してください。

```
arn:aws:ec2:us-east-1:555555555555:instance/i-b188560f
```

AWS Organizations サービスコントロールポリシー – アクセス拒否

のサービスコントロールポリシー (SCPs) を使用すると AWS Organizations、委任 GuardDuty 管理者アカウントはアクセス許可を制限し、アカウントが所有する Amazon EC2 インスタンスのオンデマンドマルウェアスキャンを開始するなどのアクションを拒否できます。

GuardDuty メンバーアカウントは、Amazon EC2 インスタンスに対してオンデマンドマルウェアスキャンを開始すると、エラーが表示されることがあります。管理アカウントに接続して、メンバーアカウントに SCP がセットアップされた理由について説明します。詳細については、「許可に対する SCP の影響」を参照してください。

以前にスキャンした Amazon EC2 インスタンスの再スキャン

スキャンが GuardDuty 実行型であるか、オンデマンドで開始さるかとは関係なく、前回のマルウェアスキャンの開始時刻から 1 時間後に、同じ Amazon EC2 インスタンスに新しいオンデマンドマルウェアスキャンを開始できます。前回のマルウェアスキャンの開始から 1 時間以内に新しいマルウェアスキャンが開始された場合、リクエストの結果として次のエラーが発生し、このリクエストのスキャン ID は生成されません。

```
A scan was started on this resource recently. You can request a scan on the same resource one hour after the previous scan start time.
```

インスタンスを再スキャンするステップは、オンデマンドマルウェアスキャンを初めて開始する場合と同じです。このステップについては、「オンデマンドマルウェアスキャンを開始する」を参照してください。

マルウェアスキャンのステータスを追跡するには、「Malware Protection for EC2 のスキャンステータスと結果のモニタリング」を参照してください。

Malware Protection for EC2 のスキャンステータスと結果のモニタリング

Amazon EC2 インスタンスでマルウェアスキャンが開始されると、GuardDuty はステータスと結果のフィールドを自動的に提供します。遷移を通じてステータスをモニタリングし、マルウェアが検出されたかどうかを確認できます。以下の表に、マルウェアスキャンに関連する可能な値を示します。

考えられる 値
Running、 Completed 、 Skipped、または Failed
Clean、または Infected
GuardDuty initiated 、または On demand

*スキャン結果は、スキャンステータスが Completed になった場合にのみ入力されます。スキャン結果 Infected は、GuardDuty がマルウェアの存在を検出したことを意味します。

マルウェアスキャンの各検索結果の保持期間は 90 日です。任意のアクセス方法を選択して、マルウェアスキャンのステータスを追跡します。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[EC2 マルウェアスキャン] を選択します。
3. [フィルター検索バー] で利用可能な次の [プロパティ] によってマルウェアスキャンをフィルターできます。

- スキャン ID – EC2 マルウェアスキャンに関連付けられた一意の識別子。
- アカウント ID – マルウェアスキャンが開始された AWS アカウント ID。
- EC2 インスタンス ARN – スキャンに関連付けられた Amazon EC2 インスタンスに関連付けられた Amazon リソースネーム (ARN)。
- スキャンステータス – 実行中、スキップ、完了 などの EBS ボリュームのスキャンステータス
- スキャンタイプ – これがオンデマンドマルウェアスキャンか GuardDuty 実行型マルウェアスキャンかを示します。

API/CLI

- マルウェアスキャンの結果が表示されると、DescribeMalwareScans を使用して、EC2_INSTANCE_ARN、SCAN_ID、ACCOUNT_ID、SCAN_TYPE、GUARDDUTY_FINDING_ID、SCA および SCAN_START_TIME に基づいてマルウェアスキャンをフィルタリングできます。

  GUARDDUTY_FINDING_ID フィルタ条件は、SCAN_TYPE が GuardDuty によって開始されたときに利用できます。

- *filter-criteria* の例は、次のコマンドで変更できます。現在、1 つの CriterionKey を基準にフィルタリングできます。CriterionKey のオプションは、EC2_INSTANCE_ARN、SCAN_ID、ACCOUNT_ID、SCAN_TYPE、GUARDDUTY_FINDING_ID、SCA です。

  max-results (最大 50) と *sort-criteria* を変更できます。AttributeName は必須であり、scanStartTime である必要があります。

  次の例で、*##*の値はプレスホルダーです。アカウントに適した値に置き換えます。例えば、例 detector-id *60b87779336485625 54d637e0e4bb3b2* を独自の有効な detector-id に置き換えます。以下のように同じ CriterionKey を使うなら、必ず EqualsValue の例を、ご自身の有効な AWS *scan-id* に置き換えてください。

```
aws guardduty describe-malware-scans --detector-
id 60b87779336485625 54d637e0e4bb3b2 --max-results 1 --sort-criteria
 '{"AttributeName": "scanStartTime", "OrderBy": "DESC"}' --filter-criteria
 '{"FilterCriterion":[{"CriterionKey":"SCAN_ID", "FilterCondition":
{"EqualsValue":"123456789012"}}] }'
```

- このコマンドの応答には、最大 1 つの結果が表示され、(Infected の場合は) 影響を受けるリソースとマルウェアの検出結果に関する詳細が含まれます。

による GuardDuty サービスアカウント AWS リージョン

スナップショットが作成されて GuardDuty サービスアカウントと共有されると、CloudTrail ログに新しいイベントが作成されます。このイベントは、対応する snapshotIdと userId (その場合は GuardDuty サービスアカウント) を指定します AWS リージョン。詳細については、「GuardDuty が EBS ボリュームをスキャンしてマルウェアを検出する方法」を参照してください。

次の例は、ModifySnapshotAttribute リクエストのリクエスト本文を示す CloudTrail イベントのスニペットです。

```
"requestParameters": {
        "snapshotId": "snap-1234567890abcdef0",
        "createVolumePermission": {
            "add": {
                "items": [
                    {
                        "userId": "111122223333"
                    }
                ]
            }
        },
        "attributeType": "CREATE_VOLUME_PERMISSION"
    }
```

次の表は、各リージョンの GuardDuty サービスアカウントを示します。userId は GuardDuty サービスアカウントであり、選択したリージョンによって異なります。

AWS リージョン	リージョンコード	GuardDuty サービスアカウント ID (**userId**)
米国東部 (バージニア北部)	us-east-1	652050842985
米国東部 (オハイオ)	us-east-2	178123968615

AWS リージョン	リージョンコード	GuardDuty サービスアカウント ID (**`userId`**)
米国西部 (北カリフォルニア)	us-west-1	669213148797
米国西部 (オレゴン)	us-west-2	447226417196
アジアパシフィック (ムンバイ)	ap-south-1	913179291432
アジアパシフィック (大阪)	ap-northeast-3	089661699081
アジアパシフィック (ソウル)	ap-northeast-2	039163547507
アジアパシフィック (東京)	ap-northeast-1	874749492622
アジアパシフィック (シンガポール)	ap-southeast-1	247460962669
アジアパシフィック (シドニー)	ap-southeast-2	124839743349
カナダ (中部)	ca-central-1	175877067165
カナダ西部 (カルガリー)	ca-west-1	894794104037
欧州 (フランクフルト)	eu-central-1	002294850712
欧州 (アイルランド)	eu-west-1	283769539786
欧州 (ロンドン)	eu-west-2	310125036783
欧州 (パリ)	eu-west-3	866607715269
欧州 (ストックホルム)	eu-north-1	693780578038
中国 (北京)	cn-north-1	448721096076

AWS リージョン	リージョンコード	GuardDuty サービスアカウント ID (**`userId`**)
中国 (寧夏)	cn-northwest-1	480864352451
南米 (サンパウロ)	sa-east-1	546914126324
アジアパシフィック (ハイデラバード) (オプトイン)	ap-south-2	682251015962
アジアパシフィック (メルボルン) (オプトイン)	ap-southeast-4	353488359550
アジアパシフィック (マレーシア) (オプトイン)	ap-southeast-5	009160069308
アジアパシフィック (タイ) (オプトイン)	ap-southeast-7	941377115582
欧州 (スペイン) (オプトイン)	eu-south-2	936182149045
欧州 (チューリッヒ) (オプトイン)	eu-central-2	867642063380
イスラエル (テルアビブ) (オプトイン)	il-central-1	619233833001
欧州 (ミラノ) (オプトイン)	eu-south-1	977238331021
アジアパシフィック (香港) (オプトイン)	ap-east-1	249472122084
中東 (バーレーン) (オプトイン)	me-south-1	404001805210
アフリカ (ケープタウン) (オプトイン)	af-south-1	957664736811

AWS リージョン	リージョンコード	GuardDuty サービスアカウント ID (**`userId`**)
アジアパシフィック (ジャカルタ) (オプトイン)	ap-southeast-3	452118225523
中東 (アラブ首長国連邦) (オプトイン)	me-central-1	828603743433
メキシコ (中部) (オプトイン)	mx-central-1	557690616787
アジアパシフィック (台北)	ap-east-2	863518437308
アジアパシフィック (ニュージーランド)	ap-southeast-6	686255982852
AWS GovCloud (米国東部)	us-gov-east-1	226283551151
AWS GovCloud (米国西部)	us-gov-west-1	226300430612

Malware Protection for EC2 のクォータ

このセクションでは、Malware Protection for EC2 の使用に関連するクォータについて説明します。GuardDuty に関連付けられたクォータについては、「GuardDuty クォータ」を参照してください。

次の表は、Malware Protection for EC2 を使用する場合のさまざまなリソースのデフォルトの可用性を示しています。

範囲	デフォルト	コメント
圧縮ファイルまたはアーカイブファイル内のデータの抽出と分析	5	アーカイブファイルで許可されるネストレベルの最大数。
アーカイブされたファイル内のファイルの数	1,000	アーカイブ内でスキャンできるファイルの最大数。この数は、アーカイブから抽出されたファイルの数と、ネストされたすべてのアーカイブから抽出されたファイルの数の合計です。
脅威の数	32	検出結果パネルに表示できる脅威の最大数。GuardDuty Malware Protection for EC2 はさらに多くの脅威名を検出した可能性があります。検出された脅威名の数がデフォルト値よりも多い場合、GuardDuty コンソールの詳細パネルで検出結果名から [検出結果 ID] を選択すると、JSON の詳細を表示できます。
検出された脅威ごとのファイル数	5	検出された脅威ごとに識別されるファイルの最大数。例えば、GuardDuty が 1 つの脅威に関連する 10 個のファイルを検出した場合、その脅威は最大 5 個のファイルを表示します。
スキャン毎、インスタンス毎の EBS ボリューム	11	GuardDuty が EC2 インスタンスごとにスキャンできる EBS ボリュームの最

範囲	デフォルト	コメント
		大数。スキャンする必要がある EBS ボリュームが 11 以上ある場合、GuardDuty Malware Protection for EC2 は `deviceName` をアルファベット順に並べ替えた上で、最初の 11 個の EBS ボリュームを選択します。
EBS ボリュームサイズ	2048 GB	Amazon EC2 インスタンスとコンテナワークロードに関連付けられている GuardDuty Malware Protection for EC2 は、最大 2048 GB のサイズの各 Amazon EBS ボリュームをスキャンできます。このクォータは、Malware Protection for EC2 のサポート AWS リージョン が利用可能な各 に適用されます。

範囲	デフォルト	コメント
サポートされているファイルの種類	GuardDuty Malware Protection for EC2 は、次のファイルシステムの種類をスキャンできます。 • 新技術ファイルシステム (NTFS) • X ファイルシステム (XFS) • ext2 ファイルシステム • ext4 ファイルシステム • File Allocation Table (FAT) ファイルシステム • Virtual File Allocation Table (VFAT) ファイルシステム	該当なし
スキャンオプションタグ	50	マルウェアスキャンオプションの設定をカスタマイズするために追加できるリソースタグの最大数。詳細については、「ユーザー定義タグ付きのスキャンオプション」を参照してください。
保持期間の検索	90	GuardDuty が検出結果を保持する最大日数。最新情報については、「Amazon GuardDuty クォータ」を参照してください。

範囲	デフォルト	コメント
マルウェアウェアスキャンの保持期間	90	GuardDuty Malware Protection for EC2 がスキャンの履歴を保持する最大日数。最新のマルウェアスキャンを表示する方法については、「Malware Protection for EC2 のスキャンステータスと結果のモニタリング」を参照してください。
オンデマンドのマルウェアスキャンの 1 秒当たりのトランザクション数 (TPS)	1	各リージョンで 1 秒ごとに開始可能なオンデマンドのマルウェアスキャンリクエスト数。
オンデマンドのマルウェアスキャンのバースト制限	1	各リージョンで 1 秒ごとに同時に開始可能なオンデマンドのマルウェアスキャンリクエスト数。

GuardDuty Malware Protection for S3

Malware Protection for S3 は、選択した Amazon Simple Storage Service (Amazon S3) バケットに新しくアップロードされたオブジェクトをスキャンすることで、マルウェアが存在する可能性を検出できます。選択したバケットに S3 オブジェクトまたは既存の S3 オブジェクトの新しいバージョンがアップロードされると、GuardDuty は自動的にマルウェアスキャンを開始します。

Malware Protection for S3 - Overview and Demo

Malware Protection for S3 を有効にする 2 つの方法

S3 で GuardDuty サービス AWS アカウント を有効にし、GuardDuty エクスペリエンス全体の一部として Malware Protection for S3 を使用する場合、または GuardDutyサービスを有効にせずに Malware Protection for S3 機能を単独で使用する場合は、Malware Protection for S3 を有効にできます。Malware Protection for S3 を単独で有効にする場合のことを、GuardDuty ドキュメントでは、Malware Protection for S3 を独立した機能として使用すると表現します。

Malware Protection for S3 を独立して使用する際の考慮事項

- GuardDuty セキュリティ検出結果 – ディテクター ID は、リージョン内のアカウントに関連付けられている一意の識別子です。アカウントの 1 つ以上のリージョンで GuardDuty を有効に

すると、GuardDuty を有効にするリージョンごとに、このアカウントに対するディテクター ID が自動的に作成されます。詳細については、「Amazon GuardDuty の概念と主要な用語」ドキュメントの「ディテクター」を参照してください。

アカウントで Malware Protection for S3 を独立して有効にする場合、そのアカウントにはディテクター ID が関連付けられません。これは、使用可能な GuardDuty の機能に影響します。たとえば、S3 マルウェアスキャンがマルウェアの存在を検出すると、すべての GuardDuty 検出結果がディテクター ID に関連付けられている AWS アカウント ため、GuardDuty 検出結果はで生成されません。

- スキャンされたオブジェクトに悪意があるかどうかを確認する – デフォルトでは、GuardDuty はマルウェアスキャン結果をデフォルトの Amazon EventBridge イベントバスと Amazon CloudWatch 名前空間に発行します。バケットの Malware Protection for S3 を有効にするときにタグ付けを有効にすると、スキャンされた S3 オブジェクトはスキャン結果に言及するタグを取得します。タグ付けの詳細については、「必要に応じてスキャン結果に基づいてオブジェクトにタグを追加する」を参照してください。

Malware Protection for S3 を有効にする際の一般的な考慮事項

次の一般的な考慮事項は、Malware Protection for S3 を独立して使用する場合にも、GuardDuty エクスペリエンスの一部として使用する場合にも適用されます。

- 自分のアカウントに属する Amazon S3 バケットに対して Malware Protection for S3 を有効にできます。委任 GuardDuty 管理者アカウントとしては、メンバーアカウントに属する Amazon S3 バケットでこの機能を有効にすることはできません。
- この機能は、GuardDuty コンソールで現在選択されているリージョンと同じリージョンに属する S3 バケットで有効にできます。GuardDuty は、クロスリージョン S3 バケットでこの機能を有効にすることはサポートしていません。
- 委任 GuardDuty 管理者アカウントには、組織のメンバーアカウントの 1 つでこの機能を設定した S3 バケットの 保護されたバケットのステータスの表示と理解に変更があるたびに、Amazon EventBridge 通知が送信されます。

内容

- Malware Protection for S3 の料金と使用コスト
- Malware Protection for S3 の仕組み
- Malware Protection for S3 の機能
- (オプション) GuardDuty Malware Protection for S3 の使用を個別に開始する (コンソールのみ)
- バケットの Malware Protection for S3 の設定

- Malware Protection for S3 を有効にした後の手順
- GuardDuty でのオンデマンド S3 マルウェアスキャン
- Malware Protection for S3 によるタグベースのアクセスコントロール (TBAC) の使用
- 保護されたバケットのステータスの表示と理解
- Malware Protection for S3 での S3 オブジェクトスキャンのモニタリング
- トラブルシューティング
- 保護されたバケットの Malware Protection プランの編集
- 保護されたバケットの Malware Protection for S3 の無効化
- Amazon S3 機能のサポート状況
- Malware Protection for S3 のクォータ

Malware Protection for S3 の料金と使用コスト

Malware Protection for S3 の料金の仕組みは、GuardDuty の他の保護プランとは異なります。GuardDuty 保護プランのほとんどは 30 日間の短期無料トライアルに従いますが、Malware Protection for S3 には無料利用枠が含まれています AWS。GuardDuty の料金については、「GuardDuty の料金」を参照してください。

次に、Malware Protection for S3 を使用した場合に発生する料金コストを示します。

無料利用枠 (スキャンコスト)

各 は、各リージョンの 1 か月あたりの特定の制限までの使用量を含む無料利用枠 AWS アカウント を取得します。各 AWS アカウント は、最大 1,000 件のリクエストと 1 GB のスキャンされたデータを毎月無料利用枠で利用できます。使用量が指定された上限を超えると、その超えた分から使用コストが発生します。料金の詳細については、GuardDuty 保護プランの料金を参照してください。

> **Important**
>
> オンデマンドスキャンは無料利用枠には含まれません。

Malware Protection for S3 を有効にした後の使用コストについては、「Malware Protection for S3 の使用コストの確認」を参照してください。

S3 オブジェクトのタグ付けの使用コスト

Malware Protection for S3 を有効にするときに、スキャンされた S3 オブジェクトに対してタグ付けを有効にするかどうかは任意です。S3 オブジェクトのタグ付けを有効にすることにした場合、その使用に伴うコストが発生します。コストの詳細については、「Amazon S3 の料金」ページの「管理と洞察」タブを参照してください。

S3 オブジェクトタグ付けの使用コストは無料利用枠に含まれません。

Amazon S3 API - GET と PUT の使用コスト

GuardDuty が IAM ロールに基づいて Amazon S3 API を実行すると、使用コストが発生します。例えば、GuardDuty は IAM ロールを引き受けると、`PutObject` API を実行してテストオブジェクトを選択されたバケットに追加します。これにより、GuardDuty は機能の有効ステータスを評価できるようになります。

での S3 API コールの料金については AWS リージョン、Amazon S3 料金ページの「ストレージとリクエスト」タブの「リクエストとデータの取得」を参照してください。

Amazon S3 APIs の使用コストは無料利用枠には含まれません。

EventBridge マネージドルールのコスト

GuardDuty は EventBridge マネージドルールを使用して、保護されたバケットに新しいオブジェクトがいつアップロードされるかを検出します。EventBridge によって取り込まれた Amazon S3 オプトインイベントのコストが発生します。イベント配信コストは発生しません。EventBridge の料金の詳細については、「EventBridge の料金」を参照してください。

EventBridge マネージドルールのコストは無料利用枠に含まれません。

Malware Protection for S3 の使用コストの確認

Malware Protection for S3 を 無料利用枠の特定の制限を超えて使用すると、アカウントで使用コストが発生します。無料利用枠の詳細については、「」を参照してくださいMalware Protection for S3 の料金と使用コスト。無料利用枠は Malware Protection for S3 オンデマンドオブジェクトスキャンには適用されません。

GuardDuty コンソールでは、Malware Protection for S3 の使用コストを確認できません。使用コストを表示するには、https://console.aws.amazon.com/costmanagement/ コンソールで Cost Explorer に移動します。 AWS アカウント 請求の詳細については、AWS Billing 「 ユーザーガイド」を参照してください。

GuardDuty の推定使用コストについては、「使用状況のモニタリングとコストの見積もり」を参照してください。

Malware Protection for S3 の仕組み

このセクションでは、Malware Protection for S3 のコンポーネント、S3 バケットでコンポーネントを有効にした後の動作、マルウェアスキャンのステータスと結果を確認する方法について説明します。

概要

自分の AWS アカウントに属する Amazon S3 バケットに対して Malware Protection for S3 を有効にできます。GuardDuty では、この機能をバケット全体で有効にしたり、マルウェアスキャンの範囲を、選択したプレフィックスのいずれかで始まるアップロードされた各オブジェクトのコンテンツを GuardDuty がスキャンする特定のオブジェクトプレフィックスに制限したりできます。最大 5 個のプレフィックスを追加できます。S3 バケットに対してこの機能を有効にした場合、そのバケットは保護されたバケットと呼ばれます。

IAM ロールのアクセス許可

GuardDuty がユーザーに代わってマルウェアスキャンアクションを実行できるようにするには、Malware Protection for S3 で IAM ロールを使用します。マルウェアスキャンには、選択したバケットに新規にアップロードされたオブジェクトを通知する、そうしたオブジェクトをスキャンする、スキャンしたオブジェクトにオプションでタグを追加するといったアクションがあります。これは、この機能で S3 バケットを設定するための前提条件となります。

必要に応じて既存の IAM ロールを更新したり、そのために新しいロールを作成したりできます。複数のバケットに対して Malware Protection for S3 を有効にするときに、必要に応じて既存の IAM ロールを更新して他のバケットの名前を含めることができます。詳細については、「IAM ロールポリシーの作成または更新」を参照してください。

必要に応じてスキャン結果に基づいてオブジェクトにタグを追加する

スキャンされた S3 オブジェクトへのタグ付けを有効にできる手順があり、バケットに対して Malware Protection for S3 を有効にするときに必要に応じて実行できます。IAM ロールには、スキャン後にオブジェクトにタグを追加するためのアクセス許可が既に含まれています。ただし、GuardDuty でタグを追加できるのは、セットアップの時点でこのオプションを有効にした場合だけです。

オブジェクトがアップロードされる前に、このオプションを有効にする必要があります。スキャンが終了すると、次のキーと値のペアと共に、事前定義されたタグがスキャンされた S3 オブジェクトに追加されます。

`GuardDutyMalwareScanStatus:`*`Potential scan result`*

スキャン結果にタグとして追加されうる値には、`NO_THREATS_FOUND`、`THREATS_FOUND`、`UNSUPPORTED`、`ACCESS_DENIED`、`FAILED` などがあります。これらの値の詳細については、the section called “取りうる S3 オブジェクトスキャンステータスと結果ステータス” を参照してください。

S3 オブジェクトのスキャン結果を確認できる方法の 1 つが、タグ付けを有効にすることです。さらにこうしたタグを使用してタグベースのアクセスコントロール (TBAC) S3 リソースポリシーを追加することで、悪意の可能性があるオブジェクトに対策を講じることができます。詳細については、「S3 バケットリソースへの TBAC の追加」を参照してください。

バケットに対して Malware Protection for S3 を設定するときに、タグ付けを有効にすることをお勧めします。オブジェクトがアップロードされた後でタグ付けを有効にしても、スキャンが始まっている可能性があり、スキャンされたオブジェクトにタグが追加されることはありません。関連する S3 オブジェクトのタグ付けコストについては、「Malware Protection for S3 の料金と使用コスト」を参照してください。

バケットに対して Malware Protection for S3 を有効にした後のプロセス

Malware Protection for S3 を有効にすると、選択されている S3 バケット専用の Malware Protection プランリソースが作成されます。このリソースは、保護されたリソースを一意に識別する Malware Protection プラン ID に関連付けられます。GuardDuty は、IAM アクセス許可のいずれかを使用して、EventBridge マネージドルールを `DO-NOT-DELETE-AmazonGuardDutyMalwareProtectionS3*` という名前で作成および管理します。

GuardDuty のデータ処理の仕組み - データ保護のガードレール

Malware Protection for S3 は、Amazon EventBridge 通知をリッスンします。オブジェクトが選択されているバケットまたはいずれかのプレフィックスにアップロードされると、GuardDuty が AWS PrivateLink を使用して S3 バケットからそのオブジェクトをダウンロードし、同じリージョン内の分離された環境で読み取り、復号化してスキャンします。スキャン環境は、インターネットへのアクセスがないロックダウンされた仮想プライベートクラウド (VPC) で動作します。VPC は、 AWS が所有する許可リストに登録されたドメインへの通信のみを許可する DNS Firewall ルールグループにアタッチされます。スキャンが終了するまで、ダウンロードされた S3 オブジェクトがスキャン環境内に AWS Key Management Service (AWS KMS) キーで暗号化されて一時的に保存されます。

> **Note**
>
> デフォルトでは、「Amazon S3 ユーザーガイド」の「Object Created Event type」にリストされているすべての Amazon S3 API が Malware Protection for S3 スキャンを開始します。こうしたイベントタイプには、PutObject、POST Object、CopyObject、CompleteMultipartUpload などがあります。

GuardDuty マルウェア検出方法と使用するスキャンエンジンの詳細については、「GuardDuty マルウェア検出のスキャンエンジン」を参照してください。

マルウェアスキャンが完了すると、スキャンメタデータにスキャンステータスに関する処理が行われ、ダウンロードされたオブジェクトのコピーが削除されます。

新しいスキャンが始まる前に、GuardDuty は毎回スキャン環境をクリーンにします。オペレータがスキャン環境にアクセスする際には条件付き認可を実施し、すべてのアクセスリクエストが確認、承認、監査されます。

S3 オブジェクトスキャンのステータスと結果の確認

GuardDuty は、S3 オブジェクトスキャン結果イベントを Amazon EventBridge のデフォルトのイベントバスに発行します。また、スキャンされたオブジェクトの数やスキャンされたバイトといったスキャンメトリクスを Amazon CloudWatch に送信します。タグ付けを有効にすると、事前定義の `GuardDutyMalwareScanStatus` タグが追加されるほか、スキャン結果がタグ値として追加される可能性もあります。

> **Important**
>
> GuardDuty は at-least-once 配信を使用します。つまり、同じオブジェクトに対して複数のスキャン結果を受け取る可能性があります。アプリケーションは、重複した結果を処理できるように設計することをお勧めします。スキャンされた各オブジェクトには 1 回だけ課金されます。

詳細については、「Malware Protection for S3 での S3 オブジェクトスキャンのモニタリング」を参照してください。

生成された検出結果の確認

検出結果の確認は、GuardDuty で Malware Protection for S3 を使用しているかどうかによって異なります。次のシナリオを考えてみてください。

GuardDuty サービスが有効になっているときに Malware Protection for S3 を使用する (ディテクター ID)

マルウェアスキャンによって S3 オブジェクト内で悪意の可能性があるファイルが検出されると、それに応じた検出結果が生成されます。検出結果の詳細を確認し、推奨手順に従って検出結果を修復できる可能性があります。検出結果をエクスポートする頻度に基づいて、生成された検出結果が S3 バケットと EventBridge イベントバスにエクスポートされます。

生成される検出結果タイプについては、「Malware Protection for S3 の検出結果タイプ」を参照してください。

Malware Protection for S3 を独立した機能として使用する (ディテクター ID なし)

関連付けられたディテクター ID がないため、GuardDuty は検出結果を生成できません。S3 オブジェクトマルウェアスキャンのステータスを確認するには、デフォルトのイベントバスに自動的に発行されるスキャン結果を表示します。また、CloudWatch メトリクスを表示して、GuardDuty がスキャンしようとしたオブジェクトとバイトの数を評価することもできます。CloudWatch アラームを設定して、スキャン結果が通知されるようにすることができます。S3 オブジェクトのタグ付けを有効にしている場合は、S3 オブジェクトの `GuardDutyMalwareScanStatus` タグキーとスキャン結果のタグ値をチェックして、マルウェアスキャンのステータスを確認することもできます。

S3 オブジェクトスキャンのステータスと結果については、「Malware Protection for S3 での S3 オブジェクトスキャンのモニタリング」を参照してください。

Malware Protection for S3 の機能

次に、バケットに対して Malware Protection for S3 を有効にすると、何を期待し、何をできるのかを簡単に説明します。

- スキャン対象の選択 - 選択した S3 バケットに関連付けられたすべてまたは特定のプレフィックス (最大 5 つ) にファイルがアップロードされると、そのファイルをスキャンします。
- アップロードされたオブジェクトに対する自動スキャン - バケットに対して Malware Protection for S3 を有効にすると、GuardDuty がスキャンを自動的に開始して、新規にアップロードされたオブジェクトにマルウェアが潜んでいないかを検出します。

- オンデマンドスキャン – 既存のオブジェクトのスキャンを開始するか、以前にスキャンしたオブジェクトを再スキャンできます。詳細については、「GuardDuty でのオンデマンド S3 マルウェアスキャン」を参照してください。
- コンソール、API/AWS CLI、または を使用して を有効にする CloudFormation – Malware Protection for S3 を有効にする任意の方法を選択します。

  Terraform などの Infrastructure as Code (IaC) プラットフォームを使用して、Malware Protection for S3 を有効にできます。詳細については、「リソース: aws_guardduty_malware_protection_plan」を参照してください。
- サポートされているファイル形式、Malware Protection for S3 クォータ、Amazon S3 機能 - Malware Protection for S3 は、S3 バケットにアップロードできるすべてのファイル形式をサポートしています。アップロードされたファイルがパスワードで保護されており、かつ GuardDuty がアップロードされたファイルのタイプに対してパスワード保護の存在を検出できる場合、GuardDuty は共通のパスワードを使用して元のコンテンツのスキャンを試みます。パスワードが失敗した場合、スキャンはスキップされます。GuardDuty は、すべてのファイル形式におけるパスワード保護の有無を検出できません。GuardDuty がパスワード保護の有無を検出できない場合でも、GuardDuty は暗号化されたコンテンツをスキャンします。

  オブジェクトサイズ、アーカイブの最大深度、その他の詳細に関連するクォータについては、「Malware Protection for S3 のクォータ」を参照してください。

  Amazon S3 機能のサポート状況については、「Amazon S3 機能のサポート状況」を参照してください。
- スキャンされた S3 オブジェクトに対するタグ付けのサポート -必要に応じてスキャン結果に基づいてオブジェクトにタグを追加するを実施すると、マルウェアスキャンのたびにスキャンのステータスを示すタグが追加されます。このタグを使用すると、S3 オブジェクトに対してタグベースのアクセスコントロール (TBAC) を設定できます。例えば、悪意のあることが示され、タグ値が `THREATS_FOUND` である S3 オブジェクトへのアクセスを制限できます。
- Amazon EventBridge 通知 - Malware Protection プランのリソースステータスが変わるか S3 オブジェクトのマルウェアスキャンが完了すると、GuardDuty は Amazon EventBridge にイベントを送信します。これらのイベントは、デフォルトのイベントバスに送信されます。EventBridge とこうしたイベントを使用して、そのイベントの発生をモニタリングするといった対策を講じるルールを記述できます。詳細については、「Amazon EventBridge による S3 オブジェクトスキャンのモニタリング」を参照してください。

- CloudWatch メトリクス - CloudWatch メトリクスを表示して、特定のマルウェアスキャンステータスに対してアラームを有効にします。詳細については、「CloudWatch での S3 オブジェクトスキャンのステータスメトリクス」を参照してください。

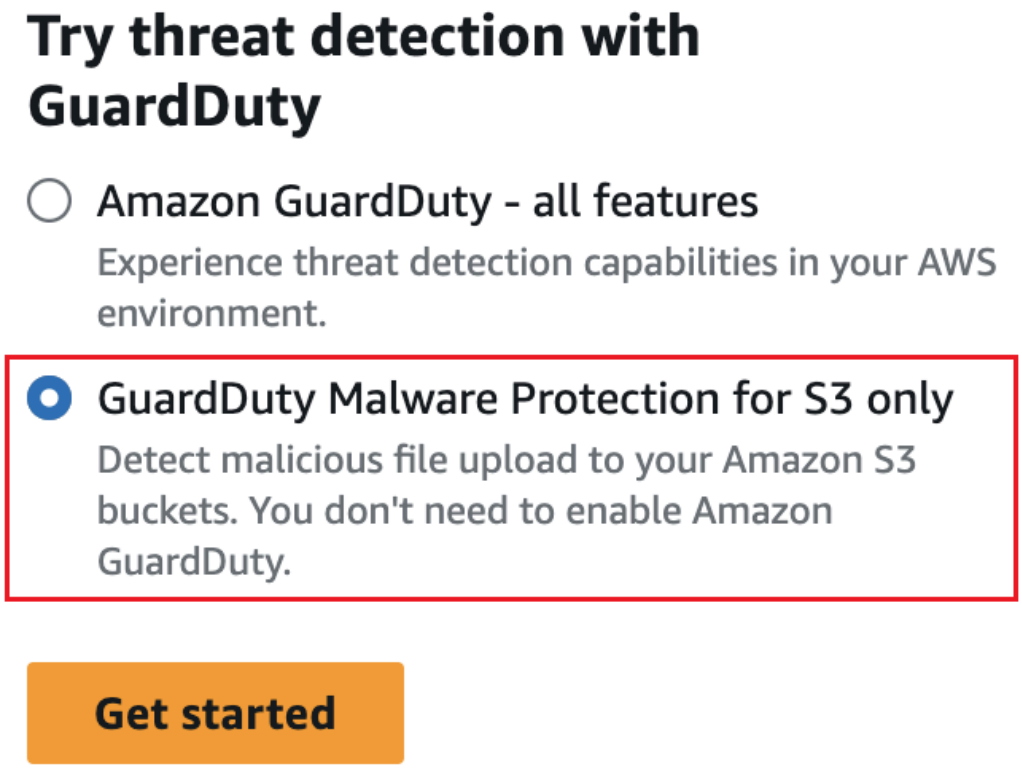
(オプション) GuardDuty Malware Protection for S3 の使用を個別に開始する (コンソールのみ)

このオプションの手順は、 AWS アカウントの GuardDuty ステータスとは関係なく Malware Protection for S3 脅威検出オプションの使用を開始する場合に使用します。

GuardDuty で他の専用の保護プランも使用する場合は、Amazon GuardDuty サービスの使用を開始する必要があります。GuardDuty 保護プランについては、「GuardDuty の機能」を参照してください。アカウントで GuardDuty を既に有効にしているときは、この手順をスキップして「バケットの Malware Protection for S3 の設定」を続行できます。

Malware Protection for S3 専用脅威検出の使用を開始するための手順

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. [GuardDuty Malware Protection for S3 のみ] を選択します。これにより、Amazon Simple Storage Service (Amazon S3) バケットに新規にアップロードされたファイルにマルウェアが含まれているかどうかを検出できます。



3. [開始する] を選択します。これで「バケットの Malware Protection for S3 の設定」の手順を続行できるようになりました。

バケットの Malware Protection for S3 の設定

Malware Protection for S3 が S3 オブジェクトをスキャンおよび (オプションで) タグを追加するには、ユーザーに代わってマルウェアスキャンアクションを実行するために必要なアクセス許可を持つサービスロールを使用できます。サービスロールを使用して Malware Protection for S3 を有効にする方法の詳細については、「Service Access」を参照してください。このロールは、GuardDuty Malware Protection サービスにリンクされたロールとは異なります。

IAM ロールを使用する場合は、S3 オブジェクトをスキャンおよび (オプションで) タグを追加するのに必要なアクセス許可を含む IAM ロールをアタッチできます。その後、GuardDuty はこの IAM ロールを引き受けて、ユーザーに代わってこれらのアクションを実行します。Amazon S3 バケットでこの保護プランを有効にするときは、この IAM ロール名が必要です。

IAM ロールを使用している場合は、Amazon S3 バケットを保護するたびに、このセクションに記載された両方のステップを実行する必要があります。

Malware Protection for S3 を有効にするには、S3 バケット名、オブジェクトプレフィックス (特定のプレフィックスを集中的に保護したい場合)、必要なアクセス許可を持つ IAM ロール名などの詳細が必要です。

Malware Protection for S3 を個別に開始する場合でも、GuardDuty サービスの一部として有効にする場合でも、ステップは同じです。

トピック

1. IAM ロールポリシーの作成または更新
2. バケットの Malware Protection for S3 の有効化
3. IAM ロールのアクセス許可エラーのトラブルシューティング

バケットの Malware Protection for S3 の有効化

このセクションでは、独自のアカウントでバケットの Malware Protection for S3 を有効にする方法の詳細な手順について説明します。以下の考慮事項を確認してから作業を進めてください。

- GuardDuty コンソールを使用してこの保護プランを有効にすると、[サービスアクセス] セクションで新しいロールを作成するか、既存のロールを使用するステップが含まれます。
- GuardDuty API または CLI を使用してこの保護プランを有効にする場合は、先に進む前に IAM ロールポリシーの作成または更新 を実行する必要があります。
- この保護プランを有効にする方法にかかわらず、必要な Malware Protection プランリソースを作成するためのアクセス許可 が必要です。

Amazon S3 バケットスロットリングの検討

S3 スロットリングは、Amazon S3 バケットとの間でデータを転送できるレートを制限する場合があります。これは、新しくアップロードしたオブジェクトのマルウェアスキャンを遅らせる可能性があります。

S3 バケットへの大量の GETおよび PUTリクエストが予想される場合は、スロットリングを防ぐための対策の実施を検討してください。これを行う方法については、「Amazon Athena ユーザーガイド」の「Amazon S3 スロットリングを防ぐ」を参照してください。

Malware Protection プランリソースを作成するためのアクセス許可

Amazon S3 バケットで Malware Protection for S3 を有効にすると、GuardDuty はバケットの保護プランの識別子として機能する Malware Protection プランリソースを作成します。AWS 管理ポリシー: AmazonGuardDutyFullAccess_v2 (推奨) をまだ使用していない場合は、次のアクセス許可を追加して、このリソースを作成する必要があります。

- `guardDuty:CreateMalwareProtectionPlan`
- `iam:PassRole`

以下のカスタムポリシーの例を使用して、*#########*を、アカウントに適した値に置き換えることができます。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ],
            "Resource": "arn:aws:iam::111122223333:role/role-name",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "malware-protection-
plan.guardduty.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "guardduty:CreateMalwareProtectionPlan"
            ],
            "Resource": "*"
        }
    ]
}
```

GuardDuty コンソールを使用して Malware Protection for S3 を有効にする

以下のセクションでは、GuardDuty コンソールで体験できるステップバイステップのチュートリアルを提供します。

GuardDuty コンソールを使用して Malware Protection for S3 を有効にするには

S3 バケットの詳細を入力する

次の手順に従って、Amazon S3 バケットの詳細を指定します。

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ページの右上隅にある AWS リージョン セレクターを使用して、Malware Protection for S3 を有効にするリージョンを選択します。
3. ナビゲーションペインで、[Malware Protection for S3] を選択します。
4. [保護されたバケット] セクションで、[有効化] を選択して、独自の AWS アカウントに属する S3 バケットの Malware Protection for S3 を有効にします。
5. [S3 バケットの詳細を入力する] で、Amazon S3 バケット名を入力します。または、[S3 を参照] を選択して S3 バケットを選択します。

   Malware Protection for S3 AWS アカウント を有効にする と S3 バケット AWS リージョン の は同じである必要があります。例えば、アカウントが `us-east-1` リージョンに属している場合、Amazon S3 バケットリージョンも `us-east-1` である必要があります。
6. [プレフィックス] では、[S3 バケット内のすべてのオブジェクト] または [特定のプレフィックスで始まるオブジェクト] を選択できます。

   - 選択したバケットに新しくアップロードされたすべてのオブジェクトを GuardDuty がスキャンできるようにするときは、[S3 バケット内のすべてのオブジェクト] を選択します。
   - 特定のプレフィックスに属する新しくアップロードされたオブジェクトをスキャンするときは、[特定のプレフィックスで始まるオブジェクト] を選択します。このオプションは、選択したオブジェクトプレフィックスのみにマルウェアスキャンの範囲を集中させるのに役立ちます。プレフィックスの使用の詳細については、「Amazon S3 ユーザーガイド」の「フォルダを使用して Amazon S3 コンソールのオブジェクトを整理する」を参照してください。

     [プレフィックスの追加] を選択し、プレフィックスを入力します。最大 5 つのプレフィックスを追加できます。

スキャンされたオブジェクトのタグ付けを有効にする

これは任意の手順です。オブジェクトがバケットにアップロードされる前にタグ付けオプションを有効にした場合、スキャンが完了すると、GuardDuty はキーを `GuardDutyMalwareScanStatus`、値をスキャン結果として事前定義されたタグを追加します。Malware Protection for S3 を最適に使用するには、スキャン終了後に S3 オブジェクトにタグを追加するオプションを有効にすることをお勧めします。標準の S3 オブジェクトのタグ付けコストが適用されます。詳細については、「Malware Protection for S3 の料金と使用コスト」を参照してください。

タグ付けを有効にする理由

- タグ付けを有効にすることは、マルウェアスキャンの結果を知る方法の 1 つです。S3 マルウェアスキャン結果の詳細については、「Malware Protection for S3 での S3 オブジェクトスキャンのモニタリング」を参照してください。
- 悪意の可能性のあるオブジェクトを含む S3 バケットにタグベースのアクセスコントロール (TBAC) ポリシーを設定します。考慮事項とタグベースのアクセスコントロール (TBAC) の実装方法については、「Malware Protection for S3 によるタグベースのアクセスコントロール (TBAC) の使用」を参照してください。

GuardDuty による S3 オブジェクトへのタグの追加に関する考慮事項:

- デフォルトでは、1 つのオブジェクトに最大 10 個のタグを関連付けることができます。詳細については、「Amazon S3 ユーザーガイド」の「タグを使用してストレージを分類する」を参照してください。

  10 個すべてのタグが既に使用されている場合、GuardDuty はスキャンされたオブジェクトに事前定義されたタグを追加できません。GuardDuty はまた、スキャン結果をデフォルトの EventBridge イベントバスにも発行します。詳細については、「Amazon EventBridge による S3 オブジェクトスキャンのモニタリング」を参照してください。
- 選択した IAM ロールに GuardDuty が S3 オブジェクトにタグ付けするアクセス許可が含まれていない場合は、保護されたバケットに対してタグ付けが有効になっていても、GuardDuty はこのスキャンされた S3 オブジェクトにタグを追加できません。タグ付けに必要な IAM ロールのアクセス許可の詳細については、「IAM ロールポリシーの作成または更新」を参照してください。

  GuardDuty はまた、スキャン結果をデフォルトの EventBridge イベントバスにも発行します。詳細については、「Amazon EventBridge による S3 オブジェクトスキャンのモニタリング」を参照してください。

[スキャンしたオブジェクトのタグ付け] でオプションを選択するには

- GuardDuty でスキャンした S3 オブジェクトにタグを追加する場合は、[オブジェクトをタグ付けする] を選択します。
- GuardDuty でスキャンした S3 オブジェクトにタグを追加しない場合は、[オブジェクトにタグ付けしない] を選択します。

サービスアクセス

次の手順に従って、既存のサービスロールを選択するか、ユーザーに代わってマルウェアスキャンアクションを実行するために必要なアクセス許可を持つ新しいサービスロールを作成します。これらのアクションには、新しくアップロードされた S3 オブジェクトのスキャンや、(オプションで) これらのオブジェクトへのタグの追加が含まれる場合があります。このロールが持つアクセス許可ポリシーについては、「IAM ロールポリシーの作成または更新」を参照してください。

[サービスアクセス] セクションで、次のいずれかを実行できます。

1. 新しいサービスロールを作成して使用する — マルウェアスキャンを実行するために必要なアクセス許可を持つ新しいサービスロールを作成および使用できます。

   [ロール名] では、GuardDuty によって事前入力された名前を使用するか、任意の意味のある名前を入力してロールを識別できます。例: `GuardDutyS3MalwareScanRole`。ロール名は 1～64 文字にする必要があります。有効な文字は、a～z、A～Z、0～9、および「+=,.@-_」です。
2. 既存のサービスロールを使用する — [サービスロール名] リストから既存のサービスロールを選択できます。

   a. [ポリシーテンプレート] では、S3 バケットのポリシーを表示できます。[S3 バケットの詳細を入力する] セクションで S3 バケットを入力または選択したことを確認します。

   b. [サービスロール名] で、サービスロールのリストからサービスロールを選択します。

要件に基づいてポリシーを変更することができます。IAM ロールの作成または更新方法の詳細については、「Create or update IAM role policy」を参照してください。

IAM ロールのアクセス許可に関する問題については、「IAM ロールのアクセス許可エラーのトラブルシューティング」を参照してください。

(オプション) Malware Protection プラン ID にタグを付ける

これは、S3 バケットリソース用に作成される Malware Protection プランのリソースにタグを追加するのに役立つオプションの手順です。

各タグには、タグキーとオプションのタグ値という 2 つの部分があります。タグ付けとその利点の詳細については、「 AWS リソースのタグ付け」を参照してください。

Malware Protection プランのリソースにタグを追加するには

1. タグのキーと、オプションで値を入力します。タグキーとタグ値は両方とも大文字と小文字が区別されます。タグキーとタグ値の名前については、「タグの命名制限と要件」を参照してください。
2. Malware Protection プランのリソースにタグを追加するには、[新しいタグを追加] を選択し、前のステップを繰り返します。 リソースごとに最大 50 個のタグを追加できます。
3. [有効化] を選択します。

API/CLI を使用して Malware Protection for S3 を有効にする

このセクションでは、 AWS 環境で Malware Protection for S3 をプログラムで有効にする手順について説明します。これには、「IAM ロールポリシーの作成または更新」のステップで作成した IAM ロールの Amazon リソースネーム (ARN) が必要です。

API/CLI を使用してプログラムにより Malware Protection for S3 を有効にするには

- API を使用する

  CreateMalwareProtectionPlan を実行して、自分のアカウントに属するバケットの Malware Protection for S3 を有効にします。

- を使用する AWS CLI

  Malware Protection for S3 を有効にする方法に応じて、次のリストに特定のユースケースのコマンド AWS CLI 例を示します。これらのコマンドを実行するときは、##################を、アカウントに適した値に置き換えます。

  AWS CLI コマンドの例

  - 次の AWS CLI コマンドを使用して、スキャンされた S3 オブジェクトのタグ付けがないバケットに対して Malware Protection for S3 を有効にします。

```
aws guardduty create-malware-protection-plan --role
 "arn:aws:iam::111122223333:role/role-name" --protected-resource
 "S3Bucket"={"BucketName"="amzn-s3-demo-bucket1"}
```

- 次の AWS CLI コマンドを使用して、特定のオブジェクトプレフィックスがあり、スキャンされた S3 オブジェクトのタグ付けがないバケットに対して Malware Protection for S3 を有効にします。

```
aws guardduty create-malware-protection-plan --role
 "arn:aws:iam::111122223333:role/role-name" --protected-resource '{"S3Bucket":
{"BucketName":"amzn-s3-demo-bucket1", "ObjectPrefixes": ["Object1","Object1"]}}'
```

- 次の AWS CLI コマンドを使用して、スキャンされた S3 オブジェクトのタグ付けが有効になっているバケットの Malware Protection for S3 を有効にします。

```
aws guardduty create-malware-protection-plan --role
 "arn:aws:iam::111122223333:role/role-name" --protected-resource
 "S3Bucket"={"BucketName"="amzn-s3-demo-bucket1"} --actions
 "Tagging"={"Status"="ENABLED"}
```

これらのコマンドの実行が成功すると、一意の Malware Protection プラン ID が生成されます。バケットの保護プランの更新や無効化などのアクションを実行するには、この Malware Protection プラン ID が必要になります。

IAM ロールのアクセス許可に関する問題については、「IAM ロールのアクセス許可エラーのトラブルシューティング」を参照してください。

IAM ロールポリシーの作成または更新

Malware Protection for S3 が S3 オブジェクトをスキャンおよび (オプションで) タグを追加するには、ユーザーに代わってマルウェアスキャンアクションを実行するために必要なアクセス許可を持つサービスロールを使用できます。サービスロールを使用して Malware Protection for S3 を有効にする方法の詳細については、「Service Access」を参照してください。このロールは、GuardDuty Malware Protection サービスにリンクされたロールとは異なります。

IAM ロールを使用する場合は、S3 オブジェクトをスキャンおよび (オプションで) タグを追加するのに必要なアクセス許可を含む IAM ロールをアタッチできます。こうしたアクセス許可を含めるには、IAM ロールを作成するか既存のロールを更新する必要があります。Malware Protection for S3

を有効にしている Amazon S3 バケットごとにこうしたアクセス許可が必要になるため、保護する Amazon S3 バケットごとにこの手順を実行する必要があります。

次に、アクセス許可を使用すると GuardDuty がユーザーに代わってどのようにマルウェアスキャンアクションを実行できるかを示します。

- Amazon EventBridge アクションにより、Malware Protection for S3 が S3 オブジェクトからの通知をリッスンできるように EventBridge マネージドルールを作成および管理できます。

  詳細については、「Amazon EventBridge ユーザーガイド」の「Amazon EventBridge マネージドルール」を参照してください。
- Amazon S3 アクションと EventBridge アクションにより、このバケットで発生するすべてのイベントについて通知を EventBridge に送信できます。

  詳細については、「Amazon S3 ユーザーガイド」の「Enabling Amazon EventBridge」を参照してください。
- Amazon S3 アクションにより、アップロードされた S3 オブジェクトにアクセスして、スキャンされた S3 オブジェクトに事前定義の `GuardDutyMalwareScanStatus` タグを追加できます。オブジェクトプレフィックスを使用するときは、ターゲットのプレフィックスに対してのみ `s3:prefix` 条件を追加してください。これにより、GuardDuty がバケット内のすべての S3 オブジェクトにアクセスするのを防ぐことができます。
- KMS キーのアクションにより、テストオブジェクトをスキャンしてバケットに配置する前にサポートされている DSSE-KMS と SSE-KMS による暗号化を使用してオブジェクトにアクセスできます。

> **Note**
>
> この手順は、アカウントのバケットに対して Malware Protection for S3 を有効にするたびに必要です。既に IAM ロールが存在する場合は、そのポリシーを更新して別の Amazon S3 バケットリソースの詳細を含めることができます。その方法を示した例については、「IAM ポリシーのアクセス許可の追加」を参照してください。

IAM ロールを作成または更新するには、次のポリシーを使用します。

ポリシー

- IAM ポリシーのアクセス許可の追加

- 信頼関係ポリシーの追加

IAM ポリシーのアクセス許可の追加

既存の IAM ロールのインラインポリシーを更新することも、新しい IAM ロールを作成することもできます。その手順については、「IAM ユーザーガイド」の「IAM ロールの作成」または「ロールのアクセス許可ポリシーの変更」を参照してください。

次に示すアクセス許可のテンプレートを任意の IAM ロールに追加します。次のプレースホルダーの値をアカウントに関連付けられた適切な値に置き換えます。

- *amzn-s3-demo-bucke* を Amazon S3 バケット名に置き換えます。

  複数の S3 バケットリソースに同じ IAM ロールを使用するには、次の例に示すとおりに既存のポリシーを更新します。

```
...
...
"Resource": [
    "arn:aws:s3:::amzn-s3-demo-bucket/*",
    "arn:aws:s3:::amzn-s3-demo-bucket2/*"
],
...
...
```

  S3 バケットに関連付けられた新しい ARN を追加する前に、必ずカンマ (,) を追加してください。ポリシーテンプレートで S3 バケット `Resource` を参照する場合は常にこれを行ってください。
- *111122223333* を AWS アカウント ID に置き換えます。
- *us-east-1* を AWS リージョンに置き換えます。
- *APKAEIBAERJR2EXAMPLE* の場合、カスタマーマネージド型キー ID に置き換えます。S3 バケットが AWS KMS キーを使用して暗号化されている場合、バケットのマルウェア保護を設定するときに新しいロールの作成オプションを選択すると、関連するアクセス許可が追加されます。

```
"Resource": "arn:aws:kms:us-east-1:111122223333:key/*"
```

IAM ロールポリシーテンプレート

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
            "Sid": "AllowManagedRuleToSendS3EventsToGuardDuty",
            "Effect": "Allow",
            "Action": [
                "events:PutRule",
                "events:DeleteRule",
                "events:PutTargets",
                "events:RemoveTargets"
            ],
            "Resource": [
                "arn:aws:events:us-east-1:111122223333:rule/DO-NOT-DELETE-
AmazonGuardDutyMalwareProtectionS3*"
            ],
            "Condition": {
                "StringLike": {
                    "events:ManagedBy": "malware-protection-
plan.guardduty.amazonaws.com"
                }
            }
        },
        {
            "Sid": "AllowGuardDutyToMonitorEventBridgeManagedRule",
            "Effect": "Allow",
            "Action": [
                "events:DescribeRule",
                "events:ListTargetsByRule"
            ],
            "Resource": [
                "arn:aws:events:us-east-1:111122223333:rule/DO-NOT-DELETE-
AmazonGuardDutyMalwareProtectionS3*"
            ]
        },
        {
            "Sid": "AllowPostScanTag",
            "Effect": "Allow",
            "Action": [
                "s3:PutObjectTagging",
                "s3:GetObjectTagging",
                "s3:PutObjectVersionTagging",
```

```
                "s3:GetObjectVersionTagging"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        },
        {
            "Sid": "AllowEnableS3EventBridgeEvents",
            "Effect": "Allow",
            "Action": [
                "s3:PutBucketNotification",
                "s3:GetBucketNotification"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket"
            ]
        },
        {
            "Sid": "AllowPutValidationObject",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket/malware-protection-resource-
validation-object"
            ]
        },
        {
            "Sid": "AllowCheckBucketOwnership",
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket"
            ]
        },
        {
           "Sid": "AllowMalwareScan",
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:GetObjectVersion"
```

```
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
        },
        {
            "Sid": "AllowDecryptForMalwareScan",
            "Effect": "Allow",
            "Action": [
                "kms:GenerateDataKey",
                "kms:Decrypt"
            ],
            "Resource": "arn:aws:kms:us-
east-1:111122223333:key/APKAEIBAERJR2EXAMPLE",
            "Condition": {
                "StringLike": {
                    "kms:ViaService": "s3.us-east-1.amazonaws.com"
                }
            }
        }
    ]
}
```

信頼関係ポリシーの追加

IAM ロールに次の信頼ポリシーをアタッチします。その手順については、「ロールの信頼ポリシーの変更」を参照してください。

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "malware-protection-plan.guardduty.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
```

```
}
```

IAM ロールのアクセス許可エラーのトラブルシューティング

Malware Protection for S3 を有効にすると、GuardDuty は IAM サービスロールに Amazon S3 バケットの所有権を検証するために必要なアクセス許可があるかどうかをチェックします。これらのアクセス許可がないか、正しく設定されていない場合は、以下のメッセージが表示されます。

```
"message": "The request was rejected because provided IAM role does not have the
 required permissions to validate S3 bucket ownership."
"type": "InvalidInputException"
```

以下のシナリオは、このエラーのトラブルシューティングに役立ちます。

IAM ロールのアクセス許可がない

- IAM ロールには、Malware Protection for S3 がロールを引き受けられるようにするために必要なアクセス許可が必要です。
- GuardDuty は、`"s3:ListBucket"` アクセス許可を使用してバケットの所有権を検証します。これは、使用する IAM ロールに存在する必要があります。

アクセス許可については、「IAM ロールポリシーの作成または更新」を参照してください。

IAM ロールの可用性

- 新しい IAM ロールを作成するときは、Malware Protection for S3 を有効にする前に、変更が結果整合性に達するまで数分かかります。ロールの作成直後に保護プランを有効にしようとすると、検証が失敗する可能性があります。
- Infrastructure as Code (IaC) デプロイの場合、GuardDuty は IAM ロールが最終的な一貫性に達するようにリソースの依存関係を宣言することをお勧めします。

  これを行う方法のサンプルテンプレートについては、「GuardDuty GitHub リポジトリ」を参照してください。

クロスリージョン有効化

Amazon S3 バケットが、GuardDuty で Malware Protection for S3 を有効にするリージョンと同じリージョンにあることを確認します。

Malware Protection for S3 を有効にした後の手順

このセクションでは、バケットに対して Malware Protection for S3 を有効にした後で実行できる手順を示します。次の手順へとスムーズに進める順序になっています。

バケットに対して Malware Protection for S3 を有効にした後で従うべき手順

1. タグベースのアクセスコントロール (TBAC) リソースポリシーを追加する - タグ付けを有効にしたときは、選択されたバケットにオブジェクトがアップロードされる前に、TBAC ポリシーを S3 バケットリソースに追加してください。詳細については、「S3 バケットリソースへの TBAC の追加」を参照してください。
2. Malware Protection プランのステータスをモニタリングする - 保護された各バケットの [ステータス] 列をモニタリングします。取りうるステータスとその意味については、「保護されたバケットのステータスの表示と理解」を参照してください。
3. 次のいずれかのオプションを選択して、スキャンを開始します。
   - オブジェクトをアップロードする:
     1. Amazon S3 コンソール (https://console.aws.amazon.com/s3/) を開きます。
     2. この機能を有効にした S3 バケットまたはオブジェクトプレフィックスにファイルをアップロードします。ファイルをアップロードする手順については、「Amazon S3 ユーザーガイド」の「バケットにオブジェクトをアップロードする」を参照してください。
   - オンデマンドスキャンを開始します。 GuardDuty でのオンデマンド S3 マルウェアスキャン
4. S3 オブジェクトスキャンのステータスと結果をモニタリングする - この手順では、S3 オブジェクトマルウェアスキャンのステータスを確認する方法について説明します。

GuardDuty と Malware Protection for S3 の両方を有効化	Malware Protection for S3 のみを有効化
• GuardDuty が有効になっていると Malware Protection for S3 の検出結果タイプが生成されることがあり、これはスキャンされた S3 オブジェクトにマルウェアが存在していることを示します。 • 「Malware Protection for S3 での S3 オブジェクトスキャンのモニタリング」に示された 1 つ以上のオプションを使用し	「Malware Protection for S3 での S3 オブジェクトスキャンのモニタリング」に示された 1 つ以上のオプションを使用して、S3 オブジェクトスキャンの結果を確認できる可能性があります。具体的には、Amazon EventBridge を使用する、Malware Protection プランに CloudWatch メトリクスを使用する、スキャ

GuardDuty と Malware Protection for S3 の両方を有効化	Malware Protection for S3 のみを有効化
て、S3 オブジェクトスキャンの結果を確認できる可能性があります。具体的には、Amazon EventBridge を使用する、Malware Protection プランに CloudWatch メトリクスを使用する、スキャンされたオブジェクトにタグ付けするといったオプションがあります。	ンされたオブジェクトにタグ付けするといったオプションがあります。

GuardDuty でのオンデマンド S3 マルウェアスキャン

GuardDuty Malware Protection for S3 は、新しい S3 アップロードを継続的にモニタリングします。保護を有効にする前に存在していたオブジェクト、または以前にスキャンしたオブジェクトを再スキャンするには、バケットの GuardDuty Malware Protection プランを有効にしたら、オンデマンド S3 マルウェアスキャンを開始できます。

オンデマンドマルウェアスキャンでは、Malware Protection Plan の IAM ロールを使用してオブジェクトへのアクセスと設定の適用を行います。スキャンは、バケットの Malware Protection Plan で設定されたプレフィックスを上書きします。

> **Note**
>
> Malware Protection for S3 のクォータは、オンデマンドのマルウェアスキャンに適用されます。詳細については、「」を参照してくださいMalware Protection for S3 のクォータ。
> 料金の詳細については、「Malware Protection for S3 の料金と使用コスト」を参照してください。

前提条件

オンデマンドマルウェアスキャンを開始する前に、アカウントが次の前提条件を満たす必要があります。

- Malware Protection for S3 は、ターゲットバケットで有効になっています。詳細については「バケットの Malware Protection for S3 の設定」を参照してください。

- AWS 管理ポリシー: AmazonGuardDutyFullAccess_v2 (推奨) ポリシーは、API を呼び出す IAM ユーザーまたは IAM ロールにアタッチされます。

オンデマンドマルウェアスキャンを開始する

SendObjectMalwareScan API オペレーションを使用します。これには、入力として S3 オブジェクトパスが必要です。

API/CLI

オブジェクトの最新バージョンをスキャンするか、スキャンする特定のバージョンを指定できます。

オブジェクトの特定のバージョンをスキャンするには:

```
aws guardduty send-object-malware-scan --s3-object '{"Bucket":
 "amzn-s3-demo-bucket", "Key": "APKAEIBAERJR2EXAMPLE", "VersionId":
 "d41d8cd98f00b204e9800998eEXAMPLE"}'
```

オブジェクトの最新バージョンをスキャンするには:

```
aws guardduty send-object-malware-scan --s3-object '{"Bucket": "amzn-s3-demo-
bucket", "Key": "APKAEIBAERJR2EXAMPLE"}'
```

> **Important**
>
> API コールが成功すると、スキャンリクエストが承諾されたことが確認されます。ただし、スキャン結果をモニタリングして正常な完了を確認し、オブジェクトへのアクセスエラーなどの問題を特定することが重要です。詳細については、「Malware Protection for S3 での S3 オブジェクトスキャンのモニタリング」を参照してください。

Malware Protection for S3 によるタグベースのアクセスコントロール (TBAC) の使用

バケットに対して Malware Protection for S3 を有効にするときに、必要に応じてタグ付けを有効にできます。選択されているバケットに新規にアップロードされた S3 オブジェクトをスキャンしよう

とすると、マルウェアスキャンのステータスを確認できるように、スキャンされたオブジェクトにタグが追加されます。タグ付けを有効にすると、すぐに使用コストが発生します。詳細については、「Malware Protection for S3 の料金と使用コスト」を参照してください。

GuardDuty は、キー付きの事前定義タグに `GuardDutyMalwareScanStatus` を使用し、値にマルウェアスキャンのステータスのいずれかを使用します。これらの値については、「the section called "取りうる S3 オブジェクトスキャンステータスと結果ステータス"」を参照してください。

GuardDuty による S3 オブジェクトへのタグの追加に関する考慮事項:

- デフォルトでは、1 つのオブジェクトに最大 10 個のタグを関連付けることができます。詳細については、「Amazon S3 ユーザーガイド」の「タグを使用してストレージを分類する」を参照してください。

  10 個すべてのタグが既に使用されている場合、GuardDuty はスキャンされたオブジェクトに事前定義されたタグを追加できません。GuardDuty はまた、スキャン結果をデフォルトの EventBridge イベントバスにも発行します。詳細については、「Amazon EventBridge による S3 オブジェクトスキャンのモニタリング」を参照してください。
- 選択した IAM ロールに GuardDuty が S3 オブジェクトにタグ付けするアクセス許可が含まれていない場合は、保護されたバケットに対してタグ付けが有効になっていても、GuardDuty はこのスキャンされた S3 オブジェクトにタグを追加できません。タグ付けに必要な IAM ロールのアクセス許可の詳細については、「IAM ロールポリシーの作成または更新」を参照してください。

  GuardDuty はまた、スキャン結果をデフォルトの EventBridge イベントバスにも発行します。詳細については、「Amazon EventBridge による S3 オブジェクトスキャンのモニタリング」を参照してください。

S3 バケットリソースへの TBAC の追加

S3 バケットリソースポリシーを使用すると、S3 オブジェクトに対するタグベースのアクセスコントロール (TBAC) を管理できます。S3 オブジェクトにアクセスして読み取るためのアクセス権を特定のユーザーに提供できます。を使用して作成された組織がある場合は AWS Organizations、GuardDuty によって追加されたタグを変更できないように強制する必要があります。詳細については、「AWS Organizations ユーザーガイド」の「Preventing tags from being modified except by authorized principals」を参照してください。このリンク先のトピックで使用されている例では、`ec2` について触れています。この例を使用するときは、*`ec2`* を `s3` に置き換えてください。

次に、TBAC を使用して何ができるかを示します。

- Malware Protection for S3 サービスプリンシパルを除くいずれのユーザーも、次のタグキーと値のペアでまだタグ付けされていない S3 オブジェクトを読み取れないようにします。

  GuardDutyMalwareScanStatus:*Potential key value*

- GuardDuty のみが、タグキーを GuardDutyMalwareScanStatus、値をスキャン結果としてスキャンされた S3 オブジェクトに追加できます。次のポリシーテンプレートを使用すると、アクセス権を持つ特定のユーザーにタグキーと値のペアの上書きを許可できます。

S3 バケットリソースポリシーの例:

サンプルポリシー内の次のプレースホルダ値を置き換えます。

- *IAM-role-name* は、バケットに Malware Protection for S3 を設定するために使用した IAM ロールを指定します。
- *555555555555* - 保護されたバケット AWS アカウント に関連付けられた を指定します。
- *amzn-s3-demo-bucket* - 保護されたバケット名を指定します。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
            "Sid": "NoReadUnlessClean",
            "Effect": "Deny",
            "NotPrincipal": {
                "AWS": [
                "arn:aws:sts::555555555555:assumed-role/IAM-role-name/
GuardDutyMalwareProtection",
                "arn:aws:iam::555555555555:role/IAM-role-name"
                ]
            },
            "Action": [
                "s3:GetObject",
                "s3:GetObjectVersion"
            ],
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
            "Condition": {
```

```
                "StringNotEquals": {
                    "s3:ExistingObjectTag/GuardDutyMalwareScanStatus":
 "NO_THREATS_FOUND"
                }
            }
        },
        {
            "Sid": "OnlyGuardDutyCanTagScanStatus",
            "Effect": "Deny",
            "NotPrincipal": {
                "AWS": [
                    "arn:aws:sts::555555555555:assumed-role/IAM-role-name/
GuardDutyMalwareProtection",
                    "arn:aws:iam::555555555555:role/IAM-role-name"
                ]
            },
            "Action": "s3:PutObjectTagging",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "s3:RequestObjectTagKeys": "GuardDutyMalwareScanStatus"
                }
            }
        }
    ]
}
```

S3 リソースのタグ付けの詳細については、「タグ付けとアクセスコントロールポリシー」を参照してください。

保護されたバケットのステータスの表示と理解

Malware Protection for S3 をバケットに対して有効にした後、ステータスは、機能が想定どおりに設定され、機能しているかどうかを示します。このステータスは、一意の Malware Protection プラン識別子 (ID) に関連付けられます。GuardDuty は、この機能を有効にしたときにこの ID を作成します。

保護されたバケットのステータスを表示するには、以下の手順に従います。

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

2. ナビゲーションペインで、[Malware Protection for S3] を選択します。
3. [保護されたバケット] テーブルで、S3 バケットの対応する [ステータス] 列を表示します。

以下の表は、Malware Protection プランリソースに関連付けられたステータス値の一覧と説明です。これらのステータスが保護されたバケットに与える影響の意味を理解することで、オブジェクトがアップロードされたときに GuardDuty が自動マルウェアスキャンを開始するよう、より確実に設定できます。

ステータス	説明
アクティブ	Malware Protection for S3 で S3 バケットが正常に設定されました。 ステータスが [アクティブ] の場合、IAM ロールへの変更 (削除またはアクセス許可の変更) はステータスを [警告] または [エラー] に更新しません。「S3 オブジェクトスキャンのモニタリング」で説明されている方法のいずれかを使用して、スキャンステータスを継続的にモニタリングすることをお勧めします。
警告*	Malware Protection for S3 は、警告が発生しても影響を受けないように設計されています。GuardDuty で新しい S3 オブジェクトが検出されると、マルウェアスキャンが開始されます。スキャンが正常に開始された後、[ステータス] 列の値が [アクティブ] に変わるまでに数分かかることがあります。[ステータス] 列の値が更新されると、EventBridge から通知が返されます。
エラー*	バケットが保護されていません。この S3 バケットに関連付けられたマルウェアスキャンが完了することはありません。根本原因は 1 つとは限りません。

*発生しうる問題とその解決手順については、「Malware Protection プランのステータスのトラブルシューティング」を参照してください。

Malware Protection for S3 での S3 オブジェクトスキャンのモニタリング

Malware Protection for S3 を GuardDuty ディテクター ID と共に使用している場合に、Amazon S3 オブジェクトに悪意の可能性があると、Malware Protection for S3 の検出結果タイプが生成されます。GuardDuty コンソールと API を使用すると、生成された検出結果を表示できます。この検出結果タイプについては、「検出結果の詳細」を参照してください。

GuardDuty を有効にせずに (ディテクター ID なしで) Malware Protection for S3 を使用した場合、スキャンした Amazon S3 オブジェクトに悪意の可能性があっても、検出結果が生成されません。

内容

- 取りうる S3 オブジェクトスキャンステータスと結果ステータス
- Amazon EventBridge による S3 オブジェクトスキャンのモニタリング
- GuardDuty マネージドタグによる S3 オブジェクトスキャンのモニタリング
- CloudWatch での S3 オブジェクトスキャンのステータスメトリクス

取りうる S3 オブジェクトスキャンステータスと結果ステータス

このセクションでは、取りうる S3 オブジェクトスキャンステータス値とスキャン結果値について説明します。

S3 オブジェクトスキャンのステータスはマルウェアスキャンのステータスを示し、完了、スキップ、失敗などがあります。

S3 オブジェクトマルウェアスキャンの結果ステータスは、スキャンのステータス値に基づいたスキャンの結果を示します。各マルウェアスキャンの結果ステータス値は、スキャンのステータスにマッピングされます。

次に、取りうる S3 オブジェクトスキャン結果値を示します。タグ付けを有効にした場合は、「S3 オブジェクトタグの使用」に従ってスキャン結果をモニタリングできます。スキャン後、タグ値は次のいずれかのスキャン結果値になります。

S3 オブジェクトマルウェアスキャンの取りうる結果ステータス値

- NO_THREATS_FOUND - GuardDuty は、スキャンされたオブジェクトに関連付けられた潜在する脅威を検出しませんでした。

- THREATS_FOUND – GuardDuty は、スキャンされたオブジェクトに関連付けられた潜在する脅威を検出しました。
- UNSUPPORTED – Malware Protection for S3 がスキャンをスキップするにはいくつかの理由があります。考えられる理由としては、パスワードで保護されたファイル、極端に高い圧縮率のアーカイブ、Malware Protection for S3 クォータ、特定の Amazon S3 機能のサポートが利用できない場合などがあります。詳細については、「Malware Protection for S3 の機能」を参照してください。
- ACCESS_DENIED – GuardDuty はこのオブジェクトにアクセスできずスキャンできません。このバケットに関連付けられている IAM ロールアクセス許可を確認してください。詳細については、「IAM ロールポリシーの作成または更新」を参照してください。

  スキャン後の S3 オブジェクトのタグ付けを有効にしている場合は、「S3 オブジェクトスキャン後のタグ失敗のトラブルシューティング」を参照してください。
- FAILED – GuardDuty は、内部エラーのためにこのオブジェクトに対するマルウェアスキャンを実行できません。

次に、S3 オブジェクトスキャンの取りうるステータス値およびその値と S3 オブジェクトスキャンの結果とのマッピングを示します。

S3 オブジェクトスキャンの取りうるステータス値

- 完了 – スキャンは正常に完了しました。S3 オブジェクトにマルウェアがあったかどうかが示されます。この場合、S3 オブジェクトスキャンの取りうる結果値は THREATS_FOUND または NO_THREATS_FOUND です。
- スキップ – この S3 オブジェクトスキャンが Malware Protection for S3 でサポートされていないか、選択されているバケットにアップロードされた S3 オブジェクトにアクセスできない場合、GuardDuty はマルウェアスキャンをスキップします。

  この場合、S3 オブジェクトスキャンの取りうる結果値は UNSUPPORTED または ACCESS_DENIED です。

  必要な IAM ロールが削除された場合、GuardDuty はスキャンをスキップします。
- 失敗 – S3 オブジェクトスキャンの結果値 FAILED と同じように、このスキャンステータスは内部エラーのために GuardDuty が S3 オブジェクトマルウェアスキャンを実行できなかったことを示します。

スキャンステータスが の場合SKIPPED、S3 オブジェクトスキャン結果の EventBridge 通知には 内のstatusReasonsフィールドが含まれますscanResultDetails。このフィールドは、スキャンが

スキップされた具体的な理由を示す文字列のリストです。次の表に、指定できるstatusReasons値を示します。

ステータス理由	スキャン結果のステータス	説明
UNAUTHORIZED_TO_GET_OBJECT	ACCESS_DENIED	Malware Protection for S3 には S3 オブジェクトを読み取るアクセス許可がないか、S3 オブジェクトが存在しません。保護されたバケットに関連付けられた IAM ロールに必要なアクセス許可があり、すべてのバケット AWS KMS ポリシーでロールがオブジェクトを復号化できることを確認します。
UNAUTHORIZED_TO_ASSUME_ROLE	ACCESS_DENIED	Malware Protection for S3 は、保護されたバケット用に設定された IAM ロールを引き受けることができません。ロール信頼ポリシーで Malware Protection for S3 がロールを引き受けることが許可されていることを確認します。
SSE_C_ENCRYPTED_OBJECT	ACCESS_DENIED	S3 オブジェクトは、お客様が用意した暗号化キー (SSE-C) で暗号化されます。GuardDuty は SSE-C で暗号化されたオブジェクトにアクセスできません。 詳細については、「」を参照してくださいAmazon S3 機能のサポート状況。
OBJECT_E_TAG_CHANGED	ACCESS_DENIED	S3 オブジェクトの ETag は、スキャンが開始されてから GuardDuty がオブジェクトの読み取りを試みたまでの間に変更されました。後続のアップロードまたは新しいバージョンがスキャンされます。
BUCKET_NOT_FOUND	ACCESS_DENIED	スキャンに関連付けられた S3 バケットが存在しなくなりました。
UNSUPPORTED_STORAGE_CLASS	UNSUPPORTED	S3 オブジェクトは、Malware Protection for S3 でサポートされていないストレージクラスを使

ステータス理由	スキャン結果のステータス	説明
		用します。詳細については、「Amazon S3 機能のサポート状況」を参照してください。
OBJECT_SIZE_LIMIT_EXCEEDED	UNSUPPORTED	S3 オブジェクトのサイズが Malware Protection for S3 の最大ファイルサイズ制限を超えています。詳細については、「Malware Protection for S3 クォータ」を参照してください。
PASSWORD_PROTECTED	UNSUPPORTED	オブジェクトはパスワードで保護されています。
EXTRACTED_FILE_LIMIT_EXCEEDED	UNSUPPORTED	アーカイブには、許可される最大制限を超えるファイルが含まれています。詳細については、「Malware Protection for S3 クォータ」を参照してください。
EXTRACTED_LEVEL_LIMIT_EXCEEDED	UNSUPPORTED	アーカイブがネストの最大許容深度を超えています。
EXTRACTED_BYTE_LIMIT_EXCEEDED	UNSUPPORTED	抽出されたアーカイブコンテンツが最大許容バイトサイズを超えています。詳細については、「Malware Protection for S3 クォータ」を参照してください。
EXTRACTION_RATIO_LIMIT_EXCEEDED	UNSUPPORTED	アーカイブの圧縮率が非常に高く、許容制限を超えています。詳細については、「Malware Protection for S3 クォータ」を参照してください。

Amazon EventBridge による S3 オブジェクトスキャンのモニタリング

Amazon EventBridge は、アプリケーションをさまざまなイベントソースのデータに簡単に接続できるようにするサーバーレスイベントバスサービスです。EventBridge は、独自のアプリケーション、Software-as-a-Service (SaaS) アプリケーション、および AWS のサービスからリアルタイムデータのストリームを配信し、そのデータを Lambda などのターゲットにルーティングします。こ

れにより、サービスで発生したイベントをモニタリングし、イベント駆動型アーキテクチャを構築できます。詳細については、「Amazon EventBridge ユーザーガイド」を参照してください。

Malware Protection for S3 で保護されている S3 バケットの所有者アカウントとして、GuardDuty は次のシナリオのときに EventBridge 通知をデフォルトのイベントバスに発行します。

- 保護されたバケットのいずれかで Malware Protection プランのリソースステータスが変更されます。さまざまなステータスについては、「保護されたバケットのステータスの表示と理解」を参照してください。

  リソースステータスに対して Amazon EventBridge (EventBridge) ルールを設定する方法については、「Malware Protection プランのリソースステータス」を参照してください。
- S3 オブジェクトスキャンの結果がデフォルトの EventBridge イベントバスに発行されます。

  `s3Throttled` フィールドは、Amazon S3 からストレージをアップロードまたは取得する際に遅延があったかどうかを示します。値 `true` は遅延があったことを示し、`false` は遅延がなかったことを示します。

  スキャン結果の `s3Throttled` が `true` の場合、Amazon S3 でプレフィックスを設定して各プレフィックスの 1 秒あたりのトランザクション数 (TPS) を削減することをお勧めします。詳細については、「Amazon S3 ユーザーガイド」の「設計パターンのベストプラクティス: Amazon S3 のパフォーマンスの最適化」を参照してください。

  S3 オブジェクトスキャンの結果に対して Amazon EventBridge (EventBridge) ルールを設定する方法については、「S3 オブジェクトスキャンの結果」を参照してください。
- スキャン後のタグ付け失敗イベントがある場合、次の理由が考えられます。
  - IAM ロールに、オブジェクトにタグ付けするアクセス許可がありません。

    IAM ポリシーのアクセス許可の追加テンプレートには、GuardDuty がオブジェクトにタグ付けするためのアクセス許可が含まれています。
  - IAM ロールに指定されているバケットリソースまたはオブジェクトが存在しなくなりました。
  - 関連付けられた S3 オブジェクトは、既にタグの最大制限に達しています。タグの制限の詳細については、「Amazon S3 ユーザーガイド」の「タグを使用したストレージの分類」を参照してください。

  スキャン後のタグ付け失敗イベントに対して Amazon EventBridge (EventBridge) ルールを設定する方法については、「スキャン後のタグ付け失敗イベント」を参照してください。

EventBridge ルールの設定

リソースステータス、スキャン後のタグ付け失敗イベント、または S3 オブジェクトスキャンの結果を別の AWS のサービスに送信するように、アカウントに EventBridge ルールを設定できます。ステータスに変化があると、委任 GuardDuty 管理者アカウントに Malware Protection プランのリソースステータス通知が返されます。

EventBridge の標準料金が適用されます。詳細については、「Amazon EventBridge の料金」を参照してください。

##で示された値はいずれも、この例のためのプレースホルダーです。これらの値は、アカウントの値と、マルウェアが検出されたかどうかに基づいて変化します。

トピック

- Malware Protection プランのリソースステータス
- S3 オブジェクトスキャンの結果
- スキャン後のタグ付け失敗イベント

Malware Protection プランのリソースステータス

次のシナリオに基づいて EventBridge イベントパターンを作成できます。

考えられる **detail-type** 値

- `"GuardDuty Malware Protection Resource Status Active"`
- `"GuardDuty Malware Protection Resource Status Warning"`
- `"GuardDuty Malware Protection Resource Status Error"`

イベントパターン

```
{
      "detail-type": ["potential detail-type"],
      "source": ["aws.guardduty"]
}
```

GuardDuty Malware Protection Resource Status Active の通知スキーマの例:

```
{
```

```
    "version": "0",
    "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
    "detail-type": "GuardDuty Malware Protection Resource Status Active",
    "source": "aws.guardduty",
    "account": "111122223333",
    "time": "2017-12-22T18:43:48Z",
    "region": "us-east-1",
    "resources": ["arn:aws:guardduty:us-east-1:111122223333:malware-protection-plan/
b4c7f464ab3a4EXAMPLE"],
    "detail": {
        "schemaVersion": "1.0",
        "eventTime": "2024-02-28T01:01:01Z",
        "s3BucketDetails": {
            "bucketName": "amzn-s3-demo-bucket"
        },
        "resourceStatus": "ACTIVE"
    }
}
```

GuardDuty Malware Protection Resource Status Warning の通知スキーマの例:

```
{
    "version": "0",
    "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
    "detail-type": "GuardDuty Malware Protection Resource Status warning",
    "source": "aws.guardduty",
    "account": "111122223333",
    "time": "2017-12-22T18:43:48Z",
    "region": "us-east-1",
    "resources": ["arn:aws:guardduty:us-east-1:111122223333:malware-protection-plan/
b4c7f464ab3a4EXAMPLE"],
    "detail": {
        "schemaVersion": "1.0",
        "eventTime": "2024-02-28T01:01:01Z",
        "s3BucketDetails": {
            "bucketName": "amzn-s3-demo-bucket"
        },
        "resourceStatus": "WARNING",
        "statusReasons": [
         {
            "code": "INSUFFICIENT_TEST_OBJECT_PERMISSIONS"
         }
        ]
```

```
    }
}
```

GuardDuty Malware Protection Resource Status Error の通知スキーマの例:

```
{
    "version": "0",
    "id": "fc7a35b7-83bd-3c1f-ecfa-1b8de9e7f7d2",
    "detail-type": "GuardDuty Malware Protection Resource Status Error",
    "source": "aws.guardduty",
    "account": "111122223333",
    "time": "2017-12-22T18:43:48Z",
    "region": "us-east-1",
    "resources": ["arn:aws:guardduty:us-east-1:111122223333:malware-protection-plan/
b4c7f464ab3a4EXAMPLE"],
    "detail": {
        "schemaVersion": "1.0",
        "eventTime": "2024-02-28T01:01:01Z",
        "s3BucketDetails": {
            "bucketName": "amzn-s3-demo-bucket"
        },
        "resourceStatus": "ERROR",
        "statusReasons": [
        {
            "code": "EVENTBRIDGE_MANAGED_EVENTS_DELIVERY_DISABLED"
        }
       ]
    }
}
```

resourceStatus ERROR の背後にある理由に基づいて、statusReasons の値が入力されます。

次の警告とエラーのトラブルシューティング手順については、「Malware Protection プランのステータスのトラブルシューティング」を参照してください。

S3 オブジェクトスキャンの結果

```
{
  "detail-type": ["GuardDuty Malware Protection Object Scan Result"],
  "source": ["aws.guardduty"]
}
```

scanStatus が の場合SKIPPED、 にはスキャンがスキップされた具体的な理由を示すstatusReasonsフィールドscanResultDetailsが含まれます。指定できる値については、「」を参照してください取りうる S3 オブジェクトスキャンステータスと結果ステータス。

NO_THREATS_FOUND の通知スキーマの例:

```
{
    "version": "0",
    "id": "72c7d362-737a-6dce-fc78-9e27a0171419",
    "detail-type": "GuardDuty Malware Protection Object Scan Result",
    "source": "aws.guardduty",
    "account": "111122223333",
    "time": "2024-02-28T01:01:01Z",
    "region": "us-east-1",
    "resources": ["arn:aws:guardduty:us-east-1:111122223333:malware-protection-plan/
b4c7f464ab3a4EXAMPLE"],
    "detail": {
        "schemaVersion": "1.0",
        "scanStatus": "COMPLETED",
        "resourceType": "S3_OBJECT",
        "s3ObjectDetails": {
            "bucketName": "amzn-s3-demo-bucket",
            "objectKey": "APKAEIBAERJR2EXAMPLE",
            "eTag": "ASIAI44QH8DHBEXAMPLE",
            "versionId" : "d41d8cd98f00b204e9800998eEXAMPLE",
            "s3Throttled": false
        },
        "scanResultDetails": {
            "scanResultStatus": "NO_THREATS_FOUND",
            "threats": null,
            "statusReasons": null
        }
    }
}
```

THREATS_FOUND の通知スキーマの例:

```
{
    "version": "0",
    "id": "72c7d362-737a-6dce-fc78-9e27a0171419",
    "detail-type": "GuardDuty Malware Protection Object Scan Result",
    "source": "aws.guardduty",
    "account": "111122223333",
```

```
    "time": "2024-02-28T01:01:01Z",
    "region": "us-east-1",
    "resources": ["arn:aws:guardduty:us-east-1:111122223333:malware-protection-plan/
b4c7f464ab3a4EXAMPLE"],
    "detail": {
        "schemaVersion": "1.0",
        "scanStatus": "COMPLETED",
        "resourceType": "S3_OBJECT",
        "s3ObjectDetails": {
            "bucketName": "amzn-s3-demo-bucket",
            "objectKey": "APKAEIBAERJR2EXAMPLE",
            "eTag": "ASIAI44QH8DHBEXAMPLE",
            "versionId" : "d41d8cd98f00b204e9800998eEXAMPLE",
            "s3Throttled": false
        },
        "scanResultDetails": {
            "scanResultStatus": "THREATS_FOUND",
            "threats": [
                {
                    "name": "EICAR-Test-File (not a virus)",
                    "source": "AMAZON",
                    "itemDetails": [{
                        "hash":
 "e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855",
                        "itemPath": "APKAEIBAERJR2EXAMPLE"
                    }]
                }
            ],
            "statusReasons": null
        }
    }
}
```

> **Note**
>
> scanResultDetails.Threats フィールドには、脅威が 1 つのみ含まれています。Malware Protection for S3 スキャンでは、デフォルトでは最初に検出された脅威が報告されます。この後、scanStatus は COMPLETED に設定されます。

スキャン結果ステータス **UNSUPPORTED** の通知スキーマの例 (スキップ):

```
{
    "version": "0",
    "id": "72c7d362-737a-6dce-fc78-9e27a0EXAMPLE",
    "detail-type": "GuardDuty Malware Protection Object Scan Result",
    "source": "aws.guardduty",
    "account": "111122223333",
    "time": "2024-02-28T01:01:01Z",
    "region": "us-east-1",
    "resources": ["arn:aws:guardduty:us-east-1:111122223333:malware-protection-plan/
b4c7f464ab3a4EXAMPLE"],
    "detail": {
        "schemaVersion": "1.0",
        "scanStatus": "SKIPPED",
        "resourceType": "S3_OBJECT",
        "s3ObjectDetails": {
            "bucketName": "amzn-s3-demo-bucket",
            "objectKey": "APKAEIBAERJR2EXAMPLE",
            "eTag": "ASIAI44QH8DHBEXAMPLE",
            "versionId" : "d41d8cd98f00b204e9800998eEXAMPLE",
            "s3Throttled": false
        },
        "scanResultDetails": {
            "scanResultStatus": "UNSUPPORTED",
            "threats": null,
            "statusReasons": ["PASSWORD_PROTECTED"]
        }
    }
}
```

スキャン結果ステータス **ACCESS_DENIED** の通知スキーマの例 (スキップ):

```
{
    "version": "0",
    "id": "72c7d362-737a-6dce-fc78-9e27a0EXAMPLE",
    "detail-type": "GuardDuty Malware Protection Object Scan Result",
    "source": "aws.guardduty",
    "account": "111122223333",
    "time": "2024-02-28T01:01:01Z",
    "region": "us-east-1",
    "resources": ["arn:aws:guardduty:us-east-1:111122223333:malware-protection-plan/
b4c7f464ab3a4EXAMPLE"],
    "detail": {
        "schemaVersion": "1.0",
```

```
        "scanStatus": "SKIPPED",
        "resourceType": "S3_OBJECT",
        "s3ObjectDetails": {
            "bucketName": "amzn-s3-demo-bucket",
            "objectKey": "APKAEIBAERJR2EXAMPLE",
            "eTag": "ASIAI44QH8DHBEXAMPLE",
            "versionId" : "d41d8cd98f00b204e9800998eEXAMPLE",
            "s3Throttled": false
        },
        "scanResultDetails": {
            "scanResultStatus": "ACCESS_DENIED",
            "threats": null,
            "statusReasons": ["SSE_C_ENCRYPTED_OBJECT"]
        }
    }
}
```

スキャン結果ステータス **FAILED** の通知スキーマの例:

```
{
    "version": "0",
    "id": "72c7d362-737a-6dce-fc78-9e27a0EXAMPLE",
    "detail-type": "GuardDuty Malware Protection Object Scan Result",
    "source": "aws.guardduty",
    "account": "111122223333",
    "time": "2024-02-28T01:01:01Z",
    "region": "us-east-1",
    "resources": ["arn:aws:guardduty:us-east-1:111122223333:malware-protection-plan/
b4c7f464ab3a4EXAMPLE"],
    "detail": {
        "schemaVersion": "1.0",
        "scanStatus": "FAILED",
        "resourceType": "S3_OBJECT",
        "s3ObjectDetails": {
            "bucketName": "amzn-s3-demo-bucket",
            "objectKey": "APKAEIBAERJR2EXAMPLE",
            "eTag": "ASIAI44QH8DHBEXAMPLE",
            "versionId" : "d41d8cd98f00b204e9800998eEXAMPLE",
            "s3Throttled": false
        },
        "scanResultDetails": {
            "scanResultStatus": "FAILED",
            "threats": null,
```

```
            "statusReasons": null
        }
    }
}
```

スキャン後のタグ付け失敗イベント

イベントパターン:

```
{
      "detail-type": "GuardDuty Malware Protection Post Scan Action Failed",
      "source": "aws.guardduty"
 }
```

ACCESS_DENIED の通知スキーマの例:

```
{
    "version": "0",
    "id": "746acd83-d75c-5b84-91d2-dad5f13ba0d7",
    "detail-type": "GuardDuty Malware Protection Post Scan Action Failed",
    "source": "aws.guardduty",
    "account": "111122223333",
    "time": "2024-06-10T16:16:08Z",
    "region": "us-east-1",
    "resources": ["arn:aws:guardduty:us-east-1:111122223333:malware-protection-plan/
b4c7f464ab3a4EXAMPLE"],
    "detail": {
        "schemaVersion": "1.0",
        "eventTime": "2024-06-10T16:16:08Z",
        "s3ObjectDetails": {
            "bucketName": "amzn-s3-demo-bucket",
            "objectKey": "2024-03-10-16-16-00-7D723DE8DBE9Y2E0",
            "eTag": "0e9eeec810ad8b61d69112c15c2a5hb6",
            "versionId" : "d41d8cd98f00b204e9800998eEXAMPLE",
            "s3Throttled": false
        },
        "postScanActions": [{
            "actionType": "TAGGING",
            "failureReason": "ACCESS_DENIED"
        }]
    }
}
```

MAX_TAG_LIMIT_EXCEEDED の通知スキーマの例:

```
{
    "version": "0",
    "id": "746acd83-d75c-5b84-91d2-dad5f13ba0d7",
    "detail-type": "GuardDuty Malware Protection Post Scan Action Failed",
    "source": "aws.guardduty",
    "account": "111122223333",
    "time": "2024-06-10T16:16:08Z",
    "region": "us-east-1",
    "resources": ["arn:aws:guardduty:us-east-1:111122223333:malware-protection-plan/
b4c7f464ab3a4EXAMPLE"],
    "detail": {
        "schemaVersion": "1.0",
        "eventTime": "2024-06-10T16:16:08Z",
        "s3ObjectDetails": {
            "bucketName": "amzn-s3-demo-bucket",
            "objectKey": "2024-03-10-16-16-00-7D723DE8DBE9Y2E0",
            "eTag": "0e9eeec810ad8b61d69112c15c2a5hb6",
            "versionId" : "d41d8cd98f00b204e9800998eEXAMPLE",
            "s3Throttled": false
        },
        "postScanActions": [{
            "actionType": "TAGGING",
            "failureReason": "MAX_TAG_LIMIT_EXCEEDED"
        }]
    }
}
```

こうした失敗理由をトラブルシューティングするには、「S3 オブジェクトスキャン後のタグ失敗のトラブルシューティング」を参照してください。

GuardDuty マネージドタグによる S3 オブジェクトスキャンのモニタリング

タグ付けを有効にするオプションを使用すると、GuardDuty でマルウェアスキャンの完了後に Amazon S3 オブジェクトにタグを追加できるようになります。

タグ付けを有効にする際の考慮事項

- GuardDuty で S3 オブジェクトにタグ付けする際、それに伴う使用コストが発生します。詳細については、「Malware Protection for S3 の料金と使用コスト」を参照してください。

- このバケットに関連付けられた優先 IAM ロールで必要なタグ付けのアクセス許可を保持する必要があります。保持しない場合、GuardDuty はスキャンしたオブジェクトにタグを追加できません。この IAM ロールには、スキャンされた S3 オブジェクトにタグを追加するためのアクセス許可が既に含まれています。詳細については、「IAM ロールポリシーの作成または更新」を参照してください。
- デフォルトでは、1 つの S3 オブジェクトに最大 10 個のタグを関連付けることができます。詳細については、「タグベースのアクセスコントロール (TBAC) の使用」を参照してください。

S3 バケットまたは特定のプレフィックスに対してタグ付けを有効にした後で、新規にアップロードされたオブジェクトをスキャンすると、そのオブジェクトに次のキーと値のペアの形式でタグが関連付けられます。

`GuardDutyMalwareScanStatus:`*`Scan-Result-Status`*

タグが取りうる値については、「取りうる S3 オブジェクトスキャンステータスと結果ステータス」を参照してください。

Malware Protection for S3 における S3 オブジェクトスキャン後のタグ失敗のトラブルシューティング

このセクションの内容は、保護されたバケットで「スキャンされたオブジェクトのタグ付けを有効にする」を実施した場合にのみ当てはまります。

GuardDuty がスキャンした S3 オブジェクトにタグを追加しようとしたときに、タグ付けのアクションが失敗することがあります。バケットでこれが発生する理由として、`ACCESS_DENIED` と `MAX_TAG_LIMIT_EXCEEDED` が考えられます。以降のトピックでは、こうしたスキャン後のタグ付け失敗の考えられる理由とそのトラブルシューティングについて説明します。

ACCESS_DENIED

次に、この問題の原因として考えられる理由を示します。

- この保護された S3 バケットで使用される IAM ロールに AllowPostScanTag アクセス許可がありません。関連付けられた IAM ロールでこのバケットポリシーが使用されていることを確認します。詳細については、「IAM ロールポリシーの作成または更新」を参照してください。
- 保護された S3 バケットポリシーが使用されている場合、GuardDuty でこのオブジェクトにタグを追加することはできません。
- スキャンされた S3 オブジェクトが存在しなくなりました。

MAX_TAG_LIMIT_EXCEEDED

デフォルトでは、1 つの S3 オブジェクトに最大 10 個のタグを関連付けることができます。詳細については、「スキャンされたオブジェクトのタグ付けを有効にする」の「GuardDuty による S3 オブジェクトへのタグの追加に関する考慮事項」を参照してください。

CloudWatch での S3 オブジェクトスキャンのステータスメトリクス

raw データを収集して読み取り可能なほぼリアルタイムのメトリクスに処理する CloudWatch を使用して GuardDuty をモニタリングできます。これらの統計は 15 か月間保持されるため、履歴情報にアクセスして Malware Protection for S3 の動作状況をより的確に把握できます。また、特定のしきい値を監視するアラームを設定し、これらのしきい値に達したときに通知を送信したりアクションを実行したりできます。詳細については、「Amazon CloudWatch ユーザーガイド」を参照してください。

Malware Protection for S3 の CloudWatch メトリクスをリソースレベルで使用できます。こうしたメトリクスを保護されたリソースごとに個別にクエリできます。メトリクスは、`AWS/GuardDuty/MalwareProtection` 名前空間に報告されます。特定のリソースにアラームを設定して、セキュリティ体制をモニタリングできます。

マルウェアスキャンのステータスメトリクス

メトリクス	説明
`CompletedScanCount`	特定の時間枠で完了した S3 オブジェクトマルウェアスキャンの数。 有効なディメンション: • `Malware Protection Plan Id` `Resource Name` 単位: カウント
`FailedScanCount`	特定の時間枠で失敗した S3 オブジェクトマルウェアスキャンの数。 有効なディメンション:

	• Malware Protection Plan Id Resource Name 単位: カウント
SkippedScanCount	特定の時間枠でスキップされた S3 オブジェクトマルウェアスキャンの数。 有効なディメンション: • Malware Protection Plan Id Resource Name Skipped Reason 考えられる 値 • Unsupported • MissingPermissions 単位: カウント
マルウェアスキャンの結果メトリクス	
InfectedScanCount	特定の時間枠で悪意の可能性があるオブジェクトを検出した S3 オブジェクトマルウェアスキャンの数。 有効なディメンション: • Malware Protection Plan Id Resource Name 単位: カウント

`CompletedScanBytes`	特定の時間枠でスキャンされた S3 オブジェクトバイトの数。 有効なディメンション: • `Malware Protection Plan Id` `Resource Name` 単位: カウント

> **Note**
>
> CloudWatch メトリクスの統計は、デフォルトでは AVG です。

Malware Protection for S3 メトリクスでは、次のディメンションがサポートされています。

ディメンション	説明
`Malware Protection ### ID`	保護されたリソースに対して Malware Protection プランのリソースが作成された場合に、そのリソースに関連付けられる一意の識別子。
`#####`	保護されたリソースの名前。
`#########`	S3 オブジェクトマルウェアスキャンがスキップされた理由。 考えられる 値 • `Unsupported` • `MissingPermissions`

こうしたメトリクスにアクセスしてクエリする方法については、「Amazon CloudWatch ユーザーガイド」の「Amazon CloudWatch Metrics の使用」を参照してください。

アラームの設定の詳細については、「Amazon CloudWatch ユーザーガイド」の「Amazon CloudWatch アラームの使用」を参照してください。

トラブルシューティング

内容

- Malware Protection プランのステータスのトラブルシューティング
- オンデマンドマルウェアスキャンのトラブルシューティング

Malware Protection プランのステータスのトラブルシューティング

GuardDuty は、保護されたバケットを対象にランキングに基づいて [ステータス] を表示します。例えば、保護されたバケットで [エラー] カテゴリと [警告] カテゴリの両方で問題が発生している場合、GuardDuty は最初に [エラー] ステータスに関連付けられている問題を表示します。

次に、Malware Protection プランのステータスのエラーと警告を示します。

エラー

- この S3 バケットに対して EventBridge 通知は無効
- S3 バケットイベントを受け取るための EventBridge マネージドルールがない
- S3 バケットがもう存在しない

警告

テストオブジェクトを配置できない

この S3 バケットに対して EventBridge 通知は無効

関連付けられたステータス理由コードは `EVENTBRIDGE_MANAGED_EVENTS_DELIVERY_DISABLED` です。

ステータスの詳細

GuardDuty は、EventBridge を使用して、この S3 バケットに新しいオブジェクトがアップロードされたら通知を受け取るようにしています。このアクセス許可は IAM ロールにありません。

トラブルシューティングするための手順

オプション 1: IAM ロールに次のアクセス許可ステートメントを追加する:

```
{
          "Sid": "AllowEnableS3EventBridgeEvents",
          "Effect": "Allow",
          "Action": [
             "s3:PutBucketNotification",
             "s3:GetBucketNotification"
             ],
          "Resource": [
             "arn:aws:s3:::amzn-s3-demo-bucket"
            ]
}
```

amzn-s3-demo-bucket を実際の Amazon S3 バケット名に置き換えてください。

オプション 2: Amazon S3 コンソールを使用して EventBridge 通知を有効にする

1. Amazon S3 コンソール (https://console.aws.amazon.com/s3/) を開きます。
2. [バケット] ページの [汎用バケット] タブで、このエラーに関連付けられているバケット名を選択します。
3. この [バケット] ページで、[プロパティ] タブを選択します。
4. [Amazon EventBridge] セクションで、[編集] を選択します。
5. [Amazon EventBridge の編集] ページで、[このバケット内のすべてのイベントについて Amazon EventBridge に通知を送信する] に対して [オン] を選択します。
6. [Save changes] (変更の保存) をクリックします。

[ステータス] 列の値が [アクティブ] に変わるまでに数分かかることがあります。

S3 バケットイベントを受け取るための EventBridge マネージドルールがない

関連付けられたステータス理由コードは `EVENTBRIDGE_MANAGED_RULE_DISABLED` です。

ステータスの詳細

EventBridge ルール設定を管理するための EventBridge マネージドルールのアクセス許可がありません。

トラブルシューティングするための手順

IAM ロールに次のアクセス許可ステートメントを追加します。

```
{
         "Sid": "AllowManagedRuleToSendS3EventsToGuardDuty",
        "Effect": "Allow",
        "Action": [
                "events:PutRule",
                "events:DeleteRule",
                "events:PutTargets",
                "events:RemoveTargets"
            ],
        "Resource": [
           "arn:aws:events:*:*:rule/DO-NOT-DELETE-
AmazonGuardDutyMalwareProtectionS3*"
           ],
        "Condition": {
           "StringEquals": {
              "events:ManagedBy": "malware-protection-plan.guardduty.amazonaws.com"
              }
           }
}
```

[ステータス] 列の値が [アクティブ] に変わるまでに数分かかることがあります。

S3 バケットがもう存在しない

関連付けられたステータス理由コードは PROTECTED_RESOURCE_DELETED です。

ステータスの詳細

この S3 バケットは、アカウントから削除されてもう存在していません。

トラブルシューティングするための手順

S3 バケットを意図せずに削除してしまった場合は、Amazon S3 コンソールを使用して新しいバケットを作成できます。

バケットを正常に作成したら、「バケットの Malware Protection for S3 の設定」ページの手順を実行して Malware Protection for S3 を有効にします。

テストオブジェクトを配置できない

関連付けられたステータス理由コードは INSUFFICIENT_TEST_OBJECT_PERMISSIONS です。

> **Note**
>
> テストオブジェクトを追加するためのアクセス許可はオプションです。このアクセス許可が IAM ロールにない場合、新規にアップロードされたオブジェクトに対して Malware Protection for S3 がマルウェアスキャンを開始するのを防ぐことができません。スキャンが正常に始まると、Malware Protection プランの [ステータス] が [警告] から [アクティブ] に変わるまでに数分かかることがあります。
> このアクセス許可が IAM ロールに既に含まれている場合、この警告は制限的な Amazon S3 バケットポリシーにより、この S3 バケットにテストオブジェクトを配置するための IAM アクセスが許可されないことを示します。

ステータスの詳細

GuardDuty では、選択されているバケットの設定を確認するために、バケットにテストオブジェクトが配置されます。

トラブルシューティングするための手順

欠落しているアクセス許可を含めるように IAM ロールを更新することもできます。選択されている IAM ロールに次のアクセス許可を追加すると、GuardDuty が選択されているリソースにテストオブジェクトを配置できるようになります。

```
{
         "Sid": "AllowPutValidationObject",
         "Effect": "Allow",
         "Action": [
           "s3:PutObject"
           ],
         "Resource": [
           "arn:aws:s3:::amzn-s3-demo-bucket/malware-protection-resource-validation-
object"
           ]
}
```

amzn-s3-demo-bucket を実際の Amazon S3 バケット名に置き換えてください。IAM ロールアクセス許可については、「IAM ロールポリシーの作成または更新」を参照してください。

[ステータス] 列の値が [アクティブ] に変わるまでに数分かかることがあります。

オンデマンドマルウェアスキャンのトラブルシューティング

スキャンを開始できません。

スキャンリクエストに有効な入力が含まれており、バケットに対して Malware Protection Plan が有効になっていることを確認してください。

保護されたバケットの Malware Protection プランの編集

優先 IAM アクセス許可ポリシーの編集、スキャンされた S3 オブジェクトのタグ付けの有効化または削除、あるいは S3 オブジェクトプレフィックスの追加または削除が必要になる場合があります。例えば、バケットの Malware Protection for S3 を有効にした場合、スキャンされた S3 オブジェクトへのスキャン結果のタグ付けを有効にしないことになります。しかし、ここで、GuardDuty で事前定義されたタグとスキャン結果をタグ値として追加する必要が生じたとします。

任意のアクセス方法を選択して、保護された S3 バケットの Malware Protection プランを更新します。

Console

Malware Protection プランを編集するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[Malware Protection for S3] を選択します。
3. [保護されたバケット] で、既存の設定を編集するバケットを選択します。
4. [編集] を選択します。
5. バケットの既存の構成と設定を更新し、変更を確認します。各セクションの説明と手順については、「バケットの Malware Protection for S3 の有効化」を参照してください。

   この保護されたバケットの [ステータス] 列をモニタリングします。[警告] または [エラー] として表示される場合は、「Malware Protection プランのステータスのトラブルシューティング」を参照してください。

API/CLI

API または を使用して Malware Protection プランを編集するには AWS CLI

- API を使用する場合

このプランのリソースに関連付けられた Malware Protection プラン ID を使用して、UpdateMalwareProtectionPlan API を実行します。

特定のリージョンの Malware Protection プラン ID を取得するには、そのリージョンで ListMalwareProtectionPlans API を実行できます。

- を使用する AWS CLI

  次のリストは、Malware Protection プランリソースを更新するためのコマンド AWS CLI の例を示しています。S3 バケットに関連付けられた Malware Protection プラン ID が必要になります。

  AWS CLI コマンドの例

  - 次の AWS CLI コマンドを使用して、S3 バケットに関連付けられた Malware Protection プランリソースのタグ付けを有効または無効にします。

    ```
    aws guardduty update-malware-protection-plan --malware-protection-plan-
    id 4cc8bf26c4d75EXAMPLE --actions "Tagging"={"Status"="ENABLED|DISABLED"}
    ```

  - 次の AWS CLI コマンドを使用して、S3 バケットに関連付けられた Malware Protection プランリソースにオブジェクトプレフィックスを追加します。

    ```
    aws guardduty update-malware-protection-plan --malware-
    protection-plan-id 4cc8bf26c4d75EXAMPLE --protected-resource
     "S3Bucket"={"ObjectPrefixes"=["amzn-s3-demo-1", "amzn-s3-demo-2"]}
    ```

    このコマンドには既存のオブジェクトプレフィックスを必ず含めてください。含めない場合、GuardDuty は Malware Protection プランリソースを編集するときにそれらのプレフィックスを削除します。

  - 次の AWS CLI コマンドを使用して、S3 バケットに関連付けられた Malware Protection プランリソースからオブジェクトプレフィックスを削除します。

    ```
    aws guardduty update-malware-protection-plan --malware-protection-plan-
    id 4cc8bf26c4d75EXAMPLE --protected-resource "S3Bucket"={"ObjectPrefixes"=[""]}
    ```

  このリソースの Malware Protection プラン ID がまだない場合は、次の AWS CLI コマンドを実行し、*us-east-1* を Malware Protection プラン IDs。

```
aws guardduty list-malware-protection-plans --region us-east-1
```

保護されたバケットの Malware Protection for S3 の無効化

保護されたバケットの Malware Protection for S3 を無効にすると、GuardDuty はそのバケットに関連付けられた Malware Protection プラン ID を削除します。新しいオブジェクトがこのバケットまたは選択したオブジェクトプレフィックスのいずれかにアップロードされた場合、GuardDuty はマルウェアスキャンを開始しません。

GuardDuty を有効にしていて、これから GuardDuty を一時停止または無効にしたい場合は、「GuardDuty の停止または無効化」を参照してください。Malware Protection for S3 にはディテクター ID の概念がないため、GuardDuty を無効または一時停止しても、アカウントの保護されたバケットのステータスには影響しません。Malware Protection for S3 機能は、関連する標準料金で引き続き単独で使用できます。詳細については、「Malware Protection for S3 の使用コストの確認」を参照してください。Malware Protection for S3 の使用を停止するには、アカウント内のすべての保護されたバケットに対して無効にする必要があります。GuardDuty を引き続き使用してバケットの Malware Protection for S3 のみを無効にする場合、次のステップは GuardDuty サービスの設定や、有効にしたその他の保護プランには影響しません。

任意のアクセス方法を選択して、保護対象の S3 バケットで Malware Protection for S3 を無効にします。

Console

GuardDuty コンソールを使用して Malware Protection for S3 を無効にするには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[Malware Protection for S3] を選択します。
3. [保護されたバケット] で、Malware Protection for S3 を無効にするバケットを選択します。

   一度に選択できる保護されたバケットは 1 つだけです。複数のバケットで Malware Protection for S3 を無効にするには、別の S3 バケットに対して再度これらのステップを実行します。
4. [無効にする] を選択して、選択を確定します。

API/CLI

API または を使用して Malware Protection for S3 を無効にするには AWS CLI

- API を使用する場合

  このプランリソースに関連付けられている Malware Protection プラン ID を使用して DeleteMalwareProtectionPlan API を実行します。

  Malware Protection プラン ID を取得するには、ListMalwareProtectionPlans API を実行できます。

- を使用する AWS CLI

  または、次の AWS CLI コマンドを実行して、*4cc8bf26c4d75EXAMPLE* をこの S3 バケットに関連付けられた Malware Protection プラン ID に置き換えることで、Malware Protection for S3 を無効にすることもできます。

```
aws guardduty delete-malware-protection-plan --malware-protection-plan-
id 4cc8bf26c4d75EXAMPLE
```

  この S3 バケットの Malware Protection プラン ID がまだない場合は、次の AWS CLI コマンドを実行し、*us-east-1* を Malware Protection プラン IDs を一覧表示するリージョンに置き換えることができます。

```
aws guardduty list-malware-protection-plans --region us-east-1
```

Amazon S3 機能のサポート状況

次の表に、Malware Protection for S3 がリストされている Amazon S3 機能をサポートしているかどうかを示します。

サポートが受けられるか	説明
はい	S3 オブジェクトは、非同期に復元しなくても取得できます。

サポートが受けられるか	説明

サポートが受けられるか	説明
条件付き	• S3 オブジェクトの場合、頻度、低頻度、アーカイブインスタンスアクセスの各階層で Intelligent Tiering のサポートを受けられます。 • オプトインアーカイブ階層とディープアーカイブ階層はサポートされていません。 • Intelligent Tiering は、常に高頻度アクセス階層に新しいオブジェクトを作成します。このため、作成時オブジェクトスキャンがサポートされています。 • Intelligent Tiering 機能は今後、アーカイブ内のオブジェクトから始まる可能性があります。そのため、これはサポートされていません。

サポートが受けられるか	説明
いいえ	GuardDuty は、Malware Protection for S3 の汎用バケットのみをサポートします。

サポートが受けられるか	説明
いいえ	S3 オブジェクトにアクセスするには、事前に復元しておく必要があります。

サポートが受けられるか	説明
いいえ	Malware Protection for S3 は、Outposts ではサポートされていません。
はい	アップロードされたすべての S3 オブジェクトが、マルウェアがないかスキャンされます。ファイルバージョン v1 でオブジェクトをアップロードした後、すぐに v2 で上書きした別のバージョンをアップロードした場合は、ファイルバージョン v1 と v2 の両方のオブジェクトがスキャンされます。ただし、スキャン開始時刻が同じ順序になるとは限りません。

サポートが受けられるか	説明
はい	レプリケート先バケットが保護されたリソースである場合、GuardDuty は保護およびモニタリングされているプレフィックスにレプリケートされているすべての S3 オブジェクトをスキャンします。

サポートが受けられるか	説明
いいえ	レプリケーションルールをスキャン結果タグに基づいて定義することはできません。Amazon S3 は、作成時を除き、タグのレプリケーションをサポートしていません。

サポートが受けられるか	説明
はい	GuardDuty は、マネージドキーおよびカスタマーマネージドキーで暗号化された S3 オブジェクトのマルウェアスキャンをサポートしています。こうしたキーを使用するためのアクセス許可を IAM ロールに必ず含めてください。詳細については、「IAM ポリシーのアクセス許可の追加」を参照してください。

サポートが受けられるか	説明
いいえ	Malware Protection for S3 は、アクセスできないキーで暗号化された S3 オブジェクトのスキャンをサポートしていません。

サポートが受けられるか	説明
いいえ	Amazon S3 Encryption Client を使用して Amazon S3 オブジェクトを暗号化すると、そのオブジェクトが AWS を含め第三者に公開されることはありません。これがサポートされていない理由の詳細については、「クライアント側の暗号化を使用したデータの保護」を参照してください。 **Note** CSE-KMS 暗号化オブジェクトは暗号化を判別できない暗号化された BLOB として受信されます。したがって、GuardDuty はそれらを受信するとすぐに処理し、暗号化された BLOB を通常のファイルとしてスキャンします。GuardDuty は、Malware Protection for S3 のクォータ のいずれかが超過しない限り、そのようなオブジェクトの `UNSUPPORTED` スキャンステータスを返しません。

サポートが受けられるか	説明
はい	ロックされた S3 オブジェクトは、Write Once Read Many (WORM) に基づいてロックされます。Malware Protection for S3 は、こうしたオブジェクトにアクセスしてスキャンできます。
はい	Malware Protection for S3 では、リクエスタ支払いで設定されたバケットをスキャンできます。リクエスタは、S3 の呼び出しに対して料金を支払います。詳細については、「Amazon S3 ユーザーガイド」の「ストレージ転送と使用量のリクエスタ支払いバケットの使用」を参照してください。

サポートが受けられるか	説明
はい	ライフサイクルポリシーをスキャン結果タグに基づいて定義できます。例えば、悪意のあるオブジェクトを自動的に削除します。ライフサイクル設定の詳細については、「Amazon S3 ユーザーガイド」の「ストレージライフサイクルの管理」を参照してください。
はい	バケットリソースポリシーを S3 オブジェクトスキャン結果タグに基づいて定義できます。例えば、まだスキャンされていない S3 オブジェクトや GuardDuty で検出された脅威へのアクセスを防ぎます。詳細については、「Malware Protection for S3 によるタグベースのアクセスコントロール (TBAC) の使用」を参照してください。

サポートが受けられるか	説明
いいえ	Malware Protection for S3 は、S3 オブジェクトコンテンツのみをスキャンします。

Malware Protection for S3 のクォータ

このセクションでは、制限とも呼ばれるデフォルトのクォータについて説明します。特に指定がない限り、各クォータはリージョン固有です。基盤となる GuardDuty サービスを使用した場合に固有のデフォルトのクォータを表示するには、「Amazon GuardDuty クォータ」を参照してください。

次の表では、 AWS アカウントに適用されるいくつかのクォータについて説明します。

AWS デフォルトのクォータ値	調整可能かどうか	説明
100 GB	いいえ	GuardDuty でマルウェアがスキャンされる S3 オブジェクトの最大サイズ。

AWS デフォルトのクォータ値	調整可能かどうか	説明
100,000	いいえ	GuardDuty が S3 オブジェクトで抽出および分析できるファイルの最大数。 **Note** 複合ファイルタイプは、これらの制限の対象となる可能性があります。ファイルタイプには、多目的インターネットメール拡張 (MIME) でエンコードされた E メールメッセージ、コンパイル済み Python (PYC) ファイル、コンパイル済み HTML ヘルプ (CHM) ファイル、すべてのインストーラ、および OpenDocument Format (ODF) ドキュメントが含まれますが、これらに限定されません。 **Note** S3 オブジェクトサイズと抽出されたコンテンツサイズの間の圧縮率が非常に高い S3 オブジェクトは、抽出されたファイルがこのクォータを下回っていても制限される可能性があります。

AWS デフォルトのクォータ値	調整可能かどうか	説明
100	いいえ	GuardDuty が S3 オブジェクトで抽出できるネストの最大レベル。
25	いいえ	Malware Protection for S3 を有効にできる S3 バケットの最大数。このクォータ制限は、各リージョンのアカウントごとに適用されます。

GuardDuty Malware Protection for AWS Backup

トピック

- 概要
- Malware Protection for Backup の仕組み
- GuardDuty Malware Protection for Backup: IAM ロールのアクセス許可
- (オプション) Malware Protection for Backup を個別に開始する (コンソールのみ)
- Malware Protection for Backup のオンデマンドスキャンの開始
- Malware Protection for Backup でのスキャンステータスと結果のモニタリング
- Malware Protection for Backup のクォータ

概要

Malware Protection for Backup は、Amazon EBS スナップショット、Amazon EC2 AMIs、Amazon S3 リカバリポイントなどの AWS Backup で保護されたリソースをスキャンして、バックアップデータにマルウェアが存在する可能性を検出するのに役立ちます。 AWS Backup が保護されたバックアップリソースを作成または更新すると、GuardDuty はそのバックアップに対してマルウェアスキャンを実行して、環境に復元される前に悪意のある可能性のあるコンテンツを特定できます。

Malware Protection for Backup の使用方法

この機能は、アカウントで GuardDuty が有効になっているかどうかに応じて、次の 2 つのモードで使用できます。

1. GuardDuty を有効にして Malware Protection for Backup を使用する

   リージョンで GuardDuty が有効になっている場合、 AWS Backup は Malware Protection を GuardDuty の検出結果ワークフローと統合します。マルウェアスキャンの結果は、Amazon EventBridge と Amazon CloudWatch に加えて GuardDuty の検出結果に表示されます。

2. GuardDuty を有効にせずに Malware Protection for Backup を使用する

   Malware Protection for Backup は、完全な GuardDuty サービスを有効にすることなく、個別に使用できます。このモードでは、EventBridge と CloudWatch を通じてスキャン結果を完全に利用できます。

Malware Protection for Backup を個別に使用する際の考慮事項

GuardDuty を有効にせずに 機能を使用する場合:

- バックアッププランの設定は、 AWS Backup で完全に管理されます。

GuardDuty は、バックアッププラン、ボールト、またはリソースタイプを選択するためのコントロールを提供しません。すべての有効化、スケジューリング、ポリシー設定は AWS Backup に残ります。

- GuardDuty の検出結果は生成されません。

検出結果には、GuardDuty が有効になっている場合にのみ作成されるディテクター ID が必要です。Malware Protection を個別に使用する場合、スキャン結果は EventBridge イベントと CloudWatch メトリクスを通じてのみ表示されます。

- GuardDuty コンソールからオンデマンドスキャンを開始できます。

GuardDuty が有効になっていない場合でも、GuardDuty コンソールには、サポートされているバックアップリソースタイプのオンデマンドマルウェアスキャンを開始するワークフローが用意されています。これにより、お客様は完全な GuardDuty サービスを使用せずに、使い慣れた GuardDuty インターフェイスを使用できます。

- GuardDuty 以外のお客様は、スキャン開始ワークフローにアクセスできます。

オンデマンドスキャンエントリポイントは、GuardDuty ディテクターがアカウントに存在するかどうかにかかわらず、Malware Protection for Backup を使用しているすべてのお客様が使用できます。

- スキャンの動作とカバレッジは同じままです。

GuardDuty が有効になっているかどうかにかかわらず、この機能は同じマルウェア検出エンジンを使用して同じ AWS Backup リソースタイプをスキャンします。唯一の違いは、結果が公開される場所です。

このモデルにより、GuardDuty のより広範な脅威検出機能を必要とすることなく、バックアップにマルウェアスキャンを採用できます。また、スキャンオペレーションを開始および表示するためのオプションの GuardDuty ベースのワークフローも提供します。

Malware Protection for Backup の仕組み

Malware Protection for Backup では、以下の AWS Backup で保護されたリソースをスキャンできます。

- Amazon EBS snapshots
- Amazon EC2 AMI
- Amazon S3 リカバリポイントと S3 継続的バックアップまたはpoint-in-time-recovery (PITR)
- サポートされているリージョンで AWS Backup Vault Lock を使用してロックされた (イミュータブル) ボールト (EBS/EC2 リカバリポイント)

増分スキャン

AWS Backup は、多くのリソースタイプの増分変更をキャプチャします。GuardDuty には、バックアップの作成時または更新時に新規または変更されたブロックまたはオブジェクトのみをスキャンする機能があり、パフォーマンスを向上させ、スキャンオーバーヘッドを削減しながら、時間の経過とともにフルカバレッジを実現します。

オンデマンドスキャン

サポートされているバックアップリソースでは、 AWS Backup または GuardDuty コンソールから直接、いつでもスキャンを開始できます。一般的なユースケースには、復元前のバックアップの検証、新しい脅威署名の公開後の古いデータの再チェック、定期的なコンプライアンススキャンの実行などがあります。

> **Note**
>
> - Malware Protection for Backup は、同じリージョンのバックアップリソースに対してのみ有効にできます。
> - GuardDuty はバックアップの読み取り専用コピーをスキャンします。バックアップコンテンツは変更されません。
> - スキャンは、標準ボールトとロック (イミュータブル) ボールトの両方で機能します。

Malware Protection for Backup の仕組み

このセクションでは、Malware Protection for Backup のコンポーネント、その仕組み、マルウェアスキャンのステータスと結果を確認する方法について説明します。

概要

Malware Protection for Backup は、EBS スナップショット、EC2 イメージ (AMI)、EBS、EC2、EC2S3 リソースタイプに属する復旧ポイントでのマルウェアの存在を検出するのに役立つ機能です。スキャンに必要なアクセス許可と、スキャンカテゴリに応じて 1 つまたは 2 つのリソース ARNs を提供する IAM ロールを渡すことで、GuardDuty コンソールまたは API を介してオンデマンドマルウェアスキャンを開始できます。フルスキャンと増分スキャンの 2 つのスキャンカテゴリがあります。

フルスキャンと増分スキャン

フルスキャンは、API がリソース ARN を受け入れ、そのリソース内のすべてのファイルをスキャンする場所です。一方、増分スキャンでは、同じリソースに属する 2 つのリソース ARNs を取得し、それらの間で変更されたファイルをスキャンします。たとえば、EBS ボリュームのスナップショットを作成するとします。snapshot-1 と呼びます。このスナップショットでフルスキャンが実行された場合、GuardDuty はこのスナップショットに含まれるすべてのファイルをスキャンします。ここで、同じボリュームにいくつかのファイルが追加され、新しいスナップショットが作成されるとします。snapshot-2 と呼びます。snapshot-1 と snapshot-2 の間で変更されたファイルはわずかであるため、これらの 2 つのスナップショットのリソース ARNs を使用して増分スキャンをトリガーできます。この場合、snapshot-2 は`target`リソースと呼ばれ、snapshot-1 は`base`リソースと呼ばれます。この用語は、ドキュメントの残りの部分で使用されます。この増分スキャンは、snapshot-1 と snapshot-2 の間で変更されたファイルをスキャンします。

Point-in-Time-Recovery (PITR) のマルウェアスキャン

AWS Backup から有効にすると、継続的バックアップまたは PITR のマルウェアスキャンは、Backup プランで設定された頻度に従って行われます。これは AWS Backup サービスによってオーケストレーションされます。たとえば、GuardDuty は t0 から t2 への最初のスキャン、t2 から t4 への次のスキャン、t4 から t6 への後続のスキャンなどを実行します。これらの各スキャンは、`GetMalwareScan`API を使用してクエリできるスキャン ID を生成し、その期間のスキャンのステータスと結果を取得します。

または、GuardDuty コンソールから、または終了タイムスタンプを渡して GuardDuty `StartMalwareScan` API を使用して、PITR リソース全体をオンデマンドでスキャンすることもで

きます。これにより、GuardDuty は最初から終了タイムスタンプまでリソース全体をスキャンします。

> **Note**
>
> - GuardDuty から PITR の部分スキャンまたは増分スキャンをトリガーすることはできません。
> - オンデマンドスキャンで渡された終了タイムスタンプは、(現在の時刻 - 15 分) より前にすることはできません。

以前に感染したファイルを増分スキャンで再スキャンする

増分スキャンの一環として、GuardDuty は以前に感染したファイルをベーススキャンから最大 365 日間再スキャンします。

増分スキャンの要件

GuardDuty が増分スキャンを実行するには、次の要件を満たす必要があります。これらの要件のいずれかが満たされない場合、GuardDuty はスキャンをスキップします。

- ベースリソースは過去 365 日以内にスキャンする必要があり、スキャン結果は `COMPLETED`または`COMPLETED_WITH_ISSUES`である必要があります。
- ベースリソースには、ターゲットリソースの作成日より前の作成日が必要です。
- スナップショットの場合、ベースリソースとターゲットリソースは同じ暗号化タイプである必要があります。
- ベースリソースとターゲットリソースは同じ系統のものである必要があります。
  - EBS スナップショットと EBS 復旧ポイントの場合、暗号化タイプを変更せずに、同じボリュームから取得されるか、同じボリュームのコピーから取得されることを意味します。
  - S3 復旧ポイントの場合、ベースリソース ARN とターゲットリソース ARNs は、同じ基盤となる S3 バケットから作成する必要があります。
  - AMIs の場合、スナップショットのペアはベース AMI とターゲット AMI 間で比較され、増分スキャンのスナップショットを識別します。スナップショットの各ペアは、上記の条件を満たす必要があります。ターゲット AMI 内で、対応する一致するスナップショットがベース AMI にないスナップショットはスキップされます。

以前にスキャンしたバックアップリソースの再スキャン

前回のマルウェアスキャンの開始時刻から 10 分後に、同じリソースで新しいオンデマンドマルウェアスキャンを開始できます。前回のマルウェアスキャンの開始から 10 分以内に新しいマルウェアスキャンが開始された場合、リクエストは次のエラーになり、このリクエストに対してスキャン ID は生成されません。インスタンスを再スキャンするステップは、オンデマンドマルウェアスキャンを初めて開始する場合と同じです。

スキャンに必要な IAM ロール

フルスキャンまたは増分スキャンを開始するには、IAM ロールを渡す必要があります。このロールは、スキャンオペレーションの実行に必要なアクセス許可を提供します。 GuardDuty Malware Protection for Backup: IAM ロールのアクセス許可は、スキャンの実行に必要な関連する信頼ポリシーとともに、必要なアクセス許可の正確なリストを提供します。

リソーススキャンのステータスと結果の確認

GuardDuty はスキャン結果イベントを Amazon EventBridge のデフォルトイベントバスに発行します。GuardDuty は at-least-once 配信を使用します。つまり、同じオブジェクトに対して複数のスキャン結果を受け取る可能性があります。アプリケーションは、重複した結果を処理できるように設計することをお勧めします。スキャンされた各オブジェクトには 1 回だけ課金されます。

詳細については、「Malware Protection for Backup でのスキャンステータスと結果のモニタリング」を参照してください。

生成された検出結果の確認

検出結果の確認は、GuardDuty で Malware Protection for Backup を使用しているかどうかによって異なります。次のシナリオを考えてみてください。

GuardDuty サービスが有効になっている場合の Malware Protection for Backup の使用 (ディテクター ID)

マルウェアスキャンがスキャンされた Backup リソース内の潜在的に悪意のあるファイルを検出すると、GuardDuty は関連する検出結果を生成します。検出結果の詳細を確認し、推奨手順に従って検出結果を修復できる可能性があります。Export の検出結果の頻度に基づいて、生成された検出結果は S3 バケットと Amazon EventBridge イベントバスにエクスポートされます。

生成される検出結果タイプについては、「Malware Protection for Backup の検出Malware Protection for Backup の検出結果タイプ結果タイプ」を参照してください。

Malware Protection for Backup を独立した機能として使用する (ディテクター ID なし)

関連付けられたディテクター ID がないため、GuardDuty は検出結果を生成できません。バックアップリソースのスキャンステータスを知るには、GuardDuty が自動的にデフォルトのイベントバスに発行するスキャン結果を表示できます。

スキャンのステータスと結果の詳細については、「」を参照してくださいMalware Protection for Backup でのスキャンステータスと結果のモニタリング。

> **Note**
>
> Malware Protection for S3 も使用している場合、S3 ファイルが以前に NO_THREATS_FOUND とタグ付けされていても、オブジェクトが属する Backup Recovery Point の脅威のリストに同じファイルが表示される可能性があります。これは、サービスがマルウェア署名を頻繁に更新するためで、ファイルのステータスが変更された可能性があります。このような場合、GuardDuty は元の S3 バケットの ファイルのタグに戻って更新しないことに注意してください。ファイルに更新されたタグを適用する唯一の方法は、オブジェクトをバケットに再アップロードするか、S3 のオンデマンドスキャン機能を使用することです。

GuardDuty Malware Protection for Backup: IAM ロールのアクセス許可

マルウェアスキャンのために提供される顧客ロール

GuardDuty Malware Protection は、バックアップリソース、つまりスナップショット、AMIsEBS/EC2/S3 リカバリポイントでスキャンが開始されたときに、顧客ロール (スキャナーロール) が提供されることを想定しています。このロールは、GuardDuty がこれらの特定のリソースでスキャンを実行するために必要なアクセス許可を提供します。このロールのアクセス許可ポリシーと信頼ポリシーは、 にありますロールのアクセス許可と信頼ポリシー。以下のセクションでは、これらの各アクセス許可が必要な理由について説明します。

アクセス許可の詳細

- `ModifySnapshotAttribute` - 暗号化されていない、カスタマーマネージドキーで暗号化されたスナップショットに GuardDuty Malware Protection サービスアカウントからアクセスできるようにします。

- `CreateGrant` - GuardDuty Malware Protection が、GuardDuty サービスアカウントがアクセスできるカスタマーマネージドキー暗号化スナップショットからカスタマーマネージドキー暗号化 EBS ボリュームを作成してアクセスできるようにします。
- `RetireGrant` - GuardDuty Malware Protection が、暗号化されたスナップショットを読み取るためにカスタマーマネージドキーで作成された許可を廃止できるようにします。
- `ReEncryptTo` および `ReEncryptFrom` - EBS がカスタマーマネージドキーで暗号化されたスナップショットへのアクセスを GuardDuty に許可し、そこから暗号化されたボリュームを作成するために必要です。お客様は、共有中のスナップショットの ReEncryption をキー移行と考える場合がありますが、スナップショットは作成後も顧客の観点から変更できません。
- `ListSnapshotBlocks` および `GetSnapshotBlock` - EBS Direct APIsは、 AWS マネージドキー暗号化スナップショットのスナップショットブロックにアクセスするために使用されます。これは、 AWS マネージドキーで暗号化されたスナップショットにクロスアカウントでアクセスできないために行われます。
- `Decrypt` - 増分スキャンの一部として EBS Direct APIs を使用してメモリにダウンロードされたときに暗号化されたカスタマーマネージドキーであるベーススナップショットの復号を許可します。
- `ListChangedBlocks` - 2 つのスナップショット間で変更されたブロックのリストを取得するための増分スナップショットスキャンで使用される EBS Direct API。
- `DescribeKey` - GuardDuty Malware Protection がカスタマーアカウントの AWS マネージドキーの keyId を決定できるようにします。
- `DescribeImages` - AMI を記述して、AMI に属するスナップショットのリストを取得できるようにします。
- `DescribeRecoveryPoint` - サービスが復旧ポイントの詳細を取得し、復旧ポイントのリソースタイプを検証できるようにします。
- `CreateBackupAccessPoint`、`DescribeBackupAccessPoint`、`DeleteBackupAccessPoint` - 復旧ポイントへのアクセスに必要なアクセスポイントの作成、説明、削除をサービスに許可します。
- `kms:Decrypt` - S3 リカバリポイントスキャン中に、サービスが S3 リカバリポイント内のオブジェクトにアクセスできるようにします。

ロールの保護

ロールは、GuardDuty Malware Protection サービスプリンシパルを信頼する信頼ポリシーで設定する必要があります。これにより、GuardDuty サービス以外のプリンシパルがこのロールを引き受けることはできません。さらに、ポリシーを ではなく特定のリソースに絞り込むことをお勧めします*。

これには、スナップショット ID とキー ID が含まれます。これにより、ロールはそれらの特定のリソースにのみアクセスできるようになります。

> **Important**
>
> 設定が正しくないと、アクセス許可が不十分であるためにスキャンが失敗する可能性があります。

GuardDuty Malware Protection が KMS AWS で許可を使用する方法

GuardDuty Malware Protection では、KMS キーを使用するための許可が必要です。

暗号化されたスナップショットまたは暗号化されたスナップショットで構成される EC2 AMI でスキャンを開始すると、GuardDuty Malware Protection は CreateGrant リクエストを KMS に送信することで、ユーザーに代わって許可を作成します。 AWS これらの許可により、GuardDuty はアカウント内の特定のキーにアクセスできます。

GuardDuty Malware Protection では、カスタマーマネージドキーを次の内部オペレーションに使用する権限が必要です。

- DescribeKey リクエストを AWS に送信して、マルウェアスキャン用に送信されたリソースが暗号化されている対称カスタマーマネージドキーに関する詳細を取得します。
- CreateVolume API を使用して暗号化されたスナップショットから EBS ボリュームを作成し、同じキーでボリュームを暗号化します。
- 増分スキャン中に GetSnapshotBlock API を使用してスナップショットのスナップショットブロックにアクセスします。
- KMS AWS に Decrypt リクエストを送信して暗号化されたデータキーを復号し、スキャン中にスナップショット上のデータを読み取るために使用できます。

作成された権限を取り消すか、カスタマーマネージドキーへのサービスのアクセスをいつでも削除できます。その場合、GuardDuty はカスタマーマネージドキーによって暗号化されたデータにアクセスできず、そのデータに依存するオペレーションに影響します。

GuardDuty Malware Protection 暗号化コンテキスト

暗号化コンテキストは、データに関する追加のコンテキスト情報を含むキーと値のペアのオプションセットです。

データを暗号化するリクエストに暗号化コンテキストを含めると、 AWS KMS; は暗号化コンテキストを暗号化されたデータにバインドします。データを復号化するには、そのリクエストに (暗号化時と) 同じ暗号化コンテキストを含めます。

GuardDuty Malware Protection は、2 つの暗号化コンテキストのいずれかを使用します。

暗号化コンテキスト 1: キーは ですaws:guardduty:id。

```
"encryptionContext": {
    "aws:guardduty:id": "snap-11112222333344"
}
```

この暗号化コンテキスト
は、CreateGrant、Decrypt、GenerateDataKeyWithoutPlaintext、ReEncryptTo、RetireGrant、DescribeKey のグラントオペレーションで使用されます。

この暗号化コンテキストと許可オペレーションを使用して、現在のリソースに 1 つの許可が作成されます。

暗号化コンテキスト 2: キーは です aws:ebs:id

```
"encryptionContext": {
    "aws:ebs:id": "snap-11112222333344"
}
```

この暗号化コンテキストは、ReEncryptFrom、Decrypt、RetireGrant、DescribeKey のグラントオペレーションで使用されます。

これらの暗号化コンテキストと権限オペレーションを使用して 3 つの権限が作成されます。ReEncryptFrom 許可オペレーションを使用したターゲットスナップショット上の 1 つ。Decrypt, RetireGrant, DescribeKey オペレーションを使用したターゲットスナップショットの 2 番目のスナップショット。また、2 番目のグラントと同じグラントオペレーションを持つベーススナップショットの 3 つ目。

ロールのアクセス許可と信頼ポリシー

アクセス許可ポリシー

```
{
    "Version": "2012-10-17",
    "Statement": [{
```

```
            "Effect": "Allow",
            "Action": [
                "ebs:ListSnapshotBlocks",
                "ebs:ListChangedBlocks",
                "ebs:GetSnapshotBlock"
            ],
            "Resource": "arn:aws:ec2:*::snapshot/*"
        },
        {
            "Sid": "CreateGrantPermissions",
            "Effect": "Allow",
            "Action": "kms:CreateGrant",
            "Resource": "arn:aws:kms:*:*:key/*",
            "Condition": {
                "ForAnyValue:StringLike": {
                    "kms:EncryptionContext:aws:guardduty:id": "snap-*",
                    "kms:ViaService": [
                        "guardduty.*.amazonaws.com",
                        "backup.*.amazonaws.com"
                    ]
                },
                "ForAllValues:StringEquals": {
                    "kms:GrantOperations": [
                        "Decrypt",
                        "CreateGrant",
                        "GenerateDataKeyWithoutPlaintext",
                        "ReEncryptFrom",
                        "ReEncryptTo",
                        "RetireGrant",
                        "DescribeKey"
                    ]
                },
                "Bool": {
                    "kms:GrantIsForAWSResource": "true"
                }
            }
        },
        {
            "Sid": "CreateGrantPermissionsForReEncryptAndDirectAPIs",
            "Effect": "Allow",
            "Action": "kms:CreateGrant",
            "Resource": "arn:aws:kms:*:*:key/*",
            "Condition": {
                "ForAnyValue:StringLike": {
```

```
                    "kms:EncryptionContext:aws:ebs:id": "snap-*",
                    "kms:ViaService": [
                        "guardduty.*.amazonaws.com",
                        "backup.*.amazonaws.com"
                    ]
                },
                "ForAllValues:StringEquals": {
                    "kms:GrantOperations": [
                        "Decrypt",
                        "ReEncryptTo",
                        "ReEncryptFrom",
                        "RetireGrant",
                        "DescribeKey"
                    ]
                },
                "Bool": {
                    "kms:GrantIsForAWSResource": "true"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeImages",
                "ec2:DescribeSnapshots"
            ],
            "Resource": "*"
        },
        {
            "Sid": "ShareSnapshotPermission",
            "Effect": "Allow",
            "Action": [
                "ec2:ModifySnapshotAttribute"
            ],
            "Resource": "arn:aws:ec2:*:*:snapshot/*"
        },
        {
            "Sid": "ShareSnapshotKMSPermission",
            "Effect": "Allow",
            "Action": [
                "kms:ReEncryptTo",
                "kms:ReEncryptFrom"
            ],
            "Resource": "arn:aws:kms:*:*:key/*",
```

```
            "Condition": {
                "StringLike": {
                    "kms:ViaService": "ec2.*.amazonaws.com"
                }
            }
        },
        {
            "Sid": "DescribeKeyPermission",
            "Effect": "Allow",
            "Action": "kms:DescribeKey",
            "Resource": "arn:aws:kms:*:*:key/*"
        },
        {
            "Sid": "DescribeRecoveryPointPermission",
            "Effect": "Allow",
            "Action": [
                "backup:DescribeRecoveryPoint"
            ],
            "Resource": "*"
        },
        {
            "Sid": "CreateBackupAccessPointPermissions",
            "Effect" : "Allow",
            "Action" : [
                "backup:CreateBackupAccessPoint"
            ],
            "Resource": "arn:aws:backup:*:*:recovery-point:*"
        },
        {
            "Sid": "ReadAndDeleteBackupAccessPointPermissions",
            "Effect" : "Allow",
            "Action" : [
                "backup:DescribeBackupAccessPoint",
                "backup:DeleteBackupAccessPoint"
            ],
            "Resource": "*"
        },
        {
            "Sid": "KMSKeyPermissionsForInstantAccess",
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "arn:aws:kms:*:*:key/*",
```

```
            "Condition": {
                "StringLike": {
                    "kms:ViaService": "backup.*.amazonaws.com"
                }
            }
        }
    ]
}
```

信頼ポリシー

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "malware-protection.guardduty.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

(オプション) Malware Protection for Backup を個別に開始する (コンソールのみ)

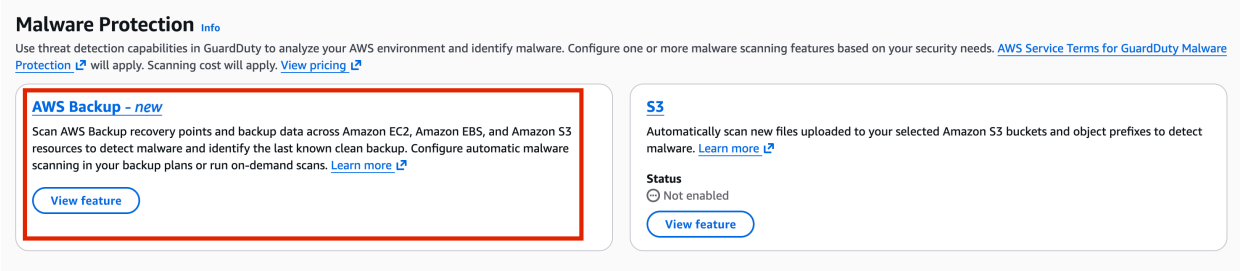
AWS アカウントの GuardDuty ステータスに関係なく、Malware Protection for Backup 脅威検出オプションの使用を開始する場合は、このオプションのステップを使用します。

GuardDuty で他の専用の保護プランも使用する場合は、Amazon GuardDuty サービスの使用を開始する必要があります。GuardDuty 保護プランの詳細については、GuardDuty の機能」を参照してください。

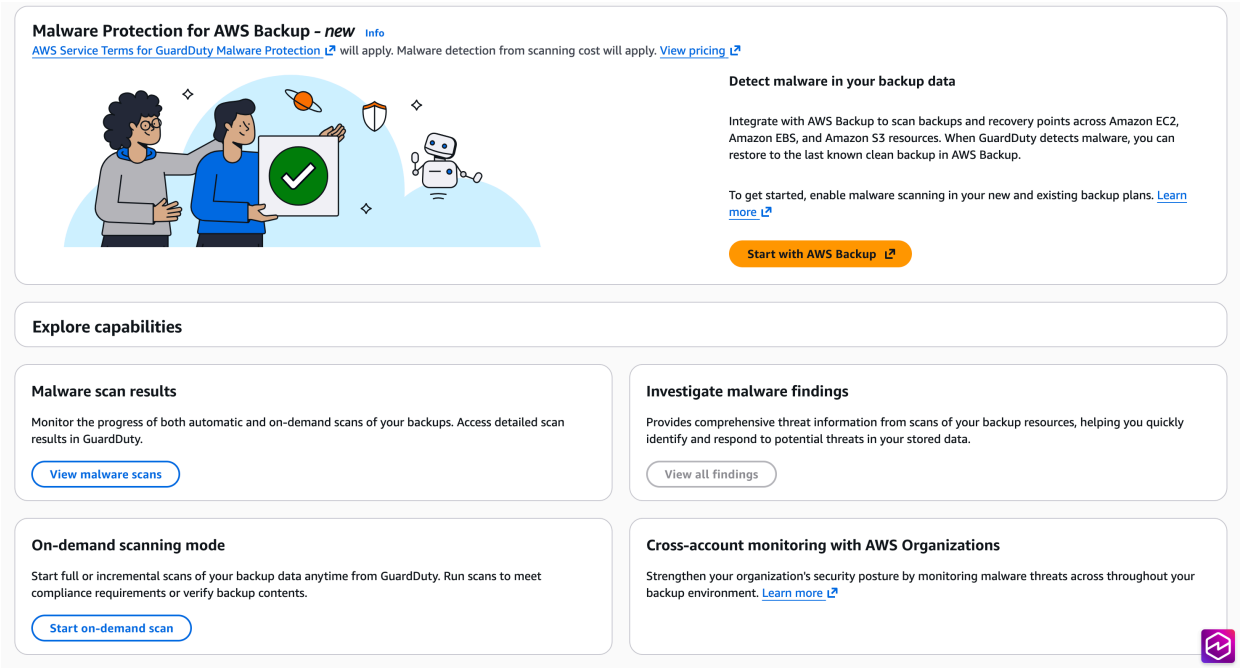
Malware Protection for Backup の使用を開始する手順

1. AWS マネジメントコンソールにサインインし、https://console.aws.amazon.com/guardduty/ で https://console.aws.amazon.com/guardduty/ コンソールを開きます。
2. Malware Protection 機能の検出を選択し、開始方法をクリックします。

3. 開始方法 をクリックすると、 AWS バックアップ機能や S3 Malware Protection 機能などのオプションから選択できます。



4. Malware Protection for Backup ページが表示され、オンデマンドスキャンを開始するか、マルウェアスキャンを表示するかを選択できます。



Malware Protection for Backup のオンデマンドスキャンの開始

コンソール

1. AWS マネジメントコンソールにサインインし、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. Malware Protection for Backup に移動し、オンデマンドスキャンの開始をクリックします。
3. フルスキャンと増分スキャンのどちらかを選択します。
    1. フルスキャンを開始するには、スキャンするリソースのリソース ARN を入力します。

2. 増分スキャンの場合は、ターゲットリソース ARN とベースラインリソース ARN を入力します。
3. スキャンされるリソースが復旧ポイントの場合は、それが属する AWS バックアップボールトの名前も入力する必要があります。
4. S3 PITR リソースの場合は、終了タイムスタンプを渡します。終了タイムスタンプを (現在の時刻 - 15 分) より前にすることはできません。

4. サービスアクセス - リソースにアクセスしてスキャンを実行するために必要なアクセス許可を持つロールを選択する必要があります。ポリシーの表示をクリックして、ロールに必要な正確なアクセス許可と、必要な信頼ポリシーを表示します。

要件に基づいてポリシーを変更するか、アクセス許可を正確なリソースに絞り込むことができます。IAM ロールを作成または更新する方法の詳細については、「」を参照してくださいGuardDuty Malware Protection for Backup: IAM ロールのアクセス許可。

IAM ロールのアクセス許可に関する問題については、「IAM ロールのアクセス許可のトラブルシューティングエラー」を参照してください。

API/CLI

オンデマンドマルウェアスキャンを開始するリソース`resourceArn`の を受け入れる StartMalwareScan を呼び出します。増分スキャンを開始する場合は、 の を渡`baselineResourceArn`します`incrementalScanDetails`。スキャン設定の一部として、スキャンの開始に必要なすべてのアクセス許可を持つ IAM ロールも指定する必要があります。スキャンするリソースが S3 PITR リソースの場合は、終了タイムスタンプを渡す必要があります。スキャンを正常に開始すると、`StartMalwareScan` は `scanId` を返します。`GetMalwareScan` API を呼び出して、開始されたスキャンの進行状況をモニタリングし、スキャンが完了したらスキャンの詳細を取得します。

Malware Protection for Backup でのスキャンステータスと結果のモニタリング

マルウェアスキャンが開始されると、GuardDuty はスキャンのステータスと結果をモニタリングするいくつかのメカニズムを提供します。次の表は、マルウェアスキャンに関連する値の一部を示しています。

考えられる 値
RUNNING、 COMPLETED 、 COMPLETED_WITH_ISSUES 、 FAILED、 または SKIPPED
FULL_SCAN 、または INCREMENTAL_SCAN
GUARDDUTY_INITIATED 、ON_DEMAND 、または BACKUP_INITIATED
NO_THREATS_FOUND 、または THREATS_FOUND

考えられる 値

*スキャンが完了していない場合、スキャン結果ステータスが存在しない可能性があることに注意してください。THREATS_FOUND のスキャン結果ステータスは、GuardDuty がマルウェアの存在を検出したことを示します。

S3 リカバリポイントの場合、COMPLETED_WITH_ISSUES は、一部のファイルがスキップまたは失敗したことを示します。AMI の場合、COMPLETED_WITH_ISSUES は、少なくとも 1 つのスナップショットがスキャンできなかったことを示します。スキップされた理由のリストについては、以下を参照してください。

スキャンは、さまざまな理由でスキップされる場合があります。次の表に、スキャンがスキップされる理由を示します。

Reason
顧客ロールには、サービスがスキャンを実行するために必要なアクセス許可がありません
スキャンしようとしているリソースがアカウントに存在しないか、スキャン中に削除されました
スナップショットのサイズが GuardDuty で現在サポートされているサイズを超えています

Reason
増分スキャンリクエストで指定されたリソースに違いはありません
リソースが想定どおりの状態ではありません。スキャンが増分の場合、基本復旧ポイントは AVAILABLE または COMPLETED 状態ではありません
増分スキャンの場合 - ベースリソースと現在のリソースが同じ系統のものではない
増分スキャンの場合 - ベースリソースが以前にスキャンされていないか、完了したスキャンが見つかりませんでした
増分スキャンの場合 - ベースリソースの作成日が現在のリソースの作成日より大きい

Reason
リクエストされたリソースタイプは増分スキャンをサポートしていません
パブリック AMI、エフェメラルストレージのみを持つ AMI、および使用可能な状態でない AMI はスキャンの対象ではありません
コールドストレージスナップショットはスキャンの対象ではありません
複合リソースタイプではスキャンはサポートされていません

Reason
リクエストされたリソースには、スキャンをサポートしていない Amazon Marketplace 製品コードが含まれています
AMI の は、40 個を超えるスナップショットのスキャンをサポートしていません
リクエストされたリソースの Ebs ブロックデバイスマッピングが見つかりませんでした

Reason
増分スキャンの場合 - ベースリソースの ARN は、予想されるリソースの ARN とは異なります。
スキャン内のすべてのファイルがスキップまたは失敗しました

スキャン結果の保持期間は 90 日間です。任意のアクセス方法を選択して、マルウェアスキャンのステータスを追跡します。

コンソールを使用したスキャンのモニタリング

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで [Malware scans] (マルウェアウェアのスキャン) を選択します。
3. [フィルター検索バー] で利用可能な次の [プロパティ] によってマルウェアスキャンをフィルターできます。
   - スキャン ID – マルウェアスキャンに関連付けられた一意の識別子。
   - アカウント ID – マルウェアスキャンが開始されたアカウント。
   - リソース ARN – スキャンに関連付けられた Amazon リソースに関連付けられた Amazon リソースネーム (ARN)。

- リソースタイプ – EC2 インスタンス、EBS スナップショット | EC2 AMI、EBS リカバリポイント、EC2 リカバリポイント、S3 リカバリポイントなど、スキャンに関連付けられたリソースのタイプ。
- ステータス – 実行中、スキップ済み、完了済み、問題ありで完了済み、失敗など、スキャンのスキャンステータス。
- スキャンタイプ — これがオンデマンド、GuardDuty 実行型、またはバックアップ実行型のマルウェアスキャンであったかどうかを示します。

API/CLI を使用したスキャンのモニタリング

- ListMalwareScans を呼び出して、RESOURCE_ARN、、SCAN_ID、、ACCOUNT_IDSCAN_TYPE GUARDDUTY_FINDING_ID、RESOURCE_TYPEおよび SCAN_STATUSでマルウェアスキャンをフィルタリングできますSCAN_START_TIME。また、GetMalwareScan を呼び出して、スキャン ID を入力として指定することで、スキャンのより詳細なメタデータを取得することもできます。GUARDDUTY_FINDING_ID フィルタ条件は、SCAN_TYPE が GuardDuty によって開始されたときに利用できます。
- 以下のコマンドで *filter-criteria* の例を変更し、CriterionKey一度に 1 つずつフィルタリングできます。のオプションCriterionKeyは、Resource_ARN、、SCAN_ID、ACCOUNT_ID、SCAN_TYPE、GUARDDUTY_FINDIおよび ですSCAN_START_TIME。*max-results* (最大 50) と *sort-criteria* を変更できます。AttributeName フィールドは に必須sort-criteriaであり、 に設定する必要がありますscanStartTime。次の例で、*##*の値はプレスホルダーです。アカウントに適した値に置き換えます。ListMalwareScans に以下CriterionKeyと同じ を使用する場合は、例をフィルタリングするEqualsValue*#######*に置き換えてください。

```
aws guardduty list-malware-scans --max-results 25 --sort-criteria
 '{"AttributeName": "scanStartTime", "OrderBy": "DESC"}' --filter-criteria
 '{"FilterCriterion":[{"CriterionKey":"RESOURCE_TYPE", "FilterCondition":
{"EqualsValue":"EBS_SNAPSHOT"}}] }'
```

```
aws guardduty get-malware-scan --scan-id abc123
```

- ListMalwareScans に対する上記のコマンドのレスポンスは、影響を受けるリソース (複数可) に関する詳細を含む最大 25 回のスキャンを返します。GetMalwareScan に対する上記のコマンドのレスポンスは、スキャンに関する詳細メタデータを含む 1 回のスキャンを返します。

EventBridge を使用したスキャンのモニタリング

Amazon EventBridge は、アプリケーションをさまざまなイベントソースのデータに簡単に接続できるようにするサーバーレスイベントバスサービスです。EventBridge は、独自のアプリケーション、Software-as-a-Service (SaaS) アプリケーション、Amazon のサービスからリアルタイムデータのストリームを配信し、そのデータを Lambda などのターゲットにルーティングします。これにより、サービスで発生したイベントをモニタリングし、イベント駆動型アーキテクチャを構築できます。詳細については、「Amazon EventBridge ユーザーガイド」を参照してください。

GuardDuty は、スキャンステータスが決定されると、EventBridge 通知をデフォルトのイベントバスに発行します。アカウントで EventBridge ルールを設定して、Amazon EventBridge と統合された他のサービスにイベントを送信できます。EventBridge の標準料金が適用されます。詳細については、「Amazon EventBridge の料金」を参照してください。

以下に示す値の多くは、この例のプレースホルダーであり、スキャンによって異なります。

Malware スキャン結果イベント

Backup の潜在的な詳細タイプ値:

- GuardDuty Malware Protection EBS スナップショットスキャン結果」
- GuardDuty Malware Protection EC2 AMI スキャン結果」
- GuardDuty Malware Protection S3 Recovery Point スキャン結果」
- GuardDuty Malware Protection EBS Recovery Point スキャン結果」
- GuardDuty Malware Protection EC2 Recovery Point スキャン結果」

サンプルイベントパターン:

```
{
      "detail-type": ["GuardDuty Malware Protection EC2 AMI Scan Result"],
      "source": ["aws.guardduty"]
}
```

脅威が見つからない EC2 AMI スキャンの通知スキーマの例:

```
{
    "version": "0",
    "id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
    "detail-type": "GuardDuty Malware Protection EC2 AMI Scan Result",
```

```
    "source": "aws.guardduty",
    "account": "1111222233334444",
    "time": "2025-11-01T00:00:00Z",
    "region": "us-east-1",
    "resources": ["arn:aws:ec2:us-east-1:1111222233334444:image/
ami-1234567890abcdef0"],
    "detail": {
        "schemaVersion": "1.0",
        "scanStatus": "COMPLETED",
        "resourceType": "EC2_AMI",
        "scanId": "d41d8cd98f00b204e9800998ecf8427e",
        "scanStatusReason": null,
        "scanType": "ON_DEMAND",
        "triggerType": "GUARDDUTY",
        "scanCategory": "FULL_SCAN",
        "scanStartTime": 1234567890123,
        "scanCompleteTime": 2345678901234,
        "scanResultDetails": {
            "scanResultStatus": "NO_THREATS_FOUND",
            "uniqueThreatCount": null
        }
    }
}
```

脅威が見つかった EC2 AMI スキャンの通知スキーマの例:

```
{
    "version": "0",
    "id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222",
    "detail-type": "GuardDuty Malware Protection EC2 AMI Scan Result",
    "source": "aws.guardduty",
    "account": "1111222233334444",
    "time": "2025-11-01T00:00:00Z",
    "region": "us-east-1",
    "resources": ["arn:aws:ec2:us-east-1:1111222233334444:image/
ami-1234567890abcdef0"],
    "detail": {
        "schemaVersion": "1.0",
        "scanStatus": "COMPLETED",
        "resourceType": "EC2_AMI",
        "scanId": "d41d8cd98f00b204e9800998ecf8427e",
        "scanStatusReason": null,
        "scanType": "ON_DEMAND",
```

```
        "triggerType": "GUARDDUTY",
        "scanCategory": "FULL_SCAN",
        "scanStartTime": 1234567890123,
        "scanCompleteTime": 2345678901234,
        "scanResultDetails": {
            "scanResultStatus": "THREATS_FOUND",
            "uniqueThreatCount": 1,
            "threats": {
                "name": "EICAR-Test-File (not a virus)",
                "source": "AMAZON",
                "count": 2,
                "itemDetails": [{
                    "resourceArn": "arn:aws:ec2:us-east-1:1111222233334444:snapshot/
snap-abcdef01234567890",
                    "hash":
 "e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855",
                    "itemPath": "/eicar.txt",
                    "additionalInfo": {
                        "versionId": null,
                        "deviceName": "/dev/sdf"
                    }
                }]
            }
        }
    }
}
```

スキップされた EC2 AMI スキャンの通知スキーマの例:

```
{
    "version": "0",
    "id": "a1b2c3d4-5678-90ab-cdef-EXAMPLE33333",
    "detail-type": "GuardDuty Malware Protection EC2 AMI Scan Result",
    "source": "aws.guardduty",
    "account": "1111222233334444",
    "time": "2025-11-01T00:00:00Z",
    "region": "us-east-1",
    "resources": ["arn:aws:ec2:us-east-1:1111222233334444:image/
ami-1234567890abcdef0"],
    "detail": {
        "schemaVersion": "1.0",
        "scanStatus": "SKIPPED",
        "resourceType": "EC2_AMI",
```

```
        "scanId": "d41d8cd98f00b204e9800998ecf8427e",
        "scanStatusReason": "UNSUPPORTED_AMI",
        "scanType": "ON_DEMAND",
        "triggerType": "GUARDDUTY",
        "scanCategory": "FULL_SCAN",
        "scanStartTime": 1234567890123,
        "scanCompleteTime": 2345678901234,
       "scanResultDetails": {
            "uniqueThreatCount": null,
            "threats": null
        }
    }
}
```

Malware Protection for Backup のクォータ

Malware Protection for Backup のクォータ

クォータ名	AWS デフォルトのクォータ値	調整可能かどうか	説明
すべてのリソースタイプの StartMalwareScan TPS 制限	10	いいえ	
サポートされるスナップショットの最大サイズ	2 TiB	いいえ	この制限は、増分スナップショットサイズではなく、完全なスナップショットサイズ (fullSnapshotSizeInBytes) に適用されます。詳細については、Amazon EC2 API リファレンス」の「スナップショット」を参照してください。 増分スキャンの場合、現在のスナップショットまたはベーススナップショットがこの制限を超えると、GuardDuty はスキャンをスキップします。

クォータ名	AWS デフォルトのクォータ値	調整可能かどうか	説明
スナップショットスキャンと AMI スキャンでサポートされるファイルシステム	• 新技術ファイルシステム (NTFS) • X ファイルシステム (XFS) • ext2 ファイルシステム • ext4 ファイルシステム • File Allocation Table (FAT) ファイルシステム • Virtual File Allocation Table (VFAT) ファイルシステム	いいえ	
マルウェアスキャンでサポートされている AMI 内のスナップショットの最大数	フルスキャンの場合は 40。40 個を超える場合、GuardDuty はすべてのスナップショットのうち 40 個のみをスキャンします。増分スキャンの場合、スキャンはスキップされます。	いいえ	

クォータ名	AWS デフォルトのクォータ値	調整可能かどうか	説明
S3 復旧ポイント内のオブジェクトの最大サイズ	100 GB	いいえ	GuardDuty でマルウェアがスキャンされる S3 オブジェクトの最大サイズ。
アーカイブバイトの抽出	100 GB	いいえ	GuardDuty がアーカイブファイルから抽出して分析できるデータの最大量。 GuardDuty は、100 GB を超えるアーカイブファイルの抽出をスキップします。

クォータ名	AWS デフォルトのクォータ値	調整可能かどうか	説明
アーカイブファイルの抽出	10,000	いいえ	GuardDuty がアーカイブファイルから抽出して分析できるファイルの最大数。アーカイブに含まれているファイルの数が 10,000 を超える場合は、GuardDuty はアーカイブファイルをスキップする必要があります。 **Note** 複合ファイルタイプは、これらの制限の対象となる可能性があります。ファイルタイプには、多目的インターネットメール拡張 (MIME) でエンコードされた E メールメッセージ、コンパイル済み Python (PYC) ファイル、コンパイル済み HTML ヘルプ (CHM) ファイル、すべてのインストーラ、および OpenDocument Format (ODF) ドキュメントが含まれますが、これらに限定されません。
アーカイブの最大深度	5	いいえ	GuardDuty が抽出できるネストされたアーカイブの最大レベル。アーカイブにこの値を超えてファイルがネストされている場合、その分のネストされたファイルはスキップされます。

GuardDuty RDS Protection

> **Note**
>
> RDS Protection と他のすべての保護プランは、GuardDuty コンソールの 1 ページから設定できます。詳細については、「保護プランの設定」を参照してください。

Amazon GuardDuty の RDS Protection は、Amazon Aurora および RDS for PostgreSQL データベースに対する潜在的な脅威について、RDS ログインとデータアクティビティを分析してプロファイリングします。

RDS Protection for ログインアクティビティは、Amazon Aurora (Amazon Aurora MySQL 互換エディションおよび Aurora PostgreSQL 互換エディション)、RDS for PostgreSQL、Amazon RDS for MySQL、Amazon RDS for MariaDB データベースへの潜在的なアクセス脅威についてログインイベントを分析します。サポートされているこれらのデータベースで疑わしいログイン動作を特定するのに役立ちます。GuardDuty は、異常なアクティビティがないか RDS ログインアクティビティ を継続的にモニタリングおよびプロファイリングします。例えば、これまで特定されなかった外部の攻撃者がデータベースに不正にアクセスした場合や、敵対者がデータベースのパスワードを推測してブルートフォースアクセスを試みた場合などです。

RDS Protection for Data Activity は、Amazon Aurora for PostgreSQL および RDS for PostgreSQL データベースのデータベースクエリパターンを分析して、潜在的なデータアクセスと操作の脅威を検出します。GuardDuty は、異常な動作RDS データアクティビティを継続的にモニタリングし、プロファイルを作成します。たとえば、アクターがそのデータベースまたはユーザーに対して通常よりも大幅に多くの行を読み取る、データを削除または削除する、既知の悪意のある IP アドレスまたは Tor IP アドレスからデータベースにアクセスするなどです。

RDS Protection をまだ有効にしていないアカウントについては、30-day free trial の詳細を確認し、この機能を有効にすることができます。この機能を有効にするには、「マルチアカウント環境での RDS Protection の有効化」または「スタンドアロンアカウントの RDS Protection の設定」を参照してください。

> **メモ**
>
> RDS for PostgreSQL リードレプリカインスタンスでは、プライマリデータベースインスタンスがサポートされているデータベースバージョンにあり、プライマリデータベース

から正常にレプリケートされている必要があります。リードレプリカの詳細については、「Amazon RDS ユーザーガイド」の「DB インスタンスのリードレプリカの使用」を参照してください。

RDS Protection は追加のインフラストラクチャが不要で、データベースインスタンスのパフォーマンスに影響を与えないように設計されています。RDS Protection が疑わしい、または異常なログイン試行、流出、またはデータ破壊アクティビティを検出すると、GuardDuty は侵害された可能性のあるデータベースに関する詳細RDS Protection の検出結果タイプを含む 1 つ以上の を生成します。

30 日間の無料トライアル

- AWS アカウント 新しいリージョンの で GuardDuty を初めて有効にすると、30 日間の無料トライアルが受けられます。この場合、GuardDuty は無料トライアルに含まれている RDS Protection も有効にします。RDS Protection ログインアクティビティモニタリングと RDS Protection データアクティビティモニタリングは、データベースのアクティビティのモニタリングを開始します。
- 既に GuardDuty を使用していて、初めて新しいリージョンで RDS Protection を有効にすると、このリージョンのアカウントに対して RDS Protection の 30 日間の無料トライアルが適用されます。
- でログインアクティビティに RDS Protection を既に使用 AWS アカウント していて、データアクティビティに RDS Protection を初めて有効にすると、アカウントはデータアクティビティに RDS Protection の 30 日間の無料トライアルを受け取ります。
- 任意のリージョンでいつでも RDS Protection または RDS Protection のデータアクティビティモニタリングを無効にすることができます。
- 30 日間の無料トライアルでは、そのアカウントとリージョンの使用コストの見積もりを取得できます。30 日間の無料トライアルが終了すると、RDS Protection は自動的に無効になることはありません。このリージョンのアカウントで使用コストが発生し始めます。詳細については、「GuardDuty の使用状況のモニタリングとコストの見積もり」を参照してください。

RDS Protection 機能が有効になっていない場合、GuardDuty は異常または疑わしいログインまたはデータ動作を検出しません。RDS Protection を無効にすると、GuardDuty は RDS アクティビティのモニタリングを直ちに停止し、サポートされているデータベースインスタンスに対する潜在的な脅威を検出したり、関連する検出結果タイプを生成したりしません。

Aurora PostgreSQL Limitless Databases がサポートされている AWS リージョン については、「Aurora PostgreSQL Limitless Database の要件」を参照してください。

サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース

次の表は、RDS Protection ログインアクティビティモニタリングとデータアクティビティモニタリングでサポートされている Aurora および Amazon RDS データベースのバージョンを示しています。GuardDuty は、将来的に追加の Amazon RDS データベースに拡張される可能性があります。

Amazon Aurora および Amazon RDS DB エンジン	サポートされているエンジンバージョンをモニタリングするログインアクティビティ	サポートされているエンジンバージョンをモニタリングするデータアクティビティ
Aurora MySQL	• 2.10.2 以降 • 3.02.1 以降 • 以降のメジャーバージョン	サポートされていません
Aurora PostgreSQL	• 10.23 以降 • 11.12 以降 • 12.7 以降 • 13.3 以降 • 14.3 以降 • 15.2 以降 • 16.1 以降 • 以降のメジャーバージョン	• 11.19 以降 • 12.14 以降 • 13.10 以降 • 14.7 以降 • 15.2 以降 • 以降のメジャーバージョン
RDS for PostgreSQL	• 11.17 以降 • 12.12 以降 • 13.8 以降 • 14.5 以降 • RDS for PostgreSQL バージョン 15 • RDS for PostgreSQL バージョン 16 • RDS for PostgreSQL バージョン 17	• 11.20 以降 • 12.15 以降 • 13.11 以降 • 14.8 以降 • 15.3 以降 • 以降のメジャーバージョン

Amazon Aurora および Amazon RDS DB エンジン	サポートされているエンジンバージョンをモニタリングするログインアクティビティ	サポートされているエンジンバージョンをモニタリングするデータアクティビティ
	• RDS for PostgreSQL バージョン 18 • 以降のメジャーバージョン	
Amazon Aurora PostgreSQL Limitless Database	• `16.4-limitless` • 以降のメジャーバージョン	サポートされていません
Amazon RDS for MariaDB	• 10.6.25 以降 • 10.11.16 以降 • 11.4.10 以降 • 11.8.6 以降 • 以降のメジャーバージョン	サポートされていません
Amazon RDS for MySQL	• 8.0.45 以降 • 8.4.8 以降 • 以降のメジャーバージョン	サポートされていません

RDS ログインアクティビティ

RDS Protection 機能を有効にすると、GuardDuty は Aurora および Amazon RDS サービスから直接データベースの RDS ログインアクティビティのモニタリングを自動的に開始します。RDS ログインアクティビティは、 AWS 環境で に対して行われたログイン試行の成功と失敗の両方をキャプチャサポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベースします。異常なログイン動作の兆候が見られる場合、GuardDuty は侵害された可能性があるデータベースの詳細情報が含まれる検出結果を生成します。RDS Protection を初めて有効にする場合、またはデータベースインスタンスを新しく作成した場合は、通常の動作をベースラインにするための学習期間があります。このため、新たに有効になったデータベースインスタンスや新しく作成されたデータベースインスタンスでは、最長で 2 週間、関連する異常ログインが検出されないことがあります。

RDS Protection が潜在的な脅威 (一連の成功、失敗、未完了のログイン試行で異常なパターンなど) を検出すると、GuardDuty は 1 つ以上の「RDS Protection の検出結果タイプ」を生成します。検出結果タイプに応じて、「RDS ログインアクティビティベースの異常」などの異常な動作に関する詳

細が含まれる場合があります。GuardDuty では、RDS ログインアクティビティログは利用できません。

RDS データアクティビティ

データアクティビティに対して RDS Protection を有効にすると、GuardDuty はサポートされている Aurora および Amazon RDS データベースのデータベースアクティビティパターンを分析して、潜在的なデータアクセスと操作の脅威を特定します。異常なデータアクティビティを検出すると、GuardDuty は侵害された可能性のあるデータベースに関する詳細を含む検出結果を生成します。RDS Protection for データアクティビティを初めて有効にする場合、または新しいデータベースインスタンスを作成する場合、通常の動作のベースラインを確立するための学習期間は最大 2 週間です。この間、関連する異常なデータアクティビティの検出結果がない可能性があります。

RDS Protection for Data Activity が潜在的な脅威を検出すると、GuardDuty は 1 つ以上の を生成しますRDS Protection の検出結果タイプ。例としては、アクターがそのデータベースまたはユーザーに対して通常よりも大幅に多くの行を読み取る、データを削除または削除する、既知の悪意のある IP アドレスまたは Tor IP アドレスからデータベースにアクセスするなどがあります。検出結果のタイプによっては、検出結果に などの異常な動作に関する詳細が含まれる場合がありますRDS データアクティビティベースの異常。GuardDuty では、RDS データアクティビティログは利用できません。

マルチアカウント環境での RDS Protection の有効化

マルチアカウント環境では、委任 GuardDuty 管理者アカウントのみが、組織内のメンバーアカウントの RDS データアクティビティモニタリングの追加設定を含む RDS Protection 機能を有効または無効にすることができます。GuardDuty メンバーアカウントからは、この設定を変更できません。委任 GuardDuty 管理者アカウントは、 を使用してメンバーアカウントを管理します AWS Organizations。この委任 GuardDuty 管理者アカウントは、組織に参加するすべての新しいアカウントの RDS ログインアクティビティモニタリングと RDS データアクティビティモニタリングを自動的に有効にすることを選択できます。マルチアカウント環境の詳細については、「複数のアカウントの管理」を参照してください。

委任 GuardDuty 管理者アカウントでの RDS Protection の有効化

任意のアクセス方法を選択して、委任 GuardDuty 管理者アカウントの RDS データアクティビティモニタリングの追加設定を含む RDS Protection を有効にします。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

2. ナビゲーションペインで、保護プランを選択します。
3. すべての有効化の設定 を選択します。
4. RDS Protection で、RDS Login Activity Monitoring エントリと RDS Data Activity Monitoring エントリを見つけます。
5. このアカウント列で、オンにするエントリごとに有効を選択するか、オフのままにするには無効を選択します。

> **Note**
>
> RDS データアクティビティモニタリングを有効にする前に、RDS ログインアクティビティモニタリングを有効にする必要があります。

API/CLI

ユーザー独自のリージョンレベルのディテクター ID を使用し、`features` オブジェクト `name` を `RDS_LOGIN_EVENTS` として、`status` を `ENABLED` として渡して、updateDetector API オペレーションを実行します。RDS データアクティビティのモニタリングも有効にするには、を `status`に設定`AdditionalConfiguration`して `name` の下に `RDS_DATA_RISK`を渡しますENABLED。

または、 AWS CLI を使用して RDS Protection を有効にすることもできます。次のコマンドを実行し、*12abc34d567e8fa901bc2d34e56789f0* をアカウントのディテクター ID に置き換え、*us-east-1* を RDS Protection を有効にするリージョンに置き換えます。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

次の例では、委任管理者アカウントで RDS ログインアクティビティのモニタリングを有効にし、 `AdditionalConfiguration` を使用して RDS データアクティビティのモニタリングも有効にします。

```
aws guardduty update-detector \
    --detector-id 12abc34d567e8fa901bc2d34e56789f0 \
    --region us-east-1 \
    --features '[
        {
```

```
            "Name": "RDS_LOGIN_EVENTS",
            "Status": "ENABLED",
            "AdditionalConfiguration": [
                {
                    "Name": "RDS_DATA_RISK",
                    "Status": "ENABLED"
                }
            ]
        }
    ]'
```

すべてのメンバーアカウントの RDS Protection を自動で有効にする

任意のアクセス方法を選択して、RDS データアクティビティモニタリングの追加設定を含む RDS Protection 機能をすべてのメンバーアカウントで有効にします。これには、既存のメンバーアカウントと、組織に参加する新しいアカウントが含まれます。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   必ず委任 GuardDuty 管理者アカウントの認証情報を使用してください。
2. ナビゲーションペインで、保護プランを選択します。
3. すべての有効化の設定 を選択します。
4. RDS Protection で、すべてのアカウントの有効化を選択します。このアクションにより、組織内の既存のアカウントと新しいアカウントの両方について RDS Protection が自動的に有効になります。
5. また、すべてのアカウントの RDS データアクティビティモニタリングを有効にするには、追加設定を展開し、RDS データアクティビティモニタリングのすべてのアカウントに対して有効化を選択します。
6. すべて保存を選択し、確認して保存を選択します。

> **Note**
>
> メンバーアカウントの設定を更新するには、最大 24 時間かかる場合があります。

API/CLI

既存のメンバーや後で参加する新しいアカウントなど、組織内のすべてのメンバーアカウントに対して RDS Protection を有効にするには、独自の`###### ID`を使用して UpdateOrganizationConfiguration API オペレーションを呼び出します。RDS_LOGIN_EVENTS 機能の ALL AutoEnableを に設定し、 を AutoEnableに設定RDS_DATA_RISKAdditionalConfigurationして RDS データアクティビティのモニタリングも自動有効化ALLします。

または、 AWS CLI を使用して RDS Protection を有効にすることもできます。次のコマンドを実行し、*12abc34d567e8fa901bc2d34e56789f0* をアカウントのディテクター ID に置き換え、*us-east-1* を RDS Protection を有効にするリージョンに置き換えます。

アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

次の例では、組織内のすべてのメンバーアカウントRDS_DATA_RISKに対して RDS_LOGIN_EVENTSと の両方を自動有効化します。

```
aws guardduty update-organization-configuration \
    --detector-id 12abc34d567e8fa901bc2d34e56789f0 \
    --region us-east-1 \
    --auto-enable-organization-members ALL \
    --features '[
        {
            "Name": "RDS_LOGIN_EVENTS",
            "AutoEnable": "ALL",
            "AdditionalConfiguration": [
                {
                    "Name": "RDS_DATA_RISK",
                    "AutoEnable": "ALL"
                }
            ]
        }
    ]'
```

> **Note**
>
> メンバーアカウントの設定を更新するには、最大 24 時間かかる場合があります。

すべての既存のアクティブなメンバーアカウントのために RDS Protection を有効にする

任意のアクセス方法を選択して、組織内のすべての既存のアクティブなメンバーアカウントに対して RDS データアクティビティモニタリングの追加設定を含む RDS Protection を有効にします。GuardDuty が既に有効になっているメンバーアカウントは、既存のアクティブメンバーと呼ばれます。

> **Note**
>
> RDS ログインアクティビティモニタリングは、そのアカウントで RDS データアクティビティモニタリングを有効にする前に、メンバーアカウントで有効にする必要があります。

Console

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   委任 GuardDuty 管理者アカウントの認証情報を使用してサインインします。
2. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
3. アカウントページで、有効にする既存のアクティブなメンバーアカウントを選択します。すべてのアカウントを選択するには、テーブルヘッダーのチェックボックスを使用します。
4. Edit Protection Plans ドロップダウンメニューから RDS Login Activity を選択し、Enable を選択します。
5. 選択したアカウントで RDS データアクティビティのモニタリングも有効にするには、保護プランの編集ドロップダウンメニューを再度開き、RDS データアクティビティを選択し、有効化を選択します。RDS Login Activity Monitoring は、選択した各アカウントで既に有効になっている必要があります。

API/CLI

ユーザー独自の`###### ID` を使用して updateMemberDetectors API オペレーションを実行します。`RDS_LOGIN_EVENTS` 機能を渡して、RDS データアクティビティのモニタリングも制御`RDS_DATA_RISKAdditionalConfiguration`します。

または、 AWS CLI を使用して RDS Protection を有効にすることもできます。次のコマンドを実行し、*12abc34d567e8fa901bc2d34e56789f0* をアカウントのディテクター ID に置き換え、*us-east-1* を RDS Protection を有効にするリージョンに置き換えます。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

次の例では、既存のアクティブなメンバーアカウントで RDS ログインアクティビティのモニタリングを有効にし、 `AdditionalConfiguration` を使用して RDS データアクティビティのモニタリングも有効にします。

```
aws guardduty update-member-detectors \
    --detector-id 12abc34d567e8fa901bc2d34e56789f0 \
    --region us-east-1 \
    --account-ids 111122223333 \
    --features '[
        {
            "Name": "RDS_LOGIN_EVENTS",
            "Status": "ENABLED",
            "AdditionalConfiguration": [
                {
                    "Name": "RDS_DATA_RISK",
                    "Status": "ENABLED"
                }
            ]
        }
    ]'
```

スペースで区切られたアカウント ID のリストを渡すこともできます。

コードが正常に実行されると、`UnprocessedAccounts` の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

新しいメンバーアカウントの RDS Protection を自動で有効にする

任意のアクセス方法を選択して、組織に参加する新しいアカウントの RDS ログインアクティビティモニタリングと RDS データアクティビティモニタリングを有効にします。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   必ず委任 GuardDuty 管理者アカウントの認証情報を使用してください。
2. ナビゲーションペインで、保護プランを選択します。
3. すべての有効化の設定 を選択します。
4. [アカウントを手動で設定] を選択します。
5. RDS Protection で、新しいメンバーアカウントに対して自動有効化を選択します。このステップにより、新しいアカウントが組織に参加するたびに、そのアカウントのために RDS Protection が自動的に有効になります。この設定を変更できるのは、組織の委任 GuardDuty 管理者アカウントだけです。
6. 新しいアカウントの RDS データアクティビティモニタリングも自動有効化するには、追加設定を展開し、RDS データアクティビティモニタリングの新しいメンバーアカウントに対して自動有効化を選択します。
7. [保存] を選択します。

API/CLI

組織に参加する新しいアカウントの RDS Protection を自動的に有効にするには、独自の*###### ID* を使用して UpdateOrganizationConfiguration API オペレーションを呼び出します。RDS_LOGIN_EVENTS 機能の NEW AutoEnableを に設定し、RDS_DATA_RISKAdditionalConfigurationを AutoEnableに設定して RDS データアクティビティのモニタリングも自動有効化NEWします。新しいアカウントの機能の自動有効化を停止するには、そのAutoEnable値を に設定しますNONE。

または、 AWS CLI を使用して RDS Protection を有効にすることもできます。次のコマンドを実行し、*12abc34d567e8fa901bc2d34e56789f0* をアカウントのディテクター ID に置き換え、*us-east-1* を RDS Protection を有効にするリージョンに置き換えます。

アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

次の例では、組織に参加する新しいアカウントRDS_DATA_RISKに対して RDS_LOGIN_EVENTSと の両方を自動有効化します。

```
aws guardduty update-organization-configuration \
```

```
    --detector-id 12abc34d567e8fa901bc2d34e56789f0 \
    --region us-east-1 \
    --auto-enable-organization-members NEW \
    --features '[
        {
            "Name": "RDS_LOGIN_EVENTS",
            "AutoEnable": "NEW",
            "AdditionalConfiguration": [
                {
                    "Name": "RDS_DATA_RISK",
                    "AutoEnable": "NEW"
                }
            ]
        }
    ]'
```

メンバー アカウントの RDS Protection を選択的に有効にする

任意のアクセス方法を選択して、メンバーアカウントの RDS ログインアクティビティと RDS データアクティビティのモニタリングを選択的に有効にします。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   必ず委任 GuardDuty 管理者アカウントの認証情報を使用してください。
2. ナビゲーションペインで、[Accounts] (アカウント) を選択します。

   アカウントページで、メンバーアカウントのステータスについて RDS ログインアクティビティと RDS データアクティビティ列を確認します。
3. RDS ログインアクティビティを選択的に有効または無効にするには

   RDS Protection を設定するアカウントを選択します。一度に複数のアカウントを選択できます。[保護プランの編集] ドロップダウンメニューで、[RDS ログインアクティビティ] を選択し、適切なオプションを選択します。
4. RDS データアクティビティを選択的に有効または無効にするには

   RDS データアクティビティモニタリングを設定するアカウントを選択します。一度に複数のアカウントを選択できます。保護プランの編集ドロップダウンメニューで、RDS データアクティビティを選択し、適切なオプションを選択します。

API/CLI

メンバーアカウントの RDS Protection を選択的に有効または無効にするには、ユーザー独自の`###### ID` を使用して updateMemberDetectors API オペレーションを起動します。RDS データアクティビティのモニタリング`AdditionalConfiguration`を個別に制御するには、`RDS_LOGIN_EVENTS`機能を`RDS_DATA_RISK`渡します。

または、 AWS CLI を使用して RDS Protection を有効にすることもできます。次のコマンドを実行し、*12abc34d567e8fa901bc2d34e56789f0* をアカウントのディテクター ID に置き換え、*us-east-1* を RDS Protection を有効にするリージョンに置き換えます。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

次の例では、`RDS_LOGIN_EVENTS` と `RDS_DATA_RISK` の両方を有効にします。

```
aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0
 --region us-east-1 --account-ids 111122223333 --features '[{"Name":
 "RDS_LOGIN_EVENTS", "Status": "ENABLED", "AdditionalConfiguration": [{"Name":
 "RDS_DATA_RISK", "Status": "ENABLED"}]}]'
```

RDS データアクティビティのモニタリングをオフにしながら RDS ログインアクティビティのモニタリングをオンにするには、 `RDS_LOGIN_EVENTS Status` を に、 `ENABLED` `RDS_DATA_RISKStatus`を に設定しますDISABLED。

```
aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0
 --region us-east-1 --account-ids 111122223333 --features '[{"Name":
 "RDS_LOGIN_EVENTS", "Status": "ENABLED", "AdditionalConfiguration": [{"Name":
 "RDS_DATA_RISK", "Status": "DISABLED"}]}]'
```

> **Note**
>
> スペースで区切られたアカウント ID のリストを渡すこともできます。

コードが正常に実行されると、`UnprocessedAccounts` の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

スタンドアロンアカウントの RDS Protection の設定

スタンドアロンアカウントは、特定の の で保護プランを有効または無効にする決定を所有 AWS アカウント します AWS リージョン。

アカウントが を通じて AWS Organizations、または招待方法によって GuardDuty 管理者アカウントに関連付けられている場合、このセクションはアカウントには適用されません。詳細については、「マルチアカウント環境での RDS Protection の有効化」を参照してください。

RDS Protection を有効にすると、GuardDuty はアカウントでサポートされているデータベースの「RDS ログインアクティビティ」のモニタリングを開始します。

任意のアクセス方法を選択して、スタンドアロンアカウントの RDS Protection を設定します。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、保護プランを選択します。
3. すべての有効化の設定 を選択します。RDS Protection で、Enable を選択して RDS Protection を有効にします。
4. すべて保存を選択し、確認して保存を選択します。

API/CLI

ユーザー独自のリージョンレベルのディテクター ID を使用し、`features` オブジェクト `name` を `RDS_LOGIN_EVENTS` として、`status` を `ENABLED` として渡して、updateDetector API オペレーションを実行します。RDS データアクティビティのモニタリングも有効にするには、を `status`に設定`AdditionalConfiguration`して `name` の下に `RDS_DATA_RISK`を渡しますENABLED。

または、 AWS CLI を使用して RDS Protection を有効にすることもできます。次のコマンドを実行し、*`12abc34d567e8fa901bc2d34e56789f0`* をアカウントのディテクター ID に置き換え、*`us-east-1`* を RDS Protection を有効にするリージョンに置き換えます。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

次の例では、`RDS_LOGIN_EVENTS` と `RDS_DATA_RISK` の両方を有効にします。

```
aws guardduty update-detector \
    --detector-id 12abc34d567e8fa901bc2d34e56789f0 \
    --region us-east-1 \
    --features '[
        {
            "Name": "RDS_LOGIN_EVENTS",
            "Status": "ENABLED",
            "AdditionalConfiguration": [
                {
                    "Name": "RDS_DATA_RISK",
                    "Status": "ENABLED"
                }
            ]
        }
    ]'
```

RDS Protection のモニタリングに関する問題のトラブルシューティング

GuardDuty RDS Protection は、サポートされているデータベース に対する潜在的なアクセス脅威がないか、ログインアクティビティを分析してプロファイリングします。セキュリティテレメトリを効果的に収集するために、GuardDuty ではデータベースを問題なく設定して運用する必要があります。データベースの設定ミスや問題が発生した場合、セキュリティモニタリングに影響する可能性があります。

以下のセクションでは、一般的な問題とトラブルシューティングの手順について説明します。

RDS ストレージがいっぱいです

RDS インスタンスのストレージが不足すると、GuardDuty はセキュリティテレメトリを収集しない場合があります。DB インスタンスストレージ容量の割り当て (`storage-full`) に達することは重大なステータスであり、RDS ではこの問題をすぐに修正することをお勧めします。詳細については、Amazon RDS ユーザーガイドの「DB インスタンスステータスの表示」を参照してください。

`storage-full` ステータスを解決するには、以下のいずれかのアクションを実行します。

- ストレージの自動スケーリングを有効にする (推奨) — Amazon RDS ストレージの自動スケーリングを有効にして、ストレージ容量を自動的に管理し、将来の `storage-full` 状態を防止します。

詳細については、「Amazon RDS ユーザーガイド」の「Amazon RDS ストレージのオートスケーリングによる容量の自動管理」を参照してください。

- ストレージ使用率のモニタリング – 以下のいずれかの方法を使用して、ストレージ使用率を確認します。
  - CloudWatch メトリクスを使用して、ストレージの詳細を表示します。詳細については、「Amazon CloudWatch ユーザーガイド」の「CloudWatch Database Insights」を参照してください。
  - 「Amazon RDS ユーザーガイド」の「Amazon RDS インスタンスのメトリクスのモニタリング」の手順に従って、ストレージメトリクスを表示します。
- ストレージ容量の変更 – インスタンスのストレージ容量を増やす方法については、「Amazon RDS ユーザーガイド」の「DB インスタンスのストレージ容量を増加する」を参照してください。

RDS for PostgreSQL のプライマリデータベースでサポートされていないバージョン

RDS for PostgreSQL リードレプリカインスタンスでは、プライマリデータベースインスタンスがサポートされているデータベースバージョンにあり、かつプライマリデータベースから正常にレプリケートされている必要があります。GuardDuty は、これらの要件が満たされた場合にのみインスタンスをモニタリングします。

サポートされていないバージョンの問題を解決するには、次のいずれかを実行します。

- データベースバージョンの互換性を検証する – プライマリ RDS for PostgreSQL データベースでサポートされているバージョンの 1 つが実行されていることを確認します。詳細については、「サポートされているデータベース」を参照してください。
- レプリケーションの潜在的な問題に対処する – プライマリインスタンスとレプリカインスタンス間のレプリケーションの問題を確認して解決します。詳細については、「Amazon RDS ユーザーガイド」の「RDS for PostgreSQL でのリードレプリカの使用」を参照してください。

セキュリティに関するその他の考慮事項

組織に厳格なコンプライアンス要件がある場合は、RDS Protection の使用に加えて、データベース監査を実装することをお勧めします。セキュリティ責任と責任共有モデルの詳細については、「Amazon RDS ユーザーガイド」の「Amazon RDS でのセキュリティ」を参照してください。

GuardDuty Lambda Protection

> **Note**
>
> Lambda Protection は、GuardDuty コンソールの 1 つのページから他のすべての保護プランとともに設定できます。詳細については、「保護プランの設定」を参照してください。

Lambda Protection は、 AWS 環境内で AWS Lambda 関数が呼び出されたときに潜在的なセキュリティ脅威を特定するのに役立ちます。Lambda Protection を有効にすると、GuardDuty が Lambda ネットワークアクティビティログのモニタリングを開始します。例えば、アカウントのすべての Lambda 関数が生成する VPC フローログ (VPC ネットワーキングを使用しないログも含む) や、Lambda 関数が呼び出されたときに生成されるログなどです。悪意の可能性があるコードが Lambda 関数に存在することを示す疑わしいネットワークトラフィックが識別されると、1 つ以上の Lambda Protection の検出結果タイプが生成されます。

30 日間の無料トライアル

次に、アカウントに適用される 30 日間無料トライアルの仕組みを示します。

- AWS アカウント 新しいリージョンの で GuardDuty を初めて有効にすると、30 日間の無料トライアルが受けられます。この場合、Lambda Protection も無料トライアルに含まれているため一緒に有効になります。
- 既に GuardDuty を使用している場合に、初めて Lambda Protection を有効にすると、このリージョンのアカウントに Lambda Protection の 30 日間無料トライアルが適用されます。
- Lambda Protection はいつでもどのリージョンでも、無効にできます。
- 30 日間の無料トライアルでは、そのアカウントとリージョンの使用コストの見積もりを取得できます。30 日間無料トライアルが終了しても、Lambda Protection が自動的に無効になることはありません。このリージョンのアカウントで使用コストが発生し始めます。詳細については、「GuardDuty の使用状況のモニタリングとコストの見積もり」を参照してください。

Lambda ネットワークアクティビティログは変更されることがあります。例えば、Lambda 関数を呼び出すと生成される DNS クエリデータなど、他のネットワークアクティビティに拡張されるといった場合です。他の形式のネットワークアクティビティモニタリングへの拡張により、GuardDuty が Lambda Protection で処理するデータ量が増加します。これは Lambda Protection の使用コストに直接影響します。GuardDuty が追加のネットワークアクティビティログのモニタリングを開始するた

びに、リリースの少なくとも 30 日前に、Lambda Protection を有効にしたアカウントに通知が届きます。

> **Note**
>
> Lambda Network Activity Monitoring には Lambda@Edge 関数のログは含まれません。

Lambda Network Activity Monitoring

Lambda Protection を有効にすると、アカウントに関連付けられた Lambda 関数が呼び出されたときに生成される Lambda ネットワークアクティビティログが GuardDuty によって監視されます。これにより、Lambda 関数に対する潜在的なセキュリティ脅威を検出できます。VPC ネットワークを使用するように設定されている Lambda 関数では、Lambda for GuardDuty で作成された Elastic Network Interface (ENI) の VPC フローログを有効にする必要はありません。GuardDuty では、検出結果を生成するために処理された Lambda ネットワークアクティビティのログデータの量 (GB 単位) に対してのみ課金されます。GuardDuty では、スマートフィルターを適用し、脅威検出に関連する Lambda ネットワークアクティビティログのサブセットを分析することでコストを最適化します。

GuardDuty は Lambda ネットワークアクティビティログ (VPC や VPC 以外のフローログなど) を管理したり、アカウントへのアクセスを可能にしたりすることはありません。

マルチアカウント環境での Lambda Protection の有効化

マルチアカウント環境で組織内のメンバーアカウントの Lambda Protection を有効または無効にできるのは、委任 GuardDuty 管理者アカウントのみです。GuardDuty メンバーアカウントからは、この設定を変更できません。委任 GuardDuty 管理者アカウントは、 を使用してメンバーアカウントを管理します AWS Organizations。委任 GuardDuty 管理者アカウントでは、組織に加わるすべての新規アカウントに対して、Lambda Network Activity Monitoring の自動有効化を選択できます。マルチアカウント環境の詳細については、「Amazon GuardDuty での複数のアカウントの管理」を参照してください。

委任 GuardDuty 管理者アカウントでの Lambda Protection の有効化

任意のアクセス方法を選択して、委任 GuardDuty 管理者アカウントに対して Lambda Network Activity Monitoring を有効または無効にします。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインの [設定] で、[Lambda Protection] を選択します。
3. [Lambda Protection] ページで、[編集] を選択します。
4. 次のいずれかを行います。

   [すべてのアカウントについて有効にする] の使用

   - [すべてのアカウントについて有効にする] を選択します。これにより、 AWS 組織に参加する新しいアカウントを含め、組織内のすべてのアクティブな GuardDuty アカウントの保護プランが有効になります。
   - [保存] を選択します。

   [アカウントを手動で設定] の使用

   - 委任 GuardDuty 管理者アカウントでのみ保護プランを有効にするには、[アカウントを手動で設定] を選択します。
   - [委任 GuardDuty 管理者 (このアカウント)] セクションで [有効にする] を選択します。
   - [保存] を選択します。

API/CLI

ユーザー独自のリージョンレベルのディテクター ID を使用し、`features` オブジェクト `name` を `LAMBDA_NETWORK_LOGS` として、`status` を `ENABLED` として渡して、updateDetector API オペレーションを実行します。

または、 AWS CLI を使用して Lambda Protection を有効にすることもできます。次のコマンドを実行し、*12abc34d567e8fa901bc2d34e56789f0* をアカウントのディテクター ID に、*us-east-1* を Lambda Protection を有効にするリージョンにそれぞれ置き換えます。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0 --
region us-east-1 --features '[{"Name": "LAMBDA_NETWORK_LOGS", "Status": "ENABLED"}]'
```

Lambda Network Activity Monitoring をすべてのメンバーアカウントで Lambda Network Activity Monitoring を自動で有効にする

任意のアクセス方法を選択して、すべてのメンバーアカウントのために Lambda ネットワークアクティビティモニタリング機能を有効にします。これには、既存のメンバーアカウントと、組織に参加する新しいアカウントが含まれます。

Console

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   必ず委任 GuardDuty 管理者アカウントの認証情報を使用してください。

2. 次のいずれかを行います。

   Protection Plans ページの使用

   1. ナビゲーションペインで、保護プランを選択します。
   2. すべての有効化の設定 を選択します。
   3. Lambda Protection で、すべてのアカウントの有効化を選択します。このアクションにより、組織内の既存のアカウントと新しいアカウントの両方について Lambda ネットワークアクティビティモニタリングが自動的に有効になります。
   4. すべて保存を選択し、確認して保存を選択します。

   > **Note**
   >
   > メンバーアカウントの設定を更新するには、最大 24 時間かかる場合があります。

   [アカウント] ページの使用

   1. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
   2. [アカウント] ページで、[招待によるアカウントの追加] の前に [自動有効化] の詳細設定を選択します。
   3. [自動有効化の詳細設定を管理] ウィンドウで、[Lambda ネットワークアクティビティモニタリング] の下の [すべてのアカウントについて有効にする] を選択します。

> **Note**
>
> デフォルトでは、この操作により [新規メンバーアカウントの GuardDuty を自動有効化] オプションが自動的に有効になります。

4. [保存] を選択します。

[すべてのアカウントについて有効にする] オプションを使用できない場合は、「メンバーアカウントで Lambda Network Activity Monitoring を選択的に有効または無効にする」を参照してください。

API/CLI

メンバーアカウントの Lambda Network Activity Monitoring を選択的に有効または無効にするには、独自の*###### ID* を使用して updateMemberDetectors API オペレーションを呼び出します。

または、 AWS CLI を使用して Lambda Protection を有効にすることもできます。次のコマンドを実行し、*12abc34d567e8fa901bc2d34e56789f0* をアカウントのディテクター ID に、*us-east-1* を Lambda Protection を有効にするリージョンにそれぞれ置き換えます。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0
 --account-ids 111122223333 --region us-east-1--features '[{"Name":
 "LAMBDA_NETWORK_LOGS", "Status": "ENABLED"}]'
```

スペースで区切られたアカウント ID のリストを渡すこともできます。

コードが正常に実行されると、`UnprocessedAccounts` の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

すべての既存のアクティブなメンバーアカウントのために Lambda ネットワークアクティビティモニタリングを有効にする

任意のアクセス方法を選択して、組織内のすべての既存のアクティブなメンバーアカウントの Lambda Network Activity Monitoring を有効にします。

Console

すべての既存のアクティブなメンバーアカウントのために Lambda ネットワークアクティビティモニタリングを設定するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   委任 GuardDuty 管理者アカウントの認証情報を使用してサインインします。
2. ナビゲーションペインで、保護プランを選択します。
3. すべての有効化の設定 を選択します。Lambda Protection では、設定の現在のステータスを表示できます。[アクティブなメンバーアカウント] セクションで、[アクション] を選択します。
4. [アクション] ドロップダウンメニューから、[すべての既存のアクティブなメンバーアカウントについて有効にする] を選択します。
5. [確認] を選択します。

API/CLI

メンバーアカウントの Lambda Network Activity Monitoring を選択的に有効または無効にするには、独自の*###### ID* を使用して updateMemberDetectors API オペレーションを呼び出します。

または、 AWS CLI を使用して Lambda Protection を有効にすることもできます。次のコマンドを実行し、*12abc34d567e8fa901bc2d34e56789f0* をアカウントのディテクター ID に、*us-east-1* を Lambda Protection を有効にするリージョンにそれぞれ置き換えます。

```
aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0
 --region us-east-1 --account-ids 111122223333 --features '[{"Name":
 "LAMBDA_NETWORK_LOGS", "Status": "ENABLED"}]'
```

スペースで区切られたアカウント ID のリストを渡すこともできます。

コードが正常に実行されると、UnprocessedAccounts の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

Lambda Network Activity Monitoring を新しいメンバーアカウントで Lambda Network Activity Monitoring を自動で有効にする

任意のアクセス方法を選択して、組織に参加する新しいアカウントの Lambda Network Activity Monitoring を有効にします。

Console

委任 GuardDuty 管理者アカウントは、[Lambda Protection] または [アカウント] ページのいずれかを使用して、組織内の新しいメンバーアカウントに対して Lambda Network Activity Monitoring を有効にすることができます。

Lambda Network Activity Monitoring を新しいメンバーアカウントで Lambda Network Activity Monitoring を自動で有効にするには

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   必ず委任 GuardDuty 管理者アカウントの認証情報を使用してください。
2. 次のいずれかを行います。
   - Protection Plans ページの使用:
     1. ナビゲーションペインで、保護プランを選択します。
     2. すべての有効化の設定 を選択します。
     3. [アカウントを手動で設定] を選択します。
     4. [新しいメンバーアカウントについて自動的に有効にする] を選択します。このステップにより、新しいアカウントが組織に参加するたびに、そのアカウントのために Lambda Protection が自動的に有効になります。この設定を変更できるのは、組織の委任 GuardDuty 管理者アカウントだけです。
     5. [保存] を選択します。
   - [Accounts] (アカウント) ページを使用する場合:
     1. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
     2. [アカウント] ページで、[自動有効化] 設定を選択します。

3. [自動有効化の詳細設定を管理] ウィンドウで、[Lambda ネットワークアクティビティモニタリング] の下の [新しいアカウントについて有効にする] を選択します。
4. [保存] を選択します。

API/CLI

新しいメンバーアカウントの Lambda Network Activity Monitoring を有効にするには、独自の*### ### ID* を使用して UpdateOrganizationConfiguration API オペレーションを呼び出します。

または、 AWS CLI を使用して Lambda Protection を有効にすることもできます。次の例では、単一のメンバーアカウントで Lambda Network Activity Monitoring を有効にする方法を示します。*12abc34d567e8fa901bc2d34e56789f0* をアカウントのディテクター ID に、*us-east-1* を Lambda Protection を有効にするリージョンにそれぞれ置き換えます。組織に参加する新規アカウントすべてに対して有効にしたくない場合は、`AutoEnable` を `NONE` に設定します。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-organization-configuration --detector-
id 12abc34d567e8fa901bc2d34e56789f0 --region us-east-1 --auto-enable --features
 '[{"Name": "LAMBDA_NETWORK_LOGS", "AutoEnable": "NEW"}]'
```

コードが正常に実行されると、`UnprocessedAccounts` の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

メンバーアカウントで Lambda Network Activity Monitoring を選択的に有効または無効にする

任意のアクセス方法を選択して、メンバーアカウントの Lambda Network Activity Monitoring を選択的に有効または無効にします。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   必ず委任 GuardDuty 管理者アカウントの認証情報を使用してください。

2. ナビゲーションペインで 設定の アカウントを選択します。

   [アカウント] ページで、[Lambda Network Activity Monitoring] 列を確認します。Lambda Network Activity Monitoring が有効になっているかどうかを示します。

3. Lambda Protection を設定するアカウントを選択します。一度に複数のアカウントを選択できます。
4. [保護プランの編集] ドロップダウンメニューから [Lambda Network Activity Monitoring] を選択し、適切なアクションを選択します。

API/CLI

独自の*###### ID* を使用して updateMemberDetectors API を呼び出します。

または、 AWS CLI を使用して Lambda Protection を有効にすることもできます。*12abc34d567e8fa901bc2d34e56789f0* をアカウントのディテクター ID に、*us-east-1* を Lambda Protection を有効にするリージョンにそれぞれ置き換えます。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-member-detectors --detector-id 12abc34d567e8fa901bc2d34e56789f0
 --region us-east-1 --account-ids 111122223333 --features '[{"Name":
 "LAMBDA_NETWORK_LOGS", "Status": "ENABLED"}]'
```

スペースで区切られたアカウント ID のリストを渡すこともできます。

コードが正常に実行されると、`UnprocessedAccounts` の空のリストが返されます。アカウントのディテクター設定を変更する際に問題が発生した場合は、そのアカウント ID と問題の概要が表示されます。

スタンドアロンアカウントに対する Lambda Protection の有効化

スタンドアロンアカウントは、特定の の で保護プランを有効または無効にする決定を所有 AWS アカウント します AWS リージョン。

アカウントが を通じて AWS Organizations、または招待方法によって GuardDuty 管理者アカウントに関連付けられている場合、このセクションはアカウントには適用されません。詳細については、「マルチアカウント環境での Lambda Protection の有効化」を参照してください。

Lambda Protection を有効にすると、アカウント内で Lambda Network Activity Monitoring のモニタリングが始まります。

スタンドアロンアカウント用に Lambda Protection を設定する際のアクセス方法を任意に選択します。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインの [設定] で、[Lambda Protection] を選択します。
3. [Lambda Protection] ページにアカウントの現在のステータスが表示されます。[有効化] を選択して、アカウントで Lambda Protection を有効にします。
4. [確認] を選択して、選択内容を保存します。

API/CLI

ユーザー独自のリージョンレベルのディテクター ID を使用し、`features` オブジェクト `name` を `LAMBDA_NETWORK_LOGS` として、`status` を `ENABLED` として渡して、updateDetector API オペレーションを実行します。

または、 AWS CLI を使用して Lambda Protection を有効にすることもできます。次のコマンドを実行し、*12abc34d567e8fa901bc2d34e56789f0* をアカウントのディテクター ID に、*us-east-1* を Lambda Protection を有効にするリージョンにそれぞれ置き換えます。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-detector --detector-id 12abc34d567e8fa901bc2d34e56789f0
 --region us-east-1 --features [{"Name" : "LAMBDA_NETWORK_LOGS", "Status" :
 "ENABLED"}]'
```

GuardDuty 調査 (プレビュー)

GuardDuty 調査は、GuardDuty の検出結果とアカウントの AI を活用したセキュリティ分析を提供します。調査を作成すると、GuardDuty はナレッジグラフを使用して、検出結果コンテキスト、過去 90 日間の関連アクティビティ、影響を受けたリソース、脅威インテリジェンス、脅威インジケータを調べます。各調査では、信頼スコア、MITRE ATT&CK® 手法分類、サポート証拠、実用的な推奨事項を含む脅威処理評価が提供されます。

各調査では、次のインサイトが生成されます。

- リスクレベル – 全体的なリスクの評価: Info、Low、Medium、High、Critical。
- 信頼度 – 評価の信頼度: Unknown、Low、Medium、High。
- 概要 – 調査結果と主要な観察結果の説明。
- 調査の詳細 – 調査に関連する追加情報とコンテキスト。
- 推奨されるアクション – 特定された問題に対処するために実行できる CLI コマンドを含む詳細なアクション。

> **Note**
>
> GuardDuty 調査は、米国東部 (バージニア北部)、米国東部 (オハイオ)、米国西部 (オレゴン)、カナダ (中部)、欧州 (フランクフルト)、欧州 (アイルランド)、欧州 (ロンドン)、欧州 (パリ)、欧州 (ストックホルム)、アジアパシフィック (東京) の 10 の商用 AWS リージョンでのみ利用できます。

分析タイプ

GuardDuty 調査では、次の 3 種類の分析がサポートされています。

- 検出結果の分析 – 検出結果 ID (32 文字の 16 進数) を指定すると、特定の GuardDuty の検出結果を分析します。プレビューでは、GuardDuty 調査はすべての拡張脅威検出 (XTD) 検出結果をサポートし、基盤、S3、ランタイムプランから検出結果を選択します。
- アカウント分析 – 12 桁の AWS アカウント ID を指定すると、 AWS アカウントの脅威体制を分析します。
- 組織分析 – 組織の脅威体制を分析します。

クロスリージョン推論

GuardDuty Investigation は Cross-Region Inference Service (CRIS) を活用します。CRIS は、調査分析を処理し、調査レポートを生成するために、地理的な AWS リージョン 最適な を自動的に選択します。これにより、利用可能なコンピューティングリソースとモデルの可用性を最大化し、最高のカスタマーエクスペリエンスを実現します。

データは、調査リクエストが発生したリージョンにのみ保存されます。ただし、調査データと概要結果は、そのリージョン外で処理される場合があります。すべてのデータは Amazon セキュアネットワーク経由で暗号化されて送信されます。

GuardDuty Investigation は、次の表に示すように、推論リクエストをリクエスト元の地理的領域内の利用可能なコンピューティングリソースに安全に転送します。

クロスリージョン推論ルーティング

サポートされている 地域	GuardDuty リージョン	推論リージョン
アメリカ	米国東部 (バージニア北部)	米国東部 (バージニア北部)、米国東部 (オハイオ)、米国西部 (オレゴン)
アメリカ	米国東部 (オハイオ)	米国東部 (バージニア北部)、米国東部 (オハイオ)、米国西部 (オレゴン)
アメリカ	米国西部 (オレゴン)	米国東部 (バージニア北部)、米国東部 (オハイオ)、米国西部 (オレゴン)
アメリカ	カナダ (中部)	カナダ (中部)、米国東部 (バージニア北部)、米国東部 (オハイオ)、米国西部 (オレゴン)
欧州	欧州 (フランクフルト)	欧州 (フランクフルト)、欧州 (ストックホルム)、欧州 (ミラノ)、欧州 (スペイン)、欧州 (アイルランド)、欧州 (パリ)
欧州	欧州 (アイルランド)	欧州 (フランクフルト)、欧州 (ストックホルム)、欧州 (ミラノ)、欧州 (スペイン)、欧州 (アイルランド)、欧州 (パリ)

サポートされている 地域	GuardDuty リージョン	推論リージョン
欧州	欧州 (ロンドン)	欧州 (フランクフルト)、欧州 (ストックホルム)、欧州 (ミラノ)、欧州 (スペイン)、欧州 (アイルランド)、欧州 (ロンドン)、欧州 (パリ)
欧州	欧州 (パリ)	欧州 (フランクフルト)、欧州 (ストックホルム)、欧州 (ミラノ)、欧州 (スペイン)、欧州 (アイルランド)、欧州 (パリ)
欧州	欧州 (ストックホルム)	欧州 (フランクフルト)、欧州 (ストックホルム)、欧州 (ミラノ)、欧州 (スペイン)、欧州 (アイルランド)、欧州 (パリ)
日本	アジアパシフィック (東京)	アジアパシフィック (東京)、アジアパシフィック (大阪)

前提条件

GuardDuty 調査を使用する前に、次の前提条件を満たしていることを確認してください。

- 調査 AWS リージョン を作成する には、アクティブな GuardDuty ディテクターが必要です。GuardDuty 有効化の詳細については、「GuardDuty の開始方法」を参照してください。
- ディテクターで GuardDuty 調査機能を有効にする必要があります。

  Console

  1. GuardDuty コンソールを開きます。
  2. ナビゲーションペインで [設定] を選択します。
  3. AI を活用した調査 - プレビューで、有効化を選択します。

  API/CLI

  UpdateDetector API を呼び出し、ディテクターでAI_ANALYSTこの機能を有効にします。

```
aws guardduty update-detector \
    --detector-id 2cb3d4e5f6a7b8c9d0e1f2a3b4c5d6e7 \
```

```
    --features '[{"Name": "AI_ANALYST", "Status": "ENABLED"}]'
```

- IAM ID には、調査アクションを実行するために必要なアクセス許可が必要です。次の IAM アクションが必要です。
  - guardduty:CreateInvestigation – 新しい調査を作成するために必要です。
  - guardduty:GetInvestigation – 調査結果を取得するために必要です。
  - guardduty:ListInvestigations – ディテクターの調査を一覧表示するために必要です。

次の IAM ポリシーの例では、すべての GuardDuty 調査アクションを使用するアクセス許可を付与します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "guardduty:CreateInvestigation",
                "guardduty:GetInvestigation",
                "guardduty:ListInvestigations"
            ],
            "Resource": "arn:aws:guardduty:us-
west-2:123456789012:detector/2cb3d4e5f6a7b8c9d0e1f2a3b4c5d6e7"
        }
    ]
}
```

管理者アカウントとメンバーアカウントのアクセスモデル

GuardDuty 調査には、管理者アカウントとメンバーアカウントのどちらを使用するかに応じて、次のアクセスルールが適用されます。

- 管理者アカウント – 自分自身とメンバーアカウントに関する調査を作成、取得、一覧表示できます。
- メンバーアカウント – 自分のアカウントの調査のみを取得して一覧表示できます。メンバーアカウントは調査を作成できず、他のアカウントまたは管理者アカウントに属する調査にアクセスできません。

調査の作成

調査を作成して、環境内の GuardDuty の検出結果とアカウントを AWS 分析できます。調査はバックグラウンドで非同期的に実行されます。調査を作成したら、調査 ID を使用してそのステータスを確認し、結果を取得します。

> **Important**
>
> プレビュー中、1 日 1 アカウントあたり最大 10 件の調査を開始し、合計 1 アカウントあたり 100 件の調査を開始できます。失敗した調査は、これらのクォータにはカウントされません。API/CLI を使用している場合、トリガープロンプトは最大 2,048 文字です。

任意のアクセス方法を選択して、調査を作成します。

Console

1. GuardDuty コンソールを開き、左側のナビゲーションで調査に移動します。
2. 調査の開始 を選択します。
3. 調査範囲を選択します。
   - 特定の検出結果 – 分析する検出結果 ID を入力します。
   - 自分のアカウント (スタンドアロンのみ) – 追加の入力は必要ありません。GuardDuty はアカウントを分析します。
   - 特定のアカウント (管理者のみ) – 12 桁の AWS アカウント ID を入力します。
   - 組織内のすべてのアカウント (管理者のみ) – 追加の入力は必要ありません。
4. 調査の開始 を選択して分析を開始します。

API/CLI

CreateInvestigation API オペレーションを実行して、新しい調査を開始します。ディテクター ID と、調査対象を説明するトリガープロンプトを指定する必要があります。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty create-investigation \
```

```
    --detector-id 2cb3d4e5f6a7b8c9d0e1f2a3b4c5d6e7 \
    --trigger-prompt "Investigate finding 1ab2c3d4e5f6a7b8c9d0e1f2a3b4c5d6 in
account 123456789012"
```

前述のコマンドで、*detector-id* を独自のディテクター ID に置き換え、*trigger-prompt* を調査対象の説明に置き換えます。

必要に応じて、べき等性の--client-tokenパラメータを含めることができます。同じクライアントトークンでリクエストを再試行すると、GuardDuty は重複を作成する代わりに既存の調査を返します。

出力の例:

```
{
    "InvestigationId": "a1b2c3d4-5678-90ab-cdef-ef1234567890"
}
```

プロンプト要件をトリガーする

トリガープロンプトは、調査対象を記述する必要があります。GuardDuty は、プロンプトの内容に基づいて分析タイプを決定します。

- 検出結果の分析 – プロンプトには、検出結果 ID (32 文字の 16 進文字列) を 1 つだけ含めます。検出結果は存在し、発信者のアカウントまたはメンバーアカウントに属している必要があります。1 つのプロンプトに複数の結果 IDs を含めることはできません。
- アカウント分析 – プロンプトには 12 桁の AWS アカウント ID を 1 つだけ含めます。発信者は、そのアカウントの管理者である必要があります。1 つのプロンプトに複数のアカウント IDs を含めることはできません。
- 組織分析 – プロンプトに組織全体のセキュリティ上の懸念を記述します。調査では、組織全体のシグナルを分析します。

GuardDuty は AI を使用して自由形式のプロンプトを解釈し、適切な分析範囲を決定します。結果 ID を含めると、結果分析が実行されます。アカウント ID を含めると、アカウント分析が実行されます。プロンプトが組織全体の懸念を説明する場合は、組織分析を実行します。プロンプトが特定の分析タイプと一致しない場合、調査はデフォルトで発信者自身のアカウントを分析します。

以下は、各分析タイプのトリガープロンプトの例です。

- 検出結果の分析 – `"Investigate finding 1ab2c3d4e5f6a7b8c9d0e1f2a3b4c5d6 in account 123456789012"`
- アカウント分析 – `"Analyze findings in account with id 123456789012"`
- 組織分析 – `"Analyze findings in my organization"`

メンバーアカウントの調査の作成

管理者アカウントの場合は、トリガープロンプトにメンバーアカウント ID を含めることで、メンバーアカウントの調査を作成できます。コマンドで管理者アカウントのディテクター ID を使用します。調査結果には、指定されたメンバーアカウントの結果が含まれます。

```
aws guardduty create-investigation \
    --detector-id 2cb3d4e5f6a7b8c9d0e1f2a3b4c5d6e7 \
    --trigger-prompt "Analyze findings in account with id 111122223333"
```

前述のコマンドで、 *detector-id* を管理者アカウントのディテクター ID に置き換えます。トリガープロンプトの 12 桁のアカウント ID は、調査するメンバーアカウントを識別します。

調査結果の表示

調査を作成したら、概要、調査の詳細、信頼度、推奨事項などの結果を取得できます。調査には、次のいずれかのステータスがあります。

- RUNNING – 調査はまだ進行中です。
- COMPLETED – 調査は正常に終了し、結果が利用可能になりました。
- FAILED – 調査でエラーが発生しました。詳細については、エラーフィールドを確認してください。

任意のアクセス方法を選択して、調査結果を表示します。

AI が生成した分析とレコメンデーションには、エラーや不完全な評価が含まれている場合があります。人間によるレビューが推奨されます。

Console

1. GuardDuty コンソールを開き、左側のナビゲーションで調査に移動します。

2. 調査テーブルで、レビューする完了した調査を見つけます。
3. 調査タイトルリンクを選択して詳細ページを開きます。

> **Note**
>
> 調査タイトルは、ステータスが完了している場合にのみクリックできます。

API/CLI

GetInvestigation API オペレーションを実行して、完了した調査の完全な詳細を取得します。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty get-investigation \
    --detector-id 2cb3d4e5f6a7b8c9d0e1f2a3b4c5d6e7 \
    --investigation-id a1b2c3d4-5678-90ab-cdef-ef1234567890
```

完了した調査の出力例:

```
{
    "Investigation": {
        "InvestigationId": "a1b2c3d4-5678-90ab-cdef-ef1234567890",
        "Status": "COMPLETED",
        "TriggerPrompt": "Investigate finding 1ab2c3d4e5f6a7b8c9d0e1f2a3b4c5d6 in
 account 123456789012",
        "TriggeredBy": "123456789012",
        "RiskLevel": "Critical",
        "Risk": "Detection logic is valid but no live resources are compromised.",
        "Confidence": "High",
        "Summary": "{\"keyObservations\":{\"title\":\"...\",\"narrative\":\"...\",
\"observations\":[...]},\"countermeasures\":[...],\"threatAssessment\":{...}}",
        "Cloud": {
            "Provider": "AWS",
            "Region": "us-east-1",
            "Account": "123456789012"
        },
        "Metadata": {
            "Product": {
```

```
                "Name": "Amazon GuardDuty AI Analyst",
                "Feature": "Investigation"
            },
            "Version": "1.0.0"
        },
        "StartTime": 1705319400.0,
        "EndTime": 1705319700.0
    }
}
```


Summary フィールドには、主要な観測値、CLI コマンドによる対策、MITRE ATT&CK® 脅威評価など、完全な調査結果を含む JSON 文字列が含まれています。

調査結果の解釈

次の表は、調査が返すことができるリスクレベルを示しています。

調査リスクレベル

リスクレベル	説明
Info	環境に対するリスクが特定されていない情報検出結果。
低	即時のアクションが必要になる可能性が低い軽微なリスク。
中	レビューする必要がある中程度のリスクがあり、修復が必要になる場合があります。
高	迅速な調査と修復を必要とする重大なリスク。
重大	さらなる侵害を防ぐために即時のアクションを必要とする重大なリスク。

次の表に、信頼レベルを示します。

調査の信頼度

信頼度	説明
不明	評価の信頼度を判断するためのデータが不十分です。

信頼度	説明
低	限られた証拠が評価をサポートします。
中	中程度の証拠は評価をサポートします。
高	強力な証拠は評価をサポートします。

調査の一覧表示

ディテクターのすべての調査を、オプションのソートとページ分割を使用して一覧表示できます。これにより、複数の調査のステータスを確認および追跡できます。

任意のアクセス方法を選択して、調査を一覧表示します。

Console

1. GuardDuty コンソールを開き、左側のナビゲーションで調査に移動します。
2. 調査テーブルには、現在のディテクターのすべての調査とそのステータス、リスクレベル、タイムスタンプが表示されます。

API/CLI

ListInvestigations API オペレーションを実行して、ディテクターの調査概要を一覧表示します。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty list-investigations \
    --detector-id 2cb3d4e5f6a7b8c9d0e1f2a3b4c5d6e7 \
    --max-results 10
```

`--sort-criteria` パラメータを指定して結果をソートできます。次の の例では、開始時刻でソートされた調査を降順で一覧表示します。

```
aws guardduty list-investigations \
    --detector-id 2cb3d4e5f6a7b8c9d0e1f2a3b4c5d6e7 \
```

```
    --sort-criteria '{"attributeName": "START_TIME", "orderBy": "DESC"}' \
    --max-results 10
```

使用可能なソート属性は、START_TIME、、END_TIMESTATUS、RISK_LEVEL、および ですCONFIDENCE。(昇順) または ASC (DESC降順) でソートできます。

出力の例:

```
{
    "Investigations": [
        {
            "InvestigationId": "a1b2c3d4-5678-90ab-cdef-ef1234567890",
            "Status": "COMPLETED",
            "TriggerPrompt": "Investigate finding 1ab2c3d4e5f6a7b8c9d0e1f2a3b4c5d6
 in account 123456789012",
            "RiskLevel": "Critical",
            "Confidence": "High",
            "StartTime": 1705319400.0,
            "EndTime": 1705319700.0,
            "AccountId": "123456789012"
        },
        {
            "InvestigationId": "b2c3d4e5-6789-01bc-def0-ef2345678901",
            "Status": "COMPLETED",
            "TriggerPrompt": "Analyze findings in account with id 123456789012",
            "RiskLevel": "High",
            "Confidence": "High",
            "StartTime": 1705315800.0,
            "EndTime": 1705316100.0,
            "AccountId": "123456789012"
        },
        {
            "InvestigationId": "c3d4e5f6-7890-12cd-ef01-ef3456789012",
            "Status": "COMPLETED",
            "TriggerPrompt": "Analyze findings in my organization",
            "RiskLevel": "Medium",
            "Confidence": "Medium",
            "StartTime": 1705312200.0,
            "EndTime": 1705312500.0,
            "AccountId": "123456789012"
        }
    ]
}
```

レスポンスにNextToken値が含まれている場合は、後続のリクエストでそれを渡して、結果の次のページを取得します。1 ページあたり最大 50 件の結果を取得できます。

GuardDuty のカスタム検出ルール

カスタム検出ルールは、 AWS 環境で発生することが想定されていないアクティビティを検出する構築済みのルールのライブラリを提供します。GuardDuty は、これらのルールを構築および維持し、MITRE ATT&CK® や Threat Technique Catalog などのカタログから抽出された確立された AWS 脅威手法語彙を使用してそれぞれを記述します。検出するアクティビティの各ルールを、ドライランモードで最初に測定するか、ライブモードで有効にして検出結果を生成します。

セキュリティチームは、特定のアカウントで特定のアクションが発生しないことをよく知っています。AMI の共有、VPC フローログの削除、MFA なしでのサインインなどのアクションは、あるアカウントではルーチンであり、別のアカウントでは侵害の兆候である可能性があります。ほとんどの GuardDuty の検出結果は、あらゆる環境で疑わしいアクティビティを識別します。カスタム検出ルールを使用すると、予期しないアカウントでのみ疑わしいアクティビティを検出できます。

カスタム検出ルールでは、次の概念を使用します。

ルール

GuardDuty が所有および維持する事前構築済みの検出。各ルールは 1 つの脅威手法をターゲットとし、それに一致する原因となる条件を定義します。

関連付け

ルールとアカウント間のリンク。関連付けは、そのアカウントに対してルールが実行されるモードを記録します。関連付けのないルールは評価されません。

モード

関連付けられたルールがアクティビティと一致するときの動作。すべてのルールは 2 つのモードをサポートしています。

- ライブ – ルールがアクティビティに一致すると、GuardDuty は結果を生成します。ライブモードの有効期限はありません。
- ドライラン – GuardDuty はルールを評価し、Amazon CloudWatch メトリクスを出力しますが、検出結果は生成しません。ライブ検出にコミットする前に、ドライランを使用してシグナルボリュームを測定します。GuardDuty は、ルールがイベントに一致する場合にのみこれらのメトリクスを発行するため、一致しないルールはドライランメトリクスを生成しません。ドライランは 14 日後に自動的に期限切れになります。

組織設定

委任 GuardDuty 管理者アカウントが組織のメンバーアカウントに適用するルール設定。GuardDuty は、後で参加するアカウントを含め、各ターゲットアカウントに個々の関連付けを作成します。詳細については、「マルチアカウント環境でのカスタム検出ルールの管理」を参照してください。

Signal

ルールが検出したアクティビティの 1 つの観測された出現。GuardDuty は、関連するシグナルを 1 つの検出結果に集約します。詳細については、「結果の集約」を参照してください。

トピック

- カスタム検出ルールの仕組み
- 使用可能なカスタム検出ルール
- カスタム検出ルールの管理
- マルチアカウント環境でのカスタム検出ルールの管理

カスタム検出ルールの仕組み

まず、GuardDuty が提供するルールを参照して、環境に関連するアクティビティを検出するルールを特定します。次に、各ルールを、アクティビティが予期しないアカウントと関連付けます。ルールと アカウントの関連付けの詳細については、「」を参照してくださいカスタム検出ルールの管理。

データソースとシグナル

ルールを関連付けると、GuardDuty はルールが依存するデータソースをモニタリングし、一致するイベントについて受信ログを評価します。カスタム検出ルール は、 AWS CloudTrail 管理イベントとランタイムモニタリングイベントをサポートしています。観察されたアクティビティがルールと一致すると、GuardDuty はシグナルを生成します。シグナルは、ルールが検出したアクティビティの単一の観測された出現を表します。

各データソースには独自の要件があります。

AWS CloudTrail 管理イベント

GuardDuty は、GuardDuty が有効になっているすべてのアカウントでこれらのイベントを分析します。追加のセットアップは必要ありません。

Runtime Monitoring イベント

Runtime Monitoring ルールは、GuardDuty セキュリティエージェントがリソースで観察するプロセス、ファイル、ネットワークアクティビティを評価します。Runtime Monitoring ルールは、次の両方の要件を満たすリソースでのみアクティビティを照合できます。

- Runtime Monitoring は、Fargate のリソースタイプ Amazon EKS、Amazon EC2、または Amazon ECS で有効になっています。詳細については、「GuardDuty Runtime Monitoring」を参照してください。
- GuardDuty セキュリティエージェントは、そのプラットフォームのルールの最小エージェントバージョン以上のリソースで実行されます。

各 Runtime Monitoring ルールには、実行できる各プラットフォームで必要な最小エージェントバージョンが記載されています。これらの要件を表示するには、 GetCustomDetectionRuleオペレーションを呼び出します。

リソースのエージェントがルールの最小バージョンよりも古い場合、ルールはそのリソースのアクティビティと一致しません。この場合、GuardDuty はエラーを報告しないため、Runtime Monitoring ルールを有効にするリソースでエージェントを最新の状態に保ちます。エージェントのバージョンについては、「」を参照してくださいGuardDuty セキュリティエージェントのリリースバージョン。

モード

すべての関連付けは、次の 2 つのモードのいずれかで実行されます。

- ライブ – ルールが一致すると、GuardDuty は結果を生成します。
- ドライラン – GuardDuty は、ルールが一致するが結果を生成しない頻度をカウントする Amazon CloudWatch メトリクスを出力します。ドライランを使用して、ルールが結果を作成したり自動レスポンスをトリガーしたりする前に、ルールが一致する頻度を測定します。ドライランの関連付けは作成から 14 日後に期限切れになり、ライブの関連付けは期限切れになりません。

詳細については、「ドライラン」および「ライブ」を参照してください。

結果の集約

各検出結果はTactic:Service/Technique、 Persistence:IAM/AccountManipulationまたは などの形式に従いますPersistence:Runtime/SuspiciousCommand。GuardDuty がシグナル

を結果にグループ化する方法は、次のリストで説明するように、シグナルを生成したデータソースによって異なります。

- AWS CloudTrail ルール – GuardDuty は、同じ戦術、サービス、手法を共有するシグナルを 1 つの検出結果に集約します。GuardDuty が同じプリンシパルから同じアクティビティを繰り返し観察する場合、1 つの検出結果に複数のシグナルを含めることができます。
- Runtime Monitoring ルール – GuardDuty は、同じルール、リソース、およびプロセスの詳細を共有するシグナルを 1 つの検出結果に集約します。各検出結果は、インスタンス、タスク、コンテナなど、1 つのリソースを対象としています。GuardDuty が比較するプロセスの詳細は、検出結果タイプによって異なります。たとえば、ファイル変更の検出結果は、ファイルオペレーションとプロセスを実行したユーザーも比較します。再一致は、新しい結果を作成する代わりに、既存の結果を更新します。ルールが同じリソースとプロセスに対して同じ検出結果タイプを生成しても、異なるルールの検出結果は分離されたままになります。

検出結果の構造とフィールドの詳細については、「」を参照してくださいカスタム検出ルールの検出結果の詳細。

編集されたイベントフィールド

AWS CloudTrail ルールは、 がイベントに記録するフィールドを評価します。 AWS CloudTrail 編集は GuardDuty のアップストリームで行われます。 AWS サービスは、発行するイベントのフィールドの値を編集できます。この秘匿化は、潜在的な機密データを保護します。フィールドが編集されると、 はフィールドが存在した AWS CloudTrail が、その値は記録しません。

イベントを発行するサービスは、編集するフィールドを決定します。サービスでは、そのフィールドが機密情報を伝達できる場合にフィールドを編集します。編集は、特定の値ではなく フィールドに適用されます。したがって、特定のリクエストの値が機密でなくても、サービスはフィールドを編集します。、 CLI、または AWS SDK AWS のどちらを使用するかに関係なく、すべてのリクエスト AWS マネジメントコンソールのこれらのフィールドを編集します。

GuardDuty は、イベントにない値を評価できません。ルールの検出ロジックが秘匿化されたフィールドに依存している場合、ルールは一致しません。その後、GuardDuty は、アクティビティが発生してもシグナルも検出結果も生成しません。イベント名、サービス、呼び出し元の ID など、サービスが編集しないフィールドのみを検査する場合、 AWS ルールは影響を受けません。

秘匿化では、ルールを 1 つのリクエストに一致させることもできますが、同じ効果を持つ別のリクエストに一致させることはできません。API オペレーションが同じ変更に対して複数のフィールドを

受け入れる場合、1 つのフィールドが秘匿化され、もう 1 つのフィールドが秘匿化されないことがあります。ルールが一致するかどうかは、リクエストが使用したフィールドによって異なります。

> **Note**
>
> カスタム検出ルールは検出機能であり、完全な監査レコードではありません。アカウントで特定の API コールが発生したかどうかを確認するには、 でイベントを確認します AWS CloudTrail。詳細については、「イベント履歴を含む AWS CloudTrail イベントの表示」を参照してください。

使用可能なカスタム検出ルール

カスタム検出ルールは、GuardDuty が提供する脅威検出を拡張します。GuardDuty には、環境に固有のリスクに基づいて有効にすることを選択できるオプトインルールのライブラリが用意されています。使用可能なルールの現在のリストを取得するには、 ListCustomDetectionRules API オペレーションを使用します。各ルールが生成する検出結果タイプについては、「」を参照してくださいカスタム検出ルールの検出結果タイプ。

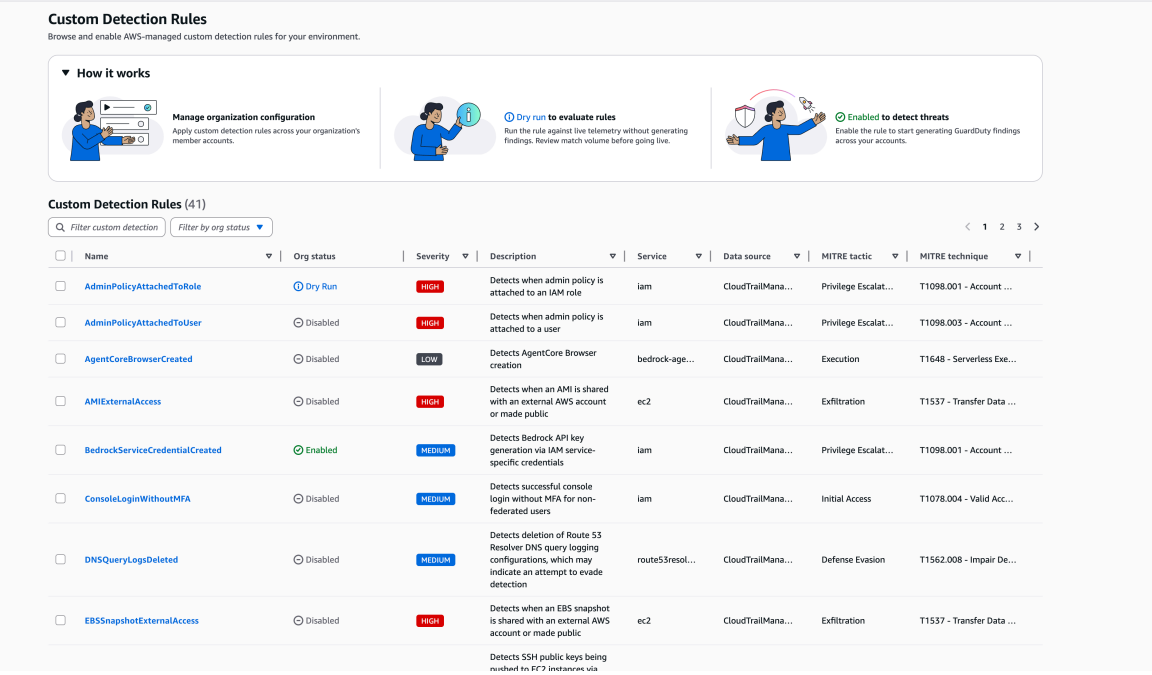
> **Note**
>
> 個々のルールの可用性は、各リージョンでの対応する AWS サービスと機能の可用性によって異なります。例えば、 AWS Organizations または Amazon Simple Email Service をターゲットとするルールは、それらのサービスが動作するリージョンでのみ使用できます。同様に、Lambda 関数 URLs または SageMaker AI ノートブックインスタンスをターゲットとするルールは、これらの機能がサポートされているリージョンでのみ使用できます。

ルール検出ロジックの表示

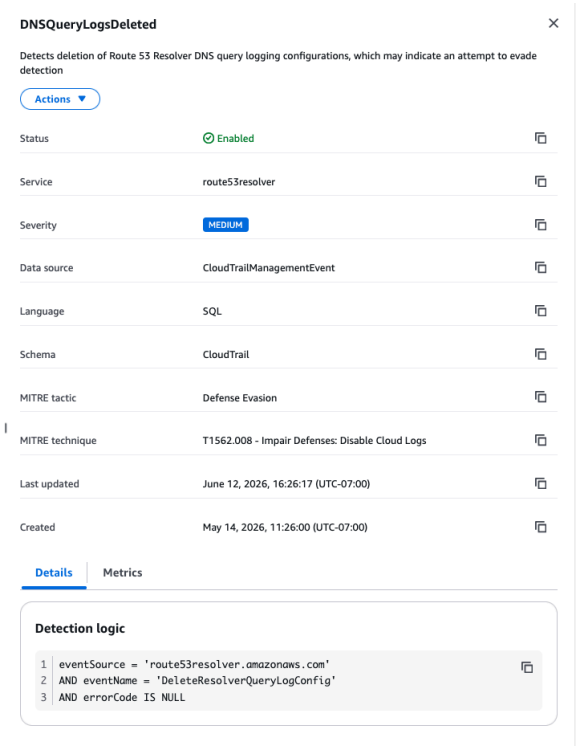
各ルールは、発生時期を決定する特定の条件を定義します。検出ロジックで任意のルールを検査できます。

- コンソール – GuardDuty コンソールで、カスタム検出ルールに移動し、ルール名を選択します。分割パネルには、ルールの検出ロジックが詳細タブに表示されます。
- API – GetCustomDetectionRuleオペレーションを呼び出します。レスポンスには、`Definition.Expression`フィールドにルールの式が含まれます。

次の図は、カスタム検出ルールカタログを示しています。



次の図は、検出ロジックを含むルールの詳細パネルを示しています。



検出ロジックで使用される関数

カスタム検出ルールは SQL 条件を使用します。 AWS CloudTrail ルールは AWS CloudTrail イベントフィールドを評価し、一部の は特殊な関数を使用して、 `requestParameters`や などのイベントフィールド内のネストされた JSON 構造を検査します`responseElements`。Runtime Monitoring ルールはランタイムイベントフィールドを評価し、多くは正規表現関数を使用して、主にプロセスコマンドラインを照合します。次の表は、ルール検出ロジックに表示される可能性のある関数を示しています。

ネストされたフィールドを検査するルールは、そのフィールドがイベントに存在するかどうかによって異なります。が記録 AWS CloudTrail しないフィールドの詳細については、「」を参照してください編集されたイベントフィールド。

関数	戻り値	説明
`get_json_object(json, path)`	文字列	ドット区切りパス (など) を使用して JSON ドキュメントから値を抽出します`$.requestParameters.bucketName`。値を文字列として返します。
`json_collect_values(json, path, field)`	配列	JSON ドキュメント内の配列に移動し、各要素から指定されたフィールドを収集します。この関数は、IAM ポリシーステートメントやセキュリティグループルールなどの繰り返し構造から値のリストを抽出します。
`get_aws_principal(policy_json, effect)`	配列	IAM、Amazon S3、または Lambda リソースポリシーを解析し、指定された効果 (許可または拒否) に一致するステートメントから AWS アカウントプリンシパルを返します。この関数は、リソースポリシーのクロスアカウントアクセスを検出します。
`policy_has_public_principal(policy_json, effect)`	ブール値	指定された効果を持つステートメントがパブリックプリンシパル () にアクセスを許可する場合に true を返します`"*"`。この関数は、パブリックアクセスを許可するポリシーを検出します。

関数	戻り値	説明
`RE2_FIND(input, pattern)`	ブール値	RE2 正規表現が のいずれかの部分`pattern`と一致する場合、true を返します`input`。一致では大文字と小文字が区別されます。`input` が null の場合、false を返します。Runtime Monitoring ルールは、主にこの関数を使用してプロセスのコマンドラインを照合します。
`IRE2_FIND(input, pattern)`	ブール値	と同じですが`RE2_FIND`、一致では大文字と小文字が区別されません。

GuardDuty セキュリティエージェントは、null 文字で区切られたプロセスのコマンドライン引数を記録します。コマンドラインに一致するほとんどの Runtime Monitoring ルールは、まずこれらの区切り文字をスペースに置き換えます。例: `RE2_FIND(REPLACE(process.cmd_line, '\u0000', ' '), pattern)`。

Runtime Monitoring ルールの検出ロジック SQL は、GuardDuty が正確なロジックではなく、アクティビティを検出する方法の近似値です。

カスタム検出ルールの管理

GuardDuty は、関連付けを通じてアカウントにルールを適用します。関連付けは、ルールをアカウントにリンクし、ルールが動作するモード、ライブまたはドライランを記録します。

組織では、委任 GuardDuty 管理者アカウントは組織設定を使用してメンバーアカウントのルールを一元管理します。組織設定では、ルールが適用されるアカウントを制御するための包含リストと除外リストがサポートされています。詳細については、「マルチアカウント環境でのカスタム検出ルールの管理」を参照してください。

使用可能なルールの一覧表示

ルールカタログには、使用可能なすべてのルールが一覧表示され、名前、重要度、データソース、MITRE ATT&CK® 戦術、手法、 AWS サービスがフィルターされます。

Console

使用可能なルールを一覧表示するには

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、カスタム検出ルールを選択します。
3. フィルターコントロールを使用して、プロパティ (重要度や戦術など) でリストを絞り込みます。

API/CLI

ListCustomDetectionRules オペレーションを実行します。フィルターを渡して、名前、重要度、データソース、戦術、手法、またはサービスで結果を絞り込むことができます。

```
aws guardduty list-custom-detection-rules \
    --filters Name=severity,Values=HIGH
```

ルールのステータスの表示

アカウントの各ルールには、次の 3 つのステータスのいずれかがあります。

- ライブ – ルールはライブモードで関連付けられ、結果を生成します。
- ドライラン – ルールはドライランモードで関連付けられ、Amazon CloudWatch メトリクスのみを出力します。
- 無効 – ルールには関連付けがなく、評価されません。

Console

カスタム検出ルールページの Status 列には、各ルールの現在のステータスが表示されます。ドロップダウンを使用してステータスでフィルタリングできます。

API/CLI

ListCustomDetectionRuleAssociations オペレーションを実行して、アカウントに関連付けられているルールとその現在のモードを確認します。

```
aws guardduty list-custom-detection-rule-associations
```

一括操作

一括オペレーションは、同じモード変更を複数のルールに適用するか、複数のアカウントにまたがる複数のルールに適用します。

Console

カスタム検出ルールページで、チェックボックスを使用して複数のルールを選択し、アクションを選択して、選択したすべてのルールを 1 つのアクションで有効または無効にします。

API/CLI

同じ変更を複数のルールに適用するには、ルールごとに を 1 CreateCustomDetectionRuleAssociation回呼び出します。

```
for RULE in rule-id-1 rule-id-2; do
    aws guardduty create-custom-detection-rule-association \
        --rule-id "$RULE" \
        --mode LIVE
done
```

名前付きリストではなくカタログ内のすべてのルールをカバーするには、ルール IDs を の出力に置き換えますListCustomDetectionRules。 `$(aws guardduty list-custom-detection-rules --query 'Rules[].RuleId' --output text)`

1 つのルールを多くのアカウントに適用するには、アカウントごとの呼び出しではなく、組織設定を使用します。組織設定は、包含リストと除外リストを持つメンバーアカウントをターゲットとし、GuardDuty は個々の関連付けを作成します。詳細については、「マルチアカウント環境でのカスタム検出ルールの管理」を参照してください。

ドライラン

ドライランモードでは、GuardDuty はルールに対してイベントを評価しますが、結果を生成しません。代わりに、GuardDuty はライブ検出を有効にする前に、シグナルボリュームとルールの動作を評価するために使用できる Amazon CloudWatch メトリクスを出力します。GuardDuty は、ルールがイベントに一致する場合にのみこれらのメトリクスを発行します。ルールが一致しない場合、ドライランメトリクスは生成されません。

ドライランモードを使用すると、自動レスポンスをトリガーしたり、コンソールで結果を生成したりすることなく、環境でルールがどのように動作するかを理解できます。

> **Note**
>
> ドライランの関連付けは、作成されてから 14 日後に期限切れになります。有効期限が切れると、GuardDuty はそのアカウントのルールの評価を停止し、ルールは無効に戻ります。ルールを再度評価するには、新しいドライランの関連付けを作成します。ライブ関連付けは期限切れになりません。

Console

ドライランモードでルールを有効にするには

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、カスタム検出ルールを選択します。
3. 1 つ以上のルールを選択します。
4. Actions を選択し、Enable (Dry Run) を選択します。

API/CLI

を mode に設定して CreateCustomDetectionRuleAssociation オペレーションを実行しますDRY_RUN。

```
aws guardduty create-custom-detection-rule-association \
    --rule-id rule-id \
    --mode DRY_RUN
```

ライブ

ライブモードでは、GuardDuty は GuardDuty コンソールに表示される検出結果を生成します。結果は Amazon EventBridge にエクスポートされ、設定する場合は、所有する Amazon S3 バケットにエクスポートされます。結果は Security Hub などの AWS 統合 AWS サービスにも送信されます。ライブモードを使用して、環境でアクティブな脅威を検出します。

Console

ライブモードでルールを有効にするには

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

2. ナビゲーションペインで、カスタム検出ルールを選択します。
3. 1 つ以上のルールを選択します。
4. アクションを選択し、有効化 (ライブ) を選択します。

既存のルールをドライランからライブに切り替えるには、ルールを選択し、アクションを選択し、有効化 (ライブ) を選択します。

API/CLI

ライブモードで新しいルールを有効にするには、 を modeに設定して CreateCustomDetectionRuleAssociationオペレーションを実行しますLIVE。

```
aws guardduty create-custom-detection-rule-association \
    --rule-id rule-id \
    --mode LIVE
```

既存のルールをドライランからライブに切り替えるには、UpdateCustomDetectionRuleAssociationオペレーションを実行します。

ルールの無効化

ルールを無効にすると、関連付けが削除され、GuardDuty がアカウントのそのルールに対するイベントの評価を停止します。

Console

カスタム検出ルールを無効にするには

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、カスタム検出ルールを選択します。
3. 無効にするルールを 1 つ以上選択します。
4. Actions を選択し、Disable を選択します。
5. プロンプトが表示されたら、アクションを確認します。

API/CLI

DeleteCustomDetectionRuleAssociation オペレーションを実行します。

```
aws guardduty delete-custom-detection-rule-association \
    --rule-id rule-id
```

マルチアカウント環境でのカスタム検出ルールの管理

組織では、委任 GuardDuty 管理者アカウントは、組織設定を使用してメンバーアカウントのカスタム検出ルールを一元的に管理します。管理者は各ルールのインテントを宣言し、GuardDuty は後で参加するアカウントを含むメンバーアカウントに自動的に適用します。

> **Note**
>
> カスタム検出ルールのマルチアカウント管理は、 を通じて管理されるアカウントでのみサポートされます AWS Organizations。招待ベースのメンバーアカウントは、一元化されたルール管理ではサポートされていません。

組織設定

メンバーアカウントは、それ自体のカスタム検出ルールを有効、無効に、または変更することはできません。メンバーアカウントから単一アカウントの関連付けオペレーションを呼び出そうとすると、GuardDuty は`AccessDeniedException`エラーを返します。管理者は、メンバーに代わってルールを管理します。委任管理者アカウントとメンバーアカウントの詳細については、GuardDuty での複数のアカウントの管理」を参照してください。

組織設定は、単一のルールをメンバーアカウントのセットにリンクします。ルールがそれらのアカウントに対して動作するモード、ライブまたはドライランを記録します。次に、GuardDuty はターゲットのメンバーアカウントごとにルールの個別の関連付けを作成します。包含リストと除外リストを使用して、すべてのメンバーまたは特定のアカウントのルールを設定できます。

Console

組織のカスタム検出ルールを設定するには

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、カスタム検出ルールを選択します。
3. ルール名を選択してルールの詳細を開き、アカウントタブを選択します。
4. Organization Configuration の横にある Configure を選択します。

5. モード (ライブ実行またはドライ実行) を選択し、含めるか除外するアカウントを選択します。

API/CLI

組織設定を作成する

CreateCustomDetectionRuleOrgConfiguration を実行して、メンバーアカウントのルール設定を作成します。

組織設定を更新する

を実行してUpdateCustomDetectionRuleOrgConfiguration、モードまたはアカウントの範囲を変更します。

組織設定を削除する

DeleteCustomDetectionRuleOrgConfiguration を実行して、組織設定を削除します。ルールはモードごとに 1 つの設定を保持できるため、 は削除する設定modeを選択します。もう 1 つの設定はそのまま残ります。

> **Note**
>
> 組織設定を削除しても、既存のメンバーアカウントの関連付けはカスケード削除されません。ルールは、GuardDuty がすでに適用したメンバーアカウントで有効のままになります。

> **Note**
>
> ドライランの組織設定は、作成してから 14 日後に期限切れになります。その設定から GuardDuty が作成したメンバーアカウントの関連付けは、各関連付けが作成されてから 14 日ではなく、その設定で期限切れになります。

同じルールで両方のモードを実行する

ルールはモードごとに最大 1 つの組織設定を保持するため、ルールは次の条件でのみライブモードとドライランモードで同時に実行できます。両方の設定でターゲットアカウントに明示的なインクルードリストを指定する必要があります。

組織全体がルールのワンショットであると主張する設定。設定は、アカウントの名前をまったく付けない場合、または除外するアカウントに名前を付ける場合に、組織全体をクレームします。他のモードの設定は共存できず、作成しようとすると が返されますConflictException。組織全体の設定を別のモードに移動するには、2 番目の設定を作成するのではなく、既存の設定を更新します。

例: すべてのメンバーアカウントにドライランモードでルールを適用する

次の AWS CLI コマンドは、後で参加するアカウントを含め、すべてのメンバーアカウントを対象とする組織設定を作成します。--include-account-ids と の両方を省略すると、組織全体が--exclude-account-idsターゲットになります。

```
aws guardduty create-custom-detection-rule-org-configuration \
    --rule-id "EXAMPLE_RULE_ID" \
    --mode DRY_RUN
```

ルールをライブ検出に移動するには、 を使用して既存の設定を更新しますUpdateCustomDetectionRuleOrgConfiguration。

組織全体にすべてのルールを適用する

コンソールは、一度に 1 つのルールを設定します。組織全体で使用可能なすべてのルールに同じモードを適用するには、 CLI を使用してルールカタログをループオーバーします AWS 。--include-account-ids と の両方を省略すると、後で参加するアカウントを含め、すべてのメンバーアカウントが--exclude-account-idsターゲットになります。

```
for RULE in $(aws guardduty list-custom-detection-rules \
    --query 'Rules[].RuleId' --output text); do
    aws guardduty create-custom-detection-rule-org-configuration \
        --rule-id "$RULE" \
        --mode DRY_RUN
done
```

このループは を使用して--mode DRY_RUN、ルールが結果を生成する前にシグナルボリュームを測定できるようにします。ループはアカウントに名前を付けないため、各設定は組織全体を主張し、そ

のルールのワンショットです。準備ができたら、既存の設定を で更新して、各ルールをライブ検出に移動しますUpdateCustomDetectionRuleOrgConfiguration。でループを再度実行すると、組織全体のドライラン設定が既に存在するためConflictException、 --mode LIVEが返されます。

Amazon GuardDuty での複数のアカウントの管理

AWS 環境に複数のアカウントがある場合は、管理者アカウント AWS アカウント として指定することで管理できます。その後、複数の をメンバーアカウントとして AWS アカウント この管理者アカウントに関連付けることができます。この設定では、指定された GuardDuty 管理者アカウントが組織全体のセキュリティを評価およびモニタリングできます。管理者アカウントは、生成されたすべての検出結果を確認したり、GuardDuty 内の保護プランを設定したりといったアカウント管理タスクを実行することもできます。

GuardDuty では、組織は、委任 GuardDuty 管理者アカウントと 1 つ以上の関連付けられたメンバーアカウントで構成されています。アカウントを関連付けるには、 と統合するか AWS Organizations、GuardDuty コンソールでメンバーシップの招待を送受信する従来の方法を使用します。GuardDuty では、 と統合することをお勧めします AWS Organizations。

> **Note**
>
> アカウントページの自動有効化セクションの編集ボタンは、レガシー設定モーダルを開くのではなく保護プランの設定、ページにリダイレクトします。

AWS Organizations は、 AWS 管理者が複数の を統合して一元管理できるようにするグローバルアカウント管理サービスです AWS アカウント。予算、セキュリティ、コンプライアンスのニーズをサポートするように設計されたアカウント管理および一括請求 (コンソリデーティッドビリング) 機能が備わっています。追加料金なしで提供され、Macie AWS のサービス、、Amazon GuardDuty などの複数の と統合 AWS Security Hub CSPMされます。詳細については、「AWS Organizations ユーザーガイド」を参照してください。

内容

- GuardDuty 管理者アカウントとメンバーアカウントの関係について理解する
- を使用した GuardDuty アカウントの管理 AWS Organizations
- 組織ポリシーを使用したアカウントの管理
- 招待による GuardDuty アカウントの管理
- メンバーアカウントの詳細を CSV 形式でエクスポートする際の GuardDuty に関する考慮事項

GuardDuty 管理者アカウントとメンバーアカウントの関係について理解する

複数アカウント環境で GuardDuty を使用すると、管理者アカウントはメンバーアカウントに代わって GuardDuty の特定の側面を管理できます。管理者アカウントは次の主な機能を実行できます。

- 関連付けられたメンバーアカウントを追加および削除する – 管理者アカウントがこれを行うプロセスは、 AWS Organizations または GuardDuty 招待方法によるアカウントの管理方法によって異なります。

  GuardDuty では、 を通じてメンバーアカウントを管理することをお勧めします AWS Organizations。
- 管理アカウントで GuardDuty を有効にする委任 GuardDuty 管理者アカウント – AWS Organizations 管理アカウントが GuardDuty を無効にすると、委任 GuardDuty 管理者アカウントは管理アカウントで GuardDuty を有効にできます。ただし、管理アカウントが GuardDuty のためのサービスにリンクされたロールの許可を明示的に削除していない必要があります。
- メンバーアカウントのステータスを設定する – 管理者アカウントは、GuardDuty 保護プランのステータスを有効または無効にでき、関連付けられたメンバーアカウントに代わって GuardDuty のステータスを有効、停止、または無効にできます。

  で管理される委任 GuardDuty 管理者アカウントは、 AWS アカウント がメンバーとして追加されると、GuardDuty を自動的に有効に AWS Organizations できます。
- 検出結果を生成するタイミングをカスタマイズする – 管理者アカウントは、抑制ルール、信頼されている IP リスト、脅威リストを作成および管理することで、GuardDuty ネットワーク内の検出結果をカスタマイズできます。マルチアカウント環境では、これらの機能の設定がサポートされるのは、委任 GuardDuty 管理者アカウントのみです。メンバーアカウントはこの設定を更新できません。

次の表では、GuardDuty の管理者アカウントとメンバーアカウントの関係を示しています。

表の説明

- 自分 - アカウントは、自分のアカウントに対してのみ、リストされたアクションを実行できます。
- 任意 – アカウントは、関連付けられた任意のアカウントに対して、リストされたアクションを実行できます。

- すべて – アカウントは、リストされたアクションを実行でき、そのアクションは関連付けられたすべてのアカウントに適用されます。通常、このアクションを実行するアカウントは、指定 GuardDuty 管理者アカウントです。
- ダッシュ (–) のセル – ダッシュ (-) の付いた表のセルは、アカウントがリストされたアクションを実行できないことを示します。

アクション	経由 AWS Organizations		招待により	
	委任 GuardDuty 管理者アカウント	関連付けられたメンバーアカウント	GuardDuty 管理者アカウント	関連付けられたメンバーアカウント
Enable GuardDuty	Any	–	Self	Self
Enable GuardDuty automatically for the entire organization (###, ##, ##)	All	–	–	–
View all Organizations member accounts regardless of GuardDuty status	Any	–	–	–
Generate sample findings	Self	Self	Self	Self
View all GuardDuty findings	Any	Self	Any	Self

Archive GuardDuty findings	Any	–	Any	–
Apply suppression rules	All	–	All	–
Create trusted IP list or threat lists	All	–	All	–
Update trusted IP list or threat lists	All	–	All	–
Delete trusted IP list or threat lists	All	–	All	–
Set EventBridge notification frequency	All	–	All	–
Set Amazon S3 location for exporting findings	All	Self	Self	Self
組織全体でオプションの 1 つ以上の保護プランを有効にする (ALL、NEW、NONE) これには Malware Protection for S3 は含まれません。	All	–	–	–

個々のアカウントの任意の GuardDuty 保護プランを有効にする これには、Malware Protection for EC2 と Malware Protection for S3 は含まれません。	Any	–	Any	–
Malware Protection for EC2	Any	–	Self	–
Malware Protection for EC2 – オンデマンドマルウェアスキャン	Any	Self	Self	Self
S3 向けのマルウェア防御	–	Self	–	Self
Disassociate a member account	Any+	–	Any	–
Disassociate from an administrator account	–	–	–	Self
Delete a disassociated member account	Any	–	Any	–

Suspend GuardDuty	Any*	–	Any*	–
Disable GuardDuty	Any*	–	Any*	Self

+ 委任 GuardDuty 管理者アカウントが、ALL 組織メンバーに対して自動有効化の詳細設定を設定していない場合にのみ、このアクションを実行できることを示します。

*委任 GuardDuty 管理者アカウントがメンバーアカウントで GuardDuty を直接無効にできないことを示します。委任 GuardDuty 管理者アカウントは、まずメンバーアカウントの関連付けを解除してから、削除する必要があります。この後、各メンバーアカウントは、自分のアカウントで GuardDuty を無効にすることができます。組織でこれらのタスクを実行する方法の詳細については、「GuardDuty 内でのメンバーアカウントの継続的管理」を参照してください。

メンバー関係のステータスについて

ListMembers や などの GuardDuty API オペレーションは、各メンバーアカウントの `relationshipStatus` フィールドGetMembersを返します。このフィールドは、管理者アカウントとメンバーアカウント間の関係の現在の状態を示します。GuardDuty がメンバーアカウントをアクティブにモニタリングしていることを確認する場合は、この値を確認します。また、メンバーアカウントがオンボーディングを完了しなかったか、モニタリングされなくなった理由のトラブルシューティングにも役立ちます。次のリストは、可能な値を示しています。

`Created`

管理者アカウントがこのアカウントをメンバーとして追加しましたが (を呼び出すなど CreateMembers)、メンバー関係はまだアクティブではありません。

`Invited`

管理者アカウントがこのメンバーアカウントに招待を送信し、招待の承諾を待っています。

`Enabled`

このメンバーアカウントは管理者アカウントに関連付けられており、GuardDuty はメンバーアカウントをアクティブにモニタリングしています。

Disabled

このメンバーアカウントは管理者アカウントに関連付けられていますが、GuardDuty はメンバーアカウントをアクティブにモニタリングしていません。

Removed

このメンバーアカウントは、管理者アカウントに関連付けられなくなりました。このステータスは通常、管理者アカウントが関連付けを削除したことを示します。

Resigned

このメンバーアカウントは、管理者アカウントに関連付けられなくなりました。このステータスは通常、メンバーアカウントが関連付けから自身を削除したことを示します。

Deleted

このメンバーアカウントは GuardDuty に存在しなくなりました。

EmailVerificationInProgress

このメンバーアカウントの招待は、E メールアドレスの検証を保留中です。

EmailVerificationFailed

このメンバーアカウントの招待は、E メールアドレスの検証に合格しませんでした。

RegionDisabled

GuardDuty が有効になっていないか、現在の で使用できないため、このメンバーアカウントをモニタリングできません AWS リージョン。

AccountSuspended

このメンバーの AWS アカウント は停止状態です。

CannotCreateDetectorInOrganizationMasterAccount

GuardDuty は、通常、必要なサービスにリンクされたロールのアクセス許可がないため、 AWS Organizations 管理アカウントにディテクターを作成できません。

を使用した GuardDuty アカウントの管理 AWS Organizations

AWS 組織では、管理アカウントは、この組織内の任意のアカウントを委任 GuardDuty 管理者アカウントとして指定できます。この管理者アカウントでは、GuardDuty は現在の でのみ自動的に有効に

なります AWS リージョン。デフォルトでは、管理者アカウントは、そのリージョン内の組織のすべてのメンバーアカウントに対して GuardDuty を有効化および管理できます。管理者アカウントは、この AWS 組織を表示してメンバーを追加できます。

以下のセクションでは、委任 GuardDuty 管理者アカウントとして実行できるさまざまなタスクについて説明します。

内容

- で GuardDuty を使用する際の考慮事項と推奨事項 AWS Organizations
- 委任 GuardDuty 管理者アカウントの指定に必要な許可
- 委任 GuardDuty 管理者アカウントの指定
- 組織の自動有効化の詳細設定の指定
- 組織へのメンバーの追加
- (オプション) 既存のメンバーアカウントの保護プランの有効化
- GuardDuty 内でのメンバーアカウントの継続的管理
- メンバーアカウントの GuardDuty の停止
- 管理者アカウントからのメンバーアカウントの関連付け解除 (削除)
- GuardDuty 組織からのメンバーアカウントの削除
- 委任 GuardDuty 管理者アカウントの変更

で GuardDuty を使用する際の考慮事項と推奨事項 AWS Organizations

以下の考慮事項と推奨事項は、委任 GuardDuty 管理者アカウントが GuardDuty でどのように機能するかを理解するのに役立ちます。

委任 GuardDuty 管理者アカウントは、最大 50,000 のメンバーを管理することができます。

> 委任 GuardDuty 管理者アカウント 1 つあたりのメンバーアカウント数は 50,000 件までとされています。これには、 を通じて追加されたメンバーアカウント AWS Organizations 、または GuardDuty 管理者アカウントの組織への招待を受け入れたメンバーアカウントが含まれます。ただし、 AWS 組織内に 50,000 を超えるアカウントが存在する可能性があります。
>
> メンバーアカウントの上限である 50,000 を超えると、CloudWatch から通知が送信され Health Dashboard、指定された委任 GuardDuty 管理者アカウントに E メールが送信されます。

委任 GuardDuty 管理者アカウントはリージョン別です。

GuardDuty とは異なり AWS Organizations、GuardDuty はリージョン別サービスです。GuardDuty 委任 GuardDuty 管理者アカウントとそのメンバーアカウントは、GuardDuty が有効になっている各希望するリージョン AWS Organizations で、 を介して追加する必要があります。組織の管理アカウントが米国東部 (バージニア北部) のみで委任 GuardDuty 管理者アカウントを指定している場合、委任 GuardDuty 管理者アカウントは、そのリージョンの組織に追加されたメンバーアカウントのみを管理します。GuardDuty が利用可能なリージョンの同等の機能の詳細については、「リージョンとエンドポイント」を参照してください。

オプトインリージョンの特殊なケース

- 委任 GuardDuty 管理者アカウントがオプトインリージョンをオプトアウトした場合、組織の GuardDuty の自動有効化設定が新しいメンバーアカウントのみ (NEW) またはすべてのメンバーアカウント (ALL) に設定されている場合でも、現在 GuardDuty が無効になっている組織内のメンバーアカウントの GuardDuty を有効にすることはできません。メンバーアカウントの設定に関する情報を確認するには、GuardDuty コンソールのナビゲーションペインの [アカウント] を開くか、ListMembers API を使用します。
- GuardDuty の自動有効化設定を NEW に設定して使用するときは、次の順序が守られていることを確認してください。

  1. メンバーアカウントがオプトインリージョンにオプトインします。
  2. AWS Organizationsで組織にメンバーアカウントを追加します。

  これらのステップの順序を変更すると、メンバーアカウントが組織にとって新規ではなくなるため、NEW の GuardDuty の自動有効化設定は特定のオプトインリージョンで機能しません。GuardDuty では 2 つの代替ソリューションがあります。

  - GuardDuty の自動有効化設定を ALL に設定します。これには、新規と既存のメンバーアカウントが含まれます。この場合、これらのステップの順序は関係ありません。
  - メンバーアカウントが既に組織の一部である場合は、GuardDuty コンソールまたは API を使用して、特定のオプトインリージョンでこのアカウントの GuardDuty の設定を個別に管理します。

AWS 組織がすべての で同じ委任 GuardDuty 管理者アカウントを持つために必要です AWS リージョン。

GuardDuty が有効になっているすべての AWS リージョン で 1 つのメンバーアカウントを委任 GuardDuty 管理者アカウントとして指定する必要があります。例えば、*## (######)* でメンバーアカウント *111122223333* を指定する場合、*### (##)* で別のメンバーアカウント

555555555555 を指定することはできません。他のすべてのリージョンで委任 GuardDuty 管理者アカウントと同じアカウントを使用する必要があります。

任意の時点で新しい委任 GuardDuty 管理者アカウントを指定できます。既存の委任 GuardDuty 管理者アカウントの削除の詳細については、「委任 GuardDuty 管理者アカウントの変更」を参照してください。

組織の管理アカウントを委任 GuardDuty 管理者アカウントとして設定することは推奨されません。

組織の管理アカウントは、委任 GuardDuty 管理者アカウントになることができます。ただし、AWS のセキュリティのベストプラクティスは最小特権の原則に従っており、この設定は推奨されていません。

委任 GuardDuty 管理者アカウントを変更しても、メンバーアカウントの GuardDuty は無効になりません。

委任 GuardDuty 管理者アカウントを削除すると、GuardDuty はこの委任 GuardDuty 管理者アカウントに関連付けられているすべてのメンバーアカウントを削除します。GuardDuty は、これらすべてのメンバーアカウントのために引き続き有効になっています。

委任 GuardDuty 管理者アカウントの指定に必要な許可

で Amazon GuardDuty の使用を開始するには AWS Organizations、組織の AWS Organizations 管理アカウントが アカウントを委任 GuardDuty 管理者アカウントとして指定します。これにより、GuardDuty を信頼されたサービスとして有効にします AWS Organizations。また、委任 GuardDuty 管理者アカウントの GuardDuty が有効になり、委任管理者アカウントが現在のリージョンの組織内の他のアカウントの GuardDuty を有効化および管理できるようになります。これらのアクセス許可の付与方法については、「 を AWS Organizations 他の AWS サービスで使用する」を参照してください。

AWS Organizations 管理アカウントとして、組織の委任 GuardDuty 管理者アカウントを指定する前に、次の GuardDuty アクションを実行できることを確認します: `guardduty:EnableOrganizationAdminAccount`。このアクションにより、GuardDuty を使用して組織の委任 GuardDuty 管理者アカウントを指定できます。また、組織に関する情報を取得するのに役立つ AWS Organizations アクションを実行できることを確認する必要があります。

これらのアクセス許可を付与するには、アカウントの AWS Identity and Access Management (IAM) ポリシーに次のステートメントを含めます。

```
{
```

```
    "Sid": "PermissionsForGuardDutyAdmin",
    "Effect": "Allow",
    "Action": [
        "guardduty:EnableOrganizationAdminAccount",
        "organizations:EnableAWSServiceAccess",
        "organizations:RegisterDelegatedAdministrator",
        "organizations:ListDelegatedAdministrators",
        "organizations:ListAWSServiceAccessForOrganization",
        "organizations:DescribeOrganizationalUnit",
        "organizations:DescribeAccount",
        "organizations:DescribeOrganization",
        "organizations:ListAccounts",
        "organizations:ListRoots"
    ],
    "Resource": "*"
}
```

AWS Organizations 管理アカウントを委任 GuardDuty 管理者アカウントとして指定する場合、アカウントには IAM アクション も必要ですCreateServiceLinkedRole。このアクションにより、管理アカウントの GuardDuty を初期化できます。ただし、許可の追加に進む前に、「で GuardDuty を使用する際の考慮事項と推奨事項 AWS Organizations」を確認してください。

管理アカウントを委任 GuardDuty 管理者アカウントとして指定し続けるには、IAM ポリシーに次のステートメントを追加し、*111122223333* を組織の管理アカウントの AWS アカウント ID に置き換えます。

```
{
 "Sid": "PermissionsToEnableGuardDuty"
 "Effect": "Allow",
 "Action": [
  "iam:CreateServiceLinkedRole"
 ],
 "Resource": "arn:aws:iam::111122223333:role/aws-service-role/guardduty.amazonaws.com/
AWSServiceRoleForAmazonGuardDuty",
 "Condition": {
  "StringLike": {
   "iam:AWSServiceName": "guardduty.amazonaws.com"
  }
 }
}
```

委任 GuardDuty 管理者アカウントの指定

このセクションでは、GuardDuty 組織の委任管理者を指定する手順について説明します。

AWS 組織の管理アカウントとして、委任 GuardDuty 管理者アカウントの運用方法考慮事項とレコメンデーションに関する を必ずお読みください。続行する前に、委任 GuardDuty 管理者アカウントの指定に必要な許可があることを確認してください。

任意のアクセス方法を選択して、組織の委任 GuardDuty 管理者アカウントを指定します。管理アカウントのみがこの手順を実行できます。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   サインインするには、 AWS Organizations 組織の管理アカウントの認証情報を使用します。
2. ページの右上隅にある AWS リージョン セレクターを使用して、組織の委任 GuardDuty 管理者アカウントを指定するリージョンを選択します。
3. GuardDuty が現在のリージョンで管理アカウントに対して有効化されているかどうかに応じて、次のいずれかを実行します。

   - GuardDuty が有効になっていない場合は、[Amazon GuardDuty - すべての機能] を選択し、[今すぐ始める] を選択します。このアクションにより、[GuardDuty にようこそ] ページに移動します。
   - GuardDuty が有効化されている場合は、ナビゲーションペインの [設定] を選択します。
4. 委任管理者に、組織の委任 GuardDuty 管理者アカウントとして指定するアカウントの 12 桁の AWS アカウント ID を入力します。
5. [委任] を選択します。
6. (推奨) 上記のステップを繰り返して、GuardDuty が有効になってい AWS リージョン る各 で委任 GuardDuty 管理者アカウントを指定します。

API/CLI

1. 組織の管理アカウントの AWS アカウント の認証情報enableOrganizationAdminAccountを使用して を実行します。

   - または、 AWS Command Line Interface を使用してこれを行うことができます。次の AWS CLI コマンドは、現在のリージョンの委任 GuardDuty 管理者アカウントのみを指

定します。次の AWS CLI コマンドを実行し、*111111111111* を委任 GuardDuty 管理者アカウントとして指定するアカウントの AWS アカウント ID に置き換えます。

```
aws guardduty enable-organization-admin-account --admin-account-
id 111111111111
```

他のリージョンの委任 GuardDuty 管理者アカウントを指定するには、 AWS CLI コマンドでリージョンを指定します。次の例では、米国西部 (オレゴン) で委任 GuardDuty 管理者アカウントを有効にする方法を示します。*us-west-2* を、委任 GuardDuty 管理者アカウントを割り当てるリージョンに置き換えてください。

```
aws guardduty enable-organization-admin-account --admin-account-
id 111111111111 --region us-west-2
```

GuardDuty AWS リージョン が利用可能な の詳細については、「」を参照してくださいリージョンとエンドポイント。

委任 GuardDuty 管理者アカウントに対して GuardDuty が無効になっている場合、いかなるアクションも実行できません。まだ有効にしていない場合は、新しく指定された委任 GuardDuty 管理者アカウントのために GuardDuty を必ず有効にしてください。

2. (推奨) 上記のステップを繰り返して、GuardDuty が有効になってい AWS リージョン る各 で委任 GuardDuty 管理者アカウントを指定します。

組織の自動有効化の詳細設定の指定

GuardDuty の組織の自動有効化機能では、組織内の ALL の既存または NEW のメンバーアカウントに対して同じ GuardDuty および保護プランのステータスを 1 つのステップで設定できます。同様に、NONE を選択して、メンバーアカウントに対してアクションを実行したくない場合を指定することもできます。次のステップでは、これらの設定について説明し、特定の設定を使用する状況も示します。

> **Note**
>
> S3 向けのマルウェア防御 を除くすべての保護プランに対して自動有効化の設定を行うことができます。

任意のアクセス方法を選択して、組織の自動有効化の詳細設定を更新します。

Console

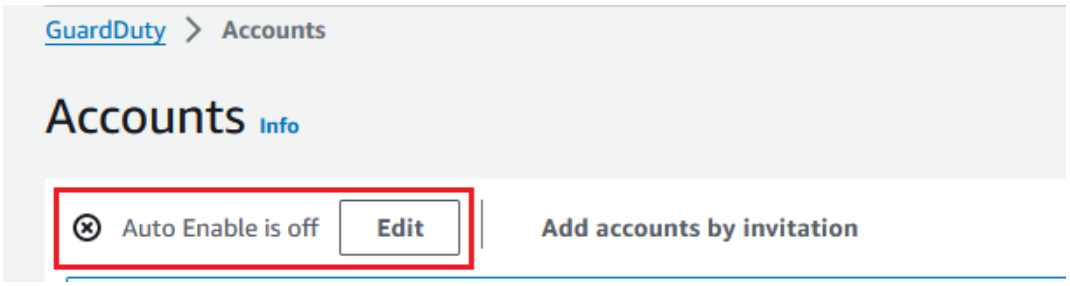
1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   サインインするには、GuardDuty 管理者アカウントの認証情報を使用します。

2. ナビゲーションペインで、[Accounts] (アカウント) を選択します。

   [アカウント] ページでは、GuardDuty 管理者アカウントは、組織に属するメンバーアカウントのために GuardDuty とオプションの保護プランを [自動有効化] するための設定オプションを使用できます。

3. 既存の自動有効化設定を更新するには、[編集] を選択します。



このサポートは、GuardDuty と でサポートされているすべてのオプション保護プランを設定するために利用できます AWS リージョン。メンバーアカウントのために、GuardDuty の次の設定オプションのいずれかを選択できます。

- [すべてのアカウントについて有効にする] (**ALL**) – 組織内のすべてのアカウントのために、対応するオプションを有効にする場合に選択します。これには、組織に参加した新しいアカウントと、組織から一時停止または削除された可能性のあるアカウントが含まれます。これには、委任 GuardDuty 管理者アカウントも含まれます。

  > **Note**
  >
  > すべてのメンバーアカウントの設定を更新するには、最大 24 時間かかる場合があります。

- [新しいアカウントの自動有効化] (**NEW**) – 新しいメンバーアカウントが組織に参加したときに、それらのアカウントのために GuardDuty またはオプションの保護プランを自動的に有効にする場合に選択します。

- [有効にしない] (**NONE**) – 組織内の新しいアカウントのために、対応するオプションを有効にしない場合に選択します。この場合、GuardDuty 管理者アカウントは各アカウントを個別に管理します。

  自動有効化設定を ALL または NEW から NONE に更新しても、このアクションは既存のアカウントの対応するオプションを無効にしません。この設定は、組織に参加する新しいアカウントに適用されます。自動有効化設定を更新すると、対応するオプションが有効になっている新しいアカウントはなくなります。

> **Note**
>
> 委任 GuardDuty 管理者アカウントがオプトインリージョンをオプトアウトした場合、組織の GuardDuty の自動有効化設定が新しいメンバーアカウントのみ (NEW) またはすべてのメンバーアカウント (ALL) に設定されている場合でも、現在 GuardDuty が無効になっている組織内のメンバーアカウントの GuardDuty を有効にすることはできません。メンバーアカウントの設定に関する情報を確認するには、GuardDuty コンソールのナビゲーションペインの [アカウント] を開くか、ListMembers API を使用します。

4. [Save changes] (変更の保存) をクリックします。
5. (オプション) 各リージョンで同じ詳細設定を使用する場合は、サポートされている各リージョンで個別に詳細設定を更新します。

   一部のオプションの保護プランは、GuardDuty AWS リージョン が利用可能なすべての で利用できない場合があります。詳細については、「リージョンとエンドポイント」を参照してください。

API/CLI

1. 委任 GuardDuty 管理者アカウントの認証情報を使用して UpdateOrganizationConfiguration を実行し、組織のそのリージョンの GuardDuty とオプションの保護プランを自動的に設定します。さまざまな自動有効化の設定については、「autoEnableOrganizationMembers」を参照してください。

   アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

リージョンでサポートされているオプションの保護プランの自動有効化の設定を行うには、各保護プランの対応するドキュメントセクションに記載されているステップに従ってください。

2. 現在のリージョンで組織の詳細設定を検証できます。describeOrganizationConfiguration を実行します。委任 GuardDuty 管理者アカウントのディテクター ID を必ず指定してください。

> **Note**
>
> すべてのメンバーアカウントの設定を更新するには、最大 24 時間かかる場合があります。

3. または、次の AWS CLI コマンドを実行して、組織に参加する新しいアカウント (NEW)、すべてのアカウント ()、ALLまたは組織内のすべてのアカウント (NONE) に対して、そのリージョンで GuardDuty を自動的に有効または無効にする設定を行います。詳細については、「autoEnableOrganizationMembers」を参照してください。必要に応じて、NEW を ALL または NONE に置き換える必要がある場合があります。ALL で保護プランを設定すると、委任 GuardDuty 管理者アカウントの保護プランも有効になります。組織の設定を管理する委任 GuardDuty 管理者アカウントのディテクター ID を必ず指定してください。

   アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty update-organization-configuration --detector-
id 12abc34d567e8fa901bc2d34e56789f0 --auto-enable-organization-members=NEW
```

4. 現在のリージョンで組織の詳細設定を検証できます。委任 GuardDuty 管理者アカウントのディテクター ID を使用して、次の AWS CLI コマンドを実行します。

```
aws guardduty describe-organization-configuration --detector-
id 12abc34d567e8fa901bc2d34e56789f0
```

(推奨) 委任 GuardDuty 管理者アカウントのディテクター ID を使用して、各リージョンで前のステップを繰り返します。

> **Note**
>
> 委任 GuardDuty 管理者アカウントがオプトインリージョンをオプトアウトした場合、組織の GuardDuty の自動有効化設定が新しいメンバーアカウントのみ (NEW) またはすべてのメンバーアカウント (ALL) に設定されている場合でも、現在 GuardDuty が無効になっている組織内のメンバーアカウントの GuardDuty を有効にすることはできません。メンバーアカウントの設定に関する情報を確認するには、GuardDuty コンソールのナビゲーションペインの [アカウント] を開くか、ListMembers API を使用します。

組織へのメンバーの追加

委任 GuardDuty 管理者アカウントとして、GuardDuty 組織に 1 AWS アカウント つ以上の を追加できます。GuardDuty メンバーとしてアカウントを追加すると、そのリージョンで GuardDuty が自動的に有効になります。組織管理アカウントには例外があります。管理アカウントアカウントを GuardDuty メンバーとして追加する前に、GuardDuty を有効にする必要があります。

任意の方法を選択して、GuardDuty 組織にメンバーアカウントを追加します。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   サインインするには、委任 GuardDuty 管理者アカウントの認証情報を使用します。

2. ナビゲーションペインで、[Accounts] (アカウント) を選択します。

   アカウントテーブルには、アクティブ (停止されていない AWS アカウント) で、委任 GuardDuty 管理者アカウントに関連付けられている可能性のあるすべてのメンバーアカウントが表示されます。メンバーアカウントが組織の管理者アカウントに関連付けられている場合、[タイプ] は [Organizations 経由] または [招待による] のいずれかになります。メンバーアカウントが組織の GuardDuty 管理者アカウントに関連付けられていない場合、このメンバーアカウントの [タイプ] は [メンバーではありません] になります。

3. メンバーとして追加する 1 つまたは複数のアカウント ID を選択します。これらのアカウント ID の [タイプ] は [Organizations 経由] である必要があります。

   招待を通じて追加されたアカウントは、組織の一部ではありません。このようなアカウントは個別に管理できます。詳細については、「招待によるアカウントの管理」を参照してください。

4. [アクション] ドロップダウンを選択し、[メンバーを追加] を選択します。このアカウントをメンバーとして追加すると、GuardDuty の自動有効化の設定が適用されます。組織の自動有効化の詳細設定の指定 の設定に基づいて、これらのアカウントの GuardDuty 設定が変更される場合があります。
5. [ステータス] 列の下矢印を選択して [メンバーではありません] のステータスでアカウントをソートし、現在のリージョンで GuardDuty が有効になっていない各アカウントを選択できます。

   アカウントテーブルにリストされているいずれのアカウントも、まだメンバーとして追加されていない場合は、現在のリージョンですべての組織アカウントのために GuardDuty を有効にすることができます。ページ上部のバナーで [有効にする] を選択します。このアクションにより、GuardDuty の [自動有効化] の設定が自動的にオンになり、組織に参加する新しいアカウントのために GuardDuty が有効になります。
6. アカウントをメンバーとして追加するには、[確認] を選択します。また、このアクションにより、選択したすべてのアカウントのために GuardDuty が有効になります。アカウントの [ステータス] が [有効] に変わります。
7. (推奨) 各ステップでこれらのステップを繰り返します AWS リージョン。これにより、委任 GuardDuty 管理者アカウントは、GuardDuty が有効になっているすべてのリージョンのメンバーアカウントの検出結果や他の設定を管理できるようになります。

   自動有効化機能は、組織の将来のメンバー全員に対して GuardDuty を有効にします。これにより、委任 GuardDuty 管理者アカウントは、組織内で作成された、または組織に追加された新しいメンバーを管理できます。メンバーアカウント数が 50,000 の上限に達すると、自動有効化機能は自動的にオフになります。メンバーアカウントを削除し、メンバーの総数が 50,000 未満になると、自動有効化機能が再度オンになります。

API/CLI

- 委任 GuardDuty 管理者アカウントの認証情報を使用して CreateMembers を実行します。

  委任 GuardDuty 管理者アカウントのリージョンディテクター ID と、GuardDuty メンバーとして追加するアカウントのアカウント詳細 (AWS アカウント IDs と対応する E メールアドレス) を指定する必要があります。この API オペレーションを使用して 1 名以上のメンバーを作成できます。

  組織で CreateMembers を実行する場合、新しいメンバーアカウントが組織に参加すると、新規メンバー用の自動有効化の詳細設定が適用されます。既存のメンバーアカウントで

CreateMembers を実行すると、組織設定は既存のメンバーにも適用されます。これにより、既存のメンバーアカウントの現在の設定が変更される場合があります。

AWS Organizations API リファレンスListAccountsで を実行して、 AWS 組織内のすべてのアカウントを表示します。

- または、 を使用することもできます AWS Command Line Interface。次の AWS CLI コマンドを実行し、必ず自分の有効なディテクター ID、 AWS アカウント ID、およびアカウント ID に関連付けられたメールアドレスを使用してください。

  アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

  ```
  aws guardduty create-members --detector-id 12abc34d567e8fa901bc2d34e56789f0
   --account-details AccountId=111122223333,Email=guardduty-member-
  name@amazon.com
  ```

  次の AWS CLI コマンドを実行して、すべての組織メンバーのリストを表示できます。

  ```
  aws organizations list-accounts
  ```

このアカウントをメンバーとして追加すると、GuardDuty の自動有効化の設定が適用されます。

(オプション) 既存のメンバーアカウントの保護プランの有効化

次の手順には、[アカウント] ページを使用して、既存のメンバーアカウントの保護プランを有効にする手順が含まれています。API または を使用してこれを行う手順については AWS CLI、特定の保護プランに関連するドキュメントを参照してください。

[アカウント] ページを通じて個々のアカウントのために保護プランを有効化できます。

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   委任 GuardDuty 管理者アカウントの認証情報を使用します。
2. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
3. 保護プランを設定するアカウントを 1 つ以上選択します。設定する保護プランごとに次のステップを繰り返します。

a. [保護プランを編集] を選択します。

b. 保護プランのリストから、設定する保護プランを 1 つ選択します。

c. この保護プランについて実行するいずれかのアクションを選択し、[確認] を選択します。

d. 選択したアカウントについて、設定された保護プランに対応する列で、更新された設定が [有効] または [無効] として表示されます。

GuardDuty 内でのメンバーアカウントの継続的管理

委任 GuardDuty 管理者アカウントには、サポートされている各 AWS リージョンの組織内のすべてのアカウントについて、GuardDuty とオプションの保護プランの設定を維持する責任があります。以下のセクションでは、GuardDuty またはオプションの保護プランの設定ステータスを維持するオプションについて説明します。

各リージョンで組織全体の設定ステータスを維持するには

- GuardDuty コンソールを使用して組織全体の自動有効化の詳細設定を指定する – 組織内のすべて (ALL) のメンバーまたは組織に参加する新しい (NEW) メンバーに対して GuardDuty を自動的に有効化するか、組織内のいずれのメンバーでも自動有効化しない (NONE) ことを選択できます。

  また、GuardDuty 内の任意の保護プランに対して同じ設定または異なる設定を指定することもできます。

  組織内のすべてのメンバーアカウントの設定を更新するには、最大 24 時間かかる場合があります。

- API を使用して自動有効化の詳細設定を更新する – UpdateOrganizationConfiguration を実行して、組織の GuardDuty とそのオプションの保護プランを自動的に設定します。CreateMembers を実行して組織内に新しいメンバーアカウントを追加すると、構成された設定が自動的に適用されます。既存のメンバーアカウントで CreateMembers を実行すると、組織設定は既存のメンバーにも適用されます。これにより、既存のメンバーアカウントの現在の設定が変更される場合があります。

  組織内のすべてのアカウントを表示するには、「AWS Organizations API Reference」の ListAccounts を実行します。

各リージョンで個別にメンバーアカウントの設定ステータスを維持するには

- 組織内のすべてのアカウントを表示するには、「AWS Organizations API Reference」の ListAccounts を実行します。
- 選択したメンバーアカウントで異なる設定ステータスにしたい場合は、メンバーアカウントごとに UpdateMemberDetectors を個別に実行します。

  GuardDuty コンソールを使用して [アカウント] ページに移動することにより、同じタスクを実行できます。

  コンソールまたは API を使用して個々のアカウントの保護プランを有効にする方法の詳細については、対応する保護プランの設定ページを参照してください。

メンバーアカウントの GuardDuty の停止

委任 GuardDuty 管理者アカウントは、組織内のメンバーアカウントの GuardDuty サービスを停止できます。これを行った場合でも、メンバーアカウントは引き続き GuardDuty 組織に残ります。これらのメンバーアカウントの GuardDuty を後で再度有効にすることもできます。ただし、最終的にこのメンバーアカウントの関連付けを解除 (削除) する場合は、このセクションの手順に従った後に、「管理者アカウントからのメンバーアカウントの関連付け解除 (削除)」の手順に従う必要があります。

メンバーアカウントで GuardDuty を停止すると、以下の変更が予想されます。

- GuardDuty は AWS 環境のセキュリティをモニタリングしたり、新しい検出結果を生成したりしなくなりました。
- メンバーアカウントの既存の検出結果はそのまま残ります。
- GuardDuty が停止したメンバーアカウントでは、GuardDuty の料金は発生しません。

  メンバーアカウントがアカウントの 1 つ以上のバケットに対して Malware Protection for S3 を有効にしている場合、GuardDuty を停止しても、Malware Protection for S3 の設定には影響しません。メンバーアカウントでは、Malware Protection for S3 の使用コストが引き続き発生します。メンバーアカウントが Malware Protection for S3 の使用を停止するには、保護されたバケットのこの機能を無効にする必要があります。詳細については、「保護されたバケットの Malware Protection for S3 の無効化」を参照してください。

任意の方法を選択して、組織内のメンバーアカウントの GuardDuty を停止します。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   サインインするには、委任 GuardDuty 管理者アカウントの認証情報を使用します。
2. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
3. アカウントページで、GuardDuty を停止するアカウントを 1 つ以上選択します。
4. [アクション] ドロップダウンメニューを選択し、[GuardDuty の停止] を選択します。
5. [GuardDuty の停止] を選択して、選択を確定します。

   これにより、メンバーアカウントの [ステータス] が [無効 (停止)] に変更されます。

   メンバーアカウントを関連付け解除または削除する追加のリージョンごとに、前述のステップを繰り返します。

API

1. GuardDuty を停止するメンバーアカウントのアカウント ID を取得するには、ListMembers API を使用します。リクエストには `OnlyAssociated` パラメータを含めます。このパラメータの値を `true` に設定した場合、GuardDuty は現在 GuardDuty メンバーであるアカウントのみの詳細を提供する `members` 配列を返します。

   または、 AWS Command Line Interface (AWS CLI) を使用して次のコマンドを実行できます。

   ```
   aws guardduty list-members --only-associated true --region us-east-1
   ```

   us-east-1 は、このアカウントの GuardDuty を停止するリージョンに置き換えます。
2. 1 つ以上の GuardDuty メンバーアカウントを停止するには、StopMonitoringMembers を実行してメンバーアカウントの GuardDuty を停止します。

   または、 AWS CLI を使用して次のコマンドを実行できます。

   ```
   aws guardduty stop-monitoring-members --detector-id
    12abc34d567e8fa901bc2d34EXAMPLE --account-ids 111122223333 --region us-east-1
   ```

   us-east-1 は、このアカウントを停止するリージョンに置き換えます。削除するアカウント ID のリストを使用する場合は、スペース文字で区切ります。

さらにこのメンバーアカウントの関連付けを解除 (削除) する場合は、「管理者アカウントからのメンバーアカウントの関連付け解除 (削除)」の手順に従います。

管理者アカウントからのメンバーアカウントの関連付け解除 (削除)

メンバーアカウントからの GuardDuty 設定の構成とデータへのアクセスを停止する場合は、そのアカウントを GuardDuty メンバーアカウントから削除します。これを行うには、GuardDuty 管理者アカウントからそのアカウントの関連付けを解除 (削除) します。

GuardDuty メンバーアカウントを解除すると、以下の処理が行われます。

- GuardDuty は現在のアカウントで有効のままですが AWS リージョン、そのアカウントは委任 GuardDuty 管理者アカウントとの関連付けが解除されます。
- 関連付けが解除されたアカウントは、アカウントインベントリに引き続き表示されます。
- GuardDuty 管理者アカウントは、このスタンドアロンアカウントの検出結果にアクセスできなくなりました。
- アカウント所有者には、関連付けの解除について通知されません。

関連付けが解除されたアカウントは、後で組織に追加できます。

任意の方法を選択して、組織からメンバーアカウントを関連付け解除 (削除) します。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   サインインするには、委任 GuardDuty 管理者アカウントの認証情報を使用します。
2. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
3. [アカウント] テーブルで、[タイプ] が [Organizations 経由]、[ステータス] が [有効] のアカウントを削除できます。

   同じ [タイプ] と [ステータス] のアカウントを 1 つ以上選択します。
4. [アクション] ドロップダウンメニューから、[アカウントの関連付けを解除する] を選択します。
5. [アカウントの関連付けを解除する] を選択して、選択を確定します。
6. 選択したアカウントの [ステータス] 値が [メンバーではありません] に変更されます。[アカウント] ページの右上隅にある [Organizations 経由 (アクティブ / すべて)] の数は、更新を反映して変更されます。

メンバーアカウントの関連付けを解除する追加のリージョンごとに、前述のステップを繰り返します。

API

1. 削除するメンバーアカウントのアカウント ID を取得するには、ListMembers API を使用します。リクエストには `OnlyAssociated` パラメータを含めます。このパラメータの値を `true` に設定した場合、GuardDuty は現在 GuardDuty メンバーであるアカウントのみの詳細を提供する `members` 配列を返します。

   または、 AWS Command Line Interface (AWS CLI) を使用して次のコマンドを実行できます。

   ```
   aws guardduty list-members --only-associated true --region us-east-1
   ```

   us-east-1 は、このアカウントを削除するリージョンに置き換えます。
2. 1 つ以上の GuardDuty メンバーアカウントを削除するには、DisassociateMembers を実行して、管理者アカウントに関連付けられているメンバーアカウントを削除します。

   または、 AWS CLI を使用して次のコマンドを実行できます。

   ```
   aws guardduty disassociate-members --detector-id 12abc34d567e8fa901bc2d34EXAMPLE
    --account-ids 111122223333 --region us-east-1
   ```

   us-east-1 は、このアカウントを削除するリージョンに置き換えます。削除するアカウント ID のリストを使用する場合は、スペース文字で区切ります。

GuardDuty 組織からのメンバーアカウントの削除

委任 GuardDuty 管理者アカウントとして、メンバーアカウントの関連付けを解除し、そのメンバーアカウントを GuardDuty 組織に保持する必要がなくなったら、そのメンバーアカウントを GuardDuty 組織から削除できます。このメンバーアカウントは、アカウントインベントリに表示されなくなります。ただし、このメンバーアカウントで GuardDuty が停止されていない場合は、GuardDuty と専用保護プランの設定は同じままです。このアカウントはスタンドアロンアカウントになり、GuardDuty を無効にすることができます。

このステップでは、 AWS 組織からメンバーアカウントは削除されません。

任意の方法を選択して、GuardDuty 組織からメンバーアカウントを削除します。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   サインインするには、委任 GuardDuty 管理者アカウントの認証情報を使用します。
2. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
3. [アカウント] テーブルで、[タイプ] が [Organizations 経由]、[ステータス] が [削除済み (関連付け解除)] のアカウントを削除できます。

   同じ [タイプ] と [ステータス] のアカウントを 1 つ以上選択します。
4. [アクション] ドロップダウンメニューから [アカウントの削除] を選択します。
5. [アカウントの削除] を選択して、選択を確定します。選択したアカウントメンバーは、[アカウント] テーブルに表示されなくなります。

   このメンバーアカウントを削除する追加のリージョンごとに、前述のステップを繰り返します。

API/CLI

1. 削除するメンバーアカウントのアカウント ID を取得するには、ListMembers API を使用します。リクエストには `OnlyAssociated` パラメータを含めます。このパラメータの値を `false` に設定した場合、GuardDuty は現在関連付けが解除された GuardDuty メンバーであるアカウントのみの詳細を提供する `members` 配列を返します。

   または、 AWS Command Line Interface (AWS CLI) を使用して次のコマンドを実行できます。

   ```
   aws guardduty list-members --detector-id 12abc34d567e8fa901bc2d34EXAMPLE --only-associated="false" --region us-east-1
   ```

   12abc34d567e8fa901bc2d34EXAMPLE を委任 GuardDuty 管理者アカウントディテクター ID に置き換え、*us-east-1* をこのアカウントを削除するリージョンに置き換えます。
2. 1 つ以上の GuardDuty メンバーアカウントを削除するには、DeleteMembers を実行して、GuardDuty 組織からメンバーアカウントを削除します。

   または、 AWS CLI を使用して次のコマンドを実行できます。

```
aws guardduty delete-members --detector-id 12abc34d567e8fa901bc2d34EXAMPLE --
account-ids 111122223333 --region us-east-1
```

12abc34d567e8fa901bc2d34EXAMPLE を委任 GuardDuty 管理者アカウントディテクター ID に置き換え、*us-east-1* をこのアカウントを削除するリージョンに置き換えます。削除するアカウント ID のリストを使用する場合は、スペース文字で区切ります。

委任 GuardDuty 管理者アカウントの変更

各リージョンで組織の委任 GuardDuty 管理者アカウントを削除し、各リージョンで新しい管理者を委任できます。リージョン内の組織のメンバーアカウントのセキュリティ体制を維持するには、そのリージョンに委任 GuardDuty 管理者アカウントが必要です。

> **メモ**
>
> 委任 GuardDuty 管理者アカウントを削除する前に、委任 GuardDuty 管理者アカウントに関連付けられているすべてのメンバーアカウントの関連付けを解除し、GuardDuty 組織から削除する必要があります。これらの手順の詳細については、次のドキュメントを参照してください。
>
> - 管理者アカウントからのメンバーアカウントの関連付け解除 (削除)
> - GuardDuty 組織からのメンバーアカウントの削除

既存の委任 GuardDuty 管理者アカウントの削除

ステップ 1 - 各リージョンの既存の委任 GuardDuty 管理者アカウントを削除するには

1. 既存の委任 GuardDuty 管理者アカウントとして、管理者アカウントにリンクされているすべてのメンバーアカウントを一覧表示します。`OnlyAssociated=false` を使用して ListMembers を実行します。
2. GuardDuty またはオプションの保護プランの自動有効化の詳細設定が `ALL` に設定されている場合は、UpdateOrganizationConfiguration を実行して組織設定を `NEW` または `NONE` に更新します。このアクションにより、次のステップですべてのメンバーアカウントの関連付けを解除するときのエラーを防止します。

3. DisassociateMembers を実行して、管理者アカウントに関連付けられているすべてのメンバーアカウントの関連付けを解除します。
4. DeleteMembers を実行して、管理者アカウントとメンバーアカウント間の関連付けを削除します。
5. 組織管理アカウントとして、DisableOrganizationAdminAccount を実行して既存の委任 GuardDuty 管理者アカウントを削除します。
6. この委任 GuardDuty 管理者アカウントがある各 AWS リージョン で、これらのステップを繰り返します。

ステップ 2 - で既存の委任 GuardDuty 管理者アカウントの登録を解除するには AWS Organizations (1 回限りのグローバルアクション)

- 「AWS Organizations API Reference」の DeregisterDelegatedAdministrator を実行して、 AWS Organizationsで既存の委任 GuardDuty 管理者アカウントを登録解除します。

  または、次の AWS CLI コマンドを実行できます。

  ```
  aws organizations deregister-delegated-administrator --account-id 111122223333 --
  service-principal guardduty.amazonaws.com
  ```

  111122223333 は既存の委任 GuardDuty 管理者アカウントに必ず置き換えてください。

  古い委任 GuardDuty 管理者アカウントを登録解除したら、新しい委任 GuardDuty 管理者アカウントにメンバーアカウントとして追加できます。

各リージョンでの新しい委任 GuardDuty 管理者アカウントの指定

1. GuardDuty コンソール、API、または AWS CLIから任意のアクセス方法を使用して、各リージョンの新しい委任 GuardDuty 管理者アカウントを指定します。詳細については、「委任 GuardDuty 管理者アカウントの指定」を参照してください。
2. DescribeOrganizationConfiguration を実行して、組織の現在の自動有効化設定を表示します。

> **Important**
>
> 新しい委任 GuardDuty 管理者アカウントにメンバーを追加する前に、組織の自動有効化設定を確認する必要があります。この設定は、新しい委任 GuardDuty 管理者アカウントおよび選択したリージョンに固有であり、 AWS Organizationsには関連しません。新し

い委任 GuardDuty 管理者アカウントの下に (新規または既存の) 組織メンバーアカウントを追加すると、GuardDuty またはオプションの保護プランが有効になるときに、新しい委任 GuardDuty 管理者アカウントの自動有効化設定が適用されます。

GuardDuty コンソール、API、または AWS CLIから任意のアクセス方法を使用して、新しい委任 GuardDuty 管理者アカウントの組織設定を変更します。詳細については、「組織の自動有効化の詳細設定の指定」を参照してください。

組織ポリシーを使用したアカウントの管理

GuardDuty 組織ポリシーを使用すると、組織内のアカウント全体で Amazon GuardDuty を一元的に有効化および管理できます。 AWS Organizations 宣言ポリシーを使用して、どの組織エンティティ (組織ルート、組織単位 (OUs)、または個々のアカウント) に GuardDuty 基盤ディテクターとその保護プランが自動的に有効になっており、委任 GuardDuty 管理者アカウントを通じて管理されているかを指定します。組織に参加するアカウント、またはポリシーがアタッチされた OU に移行するアカウントは、ポリシーを継承し、有効なポリシーで定義されている内容に従って GuardDuty とその保護プランを設定します。

GuardDuty 組織ポリシーは、「」で説明されている自動有効化設定とは異なるメカニズムですを使用した GuardDuty アカウントの管理 AWS Organizations。ポリシーの概念、構造、継承、検証、ポリシーの例については、 AWS Organizations 「 ユーザーガイド」のGuardDuty ポリシー」を参照してください。

内容

- 前提条件
- 組織ポリシーの仕組み
- コンソールでのポリシーの作成
- ポリシーを適用する前にポリシーの影響を確認する
- 考慮事項と制限事項

前提条件

GuardDuty 組織ポリシーを使用する前に、以下の前提条件を満たしてください。

- で GuardDuty サービスの信頼されたアクセスを有効にします AWS Organizations。

- 委任 GuardDuty 管理者アカウントを割り当てます。
- 委任された管理者に GuardDuty ポリシーを管理するアクセス許可を付与する委任ポリシーをアタッチします。
- 組織ルートでGUARDDUTY_POLICYポリシータイプを有効にします。これは、委任管理者が GuardDuty コンソールで最初のポリシーを作成すると自動的に発生します。

> **Warning**
>
> GUARDDUTY_POLICY ポリシータイプを有効にすると、ポリシーをアタッチする前でも、リージョン GuardDuty 自動有効化設定は適用されません。コンソールは、委任管理者が最初のポリシーを作成するときにこのポリシータイプを自動的に有効にするため、意図せずに自動有効化への依存を停止できます。最初のポリシーを作成する前に、既存の自動有効化設定を確認してください。

詳細な手順については、「 AWS Organizations ユーザーガイド」のGuardDuty ポリシー」を参照してください。

組織ポリシーの仕組み

GuardDuty 組織ポリシーは、有効になっている GuardDuty 機能、リージョン、アカウントを決定します。以下のセクションでは、ポリシーがリージョン間でどのように適用されるか、組織全体でポリシーを継承する方法、アクティブなポリシーが GuardDuty アカウントの管理方法をどのように変更するか、およびポリシーを使用する一般的な方法について説明します。

内容

- リージョン設定
- ポリシーの継承と優先順位
- ポリシーが GuardDuty アカウント管理に与える影響
- ユースケースの例

リージョン設定

GuardDuty 組織ポリシーは、リージョンごとに有効化を設定します。デフォルト設定は、GuardDuty が利用可能なすべてのリージョンに適用されます。リージョンオーバーライドを追加して、特定の

リージョンに別の設定を付与できます。デフォルト設定はオプションです。これを省略すると、指定したリージョンのみが管理され、他のすべてのリージョンは管理されません。

ポリシーがすべてのリージョンをカバーするデフォルト設定など、GuardDuty を広く有効にすると、GuardDuty は、リソースやワークロードがないリージョンを含め、有効になっているすべてのリージョンで AWS CloudTrail グローバルサービスイベントを処理します。詳細については、「GuardDuty が AWS CloudTrail グローバルイベントを処理する方法」を参照してください。

ポリシーは、GuardDuty と保護プランが利用可能なリージョンでのみ管理します。ポリシーが利用できないリージョンで GuardDuty または保護プランを有効にしている場合、ポリシーはそのリージョンでエラーなしでスキップします。ポリシーを設定する前に、GuardDuty と各保護プランが利用可能な場所を確認してください。詳細については、「リージョン固有機能の可用性」を参照してください。

リージョンオーバーライドは、そのリージョンのデフォルト設定を完全に置き換えます。デフォルトとマージされないため、そのリージョンで管理するすべての保護プランを指定する必要があります。上書きから除外した保護プランは、デフォルト設定で管理されていても、そのリージョンでは管理されません。このため、ポリシーが管理する保護プランはリージョンごとに異なる場合があります。

完全なポリシー構造とリージョンの解決方法については、 AWS Organizations 「 ユーザーガイド」のGuardDuty ポリシー」を参照してください。

ポリシーの継承と優先順位

GuardDuty 組織ポリシーは AWS Organizations 継承モデルに従います。組織ルートにアタッチされたポリシーは、組織内のすべてのアカウントに適用されます。アカウントは、親 OU にアタッチされたポリシーも継承します。複数のポリシーがアカウントに適用されると、 はそれらを 1 つの有効なポリシーに AWS Organizations マージします。アカウントの近くにアタッチされたポリシーは、階層の上位にアタッチされたポリシーよりも優先されます。

`GUARDDUTY_POLICY` ポリシータイプを有効にすると、ポリシーはリージョン GuardDuty 自動有効化設定よりも優先されます。ポリシーがアタッチされているかどうかにかかわらず、自動有効化設定は適用されません。

完全な継承ルールと有効なポリシーの解決方法については、 AWS Organizations 「 ユーザーガイド」の「管理ポリシーの継承について」を参照してください。

ポリシーが GuardDuty アカウント管理に与える影響

GuardDuty 組織ポリシーがアクティブな場合、GuardDuty アカウントの管理方法に次の変更が適用されます。

- GUARDDUTY_POLICY ポリシータイプを有効にすると、ポリシーがアタッチされているかどうかにかかわらず、リージョン GuardDuty 自動有効化設定は適用されません。
- ポリシーが保護プランを管理する場合、その保護プランは、対象アカウントの GuardDuty コンソールまたは API を使用して変更できなくなります。アカウントページには、ポリシー管理として表示されます。マネージド保護プランを変更するには、ポリシーを更新します。ポリシーが管理しない保護プランは、コンソールと API を通じて編集可能です。
- 委任管理者アカウントのみが GuardDuty ポリシーを作成および管理できます。GuardDuty コンソールの組織ポリシーページは、委任された管理者のみが使用できます。

> **Note**
>
> GuardDuty ポリシーをアタッチすると、GuardDuty メンバーとして新しいアカウントを登録し、組織全体の保護プランを有効にできます。GuardDuty とその保護プランを有効にすると、料金が発生します。詳細については、「GuardDuty の料金」を参照してください。

ユースケースの例

次の例は、GuardDuty 組織ポリシーを使用する一般的な方法を示しています。各パターンを実装するポリシー JSON については、 AWS Organizations ユーザーガイドのGuardDuty ポリシー」を参照してください。

- 組織全体で保護ベースラインを有効にします。基本ディテクターと必要な保護プランを有効にするデフォルト設定を使用して、組織ルートにポリシーをアタッチします。後で参加するアカウントを含むすべてのアカウントは、これらの機能を自動的に有効にします。
- 保護は組織単位によって異なります。ベースラインのルートで広範なポリシーを使用し、より具体的なポリシーを個々の OUs にアタッチして、コンプライアンス要件やワークロード要件が異なるアカウントの保護プランを調整します。
- 組織の成長に合わせてカバレッジを維持します。組織に参加するアカウントや対象 OU に移行するアカウントは有効なポリシーを継承するため、ポリシーは手動で有効にすることなく、新しく作成されたアカウントを保護ベースラインに合わせて維持します。

コンソールでのポリシーの作成

GuardDuty コンソールの Organization policies ページで、委任管理者アカウントから GuardDuty ポリシーを作成および管理します。 GuardDuty 最初のポリシーを作成すると、組織のGUARDDUTY_POLICYポリシータイプも自動的に有効になります。

GuardDuty ポリシーを作成するには

1. 委任管理者アカウントを使用して GuardDuty コンソールにサインインします。
2. ナビゲーションペインで、組織ポリシーを選択し、ポリシーの作成を選択します。
3. ステップ 1: ポリシーの詳細で、ポリシー名とオプションの説明を入力し、次へを選択します。
4. ステップ 2: アカウントで、組織階層でポリシーがアタッチされている場所を選択し、次へを選択します。
   - すべての組織単位とアカウントは、組織全体でポリシーをアタッチします。
   - 特定の組織単位とアカウントは、選択したエンティティにアタッチします。
   - 組織単位やアカウントは、ポリシーをアタッチせずに作成しません。
5. ステップ 3: 設定の詳細で、機能の選択で、このポリシーが管理する保護プランを選択します。Enable all または Disable all を使用してクイックスタートし、個々の保護プランを調整します。選択しない保護プランは、このポリシーの影響を受けません。他の保護プランを有効にする場合は、基本的な脅威検出を有効にする必要があります。[次へ] を選択します。
6. (オプション) ステップ 4: リージョンオーバーライドで、リージョンオーバーライドの追加を選択し、1 つ以上のリージョンを選択し、それらのリージョンの設定を設定します。リージョンオーバーライドは、リージョンのデフォルト設定を完全に置き換えるため、そこで管理するすべての保護プランを指定します。[次へ] を選択します。
7. ステップ 5: 確認して作成するには、ポリシーの詳細、アカウント、デフォルト設定、およびリージョンオーバーライドを確認します。生成されたポリシー JSON を表示するには、ポリシードキュメントを展開します。次に、[Create policy (ポリシーの作成)] を選択します。

生成された JSON は、ポリシードキュメントを展開することで、任意のステップで確認または編集できます。ポリシー構造、継承ルール、ポリシーの例については、 AWS Organizations 「 ユーザーガイド」のGuardDuty ポリシー」を参照してください。

ポリシーを適用する前にポリシーの影響を確認する

保護プランを無効にするポリシー変更を保存する前に、その保護プランが現在有効になっているアカウントとリージョンを確認し、変更がオフになる内容を理解してください。GuardDuty ポリシーはリージョンごとに適用されるため、アカウントおよびリージョンごとに現在の有効化状態を確認してください。

> **Warning**
>
> GuardDuty または保護プランを無効にするポリシーは、現在有効になっているすべての対象アカウントでそのポリシーをオフにし、それらのアカウントがコンソールまたは API を介して再びオンにならないようにします。アカウントが依存する保護を意図せずに無効にすることができます。ポリシーを作成または更新する前に、このセクションで説明されているように、変更が影響するアカウントとリージョンを特定し、影響が意図されていることを確認します。

委任管理者は、GuardDuty コンソールのアカウントページ (リージョンを切り替えて各リージョンのカバレッジを確認する) を使用するか、次の手順でプログラムを使用して、組織全体の現在の GuardDuty 有効化を確認できます。確認する各リージョンで API ステップを実行します。

ポリシーが影響を与えるアカウントとリージョンを特定するには

1. 組織内のアカウントを列挙します。`ListAccounts` API オペレーションを使用して、まだ GuardDuty メンバーではないアカウントを含め、組織内のすべてのアカウントを一覧表示します AWS Organizations 。
2. GuardDuty メンバーアカウントを列挙します。GuardDuty `ListMembers` API オペレーションを使用して、既に GuardDuty メンバーアカウントであるアカウントを一覧表示します。
3. 各メンバーアカウントの有効な保護プランを確認します。GuardDuty `GetMemberDetectors` API オペレーションを使用して、各メンバーアカウントのディテクターでどの機能が有効になっているかを確認します。
4. 現在の有効化とポリシーの変更によって無効化される保護プランを比較して、変更が影響するアカウントとリージョンを特定します。

組織に属しているが、まだ GuardDuty メンバーではないアカウントは に表示されず`ListMembers`、メンバーとして追加するまでディテクターステータスを表示できませ

ん。GuardDuty ポリシーがメンバーではないアカウントに適用されると、アカウントは自動的にメンバーとして追加されます。

ポリシーを作成してアタッチしたら、特定のアカウントに適用される最終ポリシーを確認します。有効なポリシーは、アカウントが親エンティティから継承するすべてのポリシーと、アカウントに直接アタッチされたすべてのポリシーの集約です。これを表示するには、GuardDuty ポリシータイプで API オペレーション、またはその AWS CLI または AWS SDK の同等のものを使用します AWS Organizations `DescribeEffectivePolicy`。このオペレーションを呼び出すアクセス許可を持つすべてのアカウントは、委任管理者アカウントを含め、このオペレーションを使用できます。詳細については、「 ユーザーガイド」の「有効な宣言ポリシーの表示」を参照してください。 AWS Organizations

考慮事項と制限事項

GuardDuty 組織ポリシーを計画および管理するときは、次の点に注意してください。

- ポリシーは、 のアカウントにのみ適用されます AWS Organizations。組織ポリシーは、 を通じて組織に属するアカウントでのみ GuardDuty を管理できます AWS Organizations。従来の招待方法にのみ関連付けられているアカウントなど、組織の一部ではないアカウントを管理することはできません。委任 GuardDuty 管理者アカウントまたは各アカウントとは別に、これらのアカウントで GuardDuty を管理します。2 つの関連付け方法の詳細については、「」を参照してくださいAmazon GuardDuty での複数のアカウントの管理。
- ポリシーは、サービスにリンクされたロールが既に存在する場合にのみ、管理アカウントで GuardDuty を有効にできます。GuardDuty サービスにリンクされたロールがそのアカウントに既に存在しない限り、ポリシーは組織の管理アカウントで GuardDuty を有効にすることはできません。ロールがない場合、ポリシーが有効化を宣言しても、GuardDuty は管理アカウントで有効化されません。管理ポリシーに依存する前に、GuardDuty を有効にするなどして、管理アカウントにロールを作成します。
- 委任管理者はリージョンごとに割り当てられ、アカウントを有効にすることは、アカウントの表示とは別のものです。管理アカウントは、各リージョンの GuardDuty 委任管理者を指定します。1 つのリージョンに割り当てられた委任管理者は、委任管理者ではないリージョンを含め、メンバーアカウントの GuardDuty を有効にするポリシーを作成できます。ただし、委任管理者は、管理アカウントがそのリージョンの委任管理者として指定していない限り、リージョン内のこれらのメンバーアカウントのステータスまたは結果を表示できません。リージョンのメンバーアカウントを管理およびモニタリングするには、管理アカウントがそのリージョンの委任管理者を指定する必要があります。指定はリージョンごとに行われるため、管理アカウントは異なるリージョンの委任管理

者として異なるアカウントを指定でき、各委任管理者は、指定されたリージョンでのみメンバーアカウントを管理およびモニタリングします。

- 委任管理者は、 AWS Organizations ポリシーを管理するために委任ポリシーが必要です。委任管理者を指定すると、メンバーアカウントを表示できますが、 AWS Organizations 委任ポリシーがGUARDDUTY_POLICYポリシータイプを管理するアクセス許可も付与しない限り、GuardDuty 組織ポリシーを作成、更新、またはアタッチすることはできません。これらのオペレーションが拒否された場合は、委任ポリシーが委任された管理者にポリシータイプを管理するアクセス許可を付与していることを確認します。
- マネージド保護プランは、コンソールと API では読み取り専用です。ポリシーが保護プランを管理する場合、対象アカウントは GuardDuty コンソールまたは API を使用してそのプランを変更することはできません。マネージド保護プランを変更するには、ポリシーを更新します。ポリシーが管理していない保護プランは、コンソールと API を通じて編集可能です。
- ポリシーは、検出のカスタマイズではなく、有効化のみを管理します。GuardDuty 組織ポリシーは、GuardDuty とその保護プランを有効または無効にします。エンティティリスト、IP アドレスリスト、抑制ルールなどの検出のカスタマイズは管理されません。GuardDuty を有効にした後、GuardDuty コンソールまたは API を使用してこれらを個別に設定します。詳細については、「Customizing threat detection with entity lists and IP address lists」および「GuardDuty の抑制ルール」を参照してください。
- アンマネージドリージョンは変更されず、無効化されません。ポリシーにデフォルト設定に含まれていないリージョンと、リージョンオーバーライドの名前は管理されません。ポリシーはそこで GuardDuty を有効または無効にせず、そのリージョンの既存の設定はそのまま残ります。
- GuardDuty は、有効なすべてのリージョンでグローバルイベントを処理します。GuardDuty は、IAM、、Amazon S3 AWS STS、Amazon CloudFront などのサービスから AWS CloudTrail グローバルサービスイベントをレプリケートし、リソースやワークロードがないリージョンを含め、GuardDuty が有効になっているすべてのリージョンでイベントを処理します。これにより脅威の検出が向上しますが、すべてのリージョンをカバーするデフォルト設定など、ポリシーが GuardDuty を広く有効にすると、GuardDuty はこれらのすべてのリージョンでこれらのグローバルイベントを処理するため、GuardDuty の追加のコストが発生する可能性があります。詳細については、「GuardDuty が AWS CloudTrail グローバルイベントを処理する方法」および「GuardDuty の料金」を参照してください。
- 保護プランを無効にすると、現在有効になっている場所でオフになります。GuardDuty または保護プランを無効にするポリシーは、GuardDuty または保護プランが有効になっているすべての対象アカウントで無効にし、それらのアカウントがコンソールまたは API を通じて再び有効にするのを防ぎます。ポリシーを作成または更新する前に、「」で説明されているように、変更が影響するアカウントとリージョンを特定しますポリシーを適用する前にポリシーの影響を確認する。

- ポリシーの変更は非同期的に適用されます。ポリシーの変更は組織全体に伝達され、すぐに反映されるのではなく、時間の経過とともに調整されます。変更が有効でないと判断する前に、変更を適用する時間を確保します。
- ポリシーをデタッチしても、GuardDuty は無効になりません。エンティティから GuardDuty 組織ポリシーをデタッチしても、エンティティが親エンティティから継承する他のポリシーは引き続き適用され、引き続きターゲットを管理できます。デタッチ後に有効なポリシーが残っていない場合、GuardDuty とその保護プランは現在の状態のままであり、無効になっていません。いずれの場合も、新しいアカウントが参加したり、既存のアカウントが OU に移行したりするなど、組織に対する今後の変更は、継承されたポリシーでまだカバーされていない限り、GuardDuty を自動的に有効にしなくなりました。それ以降の有効化は、手動で行うか、ポリシーを再アタッチするか、継承されたポリシーを通じて行う必要があります。
- ポリシータイプを無効にすると、すべてのポリシー管理が停止し、自動有効化が復元されます。ポリシーでの GuardDuty の無効化、ポリシーのデタッチ、GUARDDUTY_POLICYポリシータイプの無効化は、効果が異なる 3 つの異なるアクションです。組織ルートでGUARDDUTY_POLICYポリシータイプを無効にすると、GuardDuty 組織ポリシーはアカウントを管理しなくなり、GuardDuty は以前にカバーしたアカウントで有効のままになります。ポリシータイプを無効にすると、ポリシータイプを有効にすると抑制されたリージョン GuardDuty 自動有効化設定も復元されるため、自動有効化はそれらの設定に従って再開されます。

招待による GuardDuty アカウントの管理

組織外のアカウントを管理するには、従来の招待による方法を使用できます。この方法を使用する場合、他のアカウントが招待を承諾してメンバーアカウントになると、アカウントは管理者アカウントとして指定されます。

> **Note**
>
> GuardDuty では、GuardDuty の招待 AWS Organizations の代わりに を使用してメンバーアカウントを管理することをお勧めします。詳細については、「AWS Organizationsを使用したアカウントの管理」を参照してください。

アカウントが管理者アカウントでない場合、他のアカウントからの招待を承諾することができます。招待を承諾すると、自分のアカウントはメンバーアカウントになります。 AWS アカウントを GuardDuty 管理者アカウントとメンバーアカウントに同時にすることはできません。

あるアカウントからの招待を承諾する場合、別のアカウントからの招待を承諾することはできません。別のアカウントからの招待を承諾するには、まず既存の管理者アカウントからアカウントの関連付けを解除する必要があります。または、管理者アカウントは、組織からアカウントの関連付けを解除して削除することもできます。

招待によって関連付けられたアカウントは、「」で説明されているように AWS Organizations、によって関連付けられたアカウントと管理者account-to-memberの全体的な関係が似ていますGuardDuty 管理者アカウントとメンバーアカウントの関係について理解する。ただし、招待管理者アカウントのユーザーは、関連付けられたメンバーアカウントに代わって GuardDuty を有効にしたり、 AWS Organizations 組織内の他の非メンバーアカウントを表示したりすることはできません。

> **Important**
>
> この方法で GuardDuty がメンバーアカウントを作成する場合、クロスリージョン内データ転送が発生することがあります。GuardDuty は、メンバーアカウントのメールアドレスを確認するために、米国東部 (バージニア北部) リージョンでのみ動作するメール検証サービスを使用しています。

トピック

- 招待によるアカウントの追加
- GuardDuty 管理者アカウントの単一の組織への統合

招待によるアカウントの追加

GuardDuty が既に有効になっている管理者アカウントでは、メンバーを追加して GuardDuty の使用を開始できます。メンバーを追加したら、GuardDuty に参加するようにメンバーを招待することができ、メンバーは招待に応答することを選択できます。

> **Note**
>
> GuardDuty では、GuardDuty の招待 AWS Organizations の代わりに を使用してメンバーアカウントを管理することをお勧めします。詳細については、「 AWS Organizationsを使用したアカウントの管理」を参照してください。

任意のアクセス方法を選択して、GuardDuty メンバーアカウントを GuardDuty 管理者アカウントとして追加します。

Console

ステップ 1 - アカウントを追加する

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
3. 上部のペインで [招待によってアカウントを追加] を選択します。
4. [メンバーアカウントの追加] ページで、[アカウントの入力] の下で AWS アカウント ID と追加するアカウントに関連付けられている E メールアドレスを入力します。
5. アカウントの詳細を追加する別の行を 1 つずつ追加するには、[別のアカウントを追加] を選択します。[アカウントの詳細を含む.csvファイルをアップロード] を選択して、アカウントを一括で追加することもできます。

   > **Important**
   >
   > 次の例に示すように、.csv ファイルの 1 行目にヘッダー (`Account ID,Email`) を含める必要があります。後続の各行には、1 つの有効な AWS アカウント ID とそれに関連付けられた E メールアドレスが含まれている必要があります。行の形式は、AWS アカウント ID が 1 つだけ含まれていて、関連付けられた E メールアドレスがカンマで区切られている場合に有効です。
   >
   > ```
   > Account ID,Email
   > 555555555555,user@example.com
   > ```

6. すべてのアカウントの詳細を追加したら、[次へ] を選択します。新しく追加したアカウントは、[アカウント] テーブルに表示されます。これらのアカウントのステータスは [招待未送信] になります。追加したアカウントに招待状を送信する方法については、「Step 2 - Invite an account」を参照してください。

ステップ 2: アカウントを招待する

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[Accounts] (アカウント) を選択します。
3. Amazon GuardDuty に招待する 1 つまたは複数のアカウントを選択します。

4. [アクション] ドロップダウンメニューから [招待] を選択します。
5. [GuardDuty への招待] ダイアログ ボックスで、招待メッセージ (オプション) を入力します。

   招待されたアカウントが E メールにアクセスできない場合は、チェックボックスをオンにします。また、招待者の でルートユーザーに E メール通知を送信 AWS アカウント し、招待者の でアラートを生成します AWS Health Dashboard。
6. [Send invitation] (招待の送信) を選択します。招待先が指定の E メールアドレスにアクセスできる場合は、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開いて招待を確認できます。
7. 招待先が招待を承諾すると、[ステータス] 列の値が [招待済み] に変わります。招待を承諾する方法の詳細については、「Step 3 - Accept an invitation」を参照してください。

ステップ 3 - 招待を受け入れる

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   > **Important**
   >
   > メンバーシップ招待を表示または承諾するには、GuardDuty を有効にする必要があります。

2. 以下の手順は、GuardDuty をまだ有効にしていない場合にのみ行ってください。それ以外の場合は、この手順をスキップして次のステップに進んでください。

   まだ GuardDuty を有効にしていない場合は、Amazon GuardDuty ページの [今すぐ始める] を選択します。

   [GuardDuty にようこそ] ページで、[GuardDuty を有効にする] を選択します。
3. アカウントで GuardDuty を有効にしたら、以下の手順でメンバーシップへの招待を承諾します。

   a. ナビゲーションペインで [設定] を選択します。
   b. [アカウント] を選択します。
   c. [アカウント] で、招待元のアカウント所有者を確認してください。[承諾] を選択してメンバーシップへの招待を承諾します。
4. 招待を承諾すると、アカウントが GuardDuty のメンバーアカウントになります。招待を送信した所有者のアカウントが GuardDuty 管理者アカウントになります。招待が承諾されたこと

が管理者アカウントで認識されます。GuardDuty アカウントの [アカウント] テーブルが更新されます。メンバーアカウント ID に対応する [ステータス] 列の値が [有効] に変わります。管理者アカウントの所有者は、ユーザーのアカウントに代わって GuardDuty と保護プランの設定を表示および管理できるようになりました。管理者アカウントは、メンバーアカウントに対して生成された GuardDuty の検出結果を表示および管理することもできるようになりました。

API/CLI

GuardDuty 管理者アカウントを指定し、API オペレーションを通じて招待によって GuardDuty メンバーアカウントを作成または追加できます。GuardDuty の管理者アカウントとメンバーアカウントを指定するために、次の GuardDuty API オペレーションを実行します。

GuardDuty 管理者アカウントとして指定する AWS アカウント の認証情報を使用して、次の手順を実行します。

メンバーアカウントの作成または追加

1. GuardDuty が有効になっている AWS アカウントの認証情報を使用して CreateMembers API オペレーションを実行します。これは、GuardDuty 管理者アカウントにするアカウントです。

   現在の AWS アカウントのディテクター ID と、GuardDuty メンバーにするアカウントのアカウント ID と E メールアドレスを指定する必要があります。この API オペレーションを使用して 1 名以上のメンバーを作成できます。

   AWS コマンドラインツールを使用して、次の CLI コマンドを実行して管理者アカウントを指定することもできます。自身の有効なディテクター ID、アカウント ID、E メールを使用してください。

   アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

   ```
   aws guardduty create-members --detector-id 12abc34d567e8fa901bc2d34e56789f0 --
   account-details AccountId=111122223333,Email=guardduty-member@organization.com
   ```

2. GuardDuty が有効になっている AWS アカウントの認証情報InviteMembersを使用して を実行します。これは、GuardDuty 管理者アカウントにするアカウントです。

現在の AWS アカウントのディテクター ID と、GuardDuty メンバーにするアカウントのアカウント IDs を指定する必要があります。この API オペレーションにより 1 名以上のメンバーを招待できます。

> **Note**
>
> `message` リクエストパラメータを使用してオプションの招待メッセージを指定することもできます。

AWS Command Line Interface を使用して、次のコマンドを実行してメンバーアカウントを指定することもできます。招待するアカウントには、自身の有効なディテクター ID と有効なアカウント ID を使用してください。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty invite-members --detector-id 12abc34d567e8fa901bc2d34e56789f0 --
account-ids 111122223333
```

招待の受け入れ

GuardDuty メンバーアカウントとして指定する各 AWS アカウントの認証情報を使用して次の手順を実行します。

1. GuardDuty メンバー AWS アカウントに招待され、招待を受け入れるアカウントごとに CreateDetector API オペレーションを実行します。

   GuardDuty サービスを使用してディテクターリソースを有効にするかどうかを指定する必要があります。GuardDuty が動作するためには、ディテクターを作成して有効にする必要があります。招待を承諾する前に、まず GuardDuty を有効にする必要があります。

   これを行うには、次の AWS CLI コマンドを使用して コマンドラインツールを使用します。

   ```
   aws guardduty create-detector --enable
   ```

2. その AWS アカウントの認証情報を使用して、メンバーシップの招待を受け入れるアカウントごとに AcceptAdministratorInvitation API オペレーションを実行します。

   メンバー AWS アカウントのこのアカウントのディテクター ID、招待を送信した管理者アカウントのアカウント ID、および承諾する招待の招待 ID を指定する必要があります。管理者アカウントのアカウント ID は、招待メールで見つかります。または、API の ListInvitations オペレーションを使用して検索することもできます。

   次の CLI コマンドを実行して、 AWS コマンドラインツールを使用して招待を受け入れることもできます。ディテクター ID、管理者アカウント ID、招待状 ID は必ず有効なものを使用してください。

   アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

   ```
   aws guardduty accept-invitation --detector-id 12abc34d567e8fa901bc2d34e56789f0
    --administrator-id 444455556666 --invitation-
   id 84b097800250d17d1872b34c4daadcf5
   ```

GuardDuty 管理者アカウントの単一の組織への統合

GuardDuty では、 を介した関連付けを使用して AWS Organizations 、委任 GuardDuty 管理者アカウントのメンバーアカウントを管理することをお勧めします。以下に示すプロセスの例を使用して、組織内の招待によって関連付けられた管理者アカウントとメンバーを、1 つの GuardDuty の委任 GuardDuty 管理者アカウントの下に統合することができます。

> **Note**
>
> GuardDuty では、GuardDuty の招待 AWS Organizations の代わりに を使用してメンバーアカウントを管理することをお勧めします。詳細については、「AWS Organizationsを使用したアカウントの管理」を参照してください。

委任 GuardDuty 管理者アカウントによって既に管理されているアカウント、または委任 GuardDuty 管理者アカウントに関連付けられているアクティブなメンバーアカウントを、別の委任 GuardDuty 管理者アカウントに追加することはできません。各組織は、リージョンごとに 1 つのみ委任

GuardDuty 管理者アカウントを持つことができ、各メンバーアカウントは 1 つの委任 GuardDuty 管理者アカウントのみを持つことができます。

任意のアクセス方法を選択して、GuardDuty 管理者アカウントを単一の委任 GuardDuty 管理者アカウントの下に統合します。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   ログインするには、組織の管理アカウントの認証情報を使用します。

2. GuardDuty を管理するすべてのアカウントは組織の一部である必要があります。組織にアカウントを追加する方法については、「 AWS アカウント を組織に招待する」を参照してください。

3. すべてのメンバーアカウントが、単一の委任 GuardDuty 管理者アカウントとして指定するアカウントに関連付けられていることを確認してください。既存の管理者アカウントにまだ関連付けられているメンバーアカウントの関連付けを解除します。

   次のステップは、メンバーアカウントと既存の管理者アカウントの関連付けを解除するのに役立ちます。

   a. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   b. ログインするには、既存の管理者アカウントの認証情報を使用します。

   c. ナビゲーションペインで、[Accounts] (アカウント) を選択します。

   d. [アカウント] ページで、管理者アカウントとの関連付けを解除する 1 つ以上のアカウントを選択します。

   e. [アクション] を選択してから、[アカウントの関連付けを解除する] を選択します。

   f. [確認] を選択してステップを確定します。

4. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   ログインするには、管理アカウントの認証情報を使用します。

5. ナビゲーションペインで [設定] を選択します。[設定] ページで、組織の委任 GuardDuty 管理者アカウントを指定します。

6. 指定された委任 GuardDuty 管理者アカウントにログインします。

7. 組織からメンバーを追加します。詳細については、「を使用した GuardDuty アカウントの管理 AWS Organizations」を参照してください。

API/CLI

1. GuardDuty を管理するすべてのアカウントは組織の一部である必要があります。組織にアカウントを追加する方法については、「AWS アカウント を組織に招待する」を参照してください。
2. すべてのメンバーアカウントが、単一の委任 GuardDuty 管理者アカウントとして指定するアカウントに関連付けられていることを確認してください。

   a. DisassociateMembers を実行して、既存の管理者アカウントにまだ関連付けられているメンバーアカウントの関連付けを解除します。

   b. または、 AWS Command Line Interface を使用して次のコマンドを実行し、*777777777777* をメンバーアカウントの関連付けを解除する既存の管理者アカウントのディテクター ID に置き換えることができます。*666666666666* を、関連付けを解除するメンバーアカウントの AWS アカウント ID に置き換えます。

   ```
   aws guardduty disassociate-members --detector-id 777777777777 --account-ids 666666666666
   ```

3. EnableOrganizationAdminAccount を実行して、 AWS アカウント を委任 GuardDuty 管理者アカウントとして委任します。

   または、 AWS Command Line Interface を使用して次のコマンドを実行し、委任 GuardDuty 管理者アカウントを委任することもできます。

   ```
   aws guardduty enable-organization-admin-account --admin-account-id 777777777777
   ```

4. 組織からメンバーを追加します。詳細については、「Create or add member member accounts using API」を参照してください。

> **Important**
>
> リージョンサービスである GuardDuty の効果を最大限にするために、各リージョンで委任 GuardDuty 管理者アカウントを指定し、すべてのメンバーアカウントを追加することをお勧めします。

メンバーアカウントの詳細を CSV 形式でエクスポートする際の GuardDuty に関する考慮事項

GuardDuty 管理者アカウントでは、メンバーアカウントの詳細を CSV 形式でエクスポートできます。これらの詳細には、メンバーアカウント ID、名前、タイプ (AWS Organizations または招待によって追加)、GuardDuty の設定ステータス、専用保護プランが含まれます。

[CSV のエクスポート] オプションは、複数のメンバーアカウントの管理方法に基づいて GuardDuty の [アカウント] ページに表示されます。[CSV のエクスポート] オプションを使用すると、特定の保護プランが有効になっているメンバーアカウントを特定できます。

次のリストに、[CSV のエクスポート] を GuardDuty の [アカウント] ページで利用できるかどうかの基準を示します。

- は複数のメンバーアカウントの管理 AWS Organizations にのみ使用し、GuardDuty 組織内のメンバーアカウントの総数は最大 5,000 です。
- AWS Organizations と の両方の招待方法を使用し、GuardDuty 組織のメンバーアカウントの総数は最大 5,000 です。

  このシナリオでは、エクスポートされた CSV には、メンバーアカウントが を通じて追加されたか、招待ベースの方法 AWS Organizations を使用して追加されたかが含まれます。
- 招待ベースの方法のみを使用して複数のメンバーアカウントを管理する場合、[CSV のエクスポート] オプションはありません。

GuardDuty 検出結果タイプ

検出結果は、GuardDuty が AWS アカウントで疑わしいアクティビティまたは悪意のあるアクティビティの兆候を検出したときに生成する通知です。GuardDuty は、GuardDuty を有効にしたアカウントで検出結果を生成します。

新しく追加されたタイプや廃止されたタイプを含む、GuardDuty の検出結果タイプの重要な変更点については、「Amazon GuardDuty のドキュメント履歴」を参照してください。

現在廃止された検出結果タイプの詳細については、「廃止された検出結果タイプ」を参照してください。

GuardDuty EC2 の検出結果タイプ

次の検出結果は Amazon EC2 リソースに固有であり、常に `Instance` のリソースタイプを有しています。検出結果の重大度と詳細は、EC2 インスタンスが不審なアクティビティの対象であるか、不審なアクティビティを実行するアクターであるかを示すリソースロールによって異なります。

ここにリストされている検出結果には、検出結果タイプの生成に使用されるデータソースとモデルが含まれます。データソースとモデルの詳細については、「GuardDuty 基本データソース」を参照してください。

> **注意事項**
>
> - インスタンスが既に終了している場合、または基盤となる API コールが、異なるリージョンの EC2 インスタンスから行われた場合、EC2 検出結果インスタンスの詳細が欠落することがあります。
> - VPC フローログをデータソースとして使用する EC2 の検出結果は、IPv6 トラフィックをサポートしていません。

すべての EC2 の検出結果について、問題のリソースを調べて正常に動作しているかどうかを確認することをお勧めします。アクティビティが認可されると、そのリソースに対する誤検出の通知を防ぐため、抑制ルールや信頼できる IP リストを使用できます。予期しないアクティビティについては、セキュリティのベストプラクティスとしてインスタンスが侵害されていると仮定し、侵害された可能性のある Amazon EC2 インスタンスの修復 で説明されている対策をとることをお勧めします。

トピック

- Backdoor:EC2/C&CActivity.B
- Backdoor:EC2/C&CActivity.B!DNS
- Backdoor:EC2/DenialOfService.Dns
- Backdoor:EC2/DenialOfService.Tcp
- Backdoor:EC2/DenialOfService.Udp
- Backdoor:EC2/DenialOfService.UdpOnTcpPorts
- Backdoor:EC2/DenialOfService.UnusualProtocol
- Backdoor:EC2/Spambot
- Behavior:EC2/NetworkPortUnusual
- Behavior:EC2/TrafficVolumeUnusual
- CryptoCurrency:EC2/BitcoinTool.B
- CryptoCurrency:EC2/BitcoinTool.B!DNS
- DefenseEvasion:EC2/UnusualDNSResolver
- DefenseEvasion:EC2/UnusualDoHActivity
- DefenseEvasion:EC2/UnusualDoTActivity
- Impact:EC2/AbusedDomainRequest.Reputation
- Impact:EC2/BitcoinDomainRequest.Reputation
- Impact:EC2/MaliciousDomainRequest.Reputation
- Impact:EC2/MaliciousDomainRequest.Custom
- Impact:EC2/PortSweep
- Impact:EC2/SuspiciousDomainRequest.Reputation
- Impact:EC2/WinRMBruteForce
- Recon:EC2/PortProbeEMRUnprotectedPort
- Recon:EC2/PortProbeUnprotectedPort
- Recon:EC2/Portscan
- Trojan:EC2/BlackholeTraffic
- Trojan:EC2/BlackholeTraffic!DNS
- Trojan:EC2/DGADomainRequest.B
- Trojan:EC2/DGADomainRequest.C!DNS

- Trojan:EC2/DNSDataExfiltration
- Trojan:EC2/DriveBySourceTraffic!DNS
- Trojan:EC2/DropPoint
- Trojan:EC2/DropPoint!DNS
- Trojan:EC2/PhishingDomainRequest!DNS
- UnauthorizedAccess:EC2/MaliciousIPCaller.Custom
- UnauthorizedAccess:EC2/MetadataDNSRebind
- UnauthorizedAccess:EC2/RDPBruteForce
- UnauthorizedAccess:EC2/SSHBruteForce
- UnauthorizedAccess:EC2/TorClient
- UnauthorizedAccess:EC2/TorRelay

Backdoor:EC2/C&CActivity.B

EC2 インスタンスは、既知の C&C サーバーに関連付けられる IP をクエリしています。

デフォルトの重大度: [High] (高

- データソース: VPC フローログ

この検出結果は、 AWS 環境のリスト化したインスタンスが既知の C&C サーバーに関連付けられた IP をクエリしていることを知らせるものです。リスト化したインスタンスは侵害されている可能性があります。C&C サーバーは、ボットネットのメンバーにコマンドを発行するコンピュータです。

ボットネットとは、一般的なタイプのマルウェアに感染し制御されたインターネットコネクテッドデバイス (PC、サーバー、モバイルデバイス、IoT デバイスなど) のコレクションです。通常、ボットネットは、マルウェアの配布や盗用された情報 (クレジットカード番号など) の収集に使用されます。ボットネットの目的と構造によっては、C&C サーバーから分散型サービス拒否 (DDoS) 攻撃を開始するためのコマンドが発行されることもあります。

> **Note**
>
> クエリされた IP が log4j 関連の場合、関連付けられた検出結果のフィールドには次の値が含まれます。

- service.additionalInfo.threatListName = Amazon
- service.additionalInfo.threatName = Log4j Related

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Backdoor:EC2/C&CActivity.B!DNS

EC2 インスタンスが、既知の C&C サーバーに関連付けられるドメイン名をクエリしています。

デフォルトの重大度: [High] (高

- データソース; DNS ログ

この検出結果は、 AWS 環境のリスト化したインスタンスが既知の C&C サーバーに関連付けられているドメインをクエリしていることを知らせるものです。リスト化したインスタンスは侵害されている可能性があります。C&C サーバーは、ボットネットのメンバーにコマンドを発行するコンピュータです。

ボットネットとは、一般的なタイプのマルウェアに感染し制御されたインターネットコネクテッドデバイス (PC、サーバー、モバイルデバイス、IoT デバイスなど) のコレクションです。通常、ボットネットは、マルウェアの配布や盗用された情報 (クレジットカード番号など) の収集に使用されます。ボットネットの目的と構造によっては、C&C サーバーから分散型サービス拒否 (DDoS) 攻撃を開始するためのコマンドが発行されることもあります。

> **Note**
>
> クエリされたドメイン名が log4j 関連の場合、関連付けられた検出結果のフィールドには次の値が含まれます。
>
> - service.additionalInfo.threatListName = Amazon
> - service.additionalInfo.threatName = Log4j Related

> **Note**
>
> GuardDuty がこの検出結果タイプを生成する方法をテストするには、テスト用のドメイン `guarddutyc2activityb.com` に (Linux については `dig`、Windows については `nslookup` を使用して) インスタンスから DNS リクエストを実行できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Backdoor:EC2/DenialOfService.Dns

EC2 インスタンスが、DNS プロトコルを使用したサービス拒否 (DoS) 攻撃の実行に利用されている可能性があります。

デフォルトの重大度: [High] (高

- データソース: VPC フローログ

この検出結果は、大量のアウトバウンド DNS トラフィックを生成しているリスト化した EC2 インスタンスが AWS 環境にあることを知らせるものです。これは、リスト化されたインスタンスが侵害されており、DNS プロトコルを使用したサービス拒否 (DoS) 攻撃の実行に利用されていることを示しています。

> **Note**
>
> この検出結果では、パブリックにルーティング可能な IP アドレスに対する DoS 攻撃のみ検出しています。このような IP アドレスは、DoS 攻撃の主なターゲットとなっています。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Backdoor:EC2/DenialOfService.Tcp

EC2 インスタンスが TCP プロトコルを使用したサービス拒否 (DoS) 攻撃の実行に利用されている可能性があります。

デフォルトの重大度: [High] (高

- データソース: VPC フローログ

この検出結果は、大量のアウトバウンド TCP トラフィックを生成しているリスト化した EC2 インスタンスが AWS 環境にあることを知らせるものです。これは、インスタンスが侵害されており、TCP プロトコルを使用したサービス拒否 (DoS) 攻撃の実行に利用されていることを示しています。

> **Note**
>
> この検出結果では、パブリックにルーティング可能な IP アドレスに対する DoS 攻撃のみ検出しています。このような IP アドレスは、DoS 攻撃の主なターゲットとなっています。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Backdoor:EC2/DenialOfService.Udp

EC2 インスタンスが UDP プロトコルを使用したサービス拒否 (DoS) 攻撃の実行に利用されている可能性があります。

デフォルトの重大度: [High] (高

- データソース: VPC フローログ

この検出結果は、大量のアウトバウンド UDP トラフィックを生成しているリスト化した EC2 インスタンスが AWS 環境にあることを知らせるものです。これは、リスト化されたインスタンスが侵害されており、UDP プロトコルを使用したサービス拒否 (DoS) 攻撃の実行に利用されていることを示しています。

> **Note**
>
> この検出結果では、パブリックにルーティング可能な IP アドレスに対する DoS 攻撃のみ検出しています。このような IP アドレスは、DoS 攻撃の主なターゲットとなっています。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Backdoor:EC2/DenialOfService.UdpOnTcpPorts

EC2 インスタンスが、TCP ポートで UDP プロトコルを使用したサービス拒否 (DoS) 攻撃の実行に利用されている可能性があります。

デフォルトの重大度: [High] (高

- データソース: VPC フローログ

この検出結果は、TCP 通信に通常使用されるポートを対象とした大量のアウトバウンド UDP トラフィックを生成している EC2 インスタンスが AWS 環境にあることを知らせるものです。これは、リスト化されたインスタンスが侵害されており、TCP ポートで UDP プロトコルを使用したサービス拒否 (DoS) 攻撃の実行に利用されていることを示しています。

> **Note**
>
> この検出結果では、パブリックにルーティング可能な IP アドレスに対する DoS 攻撃のみ検出しています。このような IP アドレスは、DoS 攻撃の主なターゲットとなっています。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Backdoor:EC2/DenialOfService.UnusualProtocol

EC2 インスタンスが、異常なプロトコルを使用したサービス拒否 (DoS) 攻撃の実行に利用されている可能性があります。

デフォルトの重大度: [High] (高

- データソース: VPC フローログ

この検出結果は、ご利用の AWS 環境にリストされている EC2 インスタンスが、Internet Group Management Protocol などの EC2 インスタンスでは通常使用されない異常なプロトコルタイプから大量のアウトバウンドトラフィックを生成していることを知らせるものです。これは、インスタンスが侵害されており、異常なプロトコルを使用したサービス拒否 (DoS) 攻撃の実行に利用されている可能性があることを示しています。この検出結果では、パブリックにルーティング可能な IP アドレスに対する DoS 攻撃のみ検出しています。このような IP アドレスは、DoS 攻撃の主なターゲットとなっています。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Backdoor:EC2/Spambot

EC2 インスタンスがポート 25 でリモートホストと通信して異常な動作を示しています。

デフォルトの重大度: [Medium] (中

- データソース: VPC フローログ

この検出結果は、 AWS 環境のリスト化した EC2 インスタンスがポート 25 でリモートホストと通信していることを知らせるものです。この EC2 インスタンスにはポート 25 での通信履歴が以前にないため、この動作は通常と異なります。従来、ポート 25 はメールサーバーで SMTP 通信のために使用されています。この検出結果は、EC2 インスタンスが侵害されており、スパムの送信に利用されている可能性があることを示しています。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Behavior:EC2/NetworkPortUnusual

EC2 インスタンスが通常と異なるサーバーポートでリモートホストと通信しています。

デフォルトの重大度: [Medium] (中

- データソース: VPC フローログ

この検出結果は、 AWS 環境のリスト化した EC2 インスタンスが、確立されたベースラインから逸脱して動いていることを知らせるものです。この EC2 インスタンスには、このリモートポートでの通信履歴がありません。

> **Note**
>
> EC2 インスタンスがポート 389 またはポート 1389 で通信した場合、関連する検出の重大度は [High] (高) に変更され、検出結果フィールドには次の値が含まれます。
>
> - service.additionalInfo.context = Possible log4j callback

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Behavior:EC2/TrafficVolumeUnusual

EC2 インスタンスがリモートホストに対して通常と異なる大量のネットワークトラフィックを生成しています。

デフォルトの重大度: [Medium] (中

- データソース: VPC フローログ

この検出結果は、 AWS 環境のリスト化した EC2 インスタンスが、確立されたベースラインから逸脱して動いていることを知らせるものです。この EC2 インスタンスでは、このリモートホストに対してこれほど大量のトラフィックを送信した履歴がありません。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

CryptoCurrency:EC2/BitcoinTool.B

EC2 インスタンスが暗号通貨関連のアクティビティに関連付けられている IP アドレスをクエリしています。

デフォルトの重大度: [High] (高

- データソース: VPC フローログ

この検出結果は、 AWS 環境のリスト化した EC2 インスタンスでビットコインやその他の暗号通貨関連アクティビティに紐づけられた IP アドレスがクエリされていることを知らせるものです。ビットコインは、他の通貨、製品、サービスと交換できる国際的な暗号通貨およびデジタル決済システムです。ビットコインはビットコインマイニングの報酬であり、脅威アクターを高度に追及します。

修復のレコメンデーション

暗号通貨の情報を取り出して管理するためこの EC2 インスタンスがを使用する場合、またはこのインスタンスがブロックチェーンのアクティビティに関与している場合は、この検出結果はご利用の環境の想定されるアクティビティを示している可能性があります。ご利用の AWS 環境でこのような状況が発生した場合は、この検出結果に対する抑制ルールを設定することをお勧めします。抑制ルールは、2 つのフィルター条件で構成する必要があります。1 つ目の条件では、[Finding type] (結果タイプ) 属性に `CryptoCurrency:EC2/BitcoinTool.B` という値を使用します。2 つ目のフィルター条件では、ブロックチェーンのアクティビティに関係するインスタンスの [Instance ID] (インスタンス ID) を使用します。抑制ルールの作成の詳細については、「GuardDuty の抑制ルール」を参照してください。

このアクティビティが予期しないものである場合は、インスタンスが侵害されている可能性があります。「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

CryptoCurrency:EC2/BitcoinTool.B!DNS

EC2 インスタンスが暗号通貨関連のアクティビティに関連付けられているドメイン名をクエリしています。

デフォルトの重大度: [High] (高

- データソース; DNS ログ

この検出結果は、 AWS 環境のリスト化した EC2 インスタンスで、ビットコイン、またはその他の暗号通貨関連アクティビティに紐づけたドメインがクエリされていることを知らせるものです。ビットコインは、他の通貨、製品、サービスと交換できる国際的な暗号通貨およびデジタル決済システムです。ビットコインはビットコインマイニングの報酬であり、脅威アクターを高度に追及します。

修復のレコメンデーション

暗号通貨の情報を取り出して管理するためこの EC2 インスタンスがを使用する場合、またはこのインスタンスがブロックチェーンのアクティビティに関与している場合は、この検出結果はご利用の環境の想定されるアクティビティを示している可能性があります。ご利用の AWS 環境でこのような状況が発生した場合は、この検出結果に対する抑制ルールを設定することをお勧めします。抑制ルールは、2 つのフィルター条件で構成する必要があります。1 つ目の条件では、[Finding type] (結果タイプ) 属性に `CryptoCurrency:EC2/BitcoinTool.B!DNS` という値を使用します。2 つ目のフィルター条件では、ブロックチェーンのアクティビティに関係するインスタンスの [Instance ID] (インスタンス ID) を使用します。抑制ルールの作成の詳細については、「GuardDuty の抑制ルール」を参照してください。

このアクティビティが予期しないものである場合は、インスタンスが侵害されている可能性があります。「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

DefenseEvasion:EC2/UnusualDNSResolver

Amazon EC2 インスタンスでは、例外的なパブリック DNS リゾルバーと通信しています。

デフォルトの重大度: [Medium] (中

- データソース: VPC フローログ

この検出結果は、 AWS 環境のリスト化した Amazon EC2 インスタンスが、ベースラインの動作から逸脱して動いていることを知らせるものです。この EC2 インスタンスには、このパブリック DNS リゾルバーに対する最近の通信履歴がありません。GuardDuty コンソールの検出結果詳細パネルの [例外的] フィールドには、クエリを実行した DNS リゾルバーに関する情報が表示されます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

DefenseEvasion:EC2/UnusualDoHActivity

Amazon EC2 インスタンスが、例外的な DNS over HTTPS (DoH) 通信を実行しています。

デフォルトの重大度: [Medium] (中

- データソース: VPC フローログ

この検出結果は、 AWS 環境内のリスト化した Amazon EC2 インスタンスが、確立されたベースラインから逸脱して動いていることを知らせるものです。この EC2 インスタンスには、このパブリック DoH サーバーとの DNS over HTTPS (DoH) 通信の最近の履歴はありません。検出結果の詳細の [例外的] フィールドには、問い合わせた DoH サーバーに関する情報が表示されます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

DefenseEvasion:EC2/UnusualDoTActivity

Amazon EC2 インスタンスが、例外的な DNS over TLS (DoT) 通信を実行しています。

デフォルトの重大度: [Medium] (中

- データソース: VPC フローログ

この検出結果は、 AWS 環境のリスト化した EC2 インスタンスが、確立されたベースラインから逸脱して動いていることを知らせるものです。この EC2 インスタンスには、このパブリック DoT サーバーとの DNS over TLS (DoT) 通信の最近の履歴はありません。検出結果詳細パネルの [例外的] フィールドには、問い合わせた DoT サーバーに関する情報が表示されます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Impact:EC2/AbusedDomainRequest.Reputation

EC2 インスタンスが、既知の悪用されたドメインに関連付けられた評価の低いドメイン名をクエリしています。

デフォルトの重大度: [Medium] (中

- データソース; DNS ログ

この検出結果は、 AWS 環境内にリストされている Amazon EC2 インスタンスが、既知の悪用されたドメインまたは IP アドレスに関連付けられたレピュテーションの低いドメイン名をクエリしていることを知らせるものです。悪用したドメインの例としては、動的 DNS プロバイダーだけでなく、無料のサブドメイン登録を提供する最上位のドメイン名 (TLD) と第 2 位のドメイン名 (2LD) があります。脅威アクターは、無料または低コストでドメインを登録するこれらのサービスを使用する傾向があります。このカテゴリの評価の低いドメインは、レジストラのパーキング IP アドレスを決定する有効期限切れドメインであり、アクティブになっていない可能性があります。パーキング IP は、レジストラがどのサービスにもリンクされていないドメインのトラフィックを管理する場所です。脅威アクターが一般的にこれらのレジストラのサービスまたは C&C のサービス、マルウェア配布に使用するため、リストされた Amazon EC2 インスタンスは侵害される可能性があります。

[Low] (低) のレピュテーションドメインは、レピュテーションスコアモデルに基づいています。このモデルは、ドメインの特徴を評価およびランク付けし、それが悪意のあるものである可能性を判断します。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Impact:EC2/BitcoinDomainRequest.Reputation

EC2 インスタンスが、暗号通貨関連のアクティビティに関連付けられている評判の低いドメイン名をクエリしています。

デフォルトの重大度: [High] (高

- データソース; DNS ログ

この検出結果は、 AWS 環境のリスト化した Amazon EC2 インスタンスで、ビットコイン、またはその他の暗号通貨関連アクティビティに紐づけた評判の低いドメイン名がクエリされていることを知らせるものです。ビットコインは、他の通貨、製品、サービスと交換できる国際的な暗号通貨およびデジタル決済システムです。ビットコインはビットコインマイニングの報酬であり、脅威アクターを高度に追及します。

[Low] (低) のレピュテーションドメインは、レピュテーションスコアモデルに基づいています。このモデルは、ドメインの特徴を評価およびランク付けし、それが悪意のあるものである可能性を判断します。

修復のレコメンデーション

暗号通貨の情報を取り出して管理するためこの EC2 インスタンスがを使用する場合、またはこのインスタンスがブロックチェーンのアクティビティに関与している場合は、この検出結果はご利用の環境の想定されるアクティビティを示している可能性があります。ご利用の AWS 環境でこのような状況が発生した場合は、この検出結果に対する抑制ルールを設定することをお勧めします。抑制ルールは、2 つのフィルター条件で構成する必要があります。1 つ目の条件では、[Finding type] (結果タイプ) 属性に `Impact:EC2/BitcoinDomainRequest.Reputation` という値を使用します。2 つ目のフィルター条件では、ブロックチェーンのアクティビティに関係するインスタンスの [Instance ID] (インスタンス ID) を使用します。抑制ルールの作成の詳細については、「GuardDuty の抑制ルール」を参照してください。

このアクティビティが予期しないものである場合は、インスタンスが侵害されている可能性があります。「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Impact:EC2/MaliciousDomainRequest.Reputation

EC2 インスタンスが、悪意のある既知のドメインに関連付けられた評判の低いドメインをクエリしています。

デフォルトの重大度: [High] (高

- データソース; DNS ログ

この検出結果は、 AWS 環境内にリストされている Amazon EC2 インスタンスが、悪意のある既知のドメインまたは IP アドレスに関連付けられたレピュテーションの低いドメイン名をクエリしていることを知らせるものです。例えば、ドメインを既知のシンクホール IP アドレスに関連付けることができます。シンクホールドメインは、以前に脅威アクターに制御されたドメインであり、ドメインへのリクエストは、インスタンスが侵害されていることを示している場合があります。これらのドメインは、悪意のある既知のキャンペーンやドメイン生成アルゴリズムと相関している可能性もあります。

[Low] (低) のレピュテーションドメインは、レピュテーションスコアモデルに基づいています。このモデルは、ドメインの特徴を評価およびランク付けし、それが悪意のあるものである可能性を判断します。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Impact:EC2/MaliciousDomainRequest.Custom

EC2 インスタンスが、カスタム脅威エンティティリスト上のドメインをクエリしています。

デフォルトの重大度: [Medium] (中

- データソース; DNS ログ

この検出結果は、 AWS 環境内にリストされている Amazon EC2 インスタンスが、アップロードしてアクティブ化した脅威エンティティリストに含まれるドメイン名をクエリしていることを知らせるものです。GuardDuty の場合、脅威エンティティリストは悪意のある既知の ドメイン名と IP アドレ

スで構成されます。GuardDuty は、アップロードされた脅威エンティティリストに関連付けられたアクティビティに基づいて検出結果を生成します。脅威エンティティリストの名前は、検出結果の詳細で確認できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Impact:EC2/PortSweep

EC2 インスタンスが、多数の IP アドレスのポートを調査しています。

デフォルトの重大度: [High] (高

- データソース: VPC フローログ

この検出結果は、 AWS 環境内のリストされた EC2 インスタンスが、多数のパブリックにルーティング可能な IP アドレスのポートを調べていることを知らせるものです。このアクティビティタイプは、通常脆弱性ホストを見つけて悪用するのに使われます。GuardDuty コンソールの検出結果詳細パネルには、最新のリモート IP アドレスのみが表示されます

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Impact:EC2/SuspiciousDomainRequest.Reputation

EC2 インスタンスが、年齢や低人気により、本質的に疑わしい、低評判のドメイン名をクエリしています。

デフォルトの重大度: [Low] (低

- データソース; DNS ログ

この検出結果は、 AWS 環境のリスト化した Amazon EC2 インスタンスが悪意があると疑われたり、過去に悪意のあるドメインだったため評判の低いドメイン名をクエリしていることを知らせるも

のですが、当社の評判モデルは、既知の脅威と明確に関連付けることができませんでした。これらのドメインは通常、新たに観察されるか、または少量のトラフィックを受信します。

[Low] (低) のレピュテーションドメインは、レピュテーションスコアモデルに基づいています。このモデルは、ドメインの特徴を評価およびランク付けし、それが悪意のあるものである可能性を判断します。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Impact:EC2/WinRMBruteForce

EC2 インスタンスがアウトバウンドの Windows リモート管理総当たり攻撃を実行しています。

デフォルトの重大度: [Low] (低)*

> **Note**
>
> EC2 インスタンスが総当たり攻撃の対象である場合、この検出結果の重大度は「低」です。EC2 インスタンスが総当たり攻撃の動作主体である場合、この検出結果の重大度は「高」です。

- データソース: VPC フローログ

この検出結果は、 AWS 環境のリスト化した EC2 インスタンスが、Windows ベースのシステム上の Windows リモート管理サービスへのアクセスを目的とした Windows リモート管理 (WinRM) 総当たり攻撃を実行していることを知らせるものです。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Recon:EC2/PortProbeEMRUnprotectedPort

EC2 インスタンスの保護されていない EMR 関連のポートを悪意のある既知のホストが探しています。

デフォルトの重大度: [High] (高

- データソース: VPC フローログ

この検出結果は、 AWS 環境内のクラスターの一部であるリストされた EC2 インスタンスの EMR 関連の機密ポートが、セキュリティグループ、アクセスコントロールリスト (ACL)、または Linux IPTables などのホスト上のファイアウォールによってブロックされていないことを知らせるものです。この検出結果は、インターネットの既知のスキャナーがこのポートを積極的に調査していることも知らせます。ポート 8088 (YARN ウェブ UI ポート) など、この検出結果をトリガーできるポートは、リモートコード実行で使用される可能性があります。

修復のレコメンデーション

インターネットからクラスター上のポートへのオープンアクセスをブロックし、それらのポートへのアクセスを必要とする特定の IP アドレスのみにアクセスを制限する必要があります。詳細については、「Security Groups for EMR Clusters」(EMR クラスターのセキュリティグループ) を参照してください。

Recon:EC2/PortProbeUnprotectedPort

EC2 インスタンスの保護されていないポートを悪意のある既知のホストが探しています。

デフォルトの重大度: [Low] (低)*

> **Note**
>
> この検出結果のデフォルトの重大度は [Low] (低) です。ただし、調査対象のポートが Elasticsearch (9200 または 9300) で使用されている場合、検出結果の重大度は [High] (高) になります。

- データソース: VPC フローログ

この検出結果は、 AWS 環境のリスト化した EC2 インスタンスのポートが、セキュリティグループ、アクセスコントロールリスト (ACL)、Linux IPTables など、ホスト上のファイアウォールでブロックされておらず、インターネットの既知のスキャナーが積極的に調査していることを知らせるものです。

識別された保護されていないポートが 22 または 3389 であり、それらのポートを使用してインスタンスに接続している場合は、それらのポートへのアクセスを自社ネットワークの IP アドレス空間の IP アドレスのみに許可することで公開を制限することができます。Linux でポート 22 へのアクセスを制限するには、「Linux インスタンス用の受信トラフィックの認可」を参照してください。Windows でポート 3389 へのアクセスを制限するには、「Windows インスタンス用の受信トラフィックの認可」を参照してください。

GuardDuty は、ポート 443 と 80 に対してこの検出結果を生成しません。

修復のレコメンデーション

インスタンスがウェブサーバーをホストしている場合など、インスタンスが意図的に公開される場合があります。ご使用の AWS 環境でその場合は、この検出結果の抑制ルールを設定することをお勧めします。抑制ルールは、2 つのフィルター条件で構成する必要があります。1 つ目の条件では、[Finding type] (結果タイプ) 属性に `Recon:EC2/PortProbeUnprotectedPort` という値を使用します。2 番目のフィルター条件は、要塞ホストとして機能する 1 つ以上のインスタンスと一致する必要があります。これらのツールをホストするインスタンスで識別可能な条件に応じて、[Instance image ID] (インスタンスイメージ ID) 属性または [Tag] (タグ) 値の属性のいずれかを使用できます。抑制ルールの作成の詳細については、「GuardDuty の抑制ルール」を参照してください。

このアクティビティが予期しないものである場合は、インスタンスが侵害されている可能性があります。「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Recon:EC2/Portscan

EC2 インスタンスがリモートホストにアウトバウンドポートスキャンを実行しています。

デフォルトの重大度: [Medium] (中

- データソース: VPC フローログ

この検出結果は、 AWS 環境のリスト化した EC2 インスタンスが短時間にに複数のポートに接続しようとして、ポートスキャン攻撃を行っている可能性があることを知らせるものです。ポートスキャ

ン攻撃の目的は、オープンポートを見つけ、マシンで実行されているサービスを発見してそのオペレーティングシステムを特定することです。

修復のレコメンデーション

この検出結果は、ご利用の環境の EC2 インスタンスに脆弱性評価アプリケーションがデプロイされており、それらのアプリケーションがポートをスキャンして、誤ってオープンポート設定になっているものをアラートするので、誤検出される可能性があります。ご使用の AWS 環境でその場合は、この検出結果の抑制ルールを設定することをお勧めします。抑制ルールは、2 つのフィルター条件で構成する必要があります。1 つ目の条件では、[Finding type] (結果タイプ) 属性に `Recon:EC2/Portscan` という値を使用します。2 番目のフィルター条件は、これらの脆弱性評価ツールをホストする 1 つ以上のインスタンスと一致する必要があります。これらのツールをホストするインスタンスで識別可能な条件に応じて、[Instance image ID] (インスタンスイメージ ID) 属性または [Tag] (タグ) 値の属性のいずれかを使用できます。抑制ルールの作成の詳細については、「GuardDuty の抑制ルール」を参照してください。

このアクティビティが予期しないものである場合は、インスタンスが侵害されている可能性があります。「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Trojan:EC2/BlackholeTraffic

EC2 インスタンスが既知のブラックホールであるリモートホストの IP アドレスに通信しようとしています。

デフォルトの重大度: [Medium] (中

- データソース: VPC フローログ

この検出結果は、 AWS 環境内のリストされた EC2 インスタンスがブラックホール (またはシンクホール) の IP アドレスと通信しようとしているために侵害されている可能性があることを知らせるものです。ブラックホールとは、送受信トラフィックが密かに破棄されるネットワークの場所を指し、意図した受信者にデータが届いていないことは送信元に知らされません。ブラックホール IP アドレスは、稼働していないホストマシンやホストが割り当てられていないアドレスを指定します。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Trojan:EC2/BlackholeTraffic!DNS

EC2 インスタンスがブラックホールの IP アドレスにリダイレクトされるドメイン名をクエリしています。

デフォルトの重大度: [Medium] (中

- データソース; DNS ログ

この検出結果は、ブラックホール IP アドレスにリダイレクトされているドメイン名をクエリしているため、 AWS 環境内のリストされている EC2 インスタンスが侵害されている可能性があることを知らせるものです。ブラックホールとは、送受信トラフィックが密かに破棄されるネットワークの場所を指し、意図した受信者にデータが届いていないことは送信元に知らされません。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Trojan:EC2/DGADomainRequest.B

EC2 インスタンスがアルゴリズムを使用して生成されたドメインをクエリしています。このようなドメインは、マルウェアによって悪用されることが多く、EC2 インスタンスが侵害されている場合があります。

デフォルトの重大度: [High] (高

- データソース; DNS ログ

この検出結果は、 AWS 環境のリスト化した EC2 インスタンスがドメイン生成アルゴリズム (DGA) のドメインをクエリしようとしていることを知らせるものです。EC2 インスタンスは侵害されている可能性があります。

DGA は、大量のドメイン名を定期的に生成してコマンドアンドコントロール (C&C) サーバーとのランデブーポイントとするために使用されます。C&C サーバーは、ボットネットのメンバーにコマンドを発行するコンピュータであり、一般的なタイプのマルウェアに感染して制御されたインターネットのコネクテッドデバイスのコレクションです。ランデブーポイントの候補数が多いと、感染された

コンピュータは毎日これらのドメイン名の一部にアクセスしてアップデートやコマンドを受け取ろうとするため、ボットネットを効果的にシャットダウンすることが困難となります。

> **Note**
>
> この検出結果は、アドバンストな経験則を使用したドメイン名分析に基づいており、脅威インテリジェンスフィードでは検出されない新しい DGA ドメインを識別する可能性があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Trojan:EC2/DGADomainRequest.C!DNS

EC2 インスタンスがアルゴリズムを使用して生成されたドメインをクエリしています。このようなドメインは、マルウェアによって悪用されることが多く、EC2 インスタンスが侵害されている場合があります。

デフォルトの重大度: [High] (高

- データソース; DNS ログ

この検出結果は、 AWS 環境のリスト化した EC2 インスタンスがドメイン生成アルゴリズム (DGA) のドメインをクエリしようとしていることを知らせるものです。EC2 インスタンスは侵害されている可能性があります。

DGA は、大量のドメイン名を定期的に生成してコマンドアンドコントロール (C&C) サーバーとのランデブーポイントとするために使用されます。C&C サーバーは、ボットネットのメンバーにコマンドを発行するコンピュータであり、一般的なタイプのマルウェアに感染して制御されたインターネットのコネクテッドデバイスのコレクションです。ランデブーポイントの候補数が多いと、感染されたコンピュータは毎日これらのドメイン名の一部にアクセスしてアップデートやコマンドを受け取ろうとするため、ボットネットを効果的にシャットダウンすることが困難となります。

> **Note**
>
> この検出結果は、GuardDuty の脅威インテリジェンスフィードの既知の DGA ドメインに基づいています。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Trojan:EC2/DNSDataExfiltration

EC2 インスタンスが DNS クエリを使用してデータを密かに抽出しようとしています。

デフォルトの重大度: [High] (高

- データソース; DNS ログ

この検出結果は、 AWS 環境のリスト化した EC2 インスタンスが、アウトバウンドデータ転送用の DNS クエリを使用しているマルウェアであることを知らせるものです。このタイプのデータ転送は、侵害されたインスタンスを示し、データの漏洩につながる可能性があります 通常、DNS トラフィックはファイアウォールでブロックされません。例えば、侵害された EC2 インスタンスのマルウェアは、データ (クレジットカード番号など) を DNS クエリ内にエンコードし、それを攻撃者が制御するリモート DNS サーバーに送信できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Trojan:EC2/DriveBySourceTraffic!DNS

EC2 インスタンスがドライブバイダウンロード攻撃の既知の攻撃元であるリモートホストのドメイン名をクエリしています。

デフォルトの重大度: [High] (高

- データソース; DNS ログ

この検出結果は、自動ダウンロード攻撃の既知のソースであるリモートホストのドメイン名をクエリしているため、リスト化した AWS 環境の EC2 インスタンスが侵害された可能性があることを知らせるものです。これらは、インターネットから意図せずにダウンロードされるコンピュータソフトウェアであり、ウイルス、スパイウェア、マルウェアの自動インストールをトリガーする場合があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Trojan:EC2/DropPoint

EC2 インスタンスが、マルウェアによって収集された認証情報やその他の盗難されたデータを保持していることが認識されているリモートホストの IP アドレスに通信しようとしています。

デフォルトの重大度: [Medium] (中

- データソース: VPC フローログ

この検出結果は、 AWS 環境の EC2 インスタンスが、マルウェアによってキャプチャされた認証情報やその他の盗まれたデータを保持することが知られているリモートホストの IP アドレスと通信しようとしていることを知らせるものです。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Trojan:EC2/DropPoint!DNS

EC2 インスタンスが、マルウェアによって収集された認証情報やその他の盗難されたデータを保持していることが認識されているリモートホストのドメイン名をクエリしています。

デフォルトの重大度: [Medium] (中

- データソース; DNS ログ

この検出結果は、 AWS 環境内の EC2 インスタンスが、マルウェアによってキャプチャされた認証情報やその他の盗まれたデータを保持することが知られているリモートホストのドメイン名をクエリしていることを知らせるものです。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Trojan:EC2/PhishingDomainRequest!DNS

EC2 インスタンスがフィッシング攻撃に関与しているドメインをクエリしています。EC2 インスタンスは侵害されている可能性があります。

デフォルトの重大度: [High] (高

- データソース; DNS ログ

この検出結果は、 AWS 環境の EC2 インスタンスがフィッシング攻撃に関与しているドメインをクエリしようとしていることを知らせるものです。フィッシングドメインは、個人を特定できる情報、銀行やクレジットカードの詳細情報とパスワードなど、ユーザーが機密データを提供するように仕向ける、正当な機関になりすました人物によって設定されます。EC2 インスタンスがフィッシングウェブサイトに保存されている機密データを検索しようとしたり、フィッシングウェブサイトを設定しようとしたりする可能性があります。EC2 インスタンスは侵害されている可能性があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

UnauthorizedAccess:EC2/MaliciousIPCaller.Custom

EC2 インスタンスがカスタム脅威リストの IP アドレスに接続しています。

デフォルトの重大度: [Medium] (中

- データソース: VPC フローログ

この検出結果は、 AWS 環境内の EC2 インスタンスが、アップロードした脅威リストに含まれている IP アドレスと通信していることを知らせるものです。GuardDuty の場合、脅威リストは悪意のある既知の IP アドレスで構成されます。GuardDuty では、アップロード済みの脅威リストに基づいて検出結果を生成します。この検出結果を生成するために使用された脅威リストは、検出結果の詳細に表示されます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

UnauthorizedAccess:EC2/MetadataDNSRebind

EC2 インスタンスが、インスタンスメタデータサービスに解決する DNS 検索を実行しています。

デフォルトの重大度: [High] (高

- データソース; DNS ログ

この検出結果は、 AWS 環境の EC2 インスタンスが EC2 メタデータ IP アドレス (169.254.169.254) に解決されるドメインをクエリしていることを知らせるものです。この種類の DNS クエリは、インスタンスが DNS リバイディング技術の対象であることを示している可能性があります。この手法は、インスタンスに関連付けらた IAM 認証情報など、EC2 インスタンスからメタデータを取得するために使用できます。

DNS リバインディングには、EC2 インスタンスで実行されているアプリケーションをだまして URL から返されるデータをロードすることが含まれます。URL のドメイン名は EC2 メタデータ IP アドレス (169.254.169.254) に解決されます。これにより、アプリケーションは EC2 メタデータにアクセスし、攻撃者がそのメタデータを使用できるようにする可能性があります。

EC2 インスタンスが URL の追加を許可する脆弱なアプリケーションを実行している場合や、EC2 インスタンスで実行されているウェブブラウザで、誰かが URL にアクセスする場合のみ、DNS リバインディングを使用して EC2 メタデータにアクセスできます。

修復のレコメンデーション

この検出結果に応じて、EC2 インスタンスで実行されている脆弱性アプリケーションがあるか、誰が検出結果で識別したドメインへアクセスするためブラウザを使用しているかを評価する必要があります。根本的な原因が脆弱なアプリケーションである場合は、脆弱性を修復する必要があります。ユーザーが識別したドメインを閲覧した場合、ドメインをブロックするか、ユーザーがそのドメインにアクセスできないようにします。この検出結果が上記のいずれかのケースに関連していると判断した場合は、EC2 インスタンスに関連付けられたセッションを取り消す必要があります。

一部の AWS お客様は、メタデータ IP アドレスを信頼できる DNS サーバーのドメイン名に意図的にマッピングします。ご利用の環境でこのような状況が発生した場合は、この検出結果に対する抑制ルールを設定することをお勧めします。抑制ルールは、2 つのフィルター条件で構成する必要があります。1 つ目の条件では、[Finding type] (結果タイプ) 属性に `UnauthorizedAccess:EC2/MetaDataDNSRebind` という値を使用します。2 つ目のフィルター条件では、DNS リクエストのドメインを使用します。値はメタデータの IP アドレス (169.254.169.254) にマッピングしたドメインと一致する必要があります。抑制ルールの作成の詳細については、「GuardDuty の抑制ルール」を参照してください。

UnauthorizedAccess:EC2/RDPBruteForce

EC2 インスタンスが RDP ブルートフォース攻撃に巻き込まれています。

デフォルトの重大度: [Low] (低)*

> **Note**
>
> EC2 インスタンスが総当たり攻撃の対象である場合、この検出結果の重大度は「低」です。EC2 インスタンスが総当たり攻撃の動作主体である場合、この検出結果の重大度は「高」です。

- データソース: VPC フローログ

この検出結果は、 AWS 環境内の EC2 インスタンスが、Windows ベースのシステムで RDP サービスへのパスワードを取得することを目的としたブルートフォース攻撃に関与していたことを知らせるものです。これは、 AWS リソースへの未承認のアクセスを示している場合があります。

修復のレコメンデーション

インスタンスのリソースロールが `ACTOR` の場合は、インスタンスが RDP 総当たり攻撃の実行に利用されたことを示しています。このインスタンスに `Target` としてリストされた IP アドレスと通信する合理的な理由がない限りは、インスタンスが侵害されたと仮定し、侵害された可能性のある Amazon EC2 インスタンスの修復 でリストされたアクションを取ることをお勧めします。

インスタンスの [Resource Role] (リソースロール) が `TARGET` である場合は、セキュリティグループ、ACL、ファイアウォールのいずれかを使用して RDP ポートを信頼できる IP のみ保護することで、この検出結果を修復できます。詳細については、「EC2 インスタンスの保護のヒント (Linux)」を参照してください。

UnauthorizedAccess:EC2/SSHBruteForce

EC2 インスタンスが SSH ブルートフォース攻撃に巻き込まれています。

デフォルトの重大度: [Low] (低)*

> **Note**
>
> 総当たり攻撃が EC2 インスタンスのいずれかを標的にしている場合、この検出結果の重大度は「低」です。EC2 インスタンスが総当たり攻撃の動作主体である場合、この検出結果の重大度は「高」です。

- データソース: VPC フローログ

この検出結果は、 AWS 環境内の EC2 インスタンスが、Linux ベースのシステムで SSH サービスへのパスワードを取得することを目的としたブルートフォース攻撃に関与していたことを知らせるものです。これは、 AWS リソースへの未承認のアクセスを示している場合があります。

> **Note**
>
> この検出結果は、ポート 22 のモニタリングトラフィックを通じてのみ生成されます。SSH サービスが他のポートを使用するように設定されている場合には、この検出結果は生成されません。

修復のレコメンデーション

ブルートフォース試行のターゲットが踏み台ホストである場合、これは AWS 環境に対して予想される動作を表す可能性があります。このような状況が発生した場合は、この検出結果に対する抑制ルールを設定することをお勧めします。抑制ルールは、2 つのフィルター条件で構成する必要があります。1 つ目の条件では、[Finding type] (結果タイプ) 属性に `UnauthorizedAccess:EC2/SSHBruteForce` という値を使用します。2 番目のフィルター条件は、要塞ホストとして機能する 1 つ以上のインスタンスと一致する必要があります。これらのツールをホストするインスタンスで識別可能な条件に応じて、[Instance image ID] (インスタンスイメージ ID) 属性または [Tag] (タグ) 値の属性のいずれかを使用できます。抑制ルールの作成の詳細については、「GuardDuty の抑制ルール」を参照してください。

このアクティビティがご利用の環境で想定外であり、インスタンスの [Resource Role] (リソースロール) が `TARGET` である場合は、セキュリティグループ、ACL、ファイアウォールのいずれかを使用して SSH ポートを信頼できる IP のみ保護することで、この検出結果を修復できます。詳細については、「EC2 インスタンスの保護のヒント (Linux)」を参照してください。

インスタンスの [Resource Role] (リソースロール) が `ACTOR` の場合は、インスタンスが SSH 総当たり攻撃の実行に利用されたことを示しています。このインスタンスに `Target` としてリストされた IP アドレスと通信する合理的な理由がない限りは、インスタンスが侵害されたと仮定し、侵害された可能性のある Amazon EC2 インスタンスの修復 でリストされたアクションを取ることをお勧めします。

UnauthorizedAccess:EC2/TorClient

EC2 インスタンスが Tor Guard または Authority ノードに接続しています。

デフォルトの重大度: [High] (高

- データソース: VPC フローログ

この検出結果は、 AWS 環境の EC2 インスタンスが Tor Guard または Authority ノードに接続していることを知らせるものです。Tor は、匿名通信を有効にするためのソフトウェアです。Tor Guards および Authority ノードは、Tor ネットワークへの初期ゲートウェイとして動作します。このトラフィックは、この EC2 インスタンスが侵害され、Tor ネットワーク上のクライアントとして動作していることを示している場合があります。この検出結果は、攻撃者の真のアイデンティティを隠す目的で AWS リソースへの不正アクセスを示している可能性があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

UnauthorizedAccess:EC2/TorRelay

EC2 インスタンスが Tor リレーとして Tor ネットワークに接続しています。

デフォルトの重大度: [High] (高

- データソース: VPC フローログ

この検出結果は、 AWS 環境の EC2 インスタンスが Tor リレーとして動作していることを示す方法で Tor ネットワークに接続していることを知らせるものです。Tor は、匿名通信を有効にするためのソフトウェアです。ある Tor リレーから別の Tor リレーにクライアントの不正なトラフィックを転送することで、通信の匿名性を高めます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

GuardDuty IAM 検出結果タイプ

次の検出結果は、IAM エンティティとアクセスキーに特有であり、常に `AccessKey` の [Resource Type] (リソースタイプ) です。検出結果の重大度と詳細は、検出結果タイプによって異なります。

ここにリストされている検出結果には、検出結果タイプの生成に使用されるデータソースとモデルが含まれます。詳細については、「GuardDuty 基本データソース」を参照してください。

すべての IAM 関連の検出結果について、問題のエンティティを検証し、その許可が最小特権のベストプラクティスに従っていることを確認することをお勧めします。このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。検出結果の修正についての詳細は、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

トピック

- CredentialAccess:IAMUser/AnomalousBehavior
- CredentialAccess:IAMUser/CompromisedCredentials
- DefenseEvasion:IAMUser/AnomalousBehavior
- DefenseEvasion:IAMUser/BedrockLoggingDisabled
- Discovery:IAMUser/AnomalousBehavior
- Exfiltration:IAMUser/AnomalousBehavior
- Impact:IAMUser/AnomalousBehavior
- InitialAccess:IAMUser/AnomalousBehavior
- PenTest:IAMUser/KaliLinux
- PenTest:IAMUser/ParrotLinux
- PenTest:IAMUser/PentooLinux
- Persistence:IAMUser/AnomalousBehavior
- Policy:IAMUser/RootCredentialUsage
- Policy:IAMUser/ShortTermRootCredentialUsage
- PrivilegeEscalation:IAMUser/AnomalousBehavior
- Recon:IAMUser/MaliciousIPCaller
- Recon:IAMUser/MaliciousIPCaller.Custom
- Recon:IAMUser/TorIPCaller
- Stealth:IAMUser/CloudTrailLoggingDisabled
- Stealth:IAMUser/PasswordPolicyChange
- UnauthorizedAccess:IAMUser/ConsoleLoginSuccess.B
- UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.InsideAWS
- UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.OutsideAWS
- UnauthorizedAccess:IAMUser/MaliciousIPCaller
- UnauthorizedAccess:IAMUser/MaliciousIPCaller.Custom
- UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.InsideAWS

- UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.OutsideAWS
- UnauthorizedAccess:IAMUser/TorIPCaller

CredentialAccess:IAMUser/AnomalousBehavior

AWS 環境へのアクセスを取得するために使用される API が異常な方法で呼び出されました。

デフォルトの重大度: [Medium] (中

- データソース: CloudTrail 管理イベント

この検出結果は、アカウント内で異常な API リクエストが観察されたことを知らせるものです。この検出結果には、単一の API、または単一のユーザーアイデンティティで近似の一連の関連 API リクエストが含まれる場合があります。観察される API は、攻撃者がユーザーの環境のパスワード、ユーザー名、およびアクセスキーを収集しようすると、攻撃の認証情報アクセスステージに一般的に関連しています。このカテゴリの API は、`GetPasswordData`、`GetSecretValue`、`BatchGetSecretValue`、および `GenerateDbAuthToken` です。

この API リクエストは、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。機械学習モデルは、アカウント内のすべての API リクエストを評価し、攻撃者が使用するテクニックに関連する異常なイベントを特定します。機械学習モデルは、リクエストを行ったユーザー、リクエストが行われた場所、リクエストされた特定の API など、API リクエストのさまざまな要因を追跡します。リクエストしたユーザーアイデンティティにおいて API リクエストのどの要因が異常なのかという詳細については、「検出結果の詳細 」を参照してください。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

CredentialAccess:IAMUser/CompromisedCredentials

Amazon 脅威インテリジェンスにより、IAM アクセスキーが侵害された可能性があると識別されました。

デフォルトの重大度: [High] (高

- 機能: Foundational Data Source Protection に付属

詳細な説明:

この検出結果は、 AWS アカウントに関連付けられた IAM アクセスキーが Amazon 脅威インテリジェンスによって侵害された可能性があると特定されたことを知らせるものです。次に、侵害された認証情報を使用して、 AWS 環境で API オペレーションを呼び出しました。侵害された認証情報を使用して行われた API コールのリスト、各コールの回数とタイムスタンプ、関連するアクセスキー、ソース IP アドレスが検出結果の詳細に含まれています。

この検出結果の認証情報の侵害は、Amazon 脅威インテリジェンスによって識別されます。 は、使用パターンを通じて侵害された可能性のある認証情報 AWS を監視し、そのような認証情報が使用されていることがわかったときにこの検出結果を生成します。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

DefenseEvasion:IAMUser/AnomalousBehavior

防御対策を回避するために使用された API が異常な方法で呼び出されました。

デフォルトの重大度: [Medium] (中

- データソース: CloudTrail 管理イベント

この検出結果は、アカウント内で異常な API リクエストが観察されたことを知らせるものです。この検出結果には、単一の API、または単一のユーザーアイデンティティで近似の一連の関連 API リクエストが含まれる場合があります。観察された API は、攻撃者が自分のトラックをカバーし、検出を回避しようとしている防御回避戦術に一般的に関連しています。このカテゴリの API は通常、削除、無効化、停止オペレーションです (`DeleteFlowLogs`、`DisableAlarmActions`、`StopLogging` など)。

この API リクエストは、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。機械学習モデルは、アカウント内のすべての API リクエストを評価し、攻撃者が使用するテクニックに関連する異常なイベントを特定します。機械学習モデルは、リクエストを行ったユーザー、リクエストが行われた場所、リクエストされた特定の API など、API リクエストのさまざま

な要因を追跡します。リクエストしたユーザーアイデンティティにおいて API リクエストのどの要因が異常なのかという詳細については、「検出結果の詳細 」を参照してください。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

DefenseEvasion:IAMUser/BedrockLoggingDisabled

Amazon Bedrock のログ記録が無効になりました。

デフォルトの重要度: [Medium] (中)

- データソース CloudTrail 管理イベント

この検出結果は、アカウントの Bedrock モデル呼び出しのログ記録が無効になったことを知らせるものです。これは、データの不正引き出しや AI モデルの不正使用などの悪意のあるアクティビティを隠そうとする攻撃者の試みである可能性があります。ログ記録を無効にすると、モデルに送信されたデータとモデルの使用方法の可視性が失われます。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

Discovery:IAMUser/AnomalousBehavior

リソースの検出に一般的に使用される API が、異常な方法で呼び出されました。

デフォルトの重大度: [Low] (低

- データソース: CloudTrail 管理イベント

この検出結果は、アカウント内で異常な API リクエストが観察されたことを知らせるものです。この検出結果には、単一の API、または単一のユーザーアイデンティティで近似の一連の関連 API リクエストが含まれる場合があります。観察される API は、攻撃者が情報を収集して AWS 、環境がより広範な攻撃の影響を受けやすいかどうかを判断する場合、一般的に攻撃の検出段階に関連付けられます。このカテゴリの API は、get、describe、または list オペレーションです (`DescribeInstances`、`GetRolePolicy`、または `ListAccessKeys` など)。

この API リクエストは、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。機械学習モデルは、アカウント内のすべての API リクエストを評価し、攻撃者が使用するテクニックに関連する異常なイベントを特定します。機械学習モデルは、リクエストを行ったユーザー、リクエストが行われた場所、リクエストされた特定の API など、API リクエストのさまざまな要因を追跡します。リクエストしたユーザーアイデンティティにおいて API リクエストのどの要因が異常なのかという詳細については、「検出結果の詳細 」を参照してください。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

Exfiltration:IAMUser/AnomalousBehavior

AWS 環境からデータを収集するために一般的に使用される API が異常な方法で呼び出されました。

デフォルトの重大度: [High] (高

- データソース: CloudTrail 管理イベント

この検出結果は、アカウント内で異常な API リクエストが観察されたことを知らせるものです。この検出結果には、単一の API、または単一のユーザーアイデンティティで近似の一連の関連 API リクエストが含まれる場合があります。観測された API は、侵入戦術に一般的に関連していて、そこでは攻撃者がネットワークからデータを収集しようとしています。この検出結果タイプの API は管理 (コントロールプレーン)オペレーションのみであり、通常は、S3、スナップショット、およびデータベースに関連しています (`PutBucketReplication`、`CreateSnapshot`、`RestoreDBInstanceFromDBSnapshot` など)。

この API リクエストは、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。機械学習モデルは、アカウント内のすべての API リクエストを評価し、攻撃者が使用するテクニックに関連する異常なイベントを特定します。機械学習モデルは、リクエストを行ったユーザー、リクエストが行われた場所、リクエストされた特定の API など、API リクエストのさまざまな要因を追跡します。リクエストしたユーザーアイデンティティにおいて API リクエストのどの要因が異常なのかという詳細については、「検出結果の詳細 」を参照してください。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

Impact:IAMUser/AnomalousBehavior

AWS 環境内のデータまたはプロセスを改ざんするために一般的に使用される API が異常な方法で呼び出されました。

デフォルトの重大度: [High] (高

- データソース: CloudTrail 管理イベント

この検出結果は、アカウント内で異常な API リクエストが観察されたことを知らせるものです。この検出結果には、単一の API、または単一のユーザーアイデンティティで近似の一連の関連 API リクエストが含まれる場合があります。観察される API は、攻撃者がオペレーションを中断し、ユーザーのアカウント内のデータを操作、中断、または破壊しようとするインパクト戦術に一般的に関連しています。この検出結果タイプの API は、通常、delete、update、または put オペレーションです (`DeleteSecurityGroup`、`UpdateUser`、`PutBucketPolicy` など)。

この API リクエストは、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。機械学習モデルは、アカウント内のすべての API リクエストを評価し、攻撃者が使用するテクニックに関連する異常なイベントを特定します。機械学習モデルは、リクエストを行ったユーザー、リクエストが行われた場所、リクエストされた特定の API など、API リクエストのさまざまな要因を追跡します。リクエストしたユーザーアイデンティティにおいて API リクエストのどの要因が異常なのかという詳細については、「検出結果の詳細 」を参照してください。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

InitialAccess:IAMUser/AnomalousBehavior

AWS 環境への不正アクセスを得るために一般的に使用される API が異常な方法で呼び出されました。

デフォルトの重大度: [Medium] (中

- データソース: CloudTrail 管理イベント

この検出結果は、アカウント内で異常な API リクエストが観察されたことを知らせるものです。この検出結果には、単一の API、または単一のユーザーアイデンティティで近似の一連の関連

API リクエストが含まれる場合があります。攻撃者がユーザーの環境へのアクセスを確立しようとすると、観察される API は、攻撃の初期アクセス段階に一般的に関連しています。このカテゴリの API は、通常 get トークン、またはセッションオペレーションです (`StartSession` または `GetAuthorizationToken`)。

この API リクエストは、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。機械学習モデルは、アカウント内のすべての API リクエストを評価し、攻撃者が使用するテクニックに関連する異常なイベントを特定します。機械学習モデルは、リクエストを行ったユーザー、リクエストが行われた場所、リクエストされた特定の API など、API リクエストのさまざまな要因を追跡します。リクエストしたユーザーアイデンティティにおいて API リクエストのどの要因が異常なのかという詳細については、「検出結果の詳細 」を参照してください。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

PenTest:IAMUser/KaliLinux

API が Kali Linux マシンから呼び出されました。

デフォルトの重大度: [Medium] (中

- データソース: CloudTrail 管理イベント

この検出結果は、Kali Linux を実行しているマシンが、環境内のリストされた AWS アカウントに属する認証情報を使用して API コールを行っていることを知らせるものです。Kali Linux は、セキュリティプロフェッショナルがパッチ適用が必要な EC2 インスタンスの弱点を特定するために使用する一般的な侵入テストツールです。また、攻撃者はこのツールを使用して EC2 設定の弱点を見つけ、AWS 環境への不正アクセスを取得します。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

PenTest:IAMUser/ParrotLinux

API が Parrot Security Linux マシンから呼び出されました。

デフォルトの重大度: [Medium] (中

- データソース: CloudTrail 管理イベント

この検出結果は、Parrot Security Linux を実行しているマシンが、環境内のリストされた AWS アカウントに属する認証情報を使用して API コールを行っていることを知らせるものです。Parrot Security Linux は、セキュリティプロフェッショナルがパッチ適用が必要な EC2 インスタンスの弱点を特定するために使用する一般的な侵入テストツールです。また、攻撃者はこのツールを使用して EC2 設定の弱点を見つけ、 AWS 環境への不正アクセスを取得します。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

PenTest:IAMUser/PentooLinux

API が Pentoo Linux マシンから呼び出されました。

デフォルトの重大度: [Medium] (中

- データソース: CloudTrail 管理イベント

この検出結果は、Pentoo Linux を実行しているマシンが、環境内のリストされた AWS アカウントに属する認証情報を使用して API コールを行っていることを知らせるものです。Pentoo Linux は、セキュリティプロフェッショナルがパッチ適用が必要な EC2 インスタンスの弱点を特定するために使用する一般的な侵入テストツールです。また、攻撃者はこのツールを使用して EC2 設定の弱点を見つけ、 AWS 環境への不正アクセスを取得します。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

Persistence:IAMUser/AnomalousBehavior

AWS 環境への不正アクセスを維持するために一般的に使用される API が異常な方法で呼び出されました。

デフォルトの重大度: [Medium] (中

- データソース: CloudTrail 管理イベント

この検出結果は、アカウント内で異常な API リクエストが観察されたことを知らせるものです。この検出結果には、単一の API、または単一のユーザーアイデンティティで近似の一連の関連 API リクエストが含まれる場合があります。観察される API は、攻撃者がユーザーの環境へのアクセスを獲得し、そのアクセスを維持しようとするパーシスタンス戦術に一般的に関連しています。このカテゴリの API は、通常、create、インimport、または modify オペレーションです (`CreateAccessKey`、`ImportKeyPair`、`ModifyInstanceAttribute` など)。

この API リクエストは、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。機械学習モデルは、アカウント内のすべての API リクエストを評価し、攻撃者が使用するテクニックに関連する異常なイベントを特定します。機械学習モデルは、リクエストを行ったユーザー、リクエストが行われた場所、リクエストされた特定の API など、API リクエストのさまざまな要因を追跡します。リクエストしたユーザーアイデンティティにおいて API リクエストのどの要因が異常なのかという詳細については、「検出結果の詳細 」を参照してください。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

Policy:IAMUser/RootCredentialUsage

API がルートユーザーサインイン認証情報を使用して呼び出されました。

デフォルトの重大度: [Low] (低

- データソース: CloudTrail 管理イベントまたは S3 CloudTrail データイベント

この検出結果は、ユーザーの環境のリスト化された AWS アカウント のルートユーザーサインイン認証情報が AWS サービスへのリクエストに使用されていることを知らせるものです。ユーザーは、ルートユーザーのサインイン認証情報を使用して AWS サービスにアクセスしないことをお勧めします。代わりに、 AWS Security Token Service (STS) からの最小特権の一時的な認証情報を使用して AWS サービスにアクセスする必要があります。 AWS STS がサポートされていない状況では、IAM ユーザー認証情報をお勧めします。詳細については、「IAM ベストプラクティス」を参照してください。

> **Note**
>
> アカウントで S3 Protection が有効になっている場合、この検出結果は、 AWS アカウントのルートユーザーサインイン認証情報を使用して Amazon S3 リソースで S3 データプレーンオ

ペレーションを実行しようとした場合に応答して生成される可能性があります。使用された API コールは、検出結果の詳細でリスト化されます。S3 Protection が有効になっていない場合、この検出結果はイベントログ API によってのみトリガーされます。S3 Protection の詳細については、「S3 Protection」を参照してください。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

Policy:IAMUser/ShortTermRootCredentialUsage

制限付きのルートユーザー認証情報を使用して API が呼び出されました。

デフォルトの重大度: [Low] (低

- データソース: AWS CloudTrail S3 の管理イベントまたは AWS CloudTrail データイベント

この検出結果は、 環境にリスト AWS アカウント されている 用に作成された制限されたユーザー認証情報が、 へのリクエストに使用されていることを知らせるものです AWS のサービス。ルートユーザー認証情報を必要とするタスクにのみ、ルートユーザー認証情報を使用することをお勧めします。

可能であれば、 AWS Security Token Service () の一時的な認証情報を持つ最小特権の IAM ロール AWS のサービス を使用して にアクセスしますAWS STS。 AWS STS がサポートされていないシナリオの場合、ベストプラクティスは IAM ユーザー認証情報を使用することです。詳細については、「IAM ユーザーガイド」の「IAM でのセキュリティのベストプラクティス」と「 AWS アカウントでのルートユーザーベストプラクティス」を参照してください。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

PrivilegeEscalation:IAMUser/AnomalousBehavior

AWS 環境への高レベルのアクセス許可を取得するために一般的に使用される API が異常な方法で呼び出されました。

デフォルトの重要度: [Medium] (中)

- データソース CloudTrail 管理イベント

この検出結果は、アカウント内で異常な API リクエストが観察されたことを知らせるものです。この検出結果には、単一の API、または単一のユーザーアイデンティティで近似の一連の関連 API リクエストが含まれる場合があります。観察される API は、攻撃者が環境へのより高いレベルの許可を取得しようとする特権エスカレーション戦術に一般的に関連しています。このカテゴリの API は、通常、IAM ポリシー、ロール、ユーザーを変更するオペレーションを含みます (`AssociateIamInstanceProfile`、`AddUserToGroup`、`PutUserPolicy` など)。

この API リクエストは、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。機械学習モデルは、アカウント内のすべての API リクエストを評価し、攻撃者が使用するテクニックに関連する異常なイベントを特定します。機械学習モデルは、リクエストを行ったユーザー、リクエストが行われた場所、リクエストされた特定の API など、API リクエストのさまざまな要因を追跡します。リクエストしたユーザーアイデンティティにおいて API リクエストのどの要因が異常なのかという詳細については、「検出結果の詳細 」を参照してください。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

Recon:IAMUser/MaliciousIPCaller

API が悪意のある既知の IP アドレスから呼び出されました。

デフォルトの重大度: [Medium] (中

- データソース CloudTrail 管理イベント

この検出結果は、ユーザーの環境内のアカウントの AWS リソースをリスト化または記述できる API オペレーションが、脅威リストの IP アドレスから呼び出されたことを知らせるものです。攻撃者

は、盗まれた認証情報を使用して、より貴重な認証情報を見つけたり、既に持っている認証情報の機能を特定したりするために、 AWS リソースのこの種の偵察を実行する場合があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

Recon:IAMUser/MaliciousIPCaller.Custom

API が悪意のある既知の IP アドレスから呼び出されました。

デフォルトの重大度: [Medium] (中

- データソース CloudTrail 管理イベント

この検出結果は、ユーザーの環境のアカウントの AWS リソースをリスト化または説明できる API オペレーションがカスタム脅威リストの IP アドレスから呼び出されたことを知らせるものです。使用された脅威リストは、検出結果の詳細に表示されます。攻撃者は、盗まれた認証情報を使用して、より貴重な認証情報を見つけたり、既に持っている認証情報の機能を特定したりするために、 AWS リソースのこの種の偵察を実行する可能性があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

Recon:IAMUser/TorIPCaller

API が Tor 出口ノードの IP アドレスから呼び出されました。

デフォルトの重大度: [Medium] (中

- データソース CloudTrail 管理イベント

この検出結果は、ユーザーの環境のアカウントの AWS リソースをリスト化または説明できる API オペレーションが Tor 出口ノードの IP アドレスから呼び出されたことを知らせるものです。Tor は匿名通信を有効化するソフトウェアです。通信を暗号化し、一連のネットワークノード間のリレー中にランダムに通信をバウンスさせます。最後の Tor ノードは出口ノードと呼ばれます。攻撃者は真のアイデンティティを隠すために Tor を使用します。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

Stealth:IAMUser/CloudTrailLoggingDisabled

AWS CloudTrail ログ記録が無効になりました。

デフォルトの重大度: [Low] (低

- データソース CloudTrail 管理イベント

この検出結果は、 AWS 環境内の CloudTrail 証跡が無効になったことを知らせるものです。これにより、悪意ある目的でユーザーの AWS リソースへアクセスしている間、活動の痕跡を消してトラックを隠してログ記録を無効化しようとします。この検出結果は、証跡情報の削除または更新が成功することによってトリガーされる場合があります。また、この検出結果は、GuardDuty に関連付けられている証跡のログを保存する S3 バケットが削除された場合にもトリガーされます。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

Stealth:IAMUser/PasswordPolicyChange

アカウントのパスワードポリシーが弱化されています。

デフォルトの重大度: [Low] (低)*

> **Note**
>
> この検出結果の重大度は、パスワードポリシーに加えられた変更の重大度に応じて、[Low] (低)、[Medium] (中)、[High] (高) になります。

- データソース CloudTrail 管理イベント

AWS アカウントパスワードポリシーは、 AWS 環境内のリストされたアカウントで弱体化されました。例えば、アカウントの削除、必要な文字数を減らすような更新、記号や数字を不要とする更新、

パスワードの有効期限を延長するような更新が行われています。この検出結果は、 AWS アカウントのパスワードポリシーを更新または削除しようとしてトリガーすることもできます。 AWS アカウントパスワードポリシーは、IAM ユーザーに設定できるパスワードの種類を管理するルールを定義します。パスワードポリシーが弱いほど、覚えやすく、推測しやすいパスワードを作成できるため、セキュリティリスクが生じます。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

UnauthorizedAccess:IAMUser/ConsoleLoginSuccess.B

世界中でコンソールに対する複数の正常なログインが確認されました。

デフォルトの重大度: [Medium] (中

- データソース CloudTrail 管理イベント

この検出結果は、世界各地から同時に同じ IAM ユーザーによるコンソールへの複数の正常なログインが確認されたことを知らせるものです。このような異常でリスクの高いアクセス場所パターンは、AWS リソースへの不正アクセスの可能性を示します。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.InsideAWS

インスタンス起動ロールを通じて EC2 インスタンス専用に作成された認証情報は、AWS内の別のアカウントから使用されています。

デフォルトの重大度: [High] (高)*

> **Note**
>
> この検出結果のデフォルトの重大度は [High] (高) です。ただし、API が AWS 環境に関連付けられたアカウントによって呼び出された場合、重要度は中になります。

- データソース: CloudTrail 管理イベントまたは S3 CloudTrail データイベント

この検出結果は、Amazon EC2 インスタンス認証情報を使用して、関連付けられた Amazon EC2 インスタンスが実行されているアカウントとは異なる AWS アカウントが所有する IP アドレスまたは Amazon EC2 VPC エンドポイントから APIs を呼び出すときに通知します。VPC エンドポイント検出は、VPC エンドポイントのネットワークアクティビティイベントをサポートするサービスでのみ使用できます。VPC エンドポイントのネットワークアクティビティイベントをサポートするサービスの詳細については、「AWS CloudTrail ユーザーガイド」の「ネットワークアクティビティイベントのログ記録」を参照してください。

AWS では、一時的な認証情報を作成したエンティティ (AWS アプリケーション、Amazon EC2、など) の外部に再配布することはお勧めしません AWS Lambda。ただし、承認されたユーザーは Amazon EC2 インスタンスから認証情報をエクスポートして正当な API コールを行うことができます。`remoteAccountDetails.Affiliated` フィールドが `True` の場合、API は同じ管理者アカウントに関連付けられたアカウントから呼び出されました。潜在的な攻撃を除外し、アクティビティの正当性を検証するには、これらの認証情報が割り当てられている AWS アカウント 所有者または IAM プリンシパルにお問い合わせください。

> **Note**
>
> GuardDuty がリモートアカウントからの継続的なアクティビティを観察した場合、その機械学習 (ML) モデルはこれを予期される動作として識別します。したがって、GuardDuty は、そのリモートアカウントからのアクティビティに関するこの検出結果の生成を停止します。GuardDuty は、他のリモートアカウントからの新しい動作に関する検出結果を生成し続け、時間の経過とともに動作が変化するにつれて、学習したリモートアカウントを再評価します。

修復のレコメンデーション

この検出結果は、Amazon EC2 インスタンスのセッション認証情報を使用して AWS アカウント、の外部の Amazon EC2 インスタンス AWS を介して 内で API リクエストが行われたときに AWS 生成されます。ハブアンドスポーク設定の Transit Gateway アーキテクチャなどでは、 AWS サービスエンドポイントを持つ単一のハブ出力 VPC を介してトラフィックをルーティングするのが慣例である場合があります。この動作が予想される場合、GuardDuty では、抑制ルールを使用し、2 つのフィルター条件を使用するルールを作成することをお勧めします。1 つ目の条件は検出結果タイプで、この場合は UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.InsideAWS です。2 つ目のフィルター条件は、リモートアカウントの詳細のリモートアカウント ID です。

この検出結果に応じて、次のワークフローを使用して、一連のアクションを決定できます。

1. `service.action.awsApiCallAction.remoteAccountDetails.accountId` フィールドから関係するリモートアカウントを特定します。
2. そのアカウントが `service.action.awsApiCallAction.remoteAccountDetails.affiliated` フィールドから GuardDuty 環境に関連付けられているかどうかを確認します。
3. アカウントが連携している場合は、リモートアカウントの所有者と Amazon EC2 インスタンスの認証情報の所有者に問い合わせて、調査してください。

   アカウントが提携していない場合、最初の手順として、そのアカウントが組織に関連付けられているが、GuardDuty マルチアカウント環境の設定の一部となっていないかどうか、または GuardDuty がこのアカウント内でまだ有効化されていないかどうかを評価します。次に、Amazon EC2 インスタンス認証情報の所有者に問い合わせて、リモートアカウントがこれらの認証情報を使用するユースケースがあるかどうかを判断します。
4. 認証情報の所有者がリモートアカウントを認識しない場合は、 AWS内で動作する脅威アクターによって認証情報が侵害された可能性があります。ご利用の環境を保護するために、「侵害された可能性のある Amazon EC2 インスタンスの修復」で推奨されているステップを実行する必要があります。

   さらに、 AWS Trust and Safety チームに不正使用レポートを送信して、リモートアカウントの調査を開始できます。 AWS Trust and Safety にレポートを送信するときは、検出結果の完全な JSON 詳細を含めてください。

UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.OutsideAWS

インスタンス作成ロールで EC2 インスタンス専用に作成された認証情報が外部 IP アドレスから使用されています。

デフォルトの重大度: [High] (高

- データソース: CloudTrail 管理イベントまたは S3 CloudTrail データイベント

この検出結果は、 の外部のホスト AWS が、 AWS 環境の EC2 インスタンスで作成された一時的な AWS 認証情報を使用して AWS API オペレーションを実行しようとしたことを知らせるものです。リストされている EC2 インスタンスが侵害されている可能性があり、このインスタンスの一時的な認証情報が の外部にあるリモートホストに流出した可能性があります AWS。 AWS では、一時的な

認証情報を作成したエンティティ (AWS アプリケーション、EC2、Lambda など) の外部に再配布することはお勧めしません。ただし、承認されたユーザーは EC2 インスタンスから認証情報をエクスポートして正当な API コールを行うことができます。潜在的な攻撃を除外し、アクティビティの正当性を検証するには、検出結果においてリモート IP からのインスタンスの認証情報の使用が想定されるかどうかを検証します。

> **Note**
>
> GuardDuty がリモートホストからの継続的なアクティビティを観察した場合、その機械学習 (ML) モデルはこれを予期される動作として識別します。したがって、GuardDuty は、そのリモートホストからのアクティビティに関するこの検出結果の生成を停止します。GuardDuty は、他のリモートホストからの新しい動作に関する検出結果を生成し続け、時間の経過とともに動作が変化するにつれて、学習したリモートホストを再評価します。

修正のレコメンデーション

この検出結果が生成されるのは、VPC インターネットゲートウェイ (IGW) からではなく、オンプレミスのゲートウェイから排出され、インターネットトラフィックがルーティングされるように、ネットワークが構成されている場合です。AWS Outposts や VPC VPN 接続などの一般的な構成では、このようにトラフィックがルーティングされる可能性があります これが予期した動作である場合は、抑制ルールを使用して、2 つのフィルター条件で構成されるルールを作成することをお勧めします。1 つ目の条件では、[finding type] (結果タイプ) に `UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.OutsideAWS` を使用します。2 番目のフィルター条件は、オンプレミスインターネットゲートウェイの IP アドレスまたは CIDR 範囲を持つ [API caller IPv4 Address] (API 発信者の IPv4 アドレス) です。抑制ルールの作成の詳細については、「GuardDuty の抑制ルール」を参照してください。

> **Note**
>
> GuardDuty が外部ソースからの継続的なアクティビティを観察した場合、その機械学習モデルは、想定されるアクティビティとしてこれを識別し、そのソースからのアクティビティの検出結果の生成を停止します。GuardDuty は、他のソースからの新しい動作に関する検出結果を引き続き生成し、学習したソースを時間の経過とともに変化する動作として再評価します。

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。「侵害された可能性のある AWS 認証情報の修正」を参照してください。

UnauthorizedAccess:IAMUser/MaliciousIPCaller

API が悪意のある既知の IP アドレスから呼び出されました。

デフォルトの重大度: [Medium] (中

- データソース CloudTrail 管理イベント

この検出結果は、API オペレーション (EC2 インスタンスの起動、新しい IAM ユーザーの作成、また、 AWS 権限の変更などの試行など) が悪意のある既知の IP アドレスから呼び出されたことを知らせるものです。これは、環境内の AWS リソースへの不正アクセスを示している可能性があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

UnauthorizedAccess:IAMUser/MaliciousIPCaller.Custom

API がカスタム脅威リストにある IP アドレスから呼び出されました。

デフォルトの重大度: [Medium] (中

- データソース CloudTrail 管理イベント

この検出結果は、API オペレーション (EC2 インスタンスの起動、新しい IAM ユーザーの作成、AWS 権限の変更など) が、アップロードした脅威リストに含まれている IP アドレスから呼び出されたことを知らせるものです。GuardDuty の場合、脅威リストは悪意のある既知の IP アドレスで構成されます。これは、環境内の AWS リソースへの不正アクセスを示している可能性があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.InsideAWS

AWS Amazon ECS タスクまたは AWS Lambda 関数専用に作成された認証情報は、内の別のアカウントから使用されています。 AWS

デフォルトの重大度: [High] (高)*

> **Note**
>
> この検出結果のデフォルトの重大度は [High] (高) です。ただし、API が AWS 環境に関連付けられたアカウントによって呼び出された場合、重要度は中になります。

- データソース: CloudTrail 管理イベントまたは S3 CloudTrail データイベント

この検出結果は、 AWS Amazon ECS タスク認証情報または AWS Lambda 関数認証情報を使用して、関連付けられたリソースが実行されているアカウントとは異なる AWS アカウントが所有する IP アドレスまたは Amazon VPC エンドポイントから APIs を呼び出すときに通知します。VPC エンドポイント検出は、VPC エンドポイントのネットワークアクティビティイベントをサポートするサービスでのみ使用できます。VPC エンドポイントのネットワークアクティビティイベントをサポートするサービスの詳細については、「AWS CloudTrail ユーザーガイド」の「ネットワークアクティビティイベントのログ記録」を参照してください。

AWS では、一時的な認証情報を作成したエンティティ (AWS アプリケーション、Amazon EC2、Amazon Elastic Container Service など) の外部に再配布することはお勧めしません。ただし、承認されたユーザーは、リソースから認証情報をエクスポートして正当な API コールを行うことができます。`remoteAccountDetails.affiliated` フィールドが `True`の場合、API は同じ管理者アカウントに関連付けられたアカウントから呼び出されたことを意味します。潜在的な攻撃を除外し、アクティビティの正当性を検証するには、これらの認証情報が割り当てられている AWS アカウント 所有者または IAM プリンシパルにお問い合わせください。

> **Note**
>
> GuardDuty がリモートアカウントからの継続的なアクティビティを観察した場合、その機械学習 (ML) モデルはこれを予期される動作として識別します。したがって、GuardDuty は、そのリモートアカウントからのアクティビティに関するこの検出結果の生成を停止します。GuardDuty は、他のリモートアカウントからの新しい動作に関する検出結果を生成し続

け、時間の経過とともに動作が変化するにつれて、学習したリモートアカウントを再評価します。

> **Note**
>
> Amazon ECS タスクの場合、認証情報はタスクロールに関連付けられ、VPC 設定に関係なく検出が適用されます。

修復のレコメンデーション

この検出結果は、 AWS Amazon ECS タスク認証情報または AWS Lambda 関数認証情報を使用して AWS アカウント、 外のリソースを介して AWS API リクエストが内部 AWS で行われたときに生成されます。ハブアンドスポーク設定の Transit Gateway アーキテクチャなどでは、AWS サービスエンドポイントを持つ単一のハブ出力 VPC を介してトラフィックをルーティングするのが慣例である場合があります。これが予想される動作である場合は、フィルター条件の検出結果タイプ抑制ルールで を使用することをお勧めしますUnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.InsideAWS。

この検出結果に応じて、次のワークフローを使用して、一連のアクションを決定できます。

1. `service.action.awsApiCallAction.remoteAccountDetails.accountId` フィールドから関係するリモートアカウントを特定します。
2. そのアカウントが `service.action.awsApiCallAction.remoteAccountDetails.affiliated` フィールドから GuardDuty 環境に関連付けられているかどうかを確認します。
3. アカウントが関連している場合は、リモートアカウントの所有者とリソース認証情報の所有者に連絡して調査してください。

   アカウントが提携していない場合、最初の手順として、そのアカウントが組織に関連付けられているが、GuardDuty マルチアカウント環境の設定の一部となっていないかどうか、または GuardDuty がこのアカウント内でまだ有効化されていないかどうかを評価します。次に、リソース認証情報の所有者に連絡して、リモートアカウントがこれらの認証情報を使用するユースケースがあるかどうかを確認します。
4. 認証情報の所有者がリモートアカウントを認識しない場合は、 AWS内で動作する脅威アクターによって認証情報が侵害された可能性があります。ご利用の環境を保護するために、「侵害された可能性のある ECS クラスターの修復」で推奨されているステップを実行する必要があります。

さらに、 AWS Trust and Safety チームに不正使用レポートを送信して、リモートアカウントの調査を開始できます。 AWS Trust and Safety にレポートを送信するときは、検出結果の完全な JSON 詳細を含めてください。

UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.OutsideAWS

AWS リソース (Lambda 関数または Amazon ECS タスク) 専用に作成された認証情報は、外部の IP アドレスから使用されています AWS。

デフォルトの重大度: [High] (高

- データソース: CloudTrail 管理イベントまたは S3 CloudTrail データイベント

この検出結果は、 以外のホストが、 AWS 環境の AWS リソース (Lambda 関数または Amazon ECS タスク) で作成された一時的な AWS 認証情報を使用して AWS API オペレーションを実行 AWS しようとしたことを知らせるものです。リストされているリソースが侵害されている可能性があり、このリソースの一時的な認証情報が 外のリモートホストに流出した可能性があります AWS。

AWS では、一時的な認証情報を作成したエンティティ (Amazon Elastic Compute Cloud (Amazon EC2)、Amazon Elastic Container Service、 などの AWS アプリケーション) の外部に再配布することはお勧めしません AWS Lambda。ただし、承認されたユーザーは、リソースから認証情報を工クスポートして正当な API コールを行うことができます。潜在的な攻撃を除外し、アクティビティの正当性を検証するには、検出結果でリモート IP からのリソース認証情報の使用が予想されるかどうかを確認します。

> **Note**
>
> GuardDuty がリモートホストからの継続的なアクティビティを観察した場合、その機械学習 (ML) モデルはこれを予期される動作として識別します。したがって、GuardDuty は、そのリモートホストからのアクティビティに関するこの検出結果の生成を停止します。GuardDuty は、他のリモートホストからの新しい動作に関する検出結果を生成し続け、時間の経過とともに動作が変化するにつれて、学習したリモートホストを再評価します。

修復のレコメンデーション

この検出結果が生成されるのは、VPC インターネットゲートウェイ (IGW) からではなく、オンプレミスのゲートウェイから排出され、インターネットトラフィックがルーティングされるように、ネットワークが構成されている場合です。AWS Outposts や VPC VPN 接続などの一般的な構成では、このようにトラフィックがルーティングされる可能性があります。これが予想される動作である場合、GuardDuty は 抑制ルール を使用して 2 つのフィルター条件を持つルールを作成することを推奨しています。1 つ目の条件では、[finding type] (結果タイプ) に `UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.OutsideAWS` を使用します。2 番目のフィルター条件は、オンプレミスのインターネットゲートウェイの IP アドレスまたは CIDR 範囲を持つ [API 発信者の IPv4 アドレス] です。

このアクティビティが予期しないものである場合は、認証情報が侵害された可能性があります。この検出結果タイプを修正する手順については、侵害された可能性のある AWS 認証情報の修正「」または「」を参照してください侵害された可能性のある ECS クラスターの修復。

UnauthorizedAccess:IAMUser/TorIPCaller

API が Tor 出口ノードの IP アドレスから呼び出されました。

デフォルトの重大度: [Medium] (中

- データソース CloudTrail 管理イベント

この検出結果は、API オペレーション (EC2 インスタンスの起動、新しい IAM ユーザーの作成、AWS 権限の変更などの試行など) が Tor 出口ノードの IP アドレスから呼び出されたことを知らせるものです。Tor は匿名通信を有効化するソフトウェアです。通信を暗号化し、一連のネットワークノード間のリレー中にランダムに通信をバウンスさせます。最後の Tor ノードは出口ノードと呼ばれます。これは、攻撃者が真のアイデンティティを隠しているという意図により、 AWS リソースへの未承認のアクセスを示している場合があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

GuardDuty 攻撃シーケンスの検出結果タイプ

GuardDuty は、特定の一連の複数のアクションが疑わしい可能性のあるアクティビティと一致する場合に、攻撃シーケンスを検出します。攻撃シーケンスには、API アクティビティや GuardDuty の

検出結果などのシグナルが含まれます。GuardDuty が、進行中、継続中、または最近のセキュリティ脅威を示す特定のシーケンスでシグナルのグループを観察すると、GuardDuty は攻撃シーケンスの検出結果を生成します。GuardDuty は、個々の API アクティビティを潜在的な脅威として提示しないため、weak signals と見なします。

攻撃シーケンス検出は、Amazon S3 データの潜在的な侵害 (より広範なランサムウェア攻撃の一部である可能性があります)、 AWS 認証情報の侵害、Amazon EKS クラスターの侵害、Amazon ECS クラスターの侵害、Amazon EC2 インスタンスグループの侵害に焦点を当てています。以下のセクションでは、各攻撃シーケンスについて詳しく説明します。

トピック

- AttackSequence:EKS/CompromisedCluster
- AttackSequence:ECS/CompromisedCluster
- AttackSequence:EC2/CompromisedInstanceGroup
- AttackSequence:IAM/CompromisedCredentials
- AttackSequence:S3/CompromisedData

AttackSequence:EKS/CompromisedCluster

侵害された可能性のある Amazon EKS クラスターによって実行される一連の疑わしいアクション。

- デフォルトの重大度: 重大
- データソース:
  - EKS 監査ログイベント
  - Amazon EKS の Runtime Monitoring
  - Amazon EC2 の Amazon EKS マルウェア検出
  - AWS CloudTrail S3 のデータイベント
  - AWS CloudTrail 管理イベント
  - VPC フローログ
  - Route53 Resolver DNS クエリログ

この検出結果は、GuardDuty が環境内の侵害された可能性のある Amazon EKS クラスターを示す一連の疑わしいアクションを検出したことを知らせるものです。悪意のあるプロセスや悪意のあるエン

ドポイントへの接続など、複数の疑わしい異常な攻撃動作が、同じ Amazon EKS クラスターで観察されました。

GuardDuty は独自の相関アルゴリズムを使用して、IAM 認証情報を使用して実行される一連のアクションを監視および識別します。GuardDuty は、保護プランやその他のシグナルソース全体の検出結果を評価し、一般的および新たな攻撃パターンを特定します。GuardDuty は、IP の評価、API シーケンス、ユーザー設定、影響を受ける可能性のあるリソースなど、複数の要因を使用して脅威を表面化します。

修復アクション: 環境でこの動作が予期しない場合、Amazon EKS クラスターが侵害されている可能性があります。包括的な修復ガイダンスについては、「EKS Protection の検出結果の修復」および「Runtime Monitoring 検出結果の修正」を参照してください。

さらに、 AWS 認証情報が EKS クラスターを介して侵害された可能性があるため、「」を参照してください侵害された可能性のある AWS 認証情報の修正。影響を受ける可能性のある他のリソースを修復する手順については、「検出された GuardDuty セキュリティ検出結果の修復」を参照してください。

AttackSequence:ECS/CompromisedCluster

侵害された可能性のある Amazon ECS クラスターによって実行される一連の疑わしいアクション。

- デフォルトの重大度: 重大
- データソース:
  - Amazon ECS Fargate のランタイムモニタリング
  - Amazon ECS での EC2 インスタンスのランタイムモニタリング
  - Amazon EC2 の GuardDuty Malware Protection

この検出結果は、GuardDuty が環境内の侵害された可能性のある Amazon ECS クラスターを示す一連の疑わしいシグナルを検出したことを知らせるものです。これらのシグナルには、悪意のあるプロセス、悪意のあるエンドポイントとの通信、または暗号通貨のマイニング動作が含まれる場合があります。

GuardDuty は、独自の相関アルゴリズムと複数の検出要因を使用して、Amazon ECS クラスター内の疑わしいアクションのシーケンスを識別します。GuardDuty は、保護プランとさまざまなシグナルソースを横断して分析することで、一般的な攻撃パターンと新たな攻撃パターンを特定し、潜在的な侵害を高い信頼性で検出します。

修復アクション: 環境でこの動作が予期しない場合、Amazon ECS クラスターが侵害されている可能性があります。脅威封じ込めの推奨事項については、「」を参照してください侵害された可能性のある ECS クラスターの修復。侵害は、 AWS リソースの作成または変更に使用された可能性のある 1 つ以上の ECS タスクまたはコンテナワークロードにも及ぶ可能性があることに注意してください。影響を受ける可能性のあるリソースに関する包括的な修復ガイダンスについては、「」を参照してください検出された GuardDuty セキュリティ検出結果の修復。

AttackSequence:EC2/CompromisedInstanceGroup

侵害された可能性のある Amazon EC2 インスタンスを示す一連の疑わしいアクション。

- デフォルトの重大度: 重大
- データソース:
  - Amazon EC2 のランタイムモニタリング
  - Amazon EC2 のマルウェア検出
  - VPC フローログ
  - Route53 Resolver DNS クエリログ

この検出結果は、GuardDuty が環境内の Amazon EC2 インスタンスのグループ全体で潜在的な侵害を示唆する一連の疑わしいアクションを検出したことを示しています。インスタンスグループは、通常、infrastructure-as-codeで管理されるアプリケーションを表し、Auto Scaling グループ、IAM インスタンスプロファイルロール、 AWS CloudFormation スタック、Amazon EC2 起動テンプレート、AMI、VPC ID などの同様の設定を共有します。GuardDuty は、以下を含む複数の疑わしい動作を 1 つ以上のインスタンスで観察しました。

- 悪意のあるプロセス
- 悪意のあるファイル
- 疑わしいネットワーク接続
- 暗号通貨マイニングアクティビティ
- Amazon EC2 インスタンス認証情報の疑わしい使用

検出方法: GuardDuty は独自の相関アルゴリズムを使用して、Amazon EC2 インスタンス内の疑わしいアクションシーケンスを識別します。GuardDuty は、保護プランやさまざまなシグナルソースに

わたる検出結果を評価することで、IP やドメインの評価、疑わしい実行プロセスなど、複数の要因を使用して攻撃パターンを識別します。

修復アクション: 環境でこの動作が予期しない場合、Amazon EC2 インスタンスが侵害される可能性があります。侵害には、以下が含まれる可能性があります。

- 複数のプロセス
- Amazon EC2 インスタンスまたは他の AWS リソースの変更に使用された可能性のあるインスタンス認証情報

脅威封じ込めの推奨事項については、「」を参照してください侵害された可能性のある Amazon EC2 インスタンスの修復。侵害は 1 つ以上の Amazon EC2 インスタンスにまで及ぶ可能性があり、Amazon EC2 インスタンスやその他の AWS リソースを作成または変更するために使用された可能性のある侵害されたプロセスまたはインスタンス認証情報が含まれる可能性があることに注意してください。影響を受ける可能性のあるリソースに関する包括的な修復ガイダンスについては、「」を参照してください検出された GuardDuty セキュリティ検出結果の修復。

AttackSequence:IAM/CompromisedCredentials

侵害された可能性のある AWS 認証情報を使用して呼び出された一連の API リクエスト。

- デフォルトの重大度: 重大
- データソース: AWS CloudTrail 管理イベント

この検出結果は、GuardDuty が環境内の 1 つ以上のリソースに影響を与える AWS 認証情報を使用して行われた一連の疑わしいアクションを検出したことを知らせるものです。同じ認証情報によって複数の疑わしい攻撃動作と異常な攻撃動作が観察され、認証情報が悪用されているという信頼度が高くなります。

GuardDuty は独自の相関アルゴリズムを使用して、IAM 認証情報を使用して実行される一連のアクションを監視および識別します。GuardDuty は、保護プランやその他のシグナルソース全体の検出結果を評価し、一般的および新たな攻撃パターンを特定します。GuardDuty は、IP の評価、API シーケンス、ユーザー設定、影響を受ける可能性のあるリソースなど、複数の要因を使用して脅威を表面化します。

を有効にするとAI 保護、GuardDuty はこの攻撃シーケンスと相関するシグナルAI Protection の検出結果タイプに を含めることもできます。これらの検出結果は IAM ID に関連付けられています。

その結果、GuardDuty はそれらをこの攻撃シーケンスのシグナルとして表示し、侵害された可能性のある同じ認証情報で実行された他の疑わしいアクションと関連付けることができます。例えば、GuardDuty は他のシグナルを、それらの認証情報で行われた異常な Amazon Bedrock または Amazon SageMaker AI モデルの呼び出しと関連付け、この攻撃シーケンスに含めることができます。単独では、AI Protection の検出結果は攻撃シーケンスを生成しません。詳細については、「AI 保護」を参照してください。

修復アクション: 環境でこの動作が予期しない場合、 AWS 認証情報が侵害されている可能性があります。修正する手順については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。侵害された認証情報は、環境内の Amazon S3 バケット、 AWS Lambda 関数、Amazon EC2 インスタンスなどの追加のリソースを作成または変更するために使用された可能性があります。影響を受ける可能性のある他のリソースを修復する手順については、「検出された GuardDuty セキュリティ検出結果の修復」を参照してください。

AttackSequence:S3/CompromisedData

一連の API リクエストが、Amazon S3 内のデータを流出または破壊しようとして呼び出されました。

- デフォルトの重大度: 重大
- データソース: AWS CloudTrail S3 のデータイベント および AWS CloudTrail 管理イベント

この検出結果は、GuardDuty が、侵害された可能性のある AWS 認証情報を使用して、1 つ以上の Amazon Simple Storage Service (Amazon S3) バケットでデータ侵害を示す一連の疑わしいアクションを検出したことを知らせるものです。複数の疑わしい攻撃動作と異常な攻撃動作 (API リクエスト) が観察され、認証情報が悪用されているという信頼度が高くなります。

GuardDuty は相関アルゴリズムを使用して、IAM 認証情報を使用して実行される一連のアクションを観察および識別します。GuardDuty はその後、保護プランやその他のシグナルソースにわたる発見事項を評価し、共通および新たな攻撃パターンを特定します。GuardDuty は、IP の評価、API シーケンス、ユーザー設定、影響を受ける可能性のあるリソースなど、複数の要因を使用して脅威を表面化します。

修復アクション: 環境でこのアクティビティが予期しない場合、 AWS 認証情報または Amazon S3 データが流出または破棄される可能性があります。修正する手順については、「侵害された可能性のある AWS 認証情報の修正」および「侵害された可能性のある S3 バケットの修復」を参照してください。

カスタム検出ルールの検出結果タイプ

このページでは、GuardDuty がカスタム検出ルールから生成する検出結果タイプをデータソース別にグループ化して一覧表示します。各検出結果タイプは、1 つ以上のシグナルを 1 つの検出結果に集約します。カスタム検出ルールの有効化の詳細については、「」を参照してくださいGuardDuty のカスタム検出ルール。

> **Note**
>
> 個々のルールは、ターゲット AWS リージョン とする AWS サービスまたは機能をサポートする でのみ使用できます。例えば、 AWS Organizations または Amazon Simple Email Service をターゲットとするルールは、それらのサービスが動作するリージョンでのみ使用できます。同様に、Lambda 関数 URLs または SageMaker AI ノートブックインスタンスをターゲットとするルールは、これらの機能がサポートされているリージョンでのみ使用できます。

トピック

- AWS CloudTrail 管理イベントのカスタム検出ルール
- ランタイムモニタリングのカスタム検出ルール

AWS CloudTrail 管理イベントのカスタム検出ルール

以下の検出結果タイプは、MITRE ATT&CK® 戦術別に整理された AWS CloudTrail 管理イベントシグナルから生成されます。

認証情報アクセス

これらのルールは、Amazon EC2 インスタンスパスワードの取得や Amazon RDS 管理者パスワードのリセットなど、認証情報を取得またはリセットするオペレーションを検出しようとします。

結果タイプ	ルール名	緊急度
CredentialAccess:EC2/UnsecuredCredentials	EC2PasswordDataRetrieved	MEDIUM
CredentialAccess:RDS/ModifyAuthenticationProcess	RDSMasterPasswordReset	HIGH

防御障害

これらのルールは、VPC フローログの削除、組織ポリシーの無効化、組織からのアカウントの削除など、セキュリティコントロールを無効化または弱体化する変更を検出しようとします。

結果タイプ	ルール名	緊急度
DefenseImpairment:EC2/DisableOrModifyTools	EC2TerminationProtectionEnabled	LOW
	VPCFlowLogsDeleted	HIGH
DefenseImpairment:Organizations/ModifyCloudResourceHierarchy	OrganizationLeaveAttempt	HIGH
DefenseImpairment:Organizations/DisableOrModifyTools	OrganizationPolicyDisabled	HIGH
DefenseImpairment:RDS/ModifyAuthenticationProcess	RDSIAMAuthDisabled	MEDIUM
DefenseImpairment:Route53Resolver/DisableOrModifyTools	DNSQueryLogsDeleted	MEDIUM
DefenseImpairment:S3/DisableOrModifyTools	S3CustomerProvidedKeysEnabled	HIGH

実行

これらのルールは、SageMaker AI ノートブックのライフサイクル設定を変更するなど、リソースでコードを実行する変更を検出しようとします。

結果タイプ	ルール名	緊急度
Execution:SageMaker/CommandAndScriptingInterpreter	SageMakerLifecycleConfigModified	MEDIUM

Exfiltration

これらのルールは、AMI、Amazon EBS スナップショット、Amazon RDS スナップショットを他のアカウントにアクセス可能にするなど、アカウント外のリソースを共有する変更を検出しようとします。

結果タイプ	ルール名	緊急度
Exfiltration:EC2/TransferDataToCloudAccount	AMIExternalAccess	HIGH
	EBSSnapshotExternalAccess	HIGH
除外: RDS/TransferDataToCloudAccount	RDSSnapshotPubliclyShared	HIGH
除外: S3/TransferDataToCloudAccount	S3BucketPolicyExternalAccess	HIGH

Impact

これらのルールは、Amazon S3 オブジェクトをすばやく期限切れに設定する、Amazon SES 送信を有効にするなど、データとリソースを中断または破壊する可能性のある変更を検出しようとします。

結果タイプ	ルール名	緊急度
影響:S3/DataDestruction	S3LifecycleRapidExpiration	MEDIUM
影響:SES/ResourceHijacking	SESAccountSendingEnabled	LOW
	SESProductionAccessEnabled	MEDIUM

初回アクセス

これらのルールは、MFA を使用しないコンソールサインインや Amazon RDS データベースをパブリックにアクセス可能にするなど、環境へのアクセスを開くアクティビティを検出しようとします。

結果タイプ	ルール名	緊急度
InitialAccess:IAM/ValidAccounts	ConsoleLoginWithoutMFA	MEDIUM

結果タイプ	ルール名	緊急度
InitialAccess:RDS/ExploitPublicFacingApplication	RDSPubliclyAccessible	HIGH

横方向への移動

これらのルールは、SSH キーを Amazon EC2 インスタンスにプッシュするなど、リソース間のアクセスを許可するアクティビティを検出しようとします。

結果タイプ	ルール名	緊急度
LateralMovement:EC2/RemoteServices	EC2InstanceSSHKeyPushed	MEDIUM

永続的

これらのルールは、IAM アクセスキーの作成、ロールの信頼ポリシーの変更、Lambda 関数のパブリックアクセスなど、アクセスを作成または拡張する変更を検出しようとします。

結果タイプ	ルール名	緊急度
Persistence:EC2/BootOrLogonInitializationScripts	EC2UserDataModified	HIGH
Persistence:EC2/ExternalRemoteServices	EC2SecurityGroupPublicSSH	MEDIUM
永続性:IAM/AccountManipulation	IAMAccessKeyCreated	LOW
	IAMRoleTrustPolicyModified	MEDIUM
	IAMUserLoginProfileCreated	HIGH
	IAMUserLoginProfileUpdated	MEDIUM
永続性:Lambda/AccountManipulation	LambdaFunctionPublicAccess	HIGH
Persistence:Lambda/ModifyAuthenticationProcess	LambdaFunctionUrlPublicAccess	HIGH

結果タイプ	ルール名	緊急度
永続性:RolesAnywhere/AccountManipulation	RolesAnywhereTrustAnchorCreated	MEDIUM

権限昇格

これらのルールは、IAM ユーザーまたはロールに管理者ポリシーをアタッチするなど、より広範なアクセス許可を付与する変更を検出しようとします。

結果タイプ	ルール名	緊急度
PrivilegeEscalation:IAM/AccountManipulation	AdminPolicyAttachedToRole	HIGH
	AdminPolicyAttachedToUser	HIGH
	BedrockServiceCredentialCreated	MEDIUM
	SESFullAccessPolicyAttachedToRole	MEDIUM
	SESFullAccessPolicyAttachedToUser	MEDIUM

リソース開発

これらのルールは、Amazon SES ドメイン ID の検証など、後のアクティビティをサポートできるリソースの設定を検出しようとします。

結果タイプ	ルール名	緊急度
ResourceDevelopment:SES/CompromiseAccounts	SESDomainIdentityVerified	MEDIUM

ランタイムモニタリングのカスタム検出ルール

GuardDuty は、MITRE ATT&CK® 戦術別に整理された Runtime Monitoring シグナルから次のカスタム検出ルールの検出結果タイプを生成します。これらのルールには、Runtime Monitoring と、モニタリング対象リソースの GuardDuty セキュリティエージェントが必要です。詳細については、「データソースとシグナル」を参照してください。

コマンドアンドコントロール

これらのルールは、DNS トンネリングツールやプロトコルトンネリングなど、ネットワークトラフィックをトンネリングするプロセスを検出しようとします。

結果タイプ	ルール名	緊急度
CommandAndControl:Runtime/SuspiciousCommand	DnsTunnelingToolExecuted	HIGH
CommandAndControl:Runtime/SuspiciousCommand	SuspiciousProtocolTunneling	MEDIUM

認証情報アクセス

これらのルールは、`/proc`ファイルシステムを介したプロセスメモリなど、機密データを読み取るプロセスを検出しようとします。

結果タイプ	ルール名	緊急度
CredentialAccess:Runtime/SensitiveFileRead	DumpProcessMemoryFromProc	HIGH

防御障害

これらのルールは、監査ログ記録の無効化、ファイアウォールルールの変更、サービスの停止など、ホスト防御を弱めるプロセスを検出しようとします。

結果タイプ	ルール名	緊急度
DefenseEvasion:Runtime/SensitiveFileModified	FirewallConfigModified	MEDIUM

結果タイプ	ルール名	緊急度
DefenseEvasion:Runtime/SensitiveFileModified	LogManipulation	MEDIUM
DefenseEvasion:Runtime/SuspiciousCommand	AuditDisabled	MEDIUM
DefenseEvasion:Runtime/SuspiciousCommand	FirewallDisablement	MEDIUM
DefenseEvasion:Runtime/SuspiciousCommand	KernelHardeningDisabled	MEDIUM
DefenseEvasion:Runtime/SuspiciousCommand	KillCommandUsage	MEDIUM
DefenseEvasion:Runtime/SuspiciousCommand	ServiceStoppedOrDisabled	MEDIUM

発見

これらのルールは、パスワードファイルの読み取り、認証情報やキーの検索、仮想化の検出など、リソースに関する情報を収集するプロセスを検出しようとします。

結果タイプ	ルール名	緊急度
Discovery:Runtime/SensitiveFileRead	PasswordShadowFileDiscovery	MEDIUM
Discovery:Runtime/SuspiciousCommand	FindCommandReconCloudCredentials	MEDIUM
Discovery:Runtime/SuspiciousCommand	FindCommandReconKeyMaterial	MEDIUM
Discovery:Runtime/SuspiciousCommand	PromiscuousModeEnabled	MEDIUM
Discovery:Runtime/SuspiciousCommand	SSHWithHostKeyCheckingDisabled	MEDIUM
Discovery:Runtime/SuspiciousCommand	SqliteDatabaseDump	LOW
Discovery:Runtime/SuspiciousCommand	VirtualizationDetection	LOW

実行

これらのルールは、スクリプトのダウンロードと実行、コンテナ内のインタラクティブシェルを開くなど、悪意のある可能性のあるコマンドやコードを実行するプロセスを検出しようとします。

結果タイプ	ルール名	緊急度
Execution:Runtime/SuspiciousCommand	ChmodInTmpDirectory	MEDIUM
Execution:Runtime/SuspiciousCommand	CurlWgetScriptDownload	MEDIUM
Execution:Runtime/SuspiciousCommand	DataEncryptionUsingCommandLineUtilities	MEDIUM
Execution:Runtime/SuspiciousCommand	InlinePythonExecution	MEDIUM
Execution:Runtime/SuspiciousCommand	InteractiveShellInsideAContainer	MEDIUM
Execution:Runtime/SuspiciousCommand	RemoteFileCopy	MEDIUM
Execution:Runtime/SuspiciousCommand	ShellFromUtilities	MEDIUM
Execution:Runtime/SuspiciousTool	NetworkSnifferToolExecuted	MEDIUM
Execution:Runtime/SuspiciousTool	RawSocketCreated	MEDIUM

Exfiltration

これらのルールは、curl、wget、rclone を含むファイルのアップロード、転送前のデータの分割など、データをリソース外に移動するプロセスを検出しようとします。

結果タイプ	ルール名	緊急度
Exfiltration:Runtime/SuspiciousCommand	CurlWgetExfiltrateData	MEDIUM
Exfiltration:Runtime/SuspiciousCommand	DataSplittingForExfiltration	MEDIUM
Exfiltration:Runtime/SuspiciousCommand	S3DataExfiltrationWithRclone	MEDIUM

永続的

これらのルールは、ユーザーアカウント、SSH 認可キー、スケジュールされたジョブ、システムサービスの変更など、リソースへのアクセスを維持する変更を検出しようとします。

結果タイプ	ルール名	緊急度
Persistence:Runtime/SensitiveFileModified	PAMConfigModification	MEDIUM
Persistence:Runtime/SensitiveFileModified	PasswordModification	MEDIUM
Persistence:Runtime/SensitiveFileModified	SSHAuthorizedKeysModification	MEDIUM
Persistence:Runtime/SensitiveFileModified	ScheduledJobManipulation	MEDIUM
Persistence:Runtime/SensitiveFileModified	ShellRCFileManipulated	MEDIUM
Persistence:Runtime/SensitiveFileModified	SystemdServiceManipulation	MEDIUM
Persistence:Runtime/SuspiciousCommand	AuditdKeylogger	MEDIUM
Persistence:Runtime/SuspiciousCommand	InteractiveLocalAccountCreation	MEDIUM
Persistence:Runtime/SuspiciousCommand	NobodyUserManipulation	HIGH
Persistence:Runtime/SuspiciousCommand	SshFromPublicIPAddress	MEDIUM
Persistence:Runtime/SuspiciousCommand	SystemdTransientServiceCreated	MEDIUM

権限昇格

これらのルールは、sudoers ファイルの変更、setuid フラグの設定、nsenter を使用したコンテナのエスケープなど、高レベルのアクセス許可を付与する変更を検出しようとします。

結果タイプ	ルール名	緊急度
DefenseEvasion:Runtime/SuspiciousCommand	FindCommandReconSetuidSetgidFiles	MEDIUM

結果タイプ	ルール名	緊急度
PrivilegeEscalation:Runtime/SensitiveFileModified	SudoerFileLinked	MEDIUM
PrivilegeEscalation:Runtime/SensitiveFileModified	SudoerFileManipulation	MEDIUM
PrivilegeEscalation:Runtime/SensitiveFileModified	SudoerFileRenamed	MEDIUM
PrivilegeEscalation:Runtime/SuspiciousCommand	AddSetUIDCapability	HIGH
PrivilegeEscalation:Runtime/SuspiciousCommand	NewUserAddedToRootGroup	HIGH
PrivilegeEscalation:Runtime/SuspiciousCommand	NsenterContainerEscape	HIGH
PrivilegeEscalation:Runtime/SuspiciousCommand	SetUidFlagSet	MEDIUM
PrivilegeEscalation:Runtime/SuspiciousCommand	SudoPrivilegeEnumeration	MEDIUM
PrivilegeEscalation:Runtime/SuspiciousCommand	SudoerFileRead	MEDIUM

Stealth

これらのルールは、LD_PRELOAD を使用したライブラリ関数のハイジャックや Base64 を使用したデータのエンコードなど、アクティビティを隠すプロセスを検出しようとします。

結果タイプ	ルール名	緊急度
DefenseEvasion:Runtime/SensitiveFileModified	LDPreloadFunctionHijack	MEDIUM
DefenseEvasion:Runtime/SuspiciousCommand	Base64EncodeDecode	MEDIUM

結果タイプ	ルール名	緊急度
DefenseEvasion:Runtime/SuspiciousCommand	BpftoolProgramLoaded	MEDIUM
DefenseEvasion:Runtime/SuspiciousCommand	DebugfsRawDiskAccess	MEDIUM
DefenseEvasion:Runtime/SuspiciousCommand	SuspiciousExecutionPath	MEDIUM

GuardDuty S3 Protection の検出結果タイプ

次の検出結果は、Amazon S3リソースに特有のもので、データソースが S3 CloudTrail データイベントの場合、またはデータソースが CloudTrail 管理イベントや AccessKey の場合、リソースタイプは S3Bucket となります。検出結果の重大度と詳細は、検出結果タイプとバケットに関連付けられている許可によって異なります。

ここにリストされている検出結果には、検出結果タイプの生成に使用されるデータソースとモデルが含まれます。データソースとモデルの詳細については、「GuardDuty 基本データソース」を参照してください。

> **Important**
>
> S3 CloudTrail データイベントのデータソースを持つ検出結果は、S3 Protection を有効にしている場合のみ生成されます。デフォルトでは、2020 年 7 月 31 日以降、アカウントが初めて GuardDuty を有効にするか、委任 GuardDuty 管理者アカウントが既存のメンバーアカウントで GuardDuty を有効にすると、S3 Protection が有効になります。ただし、新しいメンバーが GuardDuty 組織に参加するときは、組織の自動有効化の詳細設定が適用されます。自動有効化の詳細設定については、「組織の自動有効化の詳細設定の指定」を参照してください。S3 Protection を有効にする方法については、「GuardDuty S3 Protection」を参照してください。

すべての S3Bucket タイプの検出結果では、問題のバケットに対するアクセス権限と検出結果に関係するユーザーのアクセス権限を確認することをお勧めします。予期しないアクティビティについては、「侵害された可能性のある S3 バケットの修復」に記載されている修復レコメンデーションを参照してください。

トピック

- Discovery:S3/AnomalousBehavior
- Discovery:S3/MaliciousIPCaller
- Discovery:S3/MaliciousIPCaller.Custom
- Discovery:S3/TorIPCaller
- Exfiltration:S3/AnomalousBehavior
- Exfiltration:S3/MaliciousIPCaller
- Impact:S3/AnomalousBehavior.Delete
- Impact:S3/AnomalousBehavior.Permission
- Impact:S3/AnomalousBehavior.Write
- Impact:S3/MaliciousIPCaller
- PenTest:S3/KaliLinux
- PenTest:S3/ParrotLinux
- PenTest:S3/PentooLinux
- Policy:S3/AccountBlockPublicAccessDisabled
- Policy:S3/BucketAnonymousAccessGranted
- Policy:S3/BucketBlockPublicAccessDisabled
- Policy:S3/BucketPublicAccessGranted
- Stealth:S3/ServerAccessLoggingDisabled
- UnauthorizedAccess:S3/MaliciousIPCaller.Custom
- UnauthorizedAccess:S3/TorIPCaller

Discovery:S3/AnomalousBehavior

S3 オブジェクトの検出に一般的に使用される API が、異常な方法で呼び出されました。

デフォルトの重大度: [Low] (低

- データソース: S3 CloudTrail データイベント

この検出結果は、IAM エンティティが S3 API を呼び出して、環境内の `ListObjects` などの S3 バケットを検出したことを知らせるものです。このタイプのアクティビティは、攻撃者が情報を収集し

て AWS 、環境がより広範な攻撃の影響を受けやすいかどうかを判断する攻撃の検出段階に関連付けられています。IAM エンティティが異常な方法で API を呼び出したため、このアクティビティは疑わしいものです。例えば、過去の履歴がない IAM エンティティが S3 API を呼び出したり、IAM エンティティが通常とは異なる場所から S3 API 呼び出したりした場合です。

この API は、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。機械学習モデルは、アカウント内のすべての API リクエストを評価し、攻撃者が使用するテクニックに関連する異常なイベントを特定します。リクエストを行ったユーザー、リクエストが行われた場所、リクエストされた特定の API、リクエストされたバケット、および API 呼び出しの数など、API リクエストのさまざまな要因を追跡します。リクエストを呼び出したユーザーアイデンティティにおいて API リクエストのどの要因が異常なのかという詳細については、「検出結果の詳細」を参照してください。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

Discovery:S3/MaliciousIPCaller

AWS 環境内のリソースを検出するために一般的に使用される S3 API が、既知の悪意のある IP アドレスから呼び出されました。

デフォルトの重大度: [High] (高

- データソース: S3 CloudTrail データイベント

この検出結果は、S3 API オペレーションが悪意のある既知のアクティビティと関連した IP アドレスから呼び出されたことを知らせるものです。観察された API は、攻撃者が AWS 環境に関する情報を収集している場合に、攻撃の検出段階に一般的に関連します。例には、`GetObjectAcl` や `ListObjects` が含まれます。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

Discovery:S3/MaliciousIPCaller.Custom

S3 API がカスタム脅威リストにある IP アドレスから呼び出されました。

デフォルトの重大度: [High] (高

- データソース: S3 CloudTrail データイベント

この検出結果は、`GetObjectAcl` または `ListObjects` などの S3 API が、アップロード済みの脅威リストに含まれている IP アドレスから呼び出されたことを知らせています。この検出結果に関連する脅威リストは、検出結果の詳細の [Additional information] (追加情報) セクションにリストされています。このタイプのアクティビティは攻撃の検出段階に関連しており、攻撃者は、 AWS 環境がより広範な攻撃を受けやすいかどうかを判断するために情報を収集しています。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

Discovery:S3/TorIPCaller

S3 API が Tor 出口ノードの IP アドレスから呼び出されました。

デフォルトの重大度: [Medium] (中

- データソース: S3 CloudTrail データイベント

この検出結果は、`GetObjectAcl` や `ListObjects` などの S3 API が Tor 出口ノードの IP アドレスから呼び出されたことを知らせるものです。このタイプのアクティビティは、攻撃者が情報を収集して AWS 、環境がより広範な攻撃の影響を受けやすいかどうかを判断する攻撃の検出段階に関連付けられています。Tor は、匿名通信を有効にするためのソフトウェアです。通信を暗号化し、一連のネットワークノード間のリレー中にランダムに通信をバウンスさせます。最後の Tor ノードは終了ノードと呼ばれます。これは、攻撃者の真のアイデンティティを隠す目的で AWS リソースへの不正アクセスを示している可能性があります。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

Exfiltration:S3/AnomalousBehavior

IAM エンティティが疑わしい方法で S3 API を呼び出しました。

デフォルトの重大度: [High] (高

- データソース: S3 CloudTrail データイベント

この検出結果から、IAM エンティティが S3 バケットの関連している API コールを行い、このアクティビティがそのエンティティの確立されたベースラインと異なっていることがわかります。このアクティビティで使用された API コールは、攻撃の抽出段階に関連付けられており、攻撃者はデータを収集します。IAM エンティティが異常な方法で API を呼び出したため、このアクティビティは疑わしいものです。例えば、過去の履歴がない IAM エンティティが S3 API を呼び出したり、IAM エンティティが通常とは異なる場所から S3 API 呼び出したりした場合です。

この API は、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。機械学習モデルは、アカウント内のすべての API リクエストを評価し、攻撃者が使用するテクニックに関連する異常なイベントを特定します。リクエストを行ったユーザー、リクエストが行われた場所、リクエストされた特定の API、リクエストされたバケット、および API 呼び出しの数など、API リクエストのさまざまな要因を追跡します。リクエストを呼び出したユーザーアイデンティティにおいて API リクエストのどの要因が異常なのかという詳細については、「検出結果の詳細」を参照してください。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

Exfiltration:S3/MaliciousIPCaller

AWS 環境からデータを収集するために一般的に使用される S3 API が、既知の悪意のある IP アドレスから呼び出されました。

デフォルトの重大度: [High] (高

- データソース: S3 CloudTrail データイベント

この検出結果は、S3 API オペレーションが悪意のある既知のアクティビティと関連した IP アドレスから呼び出されたことを知らせるものです。観察された API は、攻撃者がネットワークからデータを収集しようとしている侵入戦術に一般的に関連しています。例には、`GetObject` や `CopyObject` が含まれます。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

Impact:S3/AnomalousBehavior.Delete

IAM エンティティが疑わしい方法でデータを削除をしようとした S3 API を呼び出しました。

デフォルトの重大度: [High] (高

- データソース: S3 CloudTrail データイベント

この検出結果は、 AWS 環境内の IAM エンティティが S3 バケットを含む API コールを行っていることを知らせるもので、この動作はエンティティの確立されたベースラインとは異なります。このアクティビティで使用された API コールは、データを削除しようとする攻撃に関連付けられています。IAM エンティティが異常な方法で API を呼び出したため、このアクティビティは疑わしいものです。例えば、過去の履歴がない IAM エンティティが S3 API を呼び出したり、IAM エンティティが通常とは異なる場所から S3 API 呼び出したりした場合です。

この API は、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。機械学習モデルは、アカウント内のすべての API リクエストを評価し、攻撃者が使用するテクニックに関連する異常なイベントを特定します。リクエストを行ったユーザー、リクエストが行われた場所、リクエストされた特定の API、リクエストされたバケット、および API 呼び出しの数など、API リクエストのさまざまな要因を追跡します。リクエストを呼び出したユーザーアイデンティティにおいて API リクエストのどの要因が異常なのかという詳細については、「検出結果の詳細」を参照してください。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

S3 バケットのコンテンツを監査して、以前のオブジェクトバージョンを復元できるかどうか、または復元する必要があるかどうかを判断することをお勧めします。

Impact:S3/AnomalousBehavior.Permission

アクセスコントロールリスト (ACL) のアクセス許可設定に一般的に使用される API が異常な方法で呼び出されました。

デフォルトの重大度: [High] (高

- データソース: S3 CloudTrail データイベント

この検出結果は、 AWS 環境内の IAM エンティティが、リストされた S3 バケットのバケットポリシーまたは ACL を更新したことを知らせるものです。この変更により、認証されたすべての AWS ユーザーに S3 バケットが公開される可能性があります。

この API は、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。機械学習モデルは、アカウント内のすべての API リクエストを評価し、攻撃者が使用するテクニックに関連する異常なイベントを特定します。リクエストを行ったユーザー、リクエストが行われた場所、リクエストされた特定の API、リクエストされたバケット、および API 呼び出しの数など、API リクエストのさまざまな要因を追跡します。リクエストを呼び出したユーザーアイデンティティにおいて API リクエストのどの要因が異常なのかという詳細については、「検出結果の詳細」を参照してください。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

S3 バケットのコンテンツを監査して、オブジェクトが予期せずパブリックアクセスを許可されていなかったか確認することをお勧めします。

Impact:S3/AnomalousBehavior.Write

IAM エンティティが疑わしい方法でデータの書き込みをしようとした S3 API を呼び出しました。

デフォルトの重大度: [Medium] (中

- データソース: S3 CloudTrail データイベント

この検出結果は、 AWS 環境内の IAM エンティティが S3 バケットを含む API コールを行っていることを知らせるもので、この動作はエンティティの確立されたベースラインとは異なります。このアクティビティで使用された API コールは、データを書き込もうとする攻撃に関連付けられています。IAM エンティティが異常な方法で API を呼び出したため、このアクティビティは疑わしいものです。例えば、過去の履歴がない IAM エンティティが S3 API を呼び出したり、IAM エンティティが通常とは異なる場所から S3 API 呼び出したりした場合です。

この API は、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。機械学習モデルは、アカウント内のすべての API リクエストを評価し、攻撃者が使用するテクニックに関連する異常なイベントを特定します。リクエストを行ったユーザー、リクエストが行われた場所、リクエストされた特定の API、リクエストされたバケット、および API 呼び出しの数など、API リクエストのさまざまな要因を追跡します。リクエストを呼び出したユーザーアイデンティティにおいて API リクエストのどの要因が異常なのかという詳細については、「検出結果の詳細」を参照してください。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

S3 バケットのコンテンツを監査して、この API 呼び出しが悪意のあるデータや不正なデータを書き込んでいなかったか確認することをお勧めします。

Impact:S3/MaliciousIPCaller

AWS 環境内のデータまたはプロセスを改ざんするために一般的に使用される S3 API が、既知の悪意のある IP アドレスから呼び出されました。

デフォルトの重大度: [High] (高

- データソース: S3 CloudTrail データイベント

この検出結果は、S3 API オペレーションが悪意のある既知のアクティビティと関連した IP アドレスから呼び出されたことを知らせるものです。観測された API は、攻撃者が AWS 環境内のデータを操作、中断、または破壊しようとしている影響戦術に一般的に関連します。例には、`PutObject` や `PutObjectAcl` が含まれます。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

PenTest:S3/KaliLinux

S3 API が Kali Linux マシンから呼び出されました。

デフォルトの重大度: [Medium] (中

- データソース: S3 CloudTrail データイベント

この検出結果は、Kali Linux を実行しているマシンが、 AWS アカウントに属する認証情報を使用して S3 API コールを行っていることを知らせるものです。認証情報が侵害されている可能性があります。Kali Linux は、セキュリティプロフェッショナルがパッチ適用が必要な EC2 インスタンスの弱点を特定するために使用する一般的な侵入テストツールです。また、攻撃者はこのツールを使用して EC2 設定の弱点を見つけ、 AWS 環境への不正アクセスを取得します。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

PenTest:S3/ParrotLinux

S3 API が Parrot Security Linux マシンから呼び出されました。

デフォルトの重大度: [Medium] (中

- データソース: S3 CloudTrail データイベント

この検出結果は、Parrot Security Linux を実行しているマシンが、 AWS アカウントに属する認証情報を使用して S3 API コールを行っていることを知らせるものです。認証情報が侵害されている可能性があります。Parrot Security Linux は、セキュリティプロフェッショナルが EC2 インスタンスの脆弱性を特定してパッチを適用するために使う一般的な侵入テストツールです。このツールを攻撃者が利用して EC2 設定の脆弱性を探り出し、 AWS 環境への未承認のアクセスを取得する場合があります。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

PenTest:S3/PentooLinux

S3 API が Pentoo Linux マシンから呼び出されました。

デフォルトの重大度: [Medium] (中

- データソース: S3 CloudTrail データイベント

この検出結果は、Pentoo Linux を実行しているマシンが、 AWS アカウントに属する認証情報を使用して S3 API コールを行っていることを知らせるものです。認証情報が侵害されている可能性があります。Pentoo Linux は、セキュリティプロフェッショナルがパッチ適用を必要とする EC2 インスタンスの弱点を特定するために使用する一般的な侵入テストツールです。また、攻撃者はこのツールを使用して EC2 設定の弱点を見つけ、 AWS 環境への不正アクセスを取得します。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

Policy:S3/AccountBlockPublicAccessDisabled

IAM エンティティが、アカウント上の S3 ブロックパブリックアクセスを無効にするために使用される API を呼び出しました。

デフォルトの重大度: [Low] (低

- データソース CloudTrail 管理イベント

この検出結果は、Amazon S3 ブロックパブリックアクセスがアカウントレベルで無効されたことを知らせるものです。S3 ブロックパブリックアクセス設定が有効化されると、それらはデータが誤って公開されるのを防ぐためのセキュリティ対策として、バケット上でポリシー、またはアクセスコントロールリスト (ACL) をフィルタリングするために使われます。

バケット内、またはバケット内のオブジェクトへの公開アクセスを許可するために、通常、S3 ブロックパブリックアクセスは、アカウントでは無効になっています。アカウントのために S3 ブロックパブリックアクセスを無効にすると、バケットへのアクセスは、個々のバケットに適用されるポリシー、ACL、またはバケットレベルのパブリックアクセス設定によって制御されます。これは、必ずしもバケットがパブリックに共有されているということではありませんが、アクセスへの適切なレベルが提供されているのを確認するために、バケットに対して適用される許可を監査する必要があります。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

Policy:S3/BucketAnonymousAccessGranted

IAM プリンシパルは、バケットポリシーまたは ACL を変更することにより、インターネットへの S3 バケットへのアクセスを許可しています。

デフォルトの重大度: [High] (高

- データソース CloudTrail 管理イベント

この検出結果から、IAM エンティティが、そのバケット上のバケットポリシーまたは ACL を変更したため、リストされた S3 バケットがインターネット上でパブリックにアクセス可能になったことを知らせるものです。

ポリシーや ACL の変更が検出されると、GuardDuty は、Zelkova による自動推論を使用して、バケットがパブリックにアクセス可能であるかどうかを確認します。

> **Note**
>
> バケットの ACL またはバケットポリシーが明示的に拒否またはすべてを拒否するように設定されている場合、この検出結果はバケットの現在の状態を反映していない可能性があります。この検出結果には、S3 バケットで有効になっている可能性のある S3 ブロックパブリックアクセス設定は反映されません。このような場合、検出結果の `effectivePermission` 値には `UNKNOWN` としてマークされます。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

Policy:S3/BucketBlockPublicAccessDisabled

IAM エンティティが、バケットの S3 ブロックパブリックアクセスを無効にするために使用される API を呼び出しました。

デフォルトの重大度: [Low] (低

- データソース CloudTrail 管理イベント

この検出結果から、リストされた S3 バケットのために、ブロックパブリックアクセスが無効になったことを知らせるものです。有効にすると、S3 ブロックパブリックアクセス設定が使われ、バケットに適用されるポリシーまたは アクセスコントロールリスト (ACL) をフィルタリングし、データが誤って公開される安全対策となります。

通常、S3 ブロックパブリックアクセスは、バケット上で無効にされ、バケットまたはバケット内のオブジェクトへの公開アクセスを許可します。S3 ブロックパブリックアクセスがこのバケットに対して無効になっている場合、バケットに適用されるポリシーまたはそれに適用される ACL によって、バケットへのアクセスが制御されます。これは、バケットがパブリックに共有されているということではありませんが、バケットに適用されているポリシーと ACL を監査して、適切な許可が適用されていることを確認する必要があります。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

Policy:S3/BucketPublicAccessGranted

IAM プリンシパルは、バケットポリシーまたは ACL を変更することで、すべての AWS ユーザーに S3 バケットへのパブリックアクセスを許可しています。 ACLs

デフォルトの重大度: [High] (高

- データソース CloudTrail 管理イベント

この検出結果は、IAM エンティティがその S3 バケットのバケットポリシーまたは ACL を変更したために、リストされた S3 バケットがすべての認証済み AWS ユーザーに公開されたことを知らせるものです。

ポリシーや ACL の変更が検出されると、GuardDuty は、Zelkova による自動推論を使用して、バケットがパブリックにアクセス可能であるかどうかを確認します。

> **Note**
>
> バケットの ACL またはバケットポリシーが明示的に拒否またはすべてを拒否するように設定されている場合、この検出結果はバケットの現在の状態を反映していない可能性があります。この検出結果には、S3 バケットで有効になっている可能性のある S3 ブロックパブリックアクセス設定は反映されません。このような場合、検出結果の `effectivePermission` 値には `UNKNOWN` としてマークされます。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

Stealth:S3/ServerAccessLoggingDisabled

バケットのために S3 サーバーアクセスのログ記録が無効になりました。

デフォルトの重大度: [Low] (低

- データソース CloudTrail 管理イベント

この検出結果は、 AWS 環境内のバケットに対して S3 サーバーアクセスログ記録が無効になっていることを知らせるものです。無効にした場合、識別された S3 バケットにアクセスしようとする試みに対してウェブリクエストログは作成されません。ただし、DeleteBucket などのバケットへの S3 管理 API コールは、まだ追跡されています。このバケットの CloudTrail を介して S3 データイベントログが有効になっている場合、バケット内のオブジェクトに対するウェブリクエストは引き続き追跡されます。ログ記録の無効化は、許可されていないユーザーがその形跡を隠すために頻繁に使用する手法です。S3 ログの詳細については、「S3 サーバーアクセスログ」と「S3 ログオプション」を参照してください。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

UnauthorizedAccess:S3/MaliciousIPCaller.Custom

S3 API がカスタム脅威リストにある IP アドレスから呼び出されました。

デフォルトの重大度: [High] (高

- データソース: S3 CloudTrail データイベント

この検出結果は、アップロードした脅威リストに含まれたIP アドレスから S3 API オペレーション (`PutObject` や `PutObjectAcl` など) が呼び出されたことを知らせるものです。この検出結果に関連する脅威リストは、検出結果の詳細の [Additional information] (追加情報) セクションにリストされています。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

UnauthorizedAccess:S3/TorIPCaller

S3 API が Tor 出口ノードの IP アドレスから呼び出されました。

デフォルトの重大度: [High] (高

- データソース: S3 CloudTrail データイベント

この検出結果は、Tor 出口ノードの IP アドレスから S3 API オペレーション (`PutObject` や `PutObjectAcl` など) が呼び出されたことを知らせるものです。Tor は匿名通信を有効化するソフトウェアです。通信を暗号化し、一連のネットワークノード間のリレー中にランダムに通信をバウンスさせます。最後の Tor ノードは終了ノードと呼ばれます。この検出結果は、攻撃者の真のアイデンティティを隠す目的で AWS 、リソースへの不正アクセスを示している可能性があります。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

EKS Protection の検出結果タイプ

次の検出結果は Amazon EKS リソースに固有であり、resource_type は `EKSCluster` です。検出結果の重大度と詳細は、検出結果タイプによって異なります。

すべての EKS 監査ログタイプの検出結果については、問題のリソースを調べて、アクティビティが想定されたものであるのか、または悪意あるアクティビティである可能性があるのかを判断することをお勧めします。GuardDuty の検出結果によって特定された、侵害された EKS 監査ログリソースを修復するためのガイダンスについては、「EKS Protection の検出結果の修復」を参照してください。

> **Note**
>
> これらの結果を生成する原因となるアクティビティが想定される場合は、今後アラートが発生しないように GuardDuty の抑制ルール を追加することを検討してください。

トピック

- CredentialAccess:Kubernetes/MaliciousIPCaller

- CredentialAccess:Kubernetes/MaliciousIPCaller.Custom
- CredentialAccess:Kubernetes/SuccessfulAnonymousAccess
- CredentialAccess:Kubernetes/TorIPCaller
- DefenseEvasion:Kubernetes/MaliciousIPCaller
- DefenseEvasion:Kubernetes/MaliciousIPCaller.Custom
- DefenseEvasion:Kubernetes/SuccessfulAnonymousAccess
- DefenseEvasion:Kubernetes/TorIPCaller
- Discovery:Kubernetes/MaliciousIPCaller
- Discovery:Kubernetes/MaliciousIPCaller.Custom
- Discovery:Kubernetes/SuccessfulAnonymousAccess
- Discovery:Kubernetes/TorIPCaller
- Execution:Kubernetes/ExecInKubeSystemPod
- Impact:Kubernetes/MaliciousIPCaller
- Impact:Kubernetes/MaliciousIPCaller.Custom
- Impact:Kubernetes/SuccessfulAnonymousAccess
- Impact:Kubernetes/TorIPCaller
- Persistence:Kubernetes/MaliciousIPCaller
- Persistence:Kubernetes/MaliciousIPCaller.Custom
- Persistence:Kubernetes/SuccessfulAnonymousAccess
- Persistence:Kubernetes/TorIPCaller
- Policy:Kubernetes/AdminAccessToDefaultServiceAccount
- Policy:Kubernetes/AnonymousAccessGranted
- Policy:Kubernetes/ExposedDashboard
- Policy:Kubernetes/KubeflowDashboardExposed
- CredentialAccess:Kubernetes/AnomalousBehavior.SecretsAccessed
- PrivilegeEscalation:Kubernetes/AnomalousBehavior.RoleBindingCreated
- Execution:Kubernetes/AnomalousBehavior.ExecInPod
- PrivilegeEscalation:Kubernetes/AnomalousBehavior.WorkloadDeployed!PrivilegedContainer
- Persistence:Kubernetes/AnomalousBehavior.WorkloadDeployed!ContainerWithSensitiveMount
- Execution:Kubernetes/AnomalousBehavior.WorkloadDeployed

- PrivilegeEscalation:Kubernetes/AnomalousBehavior.RoleCreated
- Discovery:Kubernetes/AnomalousBehavior.PermissionChecked

> **Note**
>
> Kubernetes バージョン 1.14 より前では、`system:unauthenticated` グループは、デフォルトで `system:discovery` と `system:basic-user` ClusterRoles に関連付けられていました。この関連付けは、匿名ユーザーからの意図しないアクセスを許可する場合があります。クラスターの更新では、これらの許可は取り消されません。クラスターをバージョン 1.14 以降に更新した場合でも、これらの許可は引き続き有効になっている可能性があります。これらの許可の関連付けを `system:unauthenticated` グループから解除することをお勧めします。これらのアクセス許可を取り消す方法のガイダンスについては、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」を参照してください。

CredentialAccess:Kubernetes/MaliciousIPCaller

Kubernetes クラスターの認証情報またはシークレットにアクセスするために一般的に使用される API が悪意のある既知の IP アドレスから呼び出されました。

デフォルトの重大度: [High] (高

- 機能: EKS 監査ログ

この検出結果は、API オペレーションが悪意のある既知のアクティビティと関連した IP アドレスから呼び出されたことを知らせるものです。観察された API は、攻撃者が Kubernetes クラスターのパスワード、ユーザー名、およびアクセスキーの収集を試みている認証情報アクセス戦術に一般的に関連付けられています。

修復のレコメンデーション

`KubernetesUserDetails` セクションの検出結果で報告されたユーザーが `system:anonymous` である場合は、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」の指示に従って、匿名ユーザーが API の呼び出しを許可された理由を調べ、必要に応じてアクセス許可を取り消します。ユーザーが認証されたユーザーである場合は、そのアクティビティが正当なものであるのか、または悪意のあるものであるのかを調査します。アクティビティが悪意のあ

るものである場合は、ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

CredentialAccess:Kubernetes/MaliciousIPCaller.Custom

Kubernetes クラスターの認証情報またはシークレットにアクセスするために一般的に使用される API が、カスタム脅威リストの IP アドレスから呼び出されました。

デフォルトの重大度: [High] (高

- 機能: EKS 監査ログ

この検出結果は、API オペレーションが、アップロード済みの脅威リストに含まれている IP アドレスから呼び出されたことを知らせるものです。この検出結果に関連する脅威リストは、検出結果の詳細の [Additional Information] (追加情報) セクションにリストされています。観察された API は、攻撃者が Kubernetes クラスターのパスワード、ユーザー名、およびアクセスキーの収集を試みている認証情報アクセス戦術に一般的に関連付けられています。

修復のレコメンデーション

`KubernetesUserDetails` セクションの検出結果で報告されたユーザーが `system:anonymous` である場合は、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」の指示に従って、匿名ユーザーが API の呼び出しを許可された理由を調べ、必要に応じてアクセス許可を取り消します。ユーザーが認証されたユーザーである場合は、そのアクティビティが正当なものであるのか、または悪意のあるものであるのかを調査します。アクティビティが悪意のあるものである場合は、ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

CredentialAccess:Kubernetes/SuccessfulAnonymousAccess

Kubernetes クラスターの認証情報またはシークレットにアクセスするために一般的に使用される API が、認証されていないユーザーによって呼び出されました。

デフォルトの重大度: [High] (高

- 機能: EKS 監査ログ

この検出結果は、API オペレーションが `system:anonymous` ユーザーによって正常に呼び出されたことを知らせるものです。`system:anonymous` によって実行された API コールは認証されてい

ません。観察された API は、攻撃者が Kubernetes クラスターのパスワード、ユーザー名、およびアクセスキーの収集を試みている認証情報アクセス戦術に一般的に関連付けられています。このアクティビティは、検出結果で報告された API アクションで匿名アクセスまたは非認証アクセスが許可され、他のアクションで許可される可能性があることを示します。この動作が想定されていない場合は、設定ミスがあるか、または認証情報が侵害されている可能性があります。

修復のレコメンデーション

クラスター上の `system:anonymous` ユーザーに付与されている許可を調べて、すべての許可が必要であることを確認する必要があります。誤って、または悪意を持って許可が付与された場合は、ユーザーのアクセスを取り消し、クラスターに対して攻撃者が加えた変更を元に戻す必要があります。詳細については、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」を参照してください。

詳細については、「EKS Protection の検出結果の修復」を参照してください。

CredentialAccess:Kubernetes/TorIPCaller

Kubernetes クラスターの認証情報またはシークレットにアクセスするために一般的に使用される API が、Tor 出口ノードの IP アドレスから呼び出されました。

デフォルトの重大度: [High] (高

- 機能: EKS 監査ログ

この検出結果は、API が Tor 出口ノードの IP アドレスから呼び出されたことを知らせるものです。観察された API は、攻撃者が Kubernetes クラスターのパスワード、ユーザー名、およびアクセスキーの収集を試みている認証情報アクセス戦術に一般的に関連付けられています。Tor は、匿名通信を有効にするためのソフトウェアです。通信を暗号化し、一連のネットワークノード間のリレー中にランダムに通信をバウンスさせます。最後の Tor ノードは出口ノードと呼ばれます。これは、攻撃者が真のアイデンティティを隠す意図を持った、Kubernetes クラスターのリソースへの未承認のアクセスを示している場合があります。

修復のレコメンデーション

`KubernetesUserDetails` セクションの検出結果で報告されたユーザーが `system:anonymous` である場合は、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」の指示に従って、匿名ユーザーが API の呼び出しを許可された理由を調べ、必要に応じてアクセス許可を取り消します。ユーザーが認証されたユーザーである場合は、そのアクティビティが

正当なものであるのか、または悪意のあるものであるのかを調査します。アクティビティが悪意のあるものである場合は、ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

DefenseEvasion:Kubernetes/MaliciousIPCaller

防御対策を回避するために一般的に使用される API が、悪意のある既知の IP アドレスから呼び出されました。

デフォルトの重大度: [High] (高

- 機能: EKS 監査ログ

この検出結果は、API オペレーションが悪意のある既知のアクティビティと関連した IP アドレスから呼び出されたことを知らせるものです。観察された API は一般に、攻撃者が検出を回避するためにアクションの隠ぺいを試みている防御回避戦術に関連付けられています。

修復のレコメンデーション

`KubernetesUserDetails` セクションの検出結果で報告されたユーザーが `system:anonymous` である場合は、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」の指示に従って、匿名ユーザーが API の呼び出しを許可された理由を調べ、必要に応じてアクセス許可を取り消します。ユーザーが認証されたユーザーである場合は、そのアクティビティが正当なものであるのか、または悪意のあるものであるのかを調査します。アクティビティが悪意のあるものである場合は、ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

DefenseEvasion:Kubernetes/MaliciousIPCaller.Custom

防御対策を回避するために一般的に使用される API が、カスタム脅威リストにある IP アドレスから呼び出されました。

デフォルトの重大度: [High] (高

- 機能: EKS 監査ログ

この検出結果は、API オペレーションが、アップロード済みの脅威リストに含まれている IP アドレスから呼び出されたことを知らせるものです。この検出結果に関連する脅威リストは、検出結果の

詳細の [Additional Information] (追加情報) セクションにリストされています。観察された API は一般に、攻撃者が検出を回避するためにアクションの隠ぺいを試みている防御回避戦術に関連付けられています。

修復のレコメンデーション

KubernetesUserDetails セクションの検出結果で報告されたユーザーが system:anonymous である場合は、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」の指示に従って、匿名ユーザーが API の呼び出しを許可された理由を調べ、必要に応じてアクセス許可を取り消します。ユーザーが認証されたユーザーである場合は、そのアクティビティが正当なものであるのか、または悪意のあるものであるのかを調査します。アクティビティが悪意のあるものである場合は、ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

DefenseEvasion:Kubernetes/SuccessfulAnonymousAccess

防御対策を回避するために一般的に使用される API が、認証されていないユーザーによって呼び出されました。

デフォルトの重大度: [High] (高

- 機能: EKS 監査ログ

この検出結果は、API オペレーションが system:anonymous ユーザーによって正常に呼び出されたことを知らせるものです。system:anonymous によって実行された API コールは認証されていません。観察された API は一般に、攻撃者が検出を回避するためにアクションの隠ぺいを試みている防御回避戦術に関連付けられています。このアクティビティは、検出結果で報告された API アクションで匿名アクセスまたは非認証アクセスが許可され、他のアクションで許可される可能性があることを示します。この動作が想定されていない場合は、設定ミスがあるか、または認証情報が侵害されている可能性があります。

修復のレコメンデーション

クラスター上の system:anonymous ユーザーに付与されている許可を調べて、すべての許可が必要であることを確認する必要があります。誤って、または悪意を持って許可が付与された場合は、ユーザーのアクセスを取り消し、クラスターに対して攻撃者が加えた変更を元に戻す必要があります。詳細については、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」を参照してください。

詳細については、「EKS Protection の検出結果の修復」を参照してください。

DefenseEvasion:Kubernetes/TorIPCaller

防御対策を回避するために一般的に使用される API が、Tor 出口ノードの IP アドレスから呼び出されました。

デフォルトの重大度: [High] (高

- 機能: EKS 監査ログ

この検出結果は、API が Tor 出口ノードの IP アドレスから呼び出されたことを知らせるものです。観察された API は一般に、攻撃者が検出を回避するためにアクションの隠ぺいを試みている防御回避戦術に関連付けられています。Tor は匿名通信を有効化するソフトウェアです。通信を暗号化し、一連のネットワークノード間のリレー中にランダムに通信をバウンスさせます。最後の Tor ノードは出口ノードと呼ばれます。これは、攻撃者が真のアイデンティティを隠す意図を持った、Kubernetes クラスターへの未承認のアクセスを示している場合があります。

修復のレコメンデーション

`KubernetesUserDetails` セクションの検出結果で報告されたユーザーが `system:anonymous` である場合は、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」の指示に従って、匿名ユーザーが API の呼び出しを許可された理由を調べ、必要に応じてアクセス許可を取り消します。ユーザーが認証されたユーザーである場合は、そのアクティビティが正当なものであるのか、または悪意のあるものであるのかを調査します。アクティビティが悪意のあるものである場合は、ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

Discovery:Kubernetes/MaliciousIPCaller

Kubernetes クラスター内のリソースを見つけるために一般的に使用される API が、ある IP アドレスから呼び出されました。

デフォルトの重大度: [Medium] (中

- 機能: EKS 監査ログ

この検出結果は、API オペレーションが悪意のある既知のアクティビティと関連した IP アドレスから呼び出されたことを知らせるものです。観察される API は、攻撃の検出段階で一般的に使用され

ます。攻撃者は、情報を収集して、Kubernetes クラスターがより広範な攻撃の影響を受けやすいかどうかを判断します。

> **非認証アクセスの場合**
>
> 非認証アクセスでは、MaliciousIPCaller 検出結果は生成されません。SuccessfulAnonymousAccess 検出結果は、非認証アクセスまたは匿名アクセスで生成されます。

修復のレコメンデーション

`KubernetesUserDetails` セクションの検出結果で報告されたユーザーが `system:anonymous` である場合は、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」の指示に従って、匿名ユーザーが API の呼び出しを許可された理由を調べ、必要に応じてアクセス許可を取り消します。ユーザーが認証されたユーザーである場合は、そのアクティビティが正当なものであるのか、または悪意のあるものであるのかを調査します。アクティビティが悪意のあるものである場合は、ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

Discovery:Kubernetes/MaliciousIPCaller.Custom

Kubernetes クラスターのリソースを見つけるために一般的に使用される API が、カスタム脅威リストにある IP アドレスから呼び出されました。

デフォルトの重大度: [Medium] (中

- 機能: EKS 監査ログ

この検出結果は、API が、アップロード済みの脅威リストに含まれている IP アドレスから呼び出されたことを知らせるものです。この検出結果に関連する脅威リストは、検出結果の詳細の [Additional Information] (追加情報) セクションにリストされています。観察される API は、攻撃の検出段階で一般的に使用されます。攻撃者は、情報を収集して、Kubernetes クラスターがより広範な攻撃の影響を受けやすいかどうかを判断します。

修復のレコメンデーション

`KubernetesUserDetails` セクションの検出結果で報告されたユーザーが `system:anonymous` である場合は、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラク

ティス」の指示に従って、匿名ユーザーが API の呼び出しを許可された理由を調べ、必要に応じてアクセス許可を取り消します。ユーザーが認証されたユーザーである場合は、そのアクティビティが正当なものであるのか、または悪意のあるものであるのかを調査します。アクティビティが悪意のあるものである場合は、ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

Discovery:Kubernetes/SuccessfulAnonymousAccess

Kubernetes クラスターのリソースを見つけるために一般的に使用される API が、認証されていないユーザーによって呼び出されました。

デフォルトの重大度: [Medium] (中

- 機能: EKS 監査ログ

この検出結果は、API オペレーションが `system:anonymous` ユーザーによって正常に呼び出されたことを知らせるものです。`system:anonymous` によって実行された API コールは認証されていません。攻撃者が Kubernetes クラスターで情報を収集しているときに観測された API は、攻撃の検出段階に一般的に関連しています。このアクティビティは、検出結果で報告された API アクションで匿名アクセスまたは非認証アクセスが許可され、他のアクションで許可される可能性があることを示します。この動作が想定されていない場合は、設定ミスがあるか、または認証情報が侵害されている可能性があります。

この検出結果タイプでは、`/healthz`、`/livez`、`/readyz`、`/version` などのヘルスチェック API エンドポイントは除外されます。

修復のレコメンデーション

クラスター上の `system:anonymous` ユーザーに付与されている許可を調べて、すべての許可が必要であることを確認する必要があります。誤って、または悪意を持って許可が付与された場合は、ユーザーのアクセスを取り消し、クラスターに対して攻撃者が加えた変更を元に戻す必要があります。詳細については、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」を参照してください。

詳細については、「EKS Protection の検出結果の修復」を参照してください。

Discovery:Kubernetes/TorIPCaller

Kubernetes クラスターのリソースを見つけるために一般的に使用される API が、Tor 出口ノードの IP アドレスから呼び出されました。

デフォルトの重大度: [Medium] (中

- 機能: EKS 監査ログ

この検出結果は、API が Tor 出口ノードの IP アドレスから呼び出されたことを知らせるものです。観察される API は、攻撃の検出段階で一般的に使用されます。攻撃者は、情報を収集して、Kubernetes クラスターがより広範な攻撃の影響を受けやすいかどうかを判断します。Tor は匿名通信を有効化するソフトウェアです。通信を暗号化し、一連のネットワークノード間のリレー中にランダムに通信をバウンスさせます。最後の Tor ノードは出口ノードと呼ばれます。これは、攻撃者が真のアイデンティティを隠す意図を持った、Kubernetes クラスターへの未承認のアクセスを示している場合があります。

修復のレコメンデーション

`KubernetesUserDetails` セクションの検出結果で報告されたユーザーが `system:anonymous` である場合は、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」の指示に従って、匿名ユーザーが API の呼び出しを許可された理由を調べ、必要に応じてアクセス許可を取り消します。ユーザーが認証されたユーザーである場合は、そのアクティビティが正当なものであるのか、または悪意のあるものであるのかを調査します。アクティビティが悪意のあるものである場合は、ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

Execution:Kubernetes/ExecInKubeSystemPod

kube-system 名前空間内のポッド内でコマンドが実行されました

デフォルトの重大度: [Medium] (中

- 機能: EKS 監査ログ

この検出結果は、コマンドが `kube-system` 名前空間内のポッドで Kubernetes exec API を使用して実行されたことを示すものです。`kube-system` 名前空間はデフォルトの名前空間であり、主に `kube-dns` や `kube-proxy` などのシステムレベルのコンポーネントに使用されます。`kube-`

system 名前空間のポッドまたはコンテナ内でコマンドを実行することは非常にまれであり、疑わしいアクティビティを示している可能性があります。

修復のレコメンデーション

このコマンドの実行が想定されていない場合は、コマンドの実行に使用されたユーザーアイデンティティの認証情報が侵害される可能性があります。ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

Impact:Kubernetes/MaliciousIPCaller

Kubernetes クラスターのリソースを改ざんするために一般的に使用される API が、悪意のある既知の IP アドレスから呼び出されました。

デフォルトの重大度: [High] (高

- 機能: EKS 監査ログ

この検出結果は、API オペレーションが悪意のある既知のアクティビティと関連した IP アドレスから呼び出されたことを知らせるものです。観測された API は、攻撃者が AWS 環境内のデータを操作、中断、または破壊しようとしている影響戦術と一般的に関連しています。

修復のレコメンデーション

KubernetesUserDetails セクションの検出結果で報告されたユーザーが system:anonymous である場合は、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」の指示に従って、匿名ユーザーが API の呼び出しを許可された理由を調べ、必要に応じてアクセス許可を取り消します。ユーザーが認証されたユーザーである場合は、そのアクティビティが正当なものであるのか、または悪意のあるものであるのかを調査します。アクティビティが悪意のあるものである場合は、ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

Impact:Kubernetes/MaliciousIPCaller.Custom

Kubernetes クラスターのリソースを改ざんするために一般的に使用される API が、カスタム脅威リストにある IP アドレスから呼び出されました。

デフォルトの重大度: [High] (高

- 機能: EKS 監査ログ

この検出結果は、API オペレーションが、アップロード済みの脅威リストに含まれている IP アドレスから呼び出されたことを知らせるものです。この検出結果に関連する脅威リストは、検出結果の詳細の [Additional Information] (追加情報) セクションにリストされています。観測された API は、攻撃者が AWS 環境内のデータを操作、中断、または破壊しようとしている影響戦術に一般的に関連しています。

修復のレコメンデーション

KubernetesUserDetails セクションの検出結果で報告されたユーザーが system:anonymous である場合は、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」の指示に従って、匿名ユーザーが API の呼び出しを許可された理由を調べ、必要に応じてアクセス許可を取り消します。ユーザーが認証されたユーザーである場合は、そのアクティビティが正当なものであるのか、または悪意のあるものであるのかを調査します。アクティビティが悪意のあるものである場合は、ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

Impact:Kubernetes/SuccessfulAnonymousAccess

Kubernetes クラスターのリソースを改ざんするために一般的に使用される API が、認証されていないユーザーによって呼び出されました。

デフォルトの重大度: [High] (高

- 機能: EKS 監査ログ

この検出結果は、API オペレーションが system:anonymous ユーザーによって正常に呼び出されたことを知らせるものです。system:anonymous によって実行された API コールは認証されていません。観測された API は、攻撃者がクラスターのリソースを改ざんしているときの攻撃の影響段階に一般的に関連しています。このアクティビティは、検出結果で報告された API アクションで匿名アクセスまたは非認証アクセスが許可され、他のアクションで許可される可能性があることを示します。この動作が想定されていない場合は、設定ミスがあるか、または認証情報が侵害されている可能性があります。

修復のレコメンデーション

クラスター上の system:anonymous ユーザーに付与されている許可を調べて、すべての許可が必要であることを確認する必要があります。誤って、または悪意を持って許可が付与された場合は、

ユーザーのアクセスを取り消し、クラスターに対して攻撃者が加えた変更を元に戻す必要があります。詳細については、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」を参照してください。

詳細については、「EKS Protection の検出結果の修復」を参照してください。

Impact:Kubernetes/TorIPCaller

Kubernetes クラスターのリソースを改ざんするために一般的に使用される API が、Tor 出口ノードの IP アドレスから呼び出されました。

デフォルトの重大度: [High] (高

- 機能: EKS 監査ログ

この検出結果は、API が Tor 出口ノードの IP アドレスから呼び出されたことを知らせるものです。観測された API は、インパクト戦術に一般的に関連していて、ここでは攻撃者が AWS 環境内部でデータを操作、割り込み、または破壊しようとしています。Tor は匿名通信を有効化するソフトウェアです。通信を暗号化し、一連のネットワークノード間のリレー中にランダムに通信をバウンスさせます。最後の Tor ノードは出口ノードと呼ばれます。これは、攻撃者が真のアイデンティティを隠す意図を持った、Kubernetes クラスターへの未承認のアクセスを示している場合があります。

修復のレコメンデーション

`KubernetesUserDetails` セクションの検出結果で報告されたユーザーが `system:anonymous` である場合は、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」の指示に従って、匿名ユーザーが API の呼び出しを許可された理由を調べ、必要に応じてアクセス許可を取り消します。ユーザーが認証されたユーザーである場合は、そのアクティビティが正当なものであるのか、または悪意のあるものであるのかを調査します。アクティビティが悪意のあるものである場合は、ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

Persistence:Kubernetes/MaliciousIPCaller

Kubernetes クラスターへの永続アクセスを取得するために一般的に使用される API が、悪意のある既知の IP アドレスから呼び出されました。

デフォルトの重大度: [Medium] (中

- 機能: EKS 監査ログ

この検出結果は、API オペレーションが悪意のある既知のアクティビティと関連した IP アドレスから呼び出されたことを知らせるものです。観察される API は、攻撃者がユーザーの Kubernetes クラスターへのアクセス権を取得し、そのアクセスを維持しようとする永続化戦術に一般的に関連しています。

修復のレコメンデーション

`KubernetesUserDetails` セクションの検出結果で報告されたユーザーが `system:anonymous` である場合は、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」の指示に従って、匿名ユーザーが API の呼び出しを許可された理由を調べ、必要に応じてアクセス許可を取り消します。ユーザーが認証されたユーザーである場合は、そのアクティビティが正当なものであるのか、または悪意のあるものであるのかを調査します。アクティビティが悪意のあるものである場合は、ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

Persistence:Kubernetes/MaliciousIPCaller.Custom

Kubernetes クラスターへの永続アクセスを取得するために一般的に使用される API が、カスタム脅威リストにある IP アドレスから呼び出されました。

デフォルトの重大度: [Medium] (中

- 機能: EKS 監査ログ

この検出結果は、API オペレーションが、アップロード済みの脅威リストに含まれている IP アドレスから呼び出されたことを知らせるものです。この検出結果に関連する脅威リストは、検出結果の詳細の [Additional Information] (追加情報) セクションにリストされています。観察される API は、攻撃者がユーザーの Kubernetes クラスターへのアクセス権を取得し、そのアクセスを維持しようとする永続化戦術に一般的に関連しています。

修復のレコメンデーション

`KubernetesUserDetails` セクションの検出結果で報告されたユーザーが `system:anonymous` である場合は、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」の指示に従って、匿名ユーザーが API の呼び出しを許可された理由を調べ、必要に応じてアクセス許可を取り消します。ユーザーが認証されたユーザーである場合は、そのアクティビティが

正当なものであるのか、または悪意のあるものであるのかを調査します。アクティビティが悪意のあるものである場合は、ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

Persistence:Kubernetes/SuccessfulAnonymousAccess

Kubernetes クラスターへの高レベルの許可を取得するために一般的に使用される API が、認証されていないユーザーによって呼び出されました。

デフォルトの重大度: [High] (高

- 機能: EKS 監査ログ

この検出結果は、API オペレーションが `system:anonymous` ユーザーによって正常に呼び出されたことを知らせるものです。`system:anonymous` によって実行された API コールは認証されていません。観察される API は、攻撃者がユーザーのクラスターへのアクセス権を取得し、そのアクセスを維持しようとする永続化戦術に一般的に関連しています。このアクティビティは、検出結果で報告された API アクションで匿名アクセスまたは非認証アクセスが許可され、他のアクションで許可される可能性があることを示します。この動作が想定されていない場合は、設定ミスがあるか、または認証情報が侵害されている可能性があります。

修復のレコメンデーション

クラスター上の `system:anonymous` ユーザーに付与されている許可を調べて、すべての許可が必要であることを確認する必要があります。誤って、または悪意を持って許可が付与された場合は、ユーザーのアクセスを取り消し、クラスターに対して攻撃者が加えた変更を元に戻す必要があります。詳細については、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」を参照してください。

詳細については、「EKS Protection の検出結果の修復」を参照してください。

Persistence:Kubernetes/TorIPCaller

Kubernetes クラスターへの永続アクセスを取得するために一般的に使用される API が、Tor 出口ノードの IP アドレスから呼び出されました。

デフォルトの重大度: [Medium] (中

- 機能: EKS 監査ログ

この検出結果は、API が Tor 出口ノードの IP アドレスから呼び出されたことを知らせるものです。観察される API は、攻撃者がユーザーの Kubernetes クラスターへのアクセス権を取得し、そのアクセスを維持しようとする永続化戦術に一般的に関連しています。Tor は匿名通信を有効化するソフトウェアです。通信を暗号化し、一連のネットワークノード間のリレー中にランダムに通信をバウンスさせます。最後の Tor ノードは終了ノードと呼ばれます。これは、攻撃者の真のアイデンティティを隠す目的で、 AWS リソースへの不正アクセスを示している可能性があります。

修復のレコメンデーション

`KubernetesUserDetails` セクションの検出結果で報告されたユーザーが `system:anonymous` である場合は、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」の指示に従って、匿名ユーザーが API の呼び出しを許可された理由を調べ、必要に応じてアクセス許可を取り消します。ユーザーが認証されたユーザーである場合は、そのアクティビティが正当なものであるのか、または悪意のあるものであるのかを調査します。アクティビティが悪意のあるものである場合は、ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

Policy:Kubernetes/AdminAccessToDefaultServiceAccount

デフォルトのサービスアカウントには、Kubernetes クラスターに対する管理者権限が付与されています。

デフォルトの重大度: [High] (高

- 機能: EKS 監査ログ

この検出結果は、Kubernetes クラスター内の名前空間のデフォルトのサービスアカウントに管理者権限が付与されたことを知らせるものです。Kubernetes は、クラスター内のすべての名前空間のためにデフォルトのサービスアカウントを作成します。別のサービスアカウントに明示的に関連付けられていないポッドに、デフォルトのサービスアカウントをアイデンティティとして自動的に割り当てます。デフォルトのサービスアカウントに管理者権限がある場合、ポッドが誤って管理者権限で起動される可能性があります。この動作が想定されていない場合は、設定ミスがあるか、または認証情報が侵害されている可能性があります。

修復のレコメンデーション

ポッドに許可を付与するためにデフォルトのサービスアカウントを使用しないでください。代わりに、ワークロードごとに専用のサービスアカウントを作成し、必要に応じてそのアカウントに許可を付与する必要があります。この問題を解決するには、すべてのポッドとワークロードの専用サービス

アカウントを作成し、ポッドとワークロードを更新して、デフォルトのサービスアカウントから専用アカウントに移行する必要があります。その後、デフォルトのサービスアカウントから管理者許可を削除する必要があります。詳細については、「EKS Protection の検出結果の修復」を参照してください。

Policy:Kubernetes/AnonymousAccessGranted

system:anonymous ユーザーには、Kubernetes クラスターに対する API 許可が付与されました。

デフォルトの重大度: [High] (高

- 機能: EKS 監査ログ

この検出結果は、Kubernetes クラスター上のユーザーが、ユーザー `system:anonymous` をロールにバインドするために `ClusterRoleBinding` または `RoleBinding` を正常に作成したことを知らせるものです。これにより、ロールで許可された API オペレーションへの認証されていないアクセスを有効にします。この動作が想定されていない場合は、設定ミスがあるか、または認証情報が侵害されている可能性があります

修復のレコメンデーション

クラスター上の `system:anonymous` ユーザーまたは `system:unauthenticated` グループに付与されている許可を調べて、不要な匿名アクセスを取り消す必要があります。詳細については、「Amazon EKS ユーザーガイド」の「Amazon EKS のセキュリティベストプラクティス」を参照してください。悪意を持って許可が付与された場合は、その許可を付与したユーザーのアクセスを取り消し、クラスターに対して攻撃者が加えた変更を元に戻す必要があります。詳細については、「EKS Protection の検出結果の修復」を参照してください。

Policy:Kubernetes/ExposedDashboard

Kubernetes クラスターのダッシュボードがインターネットに公開されました

デフォルトの重大度: [Medium] (中

- 機能: EKS 監査ログ

この検出結果は、クラスターの Kubernetes ダッシュボードがロードバランサーサービスによってインターネットに公開されたことを知らせるものです。ダッシュボードが公開されると、クラスターの

管理インターフェイスがインターネットからアクセス可能になり、攻撃者は認証およびアクセスコントロールの存在する可能性のあるギャップを悪用できます。

修復のレコメンデーション

Kubernetes ダッシュボードで強力な認証および認可が強制されているようにする必要があります。また、ネットワークアクセスコントロールを実装して、特定の IP アドレスからダッシュボードへのアクセスを制限する必要があります。

詳細については、「EKS Protection の検出結果の修復」を参照してください。

Policy:Kubernetes/KubeflowDashboardExposed

Kubernetes クラスターの Kubeflow ダッシュボードがインターネットに公開されました

デフォルトの重大度: [Medium] (中

- 機能: EKS 監査ログ

この検出結果は、クラスターの Kubeflow ダッシュボードがロードバランサーサービスによってインターネットに公開されたことを知らせるものです。Kubeflow ダッシュボードが公開されると、Kubeflow 環境の管理インターフェイスがインターネットからアクセス可能になり、攻撃者は認証およびアクセスコントロールの存在する可能性のあるギャップを悪用できます。

修復のレコメンデーション

Kubeflow ダッシュボードで強力な認証および認可が強制されているようにする必要があります。また、ネットワークアクセスコントロールを実装して、特定の IP アドレスからダッシュボードへのアクセスを制限する必要があります。

詳細については、「EKS Protection の検出結果の修復」を参照してください。

CredentialAccess:Kubernetes/AnomalousBehavior.SecretsAccessed

一般的にシークレットへのアクセスに使用される Kubernetes API が、異常な方法で呼び出されました。

デフォルトの重大度: [Medium] (中

- 機能: EKS 監査ログ

この検出結果は、機密のクラスターシークレットを取得する異常な API オペレーションが、クラスター内の Kubernetes ユーザーによって呼び出されたことを知らせるものです。観察された API は一般的に、認証情報アクセス戦略と関連付けられ、特権エスカレーションやクラスター内でさらにアクセスされる可能性があります。この動作が想定されていない場合は、設定ミスがあるか、または AWS 認証情報が侵害されている可能性があります。

この観察された API は、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。ML モデルは、EKS クラスター内のユーザー API アクティビティをすべて評価し、権限のないユーザーによって使用されたテクニックに関連する異常なイベントを特定します。ML モデルは、リクエストを行ったユーザー、リクエストが行われた場所、使用ユーザーエージェント、ユーザーが操作した名前空間など、API 操作の複数の要因を追跡します。例外的な API リクエストの詳細は、GuardDuty コンソールの検出結果の詳細パネルで確認できます。

修復のレコメンデーション

クラスターの Kubernetes ユーザーに付与された許可を調べて、すべての許可が必要であることを確認してください。誤って、または悪意を持って許可が付与された場合は、ユーザーのアクセスを取り消し、クラスターに対し認証されていないユーザーが加えたすべての変更を元に戻してください。詳細については、「EKS Protection の検出結果の修復」を参照してください。

AWS 認証情報が侵害された場合は、「」を参照してください侵害された可能性のある AWS 認証情報の修正。

PrivilegeEscalation:Kubernetes/AnomalousBehavior.RoleBindingCreated

Kubernetes クラスターで、過度に許容されるロールまたは機密性の高い名前空間への RoleBinding または ClusterRoleBinding が作成または変更されました。

デフォルトの重大度: [Medium] (中)*

> **Note**
>
> この検出結果のデフォルトの重大度は [Medium] (中) です。ただし、RoleBinding または ClusterRoleBinding に、ClusterRoles `admin` または `cluster-admin` が関係している場合、重大度は [High] (高) になります。

- 機能: EKS 監査ログ

この検出結果は、Kubernetes クラスター内のユーザーが、RoleBinding または ClusterRoleBinding を作成して、管理者権限または機密性の高い名前空間を持つロールにユーザーをバインドしたことを知らせるものです。この動作が想定されていない場合は、設定ミスがあるか、または AWS 認証情報が侵害されている可能性があります。

この観察された API は、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。ML モデルは EKS クラスター内のすべてのユーザー API アクティビティを評価します。この ML モデルは、権限のないユーザーが使用したテクニックに関連する異常なイベントも特定します。また、ML モデルは、リクエストを行ったユーザー、リクエストが行われた場所、使用ユーザーエージェント、ユーザーが操作した名前空間など、API 操作の複数の要因を追跡します。例外的な API リクエストの詳細は、GuardDuty コンソールの検出結果の詳細パネルで確認できます。

修復のレコメンデーション

Kubernetes ユーザーに付与された権限を確認してください。これらの権限は、RoleBinding および ClusterRoleBinding に関係するロールとサブジェクトで定義されます。誤って、または悪意を持って許可が付与された場合は、ユーザーのアクセスを取り消し、クラスターに対し認証されていないユーザーが加えたすべての変更を元に戻してください。詳細については、「EKS Protection の検出結果の修復」を参照してください。

AWS 認証情報が侵害された場合は、「」を参照してください侵害された可能性のある AWS 認証情報の修正。

Execution:Kubernetes/AnomalousBehavior.ExecInPod

ポッド内で異常な方法によるコマンドが実行されました。

デフォルトの重大度: [Medium] (中

- 機能: EKS 監査ログ

この検出結果は、Kubernetes exec API を使用してポッド内でコマンドが実行されたことを知らせるものです。Kubernetes exec API を使用すると、ポッド内で任意のコマンドを実行できます。この動作がユーザー、名前空間、またはポッドで想定されていない場合は、設定ミスまたは AWS 認証情報が侵害されている可能性があります。

この観察された API は、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。ML モデルは EKS クラスター内のすべてのユーザー API アクティビティを評価します。この ML モデルは、権限のないユーザーが使用したテクニックに関連する異常なイベントも特定しま

す。また、ML モデルは、リクエストを行ったユーザー、リクエストが行われた場所、使用ユーザーエージェント、ユーザーが操作した名前空間など、API 操作の複数の要因を追跡します。例外的な API リクエストの詳細は、GuardDuty コンソールの検出結果の詳細パネルで確認できます。

修復のレコメンデーション

このコマンドの実行が想定されていない場合は、コマンドの実行に使用されたユーザーアイデンティティの認証情報が侵害されている可能性があります。ユーザーのアクセスを取り消し、認証されていないユーザーがクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

AWS 認証情報が侵害された場合は、「」を参照してください侵害された可能性のある AWS 認証情報の修正。

PrivilegeEscalation:Kubernetes/AnomalousBehavior.WorkloadDeployed!PrivilegedContainer

ワークロードが特権コンテナを使用して異常な方法で起動されました。

デフォルトの重大度: [High] (高

- 機能: EKS 監査ログ

この検出結果は、ワークロードが Amazon EKS クラスター内の特権コンテナを使用して起動されたこと知らせるものです。特権コンテナには、ホストへのルートレベルのアクセス権があります。認証されていないユーザーは、特権のエスカレーション戦術として特権コンテナを起動して、ホストに最初にアクセスして侵害することができます。

この観察されたコンテナの作成や変更は、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。ML モデルは EKS クラスター内のすべてのユーザー API およびコンテナイメージアクティビティを評価します。この ML モデルは、権限のないユーザーが使用したテクニックに関連する異常なイベントも特定します。また、ML モデルは、リクエストを行ったユーザー、リクエストが行われた場所、使用ユーザーエージェント、アカウントで観察されたコンテナイメージ、ユーザーが操作した名前空間など、API 操作の複数の要因を追跡します。例外的な API リクエストの詳細は、GuardDuty コンソールの検出結果の詳細パネルで確認できます。

修復のレコメンデーション

このコンテナの起動が想定されていない場合、コンテナの起動に使用されたユーザーアイデンティティの認証情報が侵害されている可能性があります。ユーザーのアクセスを取り消し、認証されてい

ないユーザーがクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

AWS 認証情報が侵害された場合は、「」を参照してください侵害された可能性のある AWS 認証情報の修正。

このコンテナの起動が想定されている場合は、`resource.KubernetesDetails.KubernetesWorkloadDetails.containers.imagePrefix` フィールドに基づくフィルター基準を使用した抑制ルールを使用することをお勧めします。フィルター基準では、`imagePrefix` フィールドの値が検出結果で指定した `imagePrefix` フィールドの値と同じである必要があります。詳細については、「GuardDuty の抑制ルール」を参照してください。

Persistence:Kubernetes/AnomalousBehavior.WorkloadDeployed!ContainerWithSensitiveMount

ワークロードが異常な方法でデプロイされ、機密性の高いホストパスがワークロード内でマウントされました。

デフォルトの重大度: [High] (高

- 機能: EKS 監査ログ

この検出結果により、`volumeMounts` セクションに機密ホストパスを含むコンテナを使用してワークロードが起動されたことが分かります。これにより、機密性の高いホストパスがコンテナ内からアクセスおよび書き込み可能となる場合があります。この手法は、認証されていないユーザーがホストのファイルシステムにアクセスするために一般的に使用されます。

この観察されたコンテナの作成や変更は、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。ML モデルは EKS クラスター内のすべてのユーザー API およびコンテナイメージアクティビティを評価します。この ML モデルは、権限のないユーザーが使用したテクニックに関連する異常なイベントも特定します。また、ML モデルは、リクエストを行ったユーザー、リクエストが行われた場所、使用ユーザーエージェント、アカウントで観察されたコンテナイメージ、ユーザーが操作した名前空間など、API 操作の複数の要因を追跡します。例外的な API リクエストの詳細は、GuardDuty コンソールの検出結果の詳細パネルで確認できます。

修復のレコメンデーション

このコンテナの起動が想定されていない場合、コンテナの起動に使用されたユーザーアイデンティティの認証情報が侵害されている可能性があります。ユーザーのアクセスを取り消し、認証されていないユーザーがクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

AWS 認証情報が侵害された場合は、「」を参照してください侵害された可能性のある AWS 認証情報の修正。

このコンテナの起動が想定されている場合は、`resource.KubernetesDetails.KubernetesWorkloadDetails.containers.imagePrefix` フィールドに基づくフィルター基準を使用した抑制ルールを使用することをお勧めします。フィルター基準では、`imagePrefix` フィールドの値が検出結果で指定した `imagePrefix` フィールドの値と同じである必要があります。詳細については、「GuardDuty の抑制ルール」を参照してください。

Execution:Kubernetes/AnomalousBehavior.WorkloadDeployed

ワークロードが異常な方法で起動されました。

デフォルトの重大度: [Low] (低)*

> **Note**
>
> デフォルトの重大度は [Low] (低) です。ただし、ワークロードに疑わしいイメージ名 (既知のペネトレーションテストなど)や、起動時に疑わしいコマンドを実行しているコンテナ (リバースシェルコマンドなど) が含まれている場合、この検出結果タイプの重大度は [Midium] (中) と見なされます。

- 機能: EKS 監査ログ

この検出結果は、Kubernetes ワークロードが Amazon EKS クラスター内で、API アクティビティ、新しいコンテナイメージ、危険なワークロード設定などの異常な方法で作成または変更されたこと知らせるものです。承認されていないユーザーは、任意のコードを実行する手法としてコンテナを起動し、最初にホストにアクセスして、侵害する可能性があります。

この観察されたコンテナの作成や変更は、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。ML モデルは EKS クラスター内のすべてのユーザー API およびコンテナイメージアクティビティを評価します。この ML モデルは、権限のないユーザーが使用したテクニッ

クに関連する異常なイベントも特定します。また、ML モデルは、リクエストを行ったユーザー、リクエストが行われた場所、使用ユーザーエージェント、アカウントで観察されたコンテナイメージ、ユーザーが操作した名前空間など、API 操作の複数の要因を追跡します。例外的な API リクエストの詳細は、GuardDuty コンソールの検出結果の詳細パネルで確認できます。

修復のレコメンデーション

このコンテナの起動が想定されていない場合、コンテナの起動に使用されたユーザーアイデンティティの認証情報が侵害されている可能性があります。ユーザーのアクセスを取り消し、認証されていないユーザーがクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

AWS 認証情報が侵害された場合は、「」を参照してください侵害された可能性のある AWS 認証情報の修正。

このコンテナの起動が想定されている場合は、`resource.KubernetesDetails.KubernetesWorkloadDetails.containers.imagePrefix` フィールドに基づくフィルター基準を使用した抑制ルールを使用することをお勧めします。フィルター基準では、`imagePrefix` フィールドの値が検出結果で指定した `imagePrefix` フィールドの値と同じである必要があります。詳細については、「GuardDuty の抑制ルール」を参照してください。

PrivilegeEscalation:Kubernetes/AnomalousBehavior.RoleCreated

許容度の高いロールまたは ClusterRole が異常な方法で作成または変更されました。

デフォルトの重大度: [Low] (低

- 機能: EKS 監査ログ

この検出結果は、過剰な権限を使用した `Role` または `ClusterRole` を作成する異常な API オペレーションが、Amazon EKS クラスター内の Kubernetes ユーザーによって呼び出されたこと知らせるものです。アクターは、強力な権限を持つロール作成を利用することで、組み込み型の管理者に似せたロールの使用を避け、検出を回避する可能性があります。過剰な権限により、権限エスカレーションやリモートコードが実行され、名前空間やクラスターが制御される可能性があります。この動作が想定されていない場合は、設定ミスがあるか、または認証情報が侵害されている可能性があります。

この観察された API は、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。ML モデルは、Amazon EKS クラスター内のユーザー API アクティビティをすべて評

価し、権限のないユーザーによって使用された手法に関連する異常なイベントを特定します。また、ML モデルは、リクエストを行ったユーザー、リクエストが行われた場所、使用ユーザーエージェント、アカウントで観察されたコンテナイメージ、ユーザーが操作した名前空間など、API 操作の複数の要因を追跡します。例外的な API リクエストの詳細は、GuardDuty コンソールの検出結果の詳細パネルで確認できます。

修復のレコメンデーション

`Role` または `ClusterRole` で定義されている権限を調べ、すべての権限が必要であることを確認し、最小特権の原則に従ってください。誤って、または悪意を持って許可が付与された場合は、ユーザーのアクセスを取り消し、クラスターに対し認証されていないユーザーが加えたすべての変更を元に戻してください。詳細については、「EKS Protection の検出結果の修復」を参照してください。

AWS 認証情報が侵害された場合は、「」を参照してください侵害された可能性のある AWS 認証情報の修正。

Discovery:Kubernetes/AnomalousBehavior.PermissionChecked

ユーザーが異常な方法でアクセス許可を確認しました。

デフォルトの重大度: [Low] (低

- 機能: EKS 監査ログ

この検出結果は、Kubernetes クラスター内のユーザーが、特権エスカレーションやリモートコード実行につながる可能性がある既知の強力な権限が許可されているかどうかの確認に成功したこと知らせるものです。例えば、ユーザーの権限を確認するのに使用される一般的なコマンドは、`kubectl auth can-i` です。この動作が想定されていない場合は、設定ミスがあるか、または認証情報が侵害されている可能性があります。

この観察された API は、GuardDuty の異常検出機械学習 (ML) モデルによって異常として識別されています。ML モデルは、Amazon EKS クラスター内のユーザー API アクティビティをすべて評価し、権限のないユーザーによって使用された手法に関連する異常なイベントを特定します。また、ML モデルは、リクエストを行ったユーザー、リクエストが行われた場所、確認された許可、ユーザーが操作した名前空間など、API 操作の複数の要因を追跡します。例外的な API リクエストの詳細は、GuardDuty コンソールの検出結果の詳細パネルで確認できます。

修復のレコメンデーション

Kubernetes ユーザーに付与された権限を調べて、すべての権限が必要であることを確認してください。誤って、または悪意を持って許可が付与された場合は、ユーザーのアクセスを取り消し、クラスターに対し認証されていないユーザーが加えたすべての変更を元に戻してください。詳細については、「EKS Protection の検出結果の修復」を参照してください。

AWS 認証情報が侵害された場合は、「」を参照してください侵害された可能性のある AWS 認証情報の修正。

GuardDuty Runtime Monitoring の検出結果タイプ

Amazon GuardDuty は、Amazon EKS クラスター、Fargate および Amazon ECS ワークロード、および Amazon EC2 インスタンス内の Amazon EC2 ホストとコンテナからのオペレーティングシステムレベルの動作に基づいて潜在的な脅威を示すために、以下の Runtime Monitoring の検出結果を生成します。

GuardDuty は、Runtime Monitoring のカスタム検出ルールを有効にした場合にのみ、一部の Runtime Monitoring 結果タイプを生成します。これらの検出結果タイプについては、「」を参照してくださいカスタム検出ルールを必要とする検出タイプ。

> **Note**
>
> Runtime Monitoring の検出結果タイプは、ホストから収集されたランタイムログに基づいています。ログには、悪意のある攻撃者によってコントロールされる可能性のあるファイルパスなどのフィールドが含まれています。これらのフィールドは、ランタイムコンテキストを提供するために GuardDuty の検出結果にも含まれています。Runtime Monitoring の検出結果を GuardDuty コンソールの外部で処理する場合、検出結果フィールドをサニタイズする必要があります。例えば、検出結果フィールドをウェブページに表示するときに、検索フィールドを HTML でエンコードできます。

トピック

- Runtime Monitoring がデフォルトで生成する検出結果タイプ
- カスタム検出ルールを必要とする検出タイプ

Runtime Monitoring がデフォルトで生成する検出結果タイプ

Runtime Monitoring を有効にすると、GuardDuty はこのセクションの結果タイプを生成します。これらの検出結果タイプを取得するために、ランタイムモニタリングのカスタム検出ルールを有効にする必要はありません。

トピック

- CryptoCurrency:Runtime/BitcoinTool.B
- Backdoor:Runtime/C&CActivity.B
- UnauthorizedAccess:Runtime/TorRelay
- UnauthorizedAccess:Runtime/TorClient
- Trojan:Runtime/BlackholeTraffic
- Trojan:Runtime/DropPoint
- CryptoCurrency:Runtime/BitcoinTool.B!DNS
- Backdoor:Runtime/C&CActivity.B!DNS
- Trojan:Runtime/BlackholeTraffic!DNS
- Trojan:Runtime/DropPoint!DNS
- Trojan:Runtime/DGADomainRequest.C!DNS
- Trojan:Runtime/DriveBySourceTraffic!DNS
- Trojan:Runtime/PhishingDomainRequest!DNS
- Impact:Runtime/AbusedDomainRequest.Reputation
- Impact:Runtime/BitcoinDomainRequest.Reputation
- Impact:Runtime/MaliciousDomainRequest.Reputation
- Impact:Runtime/SuspiciousDomainRequest.Reputation
- UnauthorizedAccess:Runtime/MetadataDNSRebind
- Execution:Runtime/NewBinaryExecuted
- PrivilegeEscalation:Runtime/DockerSocketAccessed
- PrivilegeEscalation:Runtime/RuncContainerEscape
- PrivilegeEscalation:Runtime/CGroupsReleaseAgentModified
- DefenseEvasion:Runtime/ProcessInjection.Proc

- DefenseEvasion:Runtime/ProcessInjection.Ptrace
- DefenseEvasion:Runtime/ProcessInjection.VirtualMemoryWrite
- Execution:Runtime/ReverseShell
- DefenseEvasion:Runtime/FilelessExecution
- Impact:Runtime/CryptoMinerExecuted
- Execution:Runtime/NewLibraryLoaded
- PrivilegeEscalation:Runtime/ContainerMountsHostDirectory
- PrivilegeEscalation:Runtime/UserfaultfdUsage
- Execution:Runtime/SuspiciousTool
- Execution:Runtime/SuspiciousCommand
- DefenseEvasion:Runtime/SuspiciousCommand
- DefenseEvasion:Runtime/PtraceAntiDebugging
- Execution:Runtime/MaliciousFileExecuted
- Execution:Runtime/MaliciousFileExecuted.Custom
- Execution:Runtime/SuspiciousShellCreated
- PrivilegeEscalation:Runtime/ElevationToRoot
- Discovery:Runtime/SuspiciousCommand
- Persistence:Runtime/SuspiciousCommand
- PrivilegeEscalation:Runtime/SuspiciousCommand
- DefenseEvasion:Runtime/KernelModuleLoaded
- Persistence:Runtime/SensitiveFileModified
- PrivilegeEscalation:Runtime/SensitiveFileModified
- DefenseEvasion:Runtime/SensitiveFileModified

CryptoCurrency:Runtime/BitcoinTool.B

Amazon EC2 インスタンスまたはコンテナが暗号通貨関連のアクティビティに関連付けられている IP アドレスをクエリしています。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境内のリストされた EC2 インスタンスまたはコンテナで実行されているプロセスが暗号通貨関連アクティビティに紐づけられた IP アドレスをクエリしていることを知らせるものです。脅威アクターは、コンピューティングリソースをコントロールして、不正な暗号通貨マイニングに対して悪意を持ち再利用しようとする可能性があります。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

暗号通貨の情報を取り出して管理するためこの EC2 インスタンスまたはコンテナを使用する場合、またはこれらのいずれかがブロックチェーンのアクティビティに関与している場合は、この CryptoCurrency:Runtime/BitcoinTool.B 検出結果はご利用の環境の想定されるアクティビティを示している可能性があります。 AWS 環境でこのような場合は、この検出結果の抑制ルールを設定することをお勧めします。抑制ルールは、2 つのフィルター条件で構成する必要があります。1 つ目の検索条件フィルターでは、[検出結果タイプ] 属性に `CryptoCurrency:Runtime/BitcoinTool.B` という値を使用します。2 つ目のフィルター条件は、インスタンスの [インスタンス ID]、または暗号通貨やブロックチェーン関連のアクティビティに関わるコンテナの [コンテナイメージ ID] です。詳細については、「抑制ルール」を参照してください。

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Backdoor:Runtime/C&CActivity.B

Amazon EC2 インスタンスまたはコンテナは、既知のコマンドとコントロールサーバーに関連付けられる IP をクエリしています。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境内のリストされた EC2 インスタンスまたはコンテナで実行されているプロセスが、既知のコマンドアンドコントロール (C&C) サーバーと関連する IP アドレスをクエリし

ていることを示しています。リストされているインスタンスまたはコンテナが侵害されている可能性があります。C&C サーバーは、ボットネットのメンバーにコマンドを発行するコンピュータです。

ボットネットとは、一般的なタイプのマルウェアに感染しコントロールされたインターネットコネクテッドデバイス (PC、サーバー、モバイルデバイス、IoT デバイスなど) のコレクションです。ボットネットは、マルウェアを配布し、クレジットカード番号などの悪用された情報を収集するためによく使用されます。ボットネットの目的と構造によっては、C&C サーバーが分散サービス拒否 (DDoS) 攻撃を開始するコマンドを発行することがあります。

> **Note**
>
> クエリされた IP が log4j 関連の場合、関連付けられた検出結果のフィールドには次の値が含まれます。
>
> - `service.additionalInfo.threatListName = Amazon`
> - `service.additionalInfo.threatName = Log4j Related`

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

UnauthorizedAccess:Runtime/TorRelay

Amazon EC2 インスタンスまたはコンテナが Tor リレーとして Tor ネットワークに接続しています。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

この検出結果は、リストされた EC2 インスタンスまたは AWS 環境内のコンテナで実行されているプロセスが、Tor リレーとして動作していることを示す方法で Tor ネットワークに接続していること

を知らせるものです。Tor は、匿名通信を有効にするためのソフトウェアです。ある Tor リレーから別の Tor リレーにクライアントの不正なトラフィックを転送することで、通信の匿名性を高めます。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

UnauthorizedAccess:Runtime/TorClient

Amazon EC2 インスタンスまたはコンテナが Tor Guard または Authority ノードに接続しています。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

この検出結果は、リストされた EC2 インスタンスまたは AWS 環境内のコンテナで実行されているプロセスが Tor Guard または Authority ノードに接続していることを知らせるものです。Tor は、匿名通信を有効にするためのソフトウェアです。Tor Guard と Authority ノードは、Tor ネットワークへの初期ゲートウェイとして機能します。このトラフィックは、この EC2 インスタンスまたはコンテナが侵害された可能性があり、Tor ネットワーク上のクライアントとして動作していることを示している場合があります。この検出結果は、攻撃者の真のアイデンティティを隠す目的で AWS リソースへの不正アクセスを示している可能性があります。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Trojan:Runtime/BlackholeTraffic

Amazon EC2 インスタンスまたはコンテナが既知のブラックホールであるリモートホストの IP アドレスに通信しようとしています。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

この検出結果は、リストされた EC2 インスタンスまたは AWS 環境内のコンテナで実行されているプロセスが、ブラックホール (またはシンクホール) の IP アドレスと通信しようとしているために侵害されている可能性があることを知らせるものです。ブラックホールとは、送受信トラフィックが密かに破棄されるネットワークの場所を指し、意図した受信者にデータが届いていないことは送信元に知らされません。ブラックホール IP アドレスは、実行されていないホストマシン、またはホストが割り当てられていないアドレスを指定します。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Trojan:Runtime/DropPoint

Amazon EC2 インスタンスまたはコンテナが、マルウェアによって収集された認証情報やその他の盗難されたデータを保持していることが認識されているリモートホストの IP アドレスに通信しようとしています。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

この検出結果は、リストされた EC2 インスタンスまたは AWS 環境内のコンテナで実行されているプロセスが、マルウェアによってキャプチャされた認証情報やその他の盗まれたデータを保持することが知られているリモートホストの IP アドレスと通信しようとしていることを知らせるものです。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

CryptoCurrency:Runtime/BitcoinTool.B!DNS

Amazon EC2 インスタンスまたはコンテナが暗号通貨のアクティビティに関連付けられているドメイン名をクエリしています。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境内のリストされた EC2 インスタンスまたはコンテナで実行されているプロセスが、ビットコイン、またはその他の暗号通貨関連アクティビティに紐づけたドメインをクエリしていることを知らせるものです。脅威アクターは、コンピューティングリソースを不正な暗号通貨マイニングに不正に転用するため、コンピュートリソースを乗っ取ろうとする可能性があります。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

暗号通貨の情報を取り出して管理するためこの EC2 インスタンスまたはコンテナがを使用する場合、またはこれらのいずれかがブロックチェーンのアクティビティに関与している場合は、この CryptoCurrency:Runtime/BitcoinTool.B!DNS 検出結果はご利用の環境の想定されるアクティビティを示している可能性があります。 AWS 環境でこのような場合は、この検出結果の抑制ルールを設定することをお勧めします。抑制ルールは、2 つのフィルター条件で構成する必要があります。1 つ目の条件では、[結果タイプ] 属性に `CryptoCurrency:Runtime/BitcoinTool.B!DNS` という値を使用します。2 つ目のフィルター条件では、暗号通貨またはブロックチェーンアクティビティに関与するインスタンスの [インスタンス ID]、またはコンテナの [コンテナイメージ ID] である必要があります。詳細については、「抑制ルール」を参照してください。

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Backdoor:Runtime/C&CActivity.B!DNS

Amazon EC2 インスタンスまたはコンテナが、既知のコマンドとコントロールサーバーに関連付けられるドメイン名をクエリしています。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境内のリストされた EC2 インスタンスおよびコンテナで実行されているプロセスが、既知のコマンドとコントロール (C&C) サーバーに関連付けられているドメインをクエリしていることを知らせるものです。リスト化した EC2 インスタンスまたはコンテナは侵害されている可能性があります。C&C サーバーは、ボットネットのメンバーにコマンドを発行するコンピュータです。

ボットネットとは、一般的なタイプのマルウェアに感染し制御されたインターネットコネクテッドデバイス (PC、サーバー、モバイルデバイス、IoT デバイスなど) のコレクションです。通常、ボットネットは、マルウェアの配布や盗用された情報 (クレジットカード番号など) の収集に使用されます。ボットネットの目的と構造によっては、C&C サーバーから分散型サービス拒否 (DDoS) 攻撃を開始するためのコマンドが発行されることもあります。

> **Note**
>
> クエリされたドメイン名が log4j 関連の場合、関連付けられた検出結果のフィールドには次の値が含まれます。
>
> - `service.additionalInfo.threatListName = Amazon`
> - `service.additionalInfo.threatName = Log4j Related`

> **Note**
>
> GuardDuty がこの検出結果タイプを生成する方法をテストするには、テスト用のドメイン `guarddutyc2activityb.com` に (Linux については `dig`、Windows については `nslookup` を使用して) インスタンスから DNS リクエストを実行できます。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Trojan:Runtime/BlackholeTraffic!DNS

Amazon EC2 インスタンスまたはコンテナがブラックホールの IP アドレスにリダイレクトされるドメイン名をクエリしています。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境内のリストされた EC2 インスタンスまたはコンテナで実行されているプロセスが、ブラックホール IP アドレスにリダイレクトされるドメイン名をクエリしているため、侵害されている可能性があることを知らせるものです。ブラックホールとは、送受信トラフィックが密かに破棄されるネットワークの場所を指し、意図した受信者にデータが届いていないことは送信元に知らされません。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Trojan:Runtime/DropPoint!DNS

Amazon EC2 インスタンスまたはコンテナが、マルウェアによって収集された認証情報やその他の盗難されたデータを保持していることが認識されているリモートホストのドメイン名をクエリしています。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

この検出結果は、リストされた EC2 インスタンスまたは AWS 環境内のコンテナで実行されているプロセスが、マルウェアによってキャプチャされた認証情報やその他の盗まれたデータを保持することが知られているリモートホストのドメイン名をクエリしていることを知らせるものです。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Trojan:Runtime/DGADomainRequest.C!DNS

Amazon EC2 インスタンスまたはコンテナがアルゴリズムを使用して生成されたドメインをクエリしています。このようなドメインは、マルウェアによって悪用されることが多く、EC2 インスタンスまたはコンテナが侵害されている場合があります。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境内のリストされた EC2 インスタンスまたはコンテナで実行されているプロセスが、ドメイン生成アルゴリズム (DGA) のドメインをクエリしようとしていることを知らせるものです。リソースが侵害されている可能性があります。

DGAsは、コマンドアンドコントロール (C&C) サーバーでレンデブーポイントとして使用できる多数のドメイン名を定期的に生成するために使用されます。C&C サーバーは、ボットネットのメンバーにコマンドを発行するコンピュータであり、一般的なタイプのマルウェアに感染して制御されたインターネットのコネクテッドデバイスのコレクションです。潜在的なランデブーポイントが多いと、感染したコンピュータが毎日これらのドメイン名の一部に連絡して更新やコマンドを受け取ろうとするため、ボットネットを効果的にシャットダウンすることが困難になります。

> **Note**
>
> この検出結果は、GuardDuty 脅威インテリジェンスフィードの既知の DGA ドメインに基づいています。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Trojan:Runtime/DriveBySourceTraffic!DNS

Amazon EC2 インスタンスまたはコンテナがドライブバイダウンロード攻撃の既知の攻撃元であるリモートホストのドメイン名をクエリしています。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

この検出結果は、自動ダウンロード攻撃の既知のソースであるリモートホストのドメイン名をクエリしているため、 AWS 環境内のリストされた EC2 インスタンスまたはコンテナで実行されているプロセスが侵害された可能性があることを知らせるものです。これらは、インターネットから意図せずにダウンロードされるコンピュータソフトウェアであり、ウイルス、スパイウェア、マルウェアの自動インストールを開始する場合があります。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Trojan:Runtime/PhishingDomainRequest!DNS

Amazon EC2 インスタンスまたはコンテナがフィッシング攻撃に関与しているドメインをクエリしています。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境内のリストされた EC2 インスタンスまたはコンテナで実行されているプロセスがフィッシング攻撃に関与しているドメインをクエリしようとしていることを知らせるものです。フィッシングドメインは、個人を特定できる情報、銀行やクレジットカードの詳細情報とパスワードなど、ユーザーが機密データを提供するように仕向ける、正当な機関になりすました人物によって設定されます。EC2 インスタンスまたはコンテナがフィッシングウェブサイトに保存されている機密データを検索しようとしたり、フィッシングウェブサイトをセットアップしようとしたりする可能性があります。EC2 インスタンスまたはコンテナが侵害されている可能性があります。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Impact:Runtime/AbusedDomainRequest.Reputation

Amazon EC2 インスタンスまたはコンテナが、既知の悪用されたドメインに関連付けられた評価の低いドメイン名をクエリしています。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境内のリストされた EC2 インスタンスまたはコンテナで実行されているプロセスが、既知の悪用されたドメインまたは IP アドレスに関連付けられたレピュテーションの低いドメイン名をクエリしていることを知らせるものです。悪用したドメインの例としては、動的 DNS プロバイダーだけでなく、無料のサブドメイン登録を提供する最上位のドメイン名 (TLD) と第

2 位のドメイン名 (2LD) があります。脅威アクターは、無料または低コストでドメインを登録するこれらのサービスを使用する傾向があります。このカテゴリの評価の低いドメインは、レジストラのパーキング IP アドレスを決定する有効期限切れドメインであり、アクティブになっていない可能性があります。パーキング IP は、レジストラがどのサービスにもリンクされていないドメインのトラフィックを管理する場所です。脅威アクターが一般的にこれらのレジストラのサービスまたは C&C のサービス、マルウェアディストリビューションに使用するため、リストされた Amazon EC2 インスタンスまたはコンテナは侵害される可能性があります。

[Low] (低) のレピュテーションドメインは、レピュテーションスコアモデルに基づいています。このモデルは、ドメインの特徴を評価およびランク付けし、それが悪意のあるものである可能性を判断します。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Impact:Runtime/BitcoinDomainRequest.Reputation

Amazon EC2 インスタンスまたはコンテナが、暗号通貨関連のアクティビティに関連付けられている評判の低いドメイン名をクエリしています。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境のリストされた EC2 インスタンスまたはコンテナで実行されているプロセスが、ビットコイン、またはその他の暗号通貨関連アクティビティに紐づけた評判の低いドメイン名をクエリしていることを知らせるものです。脅威アクターは、コンピューティングリソースをコントロールして、不正な暗号通貨マイニングに対して悪意を持ち再利用しようとする可能性があります。

[Low] (低) のレピュテーションドメインは、レピュテーションスコアモデルに基づいています。このモデルは、ドメインの特徴を評価およびランク付けし、それが悪意のあるものである可能性を判断します。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

暗号通貨の情報を取り出して管理するためこの EC2 インスタンスまたはコンテナを使用する場合、またはこれらのリソースがブロックチェーンのアクティビティに関与している場合は、この検出結果はご利用の環境の想定されるアクティビティを示している可能性があります。 AWS 環境でこのような場合は、この検出結果の抑制ルールを設定することをお勧めします。抑制ルールは、2 つのフィルター条件で構成する必要があります。1 つ目の検索条件フィルターでは、[検出結果タイプ] 属性に `Impact:Runtime/BitcoinDomainRequest.Reputation` という値を使用します。2 番目のフィルター条件は、インスタンスの [インスタンス ID] か、コンテナの [コンテナイメージ ID] が暗号通貨やブロックチェーン関連のアクティビティに関係しているかどうかです。詳細については、「抑制ルール」を参照してください。

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Impact:Runtime/MaliciousDomainRequest.Reputation

Amazon EC2 インスタンスまたはコンテナが、悪意のある既知のドメインに関連付けられた評判の低いドメインをクエリしています。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境内のリストされた EC2 インスタンスまたはコンテナで実行されているプロセスが、悪意のある既知のドメインまたは IP アドレスに関連付けられたレピュテーションの低いドメイン名をクエリしていることを知らせるものです。例えば、ドメインを既知のシンクホール IP アドレスに関連付けることができます。シンクホールドメインは、以前に脅威アクターに制御されたドメインであり、ドメインへのリクエストは、インスタンスが侵害されていることを示している場合があります。これらのドメインは、悪意のある既知のキャンペーンやドメイン生成アルゴリズムと相関している可能性もあります。

[Low] (低) のレピュテーションドメインは、レピュテーションスコアモデルに基づいています。このモデルは、ドメインの特徴を評価およびランク付けし、それが悪意のあるものである可能性を判断します。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Impact:Runtime/SuspiciousDomainRequest.Reputation

Amazon EC2 インスタンスまたはコンテナが、年齢や低人気により、本質的に疑わしい、低評判のドメイン名をクエリしています。

デフォルトの重大度: [Low] (低

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境内のリストされた EC2 インスタンスまたはコンテナで実行されているプロセスが、悪意があると疑われる低評価ドメイン名をクエリしていることをユーザーに知らせるものです。このドメインで観察された特性は、以前に観察された悪意のあるドメインと一致していました。ただし、評価モデルはそれを既知の脅威と明確に関連付けることができませんでした。これらのドメインは通常、新たに観察されるか、または少量のトラフィックを受信します。

[Low] (低) のレピュテーションドメインは、レピュテーションスコアモデルに基づいています。このモデルは、ドメインの特徴を評価およびランク付けし、それが悪意のあるものである可能性を判断します。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

UnauthorizedAccess:Runtime/MetadataDNSRebind

Amazon EC2 インスタンスまたはコンテナがインスタンスメタデータサービスに解決される DNS ルックアップを実行しています。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

> **Note**
>
> 現在、この検出結果タイプは AMD64 アーキテクチャでのみサポートされています。

この検出結果は、リストされた EC2 インスタンスまたは AWS 環境内のコンテナで実行されているプロセスが、EC2 メタデータ IP アドレス (169.254.169.254) に解決されるドメインをクエリしていることを知らせるものです。この種類の DNS クエリは、インスタンスが DNS リバイディング技術の対象であることを示している可能性があります。この手法は、インスタンスに関連付けらた IAM 認証情報など、EC2 インスタンスからメタデータを取得するために使用できます。

DNS リバインディングには、EC2 インスタンスで実行されているアプリケーションをだまして URL から返されるデータをロードすることが含まれます。URL のドメイン名は EC2 メタデータ IP アドレス (169.254.169.254) に解決されます。これにより、アプリケーションは EC2 メタデータにアクセスし、攻撃者がそのメタデータを使用できるようにする可能性があります。

EC2 インスタンスが URL の追加を許可する脆弱なアプリケーションを実行している場合や、EC2 インスタンスで実行されているウェブブラウザで、誰かが URL にアクセスする場合のみ、DNS リバインディングを使用して EC2 メタデータにアクセスできます。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

この検出結果に応じて、EC2 インスタンスまたはコンテナで実行されている脆弱性アプリケーションがあるか、誰が検出結果で識別したドメインへアクセスするためブラウザを使用しているかを評価する必要があります。根本的な原因が脆弱なアプリケーションである場合は、脆弱性を修復します。ユーザーが識別したドメインを閲覧した場合、ドメインをブロックするか、ユーザーがそのドメイン

にアクセスできないようにします。この検出結果が上記のいずれかのケースに関連していると判断した場合は、EC2 インスタンスに関連付けられたセッションを取り消す必要があります。

一部の AWS お客様は、メタデータ IP アドレスを信頼できる DNS サーバーのドメイン名に意図的にマッピングします。ご利用の環境でこのような状況が発生した場合は、この検出結果に対する抑制ルールを設定することをお勧めします。抑制ルールは、2 つのフィルター条件で構成する必要があります。1 つ目の検索条件フィルターでは、[検出結果タイプ] 属性に UnauthorizedAccess:Runtime/MetadataDNSRebind という値を使用します。2 番目のフィルター条件は、[DNS リクエストドメイン] またはコンテナの [コンテナイメージ ID] です。[DNS リクエストのドメイン] を使用します。値はメタデータの IP アドレス (169.254.169.254) にマッピングしたドメインと一致する必要があります。抑制ルール作成の詳細については、「抑制ルール」を参照してください。

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Execution:Runtime/NewBinaryExecuted

コンテナで新しく作成、または最近変更されたバイナリファイルが実行されました。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

この検出結果から、コンテナ内で新たに作成、または最近変更されたバイナリファイルが実行されたことがわかります。実行時にコンテナを不変に保つことがベストプラクティスであり、バイナリファイル、スクリプト、またはライブラリはコンテナの存続期間中に作成または変更しないでください。この動作は、悪意のある攻撃者が潜在的な侵害の一環としてコンテナにアクセスし、マルウェアやその他のソフトウェアをダウンロードして実行したことを示しています。このタイプのアクティビティは侵害の兆候を示している可能性がありますが、一般的な使用パターンでもあります。したがって、GuardDuty はメカニズムを使用してこのアクティビティの疑わしいインスタンスを識別し、疑わしいインスタンスに対してのみ、この検出結果タイプを生成します。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。変更プロセスと新しいバイナリを特定するには、[変更プロセス] の詳細と [プロセス] の詳細を表示します。

変更プロセスの詳細は、検出結果の JSON の service.runtimeDetails.context.modifyingProcess フィールド、または検出結果の詳

細パネルの [変更プロセス] に記載されています。この検出結果タイプでは、検出結果の JSON の `service.runtimeDetails.context.modifyingProcess.executablePath` フィールド、または検出結果の詳細パネルの [変更プロセス] の一部として識別されるように、変更プロセスは `/usr/bin/dpkg` です。

実行された新規または変更されたバイナリの詳細は、検出結果の JSON の `service.runtimeDetails.process`、または [ランタイムの詳細] の [プロセス] セクションに記載されています。この検出結果タイプでは、`service.runtimeDetails.process.executablePath` ([実行可能パス]) フィールドで示されるように、新規または変更されたバイナリは `/usr/bin/python3.8` です。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

PrivilegeEscalation:Runtime/DockerSocketAccessed

コンテナ内のプロセスは、Docker ソケットを使用して Docker デーモンと通信しています。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

Docker ソケットは、Docker デーモン (`dockerd`) がクライアントとの通信に使用する Unix ドメインソケットです。クライアントは、Docker ソケットを介して Docker デーモンと通信してコンテナを作成するなど、さまざまなアクションを実行できます。コンテナプロセスが Docker ソケットにアクセスするのは疑わしい状況です。コンテナプロセスは、Docker ソケットと通信して特権コンテナを作成することで、コンテナからエスケープしてホストレベルのアクセスを得ることができます。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

PrivilegeEscalation:Runtime/RuncContainerEscape

runC を介したコンテナエスケープの試行が検出されました。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

RunC は、Docker や Containerd などの高レベルコンテナランタイムがコンテナを生成して実行するために使用する低レベルのコンテナランタイムです。RunC はコンテナを作成する低レベルのタスクを実行する必要があるため、常にルート権限で実行されます。脅威アクターは、runC バイナリの脆弱性を変更または悪用することで、ホストレベルのアクセス権を取得することができます。

この検出結果は、runC バイナリの変更と、次の runC の脆弱性を悪用する試みの可能性を検出します。

- CVE-2019-5736 – CVE-2019-5736 の悪用には、コンテナ内から runC バイナリを上書きすることが含まれます。この検出結果は、runC バイナリがコンテナ内のプロセスによって変更されたときに呼び出されます。
- CVE-2024-21626 – CVE-2024-21626 の悪用には、現在の作業ディレクトリ (CWD) またはコンテナをオープンファイル記述子 `/proc/self/fd/`*`FileDescriptor`* に設定することが含まれます。この検出結果は、`/proc/self/fd/` 下に現在の作業ディレクトリでのコンテナプロセスが検出されたときに呼び出されます (例: `/proc/self/fd/7`)。

この検出結果は、悪意のあるアクターが次のいずれかのタイプのコンテナで悪用を試みたことを示している可能性があります。

- 攻撃者がコントロールするイメージを含んだ新しいコンテナ。
- ホストレベルの runC バイナリに対する書き込みアクセス許可を持つアクターがアクセスできた既存のコンテナ。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

PrivilegeEscalation:Runtime/CGroupsReleaseAgentModified

CGroups リリースエージェントを介したコンテナエスケープの試行が検出されました。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

この検出結果から、コントロールグループ (cgroup) リリースエージェントファイルを変更しようとした試みが検出されたことがわかります。Linux はコントロールグループ (cgroup) を使用して、プロセスの集合のリソース使用量を制限したり、説明したり、分離したりします。各 cgroup にはリリースエージェントファイル (`release_agent`) があります。これは cgroup 内のいずれかのプロセスが終了したときに Linux が実行するスクリプトです。リリースエージェントファイルは常にホストレベルで実行されます。コンテナ内の脅威アクターは、cgroup に属するリリースエージェントファイルに任意のコマンドを書き込むことで、ホストに逃げることができます。その cgroup 内のプロセスが終了すると、アクターによって書き込まれたコマンドが実行されます。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

DefenseEvasion:Runtime/ProcessInjection.Proc

proc ファイルシステムを使用したプロセスインジェクションが、コンテナまたは Amazon EC2 インスタンスで検出されました。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

プロセスインジェクションは、脅威アクターが防御を回避し潜在的にアクセス許可を昇格させるため、プロセスにコードを挿入するために使用する手法です。proc ファイルシステム (procfs) は、プ

ロセスの仮想メモリをファイルとして表示する Linux の特別なファイルシステムです。そのファイルのパスは `/proc/PID/mem` で、`PID` はプロセスの固有の ID です。脅威アクターは、このファイルに書き込んで、プロセスにコードを挿入できます。この検出結果により、このファイルへの書き込みを試みる可能性が明らかになりました。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティは予期しないものである場合、リソースタイプは予期しないものである可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

DefenseEvasion:Runtime/ProcessInjection.Ptrace

ptrace システムコールを使用したプロセスインジェクションが、コンテナまたは Amazon EC2 インスタンスで検出されました。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

プロセスインジェクションは、脅威アクターが防御を回避し潜在的にアクセス許可を昇格させるため、プロセスにコードを挿入するために使用する手法です。プロセスは ptrace システムコールを使って、別のプロセスにコードを挿入できる。この検出結果により、ptrace システムコールを使って、プロセスへのコードの挿入を試みる可能性が明らかになりました。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティは予期しないものである場合、リソースタイプは予期しないものである可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

DefenseEvasion:Runtime/ProcessInjection.VirtualMemoryWrite

仮想メモリへの直接書き込みによるプロセスインジェクションが、コンテナまたは Amazon EC2 インスタンスで検出されました。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

プロセスインジェクションは、脅威アクターが防御を回避し潜在的にアクセス許可を昇格させるため、プロセスにコードを挿入するために使用する手法です。プロセスは、`process_vm_writev` などのシステムコールを使用して、別のプロセスの仮想メモリにコードを直接挿入することができます。この検出結果により、プロセスの仮想メモリに書き込むためのシステムコールを使って、プロセスへのコードの挿入を試みる可能性が明らかになりました。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティは予期しないものである場合、リソースタイプは予期しないものである可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Execution:Runtime/ReverseShell

コンテナまたは Amazon EC2 インスタンス内のプロセスによってリバースシェルが作成されました。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

リバースシェルは、ターゲットホストからアクターのホストへの接続で作成されるシェルセッションです。これは、アクターのホストからターゲットのホストに対して開始される通常のシェルとは逆です。脅威アクターは、ターゲットへの最初のアクセス権を取得した後、リバースシェルを作成してターゲット上でコマンドを実行します。この検出結果は、潜在的に疑わしいリバースシェル接続を特定します。

GuardDuty は、関連するランタイムアクティビティとコンテキストを調べ、関連するアクティビティとコンテキストが異常または不審と判断された場合にのみ、この検出結果タイプを生成します。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

GuardDuty セキュリティエージェントは、複数のソースからのイベントをモニタリングします。影響を受けるリソースを特定するには、GuardDuty コンソールの検出結果の詳細で [リソースタイプ] を確認してください。このアクティビティは予期しないものである場合、リソースタイプは予期しないものである可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

DefenseEvasion:Runtime/FilelessExecution

コンテナまたは Amazon EC2 インスタンス内のプロセスが、メモリからコードを実行しています。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

この検出結果により、ディスク上のメモリ内の実行可能ファイルを使用してプロセスが実行されたときに通知されます。これは、ファイルシステムのスキャンによる検出を回避するために、悪意のある実行可能ファイルをディスクに書き込まないようにする、一般的な防御回避の手法です。この手法はマルウェアによって使用されていますが、正当な使用例もいくつかあります。その例の 1 つに、コンパイルされたコードをメモリに書き込み、メモリから実行するジャストインタイム (JIT) コンパイラが関連付けられていることを知らせるものです。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Impact:Runtime/CryptoMinerExecuted

コンテナまたは Amazon EC2 インスタンスが、暗号通貨マイニングアクティビティに関連するバイナリファイルを実行しています。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境内のリストされた EC2 インスタンスまたはコンテナで実行されているプロセスが、暗号通貨マイニングアクティビティに関連付けられたバイナリファイルを実行していることを知らせるものです。脅威アクターは、コンピューティングリソースをコントロールして、不正な暗号通貨マイニングに対して悪意を持ち再利用しようとする可能性があります。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

GuardDuty ランタイムエージェントは、複数のリソースからのイベントをモニタリングします。影響を受けるリソースを特定するには、GuardDuty コンソールの結果の詳細で [リソースタイプ] を表示し、「Runtime Monitoring 検出結果の修正」を参照してください。

Execution:Runtime/NewLibraryLoaded

新しく作成または最近変更されたライブラリが、コンテナ内のプロセスによってロードされました。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

この検出結果から、ライブラリが実行時にコンテナ内で作成または変更され、コンテナ内で実行されているプロセスによって読み込まれたことがわかります。ベストプラクティスは、実行時にはコンテナを不変のままにし、コンテナの存続期間中はバイナリファイル、スクリプト、またはライブラリを作成または変更しないことです。新しく作成または変更されたライブラリをコンテナにロードすると、不審なアクティビティが発生する可能性があります。この動作は、悪意のある攻撃者が潜在的な侵害の一環としてコンテナにアクセスし、マルウェアやその他のソフトウェアをダウンロードして実行した可能性があることを示しています。このタイプのアクティビティは侵害の兆候を示している可

能性がありますが、一般的な使用パターンでもあります。したがって、GuardDuty はメカニズムを使用してこのアクティビティの疑わしいインスタンスを識別し、疑わしいインスタンスに対してのみ、この検出結果タイプを生成します。

GuardDuty ランタイムエージェントは、複数のリソースからのイベントをモニタリングします。影響を受けるリソースを特定するには、GuardDuty コンソールの検出結果の詳細で [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

PrivilegeEscalation:Runtime/ContainerMountsHostDirectory

コンテナ内のプロセスが、実行時にホストファイルシステムをマウントしました。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

複数のコンテナエスケープ手法では、実行時にホストファイルシステムをコンテナ内にマウントします。この検出結果から、コンテナ内のプロセスがホストファイルシステムをマウントしようとした可能性があり、ホストへのエスケープが試みられた可能性があることがわかります。

GuardDuty ランタイムエージェントは、複数のリソースからのイベントをモニタリングします。影響を受けるリソースを特定するには、GuardDuty コンソールの検出結果の詳細で [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

PrivilegeEscalation:Runtime/UserfaultfdUsage

あるプロセスが、**`userfaultfd`** システム呼び出しを使用してユーザースペースのページフォールトを処理しました。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

通常、ページフォールトはカーネルスペースのカーネルによって処理されます。しかし、`userfaultfd` システムコールを使うと、プロセスはユーザースペースのファイルシステムのページフォールトを処理できるようになります。これはユーザースペースのファイルシステムの実装を可能にする便利な機能です。一方で、潜在的に悪意のあるプロセスによってユーザースペースからカーネルを妨害するためにも使用される可能性があります。`userfaultfd` システムコールを使ってカーネルを中断させることは、カーネルの競合状態を悪用している最中にレースウィンドウを延長するため、一般的に悪用の手法です。`userfaultfd` を使用すると、Amazon Elastic Compute Cloud (Amazon EC2) インスタンス上で不審なアクティビティが行われたことを示している可能性があります。

GuardDuty ランタイムエージェントは、複数のリソースからのイベントをモニタリングします。影響を受けるリソースを特定するには、GuardDuty コンソールの検出結果の詳細で [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Execution:Runtime/SuspiciousTool

コンテナまたは Amazon EC2 インスタンスは、ペンテストエンゲージメントなどの攻撃的なセキュリティシナリオで頻繁に使用されるバイナリファイルまたはスクリプトを実行しています。

デフォルトの重大度: 可変

この検出結果の重大度は、検出された不審なツールがデュアルユースとみなされるか、攻撃専用であるかに応じて、高または低のいずれかになります。

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境内の EC2 インスタンスまたはコンテナで疑わしいツールが実行されたことを知らせるものです。これには、バックドアツール、ネットワークスキャナー、ネットワークスニファーとも呼ばれる、ペンテストエンゲージメントに使用されるツールが含まれます。これらのツールはすべて、悪意のないコンテキストで使用されることがありますが、悪意のある脅威アクター

によっても頻繁に使用されます。攻撃的なセキュリティツールを観察すると、関連する EC2 インスタンスまたはコンテナが侵害されていることを示している場合があります。

GuardDuty は、関連するランタイムアクティビティとコンテキストを調べ、関連するアクティビティとコンテキストが潜在的に疑わしい場合にのみ、この検出結果を生成するようにします。

GuardDuty ランタイムエージェントは、複数のリソースからのイベントをモニタリングします。影響を受けるリソースを特定するには、GuardDuty コンソールの検出結果の詳細で [リソースタイプ] を確認してください。該当する場合は、プロセスやプロセスの系統情報など、追加のコンテキストが検出結果に含まれており、詳細な調査を行うことができます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Execution:Runtime/SuspiciousCommand

侵害を示す Amazon EC2 インスタンスまたはコンテナで疑わしいコマンドが実行されました。

デフォルトの重大度: 可変

観察された悪意のあるパターンの影響に応じて、この検出結果タイプの重大度は、低、中、高のいずれかになります。

- 機能: Runtime Monitoring

この検出結果は、疑わしいコマンドが実行されたこと、および AWS 環境内の Amazon EC2 インスタンスまたはコンテナが侵害されたことを示します。これは、ファイルが疑わしいソースからダウンロードされて実行されたか、実行中のプロセスでコマンドラインに既知の悪意のあるパターンが表示されることを意味している可能性があります。これはさらに、システムでマルウェアが実行されていることを示します。

GuardDuty は、関連するランタイムアクティビティとコンテキストを調べ、関連するアクティビティとコンテキストが潜在的に疑わしい場合にのみ、この検出結果を生成するようにします。

GuardDuty ランタイムエージェントは、複数のリソースからのイベントをモニタリングします。影響を受けるリソースを特定するには、GuardDuty コンソールの検出結果の詳細で [リソースタイプ] を確認してください。該当する場合は、プロセスやプロセスの系統情報など、追加のコンテキストが検出結果に含まれており、詳細な調査を行うことができます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

> **Note**
>
> プロセスコマンドライン引数には機密データが含まれている場合があります。機密データを保護するために、GuardDuty ランタイムモニタリングの結果には完全なコマンドライン引数は含まれません。代わりに、 は検出結果を生成したコマンドラインパターンの代表的な例`Service.RuntimeDetails.Context.CommandLineExample`を提供します。

DefenseEvasion:Runtime/SuspiciousCommand

リストされている Amazon EC2 インスタンスまたはコンテナでコマンドが実行され、ファイアウォールや重要なシステムサービスなどの Linux 防御メカニズムを変更または無効化しようとしています。

デフォルトの重大度: 可変

どの防御メカニズムを変更または無効にしたかに応じて、この検出結果タイプの重大度は、高、中、低のいずれかになります。

- 機能: Runtime Monitoring

この検出結果は、ローカルシステムのセキュリティサービスから攻撃を隠そうとするコマンドが実行されたことを知らせるものです。これには、Unix ファイアウォールの無効化、ローカル IP テーブルの変更、crontab エントリの削除、ローカルサービスの無効化、`LDPreload` 関数の引き継ぎなどのアクションが含まれます。いかなる変更も非常に疑わしく、侵害を示唆している可能性があります。したがって、これらのメカニズムはシステムのさらなる侵害を検出または防止します。

GuardDuty は、関連するランタイムアクティビティとコンテキストを調べ、関連するアクティビティとコンテキストが潜在的に疑わしい場合にのみ、この検出結果を生成するようにします。

GuardDuty ランタイムエージェントは、複数のリソースからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果の詳細で [リソースタイプ] を確認してください。該当する場合は、プロセスやプロセスの系統情報など、追加のコンテキストが検出結果に含まれており、詳細な調査を行うことができます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

DefenseEvasion:Runtime/PtraceAntiDebugging

コンテナまたは Amazon EC2 インスタンスのプロセスが、ptrace システムコールを使用してアンチデバッグ対策を実行しました。

デフォルトの重大度: [Low] (低

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境内のリストされた Amazon EC2 インスタンスまたはコンテナで実行されているプロセスが、PTRACE_TRACEME オプションを含む ptrace システムコールを使用したことを示しています。このアクティビティにより、アタッチされているデバッガーは実行中のプロセスからデタッチされます。デバッガーがアタッチされていない場合、影響はありません。ただし、アクティビティ自体が疑惑を生じさせます。これは、システムでマルウェアが実行されていることを示している可能性があります。マルウェアは、アンチデバッグの手法を頻繁に使用して分析を回避し、これらの手法は実行時に検出できます。

GuardDuty は、関連するランタイムアクティビティとコンテキストを調べ、関連するアクティビティとコンテキストが潜在的に疑わしい場合にのみ、この検出結果を生成するようにします。

GuardDuty ランタイムエージェントは、複数のリソースからのイベントをモニタリングします。影響を受けるリソースを特定するには、GuardDuty コンソールの検出結果の詳細で [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Execution:Runtime/MaliciousFileExecuted

既知の悪意のある実行可能ファイルが Amazon EC2 インスタンスまたはコンテナで実行されました。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

この検出結果は、既知の悪意のある実行可能ファイルが Amazon EC2 インスタンスまたは AWS 環境内のコンテナで実行されたことを知らせるものです。これは、インスタンスまたはコンテナが侵害された可能性があり、マルウェアが実行されたことを強く示唆します。

GuardDuty は、関連するランタイムアクティビティとコンテキストを調べ、関連するアクティビティとコンテキストが潜在的に疑わしい場合にのみ、この検出結果を生成するようにします。

GuardDuty ランタイムエージェントは、複数のリソースからのイベントをモニタリングします。影響を受けるリソースを特定するには、GuardDuty コンソールの検出結果の詳細で [リソースタイプ] を確認してください。該当する場合は、プロセスやプロセスの系統情報など、追加のコンテキストが検出結果に含まれており、詳細な調査を行うことができます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Execution:Runtime/MaliciousFileExecuted.Custom

カスタム脅威リストの悪意のある実行可能ファイルが Amazon EC2 インスタンスまたはコンテナで実行されました。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

この検出結果は、アップロードしてアクティブ化した脅威エンティティリストに含まれている実行可能ファイルがAmazon EC2 インスタンスまたは AWS 環境内のコンテナで実行されたことを知らせるものです。GuardDuty では、既知の悪意のある IP アドレス、ドメイン、SHA-256 ファイルハッシュを脅威エンティティリストに含めることができます。GuardDuty は、アップロードされた脅威エンティティリストに関連付けられたアクティビティに基づいて検出結果を生成します。脅威エンティティリストの名前は、検出結果の詳細で確認できます。

注: この検出結果は、脅威エンティティリストに含まれていたハッシュを持つ実行可能ファイルが実行された場合にのみトリガーされます。

GuardDuty ランタイムエージェントは、複数のリソースからのイベントをモニタリングします。影響を受けるリソースを特定するには、GuardDuty コンソールの検出結果の詳細で [リソースタイプ]

を確認してください。該当する場合は、プロセスやプロセスの系統情報など、追加のコンテキストが検出結果に含まれており、詳細な調査を行うことができます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Execution:Runtime/SuspiciousShellCreated

Amazon EC2 インスタンスまたはコンテナ内のネットワークサービスまたはネットワークからアクセス可能なプロセスがインタラクティブシェルプロセスを開始しました。

デフォルトの重大度: [Low] (低

- 機能: Runtime Monitoring

この検出結果は、Amazon EC2 インスタンスまたは AWS 環境内のコンテナでネットワークにアクセスできるサービスがインタラクティブシェルを起動したことを知らせるものです。特定の状況では、このシナリオは悪用後の動作を示している場合があります。インタラクティブシェルを使用すると、攻撃者は侵害されたインスタンスまたはコンテナで任意のコマンドを実行できます。

GuardDuty ランタイムエージェントは、複数のリソースからのイベントをモニタリングします。影響を受けるリソースを特定するには、GuardDuty コンソールの検出結果の詳細で [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。ネットワークからアクセス可能なプロセスの情報は、親プロセスの詳細で確認できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

PrivilegeEscalation:Runtime/ElevationToRoot

リストされた Amazon EC2 インスタンスまたはコンテナで実行されているプロセスがルート権限を引き受けています。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

この検出結果は、リストされた Amazon EC2 または AWS 環境内のリストされたコンテナで実行されているプロセスが、異常または疑わしいsetuidバイナリ実行によってルート権限を引き受けたことを知らせるものです。これは、EC2 インスタンスの実行中のプロセスが悪用または setuid 悪用を通じて侵害された可能性があることを示します。ルート権限を使用すると、攻撃者はインスタンスまたはコンテナでコマンドを実行できる可能性があります。

GuardDuty は、sudo コマンドの通常の使用を含むアクティビティに対してこの検出結果タイプを生成しないように設計されていますが、アクティビティが異常または不審であると識別されると、この検出結果が生成されます。

GuardDuty は、関連するランタイムアクティビティとコンテキストを調べ、関連するアクティビティとコンテキストが異常または不審な場合にのみ、この検出結果タイプを生成します。

GuardDuty ランタイムエージェントは、複数のリソースからのイベントをモニタリングします。影響を受けるリソースを特定するには、GuardDuty コンソールの検出結果の詳細で [リソースタイプ] を確認してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Discovery:Runtime/SuspiciousCommand

疑わしいコマンドが Amazon EC2 インスタンスまたはコンテナで実行されました。これにより、攻撃者はローカルシステム、周囲の AWS インフラストラクチャ、またはコンテナインフラストラクチャに関する情報を取得できます。

デフォルトの重大度: [Low] (低

機能: Runtime Monitoring

この検出結果は、 AWS 環境内のリストされた Amazon EC2 インスタンスまたはコンテナで実行されているプロセスが、攻撃を進展させる可能性のある重要な情報を攻撃者に提供するコマンドを実行したことを知らせるものです。次の情報が取得された可能性があります。

- ユーザーやネットワーク設定などのローカルシステム

- その他の利用可能な AWS リソースとアクセス許可、または
- サービスやポッドなどの Kubernetes インフラストラクチャ。

検出結果の詳細にリストされている Amazon EC2 インスタンスまたはコンテナが侵害されている可能性があります。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果の詳細で [リソースタイプ] を確認してください。不審なコマンドの詳細については、検出結果の JSON の `service.runtimeDetails.context` フィールドを参照してください。プロセスやプロセスの系統情報を含む追加のコンテキストは、さらなる調査のために検出結果で利用できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Persistence:Runtime/SuspiciousCommand

疑わしいコマンドが Amazon EC2 インスタンスまたはコンテナで実行され、攻撃者がお客様の AWS 環境でアクセスと制御を維持できるようになりました。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

この検出結果は、リストされた Amazon EC2 インスタンスまたは AWS 環境内のコンテナで実行されているプロセスが疑わしいコマンドを実行したことを知らせるものです。コマンドは永続化メソッドをインストールし、マルウェアが中断なく実行されるようにしたり、攻撃者が侵害された可能性のあるインスタンスまたはコンテナリソースタイプに継続的にアクセスできるようにしたりします。これは、システムサービスがインストールまたは変更されたこと、`crontab` が変更されたこと、または新しいユーザーがシステム設定に追加されたことを意味する可能性があります。

GuardDuty は、関連するランタイムアクティビティとコンテキストを調べ、関連するアクティビティとコンテキストが異常または不審な場合にのみ、この検出結果タイプを生成します。

検出結果の詳細にリストされている Amazon EC2 インスタンスまたはコンテナが侵害されている可能性があります。

GuardDuty ランタイムエージェントは、複数のリソースからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果の詳細で [リソースタイプ] を確認してください。不審なコマンドの詳細については、検出結果の JSON の `service.runtimeDetails.context` フィールドを参照してください。該当する場合は、プロセスやプロセスの系統情報など、追加のコンテキストが検出結果に含まれており、詳細な調査を行うことができます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

PrivilegeEscalation:Runtime/SuspiciousCommand

不審なコマンドが Amazon EC2 インスタンスまたはコンテナで実行されたことにより、攻撃者が権限を昇格できます。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

この検出結果は、リストされた Amazon EC2 インスタンスまたは AWS 環境内のコンテナで実行されているプロセスが疑わしいコマンドを実行したことを知らせるものです。コマンドは権限昇格の実行を試みます。これにより、攻撃者が権限の高いタスクを実行できます。

GuardDuty は、関連するランタイムアクティビティとコンテキストを調べ、関連するアクティビティとコンテキストが異常または不審な場合にのみ、この検出結果タイプを生成します。

検出結果の詳細にリストされている Amazon EC2 インスタンスまたはコンテナが侵害されている可能性があります。

GuardDuty ランタイムエージェントは、複数のリソースからのイベントをモニタリングします。影響を受けるリソースを特定するには、GuardDuty コンソールの検出結果の詳細で [リソースタイプ] を確認してください。該当する場合は、プロセスやプロセスの系統情報など、追加のコンテキストが検出結果に含まれており、詳細な調査を行うことができます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

DefenseEvasion:Runtime/KernelModuleLoaded

Amazon EC2 インスタンスに、カーネルモジュールが ロードされ、カーネルレベルのアクセスを取得しようとしたことを示します。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

この検出結果は、リストされた EC2 インスタンスにカーネルモジュールがロードされたことを示します。カーネルモジュールには最も高いシステムレベルの権限 (リング 0) があるため、これは脅威アクターがカーネルレベルのアクセスを取得していることを示している可能性があります。このレベルのアクセスにより、システムを完全にコントロールできます。

GuardDuty ランタイムエージェントは、複数のリソースからのイベントをモニタリングします。影響を受けるリソースを特定するには、GuardDuty コンソールの検出結果の詳細で [リソースタイプ] を確認してください。該当する場合は、プロセスやプロセスの系統情報など、追加のコンテキストが検出結果に含まれており、詳細な調査を行うことができます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Persistence:Runtime/SensitiveFileModified

セキュリティ上重要なシステムファイルが Amazon EC2 インスタンスまたはコンテナで変更され、攻撃者が永続的にアクセスできる可能性があります。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

この検出結果は、セキュリティ上重要なシステムファイルが Amazon EC2 インスタンスまたはコンテナで変更され、攻撃者がシステムで永続的な実行権限を維持できる可能性があることを知らせるものです。これには、新しい SSH キーファイルのインストール、ユーザー設定ファイルの変更、パスワードまたはシャドウファイルの変更、スケジュールされたタスクの作成、システム起動スクリプトの操作、不審なコンテキストでのウェブサービスプロセスによる実行可能ファイルまたはスクリプトファイルの変更が含まれます。これらの変更は、環境で不正な永続的アクセスを可能にする恐れがある侵害の兆候です。

GuardDutyは、攻撃者が永続的なアクセスのために狙う、セキュリティ上重要なシステムファイルを頻繁にモニタリングします。これらのファイルは正当なシステム管理や運用の際にも変更される可能性があるため、GuardDuty は関連するランタイムアクティビティやコンテキストを調べて、その変更が他の疑わしいアクティビティと関連している場合にのみこの検出結果を生成します。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。`service.runtimeDetails.context` フィールドには、ファイルパスやオペレーションタイプなど、疑わしいファイルの変更に関する詳細が含まれます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

PrivilegeEscalation:Runtime/SensitiveFileModified

セキュリティ上重要なシステムファイルが Amazon EC2 インスタンスまたはコンテナで変更され、攻撃者が権限を昇格できる可能性があります。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

この検出結果は、セキュリティ上重要なシステムファイルが Amazon EC2 インスタンスまたはコンテナで変更され、攻撃者がシステムで権限を昇格できる可能性があることを知らせるものです。これには、sudoers 設定、PAM 設定、システム認証ファイルなどのセキュリティ上重要なファイルへの変更が含まれる場合があります。これらの変更は、不正な権限の昇格を可能にする侵害の兆候であり、不正なルートアクセス、認証メカニズムの弱体化、またはセキュリティコントロールの侵害につながる可能性があります。

GuardDuty は、攻撃者が権限の昇格のために狙う、セキュリティ上重要なシステムファイルを頻繁にモニタリングします。これらのファイルは正当なシステム管理や運用の際にも変更される可能性があるため、GuardDuty は関連するランタイムアクティビティやコンテキストを調べて、その変更が他の疑わしいアクティビティと関連している場合にのみこの検出結果を生成します。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネ

ルで [リソースタイプ] を確認してください。`service.runtimeDetails.context` フィールドには、ファイルパスやオペレーションタイプなど、疑わしいファイルの変更に関する詳細が含まれます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

DefenseEvasion:Runtime/SensitiveFileModified

セキュリティ上重要なシステムファイルが Amazon EC2 インスタンスまたはコンテナで変更され、攻撃者が検出を回避できる可能性があります。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

この検出結果は、セキュリティ上重要なシステムファイルが Amazon EC2 インスタンスまたはコンテナで変更され、攻撃者がシステムで検出を回避できる可能性があることを知らせるものです。これには、システムログファイル、監査設定、コマンド履歴ファイル、またはシステムログの変更が含まれる場合があります。これらの変更は、不正なアクティビティの隠蔽、セキュリティモニタリングの中断、またはシステムアクティビティの証拠の削除を可能にする侵害の兆候です。

GuardDuty は、攻撃者が防御の回避のために狙う、セキュリティ上重要なシステムファイルを頻繁にモニタリングします。これらのファイルは正当なシステム管理や運用の際にも変更される可能性があるため、GuardDuty は関連するランタイムアクティビティやコンテキストを調べて、その変更が他の疑わしいアクティビティと関連している場合にのみこの検出結果を生成します。

GuardDuty ランタイムエージェントは、複数のリソースタイプからのイベントをモニタリングします。侵害されている可能性のあるリソースを特定するには、GuardDuty コンソールの検出結果パネルで [リソースタイプ] を確認してください。`service.runtimeDetails.context` フィールドには、ファイルパスやオペレーションタイプなど、疑わしいファイルの変更に関する詳細が含まれます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

カスタム検出ルールを必要とする検出タイプ

GuardDuty は、ランタイムモニタリングのカスタム検出ルールを有効にした場合にのみ、次の検出結果タイプを生成します。詳細については、「GuardDuty のカスタム検出ルール」を参照してください。

各検出結果には、検出結果を生成したルールの名前を含む `CustomDetectionRuleName`フィールドが含まれます。

トピック

- CommandAndControl:Runtime/SuspiciousCommand
- CredentialAccess:Runtime/SensitiveFileRead
- Discovery:Runtime/SensitiveFileRead
- Exfiltration:Runtime/SuspiciousCommand

CommandAndControl:Runtime/SuspiciousCommand

Amazon EC2 インスタンスまたはコンテナのプロセスは、攻撃者がリソースとの通信と制御に使用できる疑わしいコマンドを実行しました。

デフォルトの重大度: 可変

重要度は、検出結果を生成したルールによって異なります。 では highDnsTunnelingToolExecuted、では mediumSuspiciousProtocolTunneling。

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境内の Amazon EC2 インスタンスまたはコンテナのプロセスが、攻撃者が検出を回避しながら侵害されたリソースと通信するために使用できるコマンドを実行したことを知らせるものです。例えば、このプロセスでは、DNS 経由でトラフィックをトンネルするツールや、トンネルまたはプロキシサービスを介してリソースを公開するツールを実行している場合があります。

GuardDuty は、 DnsTunnelingToolExecuted および SuspiciousProtocolTunnelingルールからこの結果を生成します。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

CredentialAccess:Runtime/SensitiveFileRead

Amazon EC2 インスタンスまたはコンテナのプロセスが、認証情報を含む可能性のある機密ファイルにアクセスしました。

デフォルトの重大度: [High] (高

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境内の Amazon EC2 インスタンスまたはコンテナのプロセスが、認証情報やその他のシークレットを含む可能性のあるセキュリティ機密ファイルにアクセスしたことを知らせるものです。たとえば、プロセスは`/proc`ファイルシステムを介して別のプロセスのメモリにアクセスして、そこから認証情報を抽出した可能性があります。

GuardDuty はDumpProcessMemoryFromProcルールからこの結果を生成します。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Discovery:Runtime/SensitiveFileRead

Amazon EC2 インスタンスまたはコンテナのプロセスが、ローカルユーザーと認証情報に関する情報を含む機密ファイルにアクセスしました。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境内の Amazon EC2 インスタンスまたはコンテナのプロセスが、ローカルユーザーアカウントやパスワードハッシュファイルなどのセキュリティに敏感なシステムファイルにアクセスしたことを知らせるものです。攻撃者はこの情報を使用してアカウントを検出し、パスワードのクラックを試みることができます。

GuardDuty はPasswordShadowFileDiscoveryルールからこの結果を生成します。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Exfiltration:Runtime/SuspiciousCommand

Amazon EC2 インスタンスまたはコンテナのプロセスで、攻撃者が環境からデータを移動するために使用できる疑わしいコマンドが実行されました。

デフォルトの重大度: [Medium] (中

- 機能: Runtime Monitoring

この検出結果は、 AWS 環境内の Amazon EC2 インスタンスまたはコンテナのプロセスが、環境からデータを移動するために使用できる可能性のあるコマンドを実行したことを知らせるものです。例えば、転送制限を回避するために、プロセスはファイルを外部エンドポイントまたはクラウドストレージサービスにアップロードしたり、データを小さな部分に分割したりできます。

GuardDuty は、CurlWgetExfiltrateData、、DataSplittingForExfiltrationおよびS3DataExfiltrationWithRcloneルールからこの結果を生成します。

修復のレコメンデーション

このアクティビティが予期しないものである場合、リソースが侵害されている可能性があります。詳細については、「Runtime Monitoring 検出結果の修正」を参照してください。

Malware Protection for EC2 の検出結果のタイプ

GuardDuty Malware Protection for EC2 は、EC2 インスタンスまたはコンテナワークロードのスキャン中に検出された、すべての脅威に対する単一の Malware Protection for EC2 の検出結果を提供します。この検出結果には、スキャン中に行われた検出の合計数が含まれ、重大度に基づいて、検出された上位 32 個の脅威の詳細が示されます。他の GuardDuty の検出結果とは異なり、同じ EC2 インスタンスまたはコンテナワークロードが再度スキャンされても、Malware Protection for EC2 の検出結果は更新されません。

マルウェアを検出するスキャンごとに、新しい Malware Protection for EC2 の検出結果が生成されます。Malware Protection for EC2 の検出結果には、検出結果を生成した対応するスキャンに関する情

報と、このスキャンを開始した GuardDuty の検出結果が含まれます。これにより、疑わしい動作と検出されたマルウェアとの関連付けがより簡単になります。

> **Note**
>
> GuardDuty がコンテナワークロードで悪意のあるアクティビティを検出する場合、Malware Protection for EC2 は EC2 レベルの検出結果を生成しません。

次の検出結果は、GuardDuty Malware Protection for EC2 に固有です。

トピック

- Execution:EC2/MaliciousFile
- Execution:ECS/MaliciousFile
- Execution:Kubernetes/MaliciousFile
- Execution:Container/MaliciousFile
- Execution:EC2/SuspiciousFile
- Execution:ECS/SuspiciousFile
- Execution:Kubernetes/SuspiciousFile
- Execution:Container/SuspiciousFile

Execution:EC2/MaliciousFile

悪意のあるファイルが EC2 インスタンスで検出されました。

デフォルトの重大度: 検出された脅威によって異なります。

- 機能: EBS Malware Protection

この検出結果は、GuardDuty Malware Protection for EC2 スキャンが AWS 、環境内のリストされた EC2 インスタンスで 1 つ以上の悪意のあるファイルを検出したことを示します。このリスト化したインスタンスは侵害されている可能性があります。詳細については、次を参照してください。検出結果の詳細にある「脅威の検出」を参照してください。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Execution:ECS/MaliciousFile

悪意のあるファイルが ECS クラスターで検出されました。

デフォルトの重大度: 検出された脅威によって異なります。

- 機能: EBS Malware Protection

この検出結果は、GuardDuty Malware Protection for EC2 スキャンが ECS クラスターに属するコンテナワークロードで 1 つ以上の悪意のあるファイルを検出したことを示しています。詳細については、次を参照してください。検出結果の詳細にある「脅威の検出」を参照してください。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、ECS クラスターに属するコンテナが侵害されている可能性があります。詳細については、「侵害された可能性のある ECS クラスターの修復」を参照してください。

Execution:Kubernetes/MaliciousFile

悪意のあるファイルが Kubernetes クラスターで検出されました。

デフォルトの重大度: 検出された脅威によって異なります。

- 機能: EBS Malware Protection

この検出結果は、GuardDuty Malware Protection for EC2 スキャンが Kubernetes クラスターに属するコンテナワークロードで 1 つ以上の悪意のあるファイルを検出したことを示しています。これが EKS マネージドクラスターの場合、検出結果の詳細には、影響を受ける EKS リソースに関する追加情報が表示されます。詳細については、次を参照してください。検出結果の詳細にある「脅威の検出」を参照してください。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、コンテナのワークロードが侵害されている可能性があります。詳細については、「EKS Protection の検出結果の修復」を参照してください。

Execution:Container/MaliciousFile

悪意のあるファイルがスタンドアロンコンテナで検出されました。

デフォルトの重大度: 検出された脅威によって異なります。

- 機能: EBS Malware Protection

この検出結果は、GuardDuty Malware Protection for EC2 スキャンがコンテナワークロードで 1 つ以上の悪意のあるファイルを検出し、クラスター情報が特定されていないことを示しています。詳細については、次を参照してください。検出結果の詳細にある「脅威の検出」を参照してください。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、コンテナのワークロードが侵害されている可能性があります。詳細については、「侵害された可能性のあるスタンドアロンコンテナの修復」を参照してください。

Execution:EC2/SuspiciousFile

EC2 インスタンスで疑わしいファイルが検出されました。

デフォルトの重大度: 検出された脅威によって異なります。

- 機能: EBS Malware Protection

この検出結果は、GuardDuty Malware Protection for EC2 スキャンが EC2 インスタンスで 1 つ以上の疑わしいファイルを検出したことを示します。詳細については、次を参照してください。検出結果の詳細にある「脅威の検出」を参照してください。

`SuspiciousFile` タイプの検出では、アドウェア、スパイウェア、デュアルユースツールなどの望ましくないであろうプログラムが、影響を受けるリソースに存在することが示されます。これらのプログラムは、リソースに悪影響を及ぼしたり、攻撃者が悪意のある目的で使用された可能性があります。たとえば、ネットワークツールは、攻撃者がリソースを侵害しようとするハッキングツールとして正当または悪意を持って使用できます。

疑わしいファイルが検出されたら、検出されたファイルが AWS 環境で表示されるかどうかを評価します。ファイルが予期しないものである場合は、次のセクションで提供される修復の推奨事項に従ってください。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Execution:ECS/SuspiciousFile

疑わしいファイルが ECS クラスターで検出されました。

デフォルトの重大度: 検出された脅威によって異なります。

- 機能: EBS Malware Protection

この検出結果は、GuardDuty Malware Protection for EC2 スキャンが ECS クラスターに属するコンテナで 1 つ以上の疑わしいファイルを検出したことを示しています。詳細については、次を参照してください。検出結果の詳細にある「脅威の検出」を参照してください。

`SuspiciousFile` タイプの検出では、アドウェア、スパイウェア、デュアルユースツールなどの望ましくないであろうプログラムが、影響を受けるリソースに存在することが示されます。これらのプログラムは、リソースに悪影響を及ぼしたり、攻撃者が悪意のある目的で使用された可能性があります。たとえば、ネットワークツールは、攻撃者がリソースを侵害しようとするハッキングツールとして正当または悪意を持って使用できます。

疑わしいファイルが検出されたら、検出されたファイルが AWS 環境で表示されるかどうかを評価します。ファイルが予期しないものである場合は、次のセクションで提供される修復の推奨事項に従ってください。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、ECS クラスターに属するコンテナが侵害されている可能性があります。詳細については、「侵害された可能性のある ECS クラスターの修復」を参照してください。

Execution:Kubernetes/SuspiciousFile

疑わしいファイルが Kubernetes クラスターで検出されました。

デフォルトの重大度: 検出された脅威によって異なります。

- 機能: EBS Malware Protection

この検出結果は、GuardDuty Malware Protection for EC2 スキャンが Kubernetes クラスターに属するコンテナで 1 つ以上の疑わしいファイルを検出したことを示しています。これが EKS マネージドクラスターの場合、検出結果の詳細には、影響を受ける EKS に関する追加情報が表示されます。詳細については、次を参照してください。検出結果の詳細にある「脅威の検出」を参照してください。

`SuspiciousFile` タイプの検出では、アドウェア、スパイウェア、デュアルユースツールなどの望ましくないであろうプログラムが、影響を受けるリソースに存在することが示されます。これらのプログラムは、リソースに悪影響を及ぼしたり、攻撃者が悪意のある目的で使用された可能性があります。たとえば、ネットワークツールは、攻撃者がリソースを侵害しようとするハッキングツールとして正当または悪意を持って使用できます。

疑わしいファイルが検出されたら、検出されたファイルが AWS 環境で表示されるかどうかを評価します。ファイルが予期しないものである場合は、次のセクションで提供される修復の推奨事項に従ってください。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、コンテナのワークロードが侵害されている可能性があります。詳細については、「EKS Protection の検出結果の修復」を参照してください。

Execution:Container/SuspiciousFile

疑わしいファイルがスタンドアロンコンテナで検出されました。

デフォルトの重大度: 検出された脅威によって異なります。

- 機能: EBS Malware Protection

この検出結果は、GuardDuty Malware Protection for EC2 スキャンが、クラスター情報を持たないコンテナ上で 1 つ以上の疑わしいファイルを検出したことを示しています。詳細については、次を参照してください。検出結果の詳細にある「脅威の検出」を参照してください。

`SuspiciousFile` タイプの検出では、アドウェア、スパイウェア、デュアルユースツールなどの望ましくないであろうプログラムが、影響を受けるリソースに存在することが示されます。これらのプログラムは、リソースに悪影響を及ぼしたり、攻撃者が悪意のある目的で使用された可能性があります。たとえば、ネットワークツールは、攻撃者がリソースを侵害しようとするハッキングツールとして正当または悪意を持って使用できます。

疑わしいファイルが検出されたら、検出されたファイルが AWS 環境で表示されるかどうかを評価します。ファイルが予期しないものである場合は、次のセクションで提供される修復の推奨事項に従ってください。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、コンテナのワークロードが侵害されている可能性があります。詳細については、「侵害された可能性のあるスタンドアロンコンテナの修復」を参照してください。

Malware Protection for S3 の検出結果タイプ

GuardDuty は、 AWS アカウントで潜在的なセキュリティ脅威を検出した場合にのみ、検出結果を生成します。Malware Protection for S3 の検出結果は、マルウェアスキャンを開始したアップロードされたオブジェクトに悪意のある可能性があるファイルが含まれていることを示します。

Amazon GuardDuty が で結果を生成するには AWS アカウント、GuardDuty と Malware Protection for S3 の両方を有効にします。ベストプラクティスは、まず GuardDuty を有効にし、次に Malware Protection for S3 を有効にすることです。この順序が異なる場合は、保護されたバケットに S3 オブジェクトがアップロードされる前に、GuardDuty を必ず有効にしてください。

> **Note**
>
> GuardDuty は、GuardDuty を有効にする前にスキャンされた S3 オブジェクトの検出結果を生成できません。既存の S3 オブジェクトをスキャンするには、再度アップロードすることができます。

Object:S3/MaliciousFile

スキャンされた S3 オブジェクトで悪意のあるファイルが検出されました。

デフォルトの重大度: [High] (高

- 機能: Malware Protection for S3

この検出結果は、マルウェアスキャンが、リストされている S3 オブジェクトを悪意のあるものとして検出したことを示します。詳細については、検出結果の詳細パネルの [検出された脅威] セクションを参照してください。

修復のレコメンデーション:

この検出結果を予期していなかった場合、S3 オブジェクトに悪意がある可能性があります。推奨される修復手順については、「悪意のある可能性がある S3 オブジェクトの修復」を参照してください。

Malware Protection for Backup の検出結果タイプ

GuardDuty Malware Protection for Backup は、リクエストされたリソースのスキャン中に検出されたすべての脅威に対して単一の検出結果を提供します。この検出結果には、スキャン中に行われた検出の合計数が含まれ、重大度に基づいて、検出された上位 32 個の脅威の詳細が示されます。他の GuardDuty の検出結果とは異なり、同じリソースを再度スキャンしても Malware Protection for Backup の検出結果は更新されません。Malware Protection for Backup の新しい検出結果は、マルウェアを検出するスキャンごとに生成されます。

以下の検出結果は、GuardDuty Malware Protection for Backup に固有のものです。

トピック

- Execution:EC2/MaliciousFile!Snapshot
- Execution:EC2/MaliciousFile!AMI
- Execution:EC2/MaliciousFile!RecoveryPoint
- Execution:S3/MaliciousFile!RecoveryPoint

Execution:EC2/MaliciousFile!Snapshot

EBS スナップショットで悪意のあるファイルが検出されました。

デフォルトの重大度: 検出された脅威によって異なります。

- 機能: Malware Protection for Backup

この検出結果は、GuardDuty Malware Protection for Backup スキャンが、環境内の EBS スナップショットで 1 つ以上の悪意のあるファイルを検出したことを示しています。詳細については、検出結果の詳細パネルの [検出された脅威] セクションを参照してください。

修復のレコメンデーション

これが予期しないものである場合、スナップショットは侵害されている可能性があります。詳細については、「侵害された可能性のある EBS スナップショットの修復」を参照してください。

Execution:EC2/MaliciousFile!AMI

EC2 AMI で悪意のあるファイルが検出されました。

デフォルトの重大度: 検出された脅威によって異なります。

- 機能: Malware Protection for Backup

この検出結果は、GuardDuty Malware Protection for Backup スキャンが、環境内の AMI で 1 つ以上の悪意のあるファイルを検出したことを示しています。詳細については、検出結果の詳細パネルの [検出された脅威] セクションを参照してください。

修復のレコメンデーション

これが予期しないものである場合、AMI は侵害されている可能性があります。詳細については、「侵害された可能性のある EC2 AMI の修復」を参照してください。

Execution:EC2/MaliciousFile!RecoveryPoint

Backup AWS EC2 リカバリポイントで悪意のあるファイルが検出されました。

デフォルトの重大度: 検出された脅威によって異なります。

- 機能: Malware Protection for Backup

この検出結果は、GuardDuty Malware Protection for Backup スキャンが、環境内の EC2 復旧ポイントで 1 つ以上の悪意のあるファイルを検出したことを示しています。影響を受けた復旧ポイントは、EBS スナップショットまたは EC2 AMI である可能性があります。詳細については、検出結果の詳細パネルの [検出された脅威] セクションを参照してください。

修復のレコメンデーション

これが予期しないものである場合、EC2 復旧ポイントは侵害されている可能性があります。詳細については、「侵害された可能性のある EC2 復旧ポイントの修復」を参照してください。

Execution:S3/MaliciousFile!RecoveryPoint

Backup AWS S3 復旧ポイントで悪意のあるファイルが検出されました。

デフォルトの重大度: 検出された脅威によって異なります。

- 機能: Malware Protection for Backup

この検出結果は、GuardDuty Malware Protection for Backup スキャンが、環境内の S3 復旧ポイントで 1 つ以上の悪意のあるオブジェクトを検出したことを示しています。詳細については、検出結果の詳細パネルの [検出された脅威] セクションを参照してください。

修復のレコメンデーション

これが予期しないものである場合、S3 復旧ポイントは侵害されている可能性があります。詳細については、「侵害された可能性のある S3 復旧ポイントの修復」を参照してください。

GuardDuty RDS Protection の検出結果タイプ

GuardDuty RDS Protection は、データベースインスタンスでの異常なログイン動作を検出します。以下の検出結果は サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース 固有のものであり、リソースタイプは `RDSDBInstance` または `RDSLimitlessDB` です。結果の重大度と詳細は、結果タイプによって異なります。

トピック

- CredentialAccess:RDS/AnomalousBehavior.SuccessfulLogin
- CredentialAccess:RDS/AnomalousBehavior.FailedLogin
- CredentialAccess:RDS/AnomalousBehavior.SuccessfulBruteForce
- CredentialAccess:RDS/MaliciousIPCaller.SuccessfulLogin
- CredentialAccess:RDS/MaliciousIPCaller.FailedLogin
- Discovery:RDS/MaliciousIPCaller
- CredentialAccess:RDS/TorIPCaller.SuccessfulLogin
- CredentialAccess:RDS/TorIPCaller.FailedLogin
- Discovery:RDS/TorIPCaller
- Exfiltration:RDS/AnomalousBehavior

- Impact:RDS/AnomalousBehavior.Drop
- Impact:RDS/AnomalousBehavior.Delete
- Exfiltration:RDS/MaliciousIPCaller
- Exfiltration:RDS/TorIPCaller
- Impact:RDS/MaliciousIPCaller
- Impact:RDS/TorIPCaller

CredentialAccess:RDS/AnomalousBehavior.SuccessfulLogin

ユーザーがアカウント内の RDS データベースへのログインに、異常な方法で成功しました。

デフォルトの重大度: 可変

> **Note**
>
> この検出結果に関連する異常な動作に応じて、デフォルトの重大度は「低」、「中」、「高」になります。
>
> - 低 - この検出結果に関連するユーザー名が、プライベートネットワークに関連付けられた IP アドレスからログインした場合。
> - 中 - この検出結果に関連するユーザー名がパブリック IP アドレスからログインした場合。
> - 高 - パブリック IP アドレスからのログイン試行の失敗が一貫して続いている場合は、アクセスポリシーが過度に制限されていることを示します。

- 機能: RDS ログインアクティビティのモニタリング

この検出結果は、 AWS 環境内の RDS データベースで異常なログイン成功が観察されたことを知らせるものです。これは、以前に見たことのないユーザーが RDS データベースに初めてログインしたことを示している可能性があります。一般的なシナリオは、個々のユーザーではなくアプリケーションがプログラム的にアクセスするデータベースに内部ユーザーがログインする場合です。

このログイン成功は、GuardDuty の異常検出機械学習 (ML) モデルにより、異常として識別されています。機械学習モデルは、サポートされている Amazon Aurora、Amazon RDS および Aurora

Limitless データベース 内のすべてのデータベースログインイベントを評価し、攻撃者が使用するテクニックに関連する異常なイベントを特定します。機械学習モデルは、リクエストを行ったユーザー、リクエストが行われた場所、使用された特定のデータベース接続の詳細など、RDS ログインアクティビティのさまざまな要因を追跡します。潜在的に異常なログインイベントについては、「RDS ログインアクティビティベースの異常」を参照してください。

修復のレコメンデーション

関連付けられたデータベースに対するこのアクティビティが想定外の場合は、関連するデータベースユーザーのパスワードを変更し、利用可能な監査ログを確認することをお勧めします。中および高の重大度の検出結果は、データベースへのアクセスポリシーが過度に寛容であり、ユーザーの認証情報が漏洩または侵害された可能性があることを示している可能性があります。データベースをプライベート VPC に配置し、必要なソースからのみトラフィックを許可するようにセキュリティグループのルールを制限することをお勧めします。詳細については、「ログインイベントの成功により、侵害された可能性のあるデータベースの修復」を参照してください。

CredentialAccess:RDS/AnomalousBehavior.FailedLogin

アカウント内の RDS データベースで、1 回以上の異常なログイン失敗が確認されました。

デフォルトの重大度: [Low] (低

- 機能: RDS ログインアクティビティのモニタリング

この検出結果は、 AWS 環境内の RDS データベースで 1 つ以上の異常なログイン失敗が観察されたことを知らせるものです。パブリック IP アドレスからのログインに失敗した場合は、アカウントの RDS データベースが、潜在的に悪意のある攻撃者によるブルートフォース攻撃の試みを受けている可能性があります。

この失敗したログインは、GuardDuty の異常検出機械学習 (ML) モデルにより、異常として識別されています。機械学習モデルは、サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース 内のすべてのデータベースログインイベントを評価し、攻撃者が使用するテクニックに関連する異常なイベントを特定します。機械学習モデルは、リクエストを行ったユーザー、リクエストが行われた場所、使用された特定のデータベース接続の詳細など、RDS ログインアクティビティのさまざまな要因を追跡します。潜在的に異常な RDS ログインアクティビティについては、「RDS ログインアクティビティベースの異常」を参照してください。

修復のレコメンデーション

関連付けられたデータベースに対するこのアクティビティが想定外の場合は、データベースが公開されているか、データベースのアクセスポリシーが過度に許容されている可能性があります。データベースをプライベート VPC に配置し、必要なソースからのみトラフィックを許可するようにセキュリティグループのルールを制限することをお勧めします。詳細については、「ログインイベントの失敗により、侵害された可能性のあるデータベースの修正」を参照してください。

CredentialAccess:RDS/AnomalousBehavior.SuccessfulBruteForce

通常とは異なるログイン試行を繰り返したユーザーが、パブリック IP アドレスからアカウントの RDS データベースに異常な方法でのログインに成功しました。

デフォルトの重大度: [High] (高

- 機能: RDS ログインアクティビティのモニタリング

この検出結果は、 AWS 環境内の RDS データベースでブルートフォースが成功したことを示す異常なログインが観察されたことを知らせるものです。異常なログインの成功の前に、異常なログイン試行失敗の一貫したパターンが観察されました。これは、アカウント内の RDS データベースに関連付けられているユーザーとパスワードが侵害され、悪質な攻撃者によって RDS データベースにアクセスされた可能性があることを示しています。

このブルートフォースの成功は、GuardDuty の異常検出機械学習 (ML) モデルにより、異常として識別されています。機械学習モデルは、サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース 内のすべてのデータベースログインイベントを評価し、攻撃者が使用するテクニックに関連する異常なイベントを特定します。機械学習モデルは、リクエストを行ったユーザー、リクエストが行われた場所、使用された特定のデータベース接続の詳細など、RDS ログインアクティビティのさまざまな要因を追跡します。潜在的に異常な RDS ログインアクティビティについては、「RDS ログインアクティビティベースの異常」を参照してください。

修復のレコメンデーション

このアクティビティは、データベースの認証情報が漏洩または侵害された可能性があることを示しています。関連するデータベースユーザーのパスワードを変更し、利用可能な監査ログを確認して、侵害された可能性のあるユーザーが行ったアクティビティを確認することをお勧めします。通常とは異なるログイン試行が繰り返される場合は、データベースへのアクセスポリシーが過度に寛容であるか、データベースが公開されている可能性があります。データベースをプライベート VPC に配置し、必要なソースからのみトラフィックを許可するようにセキュリティグループのルールを制限することをお勧めします。詳細については、「ログインイベントの成功により、侵害された可能性のあるデータベースの修復」を参照してください。

CredentialAccess:RDS/MaliciousIPCaller.SuccessfulLogin

ユーザーが既知の悪意のある IP アドレスからアカウントの RDS データベースへのログインに成功しました。

デフォルトの重大度: [High] (高

- 機能: RDS ログインアクティビティのモニタリング

この検出結果は、 AWS 環境内の既知の悪意のあるアクティビティに関連付けられている IP アドレスから正常な RDS ログインアクティビティが発生したことを知らせるものです。これは、アカウント内の RDS データベースに関連付けられているユーザーとパスワードが侵害され、悪質な攻撃者によって RDS データベースにアクセスされた可能性があることを示しています。

修復のレコメンデーション

関連付けられたデータベースに対するこのアクティビティが想定外の場合は、ユーザーの認証情報が公開されているか、侵害されている可能性があります。関連するデータベースユーザーのパスワードを変更し、利用可能な監査ログを確認して、侵害されたユーザーが行ったアクティビティを確認することをお勧めします。このアクティビティは、データベースへのアクセスポリシーが過度に寛容であるか、データベースが公開されていることを示している可能性もあります。データベースをプライベート VPC に配置し、必要なソースからのみトラフィックを許可するようにセキュリティグループのルールを制限することをお勧めします。詳細については、「ログインイベントの成功により、侵害された可能性のあるデータベースの修復」を参照してください。

CredentialAccess:RDS/MaliciousIPCaller.FailedLogin

既知の悪意のあるアクティビティに関連する IP アドレスが、アカウントの RDS データベースへのログインに失敗しました。

デフォルトの重大度: [Medium] (中

- 機能: RDS ログインアクティビティのモニタリング

この検出結果は、既知の悪意のあるアクティビティに関連付けられた IP アドレスが AWS 環境内の RDS データベースにログインしようとしたが、正しいユーザー名またはパスワードを指定できなかったことを知らせるものです。これは、潜在的に悪意のある攻撃者がアカウントの RDS データベースを侵害しようとしている可能性があることを示しています。

修復のレコメンデーション

関連付けられたデータベースに対するこのアクティビティが想定外の場合は、データベースのアクセスポリシーが過度に許容されているか、データベースが公開されている可能性があります。データベースをプライベート VPC に配置し、必要なソースからのみトラフィックを許可するようにセキュリティグループのルールを制限することをお勧めします。詳細については、「ログインイベントの失敗により、侵害された可能性のあるデータベースの修正」を参照してください。

Discovery:RDS/MaliciousIPCaller

既知の悪意のあるアクティビティに関連する IP アドレスがアカウントの RDS データベースを調べましたが、認証は行われませんでした。

デフォルトの重大度: [Medium] (中

- 機能: RDS ログインアクティビティのモニタリング

この検出結果は、既知の悪意のあるアクティビティに関連付けられた IP アドレスが、ログイン試行は行われませんでしたが、 AWS 環境内の RDS データベースをプローブしたことを知らせるものです。これは、潜在的に悪意のある攻撃者が公的にアクセス可能なインフラストラクチャをスキャンしようとしていることを示している可能性があります。

修復のレコメンデーション

関連付けられたデータベースに対するこのアクティビティが想定外の場合は、データベースのアクセスポリシーが過度に許容されているか、データベースが公開されている可能性があります。データベースをプライベート VPC に配置し、必要なソースからのみトラフィックを許可するようにセキュリティグループのルールを制限することをお勧めします。詳細については、「ログインイベントの失敗により、侵害された可能性のあるデータベースの修正」を参照してください。

CredentialAccess:RDS/TorIPCaller.SuccessfulLogin

ユーザーが Tor 出口ノードの IP アドレスからアカウントの RDS データベースへのログインに成功しました。

デフォルトの重大度: [High] (高

- 機能: RDS ログインアクティビティのモニタリング

この検出結果は、ユーザーが Tor 出口ノードの IP アドレスから AWS 環境内の RDS データベースに正常にログインしたことを示します。Tor は匿名通信を有効化するソフトウェアです。一連のネットワークノード間のリレーを介して通信を暗号化し、ランダムにバウンスします。最後の Tor ノードは終了ノードと呼ばれます。これは、匿名ユーザーの真のアイデンティティを隠しているという意図により、アカウント内の RDS リソースへの不正アクセスを示している場合があります。

修復のレコメンデーション

関連付けられたデータベースに対するこのアクティビティが想定外の場合は、ユーザーの認証情報が公開されているか、侵害されている可能性があります。関連するデータベースユーザーのパスワードを変更し、利用可能な監査ログを確認して、侵害されたユーザーが行ったアクティビティを確認することをお勧めします。このアクティビティは、データベースへのアクセスポリシーが過度に寛容であるか、データベースが公開されていることを示している可能性もあります。データベースをプライベート VPC に配置し、必要なソースからのみトラフィックを許可するようにセキュリティグループのルールを制限することをお勧めします。詳細については、「ログインイベントの成功により、侵害された可能性のあるデータベースの修復」を参照してください。

CredentialAccess:RDS/TorIPCaller.FailedLogin

Tor IP アドレスはアカウントの RDS データベースにログインしようとしましたが、失敗しました。

デフォルトの重大度: [Medium] (中

- 機能: RDS ログインアクティビティのモニタリング

この検出結果は、Tor 終了ノードの IP アドレスが AWS 環境内の RDS データベースにログインしようとしましたが、正しいユーザー名またはパスワードを指定できなかったことを知らせるものです。Tor は、匿名通信を有効にするためのソフトウェアです。一連のネットワークノード間のリレーを介して通信を暗号化し、ランダムにバウンスします。最後の Tor ノードは終了ノードと呼ばれます。これは、匿名ユーザーの真のアイデンティティを隠しているという意図により、アカウント内の RDS リソースへの不正アクセスを示している場合があります。

修復のレコメンデーション

関連付けられたデータベースに対するこのアクティビティが想定外の場合は、データベースのアクセスポリシーが過度に許容されているか、データベースが公開されている可能性があります。データベースをプライベート VPC に配置し、必要なソースからのみトラフィックを許可するようにセキュ

リティグループのルールを制限することをお勧めします。詳細については、「ログインイベントの失敗により、侵害された可能性のあるデータベースの修正」を参照してください。

Discovery:RDS/TorIPCaller

Tor 出口ノードの IP アドレスがアカウントの RDS データベースを調べましたが、認証は行われませんでした。

デフォルトの重大度: [Medium] (中

- 機能: RDS ログインアクティビティのモニタリング

この検出結果は、Tor 出口ノードの IP アドレスがお客様の AWS 環境の RDS データベースを確認することをお勧めします。これは、潜在的に悪意のある攻撃者が公的にアクセス可能なインフラストラクチャをスキャンしようとしていることを示している可能性があります。Tor は、匿名通信を有効にするためのソフトウェアです。通信を暗号化し、一連のネットワークノード間のリレー中にランダムに通信をバウンスさせます。最後の Tor ノードは終了ノードと呼ばれます。これは、悪意のある攻撃者の真のアイデンティティを隠して、アカウント内の RDS リソースへの不正アクセスを行っていることを示している可能性があります。

修復のレコメンデーション

関連付けられたデータベースに対するこのアクティビティが想定外の場合は、データベースのアクセスポリシーが過度に許容されているか、データベースが公開されている可能性があります。データベースをプライベート VPC に配置し、必要なソースからのみトラフィックを許可するようにセキュリティグループのルールを制限することをお勧めします。詳細については、「ログインイベントの失敗により、侵害された可能性のあるデータベースの修正」を参照してください。

Exfiltration:RDS/AnomalousBehavior

アカウントの RDS データベースから大量のデータを抽出するユーザーからの異常なアクティビティ。

デフォルトの重大度: [Low] (低

> **Note**
>
> この検出結果に関連する異常な動作に応じて、デフォルトの重要度は低、中、高のいずれかになります。

- 低 – この検出結果に関連付けられたユーザー名が、プライベートネットワークに関連付けられた IP アドレスからSELECTステートメントを実行した場合。
- Medium – この検出結果に関連付けられたユーザー名がパブリック IP アドレスからSELECTステートメントを実行した場合。
- 高 – この検出結果に関連付けられたユーザー名が、異常に大量の行を返すSELECTステートメントを実行し、アクティビティが以前に見えないパブリックネットワークから発生したか、高特権データベースロールまたは管理クライアントによって実行された場合。

- 機能: RDS DB アクティビティモニタリング

この検出結果は、 AWS 環境内の RDS データベースでデータ流出と一致する異常なデータベースアクティビティが観察されたことを知らせるものです。これは、データベースユーザーが通常よりも大幅に多くの行を選択したか、新しいユーザー、IP アドレス、またはボリュームの急増に寄与したデータベース名など、以前には見られなかったデータベース属性がアクティビティに関与したことを示している可能性があります。一般的なシナリオは、侵害されたデータベース認証情報を使用して、異常なアクセスパターンとクエリボリュームの増加を組み合わせてデータを抽出することです。

このアクティビティは、GuardDuty 異常検出機械学習 (ML) モデルによって異常として識別されました。ML モデルは、 内のすべてのデータベースアクティビティイベントを評価しサポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース、攻撃者が使用する手法に関連する異常なイベントを特定します。ML モデルは、リクエストを行ったユーザー、実行されたクエリ、アクセスされたデータ量、使用された特定のデータベース接続の詳細など、RDS データベースアクティビティのさまざまな要因を追跡します。異常の可能性があるデータベースアクティビティの詳細については、「」を参照してくださいRDS データアクティビティベースの異常。

修復のレコメンデーション

関連付けられたデータベースでこのアクティビティが予期しない場合は、関連付けられたデータベースユーザーのパスワードを変更し、使用可能な監査ログを確認して、異常なユーザーがどのデータにアクセスしたかを判断することをお勧めします。データベースをプライベート VPC に配置し、必要なソースからのトラフィックのみを許可するようにセキュリティグループのルールを制限することをお勧めします。詳細については、「異常な流出アクティビティによる侵害された可能性のあるデータベースの修復」を参照してください。

Impact:RDS/AnomalousBehavior.Drop

アカウントの RDS データベースからデータを削除するユーザーからの異常なアクティビティ。

デフォルトの重大度: [Medium] (中

> **Note**
>
> この検出結果に関連する異常な動作に応じて、デフォルトの重要度は中または高になります。
>
> - Medium – この検出結果に関連付けられたユーザー名が異常なDROPステートメントを実行した場合。
> - 高 – この結果に関連付けられたユーザー名が複数のDROPステートメントを実行し、アクティビティが以前に見えないパブリックネットワークから発生したか、高特権データベースロールまたは管理クライアントによって実行された場合。

- 機能: RDS DB アクティビティモニタリング

この検出結果は、 AWS 環境内の RDS データベースで破壊的オペレーションと一致する異常なデータベースアクティビティが観察されたことを知らせるものです。これは、データベースユーザーが通常よりも大幅に多くのテーブルまたはデータベースオブジェクトを削除した、またはアクティビティが、破壊的なオペレーションに寄与した新しいユーザー、IP アドレス、アプリケーション名などの以前には見えなかったデータベース属性に関与したことを示している可能性があります。一般的なシナリオは、侵害されたデータベース認証情報を使用して、異常なアクセスパターンとステートメントDROPボリュームの増加を組み合わせてデータベース構造を破壊することです。

このアクティビティは、GuardDuty 異常検出機械学習 (ML) モデルによって異常として識別されました。ML モデルは、 内のすべてのデータベースアクティビティイベントを評価しサポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース、攻撃者が使用する手法に関連する異常なイベントを特定します。ML モデルは、リクエストを行ったユーザー、実行されたステートメント、ターゲットとするデータベースオブジェクト、使用された特定のデータベース接続の詳細など、RDS データベースアクティビティのさまざまな要因を追跡します。異常の可能性があるデータベースアクティビティの詳細については、「」を参照してくださいRDS データアクティビティベースの異常。

修復のレコメンデーション

関連付けられたデータベースでこのアクティビティが予期しない場合は、関連付けられたデータベースユーザーのパスワードを直ちに変更し、アクセスを取り消すことをお勧めします。使用可能な監査ログを確認して、削除されたオブジェクトの範囲を決定し、必要に応じてバックアップからの復旧を開始します。また、データベースをプライベート VPC に配置し、必要なソースからのトラフィックのみを許可するようにセキュリティグループのルールを制限することもお勧めします。詳細については、「異常な破壊的アクティビティによる侵害された可能性のあるデータベースの修復」を参照してください。

Impact:RDS/AnomalousBehavior.Delete

アカウントの RDS データベースからデータを削除するユーザーからの異常なアクティビティ。

デフォルトの重大度: [Low] (低

> **Note**
>
> この検出結果に関連する異常な動作に応じて、デフォルトの重要度は低、中、高のいずれかになります。
>
> - 低 – この検出結果に関連付けられたユーザー名が、プライベートネットワークに関連付けられた IP アドレスからDELETEステートメントを実行した場合。
> - Medium – この検出結果に関連付けられたユーザー名がパブリック IP アドレスからDELETEステートメントを実行した場合。
> - 高 – この検出結果に関連付けられたユーザー名が、異常に大量の行を返すDELETEステートメントを実行し、アクティビティが以前に見えないパブリックネットワークから発生したか、高特権データベースロールまたは管理クライアントによって実行された場合。

- 機能: RDS DB アクティビティモニタリング

この検出結果は、 AWS 環境内の RDS データベースで破壊的オペレーションと一致する異常なデータベースアクティビティが観察されたことを知らせるものです。これは、データベースユーザーが通常よりも大幅に多くの行を削除したこと、またはアクティビティが新しいユーザー、IP アドレス、アプリケーション名など、以前に見えなかったデータベース属性が破壊的なオペレーションに寄与し

たことを示している可能性があります。一般的なシナリオは、侵害されたデータベース認証情報を使用して、異常なアクセスパターンと増加したDELETEステートメントボリュームを組み合わせてデータを体系的に削除することです。

このアクティビティは、GuardDuty 異常検出機械学習 (ML) モデルによって異常として識別されました。ML モデルは、 内のすべてのデータベースアクティビティイベントを評価しサポートされているAmazon Aurora、Amazon RDS および Aurora Limitless データベース、攻撃者が使用する手法に関連する異常なイベントを特定します。ML モデルは、リクエストを行ったユーザー、実行されたステートメント、影響を受ける行数、使用された特定のデータベース接続の詳細など、RDS データベースアクティビティのさまざまな要因を追跡します。異常の可能性があるデータベースアクティビティの詳細については、「」を参照してくださいRDS データアクティビティベースの異常。

修復のレコメンデーション

関連付けられたデータベースでこのアクティビティが予期しない場合は、関連付けられたデータベースユーザーのパスワードを直ちに変更し、アクセスを取り消すことをお勧めします。利用可能な監査ログを確認して、削除されたデータの範囲を判断し、データ損失が確認された場合は自動バックアップからのpoint-in-timeリカバリを開始します。また、データベースをプライベート VPC に配置し、必要なソースからのトラフィックのみを許可するようにセキュリティグループのルールを制限することもお勧めします。詳細については、「異常な破壊的アクティビティによる侵害された可能性のあるデータベースの修復」を参照してください。

Exfiltration:RDS/MaliciousIPCaller

アカウントの RDS データベースで実行された既知の悪意のあるアクティビティに関連付けられた IP アドレス。

デフォルトの重大度: [High] (高

- 機能: RDS DB アクティビティモニタリング

この検出結果は、既知の悪意のあるアクティビティに関連付けられた IP アドレスが、 AWS 環境内の RDS データベースのデータ流出と一致するデータベースクエリを実行したことを知らせるものです。これは、アカウントの RDS データベースに関連付けられたユーザーとパスワードが侵害された可能性があり、RDS データベースデータが悪意のある可能性のあるアクターによって盗まれた可能性があることを示します。

修復のレコメンデーション

関連付けられたデータベースに対するこのアクティビティが想定外の場合は、ユーザーの認証情報が公開されているか、侵害されている可能性があります。関連付けられたデータベースユーザーのパスワードを変更し、侵害されたユーザーが実行したアクティビティについて利用可能な RDS データログを確認することをお勧めします。影響を受けたユーザーがアクセスできるデータを特定し、流出した可能性のあるデータのビジネスへの影響を評価することをお勧めします。また、データベースをプライベート VPC に配置し、必要なソースからのトラフィックのみを許可するようにセキュリティグループのルールを制限することもお勧めします。詳細については、「異常な流出アクティビティによる侵害された可能性のあるデータベースの修復」を参照してください。

Exfiltration:RDS/TorIPCaller

Tor 出口ノード IP アドレスは、アカウントの RDS データベースでデータ抽出クエリを実行します。

デフォルトの重大度: [High] (高

- 機能: RDS DB アクティビティモニタリング

この検出結果は、Tor 出口ノード IP アドレスが AWS 環境内の RDS データベースのデータ流出と一致するデータベースクエリを実行したことを知らせるものです。Tor は、匿名通信を有効にするためのソフトウェアです。一連のネットワークノード間のリレーを介して通信を暗号化し、ランダムにバウンスします。最後の Tor ノードは終了ノードと呼ばれます。これは、匿名ユーザーの真のアイデンティティを隠してデータを抽出する意図で、アカウント内の RDS リソースへの不正アクセスを示している可能性があります。

修復のレコメンデーション

関連付けられたデータベースに対するこのアクティビティが想定外の場合は、ユーザーの認証情報が公開されているか、侵害されている可能性があります。関連付けられたデータベースユーザーのパスワードを変更し、侵害されたユーザーが実行したアクティビティについて利用可能な RDS データログを確認することをお勧めします。影響を受けたユーザーがアクセスできるデータを特定し、流出した可能性のあるデータのビジネスへの影響を評価することをお勧めします。また、データベースをプライベート VPC に配置し、必要なソースからのトラフィックのみを許可するようにセキュリティグループのルールを制限することもお勧めします。詳細については、「異常な流出アクティビティによる侵害された可能性のあるデータベースの修復」を参照してください。

Impact:RDS/MaliciousIPCaller

アカウントの RDS データベースで実行された既知の悪意のあるアクティビティに関連付けられた IP アドレス。

デフォルトの重大度: [High] (高

- 機能: RDS DB アクティビティモニタリング

この検出結果は、既知の悪意のあるアクティビティに関連付けられた IP アドレスが、 AWS 環境内の RDS データベースに破壊的な影響を与えるデータベースステートメントを実行したことを知らせるものです。これは、アカウントの RDS データベースに関連付けられたユーザーとパスワードが侵害された可能性があり、RDS データベースが悪意のある可能性のあるアクターによって破壊的な操作を受けている可能性があることを示します。

修復のレコメンデーション

関連付けられたデータベースに対するこのアクティビティが想定外の場合は、ユーザーの認証情報が公開されているか、侵害されている可能性があります。関連付けられたデータベースユーザーのパスワードをすぐに変更し、アクセスを取り消すことをお勧めします。使用可能な RDS データログを確認して、実行された破壊的なオペレーションの範囲を決定し、必要に応じてバックアップからの復旧を開始します。また、データベースをプライベート VPC に配置し、必要なソースからのトラフィックのみを許可するようにセキュリティグループのルールを制限することもお勧めします。詳細については、「異常な破壊的アクティビティによる侵害された可能性のあるデータベースの修復」を参照してください。

Impact:RDS/TorIPCaller

Tor 出口ノード IP アドレスは、アカウントの RDS データベースで破壊的なステートメントを実行しました。

デフォルトの重大度: [High] (高

- 機能: RDS DB アクティビティモニタリング

この検出結果は、Tor 出口ノードの IP アドレスが AWS 、環境内の RDS データベースに破壊的な影響を与えるデータベースステートメントを実行したことを知らせるものです。Tor は、匿名通信を有効にするオーバーレイネットワークです。一連のネットワークノード間のリレーを介して通信を暗号

化し、ランダムにバウンスします。最後の Tor ノードは終了ノードと呼ばれます。これは、破壊的なオペレーションの実行中に匿名ユーザーの真のアイデンティティを隠す目的で、アカウント内の RDS リソースへの不正アクセスを示している可能性があります。

修復のレコメンデーション

関連付けられたデータベースに対するこのアクティビティが想定外の場合は、ユーザーの認証情報が公開されているか、侵害されている可能性があります。関連付けられたデータベースユーザーのパスワードをすぐに変更し、アクセスを取り消すことをお勧めします。使用可能な RDS データログを確認して、実行された破壊的なオペレーションの範囲を決定し、必要に応じてバックアップからの復旧を開始します。また、データベースをプライベート VPC に配置し、必要なソースからのトラフィックのみを許可するようにセキュリティグループのルールを制限することもお勧めします。詳細については、「異常な破壊的アクティビティによる侵害された可能性のあるデータベースの修復」を参照してください。

Lambda Protection の検出結果タイプ

このセクションでは、 AWS Lambda リソースに固有で、 が としてresourceTypeリストされている検出結果タイプについて説明しますLambda。すべての Lambda の検出結果について、問題のリソースを調べて正常に動作しているかどうかを確認することをお勧めします。アクティビティが認可されると、そのリソースに対する誤検出の通知を防ぐため、抑制ルールや信頼できる IP および脅威リストを使用できます。

アクティビティが予想されていなかった場合、セキュリティのベストプラクティスは、Lambda が侵害されている可能性があると想定し、修復のレコメンデーションに従うことです。

トピック

- Backdoor:Lambda/C&CActivity.B
- CryptoCurrency:Lambda/BitcoinTool.B
- Trojan:Lambda/BlackholeTraffic
- Trojan:Lambda/DropPoint
- UnauthorizedAccess:Lambda/MaliciousIPCaller.Custom
- UnauthorizedAccess:Lambda/TorClient
- UnauthorizedAccess:Lambda/TorRelay

Backdoor:Lambda/C&CActivity.B

Lambda 関数は、既知のコマンドとコントロールサーバーに関連付けられる IP アドレスをクエリしています。

デフォルトの重大度: [High] (高

- 機能: Lambda Network Activity Monitoring

この検出結果は、 AWS 環境内のリストされた Lambda 関数が、既知のコマンドアンドコントロール (C&C) サーバーに関連付けられている IP アドレスをクエリしていることを知らせるものです。生成された検出結果に関連する Lambda 関数が危険にさらされている可能性があります。C&C サーバーは、ボットネットのメンバーにコマンドを発行するコンピュータです。

ボットネットとは、一般的なタイプのマルウェアに感染しコントロールされたインターネットコネクテッドデバイス (PC、サーバー、モバイルデバイス、IoT デバイスなど) のコレクションです。ボットネットは、マルウェアを配布し、クレジットカード番号などの悪用された情報を収集するためによく使用されます。ボットネットの目的と構造によっては、C&C サーバーから分散型サービス拒否を開始するためのコマンドが発行されることもあります。

修復のレコメンデーション

このアクティビティが予期しないものである場合、Lambda 関数は侵害されている可能性があります。詳細については、「侵害された可能性のある Lambda 関数の修復」を参照してください。

CryptoCurrency:Lambda/BitcoinTool.B

Lambda 関数が暗号通貨関連のアクティビティに関連付けられている IP アドレスをクエリしています。

デフォルトの重大度: [High] (高

- 機能: Lambda Network Activity Monitoring

この検出結果は、 AWS 環境内のリストされた Lambda 関数が Bitcoin または他の暗号通貨関連のアクティビティに関連付けられている IP アドレスをクエリしていることを知らせるものです。脅威アクターは、Lambda 関数を不正な暗号通貨マイニングに不正に転用するために、Lambda 関数を乗っ取ろうとしている可能性があります。

修復のレコメンデーション

この Lambda 関数を使用して暗号通貨のマイニングまたは管理を行う場合、またはこの関数がブロックチェーンアクティビティに関与している場合は、環境で予期されるアクティビティである可能性があります。 AWS 環境でこのような場合は、この検出結果の抑制ルールを設定することをお勧めします。抑制ルールは、2 つのフィルター条件で構成する必要があります。1 つ目の検索条件では、[検出結果タイプ] 属性に CryptoCurrency:Lambda/BitcoinTool.B という値を使用します。2 つ目のフィルター条件は、ブロックチェーンアクティビティに関係する関数の Lambda 関数名である必要があります。抑制ルール作成の詳細については、「抑制ルール」を参照してください。

このアクティビティが予想されていなかった場合、Lambda 関数が侵害されている可能性があります。詳細については、「侵害された可能性のある Lambda 関数の修復」を参照してください。

Trojan:Lambda/BlackholeTraffic

Lambda 関数は、ブラックホールと呼ばれるリモートホストの IP アドレスに通信しようとしています。

デフォルトの重大度: [Medium] (中

- 機能: Lambda Network Activity Monitoring

この検出結果は、 AWS 環境内のリストされた Lambda 関数がブラックホール (またはシンクホール) の IP アドレスと通信しようとしていることを知らせるものです。ブラックホールとは、送受信トラフィックが密かに破棄されるネットワークの場所を指し、意図した受信者にデータが届いていないことは送信元に知らされません。ブラックホール IP アドレスは、実行されていないホストマシン、またはホストが割り当てられていないアドレスを指定します。記載されている Lambda 関数は侵害されている可能性があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合、Lambda 関数は侵害されている可能性があります。詳細については、「侵害された可能性のある Lambda 関数の修復」を参照してください。

Trojan:Lambda/DropPoint

Lambda 関数が、マルウェアによって収集された認証情報やその他の盗難されたデータを保持していることが認識されているリモートホストの IP アドレスに通信しようとしています。

デフォルトの重大度: [Medium] (中

- 機能: Lambda Network Activity Monitoring

この検出結果は、 AWS 環境内のリストされた Lambda 関数が、マルウェアによってキャプチャされた認証情報やその他の盗まれたデータを保持することが知られているリモートホストの IP アドレスと通信しようとしていることを知らせるものです。

修復のレコメンデーション

このアクティビティが予期しないものである場合、Lambda 関数は侵害されている可能性があります。詳細については、「侵害された可能性のある Lambda 関数の修復」を参照してください。

UnauthorizedAccess:Lambda/MaliciousIPCaller.Custom

Lambda 関数がカスタム脅威リストの IP アドレスに接続しています。

デフォルトの重大度: [Medium] (中

- 機能: Lambda Network Activity Monitoring

この検出結果は、 AWS 環境の Lambda 関数が、アップロードした脅威リストに含まれている IP アドレスと通信していることを知らせるものです。GuardDuty の場合、脅威リストは悪意のある既知の IP アドレスで構成されます。GuardDuty では、アップロード済みの脅威リストに基づいて検出結果を生成します。脅威リストの詳細は、GuardDuty コンソールの検出結果の詳細で確認できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、Lambda 関数は侵害されている可能性があります。詳細については、「侵害された可能性のある Lambda 関数の修復」を参照してください。

UnauthorizedAccess:Lambda/TorClient

Lambda 関数は Tor Guard または Authority ノードに接続しています。

デフォルトの重大度: [High] (高

- 機能: Lambda Network Activity Monitoring

この検出結果は、 AWS 環境の Lambda 関数が Tor Guard または Authority ノードに接続していることを知らせるものです。Tor は、匿名通信を有効にするためのソフトウェアです。Tor Guards および Authority ノードは、Tor ネットワークへの初期ゲートウェイとして動作します。このトラフィックは、この Lambda 関数が侵害されている可能性があることを示している可能性があります。現在は、Tor ネットワーク上のクライアントとして動作している場合があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合、Lambda 関数は侵害されている可能性があります。詳細については、「侵害された可能性のある Lambda 関数の修復」を参照してください。

UnauthorizedAccess:Lambda/TorRelay

Lambda 関数は、Tor リレーとして Tor ネットワークに接続中です。

デフォルトの重大度: [High] (高

- 機能: Lambda Network Activity Monitoring

この検出結果は、 AWS 環境の Lambda 関数が Tor リレーとして動作していることを示す方法で Tor ネットワークに接続していることを知らせるものです。Tor は、匿名通信を有効にするためのソフトウェアです。Tor は、クライアントの不正な可能性のあるトラフィックをある Tor リレーから別の Tor リレーに転送することにより、通信の匿名性を高めます。

修復のレコメンデーション

このアクティビティが予期しないものである場合、Lambda 関数は侵害されている可能性があります。詳細については、「侵害された可能性のある Lambda 関数の修復」を参照してください。

GuardDuty AI Protection の検出結果タイプ

AI Protection を有効にすると、GuardDuty は Amazon Bedrock と Amazon SageMaker AI AWS CloudTrail の管理イベントとデータイベントを分析します。GuardDuty はこのデータを使用して、AI ワークロードをターゲットとする疑わしいアクティビティを検出します。GuardDuty は潜在的な脅威を検出すると、次のいずれかの検出結果タイプを生成します。これらの検出結果`Resource type`の は でありAccessKey、検出結果はモデルを呼び出した IAM ID を識別します。

各検出結果には、検出結果の `resource` オブジェクト内の影響を受ける AI リソースに関する詳細も含まれます。Impact:IAMUser/AnomalousModelInvocation および Impact:IAMUser/CostHarvestingの結果には、呼び出された Amazon Bedrock または Amazon SageMaker AI モデルを識別する`resource.modelDetails`リストが含まれます。Impact:IAMUser/PromptInjection.Direct 検出結果には、介入した Amazon Bedrock ガードレールを記述する`resource.bedrockGuardrailDetails`オブジェクトが含まれます。これらの詳細は、GuardDuty コンソールまたは検出結果 JSON で表示できます。

Impact:IAMUser/AnomalousModelInvocation および Impact:IAMUser/CostHarvestingの検出結果タイプは、Amazon Bedrock と Amazon SageMaker AI モデルの呼び出しの両方に適用されます。Impact:IAMUser/PromptInjection.Direct 検出結果タイプは、Amazon Bedrock ガードレールを使用する Amazon Bedrock ワークロードに適用されます。

トピック

- Impact:IAMUser/AnomalousModelInvocation
- Impact:IAMUser/CostHarvesting
- Impact:IAMUser/PromptInjection.Direct

Impact:IAMUser/AnomalousModelInvocation

IAM アイデンティティが、アイデンティティまたはアカウントの確立されたベースラインから逸脱する方法で Amazon Bedrock または Amazon SageMaker AI モデルを呼び出しました。

デフォルトの重大度: [Low] (低

- 機能: AI Protection

この検出結果は、IAM アイデンティティが、アイデンティティまたはアカウントの過去のアクティビティから逸脱する方法で Amazon Bedrock または Amazon SageMaker AI モデルを呼び出したことを知らせるものです。このアクティビティは、脅威アクターが侵害された認証情報を使用してアカウントの基盤モデルにアクセスしていることを示している可能性があります。たとえば、アクターはモデルを調査したり、敵対的な入力を作成したり、自動リクエストを実行したりできます。

GuardDuty は、IAM アイデンティティごとに通常のモデル呼び出しアクティビティのベースラインを確立し、Amazon Bedrock または Amazon SageMaker AI AWS アカウント を呼び出します。GuardDuty は、以下を含む呼び出しアクティビティの特徴をプロファイリングします。

- ID が呼び出したモデル呼び出し API
- ID が呼び出したモデル
- ソース IP アドレスとそれに関連する自律システム番号 (ASN) 組織
- リクエストを行ったユーザーエージェント

GuardDuty は、このベースラインから大幅に逸脱するアクティビティを確認した場合に、この検出結果を生成します。例としては、以前には見えなかった IP アドレス、以前には見えなかったユーザーエージェント、またはアイデンティティが呼び出されたことのないモデルからの呼び出しなどがあります。検出結果には、疑わしいアクティビティに関連するモデルを識別する`resource.modelDetails`リストが含まれています。各エントリには、Amazon Bedrock 基盤モデル、推論プロファイル、プロビジョニング済み、カスタム、またはインポートされたモデル ARN、Amazon Bedrock Marketplace エンドポイント、プロンプトリソース、または Amazon SageMaker AI エンドポイント名`modelId`を表す が含まれます。

この検出結果は、MITRE ATLAS ウェブサイトの MITRE ATLAS 手法 AML.T0040 - AI Model Inference API Access にマッピングされます。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

不正なモデル呼び出しのリスクを軽減するには、モデルを呼び出すことができる ID を制限します。次のモデル呼び出し APIs へのアクセスを、それらを必要とするプリンシパルとモデルのみにスコープします。

- `bedrock:InvokeModel`
- `bedrock:InvokeModelWithResponseStream`

- `bedrock:Converse`
- `bedrock:ConverseStream`
- `bedrock-mantle:CreateInference`
- `sagemaker:InvokeEndpoint`
- `sagemaker:InvokeEndpointAsync`
- `sagemaker:InvokeEndpointWithResponseStream`

不要な組織全体でモデル推論を拒否するには、サービスコントロールポリシー (SCP) を使用できます。詳細については、「Amazon Bedrock ユーザーガイド」の「Amazon Bedrock のアイデンティティベースのポリシーの例」および「Amazon SageMaker AI デベロッパーガイド」の「Amazon SageMaker AI アイデンティティベースのポリシーの例」を参照してください。 Amazon SageMaker

Impact:IAMUser/CostHarvesting

IAM アイデンティティが、巨額の従量課金を発生させる攻撃を示す可能性のある異常なトークンボリュームで Amazon Bedrock または Amazon SageMaker AI モデルを呼び出しました。

デフォルトの重大度: [Low] (低

- 機能: AI Protection

この検出結果は、IAM アイデンティティが、アイデンティティまたはアカウントのベースラインから大幅に逸脱する入出力トークンボリュームで Amazon Bedrock または Amazon SageMaker AI モデルを呼び出したことを知らせるものです。巨額の従量課金を発生させる攻撃では、脅威アクターは計算コストの高い入力をモデルに送信して、必ずしもデータを流出させることなく、AI ワークロードの運用コストを暴騰させます。これらの入力には、過度に長いプロンプトや意図的に不正な形式のプロンプトが含まれる場合があります。

GuardDuty は、Amazon Bedrock または Amazon SageMaker AI モデルを呼び出す IAM アイデンティティとアカウントごとに、平均入力トークンボリュームと平均出力トークンボリュームのベースラインを確立します。GuardDuty がこのベースラインを超えるトークン数を持つ呼び出しを観察し、ボリュームの異常を他の異常なシグナルと関連付けると、GuardDuty はこの検出結果を生成します。検出結果は、オブジェクト内の異常なアクティビティを報告します。これには`service.detection`、観測された呼び出しと、ベースラインから逸脱したトークンボリューム

が含まれます。また、ターゲットモデルを識別するresource.modelDetailsリストも含まれています。各エントリには、Amazon Bedrock 基盤モデル、推論プロファイル、プロビジョニング済み、カスタム、またはインポートされたモデル ARN、Amazon Bedrock Marketplace エンドポイント、プロンプトリソース、または Amazon SageMaker AI エンドポイント名modelIdを表す が含まれます。

この検出結果は、MITRE ATLAS ウェブサイトの MITRE ATLAS 手法 AML.T0034 - Cost Harvesting にマッピングされます。

修復のレコメンデーション

このアクティビティが関連する ID に対して予期しないものである場合、認証情報が侵害され、AI ワークロードのコストを引き上げるために使用された可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。ID に関連付けられたモデル呼び出しを確認し、次のモデル呼び出し APIs へのアクセスを、それらを必要とするプリンシパルのみにスコープすることで、影響を受けるモデルへのアクセスを制限します。

- bedrock:InvokeModel
- bedrock:InvokeModelWithResponseStream
- bedrock:Converse
- bedrock:ConverseStream
- bedrock-mantle:CreateInference
- sagemaker:InvokeEndpoint
- sagemaker:InvokeEndpointAsync
- sagemaker:InvokeEndpointWithResponseStream

異常な支出を検出して封じ込めるには、予算しきい値を超えたときに制限付き IAM ポリシーを適用する予算アクションやAWS コスト異常検出など、 AWS Budgets を使用できます。Amazon Bedrock のコストを特定のワークロードに帰属させるには、コスト配分タグをアプリケーション推論プロファイルにアタッチします。

Impact:IAMUser/PromptInjection.Direct

Amazon Bedrock Guardrail は、IAM アイデンティティによって行われた呼び出しで直接プロンプトインジェクションの試みを検出しました。

デフォルトの重大度: [Low] (低

- 機能: AI Protection

この検出結果は、アカウントの Amazon Bedrock ワークロード用に設定された Amazon Bedrock Guardrail が直接的なプロンプトインジェクションの試みを検出したことを知らせるものです。直接的なプロンプトインジェクション攻撃では、脅威アクターが基盤モデルへの入力として悪意のあるプロンプトを作成します。プロンプトでは、モデルは元の指示を無視し、代わりに攻撃者の指示に従います。プロンプトインジェクションは、攻撃者に、データの流出、不正なツールの呼び出し、安全コントロールのバイパスなど、AI アプリケーションに対して追加のステップを実行するための初期アクセス権を付与できます。

GuardDuty は、Amazon Bedrock Guardrail が高い信頼度でプロンプト攻撃を検出したために呼び出しに介入したときに、この検出結果を生成します。GuardDuty がプロンプトの直接挿入を検出できるようにするには、プロンプト攻撃コンテンツフィルターを使用して Amazon Bedrock ガードレールを設定します。ガードレールが環境全体の呼び出しを評価するようにするには、個々のアプリケーションに依存して各リクエストにアタッチするのではなく、広く適用します。Amazon Bedrock ポリシーを使用して、アカウント内または組織全体のすべての Amazon Bedrock モデル呼び出しにガードレールを適用できます AWS Organizations。詳細については、「Amazon Bedrock ユーザーガイド」の「Amazon Bedrock ガードレールによるクロスアカウント保護の適用」を参照してください。ガードレールが呼び出しを評価すると、Amazon Bedrock は評価を AWS CloudTrail データイベントに記録し、GuardDuty はこの結果を分析して生成します。ガードレールが呼び出しの評価時に機密情報フィルターも適用した場合、検出結果には検出された機密情報のタイプに関する詳細が含まれます。

検出結果には、介入したガードレールについて記述する `resource.bedrockGuardrailDetails` オブジェクトが含まれます。次の表に、このオブジェクトのフィールドを示します。

フィールド	説明
`guardrails`	呼び出しを評価したガードレールのリスト。各エントリには、Amazon リソースネーム (arn) とガードレールversionの が含まれます。アカウント強制ガードレールと組織強制ガードレールの両方が同じ呼び出しを評価すると、複数のガードレールが表示されることがあります。
`guardrailAction`	ガードレールが実行したアクション。GuardDuty は、値が の場合にこの結果を生成しますGUARDRAIL_INTERVENED 。

フィールド	説明
guardrailSource	ガードレールが評価したコンテンツ。 INPUTはモデルに送信されたプロンプトを示し、 はモデルからのレスポンスOUTPUTを示します。
contentPolicyFilters	ガードレールが適用したコンテンツポリシーフィルターのリスト。この検出結果タイプでは、エントリの は type PROMPT_ATTACK 、は confidence ですHIGH。は、ガードレールBLOCKEDがコンテンツをブロックNONEした場合、またはガードレールがプロンプト攻撃を検出したが、それを報告するようにのみ設定されている場合、action です。
sensitiveInformationFilters	機密情報は、呼び出しを評価したときにガードレールが適用したフィルターをフィルタリングします。各エントリには、 AWS_SECRET_KEY や など、ガードレールが検出typeした機密情報の種類を識別する とPASSWORD、ガードレールがコンテンツをブロックBLOCKEDした場合、ガードレールがコンテンツをマスクANONYMIZED した場合、またはガードレールが検出したがレポートするようにのみ設定NONEした場合actionの があります。検出結果には、検出された情報のタイプが記録されます。コンテンツ自体は記録されません。

この検出結果は、MITRE ATLAS ウェブサイトの MITRE ATLAS 手法 AML.T0051 - LLM Prompt Injection にマッピングされます。

修復のレコメンデーション

ガードレールの介入をトリガーしたプロンプトと、プロンプトを送信した IAM ID を確認します。関連付けられた ID のアクティビティが予期しない場合、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。ガードレールがプロンプト攻撃を検出したが、ブロックしなかった場合は、ガードレールがコンテンツをブロックするように、プロンプト攻撃コンテンツフィルターのアクションを Detect ではなく Block に設定することを検討してください。詳細については、「Amazon Bedrock ユーザーガイド」の「Amazon Bedrock ガードレールによって検出された有害なコンテンツを処理するためのオプション」を参照してください。 影響を受ける ID の Amazon Bedrock 呼び出し APIs へのアクセスを制限することもできます。

廃止された検出結果タイプ

検出結果は、GuardDuty で検出した潜在的なセキュリティ問題に関する詳細を含む通知です。新しく追加されたタイプや廃止されたタイプを含む、GuardDuty の検出結果タイプの重要な変更点については、「Amazon GuardDuty のドキュメント履歴」を参照してください。

GuardDuty により生成された、次の検出結果タイプは廃止されています (今後生成されません)。

> **Important**
>
> GuardDuty の廃止された検出結果タイプを再アクティブ化することはできません。

トピック

- PrivilegeEscalation:Kubernetes/PrivilegedContainer
- Persistence:Kubernetes/ContainerWithSensitiveMount
- Exfiltration:S3/ObjectRead.Unusual
- Impact:S3/PermissionsModification.Unusual
- Impact:S3/ObjectDelete.Unusual
- Discovery:S3/BucketEnumeration.Unusual
- Persistence:IAMUser/NetworkPermissions
- Persistence:IAMUser/ResourcePermissions
- Persistence:IAMUser/UserPermissions
- PrivilegeEscalation:IAMUser/AdministrativePermissions
- Recon:IAMUser/NetworkPermissions
- Recon:IAMUser/ResourcePermissions
- Recon:IAMUser/UserPermissions
- ResourceConsumption:IAMUser/ComputeResources
- Stealth:IAMUser/LoggingConfigurationModified
- UnauthorizedAccess:IAMUser/ConsoleLogin
- UnauthorizedAccess:EC2/TorIPCaller

- Backdoor:EC2/XORDDOS
- Behavior:IAMUser/InstanceLaunchUnusual
- CryptoCurrency:EC2/BitcoinTool.A
- UnauthorizedAccess:IAMUser/UnusualASNCaller

PrivilegeEscalation:Kubernetes/PrivilegedContainer

ルートレベルのアクセス権を持つ特権コンテナが Kubernetes クラスターで起動されました。

デフォルトの重大度: [Medium] (中

- 機能: EKS 監査ログ

この検出結果は、特権コンテナが、クラスター内の特権コンテナを起動するためにこれまで使用されたことのないイメージを使用して Kubernetes クラスターで起動されたことを知らせるものです。特権コンテナには、ホストへのルートレベルのアクセス権があります。攻撃者は、特権のエスカレーション戦術として特権コンテナを起動して、ホストにアクセスして侵害することができます。

修復のレコメンデーション

このコンテナの起動が想定されていない場合、コンテナの起動に使用されたユーザーアイデンティティの認証情報が侵害される可能性があります。ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

Persistence:Kubernetes/ContainerWithSensitiveMount

機密性の高い外部ホストパスが内部でマウントされた状態でコンテナが起動されました。

デフォルトの重大度: [Medium] (中

- 機能: EKS 監査ログ

この検出結果は、`volumeMounts` セクションに書き込みアクセス権を持つ機密ホストパスを含む設定でコンテナが起動されたことを知らせるものです。これにより、機密性の高いホストパスがコンテ

ナ内からアクセスおよび書き込み可能になります。この手法は、攻撃者がホストのファイルシステムにアクセスするために一般的に使用されます。

修復のレコメンデーション

このコンテナの起動が想定されていない場合、コンテナの起動に使用されたユーザーアイデンティティの認証情報が侵害される可能性があります。ユーザーのアクセスを取り消し、攻撃者がクラスターに加えた変更を元に戻します。詳細については、「EKS Protection の検出結果の修復」を参照してください。

このコンテナの起動が想定されている場合は、`resource.KubernetesDetails.KubernetesWorkloadDetails.containers.imagePrefix` フィールドに基づくフィルター基準で構成される抑制ルールを使用することをお勧めします。フィルター基準では、`imagePrefix` フィールドは検出結果で指定された `imagePrefix` と同じである必要があります。抑制ルールの作成の詳細については、「Suppression rules」(抑制ルール) を参照してください。

Exfiltration:S3/ObjectRead.Unusual

IAM エンティティが疑わしい方法で S3 API を呼び出しました。

デフォルトの重大度: [Medium] (中)*

> **Note**
>
> この検出結果のデフォルトの重大度は [Medium] (中) です。ただし、API がインスタンスで作成された一時的な AWS 認証情報を使用して呼び出された場合、検出結果の重要度は高になります。

- データソース: S3 CloudTrail データイベント

この検出結果は、 AWS 環境内の IAM エンティティが S3 バケットを含む API コールを実行し、そのエンティティの確立されたベースラインとは異なることを知らせるものです。このアクティビティで使用された API コールは、攻撃の抽出段階に関連付けられており、攻撃者はデータを収集しようとしています。IAM エンティティが API を呼び出した方法が異常だったため、このアクティビティは疑わしいものです。例えば、この IAM エンティティに以前このタイプの API を呼び出した履歴がない場合や、API が異常な場所から呼び出された場合などです。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

Impact:S3/PermissionsModification.Unusual

IAM エンティティが API を呼び出して、1 つ以上の S3 リソースの許可を変更しました。

デフォルトの重大度: [Medium] (中)*

> **Note**
>
> この検出結果のデフォルトの重大度は [Medium] (中) です。ただし、API がインスタンスで作成された一時的な AWS 認証情報を使用して呼び出された場合、検出結果の重要度は高になります。

この検出結果は、IAM エンティティが、 AWS 環境内の 1 つ以上のバケットまたはオブジェクトの許可を変更するように設計された API コールを行っていることを知らせるものです。このアクションは、攻撃者により、アカウント外で情報を共有できるよう実行された可能性があります。IAM エンティティが API を呼び出した方法が異常だったため、このアクティビティは疑わしいものです。例えば、この IAM エンティティに以前このタイプの API を呼び出した履歴がない場合や、API が異常な場所から呼び出された場合などです。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

Impact:S3/ObjectDelete.Unusual

IAM エンティティが S3 バケット内のデータを削除するために使用される API を呼び出しました。

デフォルトの重大度: [Medium] (中)*

> **Note**
>
> この検出結果のデフォルトの重大度は [Medium] (中) です。ただし、API がインスタンスで作成された一時的な AWS 認証情報を使用して呼び出された場合、検出結果の重要度は高になります。

この検出結果は、 AWS 環境内の特定の IAM エンティティが、バケット自体を削除して、リストされた S3 バケット内のデータを削除するように設計された API コールを行っていることを知らせるものです。IAM エンティティが API を呼び出した方法が異常だったため、このアクティビティは疑わしいものです。例えば、この IAM エンティティに以前このタイプの API を呼び出した履歴がない場合や、API が異常な場所から呼び出された場合などです。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

Discovery:S3/BucketEnumeration.Unusual

IAM エンティティが、ネットワーク内の S3 バケットを検出するために使用される S3 API を呼び出しました。

デフォルトの重大度: [Medium] (中)*

> **Note**
>
> この検出結果のデフォルトの重大度は [Medium] (中) です。ただし、API がインスタンスで作成された一時的な AWS 認証情報を使用して呼び出された場合、検出結果の重要度は高になります。

この検出結果は、IAM エンティティが S3 API を呼び出して、環境内の `ListBuckets` などの S3 バケットを検出したことを知らせるものです。このタイプのアクティビティは、攻撃者が情報を収集して AWS 、環境がより広範な攻撃の影響を受けやすいかどうかを判断する攻撃の検出段階に関連付けられています。IAM エンティティが API を呼び出した方法が異常だったため、このアクティビティは疑わしいものです。例えば、この IAM エンティティに以前このタイプの API を呼び出した履歴がない場合や、API が異常な場所から呼び出された場合などです。

修復のレコメンデーション

関連付けられたプリンシパルに対してこのアクティビティが予期せぬ場合は、認証情報が公開されているか、S3 許可が十分に制限されていないことを示している可能性があります。詳細については、「侵害された可能性のある S3 バケットの修復」を参照してください。

Persistence:IAMUser/NetworkPermissions

IAM エンティティが、 AWS アカウントのセキュリティグループ、ルート、および ACLs のネットワークアクセス許可を変更するのに一般的に使用される API を呼び出しました。

デフォルトの重大度: [Medium] (中)*

> **Note**
>
> この検出結果のデフォルトの重大度は [Medium] (中) です。ただし、API がインスタンスで作成された一時的な AWS 認証情報を使用して呼び出された場合、検出結果の重要度は高になります。

この検出結果は、 AWS 環境内の特定のプリンシパル (AWS アカウントのルートユーザー、IAM ロール、またはユーザー) が、確立されたベースラインとは異なる動作をしていることを示しています。このプリンシパルには、この API を呼び出した履歴はありません。

この検出結果は、ネットワーク構成設定が不審な状況下で変更された場合、例えば、プリンシパルが `CreateSecurityGroup` API を呼び出したが、それ以前に呼び出しの履歴がない場合などにトリガーされます。攻撃者はよく、セキュリティグループの変更を試み、さまざまなポートで特定のインバウンドトラフィックを許可させて彼らが EC2 インスタンスにアクセスする能力を向上させようとします。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

Persistence:IAMUser/ResourcePermissions

プリンシパルは、 のさまざまなリソースのセキュリティアクセスポリシーを変更するために一般的に使用される API を呼び出しました AWS アカウント。

デフォルトの重大度: [Medium] (中)*

> **Note**
>
> この検出結果のデフォルトの重大度は [Medium] (中) です。ただし、API が呼び出され、インスタンスで作成された一時的な AWS 認証情報が使用されている場合、検出結果の重要度は高になります。

この検出結果は、 AWS 環境内の特定のプリンシパル (AWS アカウントのルートユーザー、IAM ロール、またはユーザー) が、確立されたベースラインとは異なる動作をしていることを示しています。このプリンシパルには、この API を呼び出した履歴はありません。

この検出結果は、 AWS 環境内のプリンシパルが API を呼び出したが、それの履歴`PutBucketPolicy`がない場合など、 AWS リソースにアタッチされたポリシーまたはアクセス許可への変更が検出されたときにトリガーされます。Amazon S3 など一部のサービスでは、リソースに対するプリンシパルアクセスを 1 つ以上付与する、リソースにアタッチされた許可をサポートします。盗まれた認証情報が使用されると、攻撃者はリソースにアタッチされたポリシーを変更し、自身にそのリソースに対する今後のアクセスを付与できます。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

Persistence:IAMUser/UserPermissions

プリンシパルは、 AWS アカウントの IAM ユーザー、グループ、またはポリシーを追加、変更、または削除するために一般的に使用される API を呼び出しました。

デフォルトの重大度: [Medium] (中)*

> **Note**
>
> この検出結果のデフォルトの重大度は [Medium] (中) です。ただし、API がインスタンスで作成された一時的な AWS 認証情報を使用して呼び出された場合、検出結果の重要度は高になります。

この検出結果は、 AWS 環境内の特定のプリンシパル (AWS アカウントのルートユーザー、IAM ロール、またはユーザー) が、確立されたベースラインとは異なる動作をしていることを示しています。このプリンシパルには、この API を呼び出した履歴はありません。

この検出結果は、 AWS 環境内のプリンシパルが API を呼び出した履歴`AttachUserPolicy`がない場合など、環境内の AWS ユーザー関連のアクセス許可に対する不審な変更によってトリガーされます。攻撃者は、盗まれた認証情報を使用して、新規ユーザーの作成、既存ユーザーに対するアクセスポリシーの追加、またはアクセスキーの作成により、元のアクセスポイントが閉じられている場合であってもアカウントへのアクセスを最大化することがあります。例えば、アカウントの所有者が特定の IAM ユーザーまたはパスワードが盗まれたことに気づいてアカウントから削除したとします。ただし、不正に作成された管理者プリンシパルによって作成された他のユーザーを削除せず、その AWS アカウントを攻撃者がアクセスできるままにする場合があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

PrivilegeEscalation:IAMUser/AdministrativePermissions

プリンシパルがそれ自体に非常に寛容なポリシーを割り当てようとしました。

デフォルトの重大度: [Low] (低)*

> **Note**
>
> 特権のエスカレーションに失敗した場合のこの検出結果の重大度は [Low] (低) で、特権のエスカレーションに成功した場合は [Medium] (中) です。

この検出結果は、 AWS 環境内の特定の IAM エンティティが特権エスカレーション攻撃を示す可能性のある動作を示していることを示しています。この検出結果は、IAM ユーザーまたはロールが許

容度の高いポリシーを割り当てようとするとトリガーされます。問題となるユーザーまたはロールが管理者特権を持つことを意図しない場合は、ユーザーの認証情報が危険にさらされているか、ロールの許可が正しく設定されていない可能性があることを示します。

攻撃者は、盗まれた認証情報を使用して、新規ユーザーの作成、既存ユーザーに対するアクセスポリシーの追加、またはアクセスキーの作成により、元のアクセスポイントが閉じられている場合であってもアカウントへのアクセスを最大化します。アカウントの所有者が特定の IAM ユーザーのサインイン認証情報が盗まれたことに気づいてアカウントから削除したとしても、不正に作成した管理者プリンシパルによって作成された他のユーザーを削除せず、その AWS アカウントに攻撃者がアクセスすることが可能なままになっている場合があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

Recon:IAMUser/NetworkPermissions

プリンシパルは、 AWS アカウントのセキュリティグループ、ルート、および ACLs のネットワークアクセス許可を変更するのに一般的に使用される API を呼び出しました。

デフォルトの重大度: [Medium] (中)*

> **Note**
>
> この検出結果のデフォルトの重大度は [Medium] (中) です。ただし、API がインスタンスで作成された一時的な AWS 認証情報を使用して呼び出された場合、検出結果の重要度は高になります。

この検出結果は、 AWS 環境内の特定のプリンシパル (AWS アカウントのルートユーザー、IAM ロール、またはユーザー) が、確立されたベースラインとは異なる動作をしていることを示しています。このプリンシパルには、この API を呼び出した履歴はありません。

この検出結果は、 AWS アカウントのリソース許可が不審な状況下で調査された場合にトリガーされます。例えば、プリンシパルが `DescribeInstances` API を呼び出したが、それ以前に呼び出しの履歴がない場合です。攻撃者は、盗まれた認証情報を使用して、より貴重な認証情報を見つけたり、

既に持っている認証情報の機能を特定したりするために、 AWS リソースのこの種の偵察を実行する可能性があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

Recon:IAMUser/ResourcePermissions

プリンシパルは、 AWS アカウント内のさまざまなリソースのセキュリティアクセスポリシーを変更するために一般的に使用される API を呼び出しました。

デフォルトの重大度: [Medium] (中)*

> **Note**
>
> この検出結果のデフォルトの重大度は [Medium] (中) です。ただし、API がインスタンスで作成された一時的な AWS 認証情報を使用して呼び出された場合、検出結果の重要度は高になります。

この検出結果は、 AWS 環境内の特定のプリンシパル (AWS アカウントのルートユーザー、IAM ロール、またはユーザー) が、確立されたベースラインとは異なる動作をしていることを示しています。このプリンシパルには、この API を呼び出した履歴はありません。

この検出結果は、 AWS アカウントのリソース許可が不審な状況下で調査された場合にトリガーされます。例えば、プリンシパルが `DescribeInstances` API を呼び出したが、それ以前に呼び出しの履歴がない場合です。攻撃者は、盗まれた認証情報を使用して、より貴重な認証情報を見つけたり、既に持っている認証情報の機能を特定したりするために、 AWS リソースのこの種の偵察を実行する可能性があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

Recon:IAMUser/UserPermissions

プリンシパルは、 AWS アカウントの IAM ユーザー、グループ、またはポリシーを追加、変更、または削除するために一般的に使用される API を呼び出しました。

デフォルトの重大度: [Medium] (中)*

> **Note**
>
> この検出結果のデフォルトの重大度は [Medium] (中) です。ただし、API がインスタンスで作成された一時的な AWS 認証情報を使用して呼び出された場合、検出結果の重要度は高になります。

この検出結果は、 AWS 環境内のユーザーアクセス許可が疑わしい状況で調査されたときにトリガーされます。例えば、プリンシパル (AWS アカウントのルートユーザー、IAM ロール、または IAM ユーザー) が `ListInstanceProfilesForRole` API を呼び出したが、それ以前に呼び出しの履歴がない場合です。攻撃者は、盗まれた認証情報を使用して、より貴重な認証情報を見つけたり、既に持っている認証情報の機能を特定したりするために、 AWS リソースのこの種の偵察を実行する可能性があります。

この検出結果は、 AWS 環境内の特定のプリンシパルが、確立されたベースラインとは異なる動作をしていることを示しています。このプリンシパルには、この方法でこの API を呼び出した履歴はありません。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

ResourceConsumption:IAMUser/ComputeResources

プリンシパルが、EC2 インスタンスなどのコンピューティングリソースを起動するために一般的に使用される API を呼び出しました。

デフォルトの重大度: [Medium] (中)*

> **Note**
>
> この検出結果のデフォルトの重大度は [Medium] (中) です。ただし、API がインスタンスで作成された一時的な AWS 認証情報を使用して呼び出された場合、検出結果の重要度は高になります。

この検出結果は、 AWS 環境にリストされたアカウント内の EC2 インスタンスが不審な状況下で起動された場合にトリガーされます。この検出結果は、 AWS 環境内の特定のプリンシパルが、確立されたベースラインとは異なる動作をしていることを示しています。例えば、プリンシパル (AWS アカウントのルートユーザー、IAM ロール、または IAM ユーザー) が `RunInstances` API を呼び出したが、それの履歴がない場合などです。これは、攻撃者が盗まれた認証情報を使用してコンピューティング時間を盗んでいることを示している可能性があります (暗号通貨のマイニングやパスワードのクラッキングなど) 。また、攻撃者が AWS 環境内の EC2 インスタンスとその認証情報を使用してアカウントへのアクセスを維持する兆候である可能性もあります。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

Stealth:IAMUser/LoggingConfigurationModified

プリンシパルが、CloudTrail ログ記録の停止、既存のログの削除、アカウントのアクティビティのトレースの排除に一般的に使用される API を呼び出しました AWS 。

デフォルトの重大度: [Medium] (中)*

> **Note**
>
> この検出結果のデフォルトの重大度は [Medium] (中) です。ただし、API がインスタンスで作成された一時的な AWS 認証情報を使用して呼び出された場合、検出結果の重要度は高になります。

この検出結果は、環境内でリストされた AWS アカウントのログ記録設定が不審な状況下で変更された場合にトリガーされます。この検出結果は、 AWS 環境内の特定のプリンシパルが、確立された

ベースラインとは異なる動作をしていることを知らせるものです。たとえば、プリンシパル (AWS アカウントのルートユーザー、IAM ロール、または IAM ユーザー) が StopLogging API を呼び出したが、それの履歴がない場合などです。これは、攻撃者がアクティビティのトレースを削除してトラックをカバーしようとしていることを示している可能性があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

UnauthorizedAccess:IAMUser/ConsoleLogin

AWS アカウントのプリンシパルによる異常なコンソールログインが確認されました。

デフォルトの重大度: [Medium] (中)*

> **Note**
>
> この検出結果のデフォルトの重大度は [Medium] (中) です。ただし、API がインスタンスで作成された一時的な AWS 認証情報を使用して呼び出された場合、検出結果の重要度は高になります。

この検出結果は、コンソールへのログインが不審な状況下で検出された場合にトリガーされます。例えば、プリンシパルが、これまで行ったことのない ConsoleLogin API の呼び出しを、これまで使用したことのないクライアントまたは通常とは異なる場所から行った場合などです。これは、盗まれた認証情報が AWS アカウントへのアクセスに使用されていること、または無効または安全性の低い方法でアカウントにアクセスする有効なユーザー (承認された VPN 経由ではないなど) を示している可能性があります。

この検出結果は、 AWS 環境内の特定のプリンシパルが、確立されたベースラインとは異なる動作をしていることを知らせるものです。このプリンシパルには、この特定の場所からこのクライアントアプリケーションを使用したログインアクティビティの履歴はありません。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

UnauthorizedAccess:EC2/TorIPCaller

EC2 インスタンスが Tor 出口ノードからのインバウンド接続を受信しています。

デフォルトの重大度: [Medium] (中

この検出結果は、 AWS 環境の EC2 インスタンスが Tor 終了ノードからインバウンド接続を受信していることを知らせるものです。Tor は、匿名通信を有効にするためのソフトウェアです。通信を暗号化し、一連のネットワークノード間のリレー中にランダムに通信をバウンスさせます。最後の Tor ノードは終了ノードと呼ばれます。この検出結果は、攻撃者の真のアイデンティティを隠す目的で AWS 、リソースへの不正アクセスを示している可能性があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Backdoor:EC2/XORDDOS

EC2 インスタンスが通信しようとしている IP アドレスには XOR DDoS マルウェアが関連付けられています。

デフォルトの重大度: [High] (高

この検出結果は、 AWS 環境の EC2 インスタンスが XOR DDoS マルウェアに関連付けられた IP アドレスと通信しようとしていることを知らせるものです。この EC2 インスタンスは侵害されている可能性があります。XOR DDoS は、Linux システムをハイジャックするトロイの木馬マルウェアです。システムにアクセスするために、Linux 上の Secure Shell (SSH) サービスのパスワードを検出するためにブルートフォース攻撃を開始します。SSH 認証情報を取得してログインに成功すると、ルートユーザー権限を使用して、XOR DDoS をダウンロードしてインストールするスクリプトを実行します。その後、このマルウェアはボットネットの一部として、他のターゲットに対する分散型サービス拒否 (DDoS) 攻撃を開始します。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

Behavior:IAMUser/InstanceLaunchUnusual

ユーザーが起動した EC2 インスタンスのタイプが通常と異なります。

デフォルトの重大度: [High] (高

この検出結果は、 AWS 環境内の特定のユーザーが、確立されたベースラインとは異なる動作をしていることを知らせるものです。このユーザーには、このタイプの EC2 インスタンスを起動した履歴はありません。サインイン認証情報は侵害されている可能性があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

CryptoCurrency:EC2/BitcoinTool.A

EC2 インスタンスはビットコインマイニングプールと通信しています。

デフォルトの重大度: [High] (高

この検出結果は、 AWS 環境内の EC2 インスタンスが Bitcoin マイニングプールと通信していることを知らせるものです。暗号通貨マイニングの分野では、マイニングプールは、ネットワーク上で処理能力を共有し、ブロックの解決に貢献した作業の量に応じて報酬を分割するマイナーによるリソースのプールです。この EC2 インスタンスをビットコインマイニングに使用していない限り、EC2 インスタンスは侵害されている可能性があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合、インスタンスは侵害されている可能性があります。詳細については、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

UnauthorizedAccess:IAMUser/UnusualASNCaller

API が通常とは異なるネットワークの IP アドレスから呼び出されました。

デフォルトの重大度: [High] (高

この検出結果は、特定のアクティビティが通常とは異なるネットワークの IP アドレスから呼び出されたことを知らせるものです。これは、記述されたユーザーの AWS の使用履歴全体で一度も確認さ

れていないネットワークです。このアクティビティには、コンソールログイン、EC2 インスタンスの起動、新しい IAM ユーザーの作成、 AWS 権限の変更などが含まれます。これは、 AWS リソースへの不正アクセスを示している可能性があります。

修復のレコメンデーション

このアクティビティが予期しないものである場合は、認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。

影響を受ける可能性のあるリソース別の GuardDuty 検出結果タイプ

以下のページは、GuardDuty の検出結果に関連付けられた影響を受ける可能性のあるリソースタイプ別に分類されています。

- EC2 の検出結果タイプ
- IAM の検出結果タイプ
- 攻撃シーケンスの検出結果タイプ
- カスタム検出ルールの検出結果タイプ
- S3 Protection の検出結果タイプ
- EKS Protection の検出結果タイプ
- Runtime Monitoring の検出結果タイプ
- Malware Protection for EC2 の検出結果のタイプ
- Malware Protection for S3 の検出結果タイプ
- Malware Protection for Backup の検出結果タイプ
- RDS Protection の検出結果タイプ
- Lambda Protection の検出結果タイプ
- AI Protection の検出結果タイプ

GuardDuty のアクティブな検出結果タイプ

次の表は、すべてのアクティブな検出結果タイプをソース別にグループ化したものです。まず、基本的な脅威検出の結果タイプ (データソース別にソート) 、次にExtended Threat Detection攻撃シー

ケンスの検出結果タイプ、次に各保護プランの検出結果タイプ (計画別にアルファベット順) です。次の表では、結果の一部にアスタリスク (*) またはプラス記号 (+) でマークされた[検出結果の重大度]列値があります。

*これらの検出結果タイプは重大度が「可変」です。特定のタイプの検出結果は、検出結果に固有のコンテキストによって重大度が異なることがあります。検出結果タイプについて詳しくは、それぞれの詳細な説明を参照してください。

VPC フローログをデータソースとして使用する +EC2 の検出結果は、IPv6 トラフィックをサポートしていません。

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
CredentialAccess:IAMUser/AnomalousBehavior	IAM	CloudTrail 管理イベント	[Medium] (中)
DefenseEvasion:IAMUser/AnomalousBehavior	IAM	CloudTrail 管理イベント	[Medium] (中)
DefenseEvasion:IAMUser/BedrockLoggingDisabled	IAM	CloudTrail 管理イベント	[Medium] (中)
Discovery:IAMUser/AnomalousBehavior	IAM	CloudTrail 管理イベント	[Low] (低)
Exfiltration:IAMUser/AnomalousBehavior	IAM	CloudTrail 管理イベント	[High] (高)
Impact:IAMUser/AnomalousBehavior	IAM	CloudTrail 管理イベント	[High] (高)
InitialAccess:IAMUser/AnomalousBehavior	IAM	CloudTrail 管理イベント	[Medium] (中)

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
PenTest:IAMUser/KaliLinux	IAM	CloudTrail 管理イベント	[Medium] (中)
PenTest:IAMUser/ParrotLinux	IAM	CloudTrail 管理イベント	[Medium] (中)
PenTest:IAMUser/PentooLinux	IAM	CloudTrail 管理イベント	[Medium] (中)
Persistence:IAMUser/AnomalousBehavior	IAM	CloudTrail 管理イベント	[Medium] (中)
Stealth:IAMUser/PasswordPolicyChange	IAM	CloudTrail 管理イベント	低*
UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.InsideAWS	IAM	CloudTrail 管理イベント	高*
Policy:S3/AccountBlockPublicAccessDisabled	Amazon S3	CloudTrail 管理イベント	低
Policy:S3/BucketAnonymousAccessGranted	Amazon S3	CloudTrail 管理イベント	高
Policy:S3/BucketBlockPublicAccessDisabled	Amazon S3	CloudTrail 管理イベント	低
Policy:S3/BucketPublicAccessGranted	Amazon S3	CloudTrail 管理イベント	[High] (高)

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
PrivilegeEscalation:IAMUser/AnomalousBehavior	IAM	CloudTrail 管理イベント	[Medium] (中)
Recon:IAMUser/MaliciousIPCaller	IAM	CloudTrail 管理イベント	[Medium] (中)
Recon:IAMUser/MaliciousIPCaller.Custom	IAM	CloudTrail 管理イベント	[Medium] (中)
Recon:IAMUser/TorIPCaller	IAM	CloudTrail 管理イベント	[Medium] (中)
Stealth:IAMUser/CloudTrailLoggingDisabled	IAM	CloudTrail 管理イベント	低
Stealth:S3/ServerAccessLoggingDisabled	Amazon S3	CloudTrail 管理イベント	[Low] (低)
UnauthorizedAccess:IAMUser/ConsoleLoginSuccess.B	IAM	CloudTrail 管理イベント	[Medium] (中)
UnauthorizedAccess:IAMUser/MaliciousIPCaller	IAM	CloudTrail 管理イベント	[Medium] (中)
UnauthorizedAccess:IAMUser/MaliciousIPCaller.Custom	IAM	CloudTrail 管理イベント	[Medium] (中)
UnauthorizedAccess:IAMUser/TorIPCaller	IAM	CloudTrail 管理イベント	[Medium] (中)

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
CredentialAccess:IAMUser/CompromisedCredentials	IAM	CloudTrail 管理イベントまたは S3 CloudTrail データイベント	高
Policy:IAMUser/RootCredentialUsage	IAM	CloudTrail 管理イベントまたは S3 CloudTrail データイベント	低
Policy:IAMUser/ShortTermRootCredentialUsage	IAM	CloudTrail 管理イベントまたは S3 CloudTrail データイベント	低
UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.OutsideAWS	IAM	CloudTrail 管理イベントまたは S3 CloudTrail データイベント	高
UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.InsideAWS	IAM	CloudTrail 管理イベントまたは S3 CloudTrail データイベント	高
UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.OutsideAWS	IAM	CloudTrail 管理イベントまたは S3 CloudTrail データイベント	高
Impact:EC2/MaliciousDomainRequest.Custom	Amazon EC2	DNS ログ	中

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
Backdoor:EC2/C&CActivity.B!DNS	Amazon EC2	DNS ログ	高
CryptoCurrency:EC2/BitcoinTool.B!DNS	Amazon EC2	DNS ログ	高
Impact:EC2/AbusedDomainRequest.Reputation	Amazon EC2	DNS ログ	中
Impact:EC2/BitcoinDomainRequest.Reputation	Amazon EC2	DNS ログ	高
Impact:EC2/MaliciousDomainRequest.Reputation	Amazon EC2	DNS ログ	高
Impact:EC2/SuspiciousDomainRequest.Reputation	Amazon EC2	DNS ログ	低
Trojan:EC2/BlackholeTraffic!DNS	Amazon EC2	DNS ログ	中
Trojan:EC2/DGADomainRequest.B	Amazon EC2	DNS ログ	高
Trojan:EC2/DGADomainRequest.C!DNS	Amazon EC2	DNS ログ	高
Trojan:EC2/DNSDataExfiltration	Amazon EC2	DNS ログ	高

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
Trojan:EC2/DriveBySourceTraffic!DNS	Amazon EC2	DNS ログ	高
Trojan:EC2/DropPoint!DNS	Amazon EC2	DNS ログ	中
Trojan:EC2/PhishingDomainRequest!DNS	Amazon EC2	DNS ログ	高
UnauthorizedAccess:EC2/MetadataDNSRebind	Amazon EC2	DNS ログ	高
Backdoor:EC2/C&CActivity.B	Amazon EC2	VPC フローログ±	[High] (高)
Backdoor:EC2/DenialOfService.Dns	Amazon EC2	VPC フローログ±	[High] (高)
Backdoor:EC2/DenialOfService.Tcp	Amazon EC2	VPC フローログ±	[High] (高)
Backdoor:EC2/DenialOfService.Udp	Amazon EC2	VPC フローログ±	[High] (高)
Backdoor:EC2/DenialOfService.UdpOnTcpPorts	Amazon EC2	VPC フローログ±	[High] (高)
Backdoor:EC2/DenialOfService.UnusualProtocol	Amazon EC2	VPC フローログ±	[High] (高)
Backdoor:EC2/Spambot	Amazon EC2	VPC フローログ±	[Medium] (中)

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
Behavior:EC2/NetworkPortUnusual	Amazon EC2	VPC フローログ±	[Medium] (中)
Behavior:EC2/TrafficVolumeUnusual	Amazon EC2	VPC フローログ±	[Medium] (中)
CryptoCurrency:EC2/BitcoinTool.B	Amazon EC2	VPC フローログ±	[High] (高)
DefenseEvasion:EC2/UnusualDNSResolver	Amazon EC2	VPC フローログ±	[Medium] (中)
DefenseEvasion:EC2/UnusualDoHActivity	Amazon EC2	VPC フローログ±	[Medium] (中)
DefenseEvasion:EC2/UnusualDoTActivity	Amazon EC2	VPC フローログ±	[Medium] (中)
Impact:EC2/PortSweep	Amazon EC2	VPC フローログ±	[High] (高)
Impact:EC2/WinRMBruteForce	Amazon EC2	VPC フローログ±	低*
Recon:EC2/PortProbeEMRUnprotectedPort	Amazon EC2	VPC フローログ±	[High] (高)
Recon:EC2/PortProbeUnprotectedPort	Amazon EC2	VPC フローログ±	低*
Recon:EC2/Portscan	Amazon EC2	VPC フローログ±	[Medium] (中)
Trojan:EC2/BlackholeTraffic	Amazon EC2	VPC フローログ±	[Medium] (中)

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
Trojan:EC2/DropPoint	Amazon EC2	VPC フローログ±	[Medium] (中)
UnauthorizedAccess:EC2/MaliciousIPCaller.Custom	Amazon EC2	VPC フローログ±	[Medium] (中)
UnauthorizedAccess:EC2/RDPBruteForce	Amazon EC2	VPC フローログ±	低*
UnauthorizedAccess:EC2/SSHBruteForce	Amazon EC2	VPC フローログ±	低*
UnauthorizedAccess:EC2/TorClient	Amazon EC2	VPC フローログ±	[High] (高)
UnauthorizedAccess:EC2/TorRelay	Amazon EC2	VPC フローログ±	[High] (高)
AttackSequence:EKS/CompromisedCluster	攻撃シーケンスに関連するリソース	• EKS 監査ログイベント • Amazon EKS の Runtime Monitoring • Amazon EC2 の Amazon EKS マルウェア検出 • AWS CloudTrail S3 のデータイベント • AWS CloudTrail 管理イベント • VPC フローログ • Route53 Resolver DNS クエリログ	重大

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
AttackSequence:IAM/CompromisedCredentials	攻撃シーケンスに関連するリソース	CloudTrail 管理イベント	重大
AttackSequence:S3/CompromisedData	攻撃シーケンスに関連するリソース	CloudTrail 管理イベントおよび S3 CloudTrail データイベント	重大
AttackSequence:ECS/CompromisedCluster	攻撃シーケンスに関連するリソース	• AWS Amazon ECS ランタイム検出 (Fargate) • AWS Amazon ECS ランタイム検出 (Amazon EC2 インスタンス) • AWS マルウェア検出	重大
AttackSequence:EC2/CompromisedInstanceGroup	攻撃シーケンスに関連するリソース	• AWS Amazon EC2 ランタイム検出 • AWS マルウェア検出 • VPC フローログ • Route53 Resolver DNS クエリログ	重大
Impact:IAMUser/AnomalousModelInvocation	IAM	AI 保護	低
Impact:IAMUser/CostHarvesting	IAM	AI 保護	低

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
Impact:IAMUser/PromptInjection.Direct	IAM	AI 保護	[Low] (低)
CredentialAccess:Kubernetes/AnomalousBehavior.SecretsAccessed	Kubernetes	EKS 監査ログ	中
CredentialAccess:Kubernetes/MaliciousIPCaller	Kubernetes	EKS 監査ログ	高
CredentialAccess:Kubernetes/MaliciousIPCaller.Custom	Kubernetes	EKS 監査ログ	高
CredentialAccess:Kubernetes/SuccessfulAnonymousAccess	Kubernetes	EKS 監査ログ	高
CredentialAccess:Kubernetes/TorIPCaller	Kubernetes	EKS 監査ログ	高
DefenseEvasion:Kubernetes/MaliciousIPCaller	Kubernetes	EKS 監査ログ	高
DefenseEvasion:Kubernetes/MaliciousIPCaller.Custom	Kubernetes	EKS 監査ログ	高
DefenseEvasion:Kubernetes/SuccessfulAnonymousAccess	Kubernetes	EKS 監査ログ	高

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
DefenseEvasion:Kubernetes/TorIPCaller	Kubernetes	EKS 監査ログ	高
Discovery:Kubernetes/AnomalousBehavior.PermissionChecked	Kubernetes	EKS 監査ログ	[Low] (低)
Discovery:Kubernetes/MaliciousIPCaller	Kubernetes	EKS 監査ログ	中
Discovery:Kubernetes/MaliciousIPCaller.Custom	Kubernetes	EKS 監査ログ	中
Discovery:Kubernetes/SuccessfulAnonymousAccess	Kubernetes	EKS 監査ログ	中
Discovery:Kubernetes/TorIPCaller	Kubernetes	EKS 監査ログ	中
Execution:Kubernetes/ExecInKubeSystemPod	Kubernetes	EKS 監査ログ	中
Execution:Kubernetes/AnomalousBehavior.ExecInPod	Kubernetes	EKS 監査ログ	中
Execution:Kubernetes/AnomalousBehavior.WorkloadDeployed	Kubernetes	EKS 監査ログ	[Low] (低)
Impact:Kubernetes/MaliciousIPCaller	Kubernetes	EKS 監査ログ	高

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
Impact:Kubernetes/MaliciousIPCaller.Custom	Kubernetes	EKS 監査ログ	高
Impact:Kubernetes/SuccessfulAnonymousAccess	Kubernetes	EKS 監査ログ	高
Impact:Kubernetes/TorIPCaller	Kubernetes	EKS 監査ログ	高
Persistence:Kubernetes/MaliciousIPCaller	Kubernetes	EKS 監査ログ	中
Persistence:Kubernetes/MaliciousIPCaller.Custom	Kubernetes	EKS 監査ログ	中
Persistence:Kubernetes/SuccessfulAnonymousAccess	Kubernetes	EKS 監査ログ	高
Persistence:Kubernetes/TorIPCaller	Kubernetes	EKS 監査ログ	中
Policy:Kubernetes/AdminAccessToDefaultServiceAccount	Kubernetes	EKS 監査ログ	高
Policy:Kubernetes/AnonymousAccessGranted	Kubernetes	EKS 監査ログ	高
Policy:Kubernetes/KubeflowDashboardExposed	Kubernetes	EKS 監査ログ	中

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
Policy:Kubernetes/ExposedDashboard	Kubernetes	EKS 監査ログ	中
PrivilegeEscalation:Kubernetes/AnomalousBehavior.RoleBindingCreated	Kubernetes	EKS 監査ログ	中*
PrivilegeEscalation:Kubernetes/AnomalousBehavior.RoleCreated	Kubernetes	EKS 監査ログ	[Low] (低)
Persistence:Kubernetes/AnomalousBehavior.WorkloadDeployed!ContainerWithSensitiveMount	Kubernetes	EKS 監査ログ	高
PrivilegeEscalation:Kubernetes/AnomalousBehavior.WorkloadDeployed!PrivilegedContainer	Kubernetes	EKS 監査ログ	高
Backdoor:Lambda/C&CActivity.B	Lambda	Lambda Network Activity Monitoring	高
CryptoCurrency:Lambda/BitcoinTool.B	Lambda	Lambda Network Activity Monitoring	高
Trojan:Lambda/BlackholeTraffic	Lambda	Lambda Network Activity Monitoring	中
Trojan:Lambda/DropPoint	Lambda	Lambda Network Activity Monitoring	中

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
UnauthorizedAccess:Lambda/MaliciousIPCaller.Custom	Lambda	Lambda Network Activity Monitoring	中
UnauthorizedAccess:Lambda/TorClient	Lambda	Lambda Network Activity Monitoring	高
UnauthorizedAccess:Lambda/TorRelay	Lambda	Lambda Network Activity Monitoring	高
Execution:Container/MaliciousFile	コンテナ	EBS Malware Protection	検出された脅威によって異なります
Execution:Container/SuspiciousFile	コンテナ	EBS Malware Protection	検出された脅威によって異なります
Execution:EC2/MaliciousFile	Amazon EC2	EBS Malware Protection	検出された脅威によって異なります
Execution:EC2/SuspiciousFile	Amazon EC2	EBS Malware Protection	検出された脅威によって異なります
Execution:ECS/MaliciousFile	ECS	EBS Malware Protection	検出された脅威によって異なります
Execution:ECS/SuspiciousFile	ECS	EBS Malware Protection	検出された脅威によって異なります
Execution:Kubernetes/MaliciousFile	Kubernetes	EBS Malware Protection	検出された脅威によって異なります
Execution:Kubernetes/SuspiciousFile	Kubernetes	EBS Malware Protection	検出された脅威によって異なります
Execution:EC2/MaliciousFile!Snapshot	Amazon EBS	Malware Protection for Backup	検出された脅威によって異なります

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
Execution:EC2/MaliciousFile!AMI	Amazon EC2	Malware Protection for Backup	検出された脅威によって異なります
Execution:EC2/MaliciousFile!RecoveryPoint	AWS バックアップ	Malware Protection for Backup	検出された脅威によって異なります
Execution:S3/MaliciousFile!RecoveryPoint	AWS バックアップ	Malware Protection for Backup	検出された脅威によって異なります
Object:S3/MaliciousFile	S3Object	S3 向けのマルウェア防御	高
CredentialAccess:RDS/AnomalousBehavior.FailedLogin	サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース	RDS ログインアクティビティのモニタリング	低
CredentialAccess:RDS/AnomalousBehavior.SuccessfulBruteForce	サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース	RDS ログインアクティビティのモニタリング	高
CredentialAccess:RDS/AnomalousBehavior.SuccessfulLogin	サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース	RDS ログインアクティビティのモニタリング	変数*

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
CredentialAccess:RDS/MaliciousIPCaller.FailedLogin	サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース	RDS ログインアクティビティのモニタリング	中
CredentialAccess:RDS/MaliciousIPCaller.SuccessfulLogin	サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース	RDS ログインアクティビティのモニタリング	高
CredentialAccess:RDS/TorIPCaller.FailedLogin	サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース	RDS ログインアクティビティのモニタリング	中
CredentialAccess:RDS/TorIPCaller.SuccessfulLogin	サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース	RDS ログインアクティビティのモニタリング	高
Discovery:RDS/MaliciousIPCaller	サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース	RDS ログインアクティビティのモニタリング	中
Discovery:RDS/TorIPCaller	サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース	RDS ログインアクティビティのモニタリング	中

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
Exfiltration:RDS/AnomalousBehavior	サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース	RDS DB アクティビティのモニタリング	低*
Exfiltration:RDS/MaliciousIPCaller	サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース	RDS DB アクティビティのモニタリング	高
Exfiltration:RDS/TorIPCaller	サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース	RDS DB アクティビティのモニタリング	高
Impact:RDS/AnomalousBehavior.Delete	サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース	RDS DB アクティビティのモニタリング	低*
Impact:RDS/AnomalousBehavior.Drop	サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース	RDS DB アクティビティのモニタリング	中*
Impact:RDS/MaliciousIPCaller	サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース	RDS DB アクティビティのモニタリング	高

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
Impact:RDS/TorIPCaller	サポートされている Amazon Aurora、Amazon RDS および Aurora Limitless データベース	RDS DB アクティビティのモニタリング	高
Discovery:S3/AnomalousBehavior	Amazon S3	S3 CloudTrail データイベント	低
Discovery:S3/MaliciousIPCaller	Amazon S3	S3 CloudTrail データイベント	高
Discovery:S3/MaliciousIPCaller.Custom	Amazon S3	S3 CloudTrail データイベント	高
Discovery:S3/TorIPCaller	Amazon S3	S3 CloudTrail データイベント	中
Exfiltration:S3/AnomalousBehavior	Amazon S3	S3 CloudTrail データイベント	高
Exfiltration:S3/MaliciousIPCaller	Amazon S3	S3 CloudTrail データイベント	高
Impact:S3/AnomalousBehavior.Delete	Amazon S3	S3 CloudTrail データイベント	高
Impact:S3/AnomalousBehavior.Permission	Amazon S3	S3 CloudTrail データイベント	高
Impact:S3/AnomalousBehavior.Write	Amazon S3	S3 CloudTrail データイベント	中
Impact:S3/MaliciousIPCaller	Amazon S3	S3 CloudTrail データイベント	高

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
PenTest:S3/KaliLinux	Amazon S3	S3 CloudTrail データイベント	中
PenTest:S3/ParrotLinux	Amazon S3	S3 CloudTrail データイベント	中
PenTest:S3/PentooLinux	Amazon S3	S3 CloudTrail データイベント	中
UnauthorizedAccess:S3/TorIPCaller	Amazon S3	S3 CloudTrail データイベント	高
UnauthorizedAccess:S3/MaliciousIPCaller.Custom	Amazon S3	S3 CloudTrail データイベント	高
Backdoor:Runtime/C&CActivity.B	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
Backdoor:Runtime/C&CActivity.B!DNS	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
CryptoCurrency:Runtime/BitcoinTool.B	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
CryptoCurrency:Runtime/BitcoinTool.B!DNS	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
DefenseEvasion:Runtime/FilelessExecution	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中
DefenseEvasion:Runtime/KernelModuleLoaded	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
DefenseEvasion:Runtime/SensitiveFileModified	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中
DefenseEvasion:Runtime/ProcessInjection.Proc	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
DefenseEvasion:Runtime/ProcessInjection.Ptrace	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中
DefenseEvasion:Runtime/ProcessInjection.VirtualMemoryWrite	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
DefenseEvasion:Runtime/PtraceAntiDebugging	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	低

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
DefenseEvasion:Runtime/SuspiciousCommand	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
Discovery:Runtime/SuspiciousCommand	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	低
Execution:Runtime/MaliciousFileExecuted	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
Execution:Runtime/MaliciousFileExecuted.Custom	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
Execution:Runtime/NewBinaryExecuted	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中
Execution:Runtime/NewLibraryLoaded	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中
Execution:Runtime/SuspiciousCommand	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	変数

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
Execution:Runtime/SuspiciousShellCreated	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	低
Execution:Runtime/SuspiciousTool	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	変数
Execution:Runtime/ReverseShell	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
Impact:Runtime/AbusedDomainRequest.Reputation	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中
Impact:Runtime/BitcoinDomainRequest.Reputation	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
Impact:Runtime/CryptoMinerExecuted	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
Impact:Runtime/MaliciousDomainRequest.Reputation	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
Impact:Runtime/SuspiciousDomainRequest.Reputation	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	低
Persistence:Runtime/SuspiciousCommand	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中
Persistence:Runtime/SensitiveFileModified	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中
PrivilegeEscalation:Runtime/CGroupsReleaseAgentModified	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
PrivilegeEscalation:Runtime/ContainerMountsHostDirectory	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中
PrivilegeEscalation:Runtime/DockerSocketAccessed	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中
PrivilegeEscalation:Runtime/ElevationToRoot	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
PrivilegeEscalation:Runtime/RuncContainerEscape	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
PrivilegeEscalation:Runtime/SuspiciousCommand	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中
PrivilegeEscalation:Runtime/SensitiveFileModified	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中
PrivilegeEscalation:Runtime/UserfaultfdUsage	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中
Trojan:Runtime/BlackholeTraffic	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中
Trojan:Runtime/BlackholeTraffic!DNS	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中
Trojan:Runtime/DropPoint	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中

結果タイプ	リソースタイプ	基本データソース/機能	検出結果の重大度
Trojan:Runtime/DGADomainRequest.C!DNS	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
Trojan:Runtime/DriveBySourceTraffic!DNS	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
Trojan:Runtime/DropPoint!DNS	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	中
Trojan:Runtime/PhishingDomainRequest!DNS	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
UnauthorizedAccess:Runtime/MetadataDNSRebind	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
UnauthorizedAccess:Runtime/TorClient	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高
UnauthorizedAccess:Runtime/TorRelay	インスタンス、EKS クラスター、ECS クラスター、またはコンテナ	ランタイムモニタリング	高

Amazon GuardDuty の検出結果の理解と生成

GuardDuty の検出結果は、、 AWS アカウントワークロード、およびデータ内で検出された潜在的なセキュリティ問題を表します。GuardDuty は、 AWS 環境で予期しない潜在的に悪意のあるアクティビティを検出するたびに検出結果を生成します。

GuardDuty の結果は、GuardDuty コンソールの調査結果ページで、または AWS CLI または API オペレーションを使用して表示および管理できます。GuardDuty の検出結果を管理する方法については、「Amazon GuardDuty の検出結果の管理」を参照してください。

トピック:

GuardDuty の検出結果の形式

GuardDuty の検出結果タイプの形式と GuardDuty で追跡するさまざまな脅威の目的について説明します。

サンプルの検出結果

GuardDuty コンソールで、または GuardDuty API や AWS CLI コマンドを使用して、検出結果のサンプルを生成します。生成される検出結果のサンプルには、GuardDuty の各検出結果に関連付けられた検出結果の詳細を理解するのに役立つ架空の詳細が含まれています。これらの検出結果は、[SAMPLE] のプレフィックスでマークされます。

専用アカウントでの GuardDuty の検出結果のテスト

環境で特定の GuardDuty の検出結果をテストできます。専用の非本番 AWS アカウントで `guardduty-tester` スクリプトを実行します。GuardDuty が検出結果を検出してシミュレートするために、特定のリソースを環境にデプロイします。このエクスペリエンスは検出結果サンプルの生成とは異なります。

GuardDuty コンソールで生成された検出結果を表示する

GuardDuty コンソールで生成された検出結果を確認する方法について説明します。

GuardDuty の検出結果の重大度

各 GuardDuty の検出結果には、 AWS 環境内の潜在的なリスクを反映する重要度レベルが関連付けられています。このセクションでは、各重大度レベルが意味する内容について説明します。

検出結果の詳細

アカウントで生成される GuardDuty の検出結果に関連付けられる詳細について説明します。このトピックには、GuardDuty の基本的な脅威検出と Extended Threat Detection、および専用の保護プランに関連付けられた詳細が含まれています。

GuardDuty の検出結果の集約

同じ検出結果タイプが複数出現した場合に GuardDuty がどう処理するかについて説明します。検出された同じ検出結果タイプを集計することで、GuardDuty は元の検出結果タイプを最新の詳細情報で更新します。

GuardDuty 検出結果タイプ

このセクションでは、関連付けられた基本データソースまたはマッピングされた GuardDuty 機能に基づく GuardDuty の検出結果タイプの一覧を示します。各検出結果タイプについて調べるには、その検出結果を選択して、その説明や、検出結果を修復する場合の手順などの詳細を確認してください。

GuardDuty の検出結果の形式

GuardDuty は AWS 、環境で疑わしい動作や予期しない動作を検出すると、検出結果を生成します。検出結果は、GuardDuty で検出した潜在的なセキュリティ問題に関する詳細を含む通知です。GuardDuty コンソールで生成された検出結果を表示する には、何が起こったか、疑わしいアクティビティに関与した AWS リソース、このアクティビティがいつ発生したか、根本原因を理解するのに役立つ関連情報が含まれます。

検出結果の詳細で最も役立つ情報の 1 つは、[finding type] (結果タイプ) です。検出結果タイプの目的は、潜在的なセキュリティ問題について簡潔でわかりやすい説明を提供することです。たとえば、GuardDuty Recon:EC2/PortProbeUnprotectedPort 検出結果タイプは、 AWS 環境のどこかに EC2 インスタンスに保護されていないポートがあり、潜在的な攻撃者が調査していることをすばやく知らせます。

GuardDuty では、次のフォーマットを使って、生成するさまざまな検出結果タイプに名前を付けます。

ThreatPurpose:ResourceTypeAffected/ThreatFamilyName.DetectionMechanism!Artifact

このフォーマットの各部分は、検出結果タイプの特徴を表します。これらの特徴には、次のような説明があります。

- ThreatPurpose - 攻撃型または潜在的な攻撃型の脅威の主な目的についての説明です。GuardDuty 脅威の目的の一覧表については、次のセクションを参照してください。
- ResourceTypeAffected - この検出結果でどの AWS リソースが攻撃対象の候補として特定されたかを示します。現在、GuardDuty は「GuardDuty のアクティブな検出結果タイプ」にリストされているリソースタイプの検出結果を生成できます。
- ThreatFamilyName - GuardDuty で検出された全体的な脅威または潜在的な悪意のあるアクティビティの説明です。例えば、NetworkPortUnusual の値は、GuardDuty の検出結果で識別された EC2 インスタンスに、識別された特定のリモートポートでの通信履歴がないことを示します。
- DetectionMechanism - どの GuardDuty が検出結果を検出した方法を説明します。これは、一般的な検出結果タイプの変動や、GuardDuty が特定のメカニズムを使用して検出した検出結果を示すために使用できます。例えば、Backdoor:EC2/DenialOfService.Tcp は、TCP 上でサービス拒否 (DoS) が検出されたことを示します。UDP バリアントは Backdoor:EC2/DenialOfService.Udp です。

  .Custom の値は、GuardDuty がカスタム脅威リストに基づいて検出結果を検出したことを示します。詳細については、「エンティティリストおよび IP アドレスリスト」を参照してください。

  .Reputation の値は、GuardDuty がドメインレピュテーションスコアモデルを使用して検出結果を検出したことを示します。詳細については、「 がクラウドの最大のセキュリティ脅威 AWS を追跡し、それらをシャットダウンするのに役立つ方法」を参照してください。
- Artifact - 悪意のあるアクティビティで使用されたツールが所有する特定のリソースを示します。例えば、検出結果タイプ CryptoCurrency:EC2/BitcoinTool.B!DNS の DNS は、Amazon EC2 インスタンスがビットコインに関連する既知のドメインと通信していることを示します。

> **Note**
>
> Artifact はオプションであり、すべての GuardDuty 検出結果タイプで使用できるとは限りません。

脅威の目的

GuardDuty において、[threat purpose] (脅威の目的) は、攻撃型または潜在的な攻撃の段階など脅威の主な目的について説明します。例えば、バックドアなどの脅威の目的は、攻撃型であると示します。ただし、MITRE ATT&CK 戦術と一致する [Impact] (影響) などの脅威の目的があります。MITRE ATT&CK 戦術は、攻撃者の攻撃サイクルにおける異なるフェーズを示します。現行の GuardDutyリリースの発売で、ThreatPurpose は次の値に設定できます。

Backdoor

この値は、攻撃者が AWS リソースを侵害し、そのリソースを変更して、ホームコマンドアンドコントロール (C&C) サーバーに連絡して悪意のあるアクティビティに関する詳細な指示を受けられるようにしたことを示します。

Behavior

この値は、GuardDutyは特定の AWS リソースの確立されたベースラインとは異なるアクティビティやアクティビティパターンを検出したことを示します。

CredentialAccess

この値は、ユーザーの環境からパスワード、ユーザー名、アクセスキーなどの認証情報を盗むために攻撃者が使用する可能性のあるアクティビティパターンを GuardDuty が検出したことを示します。この脅威の目的は、MITRE ATT&CK 戦術に基づいています。

Cryptocurrency

この値は、GuardDuty が環境内の AWS リソースが暗号通貨 (Bitcoin など) に関連付けられたソフトウェアをホストしていることを検出したことを示します。

DefenseEvasion

この値は、ユーザーの環境に侵入している間、GuardDuty は攻撃者が検出を回避するために使用する可能性のあるアクティビティまたはアクティビティパターンを検出したことを示します。この脅威の目的は、MITRE ATT&CK 戦術に基づいています。

Discovery

この値は、GuardDuty がシステムおよび内部ネットワークに関する知識を拡張するために使用する可能性のあるアクティビティまたはアクティビティパターンを検出したことを示します。この脅威の目的は、MITRE ATT&CK 戦術に基づいています。

実行

この値は、GuardDuty が攻撃者が AWS 環境を探索したり、データを盗んだりするために実行を試みたり、悪意のあるコードを既に実行したりする可能性があることを検出したことを示します。この脅威の目的は、MITRE ATT&CK 戦術に基づいています。

Exfiltration

この値は、GuardDuty が環境からデータを盗もうとするときに攻撃者が使用する可能性のあるアクティビティまたはアクティビティパターンを検出したことを示します。この脅威の目的は、MITRE ATT&CK 戦術に基づいています。

Impact

この値は、GuardDuty がアクティビティまたはアクティビティパターンを検出することにより、ユーザーのシステムおよびデータを操作、中断、または破壊しようとしていることを示しています。この脅威の目的は、MITRE ATT&CK 戦術に基づいています。

InitialAccess

この値は、攻撃者がユーザーの環境へのアクセスを確立しようとするときの、攻撃の初期アクセス段階に一般的に関連しています。この脅威の目的は、MITRE ATT&CK 戦術に基づいています。

Pentest

AWS リソースの所有者や承認された担当者は、 AWS アプリケーションに対して意図的にテストを実行して、オープンなセキュリティグループや過度に寛容なアクセスキーなどの脆弱性を見つける場合があります。これらの侵入テストは、攻撃者が気づく前に脆弱なリソースを特定してロックダウンする目的で実行されます。ただし、承認済みペンテスターが使用する一部のツールは一般的なものであるため、不正なユーザーや攻撃者がこのテストに便乗して利用する場合があります。GuardDuty は、このようなアクティビティの真の意図は特定できませんが、[Pentest] (侵入テスト) 値により、GuardDuty がこのようなアクティビティを検出し、および既知の侵入テストツールで生成したアクティビティと類似しユーザーのネットワークへの悪意のある調査を示します。

Persistence

この値は、GuardDuty が初期アクセスルートが切断された場合でも、攻撃者がシステムへのアクセスを維持するために使用する可能性のあるアクティビティまたはアクティビティパターンを検出したことを示します。例えば、既存のユーザーの侵入して得た認証情報を介してアクセスした後に、新しい IAM ユーザーを作成することが含まれます。既存のユーザーの認証情報が削除されると、攻撃者はオリジナルイベントの一部として検出されなかった新しいユーザーへのアクセスを保持します。この脅威の目的は、MITRE ATT&CK 戦術に基づいています。

Policy

この値は、 AWS アカウント が推奨されるセキュリティのベストプラクティスに反する動作をしていることを示します。例えば、 AWS リソースまたは環境に関連付けられたアクセス許可ポリシーの意図しない変更や、ほとんどまたはまったく使用すべきでない特権アカウントの使用などです。

PrivilegeEscalation

この値は、ユーザーの AWS 環境下の関連プリンシパルが、攻撃者にネットワークへのより高いレベルの許可を取得するために使用する可能性のあることを通知します。この脅威の目的は、MITRE ATT&CK 戦術に基づいています。

Recon

この値は、攻撃者が環境の偵察を実行するときに、アクセスを拡大したり、リソースを利用したりする方法を決定するために使用する可能性のあるアクティビティまたはアクティビティパターンを GuardDuty が検出したことを示します。例えば、このアクティビティには、ポートを調査したり、ユーザーをリストアップしたり、特にデータベーステーブルをリストアップしたりして、AWS 環境の脆弱性を探そうとすることを含めることができます。

Stealth

この値は、攻撃者が積極的にアクションを隠そうとしていることを示します。例えば、匿名化したプロキシサーバーを使用し、アクティビティの本質の判断を非常に難しくしています。

Trojan

この値は、悪意のあるアクティビティを密かに実行するトロイの木馬プログラムが攻撃に使用されていることを示します。このソフトウェアは合法的なプログラムを装う場合があります。ユーザーは、このソフトウェアを誤って実行することがあります。脆弱性を特定して自動的に実行されることもあります。

UnauthorizedAccess

この値は、承認されていない個人による不審なアクティビティまたは不審なアクティビティパターンが GuardDuty で検出されたことを示します。

GuardDuty マルウェア検出のスキャンエンジン

Amazon GuardDuty には、内部で構築および管理されるスキャンエンジンとサードパーティーベンダーがあります。どちらも、 AWSをターゲットとする可能性のあるさまざまな種類のマルウェアを可視化する多様な内部フィードから取得された侵害のインジケータ (IoC) を使用します。また、GuardDuty には、セキュリティエンジニアによって追加された YARA ルールに基づく検出定義と、ヒューリスティックおよび機械学習 (ML) モデルに基づく検出も含まれます。Amazon S3 オブジェクトをスキャンする場合、GuardDuty マルウェア保護は、同一のスキャン定義とエンジンを使用して同一オブジェクトを複数回スキャンしても、一貫した結果を生成します。署名ベースの検出には、バイトの一致だけでなく、複雑な可能性のあるコードのスニペットも含まれており、スキャナーはコンテンツを解析して決定を下すことができます。

マルウェアスキャンエンジンは、マルウェアデトネーションにより実際のシステムで実行されるサンプルをモニタリングするライブ行動分析を実行しません。GuardDuty ソリューションは主に、ファイルベースの検出です。ファイルレスマルウェアを検出するために、GuardDuty は Amazon EKS、Amazon EC2、Amazon ECS (AWS Fargateを含む) の ランタイムモニタリング などのエージェントベースのソリューションを提供します。

GuardDuty がマルウェアをスキャンするファイル形式に制限がないと、使用するスキャンエンジンでは、クリプトマイナー、ランサムウェア、Web シェルなど、さまざまなタイプのマルウェアを検出できます。フルマネージド型の GuardDuty スキャンエンジンは、マルウェア署名のリストを 15 分ごとに継続的に更新します。

スキャンエンジンは、内部マルウェアデトネーションコンポーネントを使用する GuardDuty 脅威インテリジェンスシステムの一部です。これにより、複数のソースからマルウェアと悪意のないサンプルを独立して収集することで、新しい脅威インテリジェンスが生成されます。脅威インテリジェンスシステムからのファイルハッシュ IoC タイプは、さらにマルウェアスキャンエンジンにフィードされ、既知の不正なファイルハッシュに基づいてマルウェアを検出します。

GuardDuty のサンプルの検出結果生成

Amazon GuardDuty はサンプル検出結果を生成し、生成できるさまざまな検出結果タイプを視覚化して理解することができます。サンプル検出結果を生成すると、GuardDuty は、攻撃シークエンス検出結果タイプを含む、サポートされている検出結果タイプごとに、1 つのサンプルを現在の検出結果リストに追加します。

生成されるサンプルは、プレースホルダーの値が入力された近似値です。生成されるサンプルはご利用の環境での実際の検出結果とは異なりますが、EventBridge イベントやフィルターなど、GuardDuty のさまざまな設定をテストするために使用できます。検出結果タイプに使用できる値のリストについては、GuardDuty 検出結果タイプ テーブルを参照してください。

GuardDuty の コンソールや API で使用したサンプル検出結果の生成

任意のアクセス方法を選択して、サンプルの検出結果を生成します。

> **Note**
>
> GuardDuty コンソールを使用すると、各検出結果タイプのいずれかを生成することができます。1 つ以上の特定の検出結果タイプを生成するには、関連する API/CLI ステップを実行します。

Console

サンプル検出結果を生成するには、次の手順を使用します。このプロセスでは、GuardDuty の検出タイプごとに 1 つのサンプル検出結果を生成します。

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで [Settings] (設定) を選択します。
3. [Settings] (設定) ページの [Sample findings] (結果のサンプル) で、[Generate sample findings] (結果サンプルの生成) を選択します。
4. ナビゲーションペインで [Findings] (結果) を選択します。[Current findings] (最近の結果) ページに [SAMPLE] とプレフィックスされたサンプル検出結果が表示されます。

API/CLI

GuardDuty の任意の検出タイプに一致する単一のサンプル検出結果を生成するには、CreateSampleFindings API の場合、タイプの検索に使用できる値は、GuardDuty 検出結果タイプ テーブルにリストされています。

これは、CloudWatch Events ルールのテストや検出結果に基づくオートメーションに役立ちます。次の例は、 AWS CLIを使用して `Backdoor:EC2/DenialOfService.Tcp` タイプの単一サンプル検出結果を生成する方法を示しています。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty create-sample-findings --detector-id 12abc34d567e8fa901bc2d34e56789f0
 --finding-types Backdoor:EC2/DenialOfService.Tcp
```

これらのメソッドによって生成されたサンプル検出結果のタイトルは、常にコンソール内で [SAMPLE] で始まります。サンプルの検出結果には、検出結果の JSON の詳細の additionalInfo セクションに値 `"sample": true` があります。

生成された検出結果に関連する検出結果の重大度や侵害された可能性のあるリソースなどの検出結果の詳細については、「GuardDuty の検出結果の重大度」および「検出結果の詳細」を参照してください。

環境 AWS アカウント 内で分離された専用 のシミュレートされたアクティビティに基づいて一般的な検出結果を生成するには、「」を参照してください専用アカウントでの GuardDuty の検出結果のテスト。

専用アカウントでの GuardDuty の検出結果のテスト

このドキュメントを使用して、 AWS アカウントにデプロイされるテストリソースに対して GuardDuty の検出結果を生成するテスタースクリプトを実行します。これらのステップは、特定の GuardDuty の検出結果タイプについてと、検出結果の詳細がアカウント内の実際のリソースをどのように検索するかを理解して学習したい場合に実行できます。このエクスペリエンスは、サンプルの検出結果の生成とは異なります。GuardDuty の検出結果をテストするエクスペリエンスの詳細については、「考慮事項」を参照してください。

内容

- 考慮事項
- テスタースクリプトが生成できる GuardDuty の検出結果
- ステップ 1 - 前提条件
- ステップ 2 - AWS リソースをデプロイする
- ステップ 3 - テスタースクリプトを実行する
- ステップ 4 - AWS テストリソースをクリーンアップする
- 一般的な問題のトラブルシューティング

考慮事項

先に進む前に、次の点を考慮してください。

- GuardDuty は、専用の非本番環境 AWS アカウントにテスターをデプロイすることをお勧めします。このアプローチにより、テスターによって生成された GuardDuty の検出結果を正しく識別できます。また、GuardDuty テスターは、他のアカウントで許可されている内容を超える IAM アクセス許可を必要とする可能性のあるさまざまなリソースをデプロイします。専用アカウントを使用すると、明確なアカウント境界でアクセス許可を適切にスコープできます。
- テスタースクリプトは、さまざまな AWS リソースの組み合わせで 100 を超える GuardDuty の検出結果を生成します。現在、これにはすべての GuardDuty 検出結果タイプは含まれません。このテスタースクリプトで生成できる検出結果タイプのリストについては、「テスタースクリプトが生成できる GuardDuty の検出結果」を参照してください。

> **メモ**
>
> 攻撃シーケンスの検出結果タイプを視覚化するために、テスタースクリプトは AttackSequence:EKS/CompromisedCluster と AttackSequence:S3/CompromisedData のみを生成します。AttackSequence:IAM/CompromisedCredentials を視覚化して理解するには、アカウントで サンプルの検出結果 を生成できます。

- GuardDuty テスターが想定どおりに機能するには、テスターリソースがデプロイされているアカウントで GuardDuty を有効にする必要があります。実行するテストに応じて、テスターは適切な GuardDuty 保護プランが有効になっているかどうかを評価します。有効になっていない保護プランでは、GuardDuty は、GuardDuty が検出結果を生成するテストを実行するのに十分な期間だけ必要な保護プランを有効にするアクセス許可をリクエストします。その後、テストが完了すると、GuardDuty は保護プランを無効にします。

  初めての GuardDuty の有効化

  GuardDuty が特定のリージョンで初めて専用アカウントで有効化されると、アカウントは自動的に 30 日間の無料トライアルに登録されます。

  GuardDuty には、オプションの保護プランが提供されています。GuardDuty を有効にすると、特定の保護プランも有効になり、GuardDuty の 30 日間の無料トライアルに含められます。詳細については、「GuardDuty の 30 日間無料トライアルの使用」を参照してください。

  テスタースクリプトを実行する前に GuardDuty がアカウントで既に有効になっている場合

  GuardDuty が既に有効になっている場合、テスタースクリプトはパラメータに基づいて、特定の保護プランの設定ステータスや、検出結果の生成に必要なその他のアカウントレベルの設定をチェックします。

  このテスタースクリプトを実行すると、リージョンの専用アカウントで特定の保護プランが初めて有効になる場合があります。これにより、その保護プランの 30 日間の無料トライアルが開始します。各保護プランに関連する無料トライアルの詳細については、「GuardDuty の 30 日間無料トライアルの使用」を参照してください。

- GuardDuty テスターインフラストラクチャがデプロイされている限り、ペネトレーションテストインスタンスから UnauthorizedAccess:EC2/TorClient 検出結果を受け取ることがあります。

テスタースクリプトが生成できる GuardDuty の検出結果

現在、テスタースクリプトは、Amazon EC2、Amazon EKS、Amazon S3、IAM、EKS 監査ログに関連する次の検出結果タイプを生成します。

- AttackSequence:EKS/CompromisedCluster
- AttackSequence:S3/CompromisedData
- Backdoor:EC2/C&CActivity.B!DNS
- Backdoor:EC2/DenialOfService.Dns
- Backdoor:EC2/DenialOfService.Udp
- CryptoCurrency:EC2/BitcoinTool.B!DNS
- Impact:EC2/AbusedDomainRequest.Reputation
- Impact:EC2/BitcoinDomainRequest.Reputation
- Impact:EC2/MaliciousDomainRequest.Reputation
- Impact:EC2/SuspiciousDomainRequest.Reputation
- Recon:EC2/Portscan
- Trojan:EC2/BlackholeTraffic!DNS
- Trojan:EC2/DGADomainRequest.C!DNS
- Trojan:EC2/DNSDataExfiltration
- Trojan:EC2/DriveBySourceTraffic!DNS
- Trojan:EC2/DropPoint!DNS
- Trojan:EC2/PhishingDomainRequest!DNS
- UnauthorizedAccess:EC2/MaliciousIPCaller.Custom
- UnauthorizedAccess:EC2/RDPBruteForce
- UnauthorizedAccess:EC2/SSHBruteForce
- PenTest:IAMUser/KaliLinux
- Recon:IAMUser/MaliciousIPCaller.Custom
- Recon:IAMUser/TorIPCaller
- Stealth:IAMUser/CloudTrailLoggingDisabled
- Stealth:IAMUser/PasswordPolicyChange
- UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.OutsideAWS

- UnauthorizedAccess:IAMUser/MaliciousIPCaller.Custom
- UnauthorizedAccess:IAMUser/TorIPCaller
- Discovery:Kubernetes/MaliciousIPCaller.Custom
- Discovery:Kubernetes/SuccessfulAnonymousAccess
- Discovery:Kubernetes/TorIPCaller
- Execution:Kubernetes/ExecInKubeSystemPod
- Impact:Kubernetes/MaliciousIPCaller.Custom
- Persistence:Kubernetes/ContainerWithSensitiveMount
- Policy:Kubernetes/AdminAccessToDefaultServiceAccount
- Policy:Kubernetes/AnonymousAccessGranted
- PrivilegeEscalation:Kubernetes/PrivilegedContainer
- UnauthorizedAccess:Lambda/MaliciousIPCaller.Custom
- Discovery:S3/MaliciousIPCaller.Custom
- Discovery:S3/TorIPCaller
- PenTest:S3/KaliLinux
- Policy:S3/AccountBlockPublicAccessDisabled
- Policy:S3/BucketAnonymousAccessGranted
- Policy:S3/BucketBlockPublicAccessDisabled
- Policy:S3/BucketPublicAccessGranted
- Stealth:S3/ServerAccessLoggingDisabled
- UnauthorizedAccess:S3/MaliciousIPCaller.Custom
- UnauthorizedAccess:S3/TorIPCaller
- Backdoor:Runtime/C&CActivity.B!DNS
- CryptoCurrency:Runtime/BitcoinTool.B!DNS
- DefenseEvasion:Runtime/ProcessInjection.Ptrace
- DefenseEvasion:Runtime/ProcessInjection.VirtualMemoryWrite
- Execution:Runtime/ReverseShell
- Impact:Runtime/AbusedDomainRequest.Reputation
- Impact:Runtime/BitcoinDomainRequest.Reputation

- Impact:Runtime/MaliciousDomainRequest.Reputation
- Impact:Runtime/SuspiciousDomainRequest.Reputation
- PrivilegeEscalation:Runtime/ContainerMountsHostDirectory
- PrivilegeEscalation:Runtime/DockerSocketAccessed
- Trojan:Runtime/BlackholeTraffic!DNS
- Trojan:Runtime/DGADomainRequest.C!DNS
- Trojan:Runtime/DriveBySourceTraffic!DNS
- Trojan:Runtime/DropPoint!DNS
- Trojan:Runtime/PhishingDomainRequest!DNS

ステップ 1 - 前提条件

テスト環境を準備するには、次の項目が必要です。

- Git – 使用するオペレーティングシステムに基づく git コマンドラインツールをインストールします。

  これは、`amazon-guardduty-tester` リポジトリのクローンを作成するために必要です。
- AWS Command Line Interface – コマンドラインシェルでコマンド AWS のサービス を使用して とやり取りできるオープンソースツール。詳細については、「AWS Command Line Interface ユーザーガイド」の「Get started with AWS CLI」を参照してください。
- AWS Systems Manager – を使用してマネージドノードで Session Manager セッションを開始する AWS CLI には、ローカルマシンに Session Manager プラグインをインストールする必要があります。詳細については、「AWS Systems Manager ユーザーガイド」の「AWS CLI用の Session Manager プラグインをインストールする」を参照してください。
- Node Package Manager (NPM) — NPM をインストールして、すべての依存関係をインストールします。
- Docker – Docker がインストールされている必要があります。インストール手順については、Docker のウェブサイトを参照してください。

  Docker がインストールされていることを確認するには、次のコマンドを実行し、次のような出力があることを確認します。

```
$ docker --version
Docker version 19.03.1
```

- AWS Marketplace で Kali Linux イメージをサブスクライブします。

ステップ 2 - AWS リソースをデプロイする

このセクションでは、主要な概念のリストと、特定の AWS リソースを専用アカウントにデプロイする手順について説明します。

概念

次のリストに、リソースのデプロイに役立つコマンドに関連する主要な概念を示します。

- AWS Cloud Development Kit (AWS CDK) – CDK は、コードでクラウドインフラストラクチャを定義し、それをプロビジョニングするためのオープンソースのソフトウェア開発フレームワークです CloudFormation。CDK は、いくつかのプログラミング言語をサポートして、コンストラクトと呼ばれる再利用可能なクラウドコンポーネントを定義します。これらをまとめてスタックとアプリケーションに構成できます。次に、CDK アプリケーションを にデプロイ CloudFormation して、リソースをプロビジョニングまたは更新できます。詳細については、「 とは」を参照してください AWS CDK。 「 AWS Cloud Development Kit (AWS CDK) デベロッパーガイド」の「
- ブートストラップ — AWS 環境で を使用するための準備プロセスです AWS CDK。CDK スタックを AWS 環境にデプロイする前に、まず環境をブートストラップする必要があります。によって使用される環境内の特定の AWS リソースをプロビジョニングするこのプロセス AWS CDK は、次のセクション - で実行するステップの一部ですAWS リソースをデプロイするステップ。

  ブートストラップの仕組みの詳細については、「AWS Cloud Development Kit (AWS CDK) 開発者ガイド」の「Bootstrapping」を参照してください。

AWS リソースをデプロイするステップ

リソースのデプロイを開始するには、次の手順を実行します。

1. 専用のアカウントリージョン変数が `bin/cdk-gd-tester.ts` ファイルで手動で設定されていない限り、 AWS CLI デフォルトのアカウントとリージョンを設定します。詳細については、「AWS Cloud Development Kit (AWS CDK) 開発者ガイド」の「Environments」を参照してください。
2. 以下のコマンドを実行して、リソースをデプロイします。

```
git clone https://github.com/awslabs/amazon-guardduty-tester && cd amazon-guardduty-tester
```

```
npm install
cdk bootstrap
cdk deploy
```

最後のコマンド (cdk deploy) は、ユーザーに代わって CloudFormation スタックを作成します。このスタックの名前は GuardDutyTesterStack です。

このスクリプトの一部として、GuardDuty はアカウントに GuardDuty の検出結果を生成する新しいリソースを作成します。次のタグキーと値のペアも、Amazon EC2 インスタンスに追加されます。

`CreatedBy:GuardDuty Test Script`

また、Amazon EC2 インスタンスには、EKS ノードと ECS クラスターをホストする EC2 インスタンスも含まれます。

> **インスタンスのタイプ**
>
> GuardDuty は、テストを正常に実行するために必要な最小限のパフォーマンスを提供する費用対効果の高いインスタンスタイプを使用するように設計されています。vCPU 要件のため、Amazon EKS ノードグループには `t3.medium` が必要であり、DenialOfService 検出テストに必要なネットワーク容量が増加するため、ドライバーノードには `m6i.large` が必要です。他のすべてのテストでは、GuardDuty は `t3.micro` インスタンスタイプを使用します。インスタンスタイプの詳細については、「Amazon EC2 Instances Types Guide」の「Available sizes」を参照してください。

ステップ 3 - テスタースクリプトを実行する

これは 2 ステップのプロセスであり、まずテストドライバーでセッションを開始し、次にスクリプトを実行して特定のリソースの組み合わせで GuardDuty の検出結果を生成する必要があります。

パート A - テストドライバーでセッションを開始する

1. リソースをデプロイしたら、リージョンコードを現在のターミナルセッションの変数に保存します。次のコマンドを使用し、*`us-east-1`* はリソースをデプロイしたリージョンコードに置き換えます。

   ```
   $ REGION=us-east-1
   ```

2. テスタースクリプトは AWS Systems Manager (SSM) でのみ使用できます。テスターホストインスタンスでインタラクティブシェルを起動するには、ホストの InstanceId をクエリします。
3. 次のコマンドを使用してテスタースクリプトのセッションを開始します。

```
aws ssm start-session
  --region $REGION
  --document-name AWS-StartInteractiveCommand
  --parameters command="cd /home/ssm-user/py_tester && bash -l"
  --target $(aws ec2 describe-instances
    --region $REGION
    --filters "Name=tag:Name,Values=Driver-GuardDutyTester"
    --query "Reservations[].Instances[?State.Name=='running'].InstanceId"
    --output text)
```

パート B - 検出結果を生成する

テスタースクリプトは、入力に基づいて検出結果を生成する Bash スクリプトを動的に構築する Python ベースのプログラムです。1 つ以上の AWS リソースタイプ、GuardDuty 保護プラン、脅威の目的 (戦術) 、基本データソース、または に基づいて結果を柔軟に生成できますthe section called “テスタースクリプトが生成できる GuardDuty の検出結果”。

次のコマンド例をリファレンスとして使用し、1 つ以上のコマンドを実行して、調査する検出結果を生成します。

```
python3 guardduty_tester.py
python3 guardduty_tester.py --all
python3 guardduty_tester.py --s3
python3 guardduty_tester.py --tactics discovery
python3 guardduty_tester.py --ec2 --eks --tactics backdoor policy execution
python3 guardduty_tester.py --eks --runtime only
python3 guardduty_tester.py --ec2 --runtime only --tactics impact
python3 guardduty_tester.py --log-source dns vpc-flowlogs
python3 guardduty_tester.py --finding 'CryptoCurrency:EC2/BitcoinTool.B!DNS'
```

有効なパラメータの詳細については、次のヘルプコマンドを実行して確認できます。

```
python3 guardduty_tester.py --help
```

パート C - 生成された検出結果を確認する

アカウントで生成された検出結果を表示する任意の方法を選択します。

GuardDuty console

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで 調査検出結果 を選択します。
3. 検出結果テーブルから、詳細を表示したい検出結果を選択します。これにより、検出結果の詳細パネルが開きます。詳細については、「Amazon GuardDuty の検出結果の理解と生成」を参照してください。
4. これらの検出結果をフィルタリングする場合は、リソースのタグキーと値を使用します。例えば、Amazon EC2 インスタンスに対して生成された検出結果をフィルタリングするには、[インスタンスタグキー] と [インスタンスタグキー] にタグキーと値ペア `CreatedBy:GuardDuty Test Script` を使用します。

API

- ListFindings を実行して、特定のディテクター ID の検出結果を表示します。パラメータを指定して、検出結果をフィルタリングできます。

  アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

AWS CLI

- 次の AWS CLI コマンドを実行して生成された結果を表示し、*`us-east-1`* および *`12abc34d567e8fa901bc2d34EXAMPLE`* を適切な値に置き換えます。

  ```
  aws guardduty list-findings --region us-east-1 --detector-
  id 12abc34d567e8fa901bc2d34EXAMPLE
  ```

  アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

検出結果のフィルタリングに使用できるパラメータの詳細については、「AWS CLI Command Reference」の「list-findings」を参照してください。

ステップ 4 - AWS テストリソースをクリーンアップする

「ステップ 3 - テストースクリプトを実行する」で行われたアカウントレベルの設定やその他の設定ステータスの更新は、テストースクリプトの終了時に元の状態に戻ります。

テストースクリプトを実行したら、 AWS テストリソースをクリーンアップすることを選択できます。これを選択するには、次のいずれかの方法を使用します。

- 次のコマンドを実行します。

```
cdk destroy
```

- GuardDutyTesterStack という名前の CloudFormation スタックを削除します。ステップの詳細については、「 CloudFormation コンソールでのスタックの削除」を参照してください。

一般的な問題のトラブルシューティング

GuardDuty はよくある問題を特定し、トラブルシューティングのステップを推奨しています。

- `Cloud assembly schema version mismatch` – CLI AWS CDK を、必要なクラウドアセンブリバージョンと互換性のあるバージョン、または利用可能な最新バージョンに更新します。詳細については、「AWS CDK CLI 互換性」を参照してください。
- `Docker permission denied` – 専用アカウントでコマンドを実行できるように、専用アカウントユーザーを docker または docker-users に追加します。ステップの詳細については、「デーモンソケットオプション」を参照してください。
- `Your requested instance type is not supported in your requested Availability Zone` – 一部のアベイラビリティーゾーンは特定のインスタンスタイプをサポートしていません。任意のインスタンスタイプをサポートするアベイラビリティーゾーンを特定し、AWS リソースのデプロイを再試行するには、次の手順を実行します。

  1. 任意の方法を選択して、インスタンスタイプをサポートするアベイラビリティーゾーンを確認します。

Console

希望のインスタンスタイプをサポートするアベイラビリティーゾーンを特定するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/ec2/ で Amazon EC2 コンソールを開きます。
2. ページの右上隅にある AWS リージョンセレクターを使用して、インスタンスを起動するリージョンを選択します。
3. ナビゲーションペインの [インスタンス] で、[インスタンスタイプ] を選択します。
4. [インスタンスタイプ] テーブルから、希望のインスタンスタイプを選択します。
5. [ネットワーク] で、[アベイラビリティーゾーン] にリストされているリージョンを表示します。

   この情報に基づいて、リソースをデプロイできる新しいリージョンを選択する必要がある場合があります。

AWS CLI

次のコマンドを実行して、アベイラビリティーゾーンのリストを表示します。必ず希望のインスタンスタイプとリージョン (*us-east-1*) を指定してください。

```
aws ec2 describe-instance-type-offerings --location-type availability-zone  --
filters Name=instance-type,Values=Preferred instance type --region us-east-1 --
output table
```

このコマンドの詳細については、「AWS CLI Command Reference」の「describe-instance-type-offerings」を参照してください。

このコマンドを実行するときにエラーが表示された場合は、最新バージョンの AWS CLIを使用していることを確認してください。詳細については、「AWS Command Line Interface ユーザーガイド」の「トラブルシューティング」を参照してください。

2. AWS リソースを再度デプロイし、任意のインスタンスタイプをサポートするアベイラビリティーゾーンを指定します。

AWS リソースのデプロイを再試行するには

1. `bin/cdk-gd-tester.ts` ファイルでデフォルトリージョンを設定します。

2. アベイラビリティーゾーンを指定するには、amazon-guardduty-tester/lib/common/network/vpc.ts ファイルを開きます。
3. このファイルでは、maxAzs: 2, を availabilityZones: ['*us-east-1a*', '*us-east-1c*'], に置き換え、ここでインスタンスタイプのアベイラビリティーゾーンを指定する必要があります。
4. 「AWS リソースをデプロイするステップ」の残りの手順を続行します。

GuardDuty コンソールで生成された検出結果を表示する

GuardDuty がセキュリティ問題のパターンに一致する活動を検出すると、GuardDuty は検出結果を生成します。この検出結果タイプは、このアクティビティ中に侵害された可能性があるリソースタイプに関連付けられています。GuardDuty が生成する検出結果タイプごとに関連付けられた詳細を表示できます。

GuardDuty 管理者アカウントを使用している場合は、生成された検出結果をメンバーアカウントに代わって表示できます。ただし、メンバーアカウントは、自分のアカウントで生成された検出結果を表示できます。メンバーアカウントは、他のメンバーアカウントで生成された検出結果を表示できません。

GuardDuty コンソールで検出結果を表示する手順

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. 左のナビゲーションペインで [検出結果] を選択します。

   GuardDuty は検出結果を表形式で表示します。デフォルトでは、このテーブルは [最終表示] 列の値に基づいて降順でソートされ、最新の検出結果が上部に表示されます。

   剣のアイコン

   ( )

   が付いた発見は、攻撃シーケンスの発見を表します。
3. 検出結果に関連する詳細を表示するには、その [タイトル] を選択します。これにより、検出結果の詳細サイドパネルが開きます。攻撃シーケンスの検出結果の場合、このサイドパネルには攻撃シーケンスの概要バージョンが含まれ、このビューを展開するには、[詳細を表示] を選択します。

   このサイドパネルにリストされているフィールドの詳細については、「検出結果の詳細」を参照してください。

4. (オプション) 検出結果 JSON をダウンロードする

   a. [検出結果] を選択し、次に [アクション] メニューを選択します。

   b. [アクション] メニューで、[JSON の表示とエクスポート] を選択します。

   c. JSON の結果ウィンドウで、[ダウンロード] を選択します。

> **Note**
>
> 場合によっては、GuardDuty は、ある検出結果が生成された後に、その結果が誤検出であることを認識することがあります。GuardDuty は検出結果の JSON 詳細に [Confidence] (信頼度) フィールドを設けて、その値をゼロに設定します。このようにして、GuardDuty はこうした検出結果を無視しても構わないことを知らせます。[Confidence] フィールドのない結果は誤検出とは見なされません。

検出結果ページの操作

このセクションでは、[検出結果] ページのさまざまな要素に関する重要な情報を提供します。これにより、生成された分析結果を脅威分析と対応のために分析することが可能になります。

以下のリストでは、生成された検出結果をよりよく理解するのに役立つ検出結果ページの要素について説明します。

- 脅威タイプ:

  脅威の種類には、個々の GuardDuty 検出結果と攻撃シーケンス検出結果が含まれます。デフォルトでは、このページにはすべての検出結果が表示されます。

  検出結果テーブルビューをフィルタリングするには、[脅威タイプ[メニューで、[攻撃シーケンスの検出結果のみ] または[個々の検出結果のみ] のいずれかを選択します。

- リソース列とカウント列:

  検出結果テーブルのリソース列には、侵害された可能性のある AWS リソースの名前が表示されます。攻撃シーケンスの検出結果の場合、この列には侵害された可能性のある AWS リソースの数が表示されます。リソース名を表示するには、[リソース] 列の下にある [番号] を選択します。

  [カウント] 列は、GuardDuty が特定の検出結果を観測した回数を示します。GuardDuty が、事前に特定されたセキュリティ問題に一致するアクティビティを検出すると、その特定の検出結果の力

ウントを増加させます。攻撃シーケンスの検出結果の場合、この列の値は、検出結果の生成に関連するシグナルと検出結果の合計数を示します。

- テーブル列による検出結果のソート:

  列ヘッダーの横に矢印がある場合は、列に基づいて検出結果テーブルをソートできます。列ヘッダーを選択すると、その列の値の昇順または降順で結果を並べ替えられます。

- 検出結果のフィルタリング:

  `Account ID` や `Resource type` などの特定のプロパティ属性に基づいて、検出結果テーブルをさらにフィルタリングできます。使用可能なフィルターのタイプの詳細については、GuardDuty の検出結果のフィルタリング を参照してください。

- ステータスと保存されたルール:

  [ステータス] メニューには、[現在] と [アーカイブ] の 2 つの値が含まれます。デフォルトビューは、テーブル内の[現在] の検出結果です。

  特定の条件に一致する検出結果を GuardDuty が生成しないようにしたい場合、その検出結果を抑制できます。GuardDuty はその検出結果をアーカイブします。GuardDuty が再度この検出結果を検知した場合、この観測結果について通知されません。アーカイブされた検出結果を具体的に表示するには、[ステータス] メニューで [アーカイブ済み] を選択します。

  保存されたルールは、指定した条件に一致する検出結果を自動的にフィルタリングしてアクションを実行するのに役立つ機能です。アクションには、検出結果のアーカイブや今後の通知からの抑制が含まれる場合があります。

  詳細については、「抑制ルール」を参照してください。

GuardDuty の検出結果の重大度

GuardDuty の各検出結果には、当社のセキュリティエンジニアが定めた環境の潜在的なリスクを示す重大度と値が割り当てられます。重大度の値の範囲は 1.0 ~ 10.0 です。値が大きいほど、セキュリティリスクが高いことを示します。検出結果が示す潜在的なセキュリティの問題への対応を判断するために役立つように、GuardDuty ではこの範囲を [重大]、[高]、[中]、[低] の重大度に分けています。

特定のタイプの検出結果は、検出結果に固有のコンテキストによって重大度が異なることがあります。すべての GuardDuty 検出結果タイプのデフォルトの重大度レベルの統合リストを表示するには、「GuardDuty のアクティブな検出結果タイプ」を参照してください。

以下のセクションでは、GuardDuty の検出結果に対して定義された重大度レベルについて説明します。

トピック

- 重大度重大
- 重大度高
- 重大度中
- 重大度低

重大度重大

値の範囲: 9.0 - 10.0

説明: 重大度重大レベルは、攻撃シーケンスが進行中であるか、最近発生した可能性があることを示します。IAM ユーザーのサインイン認証情報や Amazon S3 バケットなどの 1 つ以上の AWS リソースが侵害されている可能性があるか、既に侵害されている可能性があります。

推奨事項: GuardDuty では、すべての重要な重大度の検出結果の優先順位付けと修復を推奨しています。これらの問題はランサムウェア攻撃の一部であり、いつでも深刻化する可能性があるためです。関連するリソースの詳細を表示し、セキュリティ問題の対処を開始します。詳細については、「検出結果の修復」を参照してください。

重大度高

値の範囲: 7.0 - 8.9

説明: 重大度高は、問題になっているリソース (Amazon EC2 インスタンスや IAM ユーザーサインイン認証情報) が侵害され、不正な目的で活発に使用されていることを示します。

推奨: GuardDuty では、重大度高の検出結果のセキュリティの問題は、優先事項として処理し、リソースのそれ以上の不正使用を防ぐために直ちに修復を行うことをお勧めします。例えば、Amazon EC2 インスタンスをクリーンアップまたは終了するか、IAM 認証情報を更新します。「検出結果の修復」の手順に従って、検出結果を修復します。

重大度中

値の範囲: 4.0 - 6.9

説明: 重大度中は、通常観察される動作から逸脱する不審なアクティビティを示し、ユーズケースによってはリソースが侵害されていることを示します。

推奨事項: GuardDuty では、影響を受ける可能性のあるリソースをできるだけ早く調査することをお勧めします。修復手順は、リソースと検出結果ファミリーによって異なります。確立アプローチは、アクティビティが認可され、ユースケースと一致していることを確認することです。原因を特定できない場合やアクティビティが許可されたことを確認できない場合は、リソースが侵害されたとみなす必要があります。「検出結果の修復」の手順に従って、検出結果を修復します。

次に、重大度中の検出結果を確認する場合に考慮する事項を示します。

- 未承認のユーザーがインストールした新しいソフトウェアでリソースの動作が変更されていないか確認してください。例えば、通常より高いトラフィックが許可されている場合や、新しいポートの通信が有効化されている場合などがあります。
- 許可されているユーザーがコントロールプレーンの設定を変更しているかどうかを確認します (セキュリティグループの設定の変更など)。
- 該当するリソースでアンチウィルススキャンを実行し、未承認のソフトウェアを検出します。
- 該当する IAM ロール、ユーザー、グループ、または認証情報セットにアタッチされている許可を検証します。次のアクセス許可を変更または更新する必要がある場合があります。

重大度低

値の範囲: 1.0 - 3.9

説明: 重大度低は、環境が侵害されなかった不審なアクティビティが試行されたことを示します (ポートスキャンや侵入の失敗など)。

推奨事項: すぐに推奨されるアクションはありませんが、この情報は、誰かが環境の弱点を探していることを示している可能性があるので、念のためメモしておいてください。

検出結果の詳細

Amazon GuardDuty コンソールでは、検出結果の概要セクションで検出結果の詳細を見ることができます。検出結果の詳細は検出結果のタイプによって異なります。

検出結果にどのような情報が表示されるかを決める基本的な情報が 2 つあります。1 つ目はリソースタイプです。これは、`Instance`、`AccessKey`、`S3Bucket`、`S3Object`、`Kubernetes`

cluster、ECS cluster、Container、RDSDBInstance、RDSLimitlessDB、または Lambda です。情報の検出結果を決定する 2 つ目の詳細は [リソースロール] です。リソースロールは、Target である可能性があります。つまり、リソースが疑わしいアクティビティのターゲットであったことを意味します。インスタンスタイプの検出結果については、リソースロールが Actor である場合があります。つまり、リソースが不審なアクティビティを実行するアクターだったことを意味します。このトピックでは、検出結果の一般的に入手可能な詳細をいくつか説明します。「the section called “Runtime Monitoring の検出結果タイプ”」および「Malware Protection for S3 の検出結果タイプ」の場合、リソースロールは入力されません。

トピック

- 検出結果の概要
- [リソース]
- 攻撃シーケンスの検出結果の詳細
- カスタム検出ルールの検出結果の詳細
- RDS データベース (DB) ユーザーの詳細
- Runtime Monitoring の検出結果の詳細
- EBS ボリュームのスキャンの詳細
- Malware Protection for EC2 の検出結果の詳細
- Malware Protection for S3 の検出結果の詳細
- アクション
- アクターまたはターゲット
- 位置情報の詳細
- 追加情報
- 証拠
- 異常な動作

検出結果の概要

検出結果の概要のセクションには、次の情報を含む、検索条件の最も基本的な識別機能が含まれています。

- アカウント ID - GuardDuty がこの検出結果を生成する原因となったアクティビティを実行した AWS アカウントのID

- [カウント] - GuardDuty が、このパターンとこの検出結果 ID の一致するアクティビティを集約した回数。
- 作成時刻 - この検出結果が初めて生成された日時。この値が [Updated at] (更新時刻) と異なる場合は、そのアクティビティが複数回発生しており、現在も進行中の問題でありことを示しています。

> **Note**
>
> GuardDuty コンソールの検出結果のタイムスタンプはローカルタイムゾーンで表示されます。一方、JSON エクスポートおよび CLI 出力では UTC 中にタイムスタンプが表示されます。

- [Finding ID] (結果 ID) - この検出結果タイプおよびパラメータセットに対応する一意の識別子です。このパターンに一致するアクティビティが新しく出現した場合は、同じ ID に集約されます。
- [結果タイプ] - 検出結果をトリガーしたアクティビティのタイプを表す、書式設定された文字列。詳細については、「GuardDuty の検出結果の形式」を参照してください。
- リージョン – 結果が生成された AWS リージョン。サポートされるリージョンについては、「リージョンとエンドポイント」を参照してください。
- リソース ID - GuardDuty がこの検出結果を生成する原因となったアクティビティを実行した AWS リソースの ID。
- スキャン ID — GuardDuty Malware Protection for EC2 が有効な場合に検出結果に適用されます。これは、侵害された可能性がある EC2 インスタンスまたはコンテナワークロードにアタッチされた EBS ボリュームで実行されるマルウェアスキャンの識別子です。詳細については、「Malware Protection for EC2 の検出結果の詳細」を参照してください。
- 重大度 - 重大、高、中、低のいずれかで割り当てられた検出結果の重大度。詳細については、「検出結果の重大度レベル」を参照してください。
- [結果タイプ] - GuardDuty がこの検出結果を生成する原因となったパターンに一致する新しいアクティビティにより、この検出結果が最後に更新された時刻。

[リソース]

影響を受けるリソースは、開始アクティビティの対象となった AWS リソースに関する詳細を提供します。利用可能な情報は、リソースタイプとアクションタイプによって異なります。

リソースロール – 検出結果を開始した AWS リソースのロール。値は [TARGET] または [ACTOR] で、それぞれの値は、リソースが疑わしいアクティビティの対象であったか、疑わしいアクティビティを実行したアクターであったことを表します。

リソースタイプ - 該当するリソースのタイプ。複数のリソースが関係していた場合は、複数のリソースタイプが検出結果に含まれる可能性があります。リソースタイプは、Instance、AccessKey、S3Bucket、S3Object、KubernetesCluster、ECSCluster、Container、RDSDBIr および Lambda です。リソースタイプによって、使用可能な検出結果の詳細が異なります。リソースオプションタブを選択して、そのリソースで使用可能な詳細について説明します。

Instance

インスタンスの詳細:

> **Note**
>
> インスタンスが既に停止している場合、またはクロスリージョン API コールを行うときに別のリージョンの EC2 インスタンスから基盤となる API コールが発生した場合、インスタンスの詳細が欠落することがあります。

- インスタンス ID - GuardDuty がその検出結果を生成する原因となったアクティビティに含まれる EC2 インスタンスの ID。
- インスタンスタイプ - 検出結果に関連する EC2 インスタンスのタイプ。
- [起動時刻] - インスタンスが開始された日時
- Outpost ARN – の Amazon リソースネーム (ARN) AWS Outposts。 AWS Outposts インスタンスにのみ適用されます。詳細については、「Outposts ラックのユーザーガイド」の「AWS Outpostsとは」を参照してください。
- [セキュリティグループ名] - 関連するインスタンスに付属するセキュリティグループの名前。
- [セキュリティグループ ID] - 関係するインスタンスに付属するセキュリティグループの ID。
- [インスタンスの状態]- ターゲットインスタンスの現在の状態。
- [アベイラビリティーゾーン] - 関連するインスタンスが配置されている AWS リージョンアベイラビリティーゾーン。
- [イメージ ID] - アクティビティに関連するインスタンスの構築に使用される Amazon マシンイメージの ID。
- [イメージの説明] - アクティビティに関連するインスタンスの構築に使用される Amazon マシンイメージの ID に関する説明。
- [タグ] - このリソースにアタッチされているタグのリスト。リストの形式は `key:value` です。

AccessKey

アクセスキーの詳細:

- アクセスキー ID - GuardDuty がその検出結果を生成する原因となったアクティビティを行ったユーザーのアクセスキー ID。
- [プリンシパル ID] - GuardDuty がその検出結果を生成する原因となったアクティビティを行ったユーザーのプリンシパル ID。
- [ユーザータイプ] - GuardDuty がその検出結果を生成する原因となったアクティビティを行ったユーザーのタイプ。詳細については、「CloudTrail userIdentity 要素」を参照してください。
- [ユーザー名] - GuardDuty がその検出結果を生成する原因となったアクティビティを行ったユーザーの名前。

S3Bucket

Amazon S3 バケットの詳細

- [名前] - 検出結果に関連するバケットの名前。
- [ARN] – 検出結果に関連するバケットの ARN。
- [所有者] - 検出結果に関連するバケットを所有するユーザーの正規ユーザー ID。正規ユーザー ID の詳細については、「AWS アカウント ID」を参照してください。
- [タイプ] - バケット検出結果のタイプで、[送信先] または [ソース] になります。
- デフォルトのサーバー側暗号化 - バケットの暗号化に関する詳細。
- バケットタグ - このリソースにアタッチされたタグのリスト。リストの形式は `key:value` です。
- [有効な許可] - 関連するバケットが公開されているかどうかを示す、バケットに関するすべての有効な許可とポリシーの評価。値は [公開] または [非公開] です。

S3Object

- [S3 オブジェクトの詳細] – スキャンされた S3 オブジェクトに関する以下の情報が含まれます。
  - [ARN] – スキャンされた S3 オブジェクトの Amazon リソースネーム (ARN)。
  - [キー] – S3 バケットでファイルの作成時にファイルに割り当てられた名前。

  - [Version Id] – バケットのバージョニングを有効にすると、このフィールドには、スキャンされた S3 オブジェクトの最新バージョンに関連付けられたバージョン ID が表示されます。詳細については、『Amazon S3 ユーザーガイド』の「S3 バケットでのバージョニングの使用」を参照してください。
  - [eTag] – スキャンされた S3 オブジェクトの特定のバージョンを表します。
  - [ハッシュ] – この検出結果で検出された脅威のハッシュ。
- [S3 バケットの詳細] – スキャンされた S3 オブジェクトに関連付けられた Amazon S3 バケットに関する以下の情報が含まれます。
  - [名前] – オブジェクトが含まれている S3 バケットの名前を示します。
  - [ARN] – S3 バケットの Amazon リソースネーム (ARN)。
- [所有者] – S3 バケット所有者の正規 ID。

EKSCluster

Kubernetes クラスターの詳細:

- 名前 — Kubernetes クラスターの名前。
- ARN — クラスターを識別する ARN。
- 作成時刻 - このクラスターが生成された日時。

> **Note**
>
> GuardDuty コンソールの検出結果のタイムスタンプはローカルタイムゾーンで表示されます。一方、JSON エクスポートおよび CLI 出力では UTC 中にタイムスタンプが表示されます。

- VPC ID — クラスターに関連付けられている VPC ID。
- 状態 – クラスターの現在の状態。
- タグ - クラスターに適用し、クラスターの分類と整理に役立つメタデータ。各タグは、`key:value` 形式でリストされているキーとオプションの値で構成されます。キーと値の両方を定義できます。

  クラスタータグは、クラスターに関連付けられた他のリソースには伝達されません。

Kubernetes ワークロードの詳細:

- タイプ - ポッド、デプロイ、ジョブなどの Kubernetes ワークロードのタイプ。
- 名前 - Kubernetes ワークロードの名前。
- UID - Kubernetes ワークロードの固有の ID。
- 作成時刻 - このワークロードが生成された日時。
- ラベル - Kubernetes ワークロードにアタッチされたキーと値のペア。
- コンテナ - Kubernetes ワークロードの一部として実行されているコンテナの詳細。
- 名前空間 - ワークロードはこの Kubernetes 名前空間に属します。
- ボリューム - Kubernetes ワークロードが使用するボリューム。
  - ホストパス - ボリュームがマッピングされるホストマシン上の既存のファイルまたはディレクトリを示します。
  - 名前 - ボリュームの名前。
- ポッドセキュリティコンテキスト - ポッド内のすべてのコンテナの権限とアクセスコントロール設定を定義します。
- ホストネットワーク - ポッドが Kubernetes ワークロードに含まれている場合は `true` に設定します。

Kubernetes ユーザーの詳細

- グループ — 検出結果を生成したアクティビティに関与したユーザーの Kubernetes RBAC (ロールアクセスベースのコントロール) グループ。
- ID — Kubernetes ユーザーの固有の ID。
- ユーザー名 — 検出結果を生成したアクティビティに関係した Kubernetes ユーザーの名前。
- セッション名 — Kubernetes RBAC アクセス許可を持つ IAM ロールを引き受けたエンティティ。

ECSCluster

ECS クラスターの詳細

- ARN — クラスターを識別する ARN。
- 名前 - クラスターの名前。
- 状態 – クラスターの現在の状態。
- アクティブサービスの数 — `ACTIVE` 状態のクラスター内で実行されているサービスの数。ListServices でこれらのサービスを確認できます。

- 登録されたコンテナインスタンス数 — クラスターに登録されているコンテナインスタンスの数。これには、ACTIVE および DRAINING 状態の両方のコンテナインスタンスが含まれます
- 実行中のタスク数 — RUNNING 状態のクラスターのタスクの数。
- タグ - クラスターに適用し、クラスターの分類と整理に役立つメタデータ。各タグは、key:value 形式でリストされているキーとオプションの値で構成されます。キーと値の両方を定義できます。
- コンテナ – タスクに関連付けられたコンテナの詳細:
  - コンテナ名 – コンテナの名前。
  - コンテナイメージ – コンテナのイメージ。
- タスクの詳細 — クラスター内のタスクの詳細。
  - ARN – タスクの Amazon リソースネーム (ARN)。
  - 定義 ARN – タスクを作成するタスク定義の Amazon リソースネーム(ARN)。
  - バージョン– タスクのバージョンカウンター。
  - タスクの作成時刻 – タスクが作成されたときの Unix タイムスタンプ。
  - タスクの開始時刻 – タスクが開始されたときの Unix タイムスタンプ。
  - タスクの開始ユーザー – タスクの開始時に指定されたタグ。

Container

コンテナの詳細:

- コンテナランタイム — コンテナの実行に使用されたコンテナランタイム (docker または containerd など)。
- ID — コンテナインスタンスのコンテナインスタンス ID または完全な ARN エントリ。
- 名前 — コンテナの名前。
- イメージ — コンテナインスタンスのイメージ。
- ボリュームマウント — コンテナボリュームのマウントのリスト。コンテナは、そのファイルシステムの下にボリュームをマウントできます。
- セキュリティコンテキスト — コンテナセキュリティコンテキストは、コンテナの権限とアクセス制御設定を定義します。
- プロセスの詳細 — 検出結果に関連付けられているプロセスの詳細が記述されます。

RDSDBInstance

RDSDBInstance の詳細:

> **Note**
>
> このリソースは、データベースインスタンスに関連する RDS Protection の検出結果で利用できます。

- [Database Instance ID] (データベースインスタンス ID) — GuardDuty の検出に関与したデータベースインスタンスに関連付けられた識別子
- [Engine] (エンジン) — 検出に関与したデータベースインスタンスのデータベースエンジン名。指定できる値は、Aurora MySQL 互換または Aurora PostgreSQL 互換です。
- [Engine version] (エンジンバージョン) – GuardDuty の検出に関与したデータベースエンジンのバージョン
- [Database cluster ID] (データベースクラスター ID) — GuardDuty の検出に関与したデータベースインスタンス ID を含むデータベースクラスターの識別子
- [データベースインスタンス ARN] — GuardDuty の検出に関与したデータベースインスタンスを識別する ARN

RDSLimitlessDB

RDSLimitlessDB の詳細:

このリソースは、サポートされている Limitless Database のエンジンバージョンに関連する RDS Protection の検出結果で利用可能です。

- DB シャードグループ識別子 – Limitless DB シャードグループに関連付けられた名前。
- DB シャードグループのリソース ID – Limitless DB 内の DB シャードグループのリソース識別子。
- DB シャードグループ ARN – DB シャードグループを識別する Amazon リソースネーム (ARN)。
- エンジン – 検出結果に関連する Limitless DB の識別子。
- エンジンバージョン – Limitless DB エンジンのバージョン。
- DB クラスター識別子 – Limitless DB の一部であるデータベースクラスターの名前。

影響を受ける可能性のあるデータベースのユーザーと認証の詳細については、「RDS データベース (DB) ユーザーの詳細」を参照してください。

Lambda

Lambda 関数の詳細

- 関数名 - 検出結果に含まれた Lambda 関数の名前。
- 関数バージョン - 検出結果に含まれる Lambda 関数のバージョン。
- 関数の説明 - 検出結果に含まれた Lambda 関数の説明。
- 関数 ARN - 検出結果に含まれた Lambda 関数の Amazon リソースネーム (ARN)。
- リビジョン ID - Lambda 関数バージョンのリビジョン ID。
- ロール - 検出結果に関係する Lambda 関数の実行ロール。
- VPC 設定 - Lambda 関数に関連付けられた VPC ID、セキュリティグループ、サブネット ID を含む Amazon VPC 設定。
  - VPC ID - 検出結果に含まれた Lambda 関数に関連付けられた Amazon VPC の ID。
  - サブネット ID - Lambda 関数に関連付けられているサブネットの ID。
  - セキュリティグループ - 関連する Lambda 関数にアタッチされたセキュリティグループ。これには、セキュリティグループ名とグループ ID が含まれます。
- タグ - このリソースにアタッチされているタグのリスト。リストの形式は `key:value` ペアです。

攻撃シーケンスの検出結果の詳細

GuardDuty は、アカウントで生成された各検出結果の詳細を提供します。これらの詳細により、検出結果の背後にある理由を把握することができます。このセクションでは、攻撃シーケンスの検出結果タイプ に関連する詳細に焦点を当てます。これには、影響を受ける可能性のあるリソース、イベントのタイムライン、指標、シグナル、および検出結果に関連するエンドポイントなどのインサイトが含まれます。

GuardDuty の検出結果であるシグナルに関連する詳細を確認するには、このページの関連するセクションを参照してください。

GuardDuty コンソールで攻撃シーケンスの検出結果を選択すると、詳細サイドパネルは以下のタブに分割されます。

- 概要 – 攻撃シーケンスの詳細 (シグナル、MITRE 戦術、影響を受ける可能性のあるリソースなど) をコンパクトに表示します。
- シグナル – 攻撃シーケンスに関連するイベントのタイムラインを表示します。
- リソース – 影響を受ける可能性のあるリソース、またはリスクにさらされている可能性のあるリソースに関する情報を提供します。

以下のリストは、攻撃シーケンスの検出結果の詳細に関連する説明を示しています。

シグナル

シグナルとは、GuardDuty が攻撃シーケンスの検出に使用する API アクティビティまたは検出結果です。GuardDuty は、明確な脅威として現れない微弱なシグナルを考慮し、それらを組み合わせて、個別に生成された検出結果と相関させます。より詳細なコンテキストについては、[シグナル] タブには、GuardDuty が観測したシグナルのタイムラインが表示されます。

GuardDuty の検出結果である各シグナルには、独自の重大度レベルと値が割り当てられます。GuardDuty コンソールでは、各シグナルを選択して、関連する詳細を表示できます。

アクター

攻撃シーケンスの脅威アクターに関する詳細を提供します。詳細については、「Amazon GuardDuty API Reference」の「アクター」を参照してください。

エンドポイント

この攻撃シーケンスで使用されたネットワークエンドポイントの詳細を提供します。詳細については、「Amazon GuardDuty API リファレンス」の「NetworkEndpoint」を参照してください。GuardDuty が場所を決定する方法については、「位置情報の詳細」を参照してください。

インジケータ

セキュリティ問題のパターンに一致する観察データが含まれます。このデータは、GuardDuty が疑わしいアクティビティの可能性を示す理由を指定します。例えば、インジケータ名が の場合`HIGH_RISK_API`、これは脅威アクターが一般的に使用するアクション、または認証情報へのアクセスやリソースの変更など AWS アカウント、 に影響を与える可能性のある機密アクションを示します。

以下の表に、潜在的な指標とその説明のリストを示します。

インジケータ名	説明
ATTACK_TACTIC	攻撃シーケンスで脅威アクターが使用する MITRE 戦術。例としては、Discovery や Impact などがあります。
ATTACK_TECHNIQUE	攻撃シーケンスで脅威アクターが使用した MITRE 手法。例としては、リソースへのアクセス権を取得して意図しない方法で使用すること、および脆弱性の悪用することなどがあります。
CRYPTOMINING_DOMAIN	暗号通貨マイニングプールやインフラストラクチャに関連付けられたドメイン名を示します。例えば、コンテナまたは Kubernetes 環境からの DNS クエリまたはこれらのドメインへの接続は、不正な暗号化アクティビティを示している可能性があります。
CRYPTOMINING_IP	暗号通貨マイニングプールまたはインフラストラクチャに関連付けられた IP アドレスを示します。例えば、コンテナまたは Kubernetes 環境からこれらのアドレスへの接続は、不正な暗号化アクティビティを示している可能性があります。
CRYPTOMINING_PROCESS	コンテナや Kubernetes 環境内で実行されている暗号通貨マイニングソフトウェアとして識別されるプロセスを示します。例えば、これらのプロセスは過剰な CPU リソースを消費することがあります。
HIGH_RISK_API	AWS のサービス 名前を含み、脅威アクターが一般的に使用するアクションeventName を示す AWS API、または認証情報へのアクセスやリソースの変更など AWS アカウント、 に影響を与える可能性のある機密アクションです。
MALICIOUS_DOMAIN	悪意のある意図を示す脅威インテリジェンスが疑われるドメイン名を示します。例えば、これには、コマンドアンドコントロール (C&C) サーバー、マルウェア分散サイト、またはコンテナや Kubernetes 環境から問い合わせたフィッシングドメインが含まれます。
MALICIOUS_FILE	脅威インテリジェンスまたは動作分析に基づいて、悪意のある疑いのあるファイルを示します。たとえば、これには既知のマルウェアバイナリ、悪意のあるスクリプト、不正な実行可能ファイルが含まれます。

インジケータ名	説明
MALICIOUS_IP	その IP アドレスは、悪意のある意図を示す脅威インテリジェンスを確認済です。
MALICIOUS_PACKAGE	攻撃シーケンスに関係するリソースに、悪意のあるソフトウェアパッケージが含まれていることを示します。インジケータ値には、検出されたパッケージに関連付けられた OSV 指定 (MAL-*) を使用した悪意のあるパッケージ識別子が含まれます。
MALICIOUS_PROCESS	脅威インテリジェンスや行動分析に基づいて、悪意のあると疑われるプロセスを示します。たとえば、これには、既知のマルウェア、バックドア、またはインスタンスまたはコンテナ環境内で実行される不正なツールが含まれます。
MISCONFIGURATION	攻撃シーケンスに関係するリソースに、潜在的な影響を高める可能性のあるセキュリティ設定ミスがあることを示します。例としては、管理アクセスポリシー、無効なバージョニング、暗号化の欠落などがあります。これらの設定ミスは によって識別されます AWS Security Hub CSPM。詳細については、「 AWS Security Hub CSPM ユーザーガイド」の「サポートされている特性タイプ」を参照してください。
REACHABILITY	攻撃シーケンスに関係するリソースがインターネットから到達可能であるか、パブリックにアクセス可能であることを示します。例としては、インターネットにアクセスできる Amazon EC2 インスタンス、Amazon EKS クラスター、Amazon ECS クラスター、パブリックにアクセス可能な Amazon S3 バケットなどがあります。詳細については、「 AWS Security Hub CSPM ユーザーガイド」の「サポートされている特性タイプ」を参照してください。
SENSITIVE_DATA	攻撃シーケンスに関係する Amazon S3 バケットに、Amazon Macie によって識別される機密データが含まれていることを示します AWS Security Hub CSPM。
SUSPICIOUS_NETWORK	そのネットワークは、リスクの高い仮想プライベートネットワーク (VPN) プロバイダーやプロキシサービスなど、既知の低評価スコアに関連付けられています。

インジケータ名	説明
SUSPICIOUS_PROCESS	脅威インテリジェンスまたは動作分析に基づいて異常として識別された疑わしいプロセスを示します。たとえば、これには、異常なプロセス実行パターン、予期しない親子関係、インスタンス内の疑わしい特性を持つプロセス、またはコンテナ環境が含まれます。
SUSPICIOUS_USER_AGENT	ユーザーエージェントは、Amazon S3 クライアントや攻撃ツールなど、潜在的に既知の不審または悪用されたアプリケーションに関連付けられています。
TOR_IP	IP アドレスは Tor 終了ノードに関連付けられています。
UNUSUAL_API_FOR_ACCOUNT	アカウントの履歴ベースラインに基づいて API が異常に AWS 呼び出されたことを示します。詳細については、「異常な動作」を参照してください。
UNUSUAL_ASN_FOR_ACCOUNT	アカウントの履歴ベースラインに基づいて、AS 番号 (ASN) が異常として識別されたことを示します。詳細については、「異常な動作」を参照してください。
UNUSUAL_ASN_FOR_USER	ユーザーの履歴ベースラインに基づいて、AS 番号 (ASN) が異常として識別されたことを示します。詳細については、「異常な動作」を参照してください。
VULNERABILITY	攻撃シーケンスに関係するリソースに、重大度が重大、高、中、または低の既知の脆弱性があることを示します。インジケータ値には、CVE 識別子とその重要度レベルが含まれます。これらの脆弱性は Amazon Inspector によって検出され、攻撃シーケンスの潜在的な影響が増加する可能性があります。重要度レベルの詳細については、Amazon Inspector ユーザーガイド」の「重要度レベルの理解」を参照してください。

MITRE の戦術

このフィールドは、脅威アクターが攻撃シーケンスを通じて試みる MITRE ATT&CK 戦術を指定します。GuardDuty は、攻撃シーケンス全体にコンテキストを追加する MITRE ATT&ACK フレームワークを使用します。GuardDuty コンソールが脅威アクターによって使用された脅威の目

的を指定するために使用する色は、重大、高、中、低の 検出結果の重大度レベル を示す色と一致します。

ネットワークインジケータ

インジケータには、ネットワークが疑わしい動作を示す理由を説明する、ネットワークインジケータ値の組み合わせが含まれます。このセクションは、インジケータに SUSPICIOUS_NETWORK または MALICIOUS_IP が含まれている場合にのみ適用されます。以下の例は、ネットワークインジケータをインジケータに関連付ける方法を示しています。

- *AnyCompany* は自律システム (AS) です。
- TUNNEL_VPN、IS_ANONYMOUS、ALLOWS_FREE_ACCESS はネットワークインジケータです。

```
...{
    "key": "SUSPICIOUS_NETWORK",
    "values": [{
        "AnyCompany": [
            "TUNNEL_VPN",
            "IS_ANONYMOUS",
            "ALLOWS_FREE_ACCESS"
        ]
    }]
}
...
```

以下の表に、ネットワークインジケータの値とその説明を示します。これらのタグは、GuardDuty が Spur などのソースから収集する脅威インテリジェンスに基づいて追加されます

ネットワークインジケータ値	説明
TUNNEL_VPN	ネットワークまたは IP アドレスは VPN トンネルタイプに関連付けられています。これは、パブリックネットワーク上で 2 つの地点間の安全で暗号化された接続を確立するのに役立つ特定のプロトコルを指します。
TUNNEL_PROXY	ネットワークまたは IP アドレスは Proxy トンネルタイプに関連付けられています。これは、プロキシサーバーを通じて接続を確立するのに役立つ特定のプロトコルを指します。

ネットワークインジケータ値	説明
TUNNEL_RDP	ネットワークまたは IP アドレスは、リモートデスクトップ (RDP) トラフィックを別のプロトコル内にカプセル化する方法を用いることに関連しており、これにより、セキュリティを強化、ネットワーク制限を回避し、ファイアウォールを介したリモートアクセスを可能にします。
IS_ANONYMOUS	ネットワークまたは IP アドレスは、既知の匿名サービスまたはプロキシサービスに関連付けられています。これは、匿名ネットワークの背後に潜む潜在的な不審なアクティビティを示している可能性があります。
KNOWN_THREAT_OPERATOR	ネットワークまたは IP アドレスは、既知のリスクの高いトンネルプロバイダーに関連付けられています。これは、悪意のある目的で頻繁に使用される VPN、プロキシ、またはその他のトンネリングサービスにリンクされている IP アドレスから不審なアクティビティが検出されたことを示します。
ALLOWS_FREE_ACCESS	ネットワークまたは IP アドレスが、認証や支払いを必要とせずに、そのサービスへのアクセスを許可するトンネルオペレーターに関連付けられています。また、さまざまなオンラインサービスが提供されるトライアルアカウントや、利用制限付きの体験が含まれる場合もあります。
ALLOWS_CRYPTO	ネットワークまたは IP アドレスは、支払い方法として暗号通貨やその他のデジタル通貨のみを受け入れるトンネルプロバイダー (VPN やプロキシサービスなど) に関連付けられています。
ALLOWS_TORRENTS	ネットワークまたは IP アドレスは、トレントトラフィックを許可するサービスまたはプラットフォームに関連付けられています。このようなサービスは、しばしばトレントの利用やサポート、および著作権回避活動に関連付けられています。

ネットワークインジケータ値	説明
RISK_CALLBACK_PROXY	ネットワークまたは IP アドレスは、住宅用プロキシ、マルウェアプロキシ、またはその他のコールバックプロキシ型のネットワークにトラフィックをルーティングすることが知られているデバイスに関連付けられています。これは、ネットワーク上のすべてのアクティビティがプロキシ関連であることを意味するのではなく、ネットワークにこれらのプロキシネットワークに代わってトラフィックをルーティングする機能があることを意味します。
RISK_GEO_MISMATCH	このインジケータは、ネットワークのデータセンターまたはホスティング場所が、その背後にあるユーザーとデバイスの想定される場所と異なることを示唆しています。このインジケータ値が存在しない場合でも、不一致がないという意味ではありません。これは、その不一致を確認するためのデータが不十分であることを示している可能性があります。
IS_SCANNER	ネットワークまたは IP アドレスは、ウェブフォームに対する持続的なログイン試行の実行に関連付けられています。
RISK_WEB_SCRAPING	IP アドレスのネットワークは、自動ウェブクライアントおよびその他のプログラムによるウェブアクティビティに関連付けられています。
CLIENT_BEHAVIOR_FILE_SHARING	ネットワークまたは IP アドレスは、peer-to-peer (P2P) ネットワークやファイル共有プロトコルなどのファイル共有アクティビティを示すクライアントの動作に関連付けられています。
CATEGORY_COMMERCIAL_VPN	ネットワークまたは IP アドレスは、データセンタースペース内で運用されている従来の商用仮想プライベートネットワーク (VPN) サービスに分類されるトンネルオペレーターに関連付けられています。
CATEGORY_FREE_VPN	ネットワークまたは IP アドレスは、完全に無料の VPN サービスに分類されるトンネルオペレーターに関連付けられています。
CATEGORY_RESIDENTIAL_PROXY	ネットワークまたは IP アドレスは、SDK、マルウェア、または get-paid-to-sourced プロキシサービスに分類されるトンネルオペレーターに関連付けられています。

ネットワークインジケータ値	説明
OPERATOR_XXX	このトンネルを運用しているサービスプロバイダーの名前。

カスタム検出ルールの検出結果の詳細

カスタム検出ルールの検出結果の詳細は、それを生成したルールのデータソースによって異なります。

AWS CloudTrail ルールからの結果

AWS CloudTrail ルールの結果は、 と同じService.Detection.Sequence構造を使用します攻撃シーケンスの検出結果の詳細。アクター、エンドポイント、リソース、シグナルの各フィールドは、同じスキーマを共有します。以下の詳細は、カスタム検出ルールに固有のものです。

タイプ

検出結果タイプは の形式に従いますTactic:Service/Technique。例えば、Persistence:EC2/ExternalRemoteServices。

シグナル

各シグナルは 1 つの一致ルールイベントを表します。Name フィールドにはカスタム検出ルール名が含まれ、MITRE ATT&CK® 戦術と手法SignalIndicatorsが含まれています。ルールが集約ウィンドウ内の複数のイベントと一致すると、同じルールを持つ複数のシグナルを表示できます。

アクター

検出されたアクティビティを実行した IAM プリンシパルに関する詳細。ユーザータイプ (AssumedRoleや などIAMUser)、認証情報、発行ロール ARN や MFA ステータスを含むセッションの詳細が含まれます。

リソース

Amazon EC2 インスタンス、Amazon S3 バケット、IAM アクセスキーなど、検出されたアクティビティに関連する AWS リソースの詳細。

エンドポイント

アクティビティの発生元のネットワークエンドポイントに関する詳細。IP アドレス、地理的位置 (都市と国) 、自律システム (AS) の名前と番号が含まれます。

Runtime Monitoring ルールの結果

Runtime Monitoring ルールの結果は、関連するプロセスのランタイムの詳細など、他の Runtime Monitoring の結果と同じ構造を使用します。詳細については、「Runtime Monitoring の検出結果の詳細」を参照してください。以下の詳細は、カスタム検出ルールに固有のものです。

タイプ

検出結果タイプは の形式に従いますTactic:Runtime/Technique。例えば、Persistence:Runtime/SuspiciousCommand。

ルール

CustomDetectionRuleName フィールドには、検出結果を生成したカスタム検出ルールの名前が含まれます。

Context

Service.RuntimeDetails.Context フィールドは、一致したアクティビティを記述します。結果に含まれるフィールドは、ルールによって異なります。

- SensitiveFileRead および SensitiveFileModified の結果を生成するルールには、FilePathおよび が含まれますFileOperation。
- SuspiciousCommand 検出結果を生成するほとんどのルールには が含まれCommandLineExample、一部のルールには も含まれますServiceName。ファイルのアクセス許可の変更など、特定のファイルのアクティビティを検出するルールには、FilePath代わりに と FileOperation が含まれます。
- SuspiciousTool 検出結果を生成するルールには、 ToolNameと が含まれますToolCategory。

[リソース]

各検出結果は、インスタンス、タスク、コンテナなどの 1 つのリソースを対象としています。GuardDuty がシグナルをグループ化する方法の詳細については、「」を参照してください結果の集約。

RDS データベース (DB) ユーザーの詳細

> **Note**
>
> このセクションは、GuardDuty で RDS Protection 機能を有効にしたときの検出結果に適用されます。詳細については、「GuardDuty RDS Protection」を参照してください。

GuardDuty の検出結果から、侵害の可能性があるデータベースに関する、以下のユーザー情報と認証情報が得られます。

- [User] (ユーザー) - 異常なログイン試行に使用されたユーザー名
- [Application] (アプリケーション) — 異常なログイン試行に使用されたアプリケーション名
- [Database] (データベース) — 異常なログイン試行に関与したデータベースインスタンスの名前
- [SSL] — ネットワークに使用された Secure Socket Layer (SSL) のバージョン
- [認証方法] — 検出に関与したユーザーが使用した認証方法

侵害された可能性のあるリソースの詳細については、「[リソース]」を参照してください。

Runtime Monitoring の検出結果の詳細

> **Note**
>
> これらの詳細を確認できるのは、GuardDuty が次のいずれかの GuardDuty Runtime Monitoring の検出結果タイプ を生成した場合だけです。

このセクションには、プロセスの詳細や必要なコンテキストなど、ランタイムの詳細が含まれています。プロセスの詳細には、観察されたプロセスに関する情報が記述され、ランタイムのコンテキストには、潜在的に疑わしいアクティビティに関する追加情報が記述されます。

プロセスの詳細

- 名前 - プロセスの名前。
- 実行可能ファイルのパス - プロセスの実行可能ファイルの絶対パス。
- 実行可能ファイル SHA-256 - プロセスの実行可能ファイルの SHA256 ハッシュ。

- 名前空間 PID - ホストレベルの PID 名前空間以外のセカンダリ PID 名前空間内のプロセスのプロセス ID。コンテナ内のプロセスの場合、コンテナ内で確認されるプロセス ID です。
- 現在の作業ディレクトリ - プロセスの現在の作業ディレクトリ。
- プロセス ID - オペレーティングシステムによってプロセスに割り当てられた ID。
- startTime - プロセスが開始された時間。これは UTC 日付文字列形式 (`2023-03-22T19:37:20.168Z`) です。
- UUID - GuardDuty で、プロセスに割り当てられた一意の ID。
- 親 UUID - 親プロセスの固有の ID。この ID は GuardDuty によって親プロセスに割り当てられます。
- ユーザー - プロセスを実行したユーザー。
- ユーザー ID - プロセスを実行したユーザーの ID。
- 有効なユーザー ID - イベント発生時のプロセスの有効な実効ユーザー ID。
- 系列 - プロセスの先祖に関する情報。
  - プロセス ID - オペレーティングシステムによってプロセスに割り当てられた ID。
  - UUID - GuardDuty で、プロセスに割り当てられた一意の ID。
  - 実行可能ファイルのパス - プロセスの実行可能ファイルの絶対パス。
  - 有効なユーザー ID - イベント発生時のプロセスの有効な実効ユーザー ID。
  - 親 UUID - 親プロセスの固有の ID。この ID は GuardDuty によって親プロセスに割り当てられます。
  - 開始時間 - プロセスが開始された時間。
  - 名前空間 PID - ホストレベルの PID 名前空間以外のセカンダリ PID 名前空間内のプロセスのプロセス ID。コンテナ内のプロセスの場合、コンテナ内で確認されるプロセス ID です。
  - ユーザー ID - プロセスを実行したユーザーのユーザー ID。
  - 名前 - プロセスの名前。

ランタイムのコンテキスト

次のフィールドから、生成された検出結果には、その検出結果タイプに関連するフィールドのみが含まれる場合があります。

- マウントソース - コンテナによってマウントされたホスト上のパス。
- マウントターゲット - ホストディレクトリにマッピングされているコンテナ内のパス。
- ファイルシステムタイプ - マウントされたファイルシステムのタイプを示します。

- フラグ - この検出結果に関係するイベントの動作をコントロールするオプションを示します。
- プロセスの変更 - 実行時にコンテナ内でバイナリ、スクリプト、またはライブラリを作成または変更したプロセスに関する情報。
- 修正日時 - 実行時にプロセスがコンテナ内のバイナリ、スクリプト、またはライブラリを作成または変更したときのタイムスタンプ。このフィールドは、UTC 日付文字列形式 (`2023-03-22T19:37:20.168Z`)です。
- ライブラリパス - ロードされた新しいライブラリへのパス。
- LD プリロード値 - `LD_PRELOAD` 環境変数の値。
- ソケットパス - アクセスされた Docker ソケットへのパス。
- Runc バイナリパス - `runc` バイナリへのパス。
- リリースエージェントパス - `cgroup` リリースエージェントファイルへのパス。
- コマンドラインの例 – 潜在的に疑わしいアクティビティに関連するコマンドラインの例。
- ツールカテゴリ – ツールが属するカテゴリ。この例としては、Backdoor Tool、Pentest Tool、Network Scanner、Network Sniffer などがあります。
- ツール名 – 潜在的に疑わしいツールの名前。
- ファイルオペレーション – Write、Delete、Rename、Link、Symlink など、検出結果をトリガーしたファイルオペレーションのタイプ。
- ファイルパス – 変更された機密ファイルのパス。変更には、書き込み、削除、名前変更、リンク、シンボリックリンクオペレーションが含まれます。
- 関連ファイルパス – 検出結果をトリガーしたのと同じプロセスによって変更されたすべてのファイルパス。最大 25 個のパス。
- スクリプトパス – 検出結果を生成した実行済みスクリプトへのパス。
- 脅威ファイルパス – 脅威インテリジェンスの詳細が見つかった疑わしいパス。
- サービス名 – 無効化されたセキュリティサービスの名前。
- モジュール名 – カーネルにロードされるモジュールの名前。
- モジュール SHA256 – モジュールの SHA256 ハッシュ。
- モジュールファイルパス – カーネルにロードされたモジュールへのパス。

EBS ボリュームのスキャンの詳細

> **Note**
>
> このセクションは、Malware Protection for EC2 で GuardDuty 実行型マルウェアスキャンをオンにした場合の検出結果に適用されます。

EBS ボリュームスキャンは、侵害された可能性のある EC2 インスタンスまたはコンテナワークロードにアタッチされた EBS ボリュームに関する詳細を提供します。

- スキャン — マルウェアスキャンの識別子。
- スキャン開始日 — マルウェアスキャンが開始された日時。
- スキャン完了日 — 不正プログラムのスキャンの完了日時。
- 検出結果 ID — このマルウェアスキャンを開始した GuardDuty の検出結果 ID。
- ソース – 想定される値は、`Bitdefender` と `Amazon` です。

  マルウェアの検出に使用されるスキャンエンジンの詳細については、「GuardDuty マルウェア検出のスキャンエンジン」を参照してください。
- スキャン検出 — 各マルウェアスキャンの詳細と結果の全情報。
  - スキャンされたアイテム数 — スキャンされたファイルの合計数。`totalGb`、`files`、および `volumes` などの詳細を提供します。
  - 脅威が検出されたアイテム数 — スキャン中に検出された悪意のある `files` の合計数。
  - 最も重大度の高い脅威の詳細 — スキャン中に検出された、重大度が最も高い脅威の詳細と、悪意のあるファイルの数。`severity`、`threatName`、および `count` などの詳細を提供します。
  - 名前で検出された脅威 — すべての重大度レベルの脅威をグループ化するコンテナ要素。`itemCount`、`uniqueThreatNameCount`、`shortened`、および `threatNames` などの詳細を提供します。

Malware Protection for EC2 の検出結果の詳細

> **Note**
>
> このセクションは、Malware Protection for EC2 で GuardDuty 実行型マルウェアスキャンをオンにした場合の検出結果に適用されます。

Malware Protection for EC2 スキャンでマルウェアが検出された場合は、https://console.aws.amazon.com/guardduty/ コンソールで「検出結果」のページで対応する結果を選択すると、スキャンの詳細を表示できます。Malware Protection for EC2 の検出結果の重大度は、GuardDuty の検出結果の重大度によります。

次の情報は、詳細パネルの「検出された脅威」セクションでご覧になれます。

- 名前 — 検出によってファイルをグループ化することによって取得された脅威の名前。
- 重大度 — 検出された脅威の重大度。
- ハッシュ – ファイルの SHA-256 ハッシュ。
- ファイルパス — EBS ボリューム内の悪意のあるファイルの場所。
- ファイル名 — 脅威が検出されたファイルの名前。
- ボリューム ARN — スキャンされた EBS ボリュームの ARN。

次の情報は、詳細パネルの「マルウェアスキャンの詳細」のセクションでご覧になれます。

- スキャン — 不正プログラムスキャンのスキャン ID。
- スキャン開始日 — スキャンが開始された日時。
- スキャン完了日 — スキャンの完了日時。
- スキャンされたファイル — スキャンされたファイルとディレクトリの合計数。
- スキャンされた合計 GB — プロセス中にスキャンされたストレージの容量。
- 検出結果 ID — このマルウェアスキャンを開始した GuardDuty の検出結果 ID。
- 次の情報は、詳細パネルの「ボリュームの詳細」のセクションでご覧になれます。
  - ボリューム ARN — ボリュームの Amazon リソースネーム (ARN)。
  - SnapshotARN — EBS ボリュームのスナップショットの ARN。
  - ステータス — `Running`、`Skipped`、および `Completed` などのボリュームのスキャン状態。
  - 暗号化タイプ — ボリュームの暗号化に使用される暗号化のタイプ。例えば、`CMCMK`。
  - デバイス名 - デバイスの名前。例えば、`/dev/xvda`。

Malware Protection for S3 の検出結果の詳細

AWS アカウントで GuardDuty と Malware Protection for S3 の両方を有効にすると、次のマルウェアスキャンの詳細を利用できます。

- 脅威 – マルウェアスキャン中に検出された脅威のリスト。

  > **アーカイブファイル内の複数の潜在的な脅威**
  >
  > 潜在的な脅威が複数含まれるアーカイブファイルがある場合、Malware Protection for S3 は最初に検出された脅威のみを報告します。この後、スキャンステータスは完了とマークされます。GuardDuty は、関連付けられた検出結果タイプを生成し、生成した EventBridge イベントも送信します。EventBridge イベントを使用した Amazon S3 オブジェクトスキャンのモニタリングの詳細については、「S3 オブジェクトスキャンの結果」の「THREATS_FOUND の通知スキーマの例」を参照してください。

- 項目パス – スキャンされた S3 オブジェクトのネストされた項目パスとハッシュの詳細のリスト。
  - ネストされた項目パス – 脅威が検出されたスキャンされた S3 オブジェクトの項目パス。

    このフィールドの値は、最上位レベルのオブジェクトがアーカイブであり、アーカイブ内で脅威が検出された場合にのみ使用できます。
  - [ハッシュ] – この検出結果で検出された脅威のハッシュ。

    > **Note**
    >
    > ハッシュを計算できない場合でも、スキャン結果は配信され、ハッシュ値はデフォルトで空の文字列 () の SHA-256 になりますe3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855。

- ソース – 想定される値は、`Bitdefender` と `Amazon` です。

  マルウェアの検出に使用されるスキャンエンジンの詳細については、「GuardDuty マルウェア検出のスキャンエンジン」を参照してください。

アクション

検出結果の [Action] (アクション) は、その検出をトリガーしたアクティビティのタイプに関する詳細を示します。利用可能な情報は、アクションタイプによって異なります。

[アクションタイプ] - 検出結果アクティビティのタイプ。この値は、NETWORK_CONNECTION、PORT_PROBE、DNS_REQUEST、AWS_API_CALL、RDS_LOGIN_ATTのいずれかになります。利用可能な情報は、アクションタイプによって異なります。

- NETWORK_CONNECTION - ネットワークトラフィックが識別済み EC2 インスタンスとリモートホスト間で交わされたことを示します。このアクションタイプには、次の追加情報が含まれています。
  - [接続方向] - GuardDuty がその検出結果を生成する原因となったアクティビティで確認されたネットワークの接続方向。これには、次のいずれかの値を指定できます。
    - インバウンド - リモートホストがアカウントの識別済み EC2 インスタンスのローカルポートへの接続を開始したことを示します。
    - アウトバウンド - 識別済み EC2 インスタンスがリモートホストへの接続を開始したことを示します。
    - 不明 - GuardDuty が接続の方向を判別できなかったことを示します。
  - プロトコル - GuardDuty がその検出結果を生成する原因となったアクティビティで確認されたネットワークの接続プロトコル。
  - ローカル IP (ローカル IP) - 検出結果をトリガーしたトラフィックの元の送信元 IP アドレス。この情報を使用して、トラフィックが通過する中間レイヤーの IP アドレスと、検出結果をトリガーしたトラフィックの元の送信元 IP アドレスを区別します。例えば、EKS ポッドが実行されているインスタンスの IP アドレスではなく、EKS ポッドの IP アドレスです。
  - [ブロック] - ターゲットポートがブロックされているかどうかを示します。
- PORT_PROBE - リモートホストが複数のオープンポートで識別済みの EC2 インスタンスを調査したことを示します。このアクションタイプには、次の追加情報が含まれています。
  - ローカル IP - 検出結果をトリガーしたトラフィックの元の送信元 IP アドレス。この情報を使用して、トラフィックが通過する中間レイヤーの IP アドレスと、検出結果をトリガーしたトラフィックの元の送信元 IP アドレスを区別します。例えば、EKS ポッドが実行されているインスタンスの IP アドレスではなく、EKS ポッドの IP アドレスです。
  - [ブロック] - ターゲットポートがブロックされているかどうかを示します。
- DNS_REQUEST - 識別済みの EC2 インスタンスがドメイン名を照会したことを示します。このアクションタイプには、次の追加情報が含まれています。
  - プロトコル - GuardDuty がその検出結果を生成する原因となったアクティビティで確認されたネットワークの接続プロトコル。
  - [ブロック] - ターゲットポートがブロックされているかどうかを示します。
- AWS_API_CALL - AWS API が呼び出されたことを示します。このアクションタイプには、次の追加情報が含まれています。
  - API - 呼び出された API オペレーションの名前で、GuardDuty がこの検出結果を生成する原因となった API。

> **Note**
>
> これらのオペレーションは、 AWS CloudTrailによってキャプチャされる API 以外のイベントを含めることもできます。詳細については、「CloudTrail によってキャプチャされる API 以外のイベント」を参照してください。

  - [ユーザーエージェント] – API リクエストを実行したユーザーエージェント。この値は、呼び出しが 、 AWS サービス AWS マネジメントコンソール、 AWS SDKs、または のいずれから行われたかを示します AWS CLI。
  - エラーコード - API コールの失敗によって検出結果がトリガーされた場合、そのコールに対してエラーコードが表示されます。
  - サービス名 - 検出結果をトリガーした API コールを実行しようとしたサービスの DNS 名。
- RDS_LOGIN_ATTEMPT — 侵害された可能性のあるデータベースに、リモート IP アドレスからログインが試行されたことを示します。
  - [IP アドレス] — 疑わしいログイン試行に使用されたリモート IP アドレス。

アクターまたはターゲット

[リソースロール] が TARGET の場合には、検出結果に [アクター] セクションが表示されます。これは、リソースが疑わしいアクティビティの対象であったことを示します。また、[Actor] (アクター) セクションには、リソースを対象としたエンティティの詳細が含められます。

[リソースロール] が ACTOR の場合には、検出結果に [ターゲット] セクションが表示されます。これは、リソースがリモートホストに対する不審なアクティビティに関与していたことを示します。また、このセクションには、リソースのターゲットになった IP またはドメインに関する情報が含められます。

[アクター] セクションまたは [ターゲット] セクションには、次の情報を含めることができます。

- [Affiliated] (関連) – リモート API コール実行者の AWS アカウントが GuardDuty 環境に関連しているかどうかの詳細です。この値が `true` の場合、API コール実行者は何らかの形でアカウントに関連しています。`false` の場合、API コール実行者は環境外です。
- [リモートアカウント ID] – 最終的なネットワークでリソースにアクセスするために使用されたアウトバウンド IP アドレスを所有するアカウント ID。
- IP アドレス - GuardDuty がその検出結果を生成する原因となったアクティビティに関係のある IP アドレス。

- 場所 - GuardDuty がその検出結果を生成する原因となったアクティビティに関係のある IP アドレスの位置情報。
- [組織] - GuardDuty がその検出結果を生成する原因となったアクティビティに関係のある IP アドレスの ISP 組織情報。
- ポート - GuardDuty がその検出結果を生成する原因となったアクティビティに関係のあるポート番号。
- ドメイン - GuardDuty がその検出結果を生成する原因となったアクティビティに関係のあるドメイン。
- サフィックス付きのドメイン — GuardDuty がその検出結果を生成する原因となった可能性があるアクティビティに関連する 2 番目および最上位レベルのドメイン。最上位ドメインとセカンドレベルドメインのリストについては、「public suffix list」を参照してください。

位置情報の詳細

GuardDuty が MaxMind GeoIP データベースを使用してリクエストの場所とネットワークを決定します。MaxMind は、国レベルでの非常に高精度なデータを報告しますが、精度は国や IP アドレスの種類などの要因によって異なります。

MaxMind の詳細については、「MaxMind IP Geolocation」を参照してください。GeoIP データに誤りがあると思われる場合は、[MaxMind Correct GeoIP2 Data] ページから MaxMind に修正リクエストを送信できます。

追加情報

すべての検出結果には [追加情報] セクションがあり、次のような情報が含まれます。

- 脅威リスト名 - GuardDuty がこの検出結果を生成する原因となったアクティビティに関する IP アドレスまたはドメイン名が含まれている脅威リストの名前。
- サンプル - サンプル検出結果であるかどうかを示す true または false の値。
- アーカイブ - 検出結果がアーカイブ済みかどうかを示す true または false の値。
- [異常] - 履歴で確認されていないアクティビティの詳細 これらには、異常な (以前に確認されていない) ユーザー、場所、時間、バケット、ログイン動作、または ASN Org などが含まれます。
- 異常プロトコル - GuardDuty がその検出結果を生成する原因となったアクティビティに関係のあるネットワークの接続プロトコル。
- エージェント詳細 - AWS アカウントの EKS クラスターに現在導入されているセキュリティエージェントに関する詳細。これは EKS Runtime Monitoring の検出結果タイプにのみ適用されます。

- エージェントバージョン - GuardDuty セキュリティエージェントのバージョン。
- エージェント ID - GuardDuty セキュリティエージェントの固有識別子。

証拠

脅威インテリジェンスに基づく検出結果には [証拠] セクションがあり、次のような情報が含まれます。

- 脅威インテリジェンスの詳細 - 認識された `Threat name` が表示される脅威リストの名前。
- 脅威名 - 脅威に関連付けられているマルウェアファミリーの名前、またはその他の識別子。
- 脅威ファイル SHA256 – 検出結果を生成したファイルの SHA256。

異常な動作

AnomalousBehavior で終わる検出結果タイプは、GuardDuty 異常検出機械学習 (ML) モデルによって検出されたことを示します。機械学習モデルは、アカウントへのすべての API リクエストを評価し、相手が使用するタクティクスに関連する異常なイベントを特定します。機械学習モデルは、リクエストを行ったユーザー、リクエストが行われた場所、リクエストされた特定の API など、API リクエストのさまざまな要因を追跡します。

CloudTrail ユーザーアイデンティティ において、API リクエストの原因となった、異常のある要因についての詳細は、「検出結果の詳細」を参照してください。アイデンティティは、CloudTrail userIdentity エレメントで定義され、指定できる値は、`Root`、`IAMUser`、`AssumedRole`、`FederatedUser`、`AWSAccount` または `AWSService` です。

API アクティビティに関連付けられているすべての GuardDuty の検出結果で使用できる詳細に加えて、AnomalousBehavior 検出結果には、次のセクションで概説される追加の詳細が含まれています。これらの詳細はコンソールで表示でき、検出結果の JSON でも確認できます。

- 異常な API - 検出結果に関連付けられたプライマリ API リクエストに近いユーザーアイデンティティが原因の API リクエストのリスト。このペインでは、次の方法で API イベントの詳細をさらに分類します。
  - 最初にリストされている API はプライマリ API です。プライマリ API は、観測された最も高いリスクアクティビティに関連付けられた API リクエストです。これは、発見をトリガーし、検出結果タイプの攻撃ステージと相関する API です。これは、コンソールの [アクション] セクションおよび検出結果の JSON で詳述されている API でもあります。

- リストされている他の API は、プライマリ API の近くで観察されるリストされたユーザーアイデンティティからの追加の異常 API です。リストに API が 1 つしかない場合、機械学習モデルはそのユーザーアイデンティティからの追加の API リクエストを異常として識別しませんでした。
- API のリストは、API が正常に呼び出されたかどうかに基づいて分類され、API が正常に呼び出されなかった場合は、エラーレスポンスが受信されたことを意味します。受信したエラーレスポンスのタイプは、それぞれ正常に呼び出されなかった API の上に一覧表示されます。考えられるエラーレスポンスのタイプは、`access denied`、`access denied exception`、`auth failure`、`instance limit exceeded`、`invalid permission - duplicate`、`invalid permission - not found`、および `operation not permitted` です。
- API は、関連するサービスによって分類されます。
- その他のコンテキストについては、[Historical APIs] (過去の API) を選択し、上位 API の詳細を確認します。通常は、ユーザーアイデンティティとアカウント内のユーザーすべての両方で表示される最大 20 個の API です。API は、[めったにない(月に 1 回未満)]、[頻繁でない(月に数回)]、または [頻繁(毎日から毎週)] でマークされ、アカウント内で使用されている頻度によって異なります。

- [Unusual Behavior (Account)] (異常な動作 (アカウント)) - このセクションでは、アカウントでプロファイリングされた動作に関する詳細について説明します。

> **プロファイルされた動作**
>
> GuardDuty は、配信されたイベントに基づき、アカウント内のアクティビティについて継続的に学習します。これらのアクティビティと観測されたその頻度は、プロファイルされた動作と呼ばれます。

このパネルで追跡される情報は次のとおりです。

- [ASN 組織] - 異常な API コールが行われた AS 番号 (ASN) 組織。
- [ユーザー名] - 異常な API コールを行ったユーザーの名前。
- ユーザーエージェント - 異常な API コールを行うために使用されるユーザーエージェント。ユーザーエージェントは、`aws-cli` または `Botocore` などのコールを行うために使用されるメソッドです。
- [ユーザータイプ] - 異常な API コールを行ったユーザーの名前。可能な値は、`AWS_SERVICE`、`ASSUMED_ROLE`、`IAM_USER` または `ROLE` です。

  - バケット — アクセスされている S3 バケットの名前。
- [Unusual Behavior (User Identity)] (異常な動作 (ユーザー ID)) - このセクションでは、検出に関与したユーザーアイデンティティのプロファイリングされた動作の詳細について説明します。動作が履歴として識別されない場合は、GuardDuty 機械学習モデルが、トレーニング期間内にこの方法で API コールを行うユーザー ID を認識していないことを意味します。ユーザーアイデンティティに関する詳細については、次を参照してください。
  - [ASN 組織] - 異常な API コールが行われた元の ASN 組織。
  - ユーザーエージェント - 異常な API コールを行うために使用されるユーザーエージェント。ユーザーエージェントは、`aws-cli` または `Botocore` などのコールを行うために使用されるメソッドです。
  - バケット — アクセスされている S3 バケットの名前。
- [Unusual Behavior (Bucket)] (異常な動作 (バケット)) - このセクションでは、検出結果に関連する、S3 バケットのプロファイリングされた動作に関する詳細について説明します。動作が履歴として識別されない場合は、GuardDuty 機械学習モデルが、トレーニング期間内にこの方法によるバケットへの API コールを認識していないことを意味します。このセクションで追跡される情報は次のとおりです。
  - [ASN 組織] - 異常な API コールが行われた元の ASN 組織。
  - [ユーザー名] - 異常な API コールを行ったユーザーの名前。
  - ユーザーエージェント - 異常な API コールを行うために使用されるユーザーエージェント。ユーザーエージェントは、`aws-cli` または `Botocore` などのコールを行うために使用されるメソッドです。
  - [ユーザータイプ] - 異常な API コールを行ったユーザーの名前。可能な値は、`AWS_SERVICE`、`ASSUMED_ROLE`、`IAM_USER` または `ROLE` です。

> **Note**
>
> 過去の動作の詳細については、[Unusual behavior (Account)] (異常な動作 (アカウント))、[User ID] (ユーザー ID)、または [Bucket] (バケット) セクションのいずれかで、[Historical behavior] (過去の動作) を選択して、アカウント内での使用頻度に応じて、[Rare (less than once a month)] (頻度が低い (月に数回))、[Infrequent (a few times a month)] (頻繁でない (月に数回))、または [Frequent (daily to weekly)] (頻度が高い (毎日から毎週)) の各カテゴリのアカウントで予想される動作に関する詳細を表示します。

- 異常な動作 (データベース) - このセクションでは、検出結果に関連するデータベースインスタンスの、プロファイリングされた動作に関する詳細について説明します。動作が履歴として識別されな

い場合は、GuardDuty 機械学習モデルが、トレーニング期間内にこの方法によるデータベースインスタンスへのログイン試行を認識していないことを意味します。検出結果パネルで追跡されるこのセクションの情報は次のとおりです。

- [User name] (ユーザー名) - 異常なログイン試行に使用されたユーザー名
- [ASN Org] (ASN 組織) - 異常なログイン試行が行われた ASN 組織
- [Application name] (アプリケーション名) — 異常なログイン試行に使用されたアプリケーション名
- [データベース名] — 異常なログイン試行に関与したデータベースインスタンス名

[履歴動作] セクションでは、関連するデータベースの [ユーザー名]、[ASN 組織]、[アプリケーション名]、[データベース名] について、以前に確認した内容について詳しく説明しています。固有の値にはそれぞれ、ログインが成功した際にその値が確認された回数を示すカウントが関連付けられています。

- 異常動作 (アカウント Kubernetes クラスター、Kubernetes 名前空間、 Kubernetes ユーザー名) - このセクションでは、検出結果に関連する、Kubernetes クラスターと名前空間の、プロファイリングされた動作に関する詳細を説明します。動作が履歴として識別されない場合、GuardDuty ML モデルが、アカウント、クラスター、名前空間、ユーザー名を認識していないことを意味します。検出結果パネルで追跡されるこのセクションの情報は次のとおりです。
  - ユーザー名 — 検出結果に関連付けられた Kubernetes API を呼び出したユーザー。
  - 偽装されたユーザー名 – `username` に偽装されているユーザー。
  - 名前空間 — アクションが発生した Amazon EKS クラスター内の Kubernetes 名前空間。
  - ユーザーエージェント — Kubernetes API コールに関連付けられたユーザーエージェント。ユーザーエージェントは、`kubectl` などのコールを行うために使用されるメソッドです。
  - API — Amazon EKS クラスター内で `username` によって呼び出される Kubernetes API。
  - ASN 情報 — この呼び出しを行うユーザーの IP アドレスに関連付けられた、組織や ISP などの ASN 情報。
  - 曜日 — Kubernetes API コールが行われた曜日。
  - アクセス権限 — `username` が Kubernetes API を使用できるかどうかのアクセスの確認を受ける Kubernetes の動詞とリソース。
  - サービスアカウント名 – ワークロードに ID を提供する Kubernetes ワークロードに関連付けられたサービスアカウント。
  - レジストリ — Kubernetes ワークロードにデプロイした、コンテナイメージに関連付けられたコンテナレジストリ。

- イメージ — Kubernetes ワークロードにデプロイされた、関連するタグやダイジェストを含まないコンテナイメージ。
- イメージプレフィックス設定 — イメージを使用するコンテナの、hostNetwork または privileged などのコンテナとワークロードセキュリティ設定が有効になっているイメージプレフィックス。
- サブジェクト名 — RoleBinding や ClusterRoleBinding 内の参照ロールにバインドされている user、group、serviceAccountName などのサブジェクト。
- ロール名 — ロールまたは roleBinding API の作成や変更に関係するロールの名前。

S3 ボリュームベースの異常

このセクションでは、S3 ボリュームベースの異常についてのコンテキスト情報について詳しく説明します。ボリュームベースの検出結果 (Exfiltration:S3/AnomalousBehavior) は、ユーザーの S3 バケットに対する異常な数の S3 API コールをモニタリングし、データ漏えいの可能性を示します。以下の S3 の API 呼び出しは、ボリュームベースの異常検出のために監視されます。

- GetObject
- CopyObject.Read
- SelectObjectContent

以下のメトリクスは、IAM エンティティが S3 バケットにアクセスするときの通常の動作のベースラインを構築するのに役立ちます。データの流出を検出するために、ボリュームベースの異常検出結果は通常の行動ベースラインに対してすべてのアクティビティを評価します。次のメトリクスを表示するには、[異常な動作 (ユーザーアイデンティティ)]、[確認されたボリューム (ユーザーアイデンティティ)]、および [確認されたボリューム (バケット)] セクションで [過去の動作] を選択します。

- 過去 24 時間に、影響を受ける S3 バケットに関連付けられた IAM ユーザーまたは IAM ロール (どちらが発行されたかによって異なる) によって呼び出された s3-api-name API コールの数。
- 過去 24 時間に、すべての S3 バケットに関連付けられた IAM ユーザーまたは IAM ロール (どちらが発行されたかによって異なる) によって呼び出された s3-api-name API コールの数。
- 過去 24 時間に、影響を受ける S3 バケットに関連付けられた IAM ユーザーまたは IAM ロール (どちらが発行されたかによって異なる) によって呼び出された s3-api-name API コールの数。

RDS ログインアクティビティベースの異常

このセクションは、異常なアクターが行ったログイン試行回数の詳細について説明するものであり、ログイン試行の結果ごとにグループ分けされています。RDS Protection の検出結果タイプ はログインイベントをモニタリングして successfulLoginCount、failedLoginCount、incompleteConnectionCount などの異常なパターンがないかを確認し、異常な動作を特定します。

- successfulLoginCount — このカウンターは、異常なアクターがデータベースインスタンスに対して行った、成功した接続 (ログイン属性の正しい組み合わせ) の合計を表します。ログイン属性には、ユーザー名、パスワード、データベース名が含まれます。
- failedLoginCount — このカウンターは、データベースインスタンスへの接続を確立しようとして失敗した (不成功の) ログイン試行の合計を表します。これは、ユーザー名、パスワード、データベース名など、ログイン情報の組み合わせの属性が 1 つ以上、正しくなかったことを示します。
- incompleteConnectionCount — このカウンターは、成功または失敗に分類できない接続試行の回数を表します。これらの接続は、データベースが応答する前に閉じられています。例えば、データベースポートは接続されているものの情報がデータベースに送信されないポートスキャンや、ログインが成功または失敗して完了する前に接続が中止された場合などです。

RDS データアクティビティベースの異常

このセクションでは、アクターが大量の行を読み取ったり、そのデータベースとユーザーに対して通常よりも多くのデータを削除または削除したりするなど、異常なデータアクティビティについて説明します。は、読み取りまたは削除された行数とドロップステートメントの異常なパターンについてデータベースアクティビティをモニタリングすることで、異常な動作RDS Protection の検出結果タイプを識別します。

- selectedRows および selectStatement – これらのカウンターは、SELECTステートメントによって返される行数と、検出結果に関連付けられたアクティビティで観測されたSELECTステートメントの数を表します。
- deletedRows および deleteStatement – これらのカウンターは、DELETEステートメントによって削除された行数と、検出結果に関連付けられたアクティビティで観測されたDELETEステートメントの数を表します。
- dropStatement – このカウンターは、検出結果に関連付けられたアクティビティ内のデータベースオブジェクトまたはデータを削除できるTRUNCATE、 DROP や などの破壊的なオペレーションの数を表します。

GuardDuty は、データアクティビティの検出結果の有効性を向上させ、新しい RDS Protection 脅威検出を導入するために、将来的に新しいカウンターなどの追加のデータアクティビティテレメトリを収集する可能性があります。

GuardDuty の検出結果の集約

GuardDuty は生成された検出結果を動的に更新します。GuardDuty が同一のセキュリティ問題に関連する新たなアクティビティを検出した場合、新しい検出結果を作成する代わりに、GuardDuty は最新の情報を基に元の検出結果を更新します。この動作により、同様のレポートを複数確認しなくても進行中の問題を識別することが可能となり、既知のセキュリティ問題に関する全体的な検出結果の量を減らせます。

例えば、UnauthorizedAccess:EC2/SSHBruteForce の検出結果の場合、インスタンスへの複数のアクセスの試行が同じ検出結果 ID に集約され、検出結果の詳細の [カウント] 数が増加します。これは、その検出結果が、インスタンスの SSH ポートがそのタイプのアクティビティに対して適切に保護されていないことを示す単一のセキュリティの問題を示しているためです。ただし、GuardDuty で環境の新しいインスタンスをターゲットとする SSH アクセスのアクティビティが検出されると、一意の検出結果 ID を持つ新しい検出結果が作成され、新しいリソースに関連するセキュリティの問題があることがアラートで示されます。

検出結果が集計されると、そのアクティビティの最新のオカレンスの情報で更新されます。つまり、上記の例ではインスタンスが新しいアクターからのブルートフォースの試みのターゲットになった場合、検出結果の詳細は最も新しいソースのリモート IP を反映して更新され、古い情報は置き換えられることになります。個々のアクティビティの試行についての完全な詳細は、CloudTrail ログまたは VPC フローログに引き続き記録されます。

アラート GuardDuty で既存の検出結果を集約するのではなく、新しい検出結果を生成するようにトリガーする条件は、検出結果タイプによって異なります。各検出結果タイプの集約の条件は、ご利用のアカウントの個別のセキュリティの問題の概要を提供するために、当社のセキュリティエンジニアによって決定されます。

GuardDuty がアカウント内で攻撃シーケンス検出タイプを生成した場合、GuardDuty がアカウント内の同一シーケンス内で類似のシグナルを特定したときのみ、検出結果が集計されます。それ以外の場合、GuardDuty は別の攻撃シーケンスを生成します。

Amazon GuardDuty の検出結果の管理

GuardDuty は、検出結果のソート、保存、管理に役立つ重要な特徴を提供します。これらの機能は、検出結果を特定の環境に合わせて調整し、低価値の検出結果からのノイズを減らし、独自の AWS 環境への脅威に集中するのに役立ちます。環境内のセキュリティ検出結果の価値を高めるために、これらの特徴の使い方を理解し、このページのトピックを見直してください。

トピック:

[Amazon GuardDuty の [要約] ダッシュボード]

GuardDuty コンソールで利用できる概要ダッシュボードのコンポーネントについて説明します。

[GuardDuty での検出結果のフィルタリング]

指定した条件に基づいて GuardDuty の検出結果をフィルターする方法を学習します。

[GuardDuty の抑制ルール]

GuardDuty が抑制ルールでアラートする検出結果を自動的にフィルターする方法を学習します。抑制ルールは、フィルターに基づいて検出結果を自動的にアーカイブします。

[Customizing threat detection with entity lists and IP address lists]

パブリックにルーティング可能な IP アドレスに基づいて、IP リストや脅威リストを使用して、GuardDuty モニタリングスコープをカスタマイズします。信頼できる IP リストは、信頼できる IP から生成された非 DNS の検出結果を抑制し、脅威インテルのリストは、GuardDuty が原因となるユーザーが定義した IP からのアクティビティにアラートを出します。

[生成された検出結果を Amazon S3 にエクスポートする]

生成された検出結果を Amazon S3 バケットにエクスポートして、GuardDuty で 90 日間の検出結果保持期間を過ぎたレコードを保持できるようにします。この履歴データを使用して、アカウント内の潜在的に疑わしいアクティビティを追跡し、推奨される修復手順が成功したかどうかを評価します。

[Amazon EventBridge を使用した GuardDuty 検出結果の処理]

Amazon EventBridge events を通じて GuardDuty の検出結果の自動通知を設定します。EventBridge を通じて他のタスクを自動化して、検出結果への応答を支援します。

[CloudWatch Logs を監査する方法と、Malware Protection for EC2 スキャン中にリソースがスキップされる理由について]

GuardDuty Malware Protection for EC2 の CloudWatch Logs を監査する方法と、影響を受けた Amazon EC2 インスタンスまたは Amazon EBS ボリュームがスキャンプロセス中にスキップされた理由について説明します。

[Malware Protection for EC2 の誤検出の報告]

Malware Protection for EC2 で潜在的な誤検出の脅威検出を報告する方法について説明します。

[Malware Protection for S3 で S3 オブジェクトスキャンの結果を偽陽性として報告]

Malware Protection for S3 で誤検出の可能性がある脅威検出を報告する方法について説明します。

[Malware Protection for Backup での誤検出の報告]

Malware Protection for Backup で誤検出の可能性を報告する方法について説明します。

Amazon GuardDuty の [要約] ダッシュボード

GuardDuty 概要ダッシュボードには、現在の の で生成された GuardDuty の検出結果の集計ビュー AWS アカウント が表示されます AWS リージョン。

GuardDuty 管理者アカウントを使用している場合、ダッシュボードには、自分のアカウントと組織内のメンバーアカウントの集約された統計とデータが表示されます。

概要ダッシュボードの表示

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

   GuardDuty は、コンソールを開くと、デフォルトで[概要] ダッシュボードを表示します。
2. 概要ページで、コンソールの右上隅にあるリージョンセレクター AWS リージョン から目的の を選択します。
3. 日付範囲セレクタメニューから、概要を表示する日付範囲を選択します。デフォルトでは、ダッシュボードには現在の日付、今日のデータが表示されます。

> **Note**
>
> 選択した日付範囲内に検出結果が生成されなかった場合、ダッシュボードには表示するデータがありません。ダッシュボードを更新するか、日付範囲を調整できます。

トピック

- 概要
- 検出結果
- 最も一般的な検出結果タイプ
- 重大度別の検出結果
- 検出結果が最も多いアカウント
- 検出結果を含むリソース
- 最も少ない検出結果
- 保護プランカバレッジ

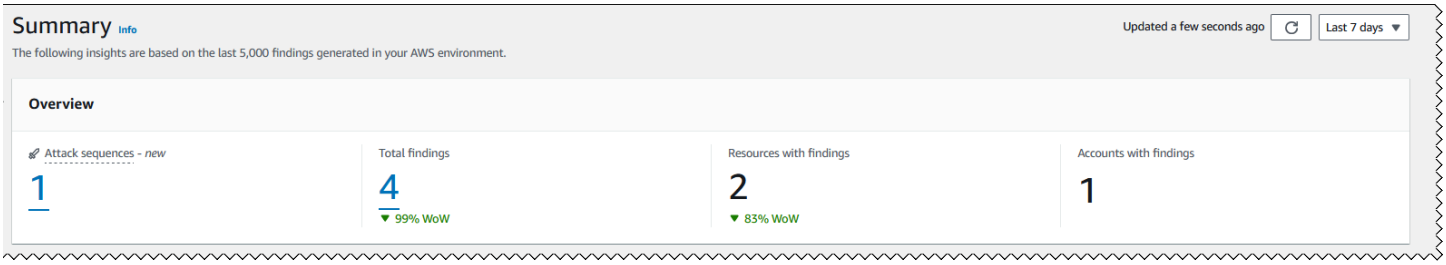
概要

このセクションでは、次のデータを提供します。

- 攻撃シーケンス: GuardDuty が現在のリージョンのアカウントで生成した攻撃シーケンスの検出結果の数を示します。

  GuardDuty は、アカウント内の潜在的なマルチステージ攻撃を検出します。[攻撃シーケンス] で [番号] を選択すると、[結果] ページでその詳細を表示できます。
- 検出結果の合計: 現在のリージョンでご自身のアカウント内に生成された検出結果の総数を示します。これには、個々の検出結果と攻撃シーケンスの検出結果の両方が含まれます。
- 検出結果を含むリソース: 検出結果に関連付けられていて、漏洩した可能性があるリソースの数を示します。
- 検出結果のあるアカウント: 少なくとも 1 つの検出結果が生成されたアカウントの数を示します。スタンドアロンアカウントの場合、このフィールドの値は 1 です。

過去 7 日間と過去 30 日の時間範囲について、[要約] ペインでは、それぞれ、週比 (WoW) や前月比 (MoM) で生成された検出結果の差が割合で表示される場合があります。前週または前月に検出結果が得られなかった場合、比較するデータがないため、差の割合は取得できない可能性があります。



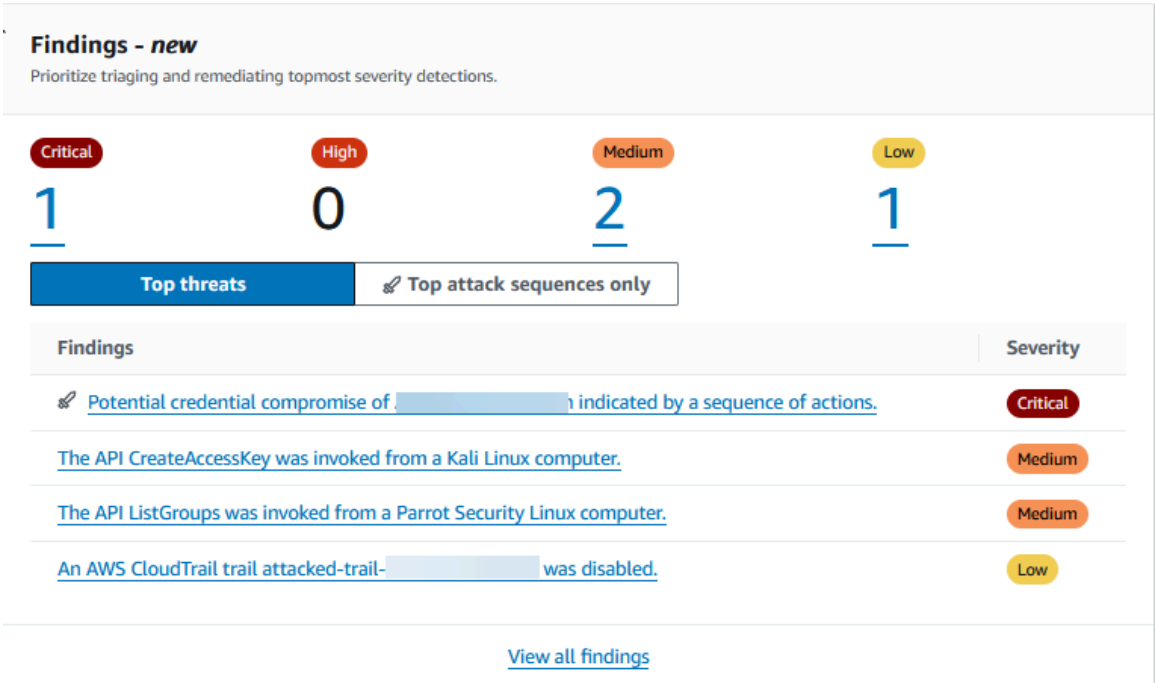
GuardDuty 管理者アカウントの場合、これらのフィールドにはすべて、自身の組織内すべてのメンバーアカウントの概要データが表示されます。

検出結果

検出結果ウィジェットには、最大 8 つの上位の検出結果が表示されます。これらの検出結果は重大度に基づいてリストされ、重大な検出結果が最初に表示されます。

デフォルトでは、すべての検出結果を表示できます。攻撃シーケンスの検出結果データのみを表示するには、[上位の攻撃シーケンスのみ] をオンにします。

このリストでは、任意の検出結果を選択して詳細を表示できます。



最も一般的な検出結果タイプ

このセクションでは、現在のリージョンで生成された最も一般的な検出結果タイプの上位 5 つを示す円グラフを示します。円グラフの各セクターにカーソルを合わせると、以下を確認できます。

- 検出結果数: 選択された日付範囲内でこの検出結果が生成された回数を示します。
- 重大度: 検出結果の重大度レベルを示します。
- パーセンテージ: 合計に対するこの検出結果タイプの割合を示します。
- 最終生成: この検出結果タイプが最後に検出されてからどれくらいの時間が経過したかを示します。

重大度別の検出結果

このセクションには、選択した日付範囲における検出結果の合計数を示す棒グラフが表示されます。このグラフは、検出結果を重大度 (重大、高、中、低) 別に分類し、範囲内の特定の日付の検出結果の数を表示するのに役立ちます。

特定の日の各深刻度レベルごとの件数を確認するには、グラフの対応する棒の上にカーソルを合わせてください。

検出結果が最も多いアカウント

このセクションでは、次のデータを提供します。

- アカウント: 結果が生成された AWS アカウント ID を示します。
- 検出結果数: このアカウント ID の検出結果が生成された回数を示します。
- 最終生成: このアカウント ID の検出結果タイプが最後に生成されてからどれくらいの時間が経過したかを示します。
- 重大度フィルター: デフォルトでは、重大度の高い検出結果タイプのデータが表示されます。このフィールドに指定できるオプションは、すべての重大度、重大度重大、重大度高、および重大度中です。

検出結果を含むリソース

このセクションでは、次のデータを提供します。

- リソース: 影響を受ける可能性のあるリソースタイプを示します。このリソースがアカウントに属している場合は、クイックリンクにアクセスしてリソースの詳細を表示できます。自身が GuardDuty 管理者アカウントの場合は、管理者メンバーアカウントアカウントの認証情報を使用して GuardDuty コンソールにアクセスすることで、影響を受ける可能性のあるリソースの詳細を表示できます。
- アカウント: このリソースが属する AWS アカウント ID を示します。
- 検出結果回数: このリソースが検出結果に関連付けられた回数を示します。
- 最終生成: このリソースに関連付けられた検出結果タイプが最後に生成されてからどれくらいの時間が経過したかを示します。
- リソースタイプフィルター: デフォルトでは、すべてのリソースタイプのデータが表示されます。このフィルターを使用すると、インスタンス、AccessKey、Lambda など、特定のリソースタイプのデータを表示できます。
- 重大度フィルター: デフォルトでは、データはすべての重大度で表示されます。このフィルターを使用すると、他の重大度レベルのデータを表示するように選択できます。選択できるオプションは、重大度重大、重大度高、重大度中、および すべての重大度です。

最も少ない検出結果

このセクションでは、 AWS 環境でまれに発生する検出結果タイプについて説明します。このウィジェットは、潜在的な新たな脅威のパターンを特定し調査するのに役立つように設計されています。

このウィジェットは次のデータを表示します。

- 検出結果タイプ: 検出結果タイプ名を示します。
- 検出結果数: 選択された時間範囲でこのタイプの検出結果が生成された回数を示します。
- 最終生成: この検出結果タイプが最後に生成されてからどれくらいの時間が経過したかを示します。
- 重大度フィルター: デフォルトでは、重大度の高い検出結果タイプのデータが表示されます。このフィールドに指定できるオプションは、重大度重大、重大度高、重大度中、および すべての重大度です。

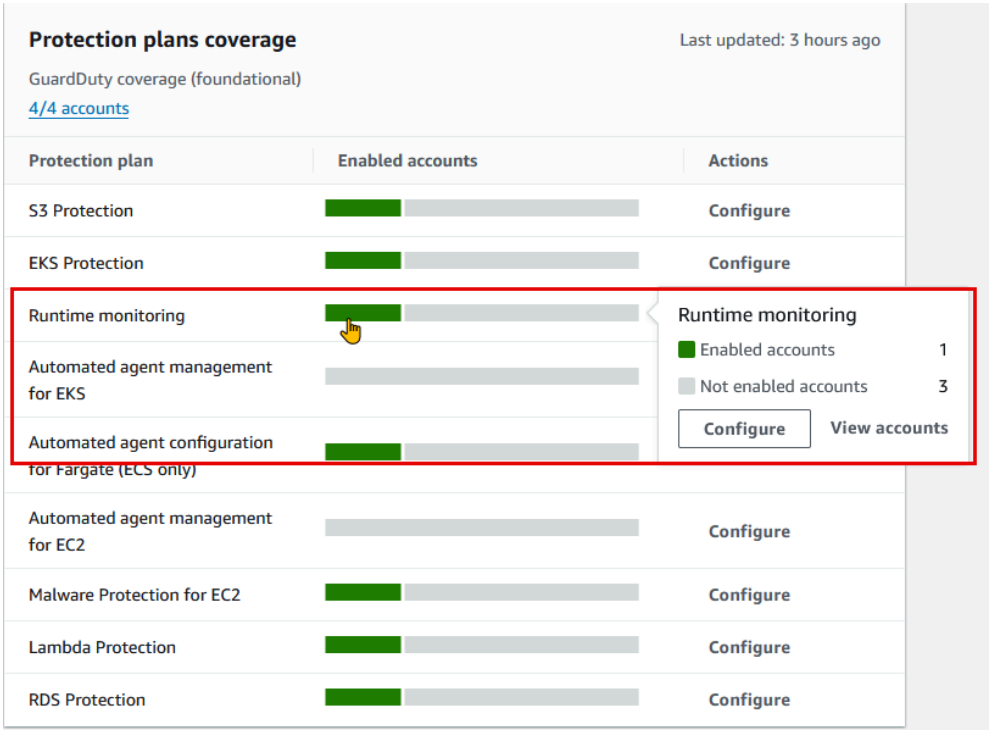
保護プランカバレッジ

このセクションでは、組織内のメンバーアカウントの統計情報を表示します。現在のリージョンで GuardDuty (基礎的な脅威検出) を有効にしたメンバーアカウントの数を示します。委任 GuardDuty

管理者のみが、組織内のメンバーアカウントの統計を表示できます。新しい AWS 組織を作成すると、組織全体の統計を生成するまでに最大 24 時間かかる場合があります。

このウィジェットの使用方法

- 設定: 保護プランが設定されていない場合は、[アクション] 列で[設定] を選択します。
- 有効なアカウントの表示: [有効なアカウント] 列のバーにカーソルを合わせると、各保護プランを有効にしたアカウントの数が表示されます。アカウントの詳細をさらに表示するには、緑色のバーを選択し、[アカウントの表示] を選択します。



GuardDuty での検出結果のフィルタリング

検出結果フィルターを使用すると、指定した条件に一致する検出結果を表示し、一致しない検出結果を除外できます。Amazon GuardDuty コンソールを使用して検出結果フィルターを簡単に作成することも、JSON を使用して CreateFilter API を使用してフィルターを作成することもできます。コンソールでフィルターを作成する方法については、次のセクションを参照してください。これらのフィルターを使用して受信した検出結果を自動的にアーカイブするには、「GuardDuty の抑制ルール」を参照してください。

フィルターを作成するときは、以下のリストを考慮してください。

- 特定のフィルターのための基準として、最少 1 から最大 50 までの属性を指定できます。

- アカウント ID などの属性値でフィルタリングする際に 等号演算子または 不等号演算子の条件を使用する最合、最大で 50 個の値を指定できます。
- 各フィルター基準の属性は AND 演算子として評価されます。同じ属性の複数の値は AND/OR として評価されます。
- 各 AWS アカウント の で作成できる保存済みフィルターの最大数については AWS リージョン、「」を参照してくださいGuardDuty クォータ。
- のフィールドservice.additionalInfoは、他のフィールドと同じ完全な JSON パスを使用して指定されます。例: { "service.additionalInfo.sample": { "Equals": ["true"] } }。
- タイムスタンプフィールドは、Unix エポックミリ秒形式 (例:) の値を受け入れます1486685375000。タイムスタンプフィールドの完全なリストについては、以下の注意事項を参照してください。

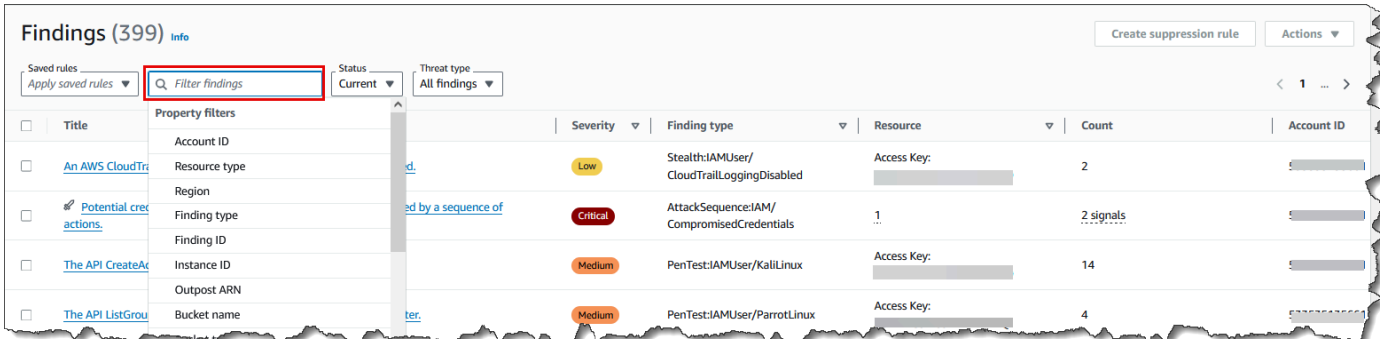
以下のセクションでは、GuardDuty コンソール、API コマンド、CLI コマンドを用いたフィルターの作成と保存手順について説明します。好みのアクセス方法を選択して続行します。

GuardDuty コンソールでのフィルターの作成と保存

検出結果フィルターは、GuardDuty コンソールを通して 作成およびテストすることができます。抑制ルールやその後のフィルターオペレーションで使用するためにコンソールを通して作成したフィルターは、保存できます。フィルターは、少なくとも 1 つのフィルター基準で構成されます。その基準は少なくとも 1 つの値と組み合わさった 1 つのフィルター属性で構成されています。

フィルター条件を作成および保存するには (コンソール)

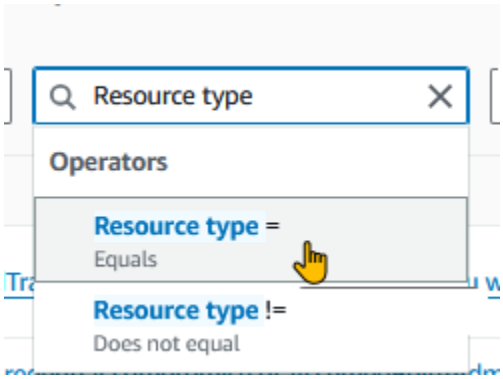
1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. 左のナビゲーションペインで [検出結果] を選択します。
3. [検出結果] ページで、[保存されたルール] メニューの横にある [フィルター検出結果] バーを選択します。これにより、プロパティフィルターの展開されたリストが表示されます。



4. フィルターの展開されたリストから、検出結果テーブルをフィルタリングする属性を選択します。

   例えば、影響を受ける可能性のあるリソースが S3Bucket である検出結果を表示するには、[リソースタイプ] を選択します。

5. 演算子 では、検出結果をフィルタリングして目的の検出結果を得るのに役立つものを選択します。前のステップの例を続行するには、[リソースタイプ =] を選択します。これにより、GuardDuty のリソースタイプのリストが表示されます。



   ユースケースで特定の検出結果を除外する必要がある場合は、不等号または != 演算子を選択できます。

6. 選択したプロパティフィルターの値を指定します。必要に応じて、[適用] を選択します。前のステップの例を続行するには、[S3Bucket] を選択します。

   これにより、適用されたフィルターに一致する検出結果が表示されます。

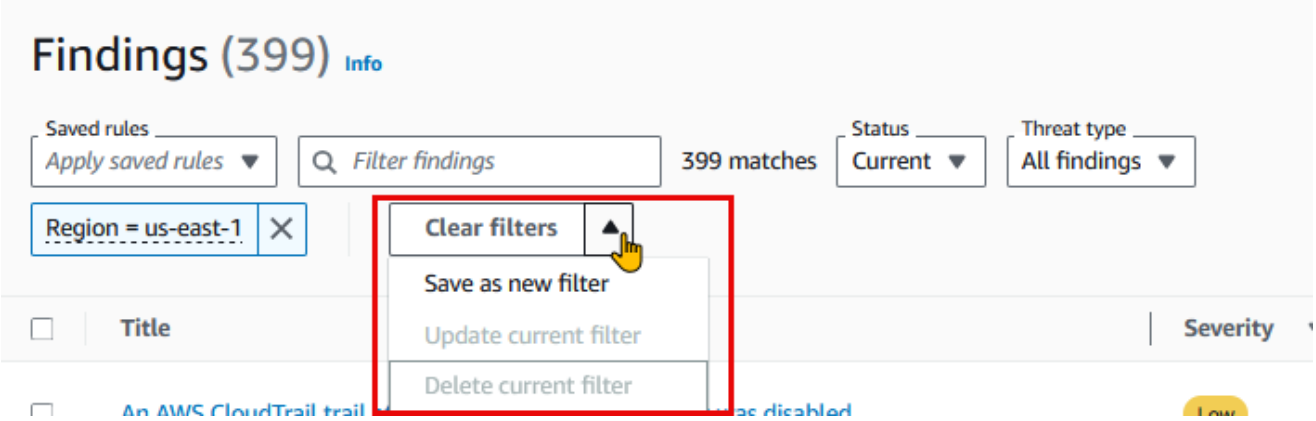
7. 複数のフィルター条件を追加するには、3 から 6 の手順を繰り返します。

   属性の詳細なリストについては、「GuardDuty のプロパティフィルター」を参照してください。

8. (オプション) 指定された属性と値をフィルターとして保存します

   このフィルターの組み合わせを将来再度適用するには、指定された属性とその値をフィルターセットとして保存できます。

   a. 1 つ以上のプロパティフィルターでフィルター条件を作成したら、[フィルターをクリア] メニューで [矢印] を選択します。



   b. フィルターセット [名前] を入力します。名前は 3 ~ 64 文字にする必要があります。有効な文字は、a-z、A ~ Z、0 ~ 9、ピリオド ( )、ハイフン (-)、アンダースコア (_) です。

   c. 説明はオプションです。説明を入力する場合、最大 512 文字まで使用できます。

   d. [作成] を選択します。

GuardDuty API と CLI を使用したフィルターセットの作成と保存

API または CLI コマンドを使用して、検出結果フィルターを作成してテストできます。フィルターは、少なくとも 1 つのフィルター基準で構成されます。その基準は少なくとも 1 つの値と組み合わさった 1 つのフィルター属性で構成されています。フィルターを保存して、後で 抑制ルール を作成したり他のフィルターオペレーションを実行したりできます。

API/CLI を使用して検出結果フィルターを作成するには

- フィルターを作成する のリージョンディテクター ID を使用して AWS アカウント CreateFilter API を実行します。

  アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

- または、create-filter CLI を使用してフィルターを作成して保存することもできます。GuardDuty のプロパティフィルター から 1 つ以上のフィルター条件を使用できます。

  赤で示されているプレースホルダー値を置き換えて、以下の例を使用します。

  例 1: 特定の検出結果タイプに一致するすべての検出結果を表示する新しいフィルターを作成する

  次の例では、特定のイメージから作成されたインスタンスのすべての PortScan 検出結果に一致するフィルターを作成します。プレースホルダー値は赤で示されます。これらの値をアカウントに適した値に置き換えます。例えば *12abc34d567e8fa901bc2d34EXAMPLE* を自分のリージョン detector-id に置き換えます。

```
aws guardduty create-filter \
--detector-id 12abc34d567e8fa901bc2d34EXAMPLE \
--name FilterExampleName \
--finding-criteria '{"Criterion": {"type": {"Equals": ["Recon:EC2/Portscan"]},
 "resource.instanceDetails.imageId": {"Equals":["ami-0a7a207083example"]}} }'
```

  例 2: 重大度レベルに一致するすべての検出結果を表示する新しいフィルターを作成する

  次の例では、HIGH 重大度レベルに関連付けられたすべての検出結果に一致するフィルターを作成します。プレースホルダー値は赤で示されます。これらの値をアカウントに適した値に置き換えます。例えば *12abc34d567e8fa901bc2d34EXAMPLE* を自分のリージョン detector-id に置き換えます。

```
aws guardduty create-filter \
--detector-id 12abc34d567e8fa901bc2d34EXAMPLE \
--name FilterExampleName \
--finding-criteria '{"Criterion": {"severity": {"Equals": ["7", "8"]}} }'
```

- API/CLI の場合、検出結果の重大度レベル は数字で表されます。深刻度レベルに基づいて結果をフィルタリングするには、以下の値を使用してください。
  - LOW 重大度レベルには、`{ "severity": { "Equals": ["1", "2", "3"] } }` を使用します
  - MEDIUM 重大度レベルには、`{ "severity": { "Equals": ["4", "5", "6"] } }` を使用します
  - HIGH 重大度レベルには、`{ "severity": { "Equals": ["7", "8"] } }` を使用します
  - CRITICAL 重大度レベルには、`{ "severity": { "Equals": ["9", "10"] } }` を使用します

- 複数の重大度レベルの検出結果については、次の例のようなプレースホルダー値を使用します。`{ "severity": { "Equals": ["7", "8", "9", "10"] } }`

  この例では、HIGH または CRITICAL の重大度レベルを持つ検出結果を示します。

  > **Note**
  >
  > 重大度レベルに関連付けられた数値のすべてではなく、1 つの数値のみを持つ例を指定した場合、API と CLI ではフィルタリングされた結果が表示されることがあります。この保存されたフィルターセットを GuardDuty コンソールで使用すると、期待どおりに動作しません。これは、GuardDuty コンソールがフィルター値を CRITICAL、HIGH、MEDIUM、および LOW と見なすためです。例えば、`{ "severity": { "Equals": ["9"] } }` を含む CLI コマンドで作成されたフィルターは、API/CLI で適切な出力を表示することが期待されます。ただし、GuardDuty コンソールで使用した場合、この保存されたフィルターには重大度レベルが部分的に含まれ、期待通りの出力は表示されません。このため、API と CLI では各重大度レベルに関連付けられたすべての値を指定する必要があります。

ライフサイクルメタデータをフィルタリングする

GetFilter API を使用してフィルターの詳細を取得すると、ライフサイクルメタデータフィールドを表示できます。これらのフィールドには、フィルターが作成された日時と、フィルターが変更された回数が表示されます。

GetFilter レスポンスには、次のライフサイクルメタデータフィールドが含まれます。

`createdAt`

フィルターが最初に作成されたタイムスタンプ。GuardDuty はこの値を作成時に設定し、変更することはありません。GuardDuty は、レガシーフィルター (この機能の起動前に作成したフィルター) に対してこのフィールドに入力しません。

`updatedAt`

フィルターが最後に更新されたタイムスタンプ。GuardDuty はUpdateFilter、呼び出しが成功するたびにこの値を更新します。作成時に、 は updatedAtに一致しますcreatedAt。レガシーフィルターの場合、このフィールドはフィルターを少なくとも 1 回更新した後にのみ表示されます。

`version`

フィルターを変更した回数をカウントする整数。フィルタの作成時に 1 から開始され、フィルタの内容が変更されるかどうかにかかわらず、UpdateFilter呼び出しが成功するたびに 1 ずつ増加します。GuardDuty はレガシーフィルターにこのフィールドを入力しません。

> **Note**
>
> これらのライフサイクルメタデータフィールドは読み取り専用です。CreateFilter および UpdateFilterオペレーションでは、これらのフィールドを入力として受け付けず、それらのレスポンスにはこれらのフィールドは含まれません。ライフサイクルメタデータを表示するには、 を呼び出しますGetFilter。

GuardDuty のプロパティフィルター

API オペレーションを使用してフィルターを作成したり、検出結果を並べ替える場合は、JSON 中でフィルター基準を特定する必要があります。これらのフィルター基準は、検出結果の詳細 JSON と相関します。次の表にフィルター属性のコンソール表示名と、それに対応する JSON フィールド名のリストを表示します。

コンソールフィールド名	JSON フィールド名
アカウント ID	accountId
検出結果 ID	id
リージョン	region
緊急度	severity 検出結果タイプの重大度レベルに基づいて検出結果タイプをフィルタリングできます。重大度の値についての詳細は、「GuardDuty の検出結果の重大度」を参照してください。API、 AWS CLI、または `severity`で を使用する場合は CloudFormation、数値が割り当てられます。詳細については、「Amazon GuardDuty API リ

コンソールフィールド名	JSON フィールド名
	ファレンス」の「findingCriteria」を参照してください。
検出結果タイプ	type
カスタム検出ルール名	customDetectionRuleName Runtime Monitoring Custom Detection Rule が生成した検出結果にのみ表示されます。詳細については、「Runtime Monitoring ルールの結果」を参照してください。
更新時刻	updatedAt
アクセスキー ID	resource.accessKeyDetails.accessKeyId
プリンシパル ID	resource.accessKeyDetails.principalId
ユーザーネーム	resource.accessKeyDetails.userName
ユーザータイプ	resource.accessKeyDetails.userType
IAM インスタンスプロファイル ID	resource.instanceDetails.iamInstanceProfile.id
インスタンス ID	resource.instanceDetails.instanceId
インスタンスイメージ ID	resource.instanceDetails.imageId
インスタンスのタグキー。	resource.instanceDetails.tags.key
インスタンスのタグ値。	resource.instanceDetails.tags.value
IPv6 アドレス	resource.instanceDetails.networkInterfaces.ipv6Addresses
プライベート IPv4 アドレス	resource.instanceDetails.networkInterfaces.privateIpAddresses.privateIpAddress

コンソールフィールド名	JSON フィールド名
パブリック DNS 名	resource.instanceDetails.networkInterfaces.publicDnsName
パブリック IP	resource.instanceDetails.networkInterfaces.publicIp
セキュリティグループ ID	resource.instanceDetails.networkInterfaces.securityGroups.groupId
セキュリティグループ名	resource.instanceDetails.networkInterfaces.securityGroups.groupName
サブネット ID	resource.instanceDetails.networkInterfaces.subnetId
VPC ID	resource.instanceDetails.networkInterfaces.vpcId
Outpost ARN	resource.instanceDetails.outpostARN
リソースタイプ	resource.resourceType
バケット許可	resource.s3BucketDetails.publicAccess.effectivePermission
バケット名	resource.s3BucketDetails.name
バケットタグキー	resource.s3BucketDetails.tags.key
バケットタグ値	resource.s3BucketDetails.tags.value
バケットタイプ	resource.s3BucketDetails.type
アクションタイプ	service.action.actionType
呼び出された API	service.action.awsApiCallAction.api
API 発信者のタイプ	service.action.awsApiCallAction.callerType

コンソールフィールド名	JSON フィールド名
API エラーコード	service.action.awsApiCallAction.errorCode
API 発信者の都市	service.action.awsApiCallAction.remoteIpDetails.city.cityName
API 発信者の国	service.action.awsApiCallAction.remoteIpDetails.country.countryName
API 発信者の IPv4 アドレス	service.action.awsApiCallAction.remoteIpDetails.ipAddressV4
API 発信者の IPv6 アドレス	service.action.awsApiCallAction.remoteIpDetails.ipAddressV6
API 発信者の ASN ID	service.action.awsApiCallAction.remoteIpDetails.organization.asn
API 発信者の ASN 名	service.action.awsApiCallAction.remoteIpDetails.organization.asnOrg
API 発信者のサービス名	service.action.awsApiCallAction.serviceName
DNS リクエストドメイン	service.action.dnsRequestAction.domain
DNS リクエストドメインサフィックス	service.action.dnsRequestAction.domainWithSuffix
ブロック済みのネットワーク接続	service.action.networkConnectionAction.blocked
ネットワーク接続の方向	service.action.networkConnectionAction.connectionDirection
ネットワーク接続のローカルポート	service.action.networkConnectionAction.localPortDetails.port
ネットワーク接続プロトコル	service.action.networkConnectionAction.protocol

コンソールフィールド名	JSON フィールド名
ネットワーク接続の都市	service.action.networkConnectionAction.remoteIpDetails.city.cityName
ネットワーク接続の国	service.action.networkConnectionAction.remoteIpDetails.country.countryName
ネットワーク接続のリモート IPv4 アドレス	service.action.networkConnectionAction.remoteIpDetails.ipAddressV4
ネットワーク接続のリモート IPv6 アドレス	service.action.networkConnectionAction.remoteIpDetails.ipAddressV6
ネットワーク接続のリモート IP ASN ID	service.action.networkConnectionAction.remoteIpDetails.organization.asn
ネットワーク接続のリモート IP ASN 名	service.action.networkConnectionAction.remoteIpDetails.organization.asnOrg
ネットワーク接続のリモートポート	service.action.networkConnectionAction.remotePortDetails.port
関連するリモートアカウント	service.action.awsApiCallAction.remoteAccountDetails.affiliated
Kubernetes API 発信者の IPv4 アドレス	service.action.kubernetesApiCallAction.remoteIpDetails.ipAddressV4
Kubernetes API 発信者の IPv6 アドレス	service.action.kubernetesApiCallAction.remoteIpDetails.ipAddressV6
Kubernetes 名前空間	service.action.kubernetesApiCallAction.namespace
Kubernetes API 発信者の ASN ID	service.action.kubernetesApiCallAction.remoteIpDetails.organization.asn
Kubernetes API 呼び出しリクエスト URI	service.action.kubernetesApiCallAction.requestUri

コンソールフィールド名	JSON フィールド名
Kubernetes API ステータスコード	service.action.kubernetesApiCallAction.statusCode
ネットワーク接続のローカル IPv4 アドレス	service.action.networkConnectionAction.localIpDetails.ipAddressV4
ネットワーク接続のローカル IPv6 アドレス	service.action.networkConnectionAction.localIpDetails.ipAddressV6
プロトコル	service.action.networkConnectionAction.protocol
API 呼び出しのサービス名	service.action.awsApiCallAction.serviceName
API 発信者アカウント ID	service.action.awsApiCallAction.remoteAccountDetails.accountId
脅威リスト名	service.additionalInfo.threatListName
リソースロール	service.resourceRole
EKS クラスター名	resource.eksClusterDetails.name
Kubernetes ワークロード名	resource.kubernetesDetails.kubernetesWorkloadDetails.name
Kubernetes ワークロード名前空間	resource.kubernetesDetails.kubernetesWorkloadDetails.namespace
Kubernetes ユーザー名	resource.kubernetesDetails.kubernetesUserDetails.username
Kubernetes コンテナイメージ	resource.kubernetesDetails.kubernetesWorkloadDetails.containers.image
Kubernetes コンテナイメージのプレフィックス	resource.kubernetesDetails.kubernetesWorkloadDetails.containers.imagePrefix

コンソールフィールド名	JSON フィールド名
スキャン ID	service.ebsVolumeScanDetails.scanId
EBS ボリュームスキャンの脅威名	service.ebsVolumeScanDetails.scanDetections.threatDetectedByName.threatNames.name
S3 オブジェクトスキャンの脅威名	service.malwareScanDetails.threats.name
脅威の重大度	service.ebsVolumeScanDetails.scanDetections.threatDetectedByName.threatNames.severity
SHA ファイル	service.ebsVolumeScanDetails.scanDetections.threatDetectedByName.threatNames.filePaths.hash
ECS クラスター名	resource.ecsClusterDetails.name
ECS コンテナイメージ	resource.ecsClusterDetails.taskDetails.containers.image
ECS タスク定義 ARN	resource.ecsClusterDetails.taskDetails.definitionArn
スタンドアロンコンテナのイメージ	resource.containerDetails.image
データベースインスタンス ID	resource.rdsDbInstanceDetails.dbInstanceIdentifier
データベースクラスター ID	resource.rdsDbInstanceDetails.dbClusterIdentifier
データベースエンジン	resource.rdsDbInstanceDetails.engine
データベースユーザー	resource.rdsDbUserDetails.user
実行可能ファイル SHA-256	service.runtimeDetails.process.executableSha256

コンソールフィールド名	JSON フィールド名
プロセス名	service.runtimeDetails.process.name
実行可能ファイルのパス	service.runtimeDetails.process.executablePath
Lambda 関数の名前	resource.lambdaDetails.functionName
Lambda 関数の ARN	resource.lambdaDetails.functionArn
Lambda 関数タグキー	resource.lambdaDetails.tags.key
Lambda 関数タグ値	resource.lambdaDetails.tags.value
DNS リクエストドメイン	service.action.dnsRequestAction.domainWithSuffix

他のすべての検出結果フィールド (以下を参照) は、抑制ルールフィルター条件としてのみ使用できます (CreateFilter と UpdateFilter を使用)。これらのフィールドは、他の API オペレーションではサポートされていません。これらのフィールドを使用する抑制ルールは、 API を使用して作成または更新する必要があります。これらのフィールドは、 `ARCHIVE`アクションを持つフィルターにのみ適用できます。

> **Note**
>
> 次のフィールドは、Unix エポックミリ秒形式のタイムスタンプ値を受け入れます (たとえば、2010 年 1 月 1 日金曜日の午前 1:23:45 GMT `1262309025000`を表します)。
>
> - createdAt
> - updatedAt
> - service.eventFirstSeen
> - service.eventLastSeen
> - resource.instanceDetails.launchTime
> - resource.lambdaDetails.lastModifiedAt
> - resource.s3BucketDetails.createdAt
> - resource.eksClusterDetails.createdAt
> - resource.ecsClusterDetails.taskDetails.createdAt
> - resource.ecsClusterDetails.taskDetails.startedAt

- service.ebsVolumeScanDetails.scanStartedAt
- service.ebsVolumeScanDetails.scanCompletedAt
- service.runtimeDetails.context.modifiedAt
- service.runtimeDetails.context.modifyingProcess.startTime
- service.runtimeDetails.context.modifyingProcess.lineage.startTime
- service.runtimeDetails.context.targetProcess.startTime
- service.runtimeDetails.context.targetProcess.lineage.startTime
- service.runtimeDetails.process.startTime
- service.runtimeDetails.process.lineage.startTime
- service.detection.sequence.actors.session.createdTime
- service.detection.sequence.signals.createdAt
- service.detection.sequence.signals.updatedAt
- service.detection.sequence.signals.firstSeenAt
- service.detection.sequence.signals.lastSeenAt
- service.detection.sequence.resources.data.s3Bucket.createdAt
- service.detection.sequence.resources.data.ecsTask.createdAt
- service.detection.sequence.resources.data.eksCluster.createdAt

JSON フィールド名
arn
associatedAttackSequenceArn
createdAt
パーティション
resource.accessKeyDetails.userIdentity.accessKeyId
resource.accessKeyDetails.userIdentity.accountId
resource.accessKeyDetails.userIdentity.arn
resource.accessKeyDetails.userIdentity.principalId
resource.accessKeyDetails.userIdentity.sessionContext.attributes.mfaAuthenticated

JSON フィールド名
resource.accessKeyDetails.userIdentity.sessionContext.ec2RoleDelivery
resource.accessKeyDetails.userIdentity.sessionContext.invokedBy
resource.accessKeyDetails.userIdentity.sessionContext.sessionIssuer.accountId
resource.accessKeyDetails.userIdentity.sessionContext.sessionIssuer.arn
resource.accessKeyDetails.userIdentity.sessionContext.sessionIssuer.principalId
resource.accessKeyDetails.userIdentity.sessionContext.sessionIssuer.type
resource.accessKeyDetails.userIdentity.sessionContext.sessionIssuer.userName
resource.accessKeyDetails.userIdentity.sessionContext.sourceIdentity
resource.accessKeyDetails.userIdentity.sessionContext.webIdFederationData.attributes
resource.accessKeyDetails.userIdentity.sessionContext.webIdFederationData.federatedProvider
resource.accessKeyDetails.userIdentity.type
resource.accessKeyDetails.userIdentity.userName
resource.bedrockGuardrailDetails.guardrailArn
resource.bedrockGuardrailDetails.guardrailVersion
resource.containerDetails.containerRuntime
resource.containerDetails.imagePrefix
resource.containerDetails.securityContext.allowPrivilegeEscalation
resource.containerDetails.securityContext.privileged
resource.containerDetails.volumeMounts.mountPath
resource.containerDetails.volumeMounts.name

JSON フィールド名
resource.ebsVolumeDetails.scannedVolumeDetails.deviceName
resource.ebsVolumeDetails.scannedVolumeDetails.encryptionType
resource.ebsVolumeDetails.scannedVolumeDetails.kmsKeyArn
resource.ebsVolumeDetails.scannedVolumeDetails.snapshotArn
resource.ebsVolumeDetails.scannedVolumeDetails.volumeArn
resource.ebsVolumeDetails.scannedVolumeDetails.volumeSizeInGB
resource.ebsVolumeDetails.scannedVolumeDetails.volumeType
resource.ebsVolumeDetails.skippedVolumeDetails.deviceName
resource.ebsVolumeDetails.skippedVolumeDetails.encryptionType
resource.ebsVolumeDetails.skippedVolumeDetails.kmsKeyArn
resource.ebsVolumeDetails.skippedVolumeDetails.snapshotArn
resource.ebsVolumeDetails.skippedVolumeDetails.volumeArn
resource.ebsVolumeDetails.skippedVolumeDetails.volumeSizeInGB
resource.ebsVolumeDetails.skippedVolumeDetails.volumeType
resource.ecsClusterDetails.activeServicesCount
resource.ecsClusterDetails.arn
resource.ecsClusterDetails.registeredContainerInstancesCount
resource.ecsClusterDetails.runningTasksCount
resource.ecsClusterDetails.status
resource.ecsClusterDetails.tags.key

JSON フィールド名
resource.ecsClusterDetails.tags.value
resource.ecsClusterDetails.taskDetails.arn
resource.ecsClusterDetails.taskDetails.containers.containerRuntime
resource.ecsClusterDetails.taskDetails.containers.id
resource.ecsClusterDetails.taskDetails.containers.imagePrefix
resource.ecsClusterDetails.taskDetails.containers.name
resource.ecsClusterDetails.taskDetails.containers.securityContext.allowPrivilegeEscalation
resource.ecsClusterDetails.taskDetails.containers.securityContext.privileged
resource.ecsClusterDetails.taskDetails.containers.volumeMounts.mountPath
resource.ecsClusterDetails.taskDetails.containers.volumeMounts.name
resource.ecsClusterDetails.taskDetails.createdAt
resource.ecsClusterDetails.taskDetails.group
resource.ecsClusterDetails.taskDetails.launchType
resource.ecsClusterDetails.taskDetails.startedAt
resource.ecsClusterDetails.taskDetails.startedBy
resource.ecsClusterDetails.taskDetails.tags.key
resource.ecsClusterDetails.taskDetails.tags.value
resource.ecsClusterDetails.taskDetails.version
resource.ecsClusterDetails.taskDetails.volumes.hostPath.path
resource.ecsClusterDetails.taskDetails.volumes.name

JSON フィールド名
resource.eksClusterDetails.arn
resource.eksClusterDetails.createdAt
resource.eksClusterDetails.status
resource.eksClusterDetails.tags.key
resource.eksClusterDetails.tags.value
resource.eksClusterDetails.vpcId
resource.instanceDetails.iamInstanceProfile.arn
resource.instanceDetails.instanceState
resource.instanceDetails.instanceType
resource.instanceDetails.launchTime
resource.instanceDetails.networkInterfaces.networkInterfaceId
resource.instanceDetails.networkInterfaces.privateDnsName
resource.instanceDetails.networkInterfaces.privateIpAddress
resource.instanceDetails.networkInterfaces.privateIpAddresses.privateDnsName
resource.instanceDetails.platform
resource.instanceDetails.productCodes.productCodeId
resource.instanceDetails.productCodes.productCodeType
resource.kubernetesDetails.kubernetesUserDetails.groups
resource.kubernetesDetails.kubernetesUserDetails.impersonatedUser.groups
resource.kubernetesDetails.kubernetesUserDetails.impersonatedUser.username

JSON フィールド名
resource.kubernetesDetails.kubernetesUserDetails.sessionName
resource.kubernetesDetails.kubernetesUserDetails.uid
resource.kubernetesDetails.kubernetesWorkloadDetails.containers.containerRuntime
resource.kubernetesDetails.kubernetesWorkloadDetails.containers.id
resource.kubernetesDetails.kubernetesWorkloadDetails.containers.name
resource.kubernetesDetails.kubernetesWorkloadDetails.containers.securityContext.allowPrivilegeEscalation
resource.kubernetesDetails.kubernetesWorkloadDetails.containers.securityContext.privileged
resource.kubernetesDetails.kubernetesWorkloadDetails.containers.volumeMounts.mountPath
resource.kubernetesDetails.kubernetesWorkloadDetails.containers.volumeMounts.name
resource.kubernetesDetails.kubernetesWorkloadDetails.hostIpc
resource.kubernetesDetails.kubernetesWorkloadDetails.hostNetwork
resource.kubernetesDetails.kubernetesWorkloadDetails.hostPid
resource.kubernetesDetails.kubernetesWorkloadDetails.serviceAccountName
resource.kubernetesDetails.kubernetesWorkloadDetails.type
resource.kubernetesDetails.kubernetesWorkloadDetails.uid
resource.kubernetesDetails.kubernetesWorkloadDetails.volumes.hostPath.path
resource.kubernetesDetails.kubernetesWorkloadDetails.volumes.name
resource.lambdaDetails.description
resource.lambdaDetails.lastModifiedAt
resource.lambdaDetails.revisionId

JSON フィールド名
resource.lambdaDetails.vpcConfig.securityGroups.groupId
resource.lambdaDetails.vpcConfig.securityGroups.groupName
resource.lambdaDetails.vpcConfig.subnetIds
resource.lambdaDetails.vpcConfig.vpcId
resource.rdsDbInstanceDetails.dbInstanceArn
resource.rdsDbInstanceDetails.dbiResourceId
resource.rdsDbInstanceDetails.dbSecurityGroups.name
resource.rdsDbInstanceDetails.dbSecurityGroups.status
resource.rdsDbInstanceDetails.engineVersion
resource.rdsDbInstanceDetails.iamDatabaseAuthenticationEnabled
resource.rdsDbInstanceDetails.publiclyAccessible
resource.rdsDbInstanceDetails.vpcId
resource.rdsDbInstanceDetails.vpcSecurityGroups.status
resource.rdsDbInstanceDetails.vpcSecurityGroups.vpcSecurityGroupId
resource.rdsDbUserDetails.application
resource.rdsDbUserDetails.authMethod
resource.rdsDbUserDetails.database
resource.rdsDbUserDetails.ssl
resource.rdsLimitlessDbDetails.dbClusterIdentifier
resource.rdsLimitlessDbDetails.dbShardGroupArn

JSON フィールド名
resource.rdsLimitlessDbDetails.dbShardGroupIdentifier
resource.rdsLimitlessDbDetails.dbShardGroupResourceId
resource.rdsLimitlessDbDetails.engine
resource.rdsLimitlessDbDetails.engineVersion
resource.rdsLimitlessDbDetails.tags.key
resource.rdsLimitlessDbDetails.tags.value
resource.s3BucketDetails.arn
resource.s3BucketDetails.createdAt
resource.s3BucketDetails.defaultServerSideEncryption.encryptionType
resource.s3BucketDetails.defaultServerSideEncryption.kmsMasterKeyArn
resource.s3BucketDetails.owner.id
resource.s3BucketDetails.publicAccess.permissionConfiguration.accountLevelPermissions.blockPublicAccess.blockPublicAcls
resource.s3BucketDetails.publicAccess.permissionConfiguration.accountLevelPermissions.blockPublicAccess.blockPublicPolicy
resource.s3BucketDetails.publicAccess.permissionConfiguration.accountLevelPermissions.blockPublicAccess.ignorePublicAcls
resource.s3BucketDetails.publicAccess.permissionConfiguration.accountLevelPermissions.blockPublicAccess.restrictPublicBuckets
resource.s3BucketDetails.publicAccess.permissionConfiguration.bucketLevelPermissions.accessControlList.allowsPublicReadAccess
resource.s3BucketDetails.publicAccess.permissionConfiguration.bucketLevelPermissions.accessControlList.allowsPublicWriteAccess

JSON フィールド名
resource.s3BucketDetails.publicAccess.permissionConfiguration.bucketLevelPermissions.blockPublicAccess.blockPublicAcls
resource.s3BucketDetails.publicAccess.permissionConfiguration.bucketLevelPermissions.blockPublicAccess.blockPublicPolicy
resource.s3BucketDetails.publicAccess.permissionConfiguration.bucketLevelPermissions.blockPublicAccess.ignorePublicAcls
resource.s3BucketDetails.publicAccess.permissionConfiguration.bucketLevelPermissions.blockPublicAccess.restrictPublicBuckets
resource.s3BucketDetails.publicAccess.permissionConfiguration.bucketLevelPermissions.bucketPolicy.allowsPublicReadAccess
resource.s3BucketDetails.publicAccess.permissionConfiguration.bucketLevelPermissions.bucketPolicy.allowsPublicWriteAccess
resource.s3BucketDetails.s3ObjectDetails.eTag
resource.s3BucketDetails.s3ObjectDetails.hash
resource.s3BucketDetails.s3ObjectDetails.key
resource.s3BucketDetails.s3ObjectDetails.objectArn
resource.s3BucketDetails.s3ObjectDetails.versionId
schemaVersion
service.action.awsApiCallAction.domainDetails.domain
service.action.awsApiCallAction.remoteIpDetails.country.countryCode
service.action.awsApiCallAction.remoteIpDetails.geoLocation.lat
service.action.awsApiCallAction.remoteIpDetails.geoLocation.lon
service.action.awsApiCallAction.remoteIpDetails.organization.isp

JSON フィールド名
service.action.awsApiCallAction.remoteIpDetails.organization.org
service.action.awsApiCallAction.userAgent
service.action.dnsRequestAction.blocked
service.action.dnsRequestAction.protocol
service.action.kubernetesApiCallAction.parameters
service.action.kubernetesApiCallAction.remoteIpDetails.country.countryCode
service.action.kubernetesApiCallAction.remoteIpDetails.geoLocation.lat
service.action.kubernetesApiCallAction.remoteIpDetails.geoLocation.lon
service.action.kubernetesApiCallAction.resource
service.action.kubernetesApiCallAction.resourceName
service.action.kubernetesApiCallAction.sourceIPs
service.action.kubernetesApiCallAction.subresource
service.action.kubernetesApiCallAction.userAgent
service.action.kubernetesApiCallAction.verb
service.action.kubernetesPermissionCheckedDetails.allowed
service.action.kubernetesPermissionCheckedDetails.namespace
service.action.kubernetesPermissionCheckedDetails.resource
service.action.kubernetesPermissionCheckedDetails.verb
service.action.kubernetesRoleBindingDetails.kind
service.action.kubernetesRoleBindingDetails.name

JSON フィールド名
service.action.kubernetesRoleBindingDetails.roleRefKind
service.action.kubernetesRoleBindingDetails.roleRefName
service.action.kubernetesRoleBindingDetails.uid
service.action.kubernetesRoleDetails.kind
service.action.kubernetesRoleDetails.name
service.action.kubernetesRoleDetails.uid
service.action.networkConnectionAction.localNetworkInterface
service.action.networkConnectionAction.localPortDetails.portName
service.action.networkConnectionAction.remoteIpDetails.country.countryCode
service.action.networkConnectionAction.remoteIpDetails.geoLocation.lat
service.action.networkConnectionAction.remoteIpDetails.geoLocation.lon
service.action.networkConnectionAction.remoteIpDetails.organization.isp
service.action.networkConnectionAction.remoteIpDetails.organization.org
service.action.networkConnectionAction.remotePortDetails.portName
service.action.portProbeAction.blocked
service.action.portProbeAction.portProbeDetails.localIpDetails.ipAddressV4
service.action.portProbeAction.portProbeDetails.localIpDetails.ipAddressV6
service.action.portProbeAction.portProbeDetails.localPortDetails.port
service.action.portProbeAction.portProbeDetails.localPortDetails.portName
service.action.portProbeAction.portProbeDetails.remoteIpDetails.city.cityName

JSON フィールド名
service.action.portProbeAction.portProbeDetails.remoteIpDetails.country.countryCode
service.action.portProbeAction.portProbeDetails.remoteIpDetails.country.countryName
service.action.portProbeAction.portProbeDetails.remoteIpDetails.geoLocation.lat
service.action.portProbeAction.portProbeDetails.remoteIpDetails.geoLocation.lon
service.action.portProbeAction.portProbeDetails.remoteIpDetails.ipAddressV4
service.action.portProbeAction.portProbeDetails.remoteIpDetails.ipAddressV6
service.action.portProbeAction.portProbeDetails.remoteIpDetails.organization.asn
service.action.portProbeAction.portProbeDetails.remoteIpDetails.organization.asnOrg
service.action.portProbeAction.portProbeDetails.remoteIpDetails.organization.isp
service.action.portProbeAction.portProbeDetails.remoteIpDetails.organization.org
service.action.rdsLoginAttemptAction.loginAttributes.application
service.action.rdsLoginAttemptAction.loginAttributes.failedLoginAttempts
service.action.rdsLoginAttemptAction.loginAttributes.successfulLoginAttempts
service.action.rdsLoginAttemptAction.loginAttributes.user
service.action.rdsLoginAttemptAction.remoteIpDetails.city.cityName
service.action.rdsLoginAttemptAction.remoteIpDetails.country.countryCode
service.action.rdsLoginAttemptAction.remoteIpDetails.country.countryName
service.action.rdsLoginAttemptAction.remoteIpDetails.geoLocation.lat
service.action.rdsLoginAttemptAction.remoteIpDetails.geoLocation.lon
service.action.rdsLoginAttemptAction.remoteIpDetails.ipAddressV4

JSON フィールド名
service.action.rdsLoginAttemptAction.remoteIpDetails.ipAddressV6
service.action.rdsLoginAttemptAction.remoteIpDetails.organization.asn
service.action.rdsLoginAttemptAction.remoteIpDetails.organization.asnOrg
service.action.rdsLoginAttemptAction.remoteIpDetails.organization.isp
service.action.rdsLoginAttemptAction.remoteIpDetails.organization.org
service.additionalInfo.agentDetails.agentId
service.additionalInfo.agentDetails.agentVersion
service.additionalInfo.anomalies.anomalousAPIs
service.additionalInfo.authenticationMethod
service.additionalInfo.averagePacketSizeIn
service.additionalInfo.averagePacketSizeOut
service.additionalInfo.context
service.additionalInfo.domain
service.additionalInfo.inBytes
service.additionalInfo.localNetworkInterfaceOwner
service.additionalInfo.localPort
service.additionalInfo.outBytes
service.additionalInfo.packetsIn
service.additionalInfo.packetsOut
service.additionalInfo.policyArn

JSON フィールド名
service.additionalInfo.policyName
service.additionalInfo.remotePort
service.additionalInfo.sample
service.additionalInfo.scannedPort
service.additionalInfo.threatFileSha256
service.additionalInfo.threatName
service.additionalInfo.totalBytesIn
service.additionalInfo.totalBytesOut
service.additionalInfo.type
service.additionalInfo.unusual.asnOrg
service.additionalInfo.unusual.port
service.additionalInfo.unusualProtocol
service.additionalInfo.userAgent.fullUserAgent
service.additionalInfo.userAgent.userAgentCategory
service.additionalInfo.value
service.additionalInfo.vpcOwnerAccountId
service.count
service.detection.sequence.actors.id
service.detection.sequence.actors.process.name
service.detection.sequence.actors.process.path

JSON フィールド名
service.detection.sequence.actors.process.sha256
service.detection.sequence.actors.session.createdTime
service.detection.sequence.actors.session.issuer
service.detection.sequence.actors.session.mfaStatus
service.detection.sequence.actors.session.uid
service.detection.sequence.actors.user.account.account
service.detection.sequence.actors.user.account.uid
service.detection.sequence.actors.user.credentialUid
service.detection.sequence.actors.user.name
service.detection.sequence.actors.user.type
service.detection.sequence.actors.user.uid
service.detection.sequence.additionalSequenceTypes
service.detection.sequence.description
service.detection.sequence.endpoints.autonomousSystem.name
service.detection.sequence.endpoints.autonomousSystem.number
service.detection.sequence.endpoints.connection.direction
service.detection.sequence.endpoints.domain
service.detection.sequence.endpoints.id
service.detection.sequence.endpoints.ip
service.detection.sequence.endpoints.location.city

JSON フィールド名
service.detection.sequence.endpoints.location.country
service.detection.sequence.endpoints.location.lat
service.detection.sequence.endpoints.location.lon
service.detection.sequence.endpoints.port
service.detection.sequence.resources.accountId
service.detection.sequence.resources.cloudPartition
service.detection.sequence.resources.data.accessKey.principalId
service.detection.sequence.resources.data.accessKey.userName
service.detection.sequence.resources.data.accessKey.userType
service.detection.sequence.resources.data.autoscalingAutoScalingGroup.ec2InstanceUids
service.detection.sequence.resources.data.cloudformationStack.ec2InstanceUids
service.detection.sequence.resources.data.container.image
service.detection.sequence.resources.data.container.imageUid
service.detection.sequence.resources.data.ec2Image.ec2InstanceUids
service.detection.sequence.resources.data.ec2Instance.availabilityZone
service.detection.sequence.resources.data.ec2Instance.ec2NetworkInterfaceUids
service.detection.sequence.resources.data.ec2Instance.iamInstanceProfile.arn
service.detection.sequence.resources.data.ec2Instance.iamInstanceProfile.id
service.detection.sequence.resources.data.ec2Instance.imageDescription
service.detection.sequence.resources.data.ec2Instance.instanceState

JSON フィールド名
service.detection.sequence.resources.data.ec2Instance.instanceType
service.detection.sequence.resources.data.ec2Instance.outpostArn
service.detection.sequence.resources.data.ec2Instance.platform
service.detection.sequence.resources.data.ec2Instance.productCodes.productCodeId
service.detection.sequence.resources.data.ec2Instance.productCodes.productCodeType
service.detection.sequence.resources.data.ec2LaunchTemplate.ec2InstanceUids
service.detection.sequence.resources.data.ec2LaunchTemplate.version
service.detection.sequence.resources.data.ec2NetworkInterface.ipv6Addresses
service.detection.sequence.resources.data.ec2NetworkInterface.privateIpAddresses.privateDnsName
service.detection.sequence.resources.data.ec2NetworkInterface.privateIpAddresses.privateIpAddress
service.detection.sequence.resources.data.ec2NetworkInterface.publicIp
service.detection.sequence.resources.data.ec2NetworkInterface.securityGroups.groupId
service.detection.sequence.resources.data.ec2NetworkInterface.securityGroups.groupName
service.detection.sequence.resources.data.ec2NetworkInterface.subNetId
service.detection.sequence.resources.data.ec2NetworkInterface.vpcId
service.detection.sequence.resources.data.ec2Vpc.ec2InstanceUids
service.detection.sequence.resources.data.ecsCluster.ec2InstanceUids
service.detection.sequence.resources.data.ecsCluster.status
service.detection.sequence.resources.data.ecsTask.containerUids

JSON フィールド名
service.detection.sequence.resources.data.ecsTask.createdAt
service.detection.sequence.resources.data.ecsTask.launchType
service.detection.sequence.resources.data.ecsTask.taskDefinitionArn
service.detection.sequence.resources.data.eksCluster.arn
service.detection.sequence.resources.data.eksCluster.createdAt
service.detection.sequence.resources.data.eksCluster.ec2InstanceUids
service.detection.sequence.resources.data.eksCluster.status
service.detection.sequence.resources.data.eksCluster.vpcId
service.detection.sequence.resources.data.iamInstanceProfile.ec2InstanceUids
service.detection.sequence.resources.data.iamInstanceProfile.id
service.detection.sequence.resources.data.kubernetesWorkload.containerUids
service.detection.sequence.resources.data.kubernetesWorkload.namespace
service.detection.sequence.resources.data.kubernetesWorkload.type
service.detection.sequence.resources.data.s3Bucket.accountPublicAccess.publicAclAccess
service.detection.sequence.resources.data.s3Bucket.accountPublicAccess.publicAclIgnoreBehavior
service.detection.sequence.resources.data.s3Bucket.accountPublicAccess.publicBucketRestrictBehavior
service.detection.sequence.resources.data.s3Bucket.accountPublicAccess.publicPolicyAccess
service.detection.sequence.resources.data.s3Bucket.bucketPublicAccess.publicAclAccess
service.detection.sequence.resources.data.s3Bucket.bucketPublicAccess.publicAclIgnoreBehavior

JSON フィールド名
service.detection.sequence.resources.data.s3Bucket.bucketPublicAccess.publicBucketRestrictBehavior
service.detection.sequence.resources.data.s3Bucket.bucketPublicAccess.publicPolicyAccess
service.detection.sequence.resources.data.s3Bucket.createdAt
service.detection.sequence.resources.data.s3Bucket.effectivePermission
service.detection.sequence.resources.data.s3Bucket.encryptionKeyArn
service.detection.sequence.resources.data.s3Bucket.encryptionType
service.detection.sequence.resources.data.s3Bucket.ownerId
service.detection.sequence.resources.data.s3Bucket.publicReadAccess
service.detection.sequence.resources.data.s3Bucket.publicWriteAccess
service.detection.sequence.resources.data.s3Bucket.s3ObjectUids
service.detection.sequence.resources.data.s3Object.eTag
service.detection.sequence.resources.data.s3Object.key
service.detection.sequence.resources.data.s3Object.versionId
service.detection.sequence.resources.name
service.detection.sequence.resources.region
service.detection.sequence.resources.resourceType
service.detection.sequence.resources.service
service.detection.sequence.resources.tags.key
service.detection.sequence.resources.tags.value
service.detection.sequence.resources.uid

JSON フィールド名
service.detection.sequence.sequenceIndicators.key
service.detection.sequence.sequenceIndicators.title
service.detection.sequence.sequenceIndicators.values
service.detection.sequence.signals.actorIds
service.detection.sequence.signals.count
service.detection.sequence.signals.createdAt
service.detection.sequence.signals.description
service.detection.sequence.signals.endpointIds
service.detection.sequence.signals.firstSeenAt
service.detection.sequence.signals.lastSeenAt
service.detection.sequence.signals.name
service.detection.sequence.signals.resourceUids
service.detection.sequence.signals.severity
service.detection.sequence.signals.signalIndicators.key
service.detection.sequence.signals.signalIndicators.title
service.detection.sequence.signals.signalIndicators.values
service.detection.sequence.signals.type
service.detection.sequence.signals.uid
service.detection.sequence.signals.updatedAt
service.detection.sequence.uid

JSON フィールド名
service.detectorId
service.ebsVolumeScanDetails.scanCompletedAt
service.ebsVolumeScanDetails.scanDetections.highestSeverityThreatDetails.count
service.ebsVolumeScanDetails.scanDetections.highestSeverityThreatDetails.severity
service.ebsVolumeScanDetails.scanDetections.highestSeverityThreatDetails.threatName
service.ebsVolumeScanDetails.scanDetections.scannedItemCount.files
service.ebsVolumeScanDetails.scanDetections.scannedItemCount.totalGb
service.ebsVolumeScanDetails.scanDetections.scannedItemCount.volumes
service.ebsVolumeScanDetails.scanDetections.threatDetectedByName.itemCount
service.ebsVolumeScanDetails.scanDetections.threatDetectedByName.shortened
service.ebsVolumeScanDetails.scanDetections.threatDetectedByName.threatNames.filePaths.fileName
service.ebsVolumeScanDetails.scanDetections.threatDetectedByName.threatNames.filePaths.filePath
service.ebsVolumeScanDetails.scanDetections.threatDetectedByName.threatNames.filePaths.volumeArn
service.ebsVolumeScanDetails.scanDetections.threatDetectedByName.threatNames.itemCount
service.ebsVolumeScanDetails.scanDetections.threatDetectedByName.uniqueThreatNameCount
service.ebsVolumeScanDetails.scanDetections.threatsDetectedItemCount.files
service.ebsVolumeScanDetails.scanStartedAt
service.ebsVolumeScanDetails.scanType
service.ebsVolumeScanDetails.sources

JSON フィールド名
service.eventFirstSeen
service.eventLastSeen
service.malwareScanDetails.scanCategory
service.malwareScanDetails.scanConfiguration.incrementalScanDetails.baselineResourceArn
service.malwareScanDetails.scanConfiguration.triggerType
service.malwareScanDetails.threats.count
service.malwareScanDetails.threats.hash
service.malwareScanDetails.threats.itemDetails.additionalInfo.deviceName
service.malwareScanDetails.threats.itemDetails.additionalInfo.versionId
service.malwareScanDetails.threats.itemDetails.hash
service.malwareScanDetails.threats.itemDetails.itemPath
service.malwareScanDetails.threats.itemDetails.resourceArn
service.malwareScanDetails.threats.itemPaths.hash
service.malwareScanDetails.threats.itemPaths.nestedItemPath
service.malwareScanDetails.threats.source
service.malwareScanDetails.uniqueThreatCount
service.runtimeDetails.context.addressFamily
service.runtimeDetails.context.commandLineExample
service.runtimeDetails.context.fileSystemType
service.runtimeDetails.context.flags

JSON フィールド名
service.runtimeDetails.context.ianaProtocolNumber
service.runtimeDetails.context.ldPreloadValue
service.runtimeDetails.context.libraryPath
service.runtimeDetails.context.memoryRegions
service.runtimeDetails.context.modifiedAt
service.runtimeDetails.context.modifyingProcess.euid
service.runtimeDetails.context.modifyingProcess.executablePath
service.runtimeDetails.context.modifyingProcess.executableSha256
service.runtimeDetails.context.modifyingProcess.lineage.euid
service.runtimeDetails.context.modifyingProcess.lineage.executablePath
service.runtimeDetails.context.modifyingProcess.lineage.name
service.runtimeDetails.context.modifyingProcess.lineage.namespacePid
service.runtimeDetails.context.modifyingProcess.lineage.parentUuid
service.runtimeDetails.context.modifyingProcess.lineage.pid
service.runtimeDetails.context.modifyingProcess.lineage.startTime
service.runtimeDetails.context.modifyingProcess.lineage.userId
service.runtimeDetails.context.modifyingProcess.lineage.uuid
service.runtimeDetails.context.modifyingProcess.name
service.runtimeDetails.context.modifyingProcess.namespacePid
service.runtimeDetails.context.modifyingProcess.parentUuid

JSON フィールド名
service.runtimeDetails.context.modifyingProcess.pid
service.runtimeDetails.context.modifyingProcess.pwd
service.runtimeDetails.context.modifyingProcess.startTime
service.runtimeDetails.context.modifyingProcess.user
service.runtimeDetails.context.modifyingProcess.userId
service.runtimeDetails.context.modifyingProcess.uuid
service.runtimeDetails.context.mountSource
service.runtimeDetails.context.mountTarget
service.runtimeDetails.context.relatedFilePaths
service.runtimeDetails.context.releaseAgentPath
service.runtimeDetails.context.runcBinaryPath
service.runtimeDetails.context.scriptPath
service.runtimeDetails.context.serviceName
service.runtimeDetails.context.shellHistoryFilePath
service.runtimeDetails.context.socketPath
service.runtimeDetails.context.targetProcess.euid
service.runtimeDetails.context.targetProcess.executablePath
service.runtimeDetails.context.targetProcess.executableSha256
service.runtimeDetails.context.targetProcess.lineage.euid
service.runtimeDetails.context.targetProcess.lineage.executablePath

JSON フィールド名
service.runtimeDetails.context.targetProcess.lineage.name
service.runtimeDetails.context.targetProcess.lineage.namespacePid
service.runtimeDetails.context.targetProcess.lineage.parentUuid
service.runtimeDetails.context.targetProcess.lineage.pid
service.runtimeDetails.context.targetProcess.lineage.startTime
service.runtimeDetails.context.targetProcess.lineage.userId
service.runtimeDetails.context.targetProcess.lineage.uuid
service.runtimeDetails.context.targetProcess.name
service.runtimeDetails.context.targetProcess.namespacePid
service.runtimeDetails.context.targetProcess.parentUuid
service.runtimeDetails.context.targetProcess.pid
service.runtimeDetails.context.targetProcess.pwd
service.runtimeDetails.context.targetProcess.startTime
service.runtimeDetails.context.targetProcess.user
service.runtimeDetails.context.targetProcess.userId
service.runtimeDetails.context.targetProcess.uuid
service.runtimeDetails.context.threatFilePath
service.runtimeDetails.context.toolCategory
service.runtimeDetails.context.toolName
service.runtimeDetails.process.euid

JSON フィールド名
service.runtimeDetails.process.lineage.euid
service.runtimeDetails.process.lineage.executablePath
service.runtimeDetails.process.lineage.name
service.runtimeDetails.process.lineage.namespacePid
service.runtimeDetails.process.lineage.parentUuid
service.runtimeDetails.process.lineage.pid
service.runtimeDetails.process.lineage.startTime
service.runtimeDetails.process.lineage.userId
service.runtimeDetails.process.lineage.uuid
service.runtimeDetails.process.namespacePid
service.runtimeDetails.process.parentUuid
service.runtimeDetails.process.pid
service.runtimeDetails.process.pwd
service.runtimeDetails.process.startTime
service.runtimeDetails.process.user
service.runtimeDetails.process.userId
service.runtimeDetails.process.uuid
service.userFeedback

GuardDuty の抑制ルール

抑制ルールは、フィルター属性と値の組み合わせで構成される基準のセットで、指定した条件に一致する新しい検出結果を自動的にアーカイブして検出結果をフィルタリングするために使用する条件のセットのことです。抑制ルールを使用して、重要ではない検出結果、誤検出の検出結果、対応を行わない脅威をフィルタリングすることにより、環境に最も影響があるセキュリティの脅威を認識しやすくなります。

抑制ルールを作成すると、その抑制ルールが適用されている限り、ルールで定義された条件に一致する新しい検出結果が自動的にアーカイブされます。既存のフィルターを使用して抑制ルールを作成したり、定義した新しいフィルターから抑制ルールを作成することもできます。検出結果タイプ全体を抑制するよう抑制ルールを設定したり、特定の検出結果タイプの特定のインスタンスのみを抑制する、よりきめ細かいフィルター条件を定義したりできます。抑制ルールはいつでも編集できます。

抑制された検出結果は AWS Security Hub CSPM、Amazon Simple Storage Service、Amazon Detective、または Amazon EventBridge に送信されないため、Security Hub CSPM、サードパーティー SIEM、またはその他のアラートおよびチケット発行アプリケーションを介して GuardDuty の検出結果を使用する場合、検出結果のノイズレベルが低減されます。Malware Protection for EC2 を有効にした場合、抑制された GuardDuty の検出結果ではマルウェアスキャンは開始されません。

GuardDuty は、抑制ルールに一致している場合でも検出結果を生成します。ただし、これらの検出結果は自動的に [archived] (アーカイブ) としてマークされます。アーカイブされた検出結果は 90 日間 GuardDuty に保存され、その期間中いつでも表示することができます。抑制された検出結果は、GuardDuty コンソールで、検出結果の表から [Archived] (アーカイブ) を選択するか、`service.archived` 等式の `findingCriteria` 条件を true にした ListFindings API を使用して GuardDuty API を通じて表示できます。

> **Note**
>
> マルチアカウント環境では、GuardDuty 管理者のみが抑制ルールを作成できます。

Extended Threat Detection での抑制ルールの使用

GuardDuty Extended Threat Detection は、 AWS アカウント内のデータソース、複数のタイプの AWS リソース、および時間にわたるマルチステージ攻撃を自動的に検出します。さまざまなデータソースのイベントを関連付けて、 AWS 環境に対する潜在的な脅威として存在するシナリオを特定し、攻撃シーケンスの検出結果を生成します。詳細については、「Extended Threat Detection の仕組み」を参照してください。

検出結果をアーカイブする抑制ルールを作成すると、Extended Threat Detection は、攻撃シーケンスのイベントを関連付けるときに、これらのアーカイブされた検出結果を使用できません。幅広い抑制ルールは、GuardDuty がマルチステージ攻撃の検出と一致する動作を検出する能力に影響を与える可能性があります。抑制ルールのためにアーカイブされた検出結果は、攻撃シーケンスのシグナルとは見なされません。例えば、特定の既知のアクティビティをターゲットにするのではなく、すべての EKS クラスター関連の検出結果をアーカイブする抑制ルールを作成すると、GuardDuty はそれらの検出結果を使用して、脅威アクターがコンテナを悪用し、特権トークンを取得し、機密リソースにアクセスする攻撃シーケンスを検出できなくなります。

GuardDuty からの次の推奨事項を検討します。

- サプレッションルールを引き続き使用して、既知の信頼できるアクティビティからのアラートを減らします。
- 抑制ルールは、GuardDuty が検出結果を生成しないようにする特定の動作に焦点を当ててください。

抑制ルールの一般的ユースケースとその例

次の検出結果タイプには、抑制ルールを適用する一般的ユースケースがあります。検出結果名を選択すると、その検出結果の詳細を確認できます。ユースケースの説明を確認して、その検出結果タイプの抑制ルールを作成するかどうかを決定します。

> **Important**
>
> GuardDuty は、環境で誤検出を繰り返し特定した検出結果に対してのみ、抑制ルールを作成することをお勧めします。

- UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.OutsideAWS - VPC のインターネットゲートウェイからではなく、オンプレミスのゲートウェイからインターネットへのトラフィックをルーティングするように VPC ネットワークが設定されている場合に、生成された検出結果を自動的にアーカイブするために抑制ルールを使用します。

  この検出結果が生成されるのは、VPC インターネットゲートウェイ (IGW) からではなく、オンプレミスのゲートウェイから排出され、インターネットトラフィックがルーティングされるように、ネットワークが構成されている場合です。AWS Outposts や VPC VPN 接続などの一般的な構成では、このようにトラフィックがルーティングされる可能性があります これが予期した動作である場合は、抑制ルールを使用して、2 つのフィルター条件で構成されるルールを作成することをお勧

めします。1 つ目の条件では、[finding type] (結果タイプ) に UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.OutsideAWS を使用します。2 番目のフィルター条件は、オンプレミスインターネットゲートウェイの IP アドレスまたは CIDR 範囲を持つ [API caller IPv4 address] (API 発信者の IPv4 アドレス) です。次の例は、API 発信者の IP アドレスに基づいてこの検出結果タイプを抑制するために使用するフィルターを示しています。

```
Finding type: UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.OutsideAWS
 API caller IPv4 address: 198.51.100.6
```

> **Note**
>
> 複数の API 発信者 IPs を含めるには、各 IP アドレスを Equals または NotEquals 条件の値のリストに追加するか、一致または NotMatches 条件にワイルドカードパターンを使用できます。Equals または NotEquals の値は、CIDR 範囲でもかまいません。CIDR 範囲は、その範囲内の任意の IP アドレスに一致します。

- Recon:EC2/Portscan - 脆弱性評価アプリケーションを使用する場合に、検出結果を自動的にアーカイブするために抑制ルールを使用します。

  抑制ルールは、2 つのフィルター条件で構成する必要があります。1 つ目の条件では、[Finding type] (結果タイプ) 属性に Recon:EC2/Portscan という値を使用します。2 番目のフィルター条件は、これらの脆弱性評価ツールをホストする 1 つ以上のインスタンスと一致する必要があります。これらのツールをホストするインスタンスで識別可能な条件に応じて、[Instance image ID] (インスタンスイメージ ID) 属性または [Tag] (タグ) 値の属性のいずれかを使用できます。次の例は、特定の AMI を持つインスタンスに基づいてこの検出結果タイプを抑制するために使用するフィルターを示しています。

  ```
  Finding type: Recon:EC2/Portscan Instance image ID: ami-999999999
  ```

- UnauthorizedAccess:EC2/SSHBruteForce - 踏み台インスタンスをターゲットとする場合に、検出結果を自動的にアーカイブするために抑制ルールを使用します。

  ブルートフォース試行のターゲットが踏み台ホストである場合、 AWS これは環境に対して予想される動作を表す可能性があります。このような状況が発生した場合は、この検出結果に対する抑制ルールを設定することをお勧めします。抑制ルールは、2 つのフィルター条件で構成する必要があります。1 つ目の条件では、[Finding type] (結果タイプ) 属性に UnauthorizedAccess:EC2/SSHBruteForce という値を使用します。2 番目のフィルター条件は、要塞ホストとして機能する 1 つ以上のインスタンスと一致する必要があります。これらのツールをホストするインスタンスで

識別可能な条件に応じて、[Instance image ID] (インスタンスイメージ ID) 属性または [Tag] (タグ) 値の属性のいずれかを使用できます。次の例は、特定のタグ値を持つインスタンスに基づいてこの検出結果タイプを抑制するために使用するフィルターを示しています。

```
Finding type: UnauthorizedAccess:EC2/SSHBruteForce Instance tag value: devops
```

- Recon:EC2/PortProbeUnprotectedPort - 意図的に公開しているインスタンスをターゲットとする場合に、検出結果を自動的にアーカイブするために抑制ルールを使用します。

  ウェブサーバーをホストしている場合など、インスタンスが意図的に公開される場合があります。ご使用の AWS 環境でその場合は、この検出結果の抑制ルールを設定することをお勧めします。抑制ルールは、2 つのフィルター条件で構成する必要があります。1 つ目の条件では、[Finding type] (結果タイプ) 属性に `Recon:EC2/PortProbeUnprotectedPort` という値を使用します。2 番目のフィルター条件は、要塞ホストとして機能する 1 つ以上のインスタンスと一致する必要があります。これらのツールをホストするインスタンスで識別可能な条件に応じて、[Instance image ID] (インスタンスイメージ ID) 属性または [Tag] (タグ) 値の属性のいずれかを使用できます。次の例は、コンソール内の特定のタグキーを持つインスタンスに基づいてこの検出結果タイプを抑制するために使用するフィルターを示しています。

  ```
  Finding type: Recon:EC2/PortProbeUnprotectedPort Instance tag key: prod
  ```

Runtime Monitoring 検出結果の抑制ルールの推奨

- PrivilegeEscalation:Runtime/DockerSocketAccessed はコンテナ内のプロセスが Docker ソケットと通信するときに生成されます。環境内に、正当な理由で Docker ソケットにアクセスする必要があるコンテナが存在する可能性があります。このようなコンテナからアクセスすると、PrivilegeEscalation:Runtime/DockerSocketAccessed 検出結果が生成されます。ご使用の AWS 環境でこれが当てはまる場合は、この検出結果タイプの抑制ルールを設定することをお勧めします。1 つ目の条件では、値が `PrivilegeEscalation:Runtime/DockerSocketAccessed` に等しい [検出結果タイプ] フィールドを使用する必要があります。2 番目のフィルター条件は、生成された検出結果のプロセスの `executablePath` と同じ値を持つ [実行可能ファイルのパス] フィールドです。別の方法として、2 番目のフィルター条件では、生成された検出結果のプロセスの `executableSha256` と同じ値を持つ [実行可能ファイル SHA-256] フィールドを使用できます。
- Kubernetes クラスターは、独自の DNS サーバーを `coredns` などのポッドとして実行します。そのため、ポッドからの DNS 検索ごとに、GuardDuty は 2 つの DNS イベントをキャプチャしま

す。1 つはポッドから、もう 1 つはサーバーポッドからキャプチャされます。これにより、以下の DNS 検出結果に重複が生じる可能性があります。

- Backdoor:Runtime/C&CActivity.B!DNS
- CryptoCurrency:Runtime/BitcoinTool.B!DNS
- Impact:Runtime/AbusedDomainRequest.Reputation
- Impact:Runtime/BitcoinDomainRequest.Reputation
- Impact:Runtime/MaliciousDomainRequest.Reputation
- Impact:Runtime/SuspiciousDomainRequest.Reputation
- Trojan:Runtime/BlackholeTraffic!DNS
- Trojan:Runtime/DGADomainRequest.C!DNS
- Trojan:Runtime/DriveBySourceTraffic!DNS
- Trojan:Runtime/DropPoint!DNS
- Trojan:Runtime/PhishingDomainRequest!DNS

重複した検出結果には、DNS サーバーポッドに対応するポッド、コンテナ、プロセスの詳細が含まれます。これらのフィールドを使用して、重複検出結果を抑制する抑制ルールをセットアップできます。最初のフィルター条件では、このセクションで前述した検出結果リストの DNS 検出結果タイプと同じ値の [検出結果タイプ] フィールドを使用する必要があります。2 つ目のフィルター条件は、お使いの DNS サーバー `executablePath` と同じ値の [実行可能ファイルのパス] か、生成された検出結果の DNS サーバー `executableSHA256` と同じ値の [実行可能ファイル SHA-256] のいずれかです。オプションの 3 番目のフィルター条件として、生成された検出結果に含まれる DNS サーバーポッドのコンテナイメージと同じ値の [Kubernetes コンテナイメージ] フィールドを使用できます。

GuardDuty で抑制ルールを作成する

抑制ルールは、フィルター属性の使用と、GuardDuty に検出結果タイプを生成させたくない値の指定を含む一連の条件です。この条件に一致する検出結果タイプは自動的にアーカイブされます。ノイズを減らすために、抑制された検出結果は、統合 AWS のサービス できる に送信されません。抑制ルールの作成の一般的なユースケースの詳細については、「抑制ルール」を参照してください。

GuardDuty コンソールのサプレッションルールページを使用して、サプレッションルールを視覚化、作成、管理できます。抑制ルールは、既存の保存済みフィルターから生成することもできます。フィルター作成の詳細については、「GuardDuty での検出結果のフィルタリング」を参照してください。

フィルター条件には、等号演算子と NotEquals 演算子を使用した完全一致、一致演算子と NotMatches 演算子を使用したワイルドカード一致、GreaterThan、GreaterThanEquals、LessThan、LessThanEquals 演算子を使用した比較一致を含めることができます。使用可能な演算子の詳細については、Conditions「」ページを参照してください。

任意のアクセス方法を選択して、GuardDuty 検出結果タイプの抑制ルールを作成します。

Console

コンソールを使用して抑制ルールを作成するには:

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. 抑制ルールページで、抑制ルールの作成をクリックして、抑制ルールの作成フォームを開きます。
3. 抑制ルールの名前を入力します。名前は 3～64 文字にする必要があります。有効な文字は、a-z、A～Z、0～9、ピリオド ( )、ハイフン (-)、アンダースコア (_) です。
4. 説明はオプションです。説明を入力する場合、最大 512 文字まで使用できます。有効な文字は、a～z、A～Z、0～9、ピリオド (.)、ハイフン (-)、コロン (:)、括弧 ({}()[])、スラッシュ (/)、スペースです。
5. Rank はオプションです。1 からフィルターと抑制ルールの合計数、さらに 1 までの数値を指定できます。
6. 属性セクションで、ドロップダウンからキーと演算子を選択します。
7. 選択したキーに基づいて、datepicker から「文字列」または「日付」のいずれかの値を入力します。文字列値の場合は、テキストを入力して Enter キーを押します。文字列値の場合、複数の値を追加できます。
8. 追加の条件を追加するには、条件を追加 を選択して、キー、演算子、値 (複数可) の別のセットを追加します。
9. 抑制ルールを作成して保存するには、抑制ルールの作成を選択します。

また、既存の保存済みフィルターから抑制ルールを作成できます。フィルター作成の詳細については、「GuardDuty での検出結果のフィルタリング」を参照してください。

保存済みフィルターから抑制ルールを作成するには、次の手順を実行します。

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

2. [検出結果] ページで、[保存されたルール] メニューから、保存されたフィルターセットルールを選択します。これにより、フィルターセットと条件に一致する検出結果が自動的に表示されます。
3. この保存されたルールにフィルター条件を追加することもできます。フィルター基準を追加する必要がない場合は、この手順をスキップします。1 つ以上のフィルター条件を追加するには、「Adding filters on Findings page」のステップ 3 ~ 7 に従い、次のステップに進みます。
4. フィルター条件を追加し、フィルタリングされた結果が要件を満たしていることを確認したら、[抑制ルールの作成] を選択します。
5. 抑制ルールの[名前] を入力します。名前は 3~64 文字にする必要があります。有効な文字は、a-z、A~Z、0~9、ピリオド ( )、ハイフン (-)、アンダースコア (_) です。
6. 説明はオプションです。説明を入力する場合、最大 512 文字まで使用できます。
7. [作成] を選択します。
8. 保存したルールにフィルター条件を追加する必要がある場合は、ステップ 4~7 に従ってフィルターを作成します。

API/CLI

API を使用して抑制ルールを作成するには

1. 抑制ルールは、CreateFilter API を使用して作成できます。これを行うには、次に示す例の形式に従って JSON ファイルでフィルター条件を指定します。次の例では、`test.example.com` ドメインへの DNS リクエストが行われた未アーカイブの重大度の低い検出結果を抑制します。重大度が中の検出結果の場合、入力リストは `["4", "5", "7"]` になります。重大度が高の検出結果の場合、入力リストは `["6", "7", "8"]` になります。重大度が高の検出結果の場合、入力リストは `["9", "10"]` になります。リスト内の任意の 1 つの値に基づいてフィルターすることもできます。

   次の例では、関数名のプレフィックスMyFunc」を持つ Lambda 関数の重要度の低い検出結果と、プレフィックスがTestTag」ではない関数タグのフィルターを追加します。

```
{
    "Criterion": {
        "service.action.dnsRequestAction.domain": {
            "Equals": [
                "test.example.com"
```

```
            ]
        },
        "severity": {
            "Equals": [
                "1",
                "2",
                "3"
            ]
        }
    }
}
```

サプレッションルールは、ワイルドカード文字 * と を使用して作成できます。。 フィルターのワイルドカードは、Matches 演算子と NotMatches 演算子のみを使用してサポートされます。任意の数の文字を一致させるには、属性値に * を使用し、1 文字を一致させるには、 を使用できますか? 属性値の 。フィルタは、1 つのワイルドカード条件で最大 5 つの属性をサポートし、1 つの属性内で最大 5 つのワイルドカード文字をサポートします。次の例では、プレフィックスMyFunc」に一致する Lambda 名のフィルターを追加しますが、プレフィックスとしてTestTag」、その後に 0 ~ 2 文字のタグを持つ Lambda 関数は追加しません。

```
{
    "Criterion": {
        "resource.lambdaDetails.functionName": {
            "Matches": [
                "MyFunc*"
            ]
        },
        "resource.lambdaDetails.tags.key": {
            "NotMatches": [
                "TestTag??"
            ]
        }
    }
}
```

JSON のフィールド名とそれに相当するコンソールのフィールド名の一覧については、「GuardDuty のプロパティフィルター」を参照してください。

フィルター基準をテストするには、ListFindings API で同じ JSON 基準を使用し、正しい検出結果が選択されていることを確認します。を使用してフィルター条件をテストするには、独自の detectorId と .json ファイルを使用して例 AWS CLI に従います。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty list-detector
```

```
aws guardduty list-findings \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--region us-east-1 \
--finding-criteria file://criteria.json
```

> **Note**
>
> ワイルドカードマッチングは、ListFindings および GetFindingsStatistics では使用できません。ワイルドカードを含む条件は、ListFindings と GetFindingsStatistics を使用して検証することはできません。

2. 抑制ルールとして使用するフィルターを CreateFilter API を使用してアップロードするか、AWS CLI で次の例に従って独自のディテクター ID、抑制ルール名、.json ファイルを使用してアップロードします。

   アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

   ```
   aws guardduty create-filter \
   --detector-id 12abc34d567e8fa901bc2d34e56789f0 \
   --region us-east-1 \
   --action ARCHIVE \
   --name yourfiltername \
   --finding-criteria file://criteria.json
   ```

ListFilter API を使用して、プログラムでフィルターのリストを表示できます。GetFilter API にフィルター名を指定すると、個々のフィルターの詳細を表示できます。UpdateFilter API を使用してフィルターを更新するか、DeleteFilter API を使用してフィルターを削除します。

GuardDuty での抑制ルールの更新

このセクションでは、特定の の AWS アカウント で抑制ルールを更新する手順について説明します AWS リージョン。

既存の抑制ルールは、GuardDuty コンソールの抑制ルールページから更新できます。GuardDuty は、GuardDuty コンソールから、または GuardDuty CLI/API を使用して、抑制フィルターの説明、ランク、フィルター条件の更新をサポートしています。 GuardDuty 抑制ルールの更新は、説明、ランク、基準のフィールド値に対する と同じ制限に従います抑制ルールを作成する。

メンバーアカウントの場合、管理者アカウントが代わりにこのアクションを実行できます。詳細については、「管理者アカウントとメンバーアカウントの関係」を参照してください。

任意のアクセス方法を選択して、GuardDuty 検出結果タイプの抑制ルールを削除します。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. 抑制ルールページで、更新する抑制ルールを選択します。
3. Actions ドロップダウンから、Update suppression rule を選択します。
4. これにより、既存の抑制ルールフォームが開きます。
5. 必要に応じて、説明、ランク、属性セクションを変更します。
6. サプレッションルールを更新する を選択して、サプレッションルールを更新します。

API/CLI

API を使用して抑制ルールを更新するには:

1. UpdateFilter API を使用して抑制ルールを更新できます。UpdateFilter API を使用して更新できるのは、説明、ランク、基準のみです。これら 3 つのフィールドはすべてオプションです。

2. 既存のフィルターを更新するには、更新する予定のフィルターの名前が必要です。
3. 既存の条件を更新する場合は、フィルターを最初に作成した方法と同様の更新された条件を使用して JSON ファイルを作成します。test.example.com ドメインへの DNS リクエストがあるアーカイブされていない重要度の低い検出結果を抑制する基準の例。重要度が中程度の検出結果の場合、入力リストは ["4"、"5"、"7"] になります。重要度の高い検出結果の場合、入力リストは ["6"、"7"、"8"] になります。重大な重要度の検出結果の場合、入力リストは ["9", "10"] になります。リスト内の任意の 1 つの値に基づいてフィルターすることもできます。次の例では、重大度の低い検出結果のフィルターを追加します。

```
{
    "Criterion": {
        "service.action.dnsRequestAction.domain": {
            "Equals": [
                "test.example.com"
            ]
        },
        "severity": {
            "Equals": [
                "1",
                "2",
                "3"
            ]
        }
    }
}
```

JSON のフィールド名とそれに相当するコンソールのフィールド名の一覧については、「GuardDuty のプロパティフィルター」を参照してください。

フィルター基準をテストするには、ListFindings API で同じ JSON 基準を使用し、正しい検出結果が選択されていることを確認します。を使用してフィルター条件をテストするには、独自の detectorId と .json ファイルを使用して例 AWS CLI に従います。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

```
aws guardduty list-detectors --region us-east-1
```

4. 説明を更新する場合は、CLI 呼び出しに説明パラメータを含めることができます。
5. ランクを更新する場合は、CLI 呼び出しに rank パラメータを含めることができます。
6. 抑制フィルターから通常のフィルターに更新する場合は、CLI 呼び出しでアクションパラメータと値を ARCHIVE として使用します。
7. 独自のディテクター ID、抑制ルールの名前、.json ファイルを使用して、 AWS CLI 既存のフィルター API または を更新します。
8. 以下は、上記のすべてのパラメータを更新する CLI の例です。コマンド - から、ユースケース用に更新する特定のパラメータを選択できます。

```
aws guardduty update-filter \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--region us-east-1 \
--action ARCHIVE \
--rank 1 \
--description "Updated description" \
--finding-criteria file://criteria.json
```

GuardDuty で抑制ルールを削除する

このセクションでは、 AWS アカウント 特定の の で抑制ルールを削除する手順について説明します AWS リージョン。

環境内で期待される動作を示さなくなった抑制ルールを削除することもできます。GuardDuty が検出結果タイプを生成できるように、関連付けられた検出結果タイプを抑制したくなくなることがあります。

メンバーアカウントの場合、管理者アカウントが代わりにこのアクションを実行できます。詳細については、「管理者アカウントとメンバーアカウントの関係」を参照してください。

任意のアクセス方法を選択して、GuardDuty 検出結果タイプの抑制ルールを削除します。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. 抑制ルールページで、削除する抑制ルールを選択します。
3. Actions ドロップダウンから、Delete suppression rule を選択します。
4. 確認ポップアップが表示されます。削除を選択して削除を続行します。または、キャンセルを選択してオペレーションをキャンセルします。

API/CLI

DeleteFilter API を実行します。フィルター名と、特定のリージョンの関連付けられたディテクター ID を指定します。

または、#でフォーマットされた値を置き換えることで、次の AWS CLI 例を使用できます。

```
aws guardduty delete-filter \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--filter-name filterName \
--region us-east-1
```

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

抑制ルールメトリクス

Amazon GuardDuty を使用すると、各抑制ルールが結果に一致してアーカイブする頻度をモニタリングできます。GuardDuty は、これらのメトリクスを AWS/GuardDuty 名前空間の Amazon CloudWatch に発行します。 GuardDuty

⚠ GuardDuty は、一致した検出結果と抑制された検出結果を異なる方法でカウントします

GuardDuty は、検出結果が抑制ルールまたは NOOP フィルター (アクションのないフィルター) と一致すると、FindingsMatchedByFilter を出力します。このメトリクスは、フィルターのアクションに関係なく増加します。GuardDuty は、ルールのアクションが

Archive で、ルールがその検出結果の最も優先度の高いアーカイブルールである場合にのみ FindingsSuppressedByFilter を発行します。

GuardDuty は、アカウント設定に基づいて抑制ルールメトリクスを発行します。

- スタンドアロンアカウント (組織の一部ではない) の場合、GuardDuty はそのアカウントにメトリクスを発行します。
- 組織の一部であるアカウントの場合、GuardDuty は委任 GuardDuty 管理者アカウントにメトリクスを発行します。GuardDuty での管理者アカウントとメンバーアカウントの仕組みの詳細については、「」を参照してくださいAmazon GuardDuty での複数のアカウントの管理。

メトリクスの詳細

次の表は、GuardDuty が抑制ルール用に発行する Amazon CloudWatch メトリクスを示しています。

メトリクス	単位	説明
`FindingsMatchedByFilter`	カウント	抑制ルールまたは NOOP フィルターの基準に一致した検出結果の数。検出結果は複数の抑制ルールまたは NOOP フィルターと一致でき、GuardDuty は一致ごとに 1 回カウントします。一致ごとに 1 つのデータポイントが生成されます。
`FindingsSuppressedByFilter`	カウント	抑制ルールによってアーカイブされた検出結果の数。複数のルールが同じ結果に一致する場合、GuardDuty はRank最小値のルールのみをカウントします。抑制ごとに 1 つのデータポイントが生成されます。

Amazon CloudWatch で抑制ルールメトリクスを表示する

抑制ルールメトリクスは、Amazon CloudWatch コンソールの AWS/GuardDuty 名前空間で、FilterName ディメンション別にグループ化されています。`FilterName` ディメンションの値は、抑制ルールまたはフィルターの名前です。Amazon CloudWatch でのメトリクスの検索とグラフ化の手順については、Amazon CloudWatch ユーザーガイド」の「使用可能なメトリクスの表示」を参照してください。

Customizing threat detection with entity lists and IP address lists

Amazon GuardDuty は、VPC フローログ、 AWS CloudTrail イベントログ、DNS ログを分析して処理することで、 AWS 環境のセキュリティをモニタリングします。1 つ以上のユースケースに焦点を当てた GuardDuty 保護プラン (を除くランタイムモニタリング) を有効にすることで、GuardDuty 内でモニタリング機能を拡張できます。

リストを使用することで、GuardDuty は環境における脅威検出の範囲をカスタマイズするのに役立ちます。GuardDuty を設定して、信頼できるソースからの検出結果の生成を停止し、脅威リストに登録された既知の悪意のあるソースからの検出結果を生成することができます。GuardDuty では従来の IP アドレスリストが引き続きサポートされるほか、IP アドレス、ドメイン、またはその両方を格納できるエンティティリスト (推奨) にもサポートが拡張されています。

トピック

- エンティティリストと IP アドレスリストについて理解する
- GuardDuty リストに関する重要な考慮事項
- リストフォーマット
- リストのステータスについて理解する
- エンティティリストと IP アドレスリストの前提条件の設定
- エンティティリストまたは IP リストの追加とアクティブ化
- エンティティリストまたは IP アドレスリストの更新
- エンティティリストまたは IP アドレスリストの非アクティブ化
- エンティティリストまたは IP アドレスリストの削除

エンティティリストと IP アドレスリストについて理解する

GuardDuty は、エンティティリスト (推奨) と IP リストの 2 つの実装アプローチを提供します。どちらのアプローチも、信頼できるソースを指定するのに役立ちます。これにより、GuardDuty はその

ソースを検出結果に含めなくなり、GuardDuty が検出結果を生成する際に参照する既知の脅威から除外できるようになります。

エンティティリストは、IP アドレス、ドメイン名、および SHA-256 ファイルハッシュをサポートします。単一の IAM 権限による Amazon Simple Storage Service (Amazon S3) への直接アクセスを利用しており、複数のリージョンにわたる IAM ポリシーのサイズ制限に影響を与えません。

IP リストは IP アドレスのみをサポートし、GuardDuty サービスにリンクされたロール (SLR) (SLR) を使用します。そのため、リージョンごとに IAM ポリシーの更新が必要となり、IAM ポリシーのサイズ制限に影響する可能性があります。

信頼できるリスト (エンティティリストと IP アドレスリストの両方) には、 AWS インフラストラクチャとの安全な通信のために信頼できるエントリが含まれています。GuardDuty は、信頼できるソースにリストされているエントリに対しては検出結果を生成しません。どの時点でも、リージョン AWS アカウント ごとに 1 つの信頼されたエンティティリストと 1 つの信頼された IP アドレスリストのみを追加できます。

脅威リスト (エンティティリストと IP アドレスリストの両方) には、既知の悪意のあるソースとして特定したエントリが含まれます。GuardDuty がこれらのソースに関連するアクティビティを検出すると、潜在的なセキュリティ問題について警告する調査結果を生成します。独自の脅威リストを作成したり、サードパーティの脅威インテリジェンスフィードを組み込んだりできます。このリストは、サードパーティの脅威インテリジェンスによって提供されるか、あるいは組織専用に特別に生成することができます。GuardDuty は、疑わしいアクティビティの可能性に基づいて検出結果を生成するだけでなく、脅威リストのエントリに関連するアクティビティに基づいても検出結果も生成します。いつでも、リージョン AWS アカウント ごとに ごとに最大 6 つの脅威エンティティリストと脅威 IP アドレスリストをアップロードできます。

> **Note**
>
> IP アドレスリストからエンティティリストに移行するには、 エンティティリストの前提条件 に従い、必要なエンティティリストを追加してアクティブ化します。その後、対応する IP アドレスリストを非アクティブ化または削除できます。

GuardDuty リストに関する重要な考慮事項

リストの操作を始める前に、以下の注意事項をお読みください。

- IP アドレスリストとエンティティリストは、パブリックにルーティング可能な IP アドレスとドメインを宛先とするトラフィックにのみ適用されます。
- エンティティリストでは、エントリは CloudTrail、Amazon VPC の VPC フローログ、Route53 Resolver DNS クエリログの検出結果に適用されます。

  IP アドレスリストのエントリは、CloudTrail と Amazon VPC の VPC フローログの検出結果に適用されますが、Route 53 Resolver DNS クエリログの検出結果には適用されません。
- 信頼できるリストと脅威リストの両方に同じ IP アドレスまたはドメインが含まれている場合、信頼できるリストのエントリが優先されます。GuardDuty は、このエントリに関連するアクティビティがある場合は、検出結果を生成しません。
- マルチアカウント環境でリストを管理できるのは、GuardDuty 管理者アカウントのみです。この設定は、メンバーアカウントに自動的に適用されます。GuardDuty は管理者アカウントの脅威ソースからの、既知の悪意ある IP アドレス (およびドメイン) を含むアクティビティに基づく検出結果を生成しますが、管理者アカウントの信頼できるソースからの IP アドレス (およびドメイン) を含むアクティビティに基づく検出結果は生成しません。詳細については、「Amazon GuardDuty での複数のアカウントの管理」を参照してください。
- IPv4 アドレスのみ受け入れられます。IPv6 アドレスはサポートされません。
- SHA-256 ファイルハッシュは、脅威エンティティリストでのみサポートされています。信頼されたエンティティリストまたは IP アドレスリストにファイルハッシュを含めることはできません。
- エンティティリストまたは IP アドレスリストをアクティブ化、非アクティブ化、または削除すると、プロセスは 15 分以内に完了すると推定されます。シナリオによっては、このプロセスが完了するまでに最大 40 分かかることがあります。
- GuardDuty は、リストのステータスがアクティブになった場合にのみ、脅威検出にリストを使用します。
- リストの S3 バケットの場所でエントリを追加または更新するたびに、リストを再度アクティブ化する必要があります。詳細については、「エンティティリストまたは IP アドレスリストの更新」を参照してください。
- エンティティリストと IP アドレスのクォータは異なります。詳細については、「GuardDuty クォータ」を参照してください。

リストフォーマット

GuardDuty は、リストとエンティティリストに複数のファイル形式 (ファイルあたり最大 35 MB) を受け入れます。各形式には固有の要件と機能があります。

プレーンテキスト (TXT)

この形式は、IP アドレス、CIDR 範囲、ドメイン名、および SHA-256 ファイルハッシュをサポートします。SHA-256 ファイルハッシュは、脅威エンティティリストでのみサポートされています。各エントリは別々の行に表示される必要があります。

Exampleエンティティリストの例

```
192.0.2.1
192.0.2.0/24
example.com
example.org
*.example.org
```

Exampleファイルハッシュを含む脅威エンティティリストの例

```
192.0.2.1
192.0.2.0/24
example.com
example.org
*.example.org
a665a45920422f9d417e4867efdc4fb8a04a1f3fff1fa07e998e86f7f7a27ae3
```

Example IP アドレスリストの例

```
192.0.2.0/24
198.51.100.1
203.0.113.1
```

脅威情報構造化記述形式 (STIX)

この形式は、IP アドレス、CIDR ブロック、ドメイン名、および SHA-256 ファイルハッシュをサポートします。STIX では、脅威インテリジェンスを使って追加のコンテキストを含めることができます。GuardDuty は、STIX インジケータから IP アドレス、CIDR 範囲、ドメイン名、および SHA-256 ファイルハッシュを処理します。SHA-256 ファイルハッシュは、脅威エンティティリストでのみサポートされています。

Exampleエンティティリストの例

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<stix:STIX_Package
    xmlns:cyboxCommon="http://cybox.mitre.org/common-2"
    xmlns:cybox="http://cybox.mitre.org/cybox-2"
    xmlns:cyboxVocabs="http://cybox.mitre.org/default_vocabularies-2"
    xmlns:stix="http://stix.mitre.org/stix-1"
    xmlns:indicator="http://stix.mitre.org/Indicator-2"
    xmlns:stixCommon="http://stix.mitre.org/common-1"
    xmlns:stixVocabs="http://stix.mitre.org/default_vocabularies-1"
    xmlns:DomainNameObj="http://cybox.mitre.org/objects#DomainNameObject-1"
    id="example:Package-a1b2c3d4-1111-2222-3333-444455556666"
    version="1.2">
    <stix:Indicators>
        <stix:Indicator
            id="example:indicator-a1b2c3d4-aaaa-bbbb-cccc-ddddeeeeffff"
            timestamp="2025-08-12T00:00:00Z"
            xsi:type="indicator:IndicatorType"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
            <indicator:Title>Malicious domain observed Example</indicator:Title>
            <indicator:Type xsi:type="stixVocabs:IndicatorTypeVocab-1.1">Domain
 Watchlist</indicator:Type>
            <indicator:Observable id="example:Observable-0000-1111-2222-3333">
                <cybox:Object id="example:Object-0000-1111-2222-3333">
                    <cybox:Properties xsi:type="DomainNameObj:DomainNameObjectType">
                        <DomainNameObj:Value condition="Equals">bad.example.com</
DomainNameObj:Value>
                    </cybox:Properties>
                </cybox:Object>
            </indicator:Observable>
        </stix:Indicator>
    </stix:Indicators>
</stix:STIX_Package>
```

Exampleファイルハッシュを含む脅威エンティティリストの例

```
<?xml version="1.0" encoding="UTF-8"?>
<stix:STIX_Package
    xmlns:cyboxCommon="http://cybox.mitre.org/common-2"
    xmlns:cybox="http://cybox.mitre.org/cybox-2"
    xmlns:cyboxVocabs="http://cybox.mitre.org/default_vocabularies-2"
    xmlns:stix="http://stix.mitre.org/stix-1"
    xmlns:indicator="http://stix.mitre.org/Indicator-2"
    xmlns:stixCommon="http://stix.mitre.org/common-1"
    xmlns:stixVocabs="http://stix.mitre.org/default_vocabularies-1"
```

```
    xmlns:DomainNameObj="http://cybox.mitre.org/objects#DomainNameObject-1"
    xmlns:FileObj="http://cybox.mitre.org/objects#FileObject-2"
    id="example:Package-a1b2c3d4-1111-2222-3333-444455556666"
    version="1.2">
    <stix:Indicators>
        <stix:Indicator
            id="example:indicator-a1b2c3d4-aaaa-bbbb-cccc-ddddeeeeffff"
            timestamp="2025-08-12T00:00:00Z"
            xsi:type="indicator:IndicatorType"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
            <indicator:Title>Malicious domain observed Example</indicator:Title>
            <indicator:Type xsi:type="stixVocabs:IndicatorTypeVocab-1.1">Domain
 Watchlist</indicator:Type>
            <indicator:Observable id="example:Observable-0000-1111-2222-3333">
                <cybox:Object id="example:Object-0000-1111-2222-3333">
                    <cybox:Properties xsi:type="DomainNameObj:DomainNameObjectType">
                        <DomainNameObj:Value condition="Equals">bad.example.com</
DomainNameObj:Value>
                    </cybox:Properties>
                </cybox:Object>
            </indicator:Observable>
        </stix:Indicator>
        <stix:Indicator
            id="example:indicator-a1b2c3d4-eeee-ffff-0000-111122223333"
            timestamp="2025-08-12T00:00:00Z"
            xsi:type="indicator:IndicatorType"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
            <indicator:Title>File hash for malware variant</indicator:Title>
            <indicator:Type xsi:type="stixVocabs:IndicatorTypeVocab-1.1">File Hash
 Watchlist</indicator:Type>
            <indicator:Observable id="example:Observable-4444-5555-6666-7777">
                <cybox:Object id="example:File-4444-5555-6666-7777">
                    <cybox:Properties xsi:type="FileObj:FileObjectType">
                        <FileObj:Hashes>
                            <cyboxCommon:Hash>
                                <cyboxCommon:Type
 xsi:type="cyboxVocabs:HashNameVocab-1.0">SHA256</cyboxCommon:Type>
                                <cyboxCommon:Simple_Hash_Value
 condition="Equals">a665a45920422f9d417e4867efdc4fb8a04a1f3fff1fa07e998e86f7f7a27ae3</
cyboxCommon:Simple_Hash_Value>
                            </cyboxCommon:Hash>
                        </FileObj:Hashes>
                    </cybox:Properties>
                </cybox:Object>
```

```
            </indicator:Observable>
        </stix:Indicator>
    </stix:Indicators>
</stix:STIX_Package>
```

Example IP アドレスリストの例

```
<?xml version="1.0" encoding="UTF-8"?>
<stix:STIX_Package
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:stix="http://stix.mitre.org/stix-1"
    xmlns:stixCommon="http://stix.mitre.org/common-1"
    xmlns:ttp="http://stix.mitre.org/TTP-1"
    xmlns:cybox="http://cybox.mitre.org/cybox-2"
    xmlns:AddressObject="http://cybox.mitre.org/objects#AddressObject-2"
    xmlns:cyboxVocabs="http://cybox.mitre.org/default_vocabularies-2"
    xmlns:stixVocabs="http://stix.mitre.org/default_vocabularies-1"
    xmlns:example="http://example.com/"
    xsi:schemaLocation="
    http://stix.mitre.org/stix-1 http://stix.mitre.org/XMLSchema/core/1.2/
stix_core.xsd
    http://stix.mitre.org/Campaign-1 http://stix.mitre.org/XMLSchema/campaign/1.2/
campaign.xsd
    http://stix.mitre.org/Indicator-2 http://stix.mitre.org/XMLSchema/indicator/2.2/
indicator.xsd
    http://stix.mitre.org/TTP-2 http://stix.mitre.org/XMLSchema/ttp/1.2/ttp.xsd
    http://stix.mitre.org/default_vocabularies-1 http://stix.mitre.org/XMLSchema/
default_vocabularies/1.2.0/stix_default_vocabularies.xsd
    http://cybox.mitre.org/objects#AddressObject-2 http://cybox.mitre.org/XMLSchema/
objects/Address/2.1/Address_Object.xsd"
    id="example:STIXPackage-a78fc4e3-df94-42dd-a074-6de62babfe16"
    version="1.2">
    <stix:Observables cybox_major_version="1" cybox_minor_version="1">
        <cybox:Observable id="example:observable-80b26f43-
dc41-43ff-861d-19aff31e0236">
            <cybox:Object id="example:object-161a5438-1c26-4275-ba44-a35ba963c245">
                <cybox:Properties xsi:type="AddressObject:AddressObjectType"
 category="ipv4-addr">

 <AddressObject:Address_Valuecondition="InclusiveBetween">192.0.2.0##comma##192.0.2.255</
AddressObject:Address_Value>
                </cybox:Properties>
            </cybox:Object>
```

```
        </cybox:Observable>
        <cybox:Observable id="example:observable-b442b399-aea4-436f-bb34-
b9ef6c5ed8ab">
            <cybox:Object id="example:object-b422417f-bf78-4b34-ba2d-de4b09590a6d">
                <cybox:Properties xsi:type="AddressObject:AddressObjectType"
 category="ipv4-addr">
                    <AddressObject:Address_Value>198.51.100.1</
AddressObject:Address_Value>
                </cybox:Properties>
            </cybox:Object>
        </cybox:Observable>
        <cybox:Observable
 id="example:observable-1742fa06-8b5e-4449-9d89-6f9f32595784">
            <cybox:Object id="example:object-dc73b749-8a31-46be-803f-71df77565391">
                <cybox:Properties xsi:type="AddressObject:AddressObjectType"
 category="ipv4-addr">
                    <AddressObject:Address_Value>203.0.113.1</
AddressObject:Address_Value>
                </cybox:Properties>
            </cybox:Object>
        </cybox:Observable>
    </stix:Observables>
</stix:STIX_Package>
```

Open Threat Exchange (OTX)™ CSV

この形式は、CIDR ブロック、個々の IP アドレス、ドメイン、およびファイルハッシュのSHA256インジケータタイプをサポートします。SHA-256 ファイルハッシュは、脅威エンティティリストでのみサポートされています。このファイル形式は、カンマで区切られた値を持っています。

Exampleエンティティリストの例

```
Indicator type, Indicator, Description
CIDR, 192.0.2.0/24, example
IPv4, 198.51.100.1, example
IPv4, 203.0.113.1, example
Domain name, example.net, example
```

Exampleファイルハッシュを含む脅威エンティティリストの例

```
Indicator type, Indicator, Description
```

```
CIDR, 192.0.2.0/24, example
IPv4, 198.51.100.1, example
IPv4, 203.0.113.1, example
Domain name, example.net, example
SHA256, a665a45920422f9d417e4867efdc4fb8a04a1f3fff1fa07e998e86f7f7a27ae3, example
```

Example IP アドレスリストの例

```
Indicator type, Indicator, Description
CIDR, 192.0.2.0/24, example
IPv4, 198.51.100.1, example
IPv4, 203.0.113.1, example
```

FireEye™ iSIGHT 脅威インテリジェンス CSV

この形式は、 sha256列で CIDR ブロック、個々の IP アドレス、ドメイン、および SHA-256 ファイルハッシュをサポートします。SHA-256 ファイルハッシュは、脅威エンティティリストでのみサポートされています。次のサンプルリストは FireEye™ CSV フォーマットを使用しています。

Exampleエンティティリストの例

```
reportId, title, threatScape, audience, intelligenceType, publishDate, reportLink,
 webLink, emailIdentifier, senderAddress, senderName, sourceDomain, sourceIp, subject,
 recipient, emailLanguage, fileName, fileSize, fuzzyHash, fileIdentifier, md5, sha1,
 sha256, description, fileType, packer, userAgent, registry, fileCompilationDateTime,
 filePath, asn, cidr, domain, domainTimeOfLookup, networkIdentifier, ip, port,
 protocol, registrantEmail, registrantName, networkType, url, malwareFamily,
 malwareFamilyId, actor, actorId, observationTime

01-00000001, Example, Test, Operational, threat, 1494944400,
 https://www.example.com/report/01-00000001, https://www.example.com/
report/01-00000001, , , , , , , , , , , , , , , , , , , , , , , , , , , , , , , , , 192.0.2.0/24, , ,
 Related, , , , , , network, , Ursnif, 21a14673-0d94-46d3-89ab-8281a0466099, , ,
 1494944400

01-00000002, Example, Test, Operational, threat, 1494944400,
 https://www.example.com/report/01-00000002, https://www.example.com/
report/01-00000002, , , , , , , , , , , , , , , , , , , ,
```

```
01-00000003, Example, Test, Operational, threat, 1494944400,
 https://www.example.com/report/01-00000003, https://www.example.com/
report/01-00000003, , , , , , , , , , , , , , , , , , , , , , , , , , , , , Related,
 203.0.113.1, , , , , network, , Ursnif, 8a78c3db-7bcb-40bc-a080-75bd35a2572d, , ,
 1494944400

 01-00000002, Malicious domain observed in test, Test, Operational, threat,
 1494944400, https://www.example.com/report/01-00000002,https://www.example.com/
report/01-00000002,,,,,,,,,,,,,,,,,,,,,,, 203.0.113.0/24, example.com,, Related,
 203.0.113.0, 8080, UDP,,, network,, Ursnif, fc13984c-c767-40c9-8329-f4c59557f73b,,,
 1494944400
```

Exampleファイルハッシュを含む脅威エンティティリストの例

```
reportId, title, threatScape, audience, intelligenceType, publishDate, reportLink,
 webLink, emailIdentifier, senderAddress, senderName, sourceDomain, sourceIp, subject,
 recipient, emailLanguage, fileName, fileSize, fuzzyHash, fileIdentifier, md5, sha1,
 sha256, description, fileType, packer, userAgent, registry, fileCompilationDateTime,
 filePath, asn, cidr, domain, domainTimeOfLookup, networkIdentifier, ip, port,
 protocol, registrantEmail, registrantName, networkType, url, malwareFamily,
 malwareFamilyId, actor, actorId, observationTime

01-00000001, Example, Test, Operational, threat, 1494944400,
 https://www.example.com/report/01-00000001, https://www.example.com/
report/01-00000001, , , , , , , , , , , , , , , , , , , , , , , , , 192.0.2.0/24, , ,
 Related, , , , , , network, , Ursnif, 21a14673-0d94-46d3-89ab-8281a0466099, , ,
 1494944400

01-00000002, Example, Test, Operational, threat, 1494944400,
 https://www.example.com/report/01-00000002, https://www.example.com/
report/01-00000002, , , , , , , , , , , , , , , , , , , , , , , , , , , , , Related,
 198.51.100.1, , , , , network, , Ursnif,
 12ab7bc4-62ed-49fa-99e3-14b92afc41bf, , ,1494944400

01-00000003, Example, Test, Operational, threat, 1494944400,
 https://www.example.com/report/01-00000003, https://www.example.com/
report/01-00000003, , , , , , , , , , , , , , , , , , , , , , , , , , , , , Related,
 203.0.113.1, , , , , network, , Ursnif, 8a78c3db-7bcb-40bc-a080-75bd35a2572d, , ,
 1494944400

 01-00000002, Malicious domain observed in test, Test, Operational, threat,
 1494944400, https://www.example.com/report/01-00000002,https://www.example.com/
report/01-00000002,,,,,,,,,,,,,,,,,,,,,,, 203.0.113.0/24, example.com,, Related,
```

```
 203.0.113.0, 8080, UDP,,, network,, Ursnif, fc13984c-c767-40c9-8329-f4c59557f73b,,,
 1494944400

01-00000005, Malicious file hash, Test, Operational, threat,
 1494944400, https://www.example.com/report/01-00000005, https://
www.example.com/report/01-00000005, , , , , , , , , , , , , , , ,
 a665a45920422f9d417e4867efdc4fb8a04a1f3fff1fa07e998e86f7f7a27ae3, , , , , , , , , , , ,
 Related, , , , , , network, , Ursnif, 9b9a6ea0-3cbf-4e12-bd3e-0c1d7b4e5f6a, , ,
 1494944400
```

Example IP アドレスリストの例

```
reportId, title, threatScape, audience, intelligenceType, publishDate, reportLink,
 webLink, emailIdentifier, senderAddress, senderName, sourceDomain, sourceIp, subject,
 recipient, emailLanguage, fileName, fileSize, fuzzyHash, fileIdentifier, md5, sha1,
 sha256, description, fileType, packer, userAgent, registry, fileCompilationDateTime,
 filePath, asn, cidr, domain, domainTimeOfLookup, networkIdentifier, ip, port,
 protocol, registrantEmail, registrantName, networkType, url, malwareFamily,
 malwareFamilyId, actor, actorId, observationTime

01-00000001, Example, Test, Operational, threat, 1494944400,
 https://www.example.com/report/01-00000001, https://www.example.com/
report/01-00000001, , , , , , , , , , , , , , , , , , , , , , , , , 192.0.2.0/24, , ,
 Related, , , , , , network, , Ursnif, 21a14673-0d94-46d3-89ab-8281a0466099, , ,
 1494944400

01-00000002, Example, Test, Operational, threat, 1494944400,
 https://www.example.com/report/01-00000002, https://www.example.com/
report/01-00000002, , , , , , , , , , , , , , , , , , , , , , , , , , , , , Related,
 198.51.100.1, , , , , network, , Ursnif,
 12ab7bc4-62ed-49fa-99e3-14b92afc41bf, , ,1494944400

01-00000003, Example, Test, Operational, threat, 1494944400,
 https://www.example.com/report/01-00000003, https://www.example.com/
report/01-00000003, , , , , , , , , , , , , , , , , , , , , , , , , , , , , Related,
 203.0.113.1, , , , , network, , Ursnif, 8a78c3db-7bcb-40bc-a080-75bd35a2572d, , ,
 1494944400
```

Proofpoint™ ET Intelligence Feed CSV

ProofPoint CSV フォーマットでは、IP アドレスまたはドメイン名を 1 つのリストに追加できます。次のサンプルリストは Proofpoint CSV フォーマットを使用しています。ports パラメータに値を指定するかどうかはオプションです。指定しない場合は、末尾にカンマ (,) を残します。

Exampleエンティティリストの例

```
domain, category, score, first_seen, last_seen, ports (|)
198.51.100.1, 1, 100, 2000-01-01, 2000-01-01,
203.0.113.1, 1, 100, 2000-01-01, 2000-01-01, 80
```

Example IP アドレスリストの例

```
ip, category, score, first_seen, last_seen, ports (|)
198.51.100.1, 1, 100, 2000-01-01, 2000-01-01,
203.0.113.1, 1, 100, 2000-01-01, 2000-01-01, 80
```

AlienVault™ Reputation Feed

次のサンプルリストは AlienVault フォーマットを使用しています。この形式は、インジケータ列の SHA-256 ファイルハッシュもサポートしています。SHA-256 ファイルハッシュは、脅威エンティティリストでのみサポートされています。

Exampleエンティティリストの例

```
192.0.2.1#4#2#Malicious Host#KR##37.5111999512,126.974098206#3
192.0.2.2#4#2#Scanning Host#IN#Gurgaon#28.4666996002,77.0333023071#3
192.0.2.3#4#2##CN#Guangzhou#23.1166992188,113.25#3
www.test.org#4#2#Malicious Host#CA#Brossard#45.4673995972,-73.4832000732#3
www.example.com#4#2#Malicious Host#PL##52.2393989563,21.0361995697#3
```

Exampleファイルハッシュを含む脅威エンティティリストの例

```
192.0.2.1#4#2#Malicious Host#KR##37.5111999512,126.974098206#3
192.0.2.2#4#2#Scanning Host#IN#Gurgaon#28.4666996002,77.0333023071#3
192.0.2.3#4#2##CN#Guangzhou#23.1166992188,113.25#3
www.test.org#4#2#Malicious Host#CA#Brossard#45.4673995972,-73.4832000732#3
www.example.com#4#2#Malicious Host#PL##52.2393989563,21.0361995697#3
a665a45920422f9d417e4867efdc4fb8a04a1f3fff1fa07e998e86f7f7a27ae3#4#2#Malicious
 Hash###0.0,0.0#3
```

Example IP アドレスリストの例

```
198.51.100.1#4#2#Malicious Host#US##0.0,0.0#3
```

```
203.0.113.1#4#2#Malicious Host#US##0.0,0.0#3
```

リストのステータスについて理解する

エンティティリストまたは IP アドレスリストを追加すると、GuardDuty はそのリストのステータスを表示します。[ステータス] 列は、リストが有効かどうか、およびアクションが必要かどうかを示します。以下のリストは、有効なステータス値を示しています。

- アクティブ – そのリストが現在、カスタム脅威検出に使用されていることを示します。
- 非アクティブ – そのリストが現在、使用されていないことを示します。GuardDuty がこのリストを環境内の脅威検出に使用する方法については、「エンティティリストまたは IP リストの追加とアクティブ化」の「ステップ 3: エンティティリストまたは IP アドレスリストをアクティブ化する」を参照してください。
- エラー – そのリストに問題があることを示します。ステータスにカーソルを合わせると、エラーの詳細を確認できます。
- アクティブ化 — GuardDuty がリストのアクティブ化プロセスを開始したことを示します。このリストのステータスを引き続きモニタリングできます。エラーがない場合、ステータスは [アクティブ] に更新されます。ステータスが [アクティブ化] の間は、このリストに対してアクションを実行することはできません。リストのステータスが [アクティブ] に変わるまでに数分かかる場合があります。
- 非アクティブ化 — GuardDuty がリストを非アクティブ化するプロセスを開始したことを示します。このリストのステータスを引き続きモニタリングできます。エラーがない場合、ステータスは [非アクティブ] に更新されます。ステータスが [非アクティブ化] の間は、このリストに対してアクションを実行することはできません。
- 削除保留中 – そのリストが削除中であることを示します。ステータスが [削除保留中] の間は、このリストに対してアクションを実行することはできません。

エンティティリストと IP アドレスリストの前提条件の設定

GuardDuty はエンティティリストと IP アドレスリストを使用して、 AWS 環境内の脅威検出をカスタマイズします。エンティティリスト (推奨) は IP アドレスとドメイン名の両方をサポートしますが、IP アドレスリストは IP アドレスのみをサポートします。これらのリストの作成を開始する前に、使用したいリストのタイプに必要なアクセス許可を追加する必要があります。

エンティティリストの前提条件

エンティティリストを追加すると、GuardDuty は S3 バケットから信頼できるリストと脅威インテリジェンスリストを読み取ります。エンティティリストの作成に使用するロールは、これらのリストを含む S3 バケットへの `s3:GetObject` アクセス許可を必ず持つ必要があります。

> **Note**
>
> マルチアカウント環境では、GuardDuty 管理者アカウントのみがリストを管理でき、これらはメンバーアカウントに自動的に適用されます。

S3 バケットの場所に対する `s3:GetObject` アクセス許可がまだない場合は、以下のポリシー例を使用して *amzn-s3-demo-bucket* を S3 バケットの場所に置き換えます。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/[object-key]"
        }
    ]
}
```

IP アドレスリストの前提条件

さまざまな IAM アイデンティティは、GuardDuty で信頼できる IP リストおよび脅威リストを用いて機能するための特別な許可を必要とします。AmazonGuardDutyFullAccess_v2 (推奨) マネージドポリシーがアタッチされている ID のみがアップロードされた信頼できる IP リストと脅威リストの名前を変更および無効化できます。

さまざまな ID に、信頼できる IP リストと脅威リストの操作 (名前の変更および非アクティブ化に加えて、リストの追加、アクティブ化、削除や、リストの場所または名前の更新が含まれます) を行うためのフルアクセスを付与するには、次のアクションがユーザー、グループ、またはロールにアタッチされた許可ポリシーに存在することを確認してください。

```
{
    "Effect": "Allow",
    "Action": [
        "iam:PutRolePolicy",
        "iam:DeleteRolePolicy"
    ],
    "Resource": "arn:aws:iam::555555555555:role/aws-service-role/
guardduty.amazonaws.com/AWSServiceRoleForAmazonGuardDuty"
}
```

> **Important**
>
> これらのアクションは AmazonGuardDutyFullAccess マネージドポリシーに含まれていません。

エンティティリストと IP リストでの SSE-KMS 暗号化の使用

GuardDuty は、リストの SSE-AES256 および SSE-KMS 暗号化をサポートしています。SSE-C はサポートされていません。S3 の暗号化タイプの詳細については、「サーバー側の暗号化によるデータの保護」を参照してください。

エンティティリストと IP リストのどちらを使用するかにかかわらず、SSE-KMS を使用する場合は、次のステートメントを AWS KMS key ポリシーに追加します。*123456789012* は自身のアカウント ID に置き換えます。

```
{
    "Sid": "AllowGuardDutyServiceRole",
    "Effect": "Allow",
    "Principal": {
    "AWS": "arn:aws:iam::123456789012:role/aws-service-role/guardduty.amazonaws.com/
AWSServiceRoleForAmazonGuardDuty"
    },
    "Action": "kms:Decrypt*",
    "Resource": "*"
}
```

エンティティリストまたは IP リストの追加とアクティブ化

エンティティリストと IP アドレスリストは、GuardDuty の脅威検出機能をカスタマイズするのに役立ちます。これらのリストの詳細については、「エンティティリストと IP アドレスリストについて理解する」を参照してください。 AWS 環境の信頼された脅威インテリジェンスデータを管理するために、GuardDuty ではエンティティリストを使用することをお勧めします。開始する前に、「エンティティリストと IP アドレスリストの前提条件の設定」を参照してください。

次のアクセス方法のいずれかを選択して、信頼できるエンティティリスト、脅威エンティティリスト、信頼できる IP リストまたは脅威 IP リストを追加してアクティブ化します。

Console

(オプション) ステップ 1: リストの場所の URL を取得する

1. Amazon S3 コンソール (https://console.aws.amazon.com/s3/) を開きます。
2. ナビゲーションペインで、[バケット] を選択します。
3. 追加する特定のリストを含む Amazon S3 バケット名を選択します。
4. オブジェクト (リスト) 名を選択すると、その詳細が表示されます。
5. [プロパティ] タブで、このオブジェクトの S3 URI をコピーします。

ステップ 2: 信頼できるデータまたは脅威インテリジェンスデータを追加する

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[リスト] を選択します。
3. [リスト] ページで、[エンティティリスト] または [IP アドレスリスト] タブを選択します。
4. 選択したタブに基づいて、信頼できるリストまたは脅威リストの追加を選択します。
5. 信頼できるリストまたは脅威リストを追加するダイアログボックスで、以下の手順を実行します。

   a. [リスト名] で、リストの名前を入力します。

      リストの命名に関する制約 - リストの名前には、小文字、大文字、数字、ダッシュ (-)、アンダースコア (_) を含めることができます。

      IP アドレスリストの場合、リストの名前は AWS アカウント および リージョン内で一意である必要があります。

b. [場所] で、リストをアップロードした場所を指定します。まだ持っていない場合は、「Step 1: Fetching location URL of your list」を参照してください。

   カスタム脅威とカスタムの信頼できるエンティティセットにのみ適用されます – 次のサポートされている形式と一致しない場所の URL を指定すると、リストの追加とアクティベーション中にエラーメッセージが表示されます。

   場所の URL の形式。

   - https://s3.amazonaws.com/bucket.name/file.txt
   - https://s3-aws-region.amazonaws.com/bucket.name/file.txt
   - http://bucket.s3.amazonaws.com/file.txt
   - http://bucket.s3-aws-region.amazonaws.com/file.txt
   - s3://bucket.name/file.txt

c. (オプション) 予想されるバケット所有者には、 Location フィールドで指定された Amazon S3 バケットを所有する AWS アカウント ID を入力できます。

   AWS アカウント ID 所有者を指定しない場合、GuardDuty はエンティティリストと IP アドレスリストの動作が異なります。エンティティリストの場合、GuardDuty は現在のメンバーアカウントが [場所] フィールドで指定された S3 バケットを所有していることを検証します。IP アドレスリストでは、 AWS アカウント ID 所有者を指定しない場合、GuardDuty は検証を実行しません。

   GuardDuty がこの S3 バケットが指定されたアカウント ID に属していないと判断した場合、リストのアクティブ化時にエラーが発生します。

d. [I agree] (同意します) チェックボックスをオンにします。

e. [Add list] (リストを追加) を選択します。デフォルトでは、追加されたリストの [ステータス] は [非アクティブ] です。リストを有効にするには、リストをアクティブ化する必要があります。

ステップ 3: エンティティリストまたは IP アドレスリストをアクティブ化する

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[リスト] を選択します。
3. [リスト[ページで、リストをアクティブ化するタブ - [エンティティリスト] または [IP アドレスリスト] を選択します。

4. アクティブ化するリストを 1 つ選択します。これにより、[アクション] メニューと[編集] メニューが有効になります。
5. [アクション] を選択し、[アクティブ化] を選択します。

API/CLI

信頼できるエンティティリストを追加およびアクティブ化するには

1. CreateTrustedEntitySet を実行します。この信頼できるエンティティリストを作成するメンバーアカウントの `detectorId` を必ず指定してください。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

   リストの命名に関する制約 - リストの名前には、小文字、大文字、数字、ダッシュ (-)、アンダースコア (_) を含めることができます。
2. あるいは、次の AWS Command Line Interface コマンドを実行することによってこれを実行できます。

```
aws guardduty create-trusted-entity-set \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--name "AnyOrganizationListEXAMPLE" \
--format TXT \
--location "https://s3.amazonaws.com/amzn-s3-demo-bucket/DOC-EXAMPLE-SOURCE-FILE.format" \
--activate
```

   `detector-id` を、信頼できるエンティティリストを作成するメンバーアカウントのディテクター ID、および`######`その他のプレースホルダー値に置き換えます。

   この新しく作成されたリストをアクティブ化しない場合は、 パラメータ `--activate` を `--no-activate` に置き換えます。

   `expected-bucket-owner` パラメータはオプションです。このパラメータの値を指定するかどうかにかかわらず、GuardDuty は、この`--detector-id`値に関連付けられた AWS アカウント ID が `--location`パラメータで指定された S3 バケットを所有していることを検証します。GuardDuty がこの S3 バケットが指定されたアカウント ID に属していないと判断した場合、このリストをアクティブ化するときにエラーが発生します。

カスタム脅威とカスタムの信頼できるエンティティセットにのみ適用されます – 次のサポートされている形式と一致しない場所の URL を指定すると、リストの追加とアクティベーション中にエラーメッセージが表示されます。

脅威エンティティリストを追加およびアクティブ化するには

1. CreateThreatEntitySet を実行します。この脅威エンティティリストを作成するメンバーアカウントの `detectorId` を必ず指定してください。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

   リストの命名に関する制約 - リストの名前には、小文字、大文字、数字、ダッシュ (-)、アンダースコア (_) を含めることができます。

2. あるいは、次の AWS Command Line Interface コマンドを実行することによってこれを実行できます。

```
aws guardduty create-threat-entity-set \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--name "AnyOrganizationListEXAMPLE" \
--format TXT \
--location "https://s3.amazonaws.com/amzn-s3-demo-bucket/DOC-EXAMPLE-SOURCE-FILE.format" \
--activate
```

   `detector-id` を、信頼できるエンティティリストを作成するメンバーアカウントのディテクター ID、および`######`その他のプレースホルダー値に置き換えます。

   この新しく作成されたリストをアクティブ化しない場合は、 パラメータ `--activate` を `--no-activate` に置き換えます。

   `expected-bucket-owner` パラメータはオプションです。このパラメータの値を指定するかどうかにかかわらず、GuardDuty は、この`--detector-id`値に関連付けられた AWS アカウント ID が `--location`パラメータで指定された S3 バケットを所有していることを検証します。GuardDuty がこの S3 バケットが指定されたアカウント ID に属していないと判断した場合、このリストをアクティブ化するときにエラーが発生します。

カスタム脅威とカスタムの信頼できるエンティティセットにのみ適用されます – 次のサポートされている形式と一致しない場所の URL を指定すると、リストの追加とアクティベーション中にエラーメッセージが表示されます。

信頼できる IP アドレスリストを追加してアクティブ化するには

1. CreateIPSet を実行します。この信頼されている IP アドレスリストを作成するメンバーアカウントの `detectorId` を必ず指定してください。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

   IP アドレスリストの場合、リストの名前は AWS アカウント および リージョン内で一意である必要があります。

   リストの命名に関する制約 - リストの名前には、小文字、大文字、数字、ダッシュ (-)、アンダースコア (_) を含めることができます。

2. または、次の AWS Command Line Interface コマンドを実行してこれを行うこともできます。このとき、`detector-id` を、信頼されている IP アドレスリストを更新するメンバーアカウントのディテクター ID に置き換えてください。

```
aws guardduty create-ip-set \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--name "AnyOrganizationListEXAMPLE" \
--format TXT \
--location "https://s3.amazonaws.com/amzn-s3-demo-bucket/DOC-EXAMPLE-SOURCE-FILE.format" \
--activate
```

   `detector-id` を、信頼できる IP リストを作成するメンバーアカウントのディテクター ID、および`######`その他のプレースホルダー値に置き換えます。

   この新しく作成されたリストをアクティブ化しない場合は、 パラメータ `--activate` を `--no-activate` に置き換えます。

   `expected-bucket-owner` パラメータはオプションです。S3 バケットを所有するアカウント ID を指定しない場合、GuardDuty は検証を実行しません。`expected-bucket-owner` パラメータのアカウント ID を指定すると、GuardDuty はこの AWS アカウント ID が `--location`パラメータで指定された S3 バケットを所有していることを検証しま

す。GuardDuty がこの S3 バケットが指定されたアカウント ID に属していないと判断した場合、このリストをアクティブ化するときにエラーが発生します。

脅威 IP リストを追加およびアクティブ化するには

1. CreateThreatIntelSet を実行します。この脅威 IP アドレスリストを作成するメンバーアカウントの detectorId を必ず指定してください。アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

   リストの命名に関する制約 - リストの名前には、小文字、大文字、数字、ダッシュ (-)、アンダースコア (_) を含めることができます。

   IP アドレスリストの場合、リストの名前は AWS アカウント および リージョン内で一意である必要があります。

2. または、次の AWS Command Line Interface コマンドを実行してこれを行うこともできます。このとき、detector-id を、脅威 IP リストを更新するメンバーアカウントのディテクター ID に置き換えてください。

```
aws guardduty create-threat-intel-set \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--name "AnyOrganizationListEXAMPLE" \
--format TXT \
--location "https://s3.amazonaws.com/amzn-s3-demo-bucket/DOC-EXAMPLE-SOURCE-FILE.format" \
--activate
```

   detector-id を脅威 IP リストを作成するメンバーアカウントのディテクター ID と、########その他のプレースホルダー値に置き換えます。

   この新しく作成されたリストをアクティブ化しない場合は、 パラメータ --activate を --no-activate に置き換えます。

   expected-bucket-owner パラメータはオプションです。S3 バケットを所有するアカウント ID を指定しない場合、GuardDuty は検証を実行しません。expected-bucket-owner パラメータのアカウント ID を指定すると、GuardDuty はこの AWS アカウント ID が --locationパラメータで指定された S3 バケットを所有していることを検証します。GuardDuty がこの S3 バケットが指定されたアカウント ID に属していないと判断した場合、このリストをアクティブ化するときにエラーが発生します。

エンティティリストまたは IP アドレスリストを有効化すると、そのリストが有効になるまでに数分かかる場合があります。詳細については、「GuardDuty リストに関する重要な考慮事項」を参照してください。

エンティティリストまたは IP アドレスリストの更新

エンティティリストと IP アドレスリストは、GuardDuty の脅威検出機能をカスタマイズするのに役立ちます。これらのリストの詳細については、「エンティティリストと IP アドレスリストについて理解する」を参照してください。

リスト名、S3 バケットの場所、想定されるバケット所有者アカウント ID、および既存リスト内のエントリを更新できます。リストのエントリを更新する場合、GuardDuty がリストの最新バージョンを使用するには、そのリストを再度アクティブ化するステップに従う必要があります。エンティティリストまたは IP アドレスリストを更新または有効化した後、このリストが有効になるまでに数分かかる場合があります。詳細については、「GuardDuty リストに関する重要な考慮事項」を参照してください。

> **Note**
>
> リストのステータスが[アクティブ化]、[非アクティブ化]、または[削除保留中] の場合は、アクションを実行する前に数分待つ必要があります。これらのステータスについては、「リストのステータスについて理解する」を参照してください。

アクセス方法のいずれかを選択して、エンティティリストまたは IP アドレスリストを更新します。

Console

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[リスト] を選択します。
3. [リスト] ページで、適切なタブ - [エンティティリスト] または [IP アドレスリスト] を選択します。
4. 更新するリスト (信頼できるリストまたは脅威リスト) を 1 つ選択します。これにより、[アクション] メニューと[編集] メニューが有効になります。
5. [編集] を選択します。
6. リストを更新するダイアログボックスで、更新する詳細を指定します。

リストの命名に関する制約 - リストの名前には、小文字、大文字、数字、ダッシュ (-)、アンダースコア (_) を含めることができます。

IP アドレスリストの場合、リストの名前は AWS アカウント および リージョン内で一意である必要があります。

カスタム脅威とカスタムの信頼できるエンティティセットにのみ適用されます – 次のサポートされている形式と一致しない場所の URL を指定すると、リストの追加とアクティベーション中にエラーメッセージが表示されます。

7. (オプション) 予想されるバケット所有者には、 Location フィールドで指定された Amazon S3 バケットを所有する AWS アカウント ID を入力できます。

   AWS アカウント ID 所有者を指定しない場合、GuardDuty はエンティティリストと IP アドレスリストの動作が異なります。エンティティリストの場合、GuardDuty は現在のメンバーアカウントが [場所] フィールドで指定された S3 バケットを所有していることを検証します。IP アドレスリストでは、 AWS アカウント ID 所有者を指定しない場合、GuardDuty は検証を実行しません。

   GuardDuty がこの S3 バケットが指定されたアカウント ID に属していないと判断した場合、リストのアクティブ化時にエラーが発生します。

8. [同意します] チェックボックスをオンにし、[リストを更新] を選択します。

API/CLI

次の手順を開始するには、更新するリストリソースに関連付けられている `trustedEntitySetId`、`threatEntitySetId`、`trustedIpSet`、`threatIpSet` などの ID が必要です。

信頼できるエンティティリストを更新してアクティブ化するには

1. UpdateTrustedEntitySet を実行します。この信頼されている IP リストを更新するメンバーアカウントの `detectorId` を必ず指定してください。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

   リストの命名に関する制約 - リストの名前には、小文字、大文字、数字、ダッシュ (-)、アンダースコア (_) を含めることができます。

2. または、次の AWS Command Line Interface コマンドを実行してリストの name を更新し、このリストをアクティブ化することもできます。

```
aws guardduty update-trusted-entity-set \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--name "AnyOrganization ListEXAMPLE" \
--trusted-entity-set-id d4b94fc952d6912b8f3060768example \
--activate
```

detector-id を、信頼できるエンティティリストを作成するメンバーアカウントのディテクター ID、および######その他のプレースホルダー値に置き換えます。

この新しく作成されたリストをアクティブ化しない場合は、 パラメータ --activate を --no-activate に置き換えます。

expected-bucket-owner パラメータはオプションです。このパラメータの値を指定するかどうかにかかわらず、GuardDuty は、この--detector-id値に関連付けられた AWS アカウント ID が --locationパラメータで指定された S3 バケットを所有していることを検証します。GuardDuty がこの S3 バケットが指定されたアカウント ID に属していないと判断した場合、このリストをアクティブ化するときにエラーが発生します。

カスタム脅威とカスタムの信頼できるエンティティセットにのみ適用されます – 次のサポートされている形式と一致しない場所の URL を指定すると、リストの追加とアクティベーション中にエラーメッセージが表示されます。

脅威エンティティリストを更新してアクティブ化するには

1. UpdateThreatEntitySet を実行します。この脅威エンティティリストを作成するメンバーアカウントの detectorId を必ず指定してください。アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

   リストの命名に関する制約 - リストの名前には、小文字、大文字、数字、ダッシュ (-)、アンダースコア (_) を含めることができます。

2. または、次の AWS Command Line Interface コマンドを実行してリストの name を更新し、このリストをアクティブ化することもできます。

```
aws guardduty update-threat-entity-set \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
```

```
--name "AnyOrganization ListEXAMPLE" \
--threat-entity-set-id d4b94fc952d6912b8f3060768example \
--activate
```

detector-id を、脅威エンティティリストを作成するメンバーアカウントのディテクター ID、および######その他のプレースホルダー値に置き換えます。

この新しく作成されたリストをアクティブ化しない場合は、 パラメータ --activate を --no-activate に置き換えます。

expected-bucket-owner パラメータはオプションです。このパラメータの値を指定するかどうかにかかわらず、GuardDuty は、この--detector-id値に関連付けられた AWS アカウント ID が --locationパラメータで指定された S3 バケットを所有していることを検証します。GuardDuty がこの S3 バケットが指定されたアカウント ID に属していないと判断した場合、このリストをアクティブ化するときにエラーが発生します。

カスタム脅威とカスタムの信頼できるエンティティセットにのみ適用されます – 次のサポートされている形式と一致しない場所の URL を指定すると、リストの追加とアクティベーション中にエラーメッセージが表示されます。

信頼できる IP アドレスリストを更新してアクティブ化するには

1. CreateIPSet を実行します。この信頼されている IP アドレスリストを更新するメンバーアカウントの detectorId を必ず指定してください。アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

   リストの命名に関する制約 - リストの名前には、小文字、大文字、数字、ダッシュ (-)、アンダースコア (_) を含めることができます。

   IP アドレスリストの場合、リストの名前は AWS アカウント および リージョン内で一意である必要があります。

2. あるいは、リストをアクティブ化する次の AWS Command Line Interface コマンドを実行することによってこれも実行できます。

```
aws guardduty update-ip-set \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--name "AnyOrganization ListEXAMPLE" \
--ip-set-id d4b94fc952d6912b8f3060768example \
```

```
--activate
```

detector-id を、信頼できる IP リストを更新するメンバーアカウントのディテクター ID、および######その他のプレースホルダー値に置き換えます。

この新しく作成されたリストをアクティブ化しない場合は、 パラメータ --activate を --no-activate に置き換えます。

expected-bucket-owner パラメータはオプションです。S3 バケットを所有するアカウント ID を指定しない場合、GuardDuty は検証を実行しません。expected-bucket-owner パラメータのアカウント ID を指定すると、GuardDuty はこの AWS アカウント ID が --locationパラメータで指定された S3 バケットを所有していることを検証します。GuardDuty がこの S3 バケットが指定されたアカウント ID に属していないと判断した場合、このリストをアクティブ化するときにエラーが発生します。

脅威 IP リストを追加およびアクティブ化するには

1. CreateThreatIntelSet を実行します。この脅威 IP アドレスリストを作成するメンバーアカウントの detectorId を必ず指定してください。アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

   リストの命名に関する制約 - リストの名前には、小文字、大文字、数字、ダッシュ (-)、アンダースコア (_) を含めることができます。

   IP アドレスリストの場合、リストの名前は AWS アカウント および リージョン内で一意である必要があります。

2. あるいは、リストをアクティブ化する次の AWS Command Line Interface コマンドを実行することによってこれも実行できます。

```
aws guardduty update-threat-intel-set \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--name "AnyOrganization ListEXAMPLE" \
--threat-intel-set-id d4b94fc952d6912b8f3060768example \
--activate
```

detector-id を脅威 IP リストを更新するメンバーアカウントのディテクター ID と、########その他のプレースホルダー値に置き換えます。

この新しく作成されたリストをアクティブ化しない場合は、 パラメータ `--activate` を `--no-activate` に置き換えます。

`expected-bucket-owner` パラメータはオプションです。S3 バケットを所有するアカウント ID を指定しない場合、GuardDuty は検証を実行しません。`expected-bucket-owner` パラメータのアカウント ID を指定すると、GuardDuty はこの AWS アカウント ID が `--location`パラメータで指定された S3 バケットを所有していることを検証します。GuardDuty がこの S3 バケットが指定されたアカウント ID に属していないと判断した場合、このリストをアクティブ化するときにエラーが発生します。

エンティティリストまたは IP アドレスリストの非アクティブ化

GuardDuty でリストを使用しない場合は、非アクティブ化できます。プロセスが完了するまでに数分かかることがあります。詳細については、「GuardDuty リストに関する重要な考慮事項」を参照してください。リストが非アクティブ化されると、エンティティリストまたは IP アドレスリストのエントリは GuardDuty での脅威検出には影響しません。

いずれかのアクセス方法を選択して、リストを非アクティブ化します。

Console

エンティティリストまたは IP アドレスリストを無効にするには

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[リスト] を選択します。
3. リストページで、リストを非アクティブ化するタブ - [エンティティリスト] または [IP アドレスリスト] を選択します。
4. 選択したタブで、非アクティブ化するリストを選択します。
5. [アクション] を選択し、[非アクティブ化] を選択します。
6. アクションを確認して、[非アクティブ化] を選択します。

API/CLI

次の手順を開始するには、非アクティブ化するリストリソースに関連付けられている `trustedEntitySetId`、`threatEntitySetId`、`trustedIpSet`、`threatIpSet` などの ID が必要です。

信頼できるエンティティリストを非アクティブ化するには

1. UpdateTrustedEntitySet を実行します。この信頼できる IP リストを作成するメンバーアカウントの `detectorId` を必ず指定してください。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。
2. あるいは、次の AWS Command Line Interface コマンドを実行することによってこれを実行できます。

```
aws guardduty update-trusted-entity-set \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--trusted-entity-set-id d4b94fc952d6912b8f3060768example \
--no-activate
```

`detector-id` を、信頼できるエンティティリストを非アクティブ化するメンバーアカウントのディテクター ID、および`######`その他のプレースホルダー値に置き換えます。

脅威エンティティリストを非アクティブ化するには

1. UpdateThreatEntitySet を実行します。この脅威リストを無効にするメンバーアカウントの `detectorId` を必ず指定してください。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。
2. あるいは、次の AWS Command Line Interface コマンドを実行することによってこれを実行できます。

```
aws guardduty update-threat-entity-set \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--threat-entity-set-id d4b94fc952d6912b8f3060768example \
--no-activate
```

`detector-id` を、脅威エンティティリストを作成するメンバーアカウントのディテクター ID、および`######`その他のプレースホルダー値に置き換えます。

信頼できる IP アドレスリストを非アクティブ化するには

1. UpdateIPSet を実行します。この信頼できる IP アドレスリストを作成するメンバーアカウントの detectorId を必ず指定してください。アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。
2. または、次の AWS Command Line Interface コマンドを実行してこれを行うこともできます。このとき、detector-id を、信頼できる IP アドレスリストを非アクティブ化するメンバーアカウントのディテクター ID に置き換えてください。

```
aws guardduty update-ip-set \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--ip-set-id d4b94fc952d6912b8f3060768example \
--no-activate
```

脅威 IP リストを非アクティブ化するには

1. UpdateThreatIntelSet を実行します。この脅威 IP アドレスリストを非アクティブ化するメンバーアカウントの detectorId を必ず指定してください。アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。
2. または、次の AWS Command Line Interface コマンドを実行してこれを行うこともできます。このとき、detector-id を、信頼できる 脅威 IP リストを非アクティブ化するメンバーアカウントのディテクター ID に置き換えてください。

```
aws guardduty update-threat-intel-set \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--threat-intel-set-id d4b94fc952d6912b8f3060768example \
--no-activate
```

エンティティリストまたは IP アドレスリストの削除

エンティティセットまたは IP アドレスセット内のリストエントリを保持しなくなった場合、それを削除できます。プロセスが完了するまでに数分かかることがあります。詳細については、「GuardDuty リストに関する重要な考慮事項」を参照してください。

リストのステータスがアクティブ化または非アクティブ化の場合は、アクションを実行する前に数分待つ必要があります。詳細については、「リストのステータスについて理解する」を参照してください。

いずれかのアクセス方法を選択して、リストを削除します。

Console

エンティティリストまたは IP アドレスリストを削除するには

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで、[リスト] を選択します。
3. [リスト] ページで、リスト - エンティティリストまたは IP アドレスリストを削除するタブを選択します。
4. 選択したタブで、削除するリストを選択します。
5. [アクション]、[削除] の順に選択します。

   リストのステータスは [削除保留中] に変わります。リストが削除されるまでに数分かかる場合があります。

API/CLI

次の手順を開始するには、削除するリストリソースに関連付けられている `trustedEntitySetId`、`threatEntitySetId`、`trustedIpSet`、`threatIpSet` などの ID が必要です。

信頼できるエンティティリストを削除するには

1. DeleteTrustedEntitySet を実行します。この信頼できる IP リストを削除するメンバーアカウントの `detectorId` を必ず指定してください。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。
2. あるいは、次の AWS Command Line Interface コマンドを実行することによってこれを実行できます。

```
aws guardduty delete-trusted-entity-set \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--trusted-entity-set-id d4b94fc952d6912b8f3060768example
```

detector-id を、信頼できるエンティティリストを削除するメンバーアカウントのディテクター ID、および######その他のプレースホルダー値に置き換えます。

脅威エンティティリストを非アクティブ化するには

1. DeleteThreatEntitySet を実行します。この脅威エンティティリストを削除するメンバーアカウントの detectorId を必ず指定してください。アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。
2. あるいは、次の AWS Command Line Interface コマンドを実行することによってこれを実行できます。

```
aws guardduty delete-threat-entity-set \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--threat-entity-set-id d4b94fc952d6912b8f3060768example
```

detector-id を、脅威エンティティリストを削除するメンバーアカウントのディテクター ID と、######その他のプレースホルダー値に置き換えます。

信頼できる IP アドレスリストを削除するには

1. DeleteIPSet を実行します。この信頼できる IP アドレスリストを削除するメンバーアカウントの detectorId を必ず指定してください。アカウントと現在のリージョンの detectorId を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。
2. または、次の AWS Command Line Interface コマンドを実行してこれを行うこともできます。このとき、detector-id を、信頼されている IP アドレスリストを更新するメンバーアカウントのディテクター ID に置き換えてください。

```
aws guardduty delete-ip-set \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--ip-set-id d4b94fc952d6912b8f3060768example
```

detector-id を、脅威エンティティリストを削除するメンバーアカウントのディテクター ID と、######その他のプレースホルダー値に置き換えます。

脅威 IP リストを削除するには

1. DeleteThreatIntelSet を実行します。この脅威 IP アドレスリストを削除するメンバーアカウントの `detectorId` を必ず指定してください。アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。
2. または、次の AWS Command Line Interface コマンドを実行してこれを行うこともできます。このとき、`detector-id` を、脅威 IP リストを削除するメンバーアカウントのディテクター ID に置き換えてください。

```
aws guardduty delete-threat-intel-set \
--detector-id 12abc34d567e8fa901bc2d34e56789f0 \
--threat-intel-set-id d4b94fc952d6912b8f3060768example
```

`detector-id` を、脅威エンティティリストを削除するメンバーアカウントのディテクター ID と、`######`その他のプレースホルダー値に置き換えます。

生成された GuardDuty の検出結果を Amazon S3 バケットにエクスポートする

GuardDuty は生成された検出結果を 90 日間保持します。GuardDuty はアクティブな検出結果を Amazon EventBridge (EventBridge) にエクスポートします。生成された検出結果は、必要に応じて Amazon Simple Storage Service (Amazon S3) バケットにエクスポートできます。これにより、アカウント内の疑わしいアクティビティの履歴データを追跡し、推奨される修復手順が成功したかどうかを評価することができます。

GuardDuty によって生成された新しいアクティブな検出結果はすべて、検出結果の生成後約 5 分以内に自動的にエクスポートされます。アクティブな検出結果の更新を EventBridge にエクスポートする頻度は設定できます。選択した頻度は、更新された既存の検出結果を EventBridge、S3 バケット (設定されている場合)、Detective (統合されている場合) にエクスポートするときに適用されます。GuardDuty が既存の検出結果の複数の出現を集計する方法については、「GuardDuty の検出結果の集約」を参照してください。

検出結果を Amazon S3 バケットにエクスポートするように設定する場合、GuardDuty は AWS Key Management Service (AWS KMS) を使用して S3 バケットの検出結果データを暗号化します。そのためには、GuardDuty がアクセス許可を使用してアカウント内の結果をエクスポートできるように、S3 バケットと AWS KMS キーにアクセス許可を追加する必要があります。

内容

- 考慮事項
- ステップ 1 – 検出結果のエクスポートに必要な許可
- ステップ 2 – KMS キーにポリシーをアタッチする
- ステップ 3: Amazon S3 バケットにポリシーをアタッチする
- ステップ 4 - S3 バケットに検出結果をエクスポートする (コンソール)
- ステップ 5 – 更新されたアクティブな検出結果をエクスポートする頻度を設定する

考慮事項

検出結果をエクスポートするための前提条件とステップに進む前に、次の主要な概念を検討します。

- エクスポート設定はリージョンレベルで行います – GuardDuty を使用するリージョンごとにエクスポートオプションを設定する必要があります。
- さまざまな AWS リージョン (クロスリージョン) の Amazon S3 バケットへの検出結果のエクスポート – GuardDuty は次のエクスポート設定をサポートしています。
  - Amazon S3 バケットまたはオブジェクト、および AWS KMS キーは、同じ に属している必要があります AWS リージョン。
  - 商用リージョンで生成された検出結果の場合、任意の商用リージョンの S3 バケットにこれらの検出結果をエクスポートすることを選択できます。ただし、これらの検出結果をオプトインリージョンの S3 バケットにエクスポートすることはできません。
  - オプトインリージョンで生成された検出結果の場合、生成されたのと同じオプトインリージョン、または任意の商用リージョンにこれらの検出結果をエクスポートすることを選択できます。ただし、あるオプトインリージョンから別のオプトインリージョンに検出結果をエクスポートすることはできません。
- 検出結果をエクスポートする許可 – アクティブな検出結果をエクスポートする設定を行うには、GuardDuty がオブジェクトをアップロードできる許可が S3 バケットに必要です。また、GuardDuty が検出結果を暗号化するために使用できる AWS KMS キーも必要です。
- アーカイブされた検出結果はエクスポートされません – デフォルトの動作では、アーカイブされた検出結果 (抑制された検出結果の新しいインスタンスを含む) はエクスポートされません。

  GuardDuty の検出結果がアーカイブ済みとして生成される場合は、元に戻す必要があります。これにより、[検出結果ステータスをフィルタリング] が [アクティブ] に変更されます。GuardDuty は、「ステップ 5 – 検出結果をエクスポートする頻度」の設定方法に基づいて、アーカイブされていない既存の検出結果に更新をエクスポートします。

- GuardDuty 管理者アカウントは、関連付けられたメンバーアカウントで生成された検出結果をエクスポートできます – 管理者アカウントで検出結果のエクスポートを設定すると、同じリージョンで生成された、関連付けられたメンバーアカウントからのすべての検出結果も、管理者アカウント用に設定したのと同じ場所にエクスポートされます。詳細については、「GuardDuty 管理者アカウントとメンバーアカウントの関係について理解する」を参照してください。

ステップ 1 – 検出結果のエクスポートに必要な許可

検出結果をエクスポートするための設定を行うときは、検出結果を保存できる Amazon S3 バケットと、データの暗号化に使用する AWS KMS キーを選択します。GuardDuty アクションに対する許可に加えて、検出結果をエクスポートする設定を正常に構成するには、次のアクションに対する許可も必要です。

- `s3:GetBucketLocation`
- `s3:PutObject`

Amazon S3 バケット内の特定のプレフィックスに検出結果をエクスポートする必要がある場合は、IAM ロールに次のアクセス許可も追加する必要があります。

- `s3:GetObject`
- `s3:ListBucket`

ステップ 2 – KMS キーにポリシーをアタッチする

GuardDuty は、 を使用してバケット内の検出結果データを暗号化します AWS Key Management Service。設定を正常に構成するには、まず、KMS キーを使用する許可を GuardDuty に付与する必要があります。KMS キーにポリシーを添付することで、許可を付与することができます。

別のアカウントの KMS キーを使用する場合は、キーを所有 AWS アカウント する にログインしてキーポリシーを適用する必要があります。検出結果をエクスポートする設定を行う場合、キーを所有するアカウントのキー ARN も必要になります。

GuardDuty が検出結果を暗号化するために KMS キーポリシーを変更するには

1. https://console.aws.amazon.com/kms で AWS KMS コンソールを開きます。
2. を変更するには AWS リージョン、ページの右上隅にあるリージョンセレクターを使用します。

3. エクスポートされた検出結果を暗号化するために使用するキーについて、既存の KMS キーを選択するか、「AWS Key Management Service 開発者ガイド」の「Create a new key」の手順を実行します。

> **Note**
>
> KMS キーと Amazon S3 バケット AWS リージョン の は同じである必要があります。

同じ S3 バケットと KMS キーのペアを使用して、該当する任意のリージョンから検出結果をエクスポートできます。詳細については、リージョン間の検出結果のエクスポートに関する「考慮事項」を参照してください。

4. [Key policy] (キーポリシー) セクションで、[Edit] (編集) を選択します。

[ポリシービューへの切り替え] が表示された場合はそれを選択して [キーポリシー] を表示してから、[編集] を選択します。

5. 次のポリシーブロックを KMS キーポリシーにコピーして、キーを使用する許可を GuardDuty に付与します。

```
{
    "Sid": "AllowGuardDutyKey",
    "Effect": "Allow",
    "Principal": {
        "Service": "guardduty.amazonaws.com"
    },
    "Action": "kms:GenerateDataKey",
    "Resource": "KMS key ARN",
    "Condition": {
        "StringEquals": {
            "aws:SourceAccount": "123456789012",
            "aws:SourceArn":
 "arn:aws:guardduty:Region2:123456789012:detector/SourceDetectorID"
        }
    }
}
```

6. ポリシーの例で#でフォーマットされている次の値を置き換えることにより、ポリシーを編集します。

1. *KMS key ARN* は、KMS キーの Amazon リソースネーム (ARN) に置き換えます。キー ARN を検索するには、「AWS Key Management Service 開発者ガイド」の「Finding the key ID and ARN」を参照してください。
2. *123456789012* は、検出結果をエクスポートする GuardDuty アカウントを所有する AWS アカウント ID に置き換えます。
3. *Region2* を、GuardDuty の結果が生成される に置き換え AWS リージョン ます。
4. *SourceDetectorID* は、検出結果が生成された特定のリージョンの GuardDuty アカウントの `detectorID` に置き換えます。

   アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

> **Note**
>
> オプトインリージョンで GuardDuty を使用している場合は、「Service」の値をそのリージョンのリージョンエンドポイントに置き換えます。例えば、中東 (バーレーン) (me-south-1) リージョンで GuardDuty を使用している場合は、`"Service": "guardduty.amazonaws.com"` を `"Service": "guardduty.me-south-1.amazonaws.com"` に置き換えます。オプトインリージョンごとのエンドポイントの詳細については、「GuardDuty エンドポイントとクォータ」を参照してください。

7. 最後のステートメントの前にポリシーステートメントを追加した場合は、このステートメントを追加する前にカンマを追加します。KMS キーポリシーの JSON 構文が有効であることを確認します。

   [保存] を選択します。
8. (オプション) 後のステップで使用するために、キー ARN をメモ帳にコピーします。

ステップ 3: Amazon S3 バケットにポリシーをアタッチする

検出結果のエクスポート先の Amazon S3 バケットに許可を追加して、GuardDuty がこの S3 バケットにオブジェクトをアップロードできるようにします。アカウントまたは別の に属する Amazon S3

バケットを使用するかどうかに関係なく AWS アカウント、これらのアクセス許可を追加する必要があります。

いずれかの時点で別の S3 バケットに検出結果をエクスポートすることを決定した場合、検出結果のエクスポートを継続するには、その S3 バケットに許可を追加し、検出結果のエクスポート設定を再度構成する必要があります。

これらの検出結果をエクスポートする Amazon S3 バケットがまだない場合は、「Amazon S3 ユーザーガイド」の「バケットの作成」を参照してください。

S3 バケットポリシーに許可をアタッチするには

1. [バケットポリシーを編集] ページが表示されるまで、「Amazon S3 ユーザーガイド」の「バケットポリシーを作成または編集するには」の手順を実行します。
2. 次のポリシー例は、Amazon S3 バケットに検出結果をエクスポートする許可を GuardDuty に付与する方法を示しています。検出結果のエクスポートを設定した後にパスを変更する場合は、新しい場所に対する許可を付与するように、ポリシーを変更する必要があります。

   次のポリシー例をコピーし、[バケットポリシーエディタ] に貼り付けます。

   最後のステートメントの前にポリシーステートメントを追加した場合は、このステートメントを追加する前にカンマを追加します。KMS キーポリシーの JSON 構文が有効であることを確認します。

   S3 バケットポリシーの例

   JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "Allow GetBucketLocation",
            "Effect": "Allow",
            "Principal": {
                "Service": "guardduty.amazonaws.com"
            },
            "Action": "s3:GetBucketLocation",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket",
            "Condition": {
                "StringEquals": {
```

```
                    "aws:SourceAccount": "123456789012",
                    "aws:SourceArn": "arn:aws:guardduty:us-
east-2:123456789012:detector/SourceDetectorID"

                }
            }
        },
        {
            "Sid": "Allow PutObject",
            "Effect": "Allow",
            "Principal": {
                "Service": "guardduty.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket[optional prefix]/
*",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "123456789012",
                    "aws:SourceArn": "arn:aws:guardduty:us-
east-2:123456789012:detector/SourceDetectorID"

                }
            }
        },
        {
            "Sid": "Deny unencrypted object uploads",
            "Effect": "Deny",
            "Principal": {
                "Service": "guardduty.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket[optional prefix]/
*",
            "Condition": {
                "StringNotEquals": {
                    "s3:x-amz-server-side-encryption": "aws:kms"
                }
            }
        },
        {
            "Sid": "Deny incorrect encryption header",
            "Effect": "Deny",
            "Principal": {
```

```
                "Service": "guardduty.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket[optional prefix]/
*",
            "Condition": {
                "StringNotEquals": {
                "s3:x-amz-server-side-encryption-aws-kms-key-id":
 "arn:aws:kms:us-east-2:111122223333:key/a1b2c3d4-5678-90ab-cdef-
EXAMPLE11111"
                }
            }
        },
        {
            "Sid": "Deny non-HTTPS access",
            "Effect": "Deny",
            "Principal": "*",
            "Action": "s3:*",
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket[optional prefix]/
*",
            "Condition": {
                "Bool": {
                    "aws:SecureTransport": "false"
                }
            }
        }
    ]
}
```

3. ポリシーの例で#でフォーマットされている次の値を置き換えることにより、ポリシーを編集します。

    1. *Amazon S3 bucket ARN* は、Amazon S3 バケットの Amazon リソースネーム (ARN) に置き換えます。[バケット ARN] は、https://console.aws.amazon.com/s3/ コンソールの [バケットポリシーを編集] ページで確認できます。
    2. *123456789012* は、検出結果をエクスポートする GuardDuty アカウントを所有する AWS アカウント ID に置き換えます。
    3. *us-east-2* は、GuardDuty の検出結果が生成される AWS リージョン に置き換えます。
    4. *SourceDetectorID* は、検出結果が生成された特定のリージョンの GuardDuty アカウントの `detectorID` に置き換えます。

アカウントと現在のリージョンの `detectorId` を検索するには、https://console.aws.amazon.com/guardduty/ コンソールの [設定] ページを参照するか、ListDetectors API を実行します。

5. *`S3 bucket ARN/[optional prefix]`* プレースホルダー値の *`[optional prefix]`* 部分は、検出結果をエクスポートするオプションのフォルダの場所に置き換えます。プレフィックスの使用の詳細については、「Amazon S3 ユーザーガイド」の「プレフィックスを使用してオブジェクトを整理する」を参照してください。

   オプションのフォルダの場所を指定する場合、そのフォルダは S3 バケットに既に存在している必要があります。GuardDuty はユーザーに代わってフォルダを作成しません。この要件は、バケットの所有権に関係なく適用されます。

6. *`KMS key ARN`* は、S3 バケットにエクスポートされた検出結果の暗号化に関連付けられた KMS キーの Amazon リソースネーム (ARN) に置き換えます。キー ARN を検索するには、「AWS Key Management Service 開発者ガイド」の「Finding the key ID and ARN」を参照してください。

> **Note**
>
> オプトインリージョンで GuardDuty を使用している場合は、「Service」の値をそのリージョンのリージョンエンドポイントに置き換えます。例えば、中東 (バーレーン) (me-south-1) リージョンで GuardDuty を使用している場合は、`"Service": "guardduty.amazonaws.com"` を `"Service": "guardduty.me-south-1.amazonaws.com"` に置き換えます。オプトインリージョンごとのエンドポイントの詳細については、「GuardDuty エンドポイントとクォータ」を参照してください。

4. [保存] を選択します。

ステップ 4 - S3 バケットに検出結果をエクスポートする (コンソール)

GuardDuty では、別の AWS アカウントの既存のバケットに検出結果をエクスポートできます。

新しい S3 バケットを作成するか、アカウントの既存のバケットを選択するときに、オプションのプレフィックスを追加できます。検出結果が書き込まれるプレフィックスとフォルダパスの両方が S3 バケットに既に存在している必要があります。GuardDuty はユーザーに代わってフォルダを作成しません。GuardDuty は検出結果オブジェクトをパス に書き込むため`/AWSLogs/`*`123456789012`*`/`

GuardDuty/*Region*、検出結果をエクスポートする前にこのフォルダ構造を作成する必要があります。

S3 オブジェクトのパス全体は *amzn-s3-demo-bucket*/*prefix-name*/UUID*.jsonl.gz* になります。UUID はランダムに生成され、ディテクター ID または検出結果 ID を表すものではありません。

> **Important**
>
> KMS キーおよび S3 バケットは同じリージョンにある必要があります。

これらのステップを完了する前に、KMS キーと既存の S3 バケットにそれぞれのポリシーをアタッチしていることを確認します。

検出結果のエクスポートを設定するには

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで [設定] を選択します。
3. [設定] ページの [検出結果のエクスポートオプション] で、[S3 バケット] の [今すぐ設定] (または必要に応じて [編集]) を選択します。
4. [S3 バケット ARN] で、**bucket ARN** を入力します。バケット ARN を検索するには、「Amazon S3 ユーザーガイド」の「S3 バケットのプロパティを表示するには」を参照してください。
5. [KMS キー ARN] で、**key ARN** を入力します。キー ARN を検索するには、「AWS Key Management Service 開発者ガイド」の「Finding the key ID and ARN」を参照してください。
6. ポリシーをアタッチします。
   - S3 バケットポリシーをアタッチする手順を実行します。詳細については、「ステップ 3: Amazon S3 バケットにポリシーをアタッチする」を参照してください。
   - KMS キーポリシーをアタッチする手順を実行します。詳細については、「ステップ 2 – KMS キーにポリシーをアタッチする」を参照してください。
7. [保存] を選択します。

ステップ 5 – 更新されたアクティブな検出結果をエクスポートする頻度を設定する

環境に応じて、更新されたアクティブな検出結果をエクスポートする頻度を設定します。デフォルトでは、更新された検出結果は 6 時間ごとにエクスポートされます。つまり、最新のエクスポート後に更新された検出結果が、次のエクスポートに含まれます。更新された検出結果が 6 時間ごとにエクスポートされ、エクスポートが 12:00 に発生した場合、12:00 以降に更新した検出結果が 18:00 にエクスポートされます。

頻度を設定するには

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. [Settings] (設定) を選択します。
3. [Findings export options] (結果のエクスポートオプション) セクションで、[Frequency for updated findings] (更新された結果の頻度) を選択します。これにより、更新されたアクティブな検出結果を EventBridge と Amazon S3 の両方にエクスポートする頻度が設定されます。次から選択できます。
   - [EventBridge と S3 を 15 分ごとに更新]
   - [EventBridge と S3 を 1 時間ごとに更新]
   - [EventBridge と S3 を 6 時間ごとに更新 (デフォルト)]
4. [Save changes] (変更の保存) をクリックします。

Amazon EventBridge を使用した GuardDuty 検出結果の処理

GuardDuty は、サーバーレスのイベントバスサービスである Amazon EventBridge (旧Amazon CloudWatch Events) に、検出結果をイベントとして自動的に公開 (送信) します。EventBridge は、アプリケーションやサービスから Amazon Simple Notification Service (Amazon SNS) トピック、AWS Lambda 関数、Amazon Kinesis ストリームなどのターゲットにほぼリアルタイムのデータストリームを配信します。詳細については、「Amazon EventBridge ユーザーガイド」を参照してください。

EventBridge は、イベントを受信することで GuardDuty の検出結果の自動モニタリングと処理を可能にします。EventBridge は、新規に生成された検出結果と集約された検出結果の両方に対するイベントを受信します。集約された検出結果では、既存の検出結果のその後の発生が元の検出結果と結合されます。Every GuardDuty の検出結果にはそれぞれ検索結果 ID が割り当てられま

す。GuardDuty によって、一意の結果 ID を持つ各検索結果に対して EventBridge イベントが作成されます。GuardDuty での集約の仕組みについては、「GuardDuty の検出結果の集約」を参照してください。

自動化されたモニタリングと処理に加えて、EventBridge を使用すると、調査結果データの長期保存が可能になります。GuardDuty は検出結果を 90 日間保存します。EventBridge を使用すると、調査結果データを好みのストレージプラットフォームに送信し、データをいつまででも保存できます。検出結果を長期間保持するために、GuardDuty は 生成された検出結果を Amazon S3 にエクスポートする をサポートしています。

トピック

- GuardDuty での EventBridge 通知頻度について
- Amazon SNS トピックとエンドポイントを設定する (E メール、Slack、Amazon Chime)
- Amazon EventBridge を使用した GuardDuty の検出結果
- GuardDuty の検出結果に対する EventBridge ルールの作成
- GuardDuty マルチアカウント環境向け EventBridge ルール

GuardDuty での EventBridge 通知頻度について

このセクションでは、EventBridge を介して検出通知を受け取る頻度と、以降の検出発生時の頻度を更新する方法について説明します。

一意の検出結果 ID を含む新しく生成された検出結果の通知

GuardDuty は、一意の検出 ID を持つ検出結果を生成した際に、これらの通知をほぼリアルタイムで送信します。通知には、通知生成プロセス中のこの結果 ID の後続のすべての出現が含まれます。

新規に生成された所見の通知頻度は、ほぼリアルタイムで行われます。デフォルトでは、この頻度を変更することはできません。

以降に検出結果が見つかった場合の通知

GuardDuty は、6 時間間隔以内で、特定の検出結果の後続のすべての発生を単一のイベントとして集計します。管理者アカウントのみが、後続の検出結果の発生について EventBridge 通知頻度を更新できます。メンバーアカウントは、自身のアカウントに対してこの頻度を更新できません。例えば、委任 GuardDuty 管理者アカウントが頻度を 1 時間に更新した場合、すべてのメンバーアカウントには、EventBridge に送信された後続の検出結果の発生に関する 1 時間の通知頻

度もあります。詳細については、「Amazon GuardDuty での複数のアカウントの管理」を参照してください。

管理者アカウントであれば、その後の検出結果の発生に関する通知のデフォルトの頻度はカスタマイズできます。有効な値は、15 分、1 時間、またはデフォルトの 6 時間です。これらの通知の頻度の設定については、ステップ 5 – 更新されたアクティブな検出結果をエクスポートする頻度を設定する を参照してください。

メンバーアカウントの EventBridge 通知を受信する管理者アカウントの詳細については、「マルチアカウント環境向け EventBridge ルール」を参照してください。

Amazon SNS トピックとエンドポイントを設定する (E メール、Slack、Amazon Chime)

Amazon Simple Notification Service (Amazon SNS) は、パブリッシャーからサブスクライバーへのメッセージ配信を提供するフルマネージドサービスです。発行者は、トピックにメッセージを送信することで、受信者と非同期的に通信します。トピックは、Amazon Simple Queue Service (Amazon SQS) AWS Lambda、HTTP/S、E メールアドレスなどの複数のエンドポイントをグループ化できる論理アクセスポイントと通信チャネルです。

> **Note**
>
> イベントルールの作成中または作成後に、Amazon SNS トピックを優先する EventBridge イベントルールに追加できます。

Amazon SNS トピックを作成する

まず、Amazon Simple SNS でトピックを設定し、エンドポイントを追加する必要があります。トピックを作成するには、「Amazon Simple Notification Service デベロッパーガイド」の「ステップ 1: トピックを作成する」の手順を実行します。トピックが作成されたら、トピック ARN をクリップボードにコピーします。このトピック ARN を使用して、推奨される設定のいずれかを続行します。

GuardDuty の検出データを送信する場所を設定する際に、希望の方法を選択してください。

Email setup

E メールエンドポイントを設定するには

Create an Amazon SNS topic の後、次のステップは、このトピックへのサブスクリプションを作成することです。「Amazon Simple Notification Service デベロッパーガイド」の「ステップ 2: Amazon SNS トピックへのサブスクリプションを作成」の手順を実行します。

1. トピック ARN には、 Create an Amazon SNS topic ステップで作成したトピック ARN を使用します。トピック ARN は以下のようになります。

   ```
   arn:aws:sns:us-east-2:123456789012:your_topic
   ```

2. [Protocol] で [Email] を選択します。
3. [エンドポイント] には、Amazon SNS からの通知を受信する E メールアドレスを入力してください。

   サブスクリプションの作成後、E メールクライアントを通じてそれを確認する必要があります。

Slack setup

チャット アプリケーションクライアント - Slack で Amazon Q Developer を設定するには

Create an Amazon SNS topic の後、次のステップは Slack 用にクライアントを設定することです。

「Amazon Q Developer in chat applications 管理者ガイド」の「チュートリアル: Slack の使用を開始する」の手順を実行します。

Chime setup

チャット アプリケーション - Chime で Amazon Q Developer を設定するには

Create an Amazon SNS topic の後、次のステップは Chime 用に Amazon Q Developer を設定することです。

「チャットアプリケーションの Amazon Q Developer 管理者ガイド」の「チュートリアル: Amazon Chime の使用を開始する」の手順を実行します。

Amazon EventBridge を使用した GuardDuty の検出結果

EventBridge では、モニタリングするイベントを指定するルールを作成します。これらのルールは、これらのイベントが発生した場合に自動化されたアクションを実行できる対象サービスおよびアプリケーションも指定します。ターゲットとは、ルールに定義されたイベントパターンにイベントがマッチしたときに、EventBridge がイベントを送信する宛先 (リソースまたあエンドポイント) です。各イベントは、 AWS イベントの EventBridge スキーマに準拠する JSON オブジェクトであり、調査結果の JSON 表現が含まれています。特定の基準を満たすイベントのみを送信するようにルールを調整できます。詳細については、「the section called “イベントスキーマ”」を参照してください。検出結果データは EventBridge イベントとして構成されているため、他のアプリケーション、サービス、およびツールを使用して、検出結果をモニタリングし、処理し、それに対してアクションを取ることができます。

イベントに基づいて GuardDuty の検出結果に関する通知を受け取るには、EventBridge ルールおよび GuardDuty のターゲットを作成する必要があります。このルールでは、EventBridge は GuardDuty が生成する検出結果の通知を、ルールで指定されたターゲットに送信できます。

> **Note**
>
> EventBridge と CloudWatch のイベントは同じ基盤のサービスと API です。ただし、EventBridge には、SaaS (Software as a Service (SaaS)) アプリケーションやユーザー独自のアプリケーションからイベントを受け取るのに役立つ追加機能が含まれています。基盤となるサービスと API が同じであるため、GuardDuty の検出結果のイベントスキーマも同じです。

GuardDuty におけるアーカイブ済みおよび非アーカイブ済みの検出結果が EventBridge と連携する方法

手動でアーカイブされた検出結果の場合、最初の検出結果と、それ以降に発生したすべての検出結果 (アーカイブ完了後に生成された検出結果) は、特定の頻度ごとに EventBridge に送信されます。詳細については、「GuardDuty での EventBridge 通知頻度について」を参照してください。

抑制ルール で自動的にアーカイブされた検出結果の場合、最初の検出結果と、それ以降に発生したすべての検出結果 (アーカイブ完了後に生成された検出結果) は、EventBridge に送信されません。自動的にアーカイブされたこれらの検出結果は GuardDuty コンソールに表示できます。

イベントスキーマ

イベントパターンは、EventBridge がターゲットにイベントを送信するかどうかを決定するために使用するデータを定義します。GuardDuty 向けの EventBridge イベントは、以下の形式になっています。

```
{
        "version": "0",
        "id": "cd2d702e-ab31-411b-9344-793ce56b1bc7",
        "detail-type": "GuardDuty Finding",
        "source": "aws.guardduty",
        "account": "111122223333",
        "time": "1970-01-01T00:00:00Z",
        "region": "us-east-1",
        "resources": [],
        "detail": {GUARDDUTY_FINDING_JSON_OBJECT}
       }
```

`detail` 値は、配列内で複数の検出結果をサポートする「検出結果」応答全体を返すのではなく、単一の検出結果の JSON の詳細をオブジェクトとして返します。

`GUARDDUTY_FINDING_JSON_OBJECT` に含まれているすべてのパラメータの完全なリストについては、「GetFindings」を参照してください。`GUARDDUTY_FINDING_JSON_OBJECT` に表示される `id` パラメータは、上記で説明した検出結果 ID です。

GuardDuty の検出結果に対する EventBridge ルールの作成

次の手順では、Amazon EventBridge コンソールと AWS Command Line Interface (AWS CLI) を使用して、GuardDuty の検出結果の EventBridge ルールを作成する方法について説明します。このルールは、GuardDuty の検出結果のイベントスキーマとパターンを使用する EventBridge イベントを検出し、それらのイベントを AWS Lambda 関数に送信して処理します。

AWS Lambda は、サーバーのプロビジョニングや管理を行わずにコードを実行するために使用できるコンピューティングサービスです。コードをパッケージ化し、Lambda 関数 AWS Lambda としてにアップロードします。 AWS Lambda その後、 は関数が呼び出されたときに関数を実行します。関数は、ユーザーが手動で呼び出したり、イベントに応答して自動的に呼び出したり、またはアプリケーションやサービスからのリクエストに応答したりすることができます。Lambda 関数の作成および呼び出しについては、 AWS Lambda 開発者ガイドを参照してください。

GuardDuty の検出結果をターゲットに送信する EventBridge ルールを作成する際に、お好みの方法を選択してください。

Console

Amazon EventBridge コンソールを使用して、すべての GuardDuty の検出結果イベントを Lambda 関数に自動的に送信して処理するルールを作成するには、次のステップに従います。このルールは、特定のイベントを受信したときに実行されるルールのデフォルト設定を使用します。ルール設定の詳細や、カスタム設定を使用するルールの作成方法については、Amazon EventBridge ユーザーガイドのイベントに反応するルールの作成を参照してください。

このルールを作成する前に、Lambda 関数を作成して、ルールがターゲットとして使用されるようにします。ルールを作成するときは、この関数をルールのターゲットとして指定する必要があります。ターゲットは、以前に作成した SNS トピックも選択できます。詳細については、「Amazon SNS トピックとエンドポイントを設定する (E メール、Slack、Amazon Chime)」を参照してください。

コンソールを使用してイベントのルールを作成するには

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/events/ で Amazon EventBridge コンソールを開きます。
2. ナビゲーションペインの [バス] で、[ルール] を選択します。
3. セクションで、ルールの作成 を選択します。
4. 詳細のルール定義 で、次の操作を行います。
    a. 名前 にルールの名前を入力します。
    b. 説明 に、認可ルールの簡単な説明を入力します。
    c. イベントバスを選択 の下で、 デフォルトのイベントバスが選択され、選択したイベントバスのルールを有効にするがオンになっていることを確認します。
    d. ルールタイプ では、イベントパターンを持つルール を選択します。
    e. 終了したら、次へ を選択します。
5. イベントパターンの作成 で、次の操作を行います。
    a. [イベントソース] で、[AWS イベントまたは EventBridge パートナーイベント] を選択してください。
    b. (オプショナル) サンプルイベント については、GuardDuty 用のサンプル検出結果イベントを確認して、イベントに含まれる可能性がある内容を確認してください。そのためには、[AWS イベント] を選択します。次に、[サンプルイベント] で [GuardDuty 検出結果] を選択します。

c. オプション 1 - EventBridge が提供するテンプレートであるパターンフォームを使用する

[イベントパターン] セクションで次を実行できます。

1. [作成方法] では [パターンフォームを使用する] を選択します。
2. イベントソース で AWS のサービス を選択してください。
3. AWS のサービス で、GuardDuty を選択します。
4. [イベントタイプ] では、[GuardDuty の検出結果] を選択します。

終了したら、次へ を選択します。

d. オプション 2 - JSON でのカスタムイベントパターンの使用

[イベントパターン] セクションで次を実行できます。

1. [作成方法] で [カスタムパターン (JSON エディター)] を選択します。
2. イベントパターンでは、以下のカスタム JSON を貼り付けてください。これにより、中程度、高、重大な検出結果に対するアラートが生成されます。詳細については、「検出結果の重大度レベル」を参照してください。

```
{
  "source": [
    "aws.guardduty"
  ],
  "detail-type": [
    "GuardDuty Finding"
  ],
  "detail": {
    "severity": [
      4,
      4.0,
      4.1,
      4.2,
      4.3,
      4.4,
      4.5,
      4.6,
      4.7,
      4.8,
      4.9,
      5,
```

```
        5.0,
        5.1,
        5.2,
        5.3,
        5.4,
        5.5,
        5.6,
        5.7,
        5.8,
        5.9,
        6,
        6.0,
        6.1,
        6.2,
        6.3,
        6.4,
        6.5,
        6.6,
        6.7,
        6.8,
        6.9,
        7,
        7.0,
        7.1,
        7.2,
        7.3,
        7.4,
        7.5,
        7.6,
        7.7,
        7.8,
        7.9,
        8,
        8.0,
        8.1,
        8.2,
        8.3,
        8.4,
        8.5,
        8.6,
        8.7,
        8.8,
        8.9,
        9,
```

```
      9.0,
      9.1,
      9.2,
      9.3,
      9.4,
      9.5,
      9.6,
      9.7,
      9.8,
      9.9,
      10,
      10.0
    ]
  }
}
```

終了したら、次へ を選択します。

6. オプション A - AWS のサービス ターゲット AWS Lambda としての選択

   [ターゲットを選択] ページで、次の操作を行います。

   a. [ターゲットタイプ] で [AWS のサービス] を選択します。

   b. [Select a target] (ターゲットを選択) では、[Lambda function] (Lambda 関数) を選択します。次に、関数で、結果イベントの送信先となる Lambda 関数を選択します。

   c. (オプション) バージョン/エイリアスを設定 で、ターゲットの Lambda 関数のバージョンまたはエイリアスの設定を入力します。

   d. (オプショナル) [追加設定] で、Lambda 関数に送信するイベントデータを指定します。関数に正常に配信されないイベントを処理する方法を指定することもできます。

   e. 終了したら、次へ を選択します。

7. オプション B - SNS トピックをターゲットとして選択

   [ターゲットを選択] ページで、次の操作を行います。

   a. [ターゲットタイプ] で [AWS のサービス] を選択します。

   b. [Select a target] (ターゲットの選択) には、[SNS topic] (SNS トピック) を選択してください。次に、ターゲットロケーションで、ターゲットロケーションに基づいて適切なオプションを選択してください。[トピック] で、作成した SNS トピックの名前を選択します。

c. [追加の設定] を展開します。[ターゲット入力の設定] では、[入力トランスフォーマー] を選択します。

d. 入力トランスフォーマーの設定 を選択します。

e. 次のコードをコピーして、[ターゲット入力トランスフォーマー] セクションの [入力パス] フィールドに貼り付けます。

```
{
    "severity": "$.detail.severity",
    "Account_ID": "$.detail.accountId",
    "Finding_ID": "$.detail.id",
    "Finding_Type": "$.detail.type",
    "region": "$.region",
    "Finding_Description": "$.detail.description"
}
```

f. 次のコードをコピーして [テンプレート] フィールドに貼り付け、E メールをフォーマットします。

```
{"message":"You have a severity <severity> GuardDuty finding type
 <Finding_Type> in the <region> Region.\n\nFinding Description:
\n<Finding_Description>\n\nFor more details open the GuardDuty console at
 https://console.aws.amazon.com/guardduty/home?region=<region>#/findings?
search=id%3D<Finding_ID>"}
```

> **Important**
>
> テンプレートを単一行の JSON オブジェクトとして指定します。前の例messageに示すように、メッセージは などのキーでラップされます。\n を使用して改行を追加します。
> などの GuardDuty 検出結果フィールドには、改行を含むエスケープされていないコントロール文字を含めるdescriptionことができます。次のいずれかのテンプレート形式を使用しないでください。
>
> - 裸引用符で囲まれた文字列
> - 別々の行で別々の引用符で囲まれた文字列を使用する複数行テンプレート

これらの形式では、置換された値が無効な JSON を生成する可能性があります。その後、EventBridge はイベントを拒否し、通知を配信しません。

> **Note**
>
> E メールサブスクリプションはテンプレート出力をそのまま受け取るため、E メールには `message` JSON オブジェクトが表示され、改行は として表示されます\n。フォーマットされた通知の場合は、チャットアプリケーション (Slack または Amazon Chime) で Amazon Q Developer に Amazon SNS トピックを配信するか、Lambda 関数をターゲットとして使用します。

8. タグの設定 ページで、ルールに割り当てる 1 つ以上のタグをオプションで入力します。続いて、次へ を選択します。
9. 確認して作成するステップでは、ジョブの設定設定を確認し、それらが正しいことを検証します。

   設定を変更するには、設定が含まれるセクションで 編集を選択し、次に正しい設定を入力します。ナビゲーションタブを使用して、設定が含まれるページに移動することもできます。
10. 設定の確認が完了したら Create rule (ルールの作成) を選択します。

API

次の手順は、 AWS CLI コマンドを使用して GuardDuty の EventBridge ルールとターゲットを作成する方法を示しています。具体的には、この手順では、GuardDuty が生成するすべての検出結果について、EventBridge がイベントを送信できるようにするルールを作成し、そのルールのターゲットとして AWS Lambda 関数に送信する方法を示しています。

> **Note**
>
> この例では、EventBridge をトリガーするルールのターゲットとして Lambda 関数を使用しています。他の AWS リソースをターゲットとして設定して、EventBridge をトリガーすることもできます。GuardDuty と EventBridge は、Amazon EC2 インスタンス、Amazon Kinesis ストリーム、Amazon ECS タスク、 AWS Step Functions ステートマシン、 `run` コマンド、組み込みターゲットのターゲットタイプをサポートしていま

す。詳細については、「Amazon EventBridge API リファレンス」の「PutTargets」を参照してください。

ルールおよびターゲットを作成するには

1. GuardDuty が生成するすべての検出結果に対して、EventBridge がイベントを送信できるようにするルールを作成するには、次の EventBridge CLI コマンドを実行します。

```
aws events put-rule --name your-rule-name --event-pattern "{\"source\":
[\"aws.guardduty\"]}"
```

GuardDuty で生成された検出結果のサブセットにのみイベントを送信するよう EventBridge に指示できるように、ルールをさらにカスタマイズできます。このサブセットは、ルールで指定されている検出結果の属性に基づきます。例えば、次の CLI コマンドを使用して、EventBridge で重大度が 5 または 8 の GuardDuty の検出結果でイベントのみ送信できるようにルールを作成します。

```
aws events put-rule --name your-rule-name --event-pattern "{\"source\":
[\"aws.guardduty\"],\"detail-type\":[\"GuardDuty Finding\"],\"detail\":
{\"severity\":[5,8]}}"
```

この目的のために、GuardDuty の検出結果のための JSON で利用可能な任意のプロパティ値を使用できます。

2. ステップ 1 で作成したルールのターゲットとして Lambda 関数をアタッチするには、次の EventBridge CLI コマンドを実行します。

```
aws events put-targets --rule your-rule-name --targets
 Id=1,Arn=arn:aws:lambda:us-east-1:111122223333:function:your-function
```

をステップ 1 で作成したルールの名前`your-rule-name`に置き換え、 を GuardDuty イベントの実際の Lambda 関数`your-function`に置き換えてください。

3. ターゲットを呼び出す上で必要な許可を追加するには、次の Lambda CLI コマンドを実行します。

```
aws lambda add-permission --function-name your-function --statement-id 1 --
action 'lambda:InvokeFunction' --principal events.amazonaws.com --source-arn
 arn:aws:events:us-east-1:111122223333:rule/your-rule-name
```

を GuardDuty イベントの実際の Lambda 関数`your-function`に置き換え、 をステップ 1 で作成したルールの名前`your-rule-name`に置き換えてください。`--source-arn` 条件は、このルールのみが関数を呼び出すことを許可します。

GuardDuty マルチアカウント環境向け EventBridge ルール

委任された GuardDuty 管理者アカウントを使用する場合、メンバーアカウントで生成されたイベントを表示し、他のアプリケーションやサービスを使用してアクションを実行できます。管理者アカウント内の EventBridge ルールは、メンバーアカウントからの該当する検出結果に基づいてトリガーされます。管理者アカウントで EventBridge を介した検出結果の設定を行うと、ご自身のアカウントとメンバーアカウントの両方からの検出結果を受け取ることができます。例えば、EventBridge を使用して、特定のタイプの検出結果を Lambda 関数に送信し、その関数でデータを処理してセキュリティインシデントおよびイベント管理 (SIEM) システムに送信することができます。

検出結果の JSON 詳細の `accountId` フィールドを使用して、GuardDuty の検出結果元となったメンバーアカウントを特定することができます。特定のメンバーアカウント向けにカスタムイベントルールを作成するには、新規ルールを作成し、イベントパターンで以下のテンプレートを使用してください。*123456789012* を、イベントをトリガーするメンバーアカウントの `accountId` に置き換えます。

```
{
  "source": [
    "aws.guardduty"
  ],
  "detail-type": [
    "GuardDuty Finding"
  ],
  "detail": {
    "accountId": [
      "123456789012"
    ]
  }
}
```

> **Note**
>
> この例は、指定されたアカウント ID からのすべての検出結果に一致するルールを作成します。複数のアカウント ID は、JSON 構文に従いカンマで区切って指定できます。

CloudWatch Logs を監査する方法と、Malware Protection for EC2 スキャン中にリソースがスキップされる理由について

GuardDuty Malware Protection for EC2 は Amazon CloudWatch ロググループ /aws/guardduty/malware-scan-events にイベントを発行します。マルウェアスキャンに関連する各イベントについて、影響を受けるリソースのステータスとスキャン結果を監視できます。特定の Amazon EC2 リソースと Amazon EBS ボリュームが、Malware Protection for EC2 スキャン中にスキップされた可能性があります。

GuardDuty Malware Protection for EC2 の CloudWatch Logs の監査

CloudWatch ロググループ /aws/guardduty/malware-scan-events でサポートされるスキャンイベントには 3 つのタイプがあります。

Malware Protection for EC2 スキャンイベント名	説明
`EC2_SCAN_STARTED`	GuardDuty Malware Protection for EC2 が、EBS ボリュームのスナップショットを作成する準備など、マルウェアスキャンのプロセスを開始するときに作成されます。
`EC2_SCAN_COMPLETED`	影響を受けるリソースの EBS ボリュームの少なくとも 1 つに対して GuardDuty Malware Protection for EC2 スキャンが完了したときに作成されます。このイベントには、スキャンされた EBS ボリュームに属する `snapshotId` も含まれます。スキャンが完了すると、スキャン結果は、`CLEAN`、`THREATS_FOUND` 、または `NOT_SCANNED` のいずれかになります。

Malware Protection for EC2 スキャンイベント名	説明
`EC2_SCAN_SKIPPED`	GuardDuty Malware Protection for EC2 スキャンが影響を受けるリソースのすべての EBS ボリュームをスキップするときに作成されます。スキップ理由を特定するには、対応するイベントを選択し、詳細を表示します。スキップの理由の詳細については、以下の「マルウェアスキャン中にリソースをスキップする理由」を参照してください。

> **Note**
>
> を使用している場合 AWS Organizations、Organizations のメンバーアカウントからの CloudWatch ログイベントは、管理者アカウントとメンバーアカウントのロググループの両方に発行されます。

任意のアクセス方法を選択して、CloudWatch イベントを表示およびクエリします。

Console

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/cloudwatch/ で CloudWatch コンソールを開きます。
2. ナビゲーションペインの [ログ] で、[ロググループ] を選択します。ロググループ /aws/guardduty/malware-scan-events を選択し、GuardDuty Malware Protection for EC2 のスキャンイベントを表示します。

   クエリを実行するには、[Log Insights] (ログのインサイト) を選択してください

   クエリの実行については、「Amazon CloudWatch Logs ユーザーガイド」の「 CloudWatch Logs Insights による ログデータの分析」を参照してください。
3. [Scan ID] (スキャン ID) を選択して、影響を受けたリソースとマルウェアの検出結果の詳細を監視します。例えば、次のクエリを実行し、CloudWatch ログイベントを `scanId` でフィルタリングできます。ご自身の有効な *`Scan-ID`* を使用してください。

```
fields @timestamp, @message, scanRequestDetails.scanId as scanId
| filter scanId like "77a6f6115da4bd95f4e4ca398492bcc0"
| sort @timestamp asc
```

API/CLI

- ロググループを操作するには、「Amazon CloudWatch ユーザーガイド」の「AWS CLIを使用したログエントリの検索」を参照してください。

  ロググループ /aws/guardduty/malware-scan-events を選択し、GuardDuty Malware Protection for EC2 のスキャンイベントを表示します。
- ログイベントを表示してフィルタリングするには、「Amazon CloudWatch API リファレンス」の「GetLogEvents」および「FilterLogEvents」をそれぞれ参照してください。

GuardDuty Malware Protectionfor EC2 ログの保持

ロググループ /aws/guardduty/malware-scan-events のデフォルトのログ保持期間は 90 日で、その後ログイベントは自動的に削除されます。CloudWatch ロググループのログ保持ポリシーを変更するには、「Amazon CloudWatch ユーザーガイド」の「CloudWatch Logs でのログデータ保持を変更する」または「Amazon CloudWatch API リファレンス」の「PutRetentionPolicy」を参照してください。

マルウェアスキャン中にリソースをスキップする理由

マルウェアスキャンに関連するイベントでは、特定の EC2 リソースと EBS ボリュームがスキャンプロセス中にスキップされた可能性があります。次の表に、GuardDuty Malware Protection for EC2 がリソースをスキャンしない理由を示します。該当する場合は、提案された手順を使用してこれらの問題を解決し、次回 GuardDuty Malware Protection for EC2 がマルウェアスキャンを開始したときにこれらのリソースをスキャンするようにします。その他の問題は、イベントの経過を知らせるために使用されるものであり、対応不要です。

スキップの理由	説明	提案されるステップ	
RESOURCE_NOT_FOUND	オンデマンドマルウェアスキャンの開始にresourceArn 提	Amazon EC2 インスタンスの resourceArn または コンテナ	

スキップの理由	説明	提案されるステップ	
	供された は、 AWS 環境で見つかりませんでした。	のワークロードを検証して、もう一度試してください。	
ACCOUNT_INELIGIBLE	オンデマンドマルウェアスキャンを開始しようとした AWS アカウント ID で GuardDuty が有効になっていません。	この AWS アカウントで GuardDuty が有効になっていることを確認します。 新しい で GuardDuty を有効にする AWS リージョン と、同期に最大 20 分かかる場合があります。	

スキップの理由	説明	提案されるステップ	
UNSUPPORTED_KEY_ENCRYPTION	GuardDuty Malware Protection for EC2 は、暗号化されていないボリュームと、カスタマーマネージドキーで暗号化されたボリュームをサポートします。Amazon EBS 暗号化を使用して暗号化された EBS ボリュームのスキャンはサポートされていません。 現在、リージョンによってはこのスキップ理由が当てはまらないことがあります。詳細については AWS リージョン、「」を参照してくださいリージョン固有機能の可用性。	暗号化キーをカスタマーマネージドキーに置き換えます。GuardDuty がサポートする暗号化タイプの詳細については、「マルウェアスキャンでサポートされている Amazon EBS ボリューム」を参照してください。	

スキップの理由	説明	提案されるステップ	
EXCLUDED_BY_SCAN_SETTINGS	EC2 インスタンスまたは EBS ボリュームは、マルウェアスキャン中に除外されました。2 つの可能性があります。タグが対象リストに追加されたが、リソースがこのタグに関連付けられていないか、タグが除外リストに追加され、リソースがそのタグに関連付けられている、または、GuardDuty Excluded タグがそのリソースに対し true に設定されています。	スキャンオプションまたは Amazon EC2 リソースに関連付けられているタグを更新します。詳細については、「ユーザー定義タグ付きのスキャンオプション」を参照してください。	
UNSUPPORTED_VOLUME_SIZE	ボリュームが 2,048 GB を超えています。	実用的ではありません。	
NO_VOLUMES_ATTACHED	GuardDuty Malware Protection for EC2 がアカウントでインスタンスを検出しましたが、このインスタンスにスキャンを続行するための EBS ボリュームがアタッチされていません。	実用的ではありません。	
UNABLE_TO_SCAN	内部サービスエラー。	実用的ではありません。	

スキップの理由	説明	提案されるステップ	
SNAPSHOT_NOT_FOUND	EBS ボリュームから作成され、サービスアカウントと共有されているスナップショットが見つかりませんでした。GuardDuty Malware Protection for EC2 はスキャンを続行できませんでした。	CloudTrail をチェックして、スナップショットが意図的に削除されていないことを確認します。	
SNAPSHOT_QUOTA_REACHED	各リージョンのスナップショットに許可されている最大ボリュームに達しました。これにより、スナップショットの保持だけでなく、新しいスナップショットの作成もできなくなります。	古いスナップショットを削除するか、クォータの増加をリクエストできます。リージョンごとのスナップショットのデフォルト制限と、クォータの増加をリクエストする方法については、「AWS 全般のリファレンスガイド」の「Service Quotas」を参照してください。	

スキップの理由	説明	提案されるステップ	
MAX_NUMBER_OF_ATTACHED_VOLUMES_REACHED	EC2 インスタンスに 11 を超えた EBS ボリュームがアタッチされました。GuardDuty Malware Protection for EC2 は、deviceName のアルファベット順に取得された EBS ボリュームの最初の 11 個をスキャンしました。	実用的ではありません。	

スキップの理由	説明	提案されるステップ	
UNSUPPORTED_PRODUCT_CODE_TYPE	GuardDuty は、productCode を marketplace として持つほとんどのインスタンスをスキャンできます。一部のマーケットプレイスインスタンスはスキャンの対象外となる場合があります。GuardDuty はそのようなインスタンスをスキップし、理由を UNSUPPORTED_PRODUCT_CODE_TYPE として口グに記録します。このサポートは、 AWS GovCloud (US) および中国リージョンによって異なります。詳細については、「リージョン固有機能の可用性」を参照してください。 詳細については、「Amazon EC2 ユーザーガイド」の「有料 AMI」を参照してください。productCode の詳細については、「Amazon EC2 API	実用的ではありません。	

スキップの理由	説明	提案されるステップ	
	リファレンス」の「ProductCode」を参照してください。		

Malware Protection for EC2 の誤検出の報告

GuardDuty Malware Protection for EC2 スキャンにより、Amazon EC2 インスタンスまたはコンテナワークロード内の無害なファイルが悪質または有害であると識別されることがあります。Malware Protection for EC2 と GuardDuty サービスのエクスペリエンスを向上させるために、スキャン中に悪質または有害であると識別されたファイルに実際にはマルウェアが含まれていないと思われる場合は、誤検出結果を報告できます。

Amazon EC2 マルウェアスキャンの結果を偽陽性として報告するには

プロセスを開始するには、 にお問い合わせください サポート。スキャンされたリソースの詳細を指定するには、次の手順に従います。

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. [EC2 マルウェアスキャン] を選択します。
3. スキャンを選択してその [Finding ID] (検出結果 ID) を表示します。
4. [Finding ID] (検出結果 ID) を指定します。ファイルの SHA-256 ハッシュを指定する必要があります。これは、GuardDuty Malware Protection for EC2 が正しいファイルを受信していることを確認するために必要です。
5. サポート チームは、悪意のある可能性のあるファイルと SHA-256 ハッシュをアップロードするために使用できる Amazon Simple Storage Service (Amazon S3) の署名付き URL を提供します。スキャンされたオブジェクトをアップロードする手順については、「Amazon S3 ユーザーガイド」の「署名付き URL を使用したオブジェクトのアップロード」を参照してください。
6. ファイルをアップロードしたら、 サポート チームに通知します。

   サポート は、ファイルを受信した後、確認応答を提供します。GuardDuty サービスのチームメンバーは提出物を分析し、Malware Protection for EC2 と GuardDuty サービスのエクスペリエンスを向上させるための適切な措置を講じます。 サポート チームはケースのステータス更新を引き続き提供します。GuardDuty で S3 オブジェクトが保持される期間は最大 30 日です。

Malware Protection for S3 で S3 オブジェクトスキャンの結果を偽陽性として報告

Malware Protection for S3 スキャンによって、オブジェクトが悪意を秘めているか害を及ぼす可能性があるものとして識別されることがあります。指摘された S3 オブジェクトにマルウェアは含まれていないと考えられる場合は、このマルウェアスキャンの結果を偽陽性として報告してください。

Malware Protection for S3 を個別に使用しているときでも、偽陽性レポートを発行できます。この場合、GuardDuty は検出結果を生成しません。スキャンのステータスと結果のステータスを確認する方法については、「S3 オブジェクトスキャンのモニタリング」を参照してください。

S3 オブジェクトマルウェアスキャンの結果を偽陽性として報告するには

プロセスを開始するには、 にお問い合わせください サポート。次の手順を使用して、スキャンされた S3 オブジェクトの詳細を提供します。

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ユースケースに応じて、適切な手順を選択します。

   Using Malware Protection for S3 with GuardDuty

   1. ナビゲーションペインで 調査検出結果 を選択します。
   2. [検出結果] ページで、偽陽性検出を選択してその詳細を表示します。
   3. 検出結果の詳細を確認して、[検出結果 ID]、[リージョン]、保護された S3 バケットの [名前]、スキャンされたオブジェクトの [キー] を指定します。

      [項目パス] の詳細から、オブジェクトの [ハッシュ] を指定します。これは、GuardDuty が正しいファイルを受信していることを確認するために必要です。

   Using Malware Protection for S3 independently

   保護された S3 バケット名、スキャンされたオブジェクト名、 AWS リージョンを指定します。
3. サポート チームは、悪意のある可能性のあるファイルとハッシュをアップロードするために使用できる Amazon Simple Storage Service (Amazon S3) の署名付き URL を提供します。スキャンされたオブジェクトをアップロードする手順については、「Amazon S3 ユーザーガイド」の「署名付き URL を使用したオブジェクトのアップロード」を参照してください。

4. S3 オブジェクトをアップロードしたら、 サポート チームに通知します。

サポート は、オブジェクトの受信を承認します。GuardDuty サービスのチームメンバーは提出物を分析し、Malware Protection for S3 と GuardDuty サービスのエクスペリエンスを向上させるための適切な措置を講じます。 サポート チームはケースのステータス更新を引き続き提供します。GuardDuty で S3 オブジェクトが保持される期間は最大 30 日です。

Malware Protection for Backup での誤検出の報告

GuardDuty Malware Protection for Backup のエクスペリエンスを向上させるために、潜在的な誤検出と誤検出を報告する場合があります。

Malware Protection for Backup で特定された潜在的な誤検出または誤検出を報告するには

プロセスを開始するには、 にお問い合わせください サポート。スキャンされたリソースの詳細を指定するには、次の手順に従います。

1. にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. [Malware Scans] (マルウェアのスキャン) を選択します。
3. スキャンを選択してその [Finding ID] (検出結果 ID) を表示します。
4. [Finding ID] (検出結果 ID) を指定します。ファイルの SHA-256 ハッシュを指定する必要があります。これは、GuardDuty が正しいファイルを受信していることを確認するために必要です。また、サンプルを提供するリージョンも指定してください。
5. サポート チームは、悪意のある可能性のあるファイルと SHA-256 ハッシュをアップロードするために使用する Amazon Simple Storage Service (Amazon S3) の署名付き URL を提供します。スキャンされたリソースをアップロードする手順については、Amazon S3ユーザーガイド」の「署名付き URLs」を参照してください。
6. ファイルをアップロードしたら、 サポート チームに通知します。

   サポート は、ファイルを受信した後、確認応答を提供します。GuardDuty サービスチームのメンバーが送信を分析し、Malware Protection for EC2 の使用体験を向上させるための適切な手順を実行します。 サポート チームはケースのステータス更新を引き続き提供します。GuardDuty で S3 オブジェクトが保持される期間は最大 30 日です。

検出された GuardDuty セキュリティ検出結果の修復

Amazon GuardDuty は、GuardDuty の基本的な脅威検出と専用の保護プランに関連付けられたセキュリティを脅かす可能性がある事例を検出結果として生成します。次のセクションでは、これらのシナリオにおいて推奨される修復ステップについて説明します。代替修復シナリオがある場合は、その説明が検出結果タイプごとに提示されます。検出結果タイプに関する完全な情報にアクセスするには、[Active findings types table] (アクティブな検出結果タイプテーブル) を選択します。

内容

- 侵害された可能性のある Amazon EC2 インスタンスの修復
- 侵害された可能性のある S3 バケットの修復
- 悪意のある可能性がある S3 オブジェクトの修復
- 侵害された可能性のある EBS スナップショットの修復
- 侵害された可能性のある EC2 AMI の修復
- 侵害された可能性のある EC2 復旧ポイントの修復
- 侵害された可能性のある S3 復旧ポイントの修復
- 侵害された可能性のある ECS クラスターの修復
- 侵害された可能性のある AWS 認証情報の修正
- 侵害された可能性のあるスタンドアロンコンテナの修復
- EKS Protection の検出結果の修復
- Runtime Monitoring 検出結果の修正
- 侵害された可能性のあるデータベースの修復
- 侵害された可能性のある Lambda 関数の修復

侵害された可能性のある Amazon EC2 インスタンスの修復

Amazon EC2 リソースの侵害を示唆する検出結果タイプが生成された場合、[リソース] が [インスタンス] になります。生成されうる検出結果タイプには EC2 の検出結果タイプ、GuardDuty Runtime Monitoring の検出結果タイプ、Malware Protection for EC2 の検出結果のタイプ があります。検出結果の原因となった動作が環境の想定内である場合は、抑制ルールの使用を検討してください。

侵害された可能性のある Amazon EC2 インスタンスを修復するには、次の手順を実行します。

1. 侵害された可能性がある Amazon EC2 インスタンスを識別する

マルウェアに侵害された可能性のあるインスタンスを調査し、検出されたマルウェアを削除します。GuardDuty でのオンデマンドマルウェアスキャン を使用して、侵害された可能性のある EC2 インスタンス内のマルウェアを特定したり、AWS Marketplace でマルウェアを特定して削除するのに役立つパートナー製品があるかどうかを確認したりできます。

2. 侵害された可能性がある Amazon EC2 インスタンスを分離する

   可能であれば、次の手順を使用して侵害された可能性のあるインスタンスを分離します。

   1. 専用の分離セキュリティグループを作成します。分離セキュリティグループでは、特定の IP アドレスからのインバウンドアクセスとアウトバウンドアクセスのみを許可します。`0.0.0.0/0` (`0-65535`) に対するトラフィックを許可するインバウンドルールもアウトバウンドルールも存在しないことを確認してください。
   2. このインスタンスに分離セキュリティグループを関連付けます。
   3. 新規に作成した分離セキュリティグループを除くすべてのセキュリティグループの関連付けを侵害された可能性のあるインスタンスから削除します。

   > **Note**
   >
   > セキュリティグループが変更されても、既に追跡済みの接続が終了することはありません。今後発生するトラフィックのみが新しいセキュリティグループによって効果的にブロックされます。
   > 疑わしい既存の接続からのトラフィックを今後ブロックする方法については、「インシデント対応プレイブック」の「Enforce NACLs based on network IoCs to prevent further traffic」を参照してください。

3. 疑わしいアクティビティのソースを特定する

   マルウェアが検出された場合は、アカウント内の検出結果のタイプに基づいて、EC2 インスタンスでの不正なアクティビティの可能性を特定して停止します。これには、開いているポートを閉じる、アクセスポリシーを変更する、脆弱性を修正するためにアプリケーションをアップグレードするなどのアクションが必要になる場合があります。

   侵害された可能性のある EC2 インスタンスでの不正なアクティビティを識別して停止できない場合は、侵害された EC2 インスタンスを削除し、必要に応じて新しいインスタンスを作成することをお勧めします。EC2 インスタンスを保護するための追加リソースを次に示します。

   - 「Amazon EC2 のベストプラクティス」の「セキュリティ」と「ネットワーク」のセクション。

- Linux インスタンス用の Amazon EC2 セキュリティグループ。
- Amazon EC2 でのセキュリティ
- EC2 インスタンスのセキュリティを確保するためのヒント (Linux)
- AWS セキュリティのベストプラクティス
- AWS セキュリティインシデント対応テクニカルガイド。

4. 参照 AWS re:Post

   不明な点については、「AWS re:Post」を参照してください。

5. テクニカルサポートリクエストを送信する

   プレミアムサポートパッケージのサブスクライバーは、テクニカルサポートをリクエストできます。

侵害された可能性のある S3 バケットの修復

GuardDuty で GuardDuty S3 Protection の検出結果タイプが生成された場合は、Amazon S3 バケットが既に侵害されています。検出結果の原因となった動作が環境の想内である場合は、抑制ルールの作成を検討してください。この動作が予期されなかった場合は、以下の推奨手順に従って、 AWS 環境内の侵害された可能性のある Amazon S3 バケットを修復します。

1. 侵害された可能性のある S3 リソースを識別します。

   S3 の GuardDuty 検索では、関連付けられた S3 バケット、その Amazon リソースネーム (ARN)、およびその所有者が検出結果の詳細に表示されます。

2. 疑わしいアクティビティのソースと使用された API コールを特定します。

   使用された API コールは、検出結果の詳細に `API` として表示されます。ソースは IAM プリンシパル (IAM ロール、ユーザーまたはアカウント) で、識別情報が検出結果に表示されます。ソースタイプに応じて、リモート IP アドレスまたはソースドメイン情報が利用可能になり、ソースが承認されたかどうかを評価するのに役立ちます。Amazon EC2 インスタンスから関連する認証情報が検出結果にあれば、そのリソースの詳細も含まれます。

3. コールソースが、識別されたリソースへのアクセスを許可されたかどうかを確認します。

   例えば、次の事項を検討します。

- IAM ユーザーが関与していた場合は、認証情報が侵害された可能性があるかどうかを確認します。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。
- API が、このタイプの API を呼び出した履歴がないプリンシパルから呼び出された場合、このソースはこのオペレーションのアクセス権を必要としますか? バケットの許可をさらに制限することはできますか?
- ユーザー名 ANONYMOUS_PRINCIPAL とユーザータイプ AWSAccount のアクセスがあった場合、これは、バケットがパブリックであり、アクセスされたことを示します。このバケットはパブリックであるべきですか? そうでない場合は、S3 リソースを共有する代替ソリューションについて、以下のセキュリティに関するレコメンデーションを確認してください。
- アクセスがユーザー名 ANONYMOUS_PRINCIPAL と ユーザータイプ AWSAccountからの 成功した PreflightRequest コールを介して行われた場合、バケットにクロスオリジンリソース共有 (CORS) ポリシーが設定されていることを示します。このバケットには CORS ポリシーが必要ですか? そうでない場合は、バケットが不用意に公開されないようにし、S3 リソースを共有する代替ソリューションについて、以下のセキュリティに関するレコメンデーションを確認してください。CORS の詳細については、「S3 ユーザーガイド」の「Cross-Origin Resource Sharing (CORS) の使用」を参照してください。

4. S3 バケットに機密データが含まれているかどうか判断します。

   Amazon Macie を使用して、S3 バケットに個人を特定できる情報 (PII)、財務データ、認証情報などの機密データが含まれているかどうかを判断します。Macie アカウントで機密データの自動検出が有効になっている場合は、S3 バケットの詳細を確認して S3 バケットのコンテンツをよりよく理解してください。Macie アカウントでこの機能が無効になっている場合、評価を早めるために有効にすることをおすすめします。または、機密データ検出ジョブを作成して実行し、S3 バケットのオブジェクトに機密データがないかを検査することもできます。詳細については、「機密ログデータをマスキングで保護する」を参照してください。

アクセスが許可されている場合は、検出結果を無視できます。https://console.aws.amazon.com/guardduty/ コンソールでは、個々の結果を表示しないように完全に抑制するルールを設定できます。詳細については、「GuardDuty の抑制ルール」を参照してください。

S3 データが許可のない当事者によって公開またはアクセスされたと判断した場合は、次の S3 セキュリティに関するレコメンデーションを確認して、許可を強化し、アクセスを制限します。適切な修復ソリューションは、特定の環境のニーズによって異なります。

特定の S3 バケットへのアクセスニーズに基づく推奨事項

次に、特定の Amazon S3 バケットへのアクセスニーズに基づく推奨事項を示します。

- S3 データへのパブリックアクセスを一元的に制限するには、S3 ブロックパブリックアクセスを使用してください。ブロックパブリックアクセス設定は、アクセスポイント、バケット、 AWS アカウントに対して 4 つの異なる設定で有効にして、アクセスの詳細度を制御できます。詳細については、「Amazon S3 ユーザーガイド」の「パブリックアクセスブロック設定」を参照してください。
- AWS アクセスポリシーを使用して、IAM ユーザーがリソースにアクセスする方法やバケットにアクセスする方法を制御できます。詳細は、「Amazon S3 ユーザーガイド」の「バケットポリシーとユーザーポリシーの使用」を参照してください。

  S3 バケットポリシーで仮想プライベートクラウド (VPC) エンドポイントを使用して、特定の VPC エンドポイントへのアクセスを制限することもできます。詳細については、「Amazon S3 ユーザーガイド」の「バケットポリシーを使用した VPC エンドポイントからのアクセスコントロール」を参照してください
- アカウント外の信頼できるエンティティへの S3 オブジェクトへのアクセスを一時的に許可するには、S3 を使用して署名済み URL を作成します。このアクセスは、アカウントの認証情報を使用して作成され、使用される認証情報に応じて 6 時間から 7 日間使用できます。詳細については、「Amazon S3 ユーザーガイド」の「署名付き URL を使用したオブジェクトのダウンロードおよびアップロード」を参照してください。
- 異なるソース間で S3 オブジェクトを共有する必要があるユースケースでは、S3 アクセスポイントを使用して、プライベートネットワーク内のオブジェクトのみへのアクセスを制限する許可セットを作成できます。詳細については、「Amazon S3 ユーザーガイド」の「アクセスポイントを使用した共有データセットへのアクセス管理」を参照してください。
- 他のアカウントに S3 リソースへのアクセスを安全に付与 AWS するには、Amazon S3ユーザーガイド」の「アクセスコントロールリスト (ACL) の概要」を参照してください。

S3 のセキュリティオプションの詳細については、「Amazon S3 ユーザーガイド」の「Amazon S3 のセキュリティのベストプラクティス」を参照してください。

悪意のある可能性がある S3 オブジェクトの修復

GuardDuty で Malware Protection for S3 の検出結果タイプが生成された場合は、Amazon S3 バケットに新規にアップロードされたオブジェクトにマルウェアが含まれています。リソースタイプは S3Object です。

次の推奨手順を使用すると、生成された検出結果を修復できる可能性があります。

1. 検出結果に関連付けられている S3ObjectDetails をチェックして、悪意のある可能性がある S3 オブジェクトを識別します。
2. こうして識別した S3 オブジェクトを分離します。関連付けられた Amazon S3 バケットに対して Malware Protection for S3 を有効にした時点で既にタグ付けを有効にしていた場合は、このオブジェクトに悪意のあるタグが割り当てられているはずです。タグベースのアクセスコントロール (TBAC) を使用して、この S3 オブジェクトへのアクセスを制限します。詳細については、「タグベースのアクセスコントロール (TBAC) の使用」を参照してください。

   あるいは、今後このオブジェクトを必要としない場合は、削除することも分離された S3 バケットに移動することもできます。S3 オブジェクトを削除する際の考慮事項については、「Amazon S3 ユーザーガイド」の「オブジェクトの削除」を参照してください。

侵害された可能性のある EBS スナップショットの修復

GuardDuty が Execution:EC2/MaliciousFile を生成する場合スナップショット検出結果タイプは、Amazon EBS スナップショットでマルウェアが検出されたことを示します。侵害された可能性のあるスナップショットを修正するには、次の手順を実行します。

1. 侵害された可能性のあるスナップショットを特定する

   1. 侵害された可能性のあるスナップショットを特定します。EBS スナップショットの GuardDuty 検出結果は、影響を受けるスナップショット ID、その Amazon リソースネーム (ARN)、および関連するマルウェアスキャンの詳細を検出結果の詳細に一覧表示します。
   2. 次のコマンドを使用して、復旧ポイントの詳細を確認します。

      ```
      aws backup describe-recovery-point –backup-vault-name 021345abcdef6789 –recovery-point-arn "arn:aws:ec2:us-east-1::snapshot/snap-abcdef01234567890"
      ```

2. 侵害されたスナップショットへのアクセスを制限する

バックアップボールトアクセスポリシーを確認して変更し、リカバリポイントのアクセスを制限し、このスナップショットを使用する可能性のある自動復元ジョブを停止します。

1. 現在の共有アクセス許可を確認します。

```
aws ec2 describe-snapshot-attribute --snapshot-id snap-abcdef01234567890 --
attribute createVolumePermission
```

2. 特定のアカウントアクセスを削除します。

```
aws ec2 modify-snapshot-attribute --snapshot-id snap-abcdef01234567890 --
attribute createVolumePermission --operation-type remove --user-ids 111122223333
```

3. その他の CLI オプションについては、modify-snapshot-attribute CLI documentation」を参照してください。

3. 修復アクションを実行する

- 削除に進む前に、すべての依存関係を特定し、必要に応じて適切なバックアップがあることを確認してください。

侵害された可能性のある EC2 AMI の修復

GuardDuty が Execution:EC2/MaliciousFile を生成する場合AMI 検出結果タイプは、Amazon マシンイメージ (AMI) でマルウェアが検出されたことを示します。侵害された可能性のある AMI を修復するには、次の手順を実行します。

1. 侵害された可能性のある AMI を特定する

1. AMIs の GuardDuty 検出結果は、影響を受ける AMI ID、その Amazon リソースネーム (ARN)、および関連するマルウェアスキャンの詳細を検出結果の詳細に一覧表示します。
2. AMI ソースイメージを確認します。

```
aws ec2 describe-images --image-ids ami-021345abcdef6789
```

2. 侵害されたリソースへのアクセスを制限する

1. バックアップボールトアクセスポリシーを確認して変更し、復旧ポイントへのアクセスを制限し、この復旧ポイントを使用する可能性のある自動復元ジョブを停止します。

2. ソース AMI アクセス許可からアクセス許可を削除する

   最初に既存のアクセス許可を表示します。

   ```
   aws ec2 describe-image-attribute --image-id ami-abcdef01234567890 --attribute
    launchPermission
   ```

   次に、個々のアクセス許可を削除します。

   ```
   aws ec2 modify-image-attribute --image-id ami-abcdef01234567890 --launch-
   permission '{"Remove":[{"UserId":"111122223333"}]}'
   ```

   その他の CLI オプションについては、「特定のアカウントと AMI を共有する - Amazon Elastic Compute Cloud」を参照してください。

3. ソースが EC2 インスタンスの場合: 侵害された可能性のある Amazon EC2 インスタンスの修正を参照してください。

3. 修復アクションを実行する

   - 削除に進む前に、すべての依存関係を特定し、必要に応じて適切なバックアップがあることを確認してください。

侵害された可能性のある EC2 復旧ポイントの修復

GuardDuty が Execution:EC2/MaliciousFile を生成する場合RecoveryPoint の検出結果タイプは、EC2 Recovery Point Backup リソースでマルウェアが検出されたことを示します。侵害された可能性のある復旧ポイントを修復するには、次の手順を実行します。

1. 侵害された可能性のある EC2 復旧ポイントを特定する

   1. EC2 Recovery Point の GuardDuty の検出結果は、Amazon リソースネーム (ARN) と関連するマルウェアスキャンの詳細を検出結果の詳細に一覧表示します。

      ```
      aws backup describe-recovery-point --backup-vault-name 021345abcdef6789
       --recovery-point-arn "arn:aws:backup:us-east-1:111122223333:recovery-
      point:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
      ```

   2. 復旧の詳細を確認して、ソースイメージを探します。

```
aws backup get-recovery-point-restore-metadata --backup-vault-
name 021345abcdef6789 --recovery-point-arn "arn:aws:backup:us-
east-1:111122223333:recovery-point:a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
```

2. 侵害されたリソースへのアクセスを制限する

- バックアップボールトアクセスポリシーを確認して変更し、復旧ポイントへのアクセスを制限し、この復旧ポイントを使用する可能性のある自動復元ジョブを停止します。環境がリソースのタグ付けを使用している場合は、復旧ポイントに適切にタグ付けして調査中であることを示すとともに、必要に応じてスケジュールされたバックアップの一時停止を検討してください。

  例:

  ```
  aws backup tag-resource --resource-arn arn:aws:backup:us-
  east-1:111122223333:recovery-point:a1b2c3d4-5678-90ab-cdef-
  EXAMPLE11111 --tags Investigation=Malware,DoNotDelete=True
  ```

3. 修復アクションを実行する

- 削除に進む前に、すべての依存関係を特定し、必要に応じて適切なバックアップがあることを確認してください。

侵害された可能性のある S3 復旧ポイントの修復

GuardDuty が Execution:S3/MaliciousFile を生成する場合RecoveryPoint の検出結果タイプは、S3 Recovery Point Backup リソースでマルウェアが検出されたことを示します。侵害された可能性のある復旧ポイントを修復するには、次の手順を実行します。

1. 侵害された可能性のある S3 復旧ポイントを特定する

   1. S3 リカバリポイントの GuardDuty 検出結果には、影響を受けるリカバリポイント ARN、バックアップボールト名、および関連するマルウェアスキャンの詳細が検出結果の詳細に一覧表示されます。
   2. 復旧ポイントの詳細を確認します。

      ```
      aws backup describe-recovery-point --backup-vault-name 021345abcdef6789
       --recovery-point-arn arn:aws:backup:us-east-1:123456789012:recovery-
      point:abcdef01234567890
      ```

2. 侵害されたリソースへのアクセスを制限する

   - バックアップボールトアクセスポリシーを確認して変更し、復旧ポイントへのアクセスを制限し、この復旧ポイントを使用する可能性のある自動復元ジョブを停止します。環境がリソースのタグ付けを使用している場合は、復旧ポイントに適切にタグ付けして調査中であることを示すとともに、必要に応じてスケジュールされたバックアップの一時停止を検討してください。

     例:

     ```
     aws backup tag-resource –resource-arn arn:aws:backup:us-
     east-1:111122223333:recovery-point:abcdef01234567890 –
     tags Investigation=Malware,DoNotDelete=True
     ```

     詳細については、「tag-resource CLI コマンドリファレンス」を参照してください。

3. 修復アクションを実行する

   - 削除に進む前に、すべての依存関係を特定し、必要に応じて適切なバックアップがあることを確認してください。

侵害された可能性のある ECS クラスターの修復

侵害された可能性のある ECS クラスターの検出結果は、Amazon ECS 環境内で疑わしいアクティビティまたは悪意のあるアクティビティが検出されたことを示します。これには、コンテナワークロードを危険にさらす不正アクセス、マルウェアの実行、またはその他の悪意のある動作が含まれます。

侵害された可能性のある Amazon ECS クラスターを修復するには、次の手順に従います。

1. 侵害された可能性のある ECS クラスターと検出された脅威を特定する (検出結果)

   影響を受ける ECS クラスターの詳細は、GuardDuty の検出結果の詳細パネルに表示されます。

2. 脅威/マルウェアの原因を評価する

   コンテナイメージにマルウェアがないか確認します。マルウェアが検出された場合は、使用されているコンテナイメージを確認します。を使用してListTasks、侵害された可能性のある同じイメージを使用する他のすべての実行中のタスクを識別します。

3. 影響を受けるタスクを分離する

影響を受けるタスクへのすべてのネットワークトラフィック (着信と発信の両方) をブロックして、脅威を停止します。このネットワーク分離は、侵害されたタスクへのすべての接続を切断することで、進行中の攻撃を防ぐのに役立ちます。

注: この検出結果が環境内の予想/正当なアクティビティによってトリガーされたと判断した場合は、同様の検出結果が表示されないように抑制ルールを設定できます。詳細については、「GuardDuty の抑制ルール」を参照してください。

侵害された可能性のある AWS 認証情報の修正

GuardDuty が を生成するとIAM の検出結果タイプ、 AWS 認証情報が侵害されたことを示します。侵害された可能性のある [リソース] のタイプは [AccessKey] です。

AWS 環境内の侵害された可能性のある認証情報を修正するには、次の手順を実行します。

1. 侵害された可能性のある IAM エンティティと使用された API コールを識別します。

   使用された API コールは、検出結果の詳細に API として表示されます。IAM エンティティ (IAM ロールまたはユーザー) とその識別情報は、検出結果の詳細の [リソース] セクションに表示されます。関連する IAM エンティティのタイプは、[User Type] (ユーザータイプ) フィールドで特定できます。IAM エンティティの名前は、[User name] (ユーザー名) フィールドに表示されます。検出結果に関連する IAM エンティティのタイプは、使用された [Access key ID] (アクセスキー ID) でも特定できます。

   AKIA で始まるキーの場合:

   このタイプのキーは、IAM ユーザーまたは AWS アカウントのルートユーザーに関連付けられているカスタマーマネージドの長期の認証情報です。IAM ユーザーのアクセスキーの管理については、「IAM ユーザーのアクセスキーの管理」を参照してください。

   ASIA で始まるキーの場合:

   このタイプのキーは、 AWS Security Token Serviceによって生成される短期の一時的な認証情報です。これらのキーは短時間のみ存在し、 AWS マネジメントコンソールで表示または管理することはできません。IAM ロールは常に AWS STS 認証情報を使用しますが、IAM ユーザー用に生成することもできます。詳細については、「IAM: 一時的なセキュリティ認証情報 AWS STS 」を参照してください。

ロールが使用された場合、[User name] (ユーザー名) フィールドには使用されたロールの名前が表示されます。CloudTrail ログエントリの `sessionIssuer`要素を調べる AWS CloudTrail ことで、 でキーがどのようにリクエストされたかを確認できます。詳細については、「IAM」およびCloudTrail AWS STS の情報」を参照してください。

2. IAM エンティティの許可を確認します。

   [IAM コンソール] を開きます。使用されたエンティティのタイプに応じて [ユーザー] タブまたは [ロール] タブを選択し、検索フィールドに識別した名前を入力して影響を受けるエンティティを見つけます。[Permission] (許可) タブと [Access Advisor] (アクセスアドバイザー) タブを使用して、そのエンティティの有効な許可を確認します。

3. IAM エンティティの認証情報が正当に使用されたかどうかを確認します。

   アクティビティが意図的なものであったかどうかを確認するには、認証情報のユーザーに問い合わせます。

   例えば、ユーザーが次のことを行ったかどうかを確認します。

   - GuardDuty の検出結果に表示された API オペレーションの呼び出し
   - GuardDuty の検出結果に表示された時刻における API オペレーションの呼び出し
   - GuardDuty の検出結果に表示された IP アドレスからの API オペレーションの呼び出し

このアクティビティが AWS 認証情報の正当な使用である場合は、GuardDuty の検出結果を無視できます。https://console.aws.amazon.com/guardduty/ コンソールでは、個々の結果を表示しないように完全に抑制するルールを設定できます。詳細については、「GuardDuty の抑制ルール」を参照してください。

正当に使用されたことが確認できない場合、このアクティビティはその特定のアクセスキー、IAM ユーザーのサインイン認証情報、または AWS アカウント全体に対する侵害の結果である可能性があります。認証情報が侵害された疑いがある場合は、「My AWS アカウント may be compromised to remediate this issue」の情報を確認してください。

侵害された可能性のあるスタンドアロンコンテナの修復

コンテナの侵害を示唆する検出結果タイプが生成された場合、[リソース] のタイプが [コンテナ] になります。検出結果の原因となった動作が環境の想定内である場合は、抑制ルールの使用を検討してください。

AWS 環境内の侵害された可能性のある認証情報を修正するには、次の手順を実行します。

1. 侵害された可能性があるコンテナを分離する

   次の手順に従うと、悪意のある可能性があるコンテナワークロードを識別できます。

   - https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
   - [検出結果] ページで、対応する検出結果を選択して検出結果パネルを表示します。
   - 検出結果パネルの [Resource affected] (影響を受けるリソース) セクションで、コンテナの [ID] と [Name] (名前) を表示できます。

   このコンテナを他のコンテナワークロードから分離します。

2. コンテナを一時停止する

   コンテナ内のすべてのプロセスを一時停止します。

   コンテナをフリーズする方法については、「Pause a container」を参照してください。

   コンテナを停止する。

   上記のステップが失敗し、コンテナが一時停止しない場合は、コンテナの実行を停止します。スナップショットの保持 機能を有効にした場合、GuardDuty はマルウェアを含む EBS ボリュームのスナップショットを保持します。

   コンテナを停止する方法については、「Stop a container」を参照してください。

3. マルウェアの有無の評価

   マルウェアがコンテナのイメージに存在したかどうかを評価します。

アクセスが許可されている場合は、検出結果を無視できます。https://console.aws.amazon.com/guardduty/ コンソールでは、個々の結果を表示しないように完全に抑制するルールを設定できます。GuardDuty コンソールでは、個々の結果が表示されないように完全に抑制するルールを設定できます。詳細については、「GuardDuty の抑制ルール」を参照してください。

EKS Protection の検出結果の修復

Amazon GuardDuty は、アカウント用に EKS Protection が有効になっている場合に、Kubernetes のセキュリティ問題の可能性を示す検出結果を生成します。詳細については、「EKS Protection」を参照してください。次のセクションでは、これらのシナリオにおいて推奨される修復ステップについて説明します。特定の修復アクションは、その特定の検出結果タイプのエントリで説明されています。

検出結果タイプに関する完全な情報にアクセスするには、[Active findings types table] (アクティブな検出結果タイプテーブル) を選択します。

EKS Protection の検出結果タイプのいずれかが予想どおりに生成された場合は、今後アラートが発生しないように GuardDuty の抑制ルールを追加することを検討してください。

さまざまなタイプの攻撃と設定の問題によって、GuardDuty EKS Protection の検出結果がトリガーされる可能性があります。このガイドは、適切な修復ガイダンスを概説するものであり、クラスターに対する GuardDuty の検出結果の根本原因を特定するのに役立ちます。GuardDuty の Kubernetes の検出結果につながる主な根本原因は次のとおりです。

- 潜在的な設定の問題
- 侵害された可能性のある Kubernetes ユーザーの修復
- 侵害された可能性のある Kubernetes ポッドの修復
- 侵害された可能性のある Kubernetes ノードの修復
- 侵害された可能性のあるコンテナイメージの修復

> **Note**
>
> Kubernetes バージョン 1.14 より前では、`system:unauthenticated` グループは、デフォルトで `system:discovery` と `system:basic-user` ClusterRoles に関連付けられていました。これは、匿名ユーザーからの意図しないアクセスを許可する場合があります。クラスターの更新では、これらの許可は取り消されません。つまり、クラスターをバージョン 1.14 以降に更新した場合でも、これらの許可は引き続き有効である可能性があります。これらの許可の関連付けを `system:unauthenticated` グループから解除することをお勧めします。
> これらのアクセス許可の削除の詳細については、「Amazon EKS ユーザーガイド」の「ベストプラクティスによる Amazon EKS クラスターの保護」を参照してください。

潜在的な設定の問題

検出結果で設定の問題が示されている場合、その特定の問題の解決に関するガイダンスについては、その検出結果の修復セクションを参照してください。詳細については、設定上の問題を示す次の検出結果タイプを参照してください。

- Policy:Kubernetes/AnonymousAccessGranted

- Policy:Kubernetes/ExposedDashboard
- Policy:Kubernetes/AdminAccessToDefaultServiceAccount
- Policy:Kubernetes/KubeflowDashboardExposed
- SuccessfulAnonymousAccess で終わる検出結果。

侵害された可能性のある Kubernetes ユーザーの修復

GuardDuty の検出結果は、検出結果で特定されたユーザーが想定されていない API アクションを実行したときに、侵害された Kubernetes ユーザーを示している場合があります。ユーザーは、コンソールの検出結果の詳細の [Kubernetes user details] (Kubernetes ユーザー詳細) セクション、または検出結果の JSON の `resource.kubernetesDetails.kubernetesUserDetails` で識別できます。これらのユーザーの詳細には、`user name`、`uid`、およびユーザーが属する Kubernetes グループが含まれます。

ユーザーが IAM エンティティを使用してワークロードにアクセスしていた場合は、この `Access Key details` セクションを使用して、IAM ロールまたはユーザーの詳細を識別できます。次のユーザータイプとその修復ガイダンスを参照してください。

> **Note**
>
> Amazon Detective を使用して、検出結果で特定された IAM ロールまたはユーザーをさらに調査できます。GuardDuty コンソールで検出結果の詳細を表示しているときに、[Detective で調査] を選択します。次に、リストされた項目から AWS ユーザーまたはロールを選択して、Detective で調査します。

組み込みの Kubernetes 管理者 – クラスターを作成した IAM アイデンティティに Amazon EKS によって割り当てられたデフォルトのユーザー。このユーザータイプは、ユーザー名 `kubernetes-admin` で識別されます。

組み込みの Kubernetes 管理者のアクセスを取り消すには:

- `Access Key details` セクションから `userType` を識別します。
  - `userType` が [Role] (ロール) であり、ロールが EC2 インスタンスロールに属している場合:
    - そのインスタンスを特定し、「侵害された可能性のある Amazon EC2 インスタンスの修復」の手順に従います。

- userType が [User] (ユーザー) の場合、またはユーザーが引き受けた [Role] (ロール) の場合:
  1. そのユーザーのアクセスキーをローテーションします。
  2. ユーザーがアクセス権を有していたシークレットをローテーションします。
  3. 詳細については、「My AWS アカウント may be compromised」の情報を確認してください。

OIDC 認証済みユーザー – OIDC プロバイダーを通じてアクセス権を付与されたユーザー。通常、OIDC ユーザーは、ユーザー名としてメールアドレスを持っています。次のコマンドを使用して、クラスターが OIDC を使用しているかどうかを確認できます: `aws eks list-identity-provider-configs --cluster-name your-cluster-name` 。

OIDC 認証済みユーザーのアクセスを取り消すには:

1. OIDC プロバイダーでそのユーザーの認証情報をローテーションします。
2. ユーザーがアクセス権を有していたシークレットをローテーションします。

AWS-Auth ConfigMap 定義ユーザー – AWS認証 ConfigMap を通じてアクセス権が付与された IAM ユーザー。詳細については、「Amazon EKS ユーザーガイド」の「Managing users or IAM roles for your cluster」を参照してください。次のコマンドを使用して、許可を確認できます: `kubectl edit configmaps aws-auth --namespace kube-system`。

AWS ConfigMap ユーザーのアクセスを取り消すには:

1. 次のコマンドを使用して、ConfigMap を開きます。

```
kubectl edit configmaps aws-auth --namespace kube-system
```

2. GuardDuty の検出結果の [Kubernetes user details] (Kubernetes ユーザー詳細) セクションで報告されたものと同じユーザー名で、mapRoles または mapUsers セクションの下のロールまたはユーザーエントリを特定します。次の例を参照してください。ここでは、管理者ユーザーが検出結果で特定されています。

```
apiVersion: v1
data:
  mapRoles: |
    - rolearn: arn:aws:iam::444455556666:role/eksctl-my-cluster-nodegroup-
standard-wo-NodeInstanceRole-1WP3NUE3O6UCF
      user name: system:node:EC2_PrivateDNSName
      groups:
```

```
        - system:bootstrappers
        - system:nodes
  mapUsers: |
    - userarn: arn:aws:iam::123456789012:user/admin
      username: admin
      groups:
        - system:masters
    - userarn: arn:aws:iam::111122223333:user/ops-user
      username: ops-user
      groups:
        - system:masters
```

3. そのユーザーを ConfigMap から削除します。管理者ユーザーが削除された次の例を参照してください。

```
apiVersion: v1
data:
  mapRoles: |
    - rolearn: arn:aws:iam::111122223333:role/eksctl-my-cluster-nodegroup-
standard-wo-NodeInstanceRole-1WP3NUE3O6UCF
      username: system:node:{{EC2PrivateDNSName}}
      groups:
        - system:bootstrappers
        - system:nodes
  mapUsers: |
    - userarn: arn:aws:iam::111122223333:user/ops-user
      username: ops-user
      groups:
        - system:masters
```

4. userType が [User] (ユーザー) の場合、またはユーザーが引き受けた [Role] (ロール) の場合:
   a. そのユーザーのアクセスキーをローテーションします。
   b. ユーザーがアクセス権を有していたシークレットをローテーションします。
   c. 詳細については、AWS 「アカウントが侵害されている可能性がある」の情報を確認してください。

検出結果に resource.accessKeyDetails セクションがない場合、ユーザーは Kubernetes サービスアカウントです。

サービスアカウント – サービスアカウントはポッドのアイデンティティを提供し、次の形式のユーザー名で識別できます: system:serviceaccount:*namespace*:*service_account_name*。

サービスアカウントへのアクセス権を取り消すには:

1. サービスアカウントの認証情報をローテーションします。
2. 次のセクションでポッドの侵害に関するガイダンスを確認します。

侵害された可能性のある Kubernetes ポッドの修復

GuardDuty が resource.kubernetesDetails.kubernetesWorkloadDetails セクション内のポッドまたはワークロードリソースの詳細を指定する場合、そのポッドまたはワークロードリソースが侵害されている可能性があります。GuardDuty の検出結果は、単一のポッドが侵害されたか、または上位レベルのリソースを介して複数のポッドが侵害されたことを示している場合があります。侵害されたポッドを特定する方法のガイダンスについては、次の侵害シナリオを参照してください。

単一のポッドの侵害

resource.kubernetesDetails.kubernetesWorkloadDetails セクション内の type フィールドが [Pod] (ポッド) である場合、検出結果は単一のポッドを特定します。名前フィールドはポッドの name であり、namespace フィールドはその名前空間です。

ポッドを実行しているワーカーノードの特定の詳細については、「Amazon EKS Best Practices Guide」の「Identify the offending pods and worker node」を参照してください。

ワークロードリソースを通じて侵害されたポッド

type セクション内の resource.kubernetesDetails.kubernetesWorkloadDetails フィールドが、Deployment などの [Workload Resource] (ワークロードリソース) を識別している場合、そのワークロードリソース内のすべてのポッドが侵害されている可能性があります。

ワークロードリソースのすべてのポッドとそれらが実行されているノードの特定の詳細については、「Amazon EKS Best Practices Guide」の「Identify the offending pods and worker nodes using workload name」を参照してください。

サービスアカウントを通じて侵害されたポッド

GuardDuty の検出結果が resource.kubernetesDetails.kubernetesUserDetails セクションでサービスアカウントを識別した場合、識別されたサービスアカウントを使用しているポッドが侵害されている可能性があります。検出結果によって報告されたユーザー名は、次の形式の場合はサービスアカウントです: system:serviceaccount:*namespace*:*service_account_name*。

サービスアカウントを使用しているすべてのポッドとそれらが実行されているノードの特定の詳細については、「Amazon EKS Best Practices Guide」の「Identify the offending pods and worker nodes using service account name」を参照してください。

侵害されたポッドとそれらが実行されているノードをすべて特定したら、「Amazon EKS Best Practices Guide」の「Isolate the pod by creating a network policy that denies all ingress and egress traffic to the pod」を参照してください。

侵害されている可能性のあるポッドを修復するには:

1. ポッドを侵害した脆弱性を特定します。
2. その脆弱性の修復を実装し、新しい代替ポッドを起動します。
3. 脆弱なポッドを削除します。

詳細については、「Amazon EKS Best Practices Guide」の「Redeploy compromised pod or workload resource」を参照してください。

ワーカーノードに Pod が他の AWS リソースにアクセスできるようにする IAM ロールが割り当てられている場合は、攻撃によるさらなる損害を防ぐために、インスタンスからそれらのロールを削除します。同様に、ポッドに IAM ロールが割り当てられている場合は、他のワークロードに影響を与えることなく、ロールから IAM ポリシーを安全に削除できるかどうかを評価します。

侵害された可能性のあるコンテナイメージの修復

GuardDuty の検出結果でポッドの侵害が示されている場合、ポッドの起動に使用されたイメージが潜在的に悪意があるものであるか、または侵害されている可能性があります。GuardDuty の検出結果では、`resource.kubernetesDetails.kubernetesWorkloadDetails.containers.image` フィールド内のコンテナイメージが識別されます。マルウェアがないかを調べるためにスキャンして、イメージが悪意のあるものであるかどうかを判断できます。

侵害されたコンテナイメージを修復するには:

1. 直ちにイメージの使用を停止し、イメージリポジトリから削除します。
2. 侵害された可能性のあるイメージを使用しているすべてのポッドを特定します。

詳細については、「Amazon EKS Best Practices Guide」の「Identify pods with vulnerable or compromised images and worker nodes」 を参照してください。

3. 侵害された可能性のあるポッドを分離し、認証情報をローテーションして、分析のためにデータを収集します。詳細については、「Amazon EKS Best Practices Guide」の「Isolate the pod by creating a network policy that denies all ingress and egress traffic to the pod」を参照してください。
4. 侵害された可能性のあるイメージを使用しているすべてのポッドを削除します。

侵害された可能性のある Kubernetes ノードの修復

GuardDuty の検出結果は、検出結果で識別されたユーザーがノードアイデンティティを表す場合、または検出結果が特権コンテナの使用を示している場合、ノードの侵害を示している場合があります。

[username] (ユーザーネーム) フィールドの形式が次の場合、ユーザーアイデンティティはワーカーノードです: `system:node:node name`。例えば、`system:node:ip-192-168-3-201.ec2.internal`。これは、攻撃者がノードへのアクセスを取得し、ノードの認証情報を使用して Kubernetes API エンドポイントと通信していることを示すものです。

検出結果では、検出結果にリストされている 1 つ以上のコンテナの `resource.kubernetesDetails.kubernetesWorkloadDetails.containers.securityContext.` 検出結果フィールドが `True` に設定されている場合に、特権コンテナの使用を示します。

侵害されている可能性のあるノードを修復するには:

1. ポッドを分離し、認証情報をローテーションして、フォレンジック分析のためにデータを収集します。

   詳細については、「Amazon EKS Best Practices Guide」の「Isolate the pod by creating a network policy that denies all ingress and egress traffic to the pod」を参照してください。
2. 侵害されている可能性のあるノードで実行されているすべてのポッドが使用するサービスアカウントを特定します。許可を確認し、必要に応じてサービスアカウントをローテーションします。
3. 侵害されている可能性のあるノードを終了します。

Runtime Monitoring 検出結果の修正

アカウントの Runtime Monitoring を有効にすると、Amazon GuardDuty GuardDuty Runtime Monitoring の検出結果タイプが AWS 環境内の潜在的なセキュリティ問題を示す を生成することが

あります。潜在的なセキュリティ問題は、侵害された Amazon EC2 インスタンス、コンテナワークロード、Amazon EKS クラスター、または AWS 環境内の侵害された認証情報のセットのいずれかを示します。セキュリティエージェントは、複数のリソースタイプからのランタイムイベントを監視します。危険にさらされている可能性のあるリソースを特定するには、GuardDuty コンソールで生成された検出結果の詳細で [リソースタイプ] を確認してください。次のセクションでは、リソースのタイプごとに推奨される修復手順について説明します。

Instance

検出結果の [リソースタイプ] が [インスタンス] の場合は、EC2 インスタンスまたは EKS ノードのいずれかが侵害されている可能性があることを示しています。

- 侵害された EKS ノードを修復するには、「侵害された可能性のある Kubernetes ノードの修復」を参照してください。
- 侵害された EC2 インスタンスを修正するには、「侵害された可能性のある Amazon EC2 インスタンスの修復」を参照してください。

EKSCluster

検出結果の詳細の [リソースタイプ] が [EKSCluster] の場合は、EKS クラスター内のポッドまたはコンテナが危険にさらされている可能性があることを示しています。

- 侵害されたポッドを修復するには、「侵害された可能性のある Kubernetes ポッドの修復」を参照ください。
- 侵害されたコンテナイメージを修復するには、「侵害された可能性のあるコンテナイメージの修復」を参照してください。

ECSCluster

検出結果の詳細の [リソースタイプ] が [ECSCluster] の場合は、ECS タスク内の ECS タスクまたはコンテナが危険にさらされている可能性があることを示しています。

1. 影響を受ける ECS クラスターを特定します

   GuardDuty Runtime Monitoring の検出結果では、検出結果の詳細パネルまたは検出結果 JSON の `resource.ecsClusterDetails` セクションに ECS クラスターの詳細が表示されます。

2. 影響を受ける ECS タスクを特定

GuardDuty Runtime Monitoring の検出結果では、検出結果の詳細パネルまたは検出結果 JSON の `resource.ecsClusterDetails.taskDetails` セクションに ECS タスクの詳細が表示されます。

3. 影響を受けるタスクを分離

   タスクへの送受信トラフィックをすべて拒否して、影響を受けたタスクを分離します。すべてのトラフィックを拒否するルールは、タスクへのすべての接続を切断することによって、すでに進行中の攻撃を阻止するのに役立ちます。

4. 侵害されたタスクを修復
   a. タスクを侵害した脆弱性を特定します。
   b. その脆弱性の修正を実装し、新しい置き換えタスクを開始します。
   c. 脆弱なタスクを停止します。

Container

検出結果の詳細の [リソースタイプ] が [コンテナ] の場合は、スタンドアロンコンテナが危険にさらされている可能性があることを示しています。

- 修正するには、「侵害された可能性のあるスタンドアロンコンテナの修復」を参照してください。
- 同じコンテナイメージを使用して複数のコンテナにわたって検出結果が生成される場合は、「侵害された可能性のあるコンテナイメージの修復」を参照してください。
- コンテナが基盤となる EC2 ホストにアクセスした場合、関連するインスタンスの認証情報が侵害されている可能性があります。詳細については、「侵害された可能性のある AWS 認証情報の修正」を参照してください。
- 潜在的に悪意のある攻撃者が基盤となる EKS ノードまたは EC2 インスタンスにアクセスした場合は、[EKSCluster] タブと [インスタンス] タブで推奨される修復方法を参照してください。

侵害されたコンテナイメージの修復

GuardDuty の検出結果でタスクの侵害が示されている場合、タスクの起動に使用されたイメージが悪意があるものであるか、または侵害されている可能性があります。GuardDuty の検出結果では、`resource.ecsClusterDetails.taskDetails.containers.image` フィールド内のコンテナイメージが識別されます。マルウェアの有無を調べるためにスキャンして、イメージが悪意のあるものであるかどうかを判断できます。

侵害されたコンテナイメージを修復

1. 直ちにイメージの使用を停止し、イメージリポジトリから削除します。
2. このイメージを使用しているタスクをすべて特定します。
3. 侵害されたイメージを使用しているすべてのタスクを停止します。侵害されたイメージの使用を停止するよう、タスクの定義を更新します。

侵害された可能性のあるデータベースの修復

RDS Protection を有効にすると、GuardDuty によって RDS Protection の検出結果タイプ が生成され、サポートされているデータベース での異常かつ疑わしいログイン動作が示されます。GuardDuty は RDS ログインアクティビティを使用して、ログイン試行の異常なパターンを特定することで脅威の分析とプロファイリングを行います。

> **Note**
>
> 検出結果タイプに関する完全な情報にアクセスするには、GuardDuty のアクティブな検出結果タイプ から選択します。

AWS 環境内の侵害された可能性のある Amazon Aurora データベースを修正するには、以下の推奨ステップに従います。

トピック

- ログインイベントの成功により、侵害された可能性のあるデータベースの修復
- ログインイベントの失敗により、侵害された可能性のあるデータベースの修正
- 異常な流出アクティビティによる侵害された可能性のあるデータベースの修復
- 異常な破壊的アクティビティによる侵害された可能性のあるデータベースの修復
- 漏えいした可能性のある認証情報の修正
- ネットワークアクセスを制限する

ログインイベントの成功により、侵害された可能性のあるデータベースの修復

以下の推奨手順は、ログインイベントの成功に関連して異常な動作をする、侵害された可能性のある Aurora データベースを修復する際に役に立ちます。

1. 影響を受けたデータベースとユーザーを特定します。

   生成された GuardDuty の検出結果には、影響を受けたデータベースの名前と対応するユーザーの詳細が表示されます。詳細については、「検出結果の詳細」を参照してください。
2. この動作が予期されるものであるか、または予期されないものであるかを確認します。

   以下のリストは、GuardDuty から検出結果が生成されるシナリオを示しています。
   - 長い時間が経過した後にデータベースにログインするユーザー。
   - データベースにときどきログインするユーザー (四半期ごとにログインする財務アナリストなど)。
   - ログイン試行に成功し、データベースを侵害した可能性がある疑わしい攻撃者。
3. 予期しない動作が発生した場合は、このステップを開始してください。

   1. データベースアクセスを制限する

      疑わしいアカウントおよびこのログインアクティビティのソースによるデータベースへのアクセスを制限します。詳細については、「漏えいした可能性のある認証情報の修正」および「ネットワークアクセスを制限する」を参照してください。
   2. 影響を評価し、アクセスされた情報を特定します。
      - 可能であれば、監査ログを確認して、アクセスされた可能性のある情報を特定します。詳細については、「Amazon Aurora ユーザーガイド」の「Amazon Aurora DB クラスターでの、イベント、ログ、およびストリーミングのモニタリング」を参照してください。
      - 機密情報または保護された情報にアクセスされたか、または変更が加えられたかどうかを確認します。

ログインイベントの失敗により、侵害された可能性のあるデータベースの修正

以下の推奨手順は、ログインイベントの失敗に関連して異常な動作をする、侵害された可能性がある Aurora データベースを修復する際に役に立ちます。

1. 影響を受けたデータベースとユーザーを特定します。

   生成された GuardDuty の検出結果には、影響を受けたデータベースの名前と対応するユーザーの詳細が表示されます。詳細については、「検出結果の詳細」を参照してください。

2. ログインに失敗したソースを特定します。

   GuardDuty の検出結果パネルの [Actor] (アクター) セクションに、[IP address] (IP アドレス) と [ASN organization] (ASN 組織) (パブリック接続の場合) が表示されます。

   自律システム (AS) は、1 つ以上のネットワークオペレーターによって実行される 1 つ以上の IP プレフィックス (ネットワーク上でアクセス可能な IP アドレスのリスト) のグループで、明確に定義された単一のルーティングポリシーを維持します。ネットワークオペレーターには、ネットワーク内のルーティングを制御したり、他のインターネットサービスプロバイダー (ISP) とルーティング情報を交換したりするための AS 番号 (ASN) が必要です。

3. この動作が予期しないものであることを確認します。

   このアクティビティが、データベースへのさらなる不正アクセスを試みているものかどうかを次のように検証します。

   - ソースが内部にある場合は、アプリケーションが誤って設定されていないかどうかを調べて、接続が繰り返し試行されていないかを確認します。
   - これが外部の攻撃者である場合は、該当するデータベースが一般公開されているかどうか、または設定が誤っていないかどうか調べ、悪意のある攻撃者が一般的なユーザー名に対してブルートフォース攻撃を仕掛けられるようになっていないかを確認します。

4. 予期しない動作が発生した場合は、このステップを開始してください。

   1. データベースアクセスを制限する

      疑わしいアカウントおよびこのログインアクティビティのソースによるデータベースへのアクセスを制限します。詳細については、「漏えいした可能性のある認証情報の修正」および「ネットワークアクセスを制限する」を参照してください。

   2. 根本原因を分析し、このアクティビティを引き起こした可能性のあるステップを特定します。

      アクティビティによってネットワークポリシーが変更され、安全でない状態になったときに通知を受けるようにアラートを設定します。詳細については、「AWS Network Firewall デベロッパーガイド」の「AWS Network Firewallにおけるファイアウォールポリシー」を参照してください。

異常な流出アクティビティによる侵害された可能性のあるデータベースの修復

以下の推奨ステップは、データ流出イベントに関連する異常な動作を示す、侵害された可能性のある Aurora データベースを修復するのに役立ちます。

1. 影響を受けたデータベースとユーザーを特定します。

   生成された GuardDuty の検出結果には、影響を受けたデータベースの名前と対応するユーザーの詳細が表示されます。詳細については、「検出結果の詳細」を参照してください。

2. アクセスされるデータの範囲を特定します。

   利用可能な RDS データアクティビティログを確認して、検出結果で示された時間枠内に異常なユーザーがどのテーブルと行にアクセスしたかを判断します。流出した可能性のあるデータの機密性を評価します。

3. この動作が予期しないものであることを確認します。

   このアクティビティが正当な運用動作を表しているかどうかを次のように調べます。

   - ソースが内部である場合は、スケジュールされた ETL ジョブ、レポートツール、またはバックアッププロセスが、昇格されたクエリボリュームを考慮する可能性があるかどうかを確認します。
   - これが外部アクターであるか、以前には見えなかったデータベース属性である場合は、対応するデータベース認証情報が外部で公開または共有されている可能性があるかどうかを確認します。
   - ボリューム属性と表示されない属性 (IP アドレス、ASN、アプリケーション名) の組み合わせが、既知の運用上の変更と一致しているかどうかを確認します。

4. 予期しない動作が発生した場合は、このステップを開始してください。

   アクセスを取り消し、認証情報をローテーションします。

   影響を受けるデータベースユーザーのパスワードを直ちに変更し、アクティブなセッションを取り消します。ユーザーに昇格された権限がある場合は、調査が完了するまで一時的に読み取り専用に制限するか、アクセスを完全に削除します。

5. ビジネスへの影響を評価します。

影響を受けたユーザーがアクセスできるデータを特定し、機密データまたは規制対象データ (PII、財務、医療) が流出した可能性があるかどうかを判断し、該当する場合は組織のデータ侵害通知手順に従います。

6. ネットワークアクセスを制限します。

   疑わしいアカウントとこのアクティビティのソースのデータベースアクセスを制限します。まだ設定されていない場合は、データベースをプライベート VPC に配置し、必要なソースからのトラフィックのみを許可するようにセキュリティグループのルールを制限します。詳細については、「ネットワークアクセスを制限する」を参照してください。

7. 根本原因を分析し、このアクティビティを引き起こした可能性のあるステップを特定します。

   アクティビティによってネットワークポリシーが変更され、安全でない状態になったときに通知を受けるようにアラートを設定します。詳細については、「AWS Network Firewall デベロッパーガイド」の「AWS Network Firewallにおけるファイアウォールポリシー」を参照してください。

異常な破壊的アクティビティによる侵害された可能性のあるデータベースの修復

以下の推奨ステップは、破壊的DELETEまたはDROPイベントに関連する異常な動作を示す、侵害された可能性のある Aurora データベースを修復するのに役立ちます。

1. 影響を受けたデータベースとユーザーを特定します。

   生成された GuardDuty の検出結果には、影響を受けたデータベースの名前と対応するユーザーの詳細が表示されます。詳細については、「検出結果の詳細」を参照してください。

2. 破壊的なオペレーションの範囲を特定します。

   利用可能な RDS データアクティビティログを確認して、検出結果で示された時間枠内に、異常なユーザーがどのテーブル、行、またはデータベースオブジェクトを削除または削除したかを判断します。データ損失が既存のバックアップまたはスナップショットから回復可能かどうかを判断します。

3. この動作が予期しないものであることを確認します。

   このアクティビティが正当な運用動作を表しているかどうかを次のように調べます。

- ソースが内部である場合は、スケジュールされたメンテナンスジョブ、クリーンアップスクリプト、または移行プロセスが DELETE または DROPオペレーションを考慮する可能性があるかどうかを確認します。
- これが外部アクターであるか、以前には見えなかったデータベース属性である場合は、対応するデータベース認証情報が公開または侵害された可能性があるかどうかを確認します。
- 破壊的なオペレーションと見えない属性 (IP アドレス、ASN、アプリケーション名) の組み合わせが、既知の運用上の変更と一致しているかどうかを確認します。

4. 予期しない動作が発生した場合は、このステップを開始してください。

   アクセスを取り消し、認証情報をローテーションします。

   影響を受けるデータベースユーザーのパスワードを直ちに変更し、アクティブなセッションを取り消します。調査が完了するまでユーザーの DROPおよび DELETE権限を削除して、データのさらなる破壊を防止します。

5. データ復旧を開始します。

   データ損失が確認された場合は、自動バックアップからpoint-in-timeリカバリを開始するか、最新のスナップショットから復元します。Amazon Aurora では、バックトラックを使用して、破壊的なアクティビティが発生する前の時間にデータベースを巻き戻すことができます。Amazon RDS の場合、point-in-timeリカバリを使用して、イベント前のポイントに復元します。

6. ネットワークアクセスを制限し、アクセス許可を強化します。

   疑わしいアカウントとこのアクティビティのソースのデータベースアクセスを制限します。まだ設定されていない場合は、データベースをプライベート VPC に配置し、必要なソースからのトラフィックのみを許可するようにセキュリティグループのルールを制限します。最小特権アクセスを適用するために IAM ポリシーとデータベースユーザー権限を確認し、まだ有効になっていない場合は RDS インスタンスで削除保護を有効にします。詳細については、「ネットワークアクセスを制限する」を参照してください。

7. 根本原因を分析し、このアクティビティを引き起こした可能性のあるステップを特定します。

   アクティビティによってネットワークポリシーが変更され、安全でない状態になったときに通知を受けるようにアラートを設定します。詳細については、「AWS Network Firewall デベロッパーガイド」の「AWS Network Firewallにおけるファイアウォールポリシー」を参照してください。

漏えいした可能性のある認証情報の修正

GuardDuty の検出結果には、検出結果で特定されたユーザーが予期しないデータベース操作を実行したときに、影響を受けたデータベースのユーザー認証情報が漏えいしたことが示される場合もあります。ユーザーを特定するには、コンソールの検出結果パネル内の [RDS DB user details] (RDS DB ユーザー詳細) セクション、または検出結果 JSON の `resource.rdsDbUserDetails` を確認します。これらのユーザーの詳細には、ユーザー名、使用したアプリケーション、アクセスしたデータベース、SSL バージョン、認証方法が含まれます。

- 検出結果に関係する特定のユーザーのアクセス権限を取り消したり、パスワードを変更したりするには、「Amazon Aurora ユーザーガイド」の「Amazon Aurora MySQL でのセキュリティ」または「Amazon Aurora PostgreSQL でのセキュリティ」を参照してください。
- を使用して AWS Secrets Manager 、Amazon Relational Database Service (RDS) データベースのシークレットを安全に保存し、自動的にローテーションします。詳細については、「AWS Secrets Manager ユーザーガイド」の「AWS Secrets Manager のチュートリアル」を参照してください。
- IAM データベース認証を使用すると、パスワードなしでデータベースユーザーのアクセスを管理できます。詳細については、「Amazon Aurora ユーザーガイド」の「IAM データベース認証」を参照してください。

  詳細については、「Amazon RDS ユーザーガイド」の「Amazon RDS のセキュリティのベストプラクティス」を参照してください。

ネットワークアクセスを制限する

GuardDuty の検出結果には、データベースがアプリケーションや仮想プライベートクラウド (VPC) の外部からのアクセスが可能であることが示されている場合があります。検出結果に示されているリモート IP アドレスが予期しない接続元である場合は、セキュリティグループを監査してください。データベースにアタッチされたセキュリティグループのリストは、https://console.aws.amazon.com/rds/ コンソールの [セキュリティグループ]、または検出結果 JSON の `resource.rdsDbInstanceDetails.dbSecurityGroups` で確認できます。セキュリティグループの設定については、「Amazon RDS ユーザーガイド」の「セキュリティグループによるアクセス制御」を参照してください。

ファイアウォールを使用している場合は、ネットワークアクセスコントロールリスト (NACL) を再設定して、データベースへのネットワークアクセスを制限します。詳細については、「AWS Network Firewall デベロッパーガイド」の「AWS Network Firewallのファイアウォール」を参照してください。

侵害された可能性のある Lambda 関数の修復

GuardDuty が Lambda Protection の検出結果タイプを生成するときに、Lambda 関数が侵害される可能性があります。この検出結果の原因となったアクティビティが想定内である場合は、抑制ルールの使用を検討してみてください。侵害された Lambda 関数を修正するには、以下の手順を実行することをお勧めします。

Lambda Protection の検出結果を修正するには

1. 侵害された可能性のある Lambda 関数のバージョンを確認します。

   Lambda Protection の GuardDuty の検出結果では、検出結果の詳細に記載されている Lambda 関数に関連する名前、Amazon リソースネーム (ARN)、関数バージョン、リビジョン ID がわかります。
2. 疑わしいアクティビティのソースを識別します。

   a. 検出結果に含まれる Lambda 関数のバージョンに関連するコードを確認してください。
   b. インポートされたライブラリと、検出結果に関係する Lambda 関数バージョンのレイヤーを確認します。
   c. Amazon Inspector で AWS Lambda 関数のスキャンを有効にしている場合は、検出結果に関連する Lambda 関数に関連付けられた Amazon Inspector の検出結果を確認します。
   d. AWS CloudTrail ログを確認して、関数の更新の原因となったプリンシパルを特定し、アクティビティが承認または予期されたことを確認します。
3. 侵害された可能性のある Lambda 関数を修復します。

   a. 検出結果に関係する Lambda 関数の実行トリガーを無効にします。詳細については、「DeleteFunctionEventInvokeConfig」を参照してください。
   b. Lambda コードを確認し、ライブラリのインポートと Lambda 関数レイヤーを更新して、潜在的に疑わしいライブラリとレイヤーを削除します。
   c. 検出結果に関係した Lambda 関数に関連する Amazon Inspector の検出結果を軽減します。

GuardDuty の使用状況のモニタリングとコストの見積もり

GuardDuty は、保護プランデータソースのログ/イベントと GuardDuty ランタイムでモニタリング VCPUs の処理を経時的に追跡する使用状況メトリクスを提供します。

このページでは、次の操作を行います。

- Amazon CloudWatch 使用状況メトリクス
- GuardDuty の使用について
- GuardDuty コストの見積もり

Amazon CloudWatch 使用状況メトリクス

GuardDuty は Amazon CloudWatch に使用状況メトリクスを発行し、次のことを可能にします。

- 時間の経過に伴う実際の使用状況を追跡する
- カスタムダッシュボードとアラームを作成する
- AWS 料金計算ツールでコスト見積もりの使用状況データをエクスポートする

GuardDuty 使用状況メトリクスは、アカウント設定に基づいて発行されます。

- すべてのアカウント (スタンドアロン、委任管理者、メンバーアカウント) は、Amazon CloudWatch で個々のアカウント使用状況メトリクスを受け取ります。
- 委任された管理者アカウントは、組織全体の集計された使用状況メトリクスをさらに受け取ります。

GuardDuty 使用状況メトリクスは、24 時間以内に Amazon CloudWatch で公開されます。

メトリクスの詳細

GuardDuty は、 `AWS/GuardDuty`名前空間の Amazon CloudWatch `Hourly` に次の使用状況メトリクスを発行します。

保護プラン	データソース	メトリクス名	[単位]	説明

基本的な脅威の検出	CloudTrailEvents	AnalyzedCount	カウント	分析された CloudTrail 管理イベントの数
基本的な脅威の検出	VPCFlowLogDNSLogEvents	AnalyzedBytes	バイト	分析された VPC フローログと DNS ログの量
EKS Protection	KubernetesAuditLogs	AnalyzedCount	カウント	分析された Amazon EKS 監査ログイベントの数
S3 Protection	S3DataEvents	AnalyzedCount	カウント	分析された S3 データイベントの数
ランタイムモニタリング	RuntimeMonitoringEC2	MonitoredVcpuHours	カウント (vCPU-時間)	Runtime Monitoring でモニタリングされる EC2 vCPU 時間
ランタイムモニタリング	RuntimeMonitoringEKS	MonitoredVcpuHours	カウント (vCPU-時間)	Runtime Monitoring でモニタリングされる Amazon EKS vCPU 時間
ランタイムモニタリング	RuntimeMonitoringFargate	MonitoredVcpuHours	カウント (vCPU-時間)	Runtime Monitoring によってモニタリングされる Fargate vCPU 時間

Malware Protection for EC2	OnDemandEBSSnapshot	ScannedBytes	バイト	スキャンされたオンデマンド EBS スナップショットデータの量
Malware Protection for EC2	OnDemandEBSVolume	ScannedBytes	バイト	スキャンされたオンデマンド EBS ボリュームデータの量
Malware Protection for EC2	MalwareProtectionEBS	ScannedBytes	バイト	Malware Protection によってスキャンされた EBS データの量
RDS Protection	RDS	MonitoredAcuHours	カウント (ACU 時間)	モニタリングされる Amazon RDS Aurora キャパシティーユニット
RDS Protection	RDSLimitless	MonitoredAcuHours	カウント (ACU 時間)	Amazon RDS Aurora Limitless ACU のモニタリング時間
RDS Protection	AuroraScaleout	MonitoredAcuHours	カウント (ACU 時間)	モニタリングされる Aurora スケールアウト ACU 時間
RDS Protection	RDS	MonitoredVcpuHours	カウント (vCPU-時間)	Amazon RDS vCPU のモニタリング時間

Lambda Protection	LambdaNetworkLogs	AnalyzedBytes	バイト	分析された Lambda ネットワークログの量
AI 保護	AIDataEvents	AnalyzedBytes	バイト	分析された AI データイベントの量

メトリクスディメンション

- すべてのアカウント (スタンドアロン、委任管理者、メンバーアカウント): メトリクスには `AccountId`および `DataSource`ディメンションが含まれます。
- 委任管理者アカウント: メトリクスには、組織内のすべてのメンバーアカウントにわたる集計`DataSource`ディメンションも含まれます。

S3 向けのマルウェア防御

GuardDuty `Malware Protection for S3`保護プランは、 `AWS/GuardDuty/MalwareProtection`名前空間の Amazon CloudWatch に次の使用状況メトリクスを発行します。

メトリクス名	[単位]	説明
CompletedScanCount	カウント	特定の時間枠で完了した S3 オブジェクトマルウェアスキャンの数。
FailedScanCount	カウント	特定の時間枠で失敗した S3 オブジェクトマルウェアスキャンの数。
SkippedScanCount	カウント	特定の時間枠でスキップされた S3 オブジェクトマルウェアスキャンの数。
InfectedScanCount	カウント	特定の時間枠で悪意の可能性があるオブジェクトを検出し

		た S3 オブジェクトマルウェアスキャンの数。
CompletedScanBytes	カウント	特定の時間枠でスキャンされた S3 オブジェクトバイトの数。

メトリクスディメンション

- すべてのメトリクスには`Malware Protection Plan Id, Resource Name`ディメンションが含まれます
- SkippedScanCount メトリクスには、追加のディメンション`Skipped Reason`として が含まれます

GuardDuty の使用について

GuardDuty イベント処理

有効にすると、GuardDuty は選択した のログソースから直接イベントとログを自動的に消費します AWS リージョン。GuardDuty は、個別の独立したデータソースからイベントを取り込んで、包括的なセキュリティ価値を提供します。

> **Important**
>
> 個々のサービスログ設定またはフィルタリングルール (VPC フローログ、DNS ログ、CloudTrail イベント、S3 データイベント、Kubernetes 監査ログ、Lambda ネットワークログの場合) は、GuardDuty によって処理されるログ/イベントには影響しません。

GuardDuty Runtime Monitoring によってモニタリングされるインスタンスの GuardDuty VPC フローログ処理料金

GuardDuty Runtime Monitoring (EC2 Runtime エージェントまたは Amazon EKS Runtime エージェント経由) によってモニタリングされるインスタンスの場合、エージェントがランタイムイベントデータをアクティブに送信している限り、GuardDuty は VPC フローログの処理に対して課金しませ

ん。エージェントがイベントデータの送信を停止すると、GuardDuty は VPC フローログを介した課金に戻ります。

Runtime Monitoring を有効にすると、GuardDuty Amazon CloudWatch 使用状況メトリクスの VPC フローログの使用量が減少します。Runtime Monitoring を無効にすると、VPC フローログの使用状況が復元されます。

GuardDuty コストの見積もり

GuardDuty は、ほとんどの保護プランで AWS アカウントごとに 30 日間の無料トライアルを提供しています。このトライアル期間中は、次のことができます。

- GuardDuty 使用状況メトリクスを使用して実際の使用状況をモニタリングする
- 観測された使用パターンに基づいて AWS 料金計算ツールを使用して月額コストを見積もる

次の保護プランには、30 日間の無料トライアルが含まれています。

- 基本的な脅威の検出
- S3 Protection
- EKS Protection
- ランタイムモニタリング
- RDS Protection
- Lambda Protection
- AI 保護
- Malware Protection for EC2 (基本的な脅威検出で有効になっている場合、GuardDuty が開始するスキャンのみ)

AWS 料金計算ツールの使用

AWS 料金計算ツールを使用して、観測された使用パターンに基づいて毎月の GuardDuty コストを見積もることができます。GuardDuty の見積りを作成するには:

1. AWS 料金計算ツールを開きます。
2. [Create estimate (見積りの作成)] を選択します。
3. サービスの追加ページで、Amazon GuardDuty を検索し、設定を選択します。

4. GuardDuty が有効になってい AWS リージョン る を選択します。
5. サービス設定セクションで、Amazon CloudWatch または GuardDuty コンソールで観測された使用状況メトリクスに基づいて、各保護プランの推定使用量を入力します。
6. 見積りに追加を選択して、予想される月額コストを表示します。

> **Note**
>
> 一部の GuardDuty 使用状況メトリクスはバイト単位で報告されます。 AWS 料金計算ツールに値を入力するときは、バイトを適切な単位 (MB、GB、TB) に変換する必要がある場合があります。次の変換を使用します。
>
> - 1 MB = 1,048,576 バイト
> - 1 GB = 1,073,741,824 バイト
> - 1 TB = 1,099,511,627,776 バイト

Security Hub のお客様

Security Hub は、アドオン Threat Analytics プランを使用して GuardDuty Threat Detection の簡素化された料金モデルを提供し、複数の GuardDuty DataSources。Security Hub Threat Analytics プラン (Security Hub with GuardDuty) を使用する場合:

- 複数の GuardDuty DataSources統合されている
- 特に、わかりやすくするために、Amazon EKS 監査ログイベントと S3 データイベントは、固定変換レートを使用して GB に変換されます。

Security Hub のコスト見積もりを作成するには、AWS Security Hub のドキュメントを参照してください。

注: GuardDuty の 30 日間の無料トライアルステータスは、Security Hub の統合とは無関係です。Security Hub の有効化または無効化:

- GuardDuty のトライアル期間をすでに使用している場合、新しい無料トライアルは付与されません
- 進行中の無料トライアルを中断または再開しない
- 既存のトライアル期間を延長しない

GuardDuty API の保護プランの機能名

Amazon GuardDuty を初めて有効にすると、 AWS 環境内で基本データソースの処理が開始されます。GuardDuty は、これらのデータソースを使用して、VPC フローログ、DNS ログ、 AWS CloudTrail 管理イベントなどのイベントの独立したストリームを処理します。次に、これらのイベントを分析して潜在的なセキュリティ脅威を特定し、アカウントに関する検出結果を生成します。

1 つ以上の保護プランが有効になっている場合、GuardDuty は AWS 環境内の他の AWS サービスの追加のデータを使用して、潜在的なセキュリティ脅威をモニタリングおよび分析します。これらの追加のデータソースは、機能と呼ばれます。

データソースから機能への変更

S3 Protection、Runtime Monitoring、Lambda Protection などの GuardDuty 保護を追加すると、保護プランに対応する GuardDuty 機能を設定できます。従来、GuardDuty 保護は API で `dataSources` が呼び出されていました。ただし、2023 年 3 月以降、新しい GuardDuty 保護プランは `dataSources` ではなく、`features` として設定されるようになりました。GuardDuty は、`dataSources` として API 経由で 2023 年 3 月より前にリリースされた保護プランの設定を引き続きサポートしていますが、新しい保護プランは `features` としてのみ利用できます。影響を受ける保護プランについては、「GuardDuty API の変更」を参照してください。

コンソールから GuardDuty の設定と保護プランを管理する場合、この変更による直接の影響は受けないため、何もする必要はありません。この変更は、GuardDuty または GuardDuty 内の保護プランを有効にするために呼び出される API の動作に影響します。APIs または を使用して保護プランの設定 AWS CLI を有効化または編集する場合は、関連する機能名を使用する必要があります。詳細については、「`dataSources` を `features` へマッピング」を参照してください。

2023 年 3 月の GuardDuty API の変更

GuardDuty API は、GuardDuty 基本データソース のリストに含まれていない保護機能を設定します。機能オブジェクトには、機能名やステータスなどの機能の詳細が含まれ、一部の保護プランの追加設定が含まれる場合があります。この移行は、「Amazon GuardDuty API リファレンス」の次の API に影響します。

- CreateDetector
- GetDetector

- UpdateDetector
- GetMemberDetectors
- UpdateMemberDetectors
- DescribeOrganizationConfiguration
- UpdateOrganizationConfiguration
- GetRemainingFreeTrialDays
- GetUsageStatistics

データソースと比較した機能

これまで、GuardDuty の機能はすべて API の dataSources オブジェクトを介して渡されていました。2023 年 3 月以降、GuardDuty は API 内の dataSources オブジェクトではなく features オブジェクトを優先するようになりました。以前のデータソースにはすべて対応する機能がありますが、新しい機能には対応するデータソースがない場合があります。

次のリストは、API を介して渡されたときの dataSources および features オブジェクトとの比較を示しています。

- dataSources オブジェクトには、各保護タイプとそのステータスのオブジェクトが含まれています。features オブジェクトは、GuardDuty 内の各保護タイプに対応する利用可能な機能のリストです。

  2023 年 3 月以降、機能のアクティベーションは、 AWS 環境で新しい GuardDuty 機能を設定する唯一の方法です。
- API リクエストまたはレスポンス内の dataSources スキーマは、GuardDuty が利用可能な各 AWS リージョン で同じです。ただし、すべての機能が各リージョンで利用できるわけではありません。そのため、利用可能な機能名は地域によって異なる場合があります。

機能を使用する API の仕組みを理解する

GuardDuty API は引き続き該当する dataSources オブジェクトを返し、同じ情報を含む features オブジェクトを異なる形式で返します。2023 年 3 月以前にリリースされた GuardDuty 機能は、dataSources オブジェクトと features オブジェクトを通じて利用可能になります。2023 年 3 月以降に GuardDuty がリリースされた機能は、features オブジェクトを通じてのみ利用できます。ディテクターを作成または更新したり、同じ API リクエストで dataSourcesと

features オブジェクトの両方の表記 AWS Organizations を使用して を記述したりすることはできません。GuardDuty 保護タイプを有効にするには、features オブジェクトも含めた同じ API を使用して、既存のデータソースを features オブジェクトに移行する必要があります。

> **Note**
>
> GuardDuty は、この変更後に新しいデータソースを追加しません。

GuardDuty は、保護プランに関連付けられているデータソースの使用を廃止しました。ただし、まだ GuardDuty 基本データソース をサポートしています。GuardDuty のベストプラクティスでは、アカウント内の保護プランの設定を有効化または編集するために機能を使用することをお勧めします。

API への機能の変更の組み込み

- APIs、SDKs、または CloudFormation テンプレートを使用して GuardDuty 設定を管理し、潜在的な新しい GuardDuty 機能を有効にする場合は、それぞれコードとテンプレートを変更する必要があります。詳細については、「Amazon GuardDuty API リファレンス」の更新済み API を参照してください。
- このアップグレード前に設定された GuardDuty 機能の場合、APIs、SDKs、または CloudFormation テンプレートを引き続き使用できます。ただし、feature オブジェクトの使用に切り替えることをお勧めします。

  すべてのデータソースには同等の機能オブジェクトがあります。詳細については、「dataSources を features へマッピング」を参照してください。
- 現在、features オブジェクトの additionalConfiguration は特定の保護タイプでのみ利用できます。
  - このような保護タイプの場合、機能の AdditionalConfiguration status が ENABLED に設定されていても、機能の設定 status が ENABLED に設定されていないときには、GuardDuty はいかなるアクションも実行しません。
  - 次の API はこの影響を受けます。
    - UpdateDetector
    - UpdateMemberDetectors
    - UpdateOrganizationConfiguration

dataSources を features へマッピング

次の表に、保護タイプである dataSources および features のマッピングを示します。

GuardDuty 保護タイプ	データソース名 *	機能名
VPC フローログ	flowLogs (読み込み専用、変更不可)	FLOW_LOGS (読み込み専用、変更不可)
Route53 Resolver DNS クエリログ	dnsLogs (読み込み専用、変更不可)	DNS_LOGS (読み込み専用、変更不可)
'CloudTrail イベント	cloudTrail (読み込み専用、変更不可)	CLOUD_TRAIL (読み込み専用、変更不可)
S3	s3Logs	S3_DATA_EVENTS
EKS Protection	kubernetes.auditlogs	EKS_AUDIT_LOGS
Malware Protection for EC2	malwareProtection.scanEc2InstanceWithFindings.ebsVolumes	EBS_MALWARE_PROTECTION
RDS ログインイベント	GuardDuty は、これらの保護タイプの機能アクティベーションのみをサポートします。	RDS_LOGIN_EVENTS
EKS Runtime Monitoring		EKS_RUNTIME_MONITORING
Runtime Monitoring		RUNTIME_MONITORING
Amazon EKS クラスター用の GuardDuty セキュリティエージェント		EKS_RUNTIME_MONITORING.additionalConfiguratio

GuardDuty 保護タイプ	データソース名 *	機能名
		n.EKS_ADDON_MANAGEMENT RUNTIME_MONITORING.additionalConfiguration.EKS_ADDON_MANAGEMENT
Amazon ECS-Fargate クラスター用の GuardDuty セキュリティエージェント		RUNTIME_MONITORING.additionalConfiguration.ECS_FARGATE_AGENT_MANAGEMENT
Amazon EC2 インスタンス用の GuardDuty セキュリティエージェント		RUNTIME_MONITORING.additionalConfiguration.EC2_AGENT_MANAGEMENT
Lambda Protection		LAMBDA_NETWORK_LOGS

*GetUsageStatistics は独自の `dataSource` 名前を使用します。詳細については、GuardDuty の使用状況のモニタリングとコストの見積もりまたはGetUsageStatisticsを参照してください。

Amazon GuardDuty のセキュリティ

でのクラウドセキュリティが最優先事項 AWS です。お客様は AWS 、セキュリティを最も重視する組織の要件を満たすように構築されたデータセンターとネットワークアーキテクチャを活用できます。

セキュリティは、 AWS とお客様の間の責任共有です。責任共有モデルでは、これをクラウドのセキュリティおよびクラウド内のセキュリティと説明しています。

- クラウドのセキュリティ – AWS クラウドで AWS サービスを実行するインフラストラクチャを保護する AWS 責任があります。 AWS また、 では、安全に使用できるサービスも提供しています。サードパーティーの監査者は、AWS コンプライアンスプログラムコンプライアンスプログラムの一環として、当社のセキュリティの有効性を定期的にテストおよび検証します。GuardDuty に適用されるコンプライアンスプログラムの詳細については、「コンプライアンスAWS プログラムによる対象範囲内のサービスコンプライアンス」を参照してください。
- クラウドのセキュリティ – お客様の責任は、使用する AWS サービスによって決まります。また、お客様は、お客様のデータの機密性、企業の要件、および適用可能な法律および規制などの他の要因についても責任を担います。

このドキュメントは、GuardDuty を使用する際に責任共有モデルを適用する方法を理解するのに役立ちます。ここでは、セキュリティとコンプライアンスの目標を満たすように GuardDuty を設定する方法を説明します。また、GuardDuty リソースのモニタリングや保護に役立つ他の AWS サービスの使用方法についても説明します。

内容

- Amazon GuardDuty におけるデータ保護
- を使用した Amazon GuardDuty API コールのログ記録 AWS CloudTrail
- Amazon GuardDuty のための Identity and Access Management
- Amazon GuardDuty のコンプライアンス検証
- Amazon GuardDuty の回復力
- Amazon GuardDuty でのインフラストラクチャセキュリティ
- Amazon GuardDuty とインターフェイス VPC エンドポイント (AWS PrivateLink)

Amazon GuardDuty におけるデータ保護

責任共有モデル、Amazon AWS GuardDuty でのデータ保護に適用されます。 Amazon GuardDuty このモデルで説明されているように、 AWS はすべての を実行するグローバルインフラストラクチャを保護する責任があります AWS クラウド。ユーザーは、このインフラストラクチャでホストされるコンテンツに対する管理を維持する責任があります。また、使用する「 AWS のサービス 」のセキュリティ設定と管理タスクもユーザーの責任となります。データプライバシーの詳細については、「Data Privacy FAQChina」を参照してください。 欧州におけるデータ保護に関する情報については、General Data Protection Regulation (GDPR) Center を参照してください。

データ保護の目的で、認証情報を保護し AWS アカウント 、 AWS IAM アイデンティティセンターまたは AWS Identity and Access Management (IAM) を使用して個々のユーザーを設定することをお勧めします。この方法により、それぞれのジョブを遂行するために必要な権限のみが各ユーザーに付与されます。また、次の方法でデータを保護することもお勧めします:

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用して AWS リソースと通信します。TLS 1.2 は必須ですが、TLS 1.3 を推奨します。
- で API とユーザーアクティビティのログ記録を設定します AWS CloudTrail。CloudTrail 証跡を使用して AWS アクティビティをキャプチャする方法については、「 AWS CloudTrail ユーザーガイド」のCloudTrail 証跡の使用」を参照してください。
- AWS 暗号化ソリューションと、その中のすべてのデフォルトのセキュリティコントロールを使用します AWS のサービス。
- Amazon Macie などの高度な管理されたセキュリティサービスを使用します。これらは、Amazon S3 に保存されている機密データの検出と保護を支援します。
- コマンドラインインターフェイスまたは API AWS を介して にアクセスするときに FIPS 140-3 検証済み暗号化モジュールが必要な場合は、FIPS エンドポイントを使用します。利用可能な FIPS エンドポイントの詳細については、「連邦情報処理規格 (FIPS) 140-3」を参照してください。

お客様の E メールアドレスなどの極秘または機密情報を、タグ、または [名前] フィールドなどの自由形式のテキストフィールドに含めないことを強くお勧めします。これは、コンソール、API、または SDK を使用して GuardDuty AWS CLIまたは他の AWS のサービス を使用する場合も同様です。AWS SDKs タグ、または名前に使用される自由記述のテキストフィールドに入力したデータは、請求または診断ログに使用される場合があります。外部サーバーに URL を提供する場合、そのサーバーへのリクエストを検証できるように、認証情報を URL に含めないことを強くお勧めします。

保管中の暗号化

GuardDuty のすべての顧客データは、保管時に暗号化ソリューションを使用して AWS 暗号化されます。

検出結果などの GuardDuty データは、 所有キーで AWS Key Management Service AWS (AWS KMS) を使用して保管時に暗号化されます。

転送中の暗号化

GuardDuty は、他のサービスからのログデータを分析します。GuardDuty は、これらのサービスから転送中のデータすべてを HTTPS および KMS で暗号化します。GuardDuty が ログから必要な情報を抽出すると、それらは破棄されます。GuardDuty で他のサービスの情報を使用する方法の詳細については、「GuardDuty data sources」(GuardDuty データソース) を参照してください。

GuardDuty データは、サービス間で転送時に暗号化されます。

サービス改善のためのデータ使用をオプトアウトする

オプトアウトポリシーを使用して、GuardDuty およびその他の AWS セキュリティサービスの開発と改善にデータを使用することを AWS Organizations オプトアウトできます。GuardDuty が現在そのようなデータを収集していない場合でも、オプトアウトすることができます。オプトアウトする方法の詳細については、「AWS Organizations ユーザーガイド」の「AI サービスのオプトアウトポリシー」を参照してください。

> **Note**
>
> オプトアウトポリシーを使用するには、 AWS アカウントが によって一元管理されている必要があります AWS Organizations。 AWS アカウントの組織をまだ作成していない場合は、AWS Organizations 「 ユーザーガイド」の「組織の作成と管理」を参照してください。

オプトアウトには次のような効果があります。

- GuardDuty は、オプトアウト前にサービス向上を目的として収集して保存したデータ (ある場合) を削除します。
- オプトアウトすると、GuardDuty はこのデータをサービス向上を目的として収集または保存しなくなります。

以下のトピックでは、GuardDuty 内の各機能によるサービスの向上を目的としたデータの処理方法を説明します。

内容

- GuardDuty Runtime Monitoring
- GuardDuty Malware Protection

GuardDuty Runtime Monitoring

GuardDuty Runtime Monitoring は、 AWS 環境内の Amazon Elastic Kubernetes Service (Amazon EKS) クラスター、 AWS Fargate Amazon Elastic Container Service (Amazon ECS) のみ、Amazon Elastic Compute Cloud (Amazon EC2) の各インスタンスのランタイム脅威検出を可能にします。Runtime Monitoring を有効にし、リソースの GuardDuty セキュリティエージェントをデプロイすると、GuardDuty ではリソースに関連するランタイムイベントの監視と分析が開始されます。これらのランタイムイベントタイプには、プロセスイベント、コンテナイベント、DNS イベントなどが含まれます。詳細については、「GuardDuty が使用する収集されたランタイムイベントタイプ」を参照してください。

GuardDuty は、ワークロードからコマンド (`curl`、`systemctl`、`cron` など) とそれに関連する引数 (`start`、`stop`、`disable` など) の両方を収集します。例えば、誰かが `systemctl stop` *`service-name`* を実行すると、GuardDuty はコマンド `systemctl`とその引数 `stop` *`service-name`* の両方をキャプチャします。この詳細情報は、コマンドパターンを分析し、複数のイベントを関連付けることで、GuardDuty が高度な脅威を検出するのに役立ちます。例えば、GuardDuty は、攻撃者がセキュリティサービスを無効にしようとしたり、既知の悪意のあるファイルを実行したりするタイミングを特定できます。GuardDuty は脅威検出のためにこのデータを積極的に使用しますが、現在、サービス改善の目的でこれらのコマンドと引数を使用しません (将来使用される場合があります)。信頼、プライバシー、およびコンテンツのセキュリティが当社の最優先事項であり、当社の使用に際してユーザーへの約束に確実に従うことを保証します。詳細については、データプライバシーのよくある質問を参照してください。

GuardDuty Malware Protection

GuardDuty Malware Protection は、侵害された可能性のある Amazon EC2 インスタンスとコンテナワークロード、選択した Amazon S3 バケットに新しくアップロードされたファイル、バックアップリソースにアタッチされた EBS ボリュームに含まれるマルウェアをスキャンして検出します。現在、GuardDuty は、検出されたマルウェアをサービス改善のために収集または使用しません。ただし、将来、GuardDuty Malware Protection が EBS ボリュームファイル、バックアップファイル、ま

たは S3 ファイルを悪意のあるファイルまたは有害ファイルとして識別すると、GuardDuty Malware Protection はこのファイルを収集して保存し、マルウェア検出と GuardDuty サービスを開発および改善します。このファイルは、その他の AWS サービスの開発と改善にも使用される場合があります。信頼、プライバシー、およびコンテンツのセキュリティが当社の最優先事項であり、当社の使用に際してユーザーへの約束に確実に従うことを保証します。詳細については、データプライバシーのよくある質問を参照してください。

を使用した Amazon GuardDuty API コールのログ記録 AWS CloudTrail

Amazon GuardDuty は AWS CloudTrail、GuardDuty のユーザー、ロール、または のサービスによって実行されたアクションを記録する AWS サービスである と統合されています。CloudTrail は、GuardDuty コンソールからの呼び出しや GuardDuty API へのコード呼び出しを含む、GuardDuty のすべての API コールをイベントとしてキャプチャします。追跡を作成する場合は、GuardDuty に関するイベントを含めた CloudTrail のイベントの Amazon Simple Storage Service (Amazon S3) バケットへの継続的な配信を有効にすることができます。証跡を設定しない場合でも、CloudTrail コンソールの [イベント履歴] で最新のイベントを表示できます。CloudTrail で収集された情報を使用して、GuardDuty に対するリクエスト、そのリクエストが発信された IP アドレス、リクエストの作成者、リクエスト作成日時、その他の詳細情報などを確認できます。

CloudTrail を設定して有効にする方法などの詳細については、AWS CloudTrail ユーザーガイドを参照してください。

CloudTrail での GuardDuty 情報

CloudTrail は、 AWS アカウントの作成時にアカウントで有効になります。GuardDuty でサポートされているイベントアクティビティが発生すると、そのアクティビティはイベント履歴の他の AWS サービスイベントとともに CloudTrail イベントに記録されます。 AWS アカウントで最近のイベントを表示、検索、ダウンロードできます。詳細については、「CloudTrail Event 履歴でのイベントの表示」を参照してください。

GuardDuty のイベントなど、 AWS アカウントのイベントの継続的な記録については、証跡を作成します。証跡により、ログファイルを CloudTrail で Amazon S3 バケットに配信できます。デフォルトでは、コンソールで証跡を作成すると、すべての リージョンに証跡が適用されます。証跡は、AWS パーティション内のすべてのリージョンからのイベントをログに記録し、指定した Amazon S3 バケットにログファイルを配信します。さらに、CloudTrail ログで収集されたイベントデータをさらに分析して処理するように他の AWS サービスを設定できます。詳細については、以下を参照してください。

- 証跡を作成するための概要
- CloudTrail がサポートされているサービスと統合
- 「CloudTrail の Amazon SNS 通知の設定」
- 複数のリージョンから CloudTrail ログファイルを受け取るおよび複数のアカウントから CloudTrail ログファイルを受け取る

各イベントまたはログエントリには、誰がリクエストを生成したかという情報が含まれます。アイデンティティ情報は、以下を判別するのに役立ちます。

- リクエストが、ルートユーザーまたは IAM ユーザーのどちらのサインイン認証情報を使用して送信されたか
- リクエストの送信に使用された一時的なセキュリティ認証情報に、ロールとフェデレーテッドユーザーのどちらが使用されたか
- リクエストが別の AWS サービスによって行われたかどうか

詳細については、CloudTrail userIdentity 要素を参照してください。

CloudTrail の GuardDuty コントロールプレーンイベント

デフォルトでは、CloudTrail は Amazon GuardDuty API リファレンスで提供されているすべての GuardDuty API オペレーションを CloudTrail ファイルにイベントとして記録します。

CloudTrail の GuardDuty データイベント

GuardDuty Runtime Monitoring は、Amazon Elastic Kubernetes Service (Amazon EKS) クラスター、Amazon Elastic Compute Cloud (Amazon EC2) インスタンス、および AWS Fargate (Amazon Elastic Container Service (Amazon ECS) のみ) タスクにデプロイされた GuardDuty セキュリティエージェントを使用して、 AWS ワークロード収集されたランタイムイベントタイプ用に が収集するアドオン (`aws-guardduty-agent`) を収集し、脅威の検出と分析のために GuardDuty に送信します。

データイベントのログとモニタリング

オプションで、GuardDuty セキュリティエージェントのデータイベントを表示するように AWS CloudTrail ログを設定できます。

CloudTrail を作成して設定するには、「AWS CloudTrail ユーザーガイド」の「データイベント」を参照して、「 AWS マネジメントコンソールの高度なイベントセレクターによるデータイベントのロギング」の手順に従ってください。トレイルを記録するには、以下の変更を実行します。

- [データイベントタイプ] には、[GuardDuty ディテクター] を選択します。
- [ログセレクターテンプレート] では、[すべてのイベントをログに記録する] を選択します。
- 設定の [JSON ビュー] を展開します。次の JSON と同じようになります。

```
[
  {
    "name": "",
    "fieldSelectors": [
      {
        "field": "eventCategory",
        "equals": [
          "Data"
        ]
      },
      {
        "field": "resources.type",
        "equals": [
          "AWS::GuardDuty::Detector"
        ]
      }
    ]
  }
]
```

トレイルのセレクターを有効にした後、https://console.aws.amazon.com/s3/ にある Amazon S3 コンソールに移動します。CloudTrail ログの設定時に選択した S3 バケットからデータイベントをダウンロードできます。

例: GuardDuty ログアーカイブエントリ

追跡は、指定した Amazon S3 バケットにイベントをログファイルとして配信できるようにする設定です。CloudTrail のログファイルは、単一か複数のログエントリを含みます。イベントは、任意の出典からの単一のリクエストを表し、リクエストされたアクション、アクションの日時、リクエストパラメータなどに関する情報が含まれます。CloudTrail ログファイルは、パブリック API 呼び出しの順序付けられたスタックトレースではないため、特定の順序では表示されません。

次は、データプレーンイベントを示す CloudTrail ログエントリの例です。


```
{
        "eventVersion": "1.08",
        "userIdentity": {
            "type": "AssumedRole",
            "principalId": "111122223333:aws:ec2-instance:i-123412341234example",
            "arn": "arn:aws:sts::111122223333:assumed-role/aws:ec2-
instance/i-123412341234example",
            "accountId": "111122223333",
            "accessKeyId": "AKIAI44QH8DHBEXAMPLE",
            "sessionContext": {
                "sessionIssuer": {
                    "type": "Role",
                    "principalId": "111122223333:aws:ec2-instance",
                    "arn": "arn:aws:iam::111122223333:role/aws:ec2-instance",
                    "accountId": "111122223333",
                    "userName": "aws:ec2-instance"
                },
                "attributes": {
                    "creationDate": "2023-03-05T04:00:21Z",
                    "mfaAuthenticated": "false"
                },
                "ec2RoleDelivery": "2.0"
            }
        },
        "eventTime": "2023-03-05T06:03:49Z",
        "eventSource": "guardduty.amazonaws.com",
        "eventName": "SendSecurityTelemetry",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "54.240.230.177",
        "userAgent": "aws-sdk-rust/0.54.1 os/linux lang/rust/1.66.0",
        "requestParameters": null,
        "responseElements": null,
        "requestID": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111",
        "eventID": "a1b2c3d4-5678-90ab-cdef-EXAMPLEbbbbb",
        "readOnly": false,
        "resources": [{
            "accountId": "111122223333",
            "type": "AWS::GuardDuty::Detector",
            "ARN": "arn:aws:guardduty:us-
west-2:111122223333:detector/12abc34d567e8fa901bc2d34e56789f0"
        }],
        "eventType": "AwsApiCall",
```

```
        "managementEvent": false,
        "recipientAccountId": "111122223333",
        "eventCategory": "Data",
        "tlsDetails": {
            "tlsVersion": "TLSv1.2",
            "cipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
            "clientProvidedHostHeader": "guardduty-data.us-east-1.amazonaws.com"
        }
    }
```

次の例では、CreateIPThreatIntelSet アクション (コントロールプレーンイベント) を示す CloudTrail ログエントリを示しています。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::444455556666:user/Alice",
        "accountId": "444455556666",
        "accessKeyId": "AKIAI44QH8DHBEXAMPLE",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-06-14T22:54:20Z"
            },
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AIDACKCEVSQ6C2EXAMPLE",
                "arn": "arn:aws:iam::444455556666:user/Alice",
                "accountId": "444455556666",
                "userName": "Alice"
            }
        }
    },
    "eventTime": "2018-06-14T22:57:56Z",
    "eventSource": "guardduty.amazonaws.com",
    "eventName": "CreateThreatIntelSet",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "54.240.230.177",
    "userAgent": "console.amazonaws.com",
    "requestParameters": {
        "detectorId": "12abc34d567e8fa901bc2d34e56789f0",
```

```
        "name": "Example",
        "format": "TXT",
        "activate": false,
        "location": "https://s3.amazonaws.com/bucket.name/file.txt"
    },
    "responseElements": {
        "threatIntelSetId": "1ab200428351c99d859bf61992460d24"
    },
    "requestID": "5f6bf981-7026-11e8-a9fc-5b37d2684c5c",
    "eventID": "81337b11-e5c8-4f91-b141-deb405625bc9",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "recipientAccountId": "444455556666"
}
```


このイベント情報からは、GuardDuty で脅威リスト `Example` を作成するリクエストが行われたことを判断できます。また、このリクエストは、Alice という名前のユーザーによって 2018 年 6 月 14 日に行われたことも確認できます。

Amazon GuardDuty のための Identity and Access Management

AWS Identity and Access Management (IAM) は、管理者が AWS リソースへのアクセスを安全に制御 AWS のサービス するのに役立つ です。IAM 管理者は、誰を認証したか (サインイン)、誰が GuardDuty リソースを使う認証 (許可) を得たのかを管理します。IAM は、追加料金なしで使用できる AWS のサービス です。

トピック

- オーディエンス
- アイデンティティを使用した認証
- ポリシーを使用したアクセスの管理
- Amazon GuardDuty が IAM で機能する方法
- Amazon GuardDuty のためのアイデンティティベースのポリシーの例
- Amazon GuardDuty でのサービスにリンクされたロールの使用
- AWS Amazon GuardDuty の マネージドポリシー
- Amazon GuardDuty アイデンティティとアクセスのトラブルシューティング

オーディエンス

AWS Identity and Access Management (IAM) の使用方法は、ロールによって異なります。

- サービスユーザー - 機能にアクセスできない場合は、管理者にアクセス許可をリクエストします (「Amazon GuardDuty アイデンティティとアクセスのトラブルシューティング」を参照)。
- サービス管理者 - ユーザーアクセスを決定し、アクセス許可リクエストを送信します (「Amazon GuardDuty が IAM で機能する方法」を参照)
- IAM 管理者 - アクセスを管理するためのポリシーを作成します (「Amazon GuardDuty のためのアイデンティティベースのポリシーの例」を参照)

アイデンティティを使用した認証

認証とは、ID 認証情報 AWS を使用して にサインインする方法です。、IAM ユーザー AWS アカウントのルートユーザー、または IAM ロールを引き受けることで認証される必要があります。

AWS IAM アイデンティティセンター (IAM Identity Center)、シングルサインオン認証、Google/Facebook 認証情報などの ID ソースからの認証情報を使用して、フェデレーティッド ID としてサインインできます。サインインの詳細については、「AWS サインイン ユーザーガイド」の「AWS アカウントにサインインする方法」を参照してください。

プログラムによるアクセスの場合、 は SDK と CLI AWS を提供してリクエストを暗号化して署名します。詳細については、「IAM ユーザーガイド」の「API リクエストに対するAWS 署名バージョン 4」を参照してください。

AWS アカウント ルートユーザー

を作成するときは AWS アカウント、すべての AWS のサービス および リソースへの完全なアクセス権を持つ AWS アカウント ルートユーザーと呼ばれる 1 つのサインインアイデンティティから始めます。日常的なタスクには、ルートユーザーを使用しないことを強くお勧めします。ルートユーザー認証情報を必要とするタスクについては、「IAM ユーザーガイド」の「ルートユーザー認証情報が必要なタスク」を参照してください。

フェデレーテッドアイデンティティ

ベストプラクティスとして、人間のユーザーが一時的な認証情報 AWS のサービス を使用して にアクセスするには、ID プロバイダーとのフェデレーションを使用する必要があります。

フェデレーティッド ID は、エンタープライズディレクトリ、ウェブ ID プロバイダー、または ID Directory Service ソースの認証情報 AWS のサービス を使用して にアクセスするユーザーです。フェデレーテッドアイデンティティは、一時的な認証情報を提供するロールを引き受けます。

アクセスを一元管理する場合は、 AWS IAM アイデンティティセンターをお勧めします。詳細については、「AWS IAM アイデンティティセンター ユーザーガイド」の「IAM アイデンティティセンターとは」を参照してください。

IAM ユーザーとグループ

IAM ユーザーは、特定の個人やアプリケーションに対する特定のアクセス許可を持つアイデンティティです。長期認証情報を持つ IAM ユーザーの代わりに一時的な認証情報を使用することをお勧めします。詳細については、IAM ユーザーガイドの「ID プロバイダーとのフェデレーションを使用して にアクセスする必要がある AWS」を参照してください。

IAM グループは、IAM ユーザーの集合を指定し、大量のユーザーに対するアクセス許可の管理を容易にします。詳細については、「IAM ユーザーガイド」の「IAM ユーザーに関するユースケース」を参照してください。

IAM ロール

IAM ロールは、特定のアクセス許可を持つアイデンティであり、一時的な認証情報を提供します。ユーザーから IAM ロール (コンソール) に切り替えるか、 または API オペレーションを呼び出すことで、ロールを引き受けることができます。 AWS CLI AWS 詳細については、「IAM ユーザーガイド」の「ロールを引き受けるための各種方法」を参照してください。

IAM ロールは、フェデレーションユーザーアクセス、一時的な IAM ユーザーのアクセス許可、クロスアカウントアクセス、クロスサービスアクセス、および Amazon EC2 で実行するアプリケーションに役立ちます。詳細については、IAM ユーザーガイド の IAM でのクロスアカウントリソースアクセス を参照してください。

ポリシーを使用したアクセスの管理

でアクセスを制御する AWS には、ポリシーを作成し、ID AWS またはリソースにアタッチします。ポリシーは、アイデンティティまたはリソースに関連付けられている場合のアクセス許可を定義します。 は、プリンシパルがリクエストを行うときにこれらのポリシー AWS を評価します。ほとんどのポリシーは JSON ドキュメント AWS として に保存されます。JSON ポリシードキュメントの詳細については、「IAM ユーザーガイド」の「JSON ポリシー概要」を参照してください。

管理者は、ポリシーを使用して、どのプリンシパルがどのリソースに対して、どのような条件でアクションを実行できるかを定義することで、誰が何にアクセスできるかを指定します。

デフォルトでは、ユーザーやロールにアクセス許可はありません。IAM 管理者は IAM ポリシーを作成してロールに追加し、このロールをユーザーが引き受けられるようにします。IAM ポリシーは、オペレーションの実行方法を問わず、アクセス許可を定義します。

アイデンティティベースのポリシー

アイデンティティベースのポリシーは、アイデンティティ (ユーザー、グループ、またはロール) にアタッチできる JSON アクセス許可ポリシードキュメントです。これらのポリシーは、アイデンティティがどのリソースに対してどのような条件下でどのようなアクションを実行できるかを制御します。アイデンティティベースポリシーの作成方法については、IAM ユーザーガイド の カスタマー管理ポリシーでカスタム IAM アクセス許可を定義する を参照してください。

アイデンティティベースのポリシーは、インラインポリシー (単一の ID に直接埋め込む) または管理ポリシー (複数の ID にアタッチされたスタンドアロンポリシー) にすることができます。管理ポリシーとインラインポリシーのいずれかを選択する方法については、「IAM ユーザーガイド」の「管理ポリシーとインラインポリシーのいずれかを選択する」を参照してください。

リソースベースのポリシー

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。例としては、IAM ロール信頼ポリシーや Amazon S3 バケットポリシーなどがあります。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスを制御できます。リソースベースのポリシーでは、プリンシパルを指定する必要があります。

リソースベースのポリシーは、そのサービス内にあるインラインポリシーです。リソースベースのポリシーでは、IAM の AWS マネージドポリシーを使用できません。

その他のポリシータイプ

AWS は、より一般的なポリシータイプによって付与されるアクセス許可の上限を設定できる追加のポリシータイプをサポートしています。

- アクセス許可の境界 – アイデンティティベースのポリシーで IAM エンティティに付与することのできるアクセス許可の数の上限を設定します。詳細については、「IAM ユーザーガイド」の「IAM エンティティのアクセス許可境界」を参照してください。

- サービスコントロールポリシー (SCP) - AWS Organizations内の組織または組織単位の最大のアクセス許可を指定します。詳細については、「AWS Organizations ユーザーガイド」の「サービスコントロールポリシー」を参照してください。
- リソースコントロールポリシー (RCP) – は、アカウント内のリソースで利用できる最大数のアクセス許可を定義します。詳細については、「AWS Organizations ユーザーガイド」の「リソースコントロールポリシー (RCP)」を参照してください。
- セッションポリシー – ロールまたはフェデレーションユーザーの一時セッションを作成する際にパラメータとして渡される高度なポリシーです。詳細については、「IAM ユーザーガイド」の「セッションポリシー」を参照してください。

複数のポリシータイプ

1 つのリクエストに複数のタイプのポリシーが適用されると、結果として作成されるアクセス許可を理解するのがさらに難しくなります。が複数のポリシータイプが関与する場合にリクエストを許可するかどうか AWS を決定する方法については、「IAM ユーザーガイド」の「ポリシー評価ロジック」を参照してください。

Amazon GuardDuty が IAM で機能する方法

IAM を使用して GuardDutyへのアクセスを管理する前に、GuardDuty で利用できる IAM の機能について学びます。

Amazon GuardDuty で使用できる IAM の機能

IAM 機能	GuardDuty のサポート
アイデンティティベースのポリシー	あり
リソースベースのポリシー	なし
ポリシーアクション	あり
ポリシーリソース	あり
ポリシー条件キー	あり
ACL	なし

IAM 機能	GuardDuty のサポート
ABAC (ポリシー内のタグ)	部分的
一時認証情報	あり
プリンシパルアクセス権限	あり
サービスロール	あり
サービスリンクロール	はい

GuardDuty およびその他の AWS のサービスがほとんどの IAM 機能と連携する方法の概要については、IAM ユーザーガイドのAWS 「IAM と連携する のサービス」を参照してください。

GuardDuty のためのアイデンティティベースのポリシー

アイデンティティベースのポリシーのサポート: あり

アイデンティティベースポリシーは、IAM ユーザー、ユーザーグループ、ロールなど、アイデンティティにアタッチできる JSON 許可ポリシードキュメントです。これらのポリシーは、ユーザーとロールが実行できるアクション、リソース、および条件をコントロールします。アイデンティティベースポリシーの作成方法については、「IAM ユーザーガイド」の「カスタマー管理ポリシーでカスタム IAM アクセス許可を定義する」を参照してください。

IAM アイデンティティベースのポリシーでは、許可または拒否するアクションとリソース、およびアクションを許可または拒否する条件を指定できます。JSON ポリシーで使用できるすべての要素について学ぶには、「IAM ユーザーガイド」の「IAM JSON ポリシーの要素のリファレンス」を参照してください。

GuardDuty のためのアイデンティティベースのポリシーの例

GuardDuty のアイデンティティベースのポリシーの例を表示するには、「Amazon GuardDuty のためのアイデンティティベースのポリシーの例」を参照してください。

GuardDuty 内のリソースベースのポリシー

リソースベースのポリシーのサポート: なし

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーには例として、IAM ロールの信頼ポリシー や Amazon S3 バケットポリシー があげられます。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスをコントロールできます。ポリシーがアタッチされているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーで、プリンシパルを指定する必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーティッドユーザー、または を含めることができます AWS のサービス。

クロスアカウントアクセスを有効にするには、全体のアカウント、または別のアカウントの IAM エンティティを、リソースベースのポリシーのプリンシパルとして指定します。詳細については、IAM ユーザーガイドのIAM でのクロスアカウントリソースアクセスを参照してください。

GuardDuty のポリシーアクション

ポリシーアクションのサポート: あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

JSON ポリシーの `Action` 要素にはポリシー内のアクセスを許可または拒否するために使用できるアクションが記述されます。このアクションは関連付けられたオペレーションを実行するためのアクセス許可を付与するポリシーで使用されます。

GuardDuty アクションのリストを確認するには、「サービス認可リファレンス」の「Actions defined by Amazon GuardDuty」を参照してください。

GuardDuty のポリシーアクションは、アクションの前に次のプレフィックスを使用します。

```
guardduty
```

単一のステートメントで複数のアクションを指定するには、アクションをカンマで区切ります。

```
"Action": [
      "guardduty:action1",
      "guardduty:action2"
         ]
```

GuardDuty のアイデンティティベースのポリシーの例を表示するには、「Amazon GuardDuty のためのアイデンティティベースのポリシーの例」を参照してください。

GuardDuty のポリシーリソース

ポリシーリソースのサポート: あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

`Resource` JSON ポリシー要素はアクションが適用されるオブジェクトを指定します。ベストプラクティスとして、Amazon リソースネーム (ARN) を使用してリソースを指定します。リソースレベルのアクセス許可をサポートしないアクションの場合は、ステートメントがすべてのリソースに適用されることを示すために、ワイルドカード (*) を使用します。

```
"Resource": "*"
```

GuardDuty リソースのタイプとその ARN のリストを確認するには、「サービス認可リファレンス」の「Resources defined by Amazon GuardDuty」を参照してください。どのアクションで各リソースの ARN を指定できるかについて学ぶには、「Amazon GuardDuty で定義されるアクション」を参照してください。

GuardDuty のアイデンティティベースのポリシーの例を表示するには、「Amazon GuardDuty のためのアイデンティティベースのポリシーの例」を参照してください。

GuardDuty のポリシー条件キー

サービス固有のポリシー条件キーのサポート: あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

`Condition` 要素は、定義された基準に基づいてステートメントが実行される時期を指定します。イコールや未満などの条件演算子を使用して条件式を作成して、ポリシーの条件とリクエスト内の値を

一致させることができます。すべての AWS グローバル条件キーを確認するには、「IAM ユーザーガイド」のAWS 「グローバル条件コンテキストキー」を参照してください。

GuardDuty の条件キーのリストを確認するには、「サービス認可リファレンス」の「Condition keys for Amazon GuardDuty」を参照してください。どのアクションおよびリソースと条件キーを使用できるかについては、「Amazon GuardDuty で定義されるアクション」を参照してください。

GuardDuty のアイデンティティベースのポリシーの例を表示するには、「Amazon GuardDuty のためのアイデンティティベースのポリシーの例」を参照してください。

GuardDuty のアクセスコントロールリスト (ACL)

ACL のサポート: なし

アクセスコントロールリスト (ACL) は、どのプリンシパル (アカウントメンバー、ユーザー、またはロール) がリソースにアクセスするためのアクセス許可を持つかを制御します。ACL はリソースベースのポリシーに似ていますが、JSON ポリシードキュメント形式は使用しません。

GuardDuty での属性ベースのアクセス制御 (ABAC)

ABAC (ポリシー内のタグ) のサポート: 一部

属性ベースのアクセス制御 (ABAC) は、タグと呼ばれる属性に基づいてアクセス許可を定義する認可戦略です。IAM エンティティと AWS リソースにタグをアタッチし、プリンシパルのタグがリソースのタグと一致するときにオペレーションを許可するように ABAC ポリシーを設計できます。

タグに基づいてアクセスを管理するには、`aws:ResourceTag/key-name`、`aws:RequestTag/key-name`、または `aws:TagKeys` の条件キーを使用して、ポリシーの条件要素でタグ情報を提供します。

サービスがすべてのリソースタイプに対して 3 つの条件キーすべてをサポートする場合、そのサービスの値はありです。サービスが一部のリソースタイプに対してのみ 3 つの条件キーのすべてをサポートする場合、値は「部分的」になります。

ABAC の詳細については、「IAM ユーザーガイド」の「ABAC 認可でアクセス許可を定義する」を参照してください。ABAC をセットアップする手順を説明するチュートリアルについては、「IAM ユーザーガイド」の「属性ベースのアクセス制御 (ABAC) を使用する」を参照してください。

GuardDuty での一時的な認証情報の使用

一時的な認証情報のサポート: あり

一時的な認証情報は AWS 、リソースへの短期的なアクセスを提供し、フェデレーションまたはスイッチロールの使用時に自動的に作成されます。長期的なアクセスキーを使用する代わりに、一時的な認証情報を動的に生成 AWS することをお勧めします。詳細については、「IAM ユーザーガイド」の「IAM の一時的な認証情報」および「AWS のサービス と IAM との連携」を参照してください。

GuardDuty のクロスサービスプリンシパル許可

転送アクセスセッション (FAS) のサポート: あり

転送アクセスセッション (FAS) は、 を呼び出すプリンシパルのアクセス許可と AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストをリクエストする を使用します。FAS リクエストを行う際のポリシーの詳細については、「転送アクセスセッション」を参照してください。

GuardDuty のサービスロール

サービスロールのサポート: あり

サービスロールとは、サービスがユーザーに代わってアクションを実行するために引き受ける IAM ロールです。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細については、IAM ユーザーガイドの AWS のサービスに許可を委任するロールを作成するを参照してください。

> **Warning**
>
> サービスロールの許可を変更すると、GuardDuty の機能が破損する可能性があります。GuardDuty が指示する場合以外は、サービスロールを編集しないでください。

GuardDuty のためにサービスにリンクされたロール

サービスリンクロールのサポート: あり

サービスにリンクされたロールは、 にリンクされたサービスロールの一種です AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント 、サービスによって所有されます。IAM 管理者は、サービスリンクロールのアクセス許可を表示できますが、編集することはできません。

GuardDuty サービスにリンクされたロールの作成または管理の詳細については、「Amazon GuardDuty でのサービスにリンクされたロールの使用」を参照してください。

サービスにリンクされたロールの作成または管理の詳細については、IAM と提携するAWS のサービス」を参照してください。表の「サービスリンクロール」列に Yes と記載されたサービスを見つけます。サービスにリンクされたロールに関するドキュメントをサービスで表示するには、[はい] リンクを選択します。

Amazon GuardDuty のためのアイデンティティベースのポリシーの例

デフォルトでは、ユーザーおよびロールには、GuardDuty リソースを作成または変更する許可はありません。IAM 管理者は、リソースで必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。

これらのサンプルの JSON ポリシードキュメントを使用して IAM アイデンティティベースのポリシーを作成する方法については、「IAM ユーザーガイド」の「IAM ポリシーを作成する (コンソール)」を参照してください。

GuardDuty が定義するアクションとリソースタイプ (リソースタイプごとの ARN の形式を含む) の詳細については、「サービス認可リファレンス」の「Amazon GuardDuty のアクション、リソース、および条件キー」を参照してください。

トピック

- ポリシーに関するベストプラクティス
- GuardDuty コンソールの使用
- GuardDuty の有効化に必要な許可
- 自分の権限の表示をユーザーに許可する
- GuardDuty への読み取り専用アクセス権を付与するカスタム IAM ポリシー
- GuardDuty の検出結果へのアクセスを拒否する
- GuardDuty のリソースへのアクセスを制限するためのカスタム IAM ポリシーの使用

ポリシーに関するベストプラクティス

ID ベースのポリシーは、あるユーザーがアカウント内で GuardDuty リソースを作成、アクセス、または削除できるどうかを決定します。これらのアクションでは、 AWS アカウントに費用が発生する場合があります。アイデンティティベースポリシーを作成したり編集したりする際には、以下のガイドラインと推奨事項に従ってください:

- AWS 管理ポリシーを開始し、最小特権のアクセス許可に移行する – ユーザーとワークロードにアクセス許可の付与を開始するには、多くの一般的なユースケースにアクセス許可を付与するAWS

管理ポリシーを使用します。これらは で使用できます AWS アカウント。ユースケースに固有の AWS カスタマー管理ポリシーを定義することで、アクセス許可をさらに減らすことをお勧めします。詳細については、IAM ユーザーガイド の AWS マネージドポリシー または ジョブ機能のAWS マネージドポリシー を参照してください。

- 最小特権を適用する – IAM ポリシーでアクセス許可を設定する場合は、タスクの実行に必要な許可のみを付与します。これを行うには、特定の条件下で特定のリソースに対して実行できるアクションを定義します。これは、最小特権アクセス許可とも呼ばれています。IAM を使用して許可を適用する方法の詳細については、IAM ユーザーガイド の IAM でのポリシーとアクセス許可 を参照してください。
- IAM ポリシーで条件を使用してアクセスをさらに制限する - ポリシーに条件を追加して、アクションやリソースへのアクセスを制限できます。たとえば、ポリシー条件を記述して、すべてのリクエストを SSL を使用して送信するように指定できます。条件を使用して、サービスアクションが などの特定の を通じて使用されている場合に AWS のサービス、サービスアクションへのアクセスを許可することもできます CloudFormation。詳細については、IAM ユーザーガイド の IAM JSON ポリシー要素:条件 を参照してください。
- IAM アクセスアナライザー を使用して IAM ポリシーを検証し、安全で機能的な権限を確保する - IAM アクセスアナライザー は、新規および既存のポリシーを検証して、ポリシーが IAM ポリシー言語 (JSON) および IAM のベストプラクティスに準拠するようにします。IAM アクセスアナライザーは 100 を超えるポリシーチェックと実用的な推奨事項を提供し、安全で機能的なポリシーの作成をサポートします。詳細については、IAM ユーザーガイド の IAM Access Analyzer でポリシーを検証する を参照してください。
- 多要素認証 (MFA) を要求する – で IAM ユーザーまたはルートユーザーを必要とするシナリオがある場合は AWS アカウント、MFA をオンにしてセキュリティを強化します。API オペレーションが呼び出されるときに MFA を必須にするには、ポリシーに MFA 条件を追加します。詳細については、IAM ユーザーガイド の MFA を使用した安全な API アクセス を参照してください。

IAM でのベストプラクティスの詳細については、IAM ユーザーガイド の IAM でのセキュリティのベストプラクティス を参照してください。

GuardDuty コンソールの使用

Amazon GuardDuty コンソールにアクセスするには、最小限の許可セットが必要です。これらのアクセス許可により、 の GuardDuty リソースの詳細を一覧表示および表示できます AWS アカウント。最小限必要な許可よりも制限が厳しいアイデンティティベースのポリシーを作成すると、そのポリシーを持つエンティティ (ユーザーまたはロール) に対してコンソールが意図したとおりに機能しません。

AWS CLI または AWS API のみを呼び出すユーザーには、最小限のコンソールアクセス許可を付与する必要はありません。代わりに、実行しようとしている API オペレーションに一致するアクションのみへのアクセスが許可されます。

ユーザーとロールが引き続き GuardDuty コンソールを使用できるようにするには、エンティティに GuardDuty `ConsoleAccess`または `ReadOnly` AWS 管理ポリシーもアタッチします。詳細については、「IAM ユーザーガイド」の「ユーザーへの許可の追加」を参照してください。

GuardDuty の有効化に必要な許可

さまざまな IAM ID (ユーザー、グループ、およびロール) に必要な許可を付与するには、必要な AWS 管理ポリシー: AmazonGuardDutyFullAccess_v2 (推奨) ポリシーをアタッチして GuardDuty を有効にします。

自分の権限の表示をユーザーに許可する

この例では、ユーザーアイデンティティにアタッチされたインラインおよびマネージドポリシーの表示を IAM ユーザーに許可するポリシーの作成方法を示します。このポリシーには、コンソールで、または AWS CLI または AWS API を使用してプログラムでこのアクションを実行するアクセス許可が含まれています。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
```

```
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

GuardDuty への読み取り専用アクセス権を付与するカスタム IAM ポリシー

GuardDuty への読み取り専用アクセス権を付与するには、AmazonGuardDutyReadOnlyAccess マネージドポリシーが使用できます。

IAM ロール、ユーザー、またはグループに GuardDuty への読み込み専用アクセス権を付与するカスタムポリシーを作成するには、次のステートメントを使用します。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "guardduty:ListMembers",
                "guardduty:GetMembers",
                "guardduty:ListInvitations",
                "guardduty:ListDetectors",
                "guardduty:GetDetector",
                "guardduty:ListFindings",
                "guardduty:GetFindings",
                "guardduty:ListIPSets",
                "guardduty:GetIPSet",
                "guardduty:ListThreatIntelSets",
                "guardduty:GetThreatIntelSet",
                "guardduty:GetMasterAccount",
                "guardduty:GetInvitationsCount",
                "guardduty:GetFindingsStatistics",
```

```
                "guardduty:DescribeMalwareScans",
                "guardduty:UpdateMalwareScanSettings",
                "guardduty:GetMalwareScanSettings"
            ],
            "Resource": "*"
        }
    ]
}
```

GuardDuty の検出結果へのアクセスを拒否する

次のポリシーを使用し、IAM ロール、ユーザーまたはグループに対して、GuardDuty の検出結果へのアクセスを拒否できます。ユーザーは検出結果やその詳細は表示できませんが、その他すべての GuardDuty のオペレーションにはアクセスできます。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "guardduty:CreateDetector",
                "guardduty:DeleteDetector",
                "guardduty:UpdateDetector",
                "guardduty:GetDetector",
                "guardduty:ListDetectors",
                "guardduty:CreateIPSet",
                "guardduty:DeleteIPSet",
                "guardduty:UpdateIPSet",
                "guardduty:GetIPSet",
                "guardduty:ListIPSets",
                "guardduty:CreateThreatIntelSet",
                "guardduty:DeleteThreatIntelSet",
                "guardduty:UpdateThreatIntelSet",
                "guardduty:GetThreatIntelSet",
                "guardduty:ListThreatIntelSets",
                "guardduty:ArchiveFindings",
                "guardduty:UnarchiveFindings",
                "guardduty:CreateSampleFindings",
```

```
                "guardduty:CreateMembers",
                "guardduty:InviteMembers",
                "guardduty:GetMembers",
                "guardduty:DeleteMembers",
                "guardduty:DisassociateMembers",
                "guardduty:StartMonitoringMembers",
                "guardduty:StopMonitoringMembers",
                "guardduty:ListMembers",
                "guardduty:GetMasterAccount",
                "guardduty:DisassociateFromMasterAccount",
                "guardduty:AcceptAdministratorInvitation",
                "guardduty:ListInvitations",
                "guardduty:GetInvitationsCount",
                "guardduty:DeclineInvitations",
                "guardduty:DeleteInvitations"
            ],
            "Resource": "*"
        },
         {
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole"
            ],
            "Resource": "arn:aws:iam::123456789012:role/aws-service-role/
guardduty.amazonaws.com/AWSServiceRoleForAmazonGuardDuty",
            "Condition": {
                "StringLike": {
                    "iam:AWSServiceName": "guardduty.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:PutRolePolicy",
                "iam:DeleteRolePolicy"
            ],
            "Resource": "arn:aws:iam::123456789012:role/aws-service-role/
guardduty.amazonaws.com/AWSServiceRoleForAmazonGuardDuty"
        }
    ]
}
```

GuardDuty のリソースへのアクセスを制限するためのカスタム IAM ポリシーの使用

ディテクター ID に基づいて GuardDuty に対するユーザーのアクセスを定義するには、次のオペレーションを除くすべての GuardDuty API のアクションをカスタム IAM ポリシー内で使用できます。

- `guardduty:CreateDetector`
- `guardduty:DeclineInvitations`
- `guardduty:DeleteInvitations`
- `guardduty:GetInvitationsCount`
- `guardduty:ListDetectors`
- `guardduty:ListInvitations`

IAM ポリシーの次のオペレーションを使用し、IPSet ID および ThreatIntelSet ID に基づいた GuardDuty に対するユーザーのアクセス権を定義します。

- `guardduty:DeleteIPSet`
- `guardduty:DeleteThreatIntelSet`
- `guardduty:GetIPSet`
- `guardduty:GetThreatIntelSet`
- `guardduty:UpdateIPSet`
- `guardduty:UpdateThreatIntelSet`

次の例では、前述のオペレーションのいくつかを使用してポリシーを作成する方法を示します。

- このポリシーでは、us-east-1 リージョンでディテクター ID として 1234567 を使用し、`guardduty:UpdateDetector` オペレーションを実行できます。

  JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "guardduty:UpdateDetector"
            ],
```

```
            "Resource": "arn:aws:guardduty:us-
east-1:123456789012:detector/1234567"
        }
    ]
}
```

- このポリシーでは、us-east-1 リージョンでディテクター ID として 1234567、および IPSet ID として 000000 を使用し、guardduty:UpdateIPSet オペレーションを実行できます。

> **Note**
>
> GuardDuty の信頼できる IP のリストと脅威リストにアクセスするために必要な許可が、そのユーザーにあることを確認します。詳細については、「エンティティリストと IP アドレスリストの前提条件の設定」を参照してください。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "guardduty:UpdateIPSet"
            ],
            "Resource": "arn:aws:guardduty:us-
east-1:123456789012:detector/1234567/ipset/000000"
        }
    ]
}
```

- このポリシーでは、us-east-1 リージョンで任意のディテクター ID、および IPSet ID として 000000 を使用し、guardduty:UpdateIPSet オペレーションを実行できます。

> **Note**
>
> GuardDuty の信頼できる IP のリストと脅威リストにアクセスするために必要な許可が、そのユーザーにあることを確認します。詳細については、「エンティティリストと IP アドレスリストの前提条件の設定」を参照してください。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "guardduty:UpdateIPSet"
            ],
            "Resource": "arn:aws:guardduty:us-east-1:123456789012:detector/*/
ipset/000000"
        }
    ]
}
```

- このポリシーでは、us-east-1 リージョンでディテクター ID および任意の IPSet ID を使用し、guardduty:UpdateIPSet オペレーションを実行できます。

> **Note**
>
> GuardDuty の信頼できる IP のリストと脅威リストにアクセスするために必要な許可が、そのユーザーにあることを確認します。詳細については、「エンティティリストと IP アドレスリストの前提条件の設定」を参照してください。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "guardduty:UpdateIPSet"
            ],
            "Resource": "arn:aws:guardduty:us-
east-1:123456789012:detector/1234567/ipset/*"
        }
    ]
```

```
}
```

GuardDuty アクションの IAM リソース ARN は、そのアクションの API リソースパスと一致します。ほとんどのアクションは、ディテクターまたはサブリソースにマッピングされます。たとえば、guardduty:GetDetectorは を使用しdetector/detectorId、 は guardduty:GetFilterを使用しますdetector/detectorId/filter/filterName。

ほとんどの検出結果とメンバーアクションは、リソースパスのサフィックスとしてオペレーション名を追加します。リストオペレーションは例外です。 guardduty:ListFindings および は、サフィックスなしでベースfindingsとmemberパスguardduty:ListMembersを使用します。これらのアクションにポリシーをスコープする場合は、フルパスを含めます。

- guardduty:GetFindings は を使用します detector/detectorId/findings/get
- guardduty:GetFindingsStatistics は を使用します detector/detectorId/findings/statistics
- guardduty:ArchiveFindings は を使用します detector/detectorId/findings/archive
- guardduty:GetMembers は を使用します detector/detectorId/member/get
- guardduty:ListFindings を使用する detector/detectorId/findings (サフィックスなし)
- guardduty:ListMembers を使用する detector/detectorId/member (サフィックスなし)

次のポリシーは、完全なリソースパスを使用して、us-east-1 リージョンの特定のディテクターguardduty:GetFindingsに を付与します。

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": "guardduty:GetFindings",
    "Resource": "arn:aws:guardduty:us-
east-1:111122223333:detector/a1b2c3d4e5f6g7h8i9j0k1l2m3n4o5p6/findings/get"
  }]
}
```

Amazon GuardDuty でのサービスにリンクされたロールの使用

Amazon GuardDuty は AWS Identity and Access Management (IAM) サービスにリンクされたロールを使用します。サービスにリンクされたロール (SLR) は、直接 GuardDuty にリンクされた IAM ロールの一意のタイプです。サービスにリンクされたロールは GuardDuty によって事前定義されており、GuardDuty がユーザーに代わって他の AWS サービスを呼び出すために必要なすべてのアクセス許可が含まれています。

サービスにリンクされたロールを使用すると、必要な許可を手動で追加せずに GuardDuty を設定できます。GuardDuty は、サービスにリンクされたロールの許可を定義します。特に定義された許可が無い限り、GuardDuty のみがそのロールを引き受けることができます。定義される許可には、信頼ポリシーや許可ポリシーなどがあり、その許可ポリシーを他の IAM エンティティにアタッチすることはできません。

GuardDuty は、 GuardDuty が利用できるすべてのリージョンで、サービスにリンクされたロールの使用をサポートします。詳細については、「リージョンとエンドポイント」を参照してください。

GuardDuty サービスにリンクされたロールは、それが有効になっているすべてのリージョンで GuardDuty を無効にした後にのみ削除できます。これにより、アクセスに必要な許可の誤った削除を防ぐことができ、GuardDuty リソースが保護されます。

サービスにリンクされたロールをサポートするその他のサービスについては、「IAM ユーザーガイド」の「AWS IAM と連携するサービス」を参照し、「サービスにリンクされたロール」列が「はい」となっているサービスを探してください。そのサービスについて、サービスにリンクされたロールのドキュメントを表示するには、「Yes」を選択します。これにはリンクが設定されています。

GuardDuty のためのサービスにリンクされたロールの許可

GuardDuty は、`AWSServiceRoleForAmazonGuardDuty` と名付けられたサービスにリンクされたロール (SLR) を使用します。SLR により、GuardDuty は次のタスクを実行できます。また、GuardDuty は、潜在的な脅威に関して GuardDuty が生成する可能性のある検出結果に、EC2 インスタンスに属する取得されたメタデータを含めることができます。`AWSServiceRoleForAmazonGuardDuty` サービスにリンクされたロールは、ロールを継承するために `guardduty.amazonaws.com` のサービスを信頼します。

アクセス許可ポリシーは、GuardDuty で次のタスクを実行するのに役立ちます。

- Amazon EC2 アクションを使用して、EC2 インスタンス、イメージ、および VPC、サブネット、トランジットゲートウェイなどのネットワークコンポーネントに関する情報を管理および取得します。

- Amazon EC2 の自動エージェントで GuardDuty Runtime Monitoring を有効にする場合、 AWS Systems Manager アクションを使用して Amazon EC2 インスタンスの SSM 関連付けを管理します。GuardDuty 自動エージェント設定が無効になっている場合、GuardDuty は包含タグ (`GuardDutyManaged:true`) を持つ EC2 インスタンスのみを考慮します。
- AWS Organizations アクションを使用して、関連するアカウントと組織 ID を記述します。
- Amazon S3 アクションを使用して S3 バケットとオブジェクトに関する情報を取得します。
- AWS Lambda アクションを使用して、Lambda 関数とタグに関する情報を取得します。
- Amazon EKS アクションを使用して、EKS クラスターに関する情報を管理および取得し、EKS クラスター上の Amazon EKS アドオンを管理します。EKS アクションは、GuardDuty に関連するタグに関する情報も取得します。
- Malware Protection for EC2 が有効になった後、IAM を使用して、Malware Protection for EC2 のためのサービスにリンクされたロールの許可を作成します。
- Amazon ECS アクションを使用して、Amazon ECS クラスターに関する情報の管理と取得、タスクとタスク定義に関する情報の取得、 を使用した Amazon ECS アカウント設定の管理を行いますguarddutyActivate。Amazon ECS に関するアクションでも、GuardDuty に関連するタグの情報も取得します。
- Amazon VPC Lattice アクションを使用して、Runtime Monitoring エージェントの接続を有効にします。

ロールは、`AmazonGuardDutyServiceRolePolicy` と名付けられた次の AWS マネージドポリシーで構成されます。

このポリシーの許可を確認するには、「AWS マネージドポリシーリファレンス」の「AmazonGuardDutyServiceRolePolicy」を参照してください。

`AWSServiceRoleForAmazonGuardDuty` サービスにリンクされたロールにアタッチされている信頼ポリシーは次のとおりです。

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "guardduty.amazonaws.com"
```

```
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

AmazonGuardDutyServiceRolePolicy ポリシーへの更新詳細については、GuardDuty による AWS 管理ポリシーの更新 を参照してください。このポリシーへの変更に関する自動アラートについては、ドキュメント履歴 ページの RSS フィードを購読してください。

GuardDuty 用のサービスにリンクされたロールの作成

GuardDuty を初めて有効にするか、以前に有効にしていなかったサポート対象リージョン中で GuardDuty を有効にしていなかった場合は、AWSServiceRoleForAmazonGuardDuty のサービスにリンクされたロールが自動的に生成されます。IAM コンソール、、または IAM API を使用して AWS CLI、サービスにリンクされたロールを手動で作成することもできます。

> **Important**
>
> GuardDuty の委任されたアカウント用に作成されたサービスにリンクされたロールは、メンバーの GuardDuty アカウントには適用されません。

サービスにリンクされたロールの作成、編集、削除をIAM プリンシパル (ユーザー、グループ、ロールなど) に許可するには、許可を設定する必要があります。AWSServiceRoleForAmazonGuardDuty サービスにリンクされたロールを正常に作成するには、GuardDuty で使用する IAM プリンシパルに必要な許可が付与されている必要があります。必要なアクセス許可を付与するには、次のポリシーをこの ユーザー、グループ、またはロールにアタッチします。

> **Note**
>
> 次の例のサンプル*##### ID* を実際の AWS アカウント ID に置き換えます。

JSON

```
{
```

```
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "guardduty:*"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:CreateServiceLinkedRole"
            ],
            "Resource": "arn:aws:iam::123456789012:role/aws-service-role/
guardduty.amazonaws.com/AWSServiceRoleForAmazonGuardDuty",
            "Condition": {
                "StringLike": {
                    "iam:AWSServiceName": "guardduty.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:PutRolePolicy",
                "iam:DeleteRolePolicy"
            ],
            "Resource": "arn:aws:iam::123456789012:role/aws-service-role/
guardduty.amazonaws.com/AWSServiceRoleForAmazonGuardDuty"
        }
    ]
}
```

IAM ロールを手動で作成することの詳細については、「IAM ユーザーガイド」の「Creating a service-linked role」を参照してください。

GuardDuty のためにサービスにリンクされたロールの編集

GuardDuty では、AWSServiceRoleForAmazonGuardDuty サービスにリンクされたロールを編集できません。サービスにリンクされたロールを作成した後は、多くのエンティティによってロールが参照される可能性があるため、ロール名を変更することはできません。ただし、IAM を使用した

ロール記述の編集はできます。詳細については、「IAM ユーザーガイド」の「サービスにリンクされたロールの編集」を参照してください。

GuardDuty 用のサービスにリンクされたロールの削除

サービスにリンクされたロールが必要な機能またはサービスが不要になった場合には、そのロールを削除することをお勧めします。これにより、使用していないエンティティがアクティブにモニタリングされたり、メンテナンスされたりすることがなくなります。

> **Important**
>
> Malware Protection for EC2 を有効にしている場合、`AWSServiceRoleForAmazonGuardDuty` を削除しても `AWSServiceRoleForAmazonGuardDutyMalwareProtection` は自動的に削除されません。`AWSServiceRoleForAmazonGuardDutyMalwareProtection` を削除するには、「Malware Protection for EC2 のサービスにリンクされたロールの削除」を参照してください。

`AWSServiceRoleForAmazonGuardDuty` を削除するために有効になっているすべてのリージョンで、最初に GuardDuty を削除する必要があります。サービスにリンクされたロールを削除しようとしたときに GuardDuty サービスが無効になっていない場合、削除は失敗します。詳細については、「GuardDuty の停止または無効化」を参照してください。

GuardDuty を無効にしても、`AWSServiceRoleForAmazonGuardDuty` は自動的に削除されません。GuardDuty を再度有効にする場合、既存の `AWSServiceRoleForAmazonGuardDuty` を使用して起動します。

サービスリンクロールを IAM で手動削除するには

IAM コンソール、 AWS CLI、または IAM API を使用して、`AWSServiceRoleForAmazonGuardDuty`サービスにリンクされたロールを削除します。詳細については、「IAM ユーザーガイド」の「サービスにリンクされたロールの削除」を参照してください。

サポートされている AWS リージョン

Amazon GuardDuty は、GuardDuty AWS リージョン が利用可能なすべての`AWSServiceRoleForAmazonGuardDuty`でサービスにリンクされたロールの使用をサポートして

います。GuardDuty が現在利用可能なリージョンの一覧については、「Amazon Web Services 全般のリファレンス」の「Amazon GuardDuty のエンドポイントとクォータ」を参照してください。

Malware Protection for EC2 のためのサービスにリンクされたロールの許可

Malware Protection for EC2 は、`AWSServiceRoleForAmazonGuardDutyMalwareProtection` というサービスにリンクされたロール (SLR) を使用します。この SLR により、Malware Protection for EC2 はエージェントレススキャンを実行して GuardDuty アカウント内のマルウェアを検出できます。これにより、GuardDuty はアカウントに EBS ボリュームスナップショットを作成し、そのスナップショットを GuardDuty サービスアカウントと共有できます。GuardDuty がスナップショットを評価すると、取得された EC2 インスタンスとコンテナワークロードのメタデータが Malware Protection for EC2 の検出結果に含まれます。`AWSServiceRoleForAmazonGuardDutyMalwareProtection` サービスにリンクされたロールは、ロールを継承するために `malware-protection.guardduty.amazonaws.com` のサービスを信頼します。

このロールのアクセス許可ポリシーは、Malware Protection for EC2 で次のタスクを実行するのに役立ちます。

- Amazon Elastic Compute Cloud (Amazon EC2) アクションを使用して、Amazon EC2 インスタンス、ボリューム、スナップショットに関する情報を取得します。Malware Protection for EC2 は、Amazon EKS と Amazon ECS クラスターのメタデータにアクセスする許可も与えます。
- `GuardDutyExcluded` タグが `true` に設定されていない EBS ボリュームのスナップショットを作成してください。デフォルトでは、`GuardDutyScanId` タグを持つスナップショットが作成されます。このタグを削除しないでください。削除すると、Malware Protection for EC2 がスナップショットにアクセスできなくなります。

> **Important**
>
> `GuardDutyExcluded` を `true` に設定すると、GuardDuty サービスは今後これらのスナップショットにアクセスできなくなります。これは、このサービスにリンクされたロールの他のステートメントにより、`GuardDutyExcluded` が `true` に設定されたスナップショットに対するどのアクションも GuardDuty が実行できなくなるためです。

- スナップショットの共有と削除を許可するのは、`GuardDutyScanId` タグが存在し、`GuardDutyExcluded` タグが `true` に設定されていない場合のみです。

> **Note**
>
> Malware Protection for EC2 がスナップショットを公開することを許可しません。

- カスタマーマネージドキー (`GuardDutyExcluded` タグが `true` に設定されているキーを除く) にアクセスして `CreateGrant` を呼び出し、GuardDuty サービスアカウントと共有される暗号化スナップショットから暗号化 EBS ボリュームを作成してアクセスします。各リージョンの GuardDuty サービスアカウントのリストについては、「による GuardDuty サービスアカウント AWS リージョン」を参照してください。
- 顧客の CloudWatch ログにアクセスして Malware Protection for EC2 ロググループを作成し、マルウェアスキャンイベントログを `/aws/guardduty/malware-scan-events` ロググループの下に配置します。
- マルウェアが検出されたスナップショットを自分のアカウントに保持するかどうかをお客様が決定できるようにします。スキャンでマルウェアが検出された場合、サービスにリンクされたロールにより、GuardDuty はスナップショットに `GuardDutyFindingDetected` および `GuardDutyExcluded` の 2 つのタグが追加できます。

> **Note**
>
> `GuardDutyFindingDetected` タグは、スナップショットにマルウェアが含まれていると指定します。

- ボリュームが EBS マネージドキーで暗号化されているかどうかを判断します。GuardDuty は `DescribeKey` アクションを実行し、アカウントの EBS マネージドキーの `key Id` を特定します。
- を使用して暗号化された EBS ボリュームのスナップショットを から取得 AWS アカウント し AWS マネージドキー、 にコピーしますGuardDuty サービスアカウント。この目的のため、`GetSnapshotBlock` と `ListSnapshotBlocks` のアクセス許可を使用します。その後、GuardDuty はサービスアカウントのスナップショットをスキャンします。現在、 で暗号化された EBS ボリュームのスキャンに対する Malware Protection for EC2 サポートは、すべての で使用できない AWS マネージドキー 場合があります AWS リージョン。詳細については、「リージョン固有機能の可用性」を参照してください。
- Amazon EC2 が Malware Protection for EC2 AWS KMS に代わって を呼び出し、カスタマーマネージドキーに対していくつかの暗号化アクションを実行できるようにします。`kms:ReEncryptTo` や `kms:ReEncryptFrom` のようなアクションは、カスタマーマネージ

ドキーで暗号化されたスナップショットを共有するために必要です。GuardDutyExcluded タグが true に設定されていないキーだけがアクセス可能です。

ロールは、AmazonGuardDutyMalwareProtectionServiceRolePolicy と名付けられた次の AWS マネージドポリシーで構成されます。

このポリシーのアクセス許可を確認するには、「AWS マネージドポリシーリファレンス」の「AmazonGuardDutyMalwareProtectionServiceRolePolicy」を参照してください。

AWSServiceRoleForAmazonGuardDutyMalwareProtection サービスにリンクされたロールにアタッチされている信頼ポリシーは次のとおりです。

JSON

```
{
  "Version":"2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "malware-protection.guardduty.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Malware Protection for EC2 のサービスにリンクされたロールの作成

Malware Protection for EC2 を初めて有効にするか、以前に有効にしていなかったサポート対象リージョンで Malware Protection for EC2 を有効にした場合は、AWSServiceRoleForAmazonGuardDutyMalwareProtection のサービスにリンクされたロールが自動的に生成されます。IAM コンソール、IAM CLI、あるいは IAM API を使って、AWSServiceRoleForAmazonGuardDutyMalwareProtection サービスにリンクされたロールを手動で作成することもできます。

> **Note**
>
> デフォルトでは、Amazon GuardDuty を初めて使用する場合、Malware Protection for EC2 は自動的に有効になります。

> **Important**
>
> 委任 GuardDuty 管理者アカウント用に作成されたサービスにリンクされたロールは、メンバーの GuardDuty アカウントには適用されません。

サービスにリンクされたロールの作成、編集、削除をIAM プリンシパル (ユーザー、グループ、ロールなど) に許可するには、許可を設定する必要があります。AWSServiceRoleForAmazonGuardDutyMalwareProtection サービスにリンクされたロールを正常に作成するには、GuardDuty で使用する IAM ID に必要な許可が付与されている必要があります。必要なアクセス許可を付与するには、次のポリシーをこの ユーザー、グループ、またはロールにアタッチします。

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [{
            "Effect": "Allow",
            "Action": "guardduty:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "iam:AWSServiceName": [
                        "malware-protection.guardduty.amazonaws.com"
                    ]
                }
            }
        },
```

```
        {
            "Effect": "Allow",
            "Action": [
                "organizations:EnableAWSServiceAccess",
                "organizations:RegisterDelegatedAdministrator",
                "organizations:ListDelegatedAdministrators",
                "organizations:ListAWSServiceAccessForOrganization",
                "organizations:DescribeOrganizationalUnit",
                "organizations:DescribeAccount",
                "organizations:DescribeOrganization"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:GetRole",
            "Resource": "arn:aws:iam::*:role/
*AWSServiceRoleForAmazonGuardDutyMalwareProtection"
        }
    ]
}
```

IAM ロールを手動で作成する方法の詳細は、「IAM ユーザーガイド」の「サービスにリンクされたロールを作成する」を参照してください。

Malware Protection for EC2 のサービスにリンクされたロールの編集

Malware Protection for EC2 では、`AWSServiceRoleForAmazonGuardDutyMalwareProtection` サービスにリンクされたロールを編集できません。サービスリンクロールを作成すると、多くのエンティティによってロールが参照される可能性があるため、ロール名を変更することはできません。ただし、IAM を使用したロール記述の編集はできます。詳細については、「IAM ユーザーガイド」の「サービスリンクロールの編集」を参照してください。

Malware Protection for EC2 のサービスにリンクされたロールの削除

サービスリンクロールが必要な機能またはサービスが不要になった場合にはそのロールを削除することをお勧めします。これにより、使用していないエンティティがアクティブにモニタリングされたり、メンテナンスされたりすることがなくなります。

> **Important**
>
> AWSServiceRoleForAmazonGuardDutyMalwareProtection を削除するためには、有効になっているすべてのリージョンで、最初に Malware Protection for EC2 を無効にする必要があります。
> サービスにリンクされたロールを削除しようとしたときに Malware Protection for EC2 サービスが無効になっていない場合、削除は失敗します。アカウントで最初に Malware Protection for EC2 を必ず無効にしてください。

[無効] を選択して Malware Protection for EC2 サービスを停止しても、AWSServiceRoleForAmazonGuardDutyMalwareProtection は自動的に削除されません。[有効] を選択して Malware Protection for EC2 サービスを再び開始すると、GuardDuty は既存の AWSServiceRoleForAmazonGuardDutyMalwareProtection を使用して動作を開始します。

サービスリンクロールを IAM で手動削除するには

IAM コンソール、 CLI、または IAM API AWS を使用して、AWSServiceRoleForAmazonGuardDutyMalwareProtectionサービスにリンクされたロールを削除します。詳細については、「IAM ユーザーガイド」の「サービスにリンクされたロールの削除」を参照してください。

サポートされている AWS リージョン

Amazon GuardDuty は、Malware Protection for EC2 AWS リージョン が利用可能なすべてので、AWSServiceRoleForAmazonGuardDutyMalwareProtectionサービスにリンクされたロールの使用をサポートしています。

GuardDuty が現在利用可能なリージョンの一覧については、「Amazon Web Services 全般のリファレンス」の「Amazon GuardDuty のエンドポイントとクォータ」を参照してください。

AWS Amazon GuardDuty の マネージドポリシー

ユーザー、グループ、ロールにアクセス許可を追加するには、自分でポリシーを記述するよりも AWS 管理ポリシーを使用する方が簡単です。チームに必要な権限のみを提供する IAM カスタマーマネージドポリシーを作成する には時間と専門知識が必要です。すぐに開始するには、 AWS マネージドポリシーを使用できます。これらのポリシーは、一般的なユースケースをターゲット範囲に含めており、 AWS アカウントで利用できます。 AWS 管理ポリシーの詳細については、IAM ユーザーガイドの「 AWS 管理ポリシー」を参照してください。

AWS サービスは、 AWS 管理ポリシーを維持および更新します。 AWS 管理ポリシーのアクセス許可は変更できません。サービスは、新機能をサポートするために、 AWS 管理ポリシーに追加のアクセス許可を追加することがあります。この種類の更新はポリシーがアタッチされている、すべてのアイデンティティ (ユーザー、グループおよびロール) に影響を与えます。サービスは、新機能が起動されたとき、または新しいオペレーションが利用可能になったときに、 AWS マネージドポリシーを更新する可能性が最も高いです。サービスは AWS 管理ポリシーからアクセス許可を削除しないため、ポリシーの更新によって既存のアクセス許可が損なわれることはありません。

さらに、 は、複数のサービスにまたがるジョブ関数の マネージドポリシー AWS をサポートしています。例えば、ReadOnlyAccess AWS 管理ポリシーは、すべての AWS サービスとリソースへの読み取り専用アクセスを提供します。サービスが新機能を起動すると、 は新しいオペレーションとリソースの読み取り専用アクセス許可 AWS を追加します。ジョブ機能のポリシーの一覧および詳細については、「IAM ユーザーガイド」の「AWS のジョブ機能のマネージドポリシー」を参照してください。

`Version` ポリシー要素は、このポリシーを処理するために使用される言語構文ルールを指定します。次のポリシーには、IAM でサポートされている現在のバージョンが含まれています。ポリシー要素の詳細については、「IAM JSON ポリシーエレメント: バージョン」を参照してください。

AWS 管理ポリシー: AmazonGuardDutyFullAccess_v2 (推奨)

AmazonGuardDutyFullAccess_v2 ポリシーを IAM アイデンティティにアタッチできます。このポリシーにより、ユーザーは、すべての GuardDuty アクションを実行し、必要なリソースにアクセスするための完全なアクセス許可を持つことができます。AmazonGuardDutyFullAccess_v2 と AmazonGuardDutyFullAccess の間では、GuardDuty を強化し、管理アクションを GuardDuty サービスプリンシパルに制限するため、AmazonGuardDutyFullAccess_v2 をアタッチすることをお勧めします。

アクセス許可の詳細

AmazonGuardDutyFullAccess_v2 ポリシーには以下のアクセス許可が含まれています。

- `GuardDuty` — すべての GuardDuty アクションに対するフルアクセスをユーザーに許可します。
- `IAM`:
  - GuardDuty サービスリンクロールの作成をユーザーに許可します。
  - GuardDuty の IAM ロールとそのポリシーの表示と管理を許可します。

  - GuardDuty にロールを渡すことをユーザーに許可します。GuardDuty はこのロールを使用して Malware Protection for S3 を有効にし、S3 オブジェクトをスキャンしてマルウェアを検出します。また、GuardDuty はこのロールを使用して Malware Protection for AWS Backup のスキャンを開始します。
  - `AWSServiceRoleForAmazonGuardDutyMalwareProtection` で `iam:GetRole` アクションを実行するアクセス許可によって、Malware Protection for EC2 のサービスリンクロール (SLR) がアカウントに存在するかどうかが明らかになります。
- `Organizations`:
  - ユーザーに GuardDuty の組織構造とアカウントの読み取り (表示) を許可します。
  - GuardDuty Organization の委任された管理者やマネージドメンバーを指定できるようになります。

このポリシーの許可を確認するには、「AWS マネージドポリシーリファレンスガイド」の「AmazonGuardDutyFullAccess_v2」を参照してください。

AWS 管理ポリシー: AmazonGuardDutyFullAccess

`AmazonGuardDutyFullAccess` ポリシーは IAM アイデンティティにアタッチできます。

> **Important**
>
> GuardDuty サービスプリンシパルに対するセキュリティを強化し、アクセス許可を制限するには、AWS 管理ポリシー: AmazonGuardDutyFullAccess_v2 (推奨) を使用することをお勧めします。

このポリシーにより、ユーザーにすべての GuardDuty アクションとリソースを実行するためのフルアクセスを許可する管理者許可を付与します。

アクセス許可の詳細

このポリシーには、次の許可が含まれています。

- `GuardDuty` — すべての GuardDuty アクションに対するフルアクセスをユーザーに許可します。
- `IAM`:

  - GuardDuty サービスリンクロールの作成をユーザーに許可します。
  - 管理者アカウントはメンバーアカウントに対して GuardDuty を有効にできます。
  - GuardDuty にロールを渡すことをユーザーに許可します。GuardDuty はこのロールを使用して Malware Protection for S3 を有効にし、S3 オブジェクトをスキャンしてマルウェアを検出します。また、GuardDuty はこのロールを使用して Malware Protection for AWS Backup のスキャンを開始します。
- `Organizations` — GuardDuty Organization の委任された管理者やマネージドメンバーを指定できるようになります。

`AWSServiceRoleForAmazonGuardDutyMalwareProtection` で `iam:GetRole` アクションを実行するアクセス許可によって、Malware Protection for EC2 のサービスリンクロール (SLR) がアカウントに存在するかどうかが明らかになります。

このポリシーの許可を確認するには、「AWS マネージドポリシーリファレンスガイド」の「AmazonGuardDutyFullAccess」を参照してください。

AWS 管理ポリシー: AmazonGuardDutyReadOnlyAccess

`AmazonGuardDutyReadOnlyAccess` ポリシーは IAM アイデンティティにアタッチできます。

このポリシーは、ユーザーが GuardDuty の検出結果と GuardDuty 組織の詳細を表示できるようにするための読み取り専用アクセスを許可します。

許可の詳細

このポリシーには、次の許可が含まれています。

- `GuardDuty` — ユーザーが GuardDuty の検出結果を表示し、`Get`、`List`、または `Describe` で始まる API オペレーションを実行できるようにします。
- `Organizations` — ユーザーは委任された管理者のアカウントの詳細を含む GuardDuty organization の構成に関する情報を取得できるようになります。

このポリシーの許可を確認するには、「AWS マネージドポリシーリファレンスガイド」の「AmazonGuardDutyReadOnlyAccess」を参照してください。

AWS 管理ポリシー: AmazonGuardDutyServiceRolePolicy

IAM エンティティに AmazonGuardDutyServiceRolePolicy をアタッチすることはできません。この AWS 管理ポリシーは、GuardDuty がユーザーに代わってアクションを実行できるようにするサービスにリンクされたロールにアタッチされます。詳細については、「GuardDuty のためのサービスにリンクされたロールの許可」を参照してください。

GuardDuty による AWS 管理ポリシーの更新

このサービスがこれらの変更の追跡を開始してからの GuardDuty の AWS マネージドポリシーの更新に関する詳細を表示します。このページの変更に関する自動アラートについては、GuardDuty の [Document history] (ドキュメントの履歴) ページの RSS フィードをサブスクライブしてください。

変更	説明	日付
AmazonGuardDutyServiceRolePolicy - 既存のポリシーへの更新	vpc-lattice:CreateServiceNetworkVpcAssociation および アクセスvpc-lattice:AssociateViaAWSService 許可を追加しました。これにより、GuardDuty は Amazon VPC Lattice アクションを使用して Runtime Monitoring エージェントの接続を有効にできます。このGdLatticeCreateManagedSnvaAssociation ステートメントにより、GuardDuty は同じアカウント内のリソースのサービスネットワーク VPC 関連付けを作成できます。GdLatticeAssociateViaAwsService ステートメントにより、GuardDuty は Amazon VPC Lattice を介して関連付けを完了できます。	2026 年 9 月 9 日

変更	説明	日付

	{
	
	"Sid": "GdLatticeCreateManagedSnvaAssociation",
	
	"Effect": "Allow",
	
	"Action": "vpc-lattice:CreateServiceNetworkVpcAssociation",
	
	"Resource": "arn:aws:vpc-lattice:*:*:servicenetworkvpcassociation/*",
	
	"Condition": {
	
	"StringEquals": {
	
	"aws:ResourceAccount": "${aws:PrincipalAccount}"
	
	}
	
	}
	
	},
	
	{
	
	"Sid": "GdLatticeAssociateViaAwsService",	

変更	説明	日付
	"Effect": "Allow", "Action": "vpc-lattice:AssociateViaAWSService", "Resource": "*" }	
AmazonGuardDutyServiceRolePolicy - 既存のポリシーへの更新	S3 バケットのリストを取得するs3:ListBucket アクセス許可と、Amazon ECS タスクecs:DescribeTasks とタスク定義に関する情報を取得するecs:DescribeTaskDefinition アクセス許可を追加しました。	2026 年 4 月 23 日

変更	説明	日付

AmazonGuardDutyServiceRolePolicy - 既存のポリシーへの更新	AWS CloudTrail イベントを消費するための追加のメカニズムを有効にするアクセスcloudtrail:CreateServiceLinkedChannel 許可を追加しました。
	
	{
	
	"Sid": "CloudTrailCreateServiceLinkedChannelSid",
	
	"Effect": "Allow",
	
	"Action": [
	
	"cloudtrail:CreateServiceLinkedChannel"
	
	],
	
	"Resource": "arn:aws:cloudtrail:*:*:channel/aws-service-channel/guardduty/*",
	
	"Condition": {
	
	"StringEquals": {
	
	"aws:ResourceAccount": "${aws:PrincipalAccount}"
		2026 年 3 月 25 日

変更	説明	日付
	`}` `}` `}`	
AmazonGuardDutyFullAccess - 廃止済み	このポリシーは、AmazonGuardDutyFullAccess_v2 という名前のスコープダウンポリシーに置き換えられました。 2026 年 3 月 13 日以降、新しいユーザー、グループ、またはロールにAmazonGuardDutyFullAccess ポリシーをアタッチすることはできません。詳細については、「AWS 管理ポリシー: AmazonGuardDutyFullAccess_v2 (推奨)」を参照してください。	2026 年 3 月 13 日

変更	説明	日付

AmazonGuardDutyFullAccess_v2 – 既存ポリシーへの更新	Malware Protection for AWS Backup を有効にするときに IAM ロールを GuardDuty に渡すことができるアクセス許可を追加しました。
	
	{
	
	"Sid": "AllowPassRoleToMalwareProtection",
	
	"Effect": "Allow",
	
	"Action": [
	
	"iam:PassRole"
	
	],
	
	"Resource": "arn:aws:iam::*:role/*",
	
	"Condition": {
	
	"StringEquals": {
	
	"iam:PassedToService": [
	
	"malware-protection-plan.guardduty.amazonaws.com",
		2025 年 11 月 19 日

変更	説明	日付
	"malware-protection.guardduty.amazonaws.com"] } } }	

変更	説明	日付

AmazonGuardDutyFullAccess – 既存ポリシーへの更新	Malware Protection for AWS Backup を有効にするときに IAM ロールを GuardDuty に渡すことができるアクセス許可を追加しました。
	
	{
	
	"Sid": "AllowPassRoleToMalwareProtection",
	
	"Effect": "Allow",
	
	"Action": [
	
	"iam:PassRole"
	
	],
	
	"Resource": "arn:aws:iam::*:role/*",
	
	"Condition": {
	
	"StringEquals": {
	
	"iam:PassedToService": [
	
	"malware-protection-plan.guardduty.amazonaws.com",
		2025 年 11 月 19 日

変更	説明	日付
	"malware-protection.guardduty.amazonaws.com"] } } }	
AmazonGuardDutyFullAccess_v2 – は新しいポリシーを追加しました。	新しい AmazonGuardDutyFullAccess_v2 ポリシーを追加しました。これは、IAM ロールとポリシー、および AWS Organizations 統合に基づいて管理アクションを GuardDuty サービスプリンシパルに制限することで、アクセス許可によってセキュリティが強化されるため、推奨されます。	2025 年 6 月 4 日
AmazonGuardDutyServiceRolePolicy - 既存のポリシーへの更新	`ec2:DescribeVpcs` アクセス許可を追加しました。これにより、GuardDuty は VPC CIDR を取得するなど VPC の更新を追跡できます。	2024 年 8 月 22 日

変更	説明	日付
AmazonGuardDutyFullAccess – 既存のポリシーの更新	Malware Protection for S3 を有効にするときに IAM ロールを GuardDuty に渡すことができるアクセス許可を追加しました。 `{ "Sid": "AllowPassRoleToMalwareProtectionPlan", "Effect": "Allow", "Action": [ "iam:PassRole" ], "Resource": "arn:aws:iam::*:role/*", "Condition": { "StringEquals": { "iam:PassedToService": "malware-protection-plan.guardduty.amazonaws.com" } } }`	2024 年 6 月 10 日

変更	説明	日付
AmazonGuardDutyServiceRolePolicy - 既存のポリシーへの更新。	Amazon EC2 の自動エージェントで GuardDuty Runtime Monitoring を有効にする場合、 AWS Systems Manager アクションを使用して Amazon EC2 インスタンスの SSM 関連付けを管理します。GuardDuty 自動エージェント設定が無効になっている場合、GuardDuty は包含タグ (`GuardDutyManaged :true`) を持つ EC2 インスタンスのみを考慮します。	2024 年 3 月 26 日
AmazonGuardDutyServiceRolePolicy - 既存のポリシーへの更新。	GuardDuty に新しいアクセス許可 `organization:DescribeOrganization` を追加しました。共有 Amazon VPC アカウントの組織 ID を取得し、組織 ID で Amazon VPC エンドポイントポリシーを設定するのが目的です。	2024 年 2 月 9 日
AmazonGuardDutyMalwareProtectionServiceRolePolicy – 既存のポリシーへの更新。	Malware Protection for EC2 に 2 つのアクセス許可が追加されました。 `GetSnapshotBlock` と `ListSnapshotBlocks` は、 から AWS アカウント EBS ボリュームのスナップショット (を使用して暗号化 AWS マネージドキー) を取得し、マルウェアスキャンを開始する前に GuardDuty サービスアカウントにコピーします。	2024 年 1 月 25 日

変更	説明	日付
AmazonGuardDutyServiceRolePolicy - 既存のポリシーへの更新	新たな許可を追加したことで、GuardDuty による guarddutyActivate Amazon ECS アカウント設定の追加が可能となり、Amazon ECS クラスターでリスト操作と説明操作を実行できるようになりました。	2023 年 11 月 26 日
AmazonGuardDutyReadOnlyAccess – 既存ポリシーへの更新	GuardDuty は新しいポリシー organizations を ListAccounts に追加しました。	2023 年 11 月 16 日
AmazonGuardDutyFullAccess – 既存ポリシーへの更新	GuardDuty は新しいポリシー organizations を ListAccounts に追加しました。	2023 年 11 月 16 日
AmazonGuardDutyServiceRolePolicy – 既存ポリシーへの更新	GuardDuty に、近日公開予定の GuardDuty EKS Runtime Monitoring 機能をサポートする新しい許可を追加されました。	2023 年 3 月 8 日

変更	説明	日付
AmazonGuardDutyServiceRolePolicy - 既存のポリシーへの更新	Malware Protection for EC2 のサービスリンクロールを作成できるようにする新しい許可を GuardDuty に追加しました。これにより、GuardDuty は Malware Protection for EC2 を有効にするプロセスを効率化できます。 GuardDuty は次の IAM アクションを実行できるようになりました。 `{ "Effect": "Allow", "Action": "iam:CreateServiceLinkedRole", "Resource": "*", "Condition": { "StringEquals": { "iam:AWSServiceName": "malware-protection.guardduty.amazonaws.com" } } }`	2023 年 2 月 21 日
AmazonGuardDutyFullAccess – 既存ポリシーへの更新	GuardDuty が `iam:GetRole` の ARN を `*AWSServiceRoleForAmazonGuardDutyMalwareProtection` に更新しました。	2022 年 7 月 26 日

変更	説明	日付
AmazonGuardDutyFullAccess – 既存ポリシーへの更新	GuardDuty に新しい `AWSServiceName` を追加しました。GuardDuty Malware Protection for EC2 サービスの `iam:CreateServiceLinkedRole` を使用して、サービスリンクロールを作成できるようにするのが目的です。 GuardDuty で、`AWSServiceRole` の情報を得るための `iam:GetRole` アクションを実行できるようになりました。	2022 年 7 月 26 日

変更	説明	日付
AmazonGuardDutyServiceRolePolicy – 既存ポリシーへの更新	GuardDuty は、GuardDuty が Amazon EC2 ネットワークアクションを使用して検出結果を改善できるようにするための新しい許可を追加しました。 GuardDuty は次の EC2 アクションを実行して、EC2 インスタンスの通信方法に関する情報を取得できるようになりました。この情報は、検出結果の精度を向上させるために使用されます。 • `ec2:DescribeVpcEndpoints` • `ec2:DescribeSubnets` • `ec2:DescribeVpcPeeringConnections` • `ec2:DescribeTransitGatewayAttachments`	2021 年 8 月 3 日
GuardDuty が変更のトラッキングを開始しました	GuardDuty は AWS 、管理ポリシーの変更の追跡を開始しました。	2021 年 8 月 3 日

Amazon GuardDuty アイデンティティとアクセスのトラブルシューティング

次の情報は、GuardDuty と IAM の使用に伴って生じる可能性のある一般的な問題の診断や修復に役立ちます。

トピック

- GuardDuty でアクションを実行する権限がありません。
- iam:PassRole を実行する権限がない
- 自分の 以外のユーザーに GuardDuty リソース AWS アカウント へのアクセスを許可したい。

GuardDuty でアクションを実行する権限がありません。

アクションを実行する権限がないというエラーが表示された場合は、そのアクションを実行できるようにポリシーを更新する必要があります。

次のエラー例は、mateojackson IAM ユーザーがコンソールを使用して、ある *my-example-widget* リソースに関する詳細情報を表示しようとしたことを想定して、その際に必要な guardduty:*GetWidget* アクセス許可を持っていない場合に発生するものです。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 guardduty:GetWidget on resource: my-example-widget
```

この場合、guardduty:*GetWidget* アクションを使用して *my-example-widget* リソースへのアクセスを許可するように、mateojackson ユーザーのポリシーを更新する必要があります。

サポートが必要な場合は、 AWS 管理者にお問い合わせください。サインイン資格情報を提供した担当者が管理者です。

iam:PassRole を実行する権限がない

iam:PassRole アクションを実行する権限がないというエラーが表示された場合は、ポリシーを更新して GuardDuty にロールを渡すことができるようにする必要があります。

一部の AWS のサービス では、新しいサービスロールまたはサービスにリンクされたロールを作成する代わりに、既存のロールをそのサービスに渡すことができます。そのためには、サービスにロールを渡すアクセス許可が必要です。

次の例のエラーは、 GuardDuty で marymajor という名前の IAM ユーザーがコンソールを使用するためにアクションを実行しようする場合に発生します。ただし、このアクションをサービスが実行するには、サービスロールから付与されたアクセス許可が必要です。Mary には、ロールをサービスに渡すアクセス許可がありません。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

この場合、Mary のポリシーを更新してメアリーに `iam:PassRole` アクションの実行を許可する必要があります。

サポートが必要な場合は、 AWS 管理者にお問い合わせください。サインイン資格情報を提供した担当者が管理者です。

自分の 以外のユーザーに GuardDuty リソース AWS アカウント へのアクセスを許可したい。

他のアカウントのユーザーや組織外の人が、リソースにアクセスするために使用できるロールを作成できます。ロールの引き受けを委託するユーザーを指定できます。リソースベースのポリシーまたはアクセスコントロールリスト (ACL) をサポートするサービスの場合、それらのポリシーを使用して、リソースへのアクセスを付与できます。

詳細については、次を参照してください。

- GuardDuty がこれらの機能をサポートしているかどうかを確認するには、「Amazon GuardDuty が IAM で機能する方法」を参照してください。
- 所有 AWS アカウント している のリソースへのアクセスを提供する方法については、「IAM ユーザーガイド」の「所有 AWS アカウント している別の の IAM ユーザーへのアクセスを提供する」を参照してください。
- リソースへのアクセスをサードパーティーに提供する方法については AWS アカウント、IAM ユーザーガイドの「サードパーティー AWS アカウント が所有する へのアクセスを提供する」を参照してください。
- ID フェデレーションを介してアクセスを提供する方法については、IAM ユーザーガイド の 外部で認証されたユーザー (ID フェデレーション) へのアクセスの許可 を参照してください。
- クロスアカウントアクセスにおけるロールとリソースベースのポリシーの使用方法の違いについては、「IAM ユーザーガイド」の「IAM でのクロスアカウントのリソースへのアクセス」を参照してください。

Amazon GuardDuty のコンプライアンス検証

新しい AWS サインアップエクスペリエンスは、規制されたワークロード向けには設計されていません。新しい AWS サインアップエクスペリエンスを使用しているが、規制対象のワークロード AWS に を使用する場合は、 AWS (アドバンスド) にサインアップするか、 AWS 環境の高度な機能をアクティブ化できます。

AWS のサービス が特定のコンプライアンスプログラムの範囲内にあるかどうかを確認するには、「コンプライアンスAWS のサービス プログラムによるスコープ」の「コンプライアンス」を参照して、関心のあるコンプライアンスプログラムを選択します。一般的な情報については、AWS「コンプライアンスプログラム」を参照してください。

を使用して、サードパーティーの監査レポートをダウンロードできます AWS Artifact。詳細については、「Downloading Reports in AWS Artifact」を参照してください。

を使用する際のお客様のコンプライアンス責任 AWS のサービス は、お客様のデータの機密性、貴社のコンプライアンス目的、適用される法律および規制によって決まります。を使用する際のコンプライアンス責任の詳細については AWS のサービス、AWS「 セキュリティドキュメント」を参照してください。

Amazon GuardDuty の回復力

AWS グローバルインフラストラクチャは、 AWS リージョンとアベイラビリティーゾーンを中心に構築されています。リージョンには、低レイテンシー、高いスループット、そして高度の冗長ネットワークで接続されている複数の物理的に独立および隔離されたアベイラビリティーゾーンがあります。アベイラビリティーゾーンでは、ゾーン間で中断することなく自動的にフェイルオーバーするアプリケーションとデータベースを設計および運用することができます。アベイラビリティーゾーンは、従来の単一または複数のデータセンターインフラストラクチャよりも可用性、フォールトトレランス、および拡張性が優れています。

AWS リージョンとアベイラビリティーゾーンの詳細については、AWS「 グローバルインフラストラクチャ」を参照してください。

Amazon GuardDuty でのインフラストラクチャセキュリティ

マネージドサービスである Amazon GuardDuty は、 AWS グローバルネットワークセキュリティで保護されています。 AWS セキュリティサービスと がインフラストラクチャ AWS を保護する方法については、AWS「 クラウドセキュリティ」を参照してください。インフラストラクチャセキュリティのベストプラクティスを使用して環境を AWS 設計するには、「Security Pillar AWS Well-Architected Framework」の「Infrastructure Protection」を参照してください。

AWS 公開された API コールを使用して、ネットワーク経由で GuardDuty にアクセスします。クライアントは次をサポートする必要があります。

- Transport Layer Security (TLS)。TLS 1.2 が必須で、TLS 1.3 をお勧めします。

- DHE (楕円ディフィー・ヘルマン鍵共有) や ECDHE (楕円曲線ディフィー・ヘルマン鍵共有) などの完全前方秘匿性 (PFS) による暗号スイート。これらのモードは、Java 7 以降など、最近のほとんどのシステムでサポートされています。

Amazon GuardDuty とインターフェイス VPC エンドポイント (AWS PrivateLink)

VPC と Amazon GuardDuty とのプライベート接続を確立するには、インターフェイス VPC エンドポイントを作成します。インターフェイスエンドポイントは、AWS PrivateLink を利用しており、インターネットゲートウェイ、NAT デバイス、VPN 接続、 AWS Direct Connect 接続のいずれも使用することなく、GuardDuty API へのプライベートアクセスを可能にする技術です。VPC のインスタンスは、パブリック IP アドレスがなくても GuardDuty API と通信できます。VPC と GuardDuty 間のトラフィックは、Amazon ネットワークを離れません。

各インターフェイスエンドポイントは、サブネット内の 1 つ以上の Elastic Network Interface によって表されます。

詳細については、「AWS PrivateLink ガイド」の「インターフェイス VPC エンドポイント (AWS PrivateLink)」を参照してください。

GuardDuty VPC エンドポイントに関する考慮事項

GuardDuty 用の VPC エンドポイントを設定する前に、「AWS PrivateLink ガイド」の「インターフェイスエンドポイントのプロパティと制限」を確認してください。

GuardDuty は、VPC からのすべての API アクションの呼び出しをサポートしています。

GuardDuty 用のインターフェイス VPC エンドポイントの作成

Amazon VPC コンソールまたは AWS Command Line Interface () を使用して、GuardDuty サービスの VPC エンドポイントを作成できますAWS CLI。詳細については、「AWS PrivateLink ガイド」の「インターフェイスエンドポイントを作成」を参照してください。

GuardDuty 用の VPC エンドポイントを作成する場合は、以下のサービス名を使用します。

- com.amazonaws.*region*.guardduty
- com.amazonaws.*region*.guardduty-fips (FIPS エンドポイント)

エンドポイントのプライベート DNS を有効にすると、リージョンのデフォルト DNS 名 (`guardduty.us-east-1.amazonaws.com` など) を使用して、GuardDuty への API リクエストを実行できます。

詳細については、AWS PrivateLink ガイドのインターフェイスエンドポイントを介したサービスへのアクセスを参照してください。

GuardDuty 用の VPC エンドポイントポリシーの作成

VPC エンドポイントには、GuardDuty へのアクセスを制御するエンドポイントポリシーをアタッチできます。このポリシーでは、以下の情報を指定します。

- アクションを実行できるプリンシパル。
- 実行可能なアクション。
- このアクションを実行できるリソース。

詳細については、「AWS PrivateLink ガイド」の「Control access to services with VPC endpoints」を参照してください。

例: GuardDuty アクション用の VPC エンドポイントポリシー

以下は、GuardDuty 用のエンドポイントポリシーの例です。エンドポイントにアタッチされると、このポリシーは、すべてのリソースですべてのプリンシパルに、リストされている GuardDuty アクションへのアクセス権を付与します。

```
{
   "Statement":[
      {
         "Principal":"*",
         "Effect":"Allow",
         "Action":[
            "guardduty:listDetectors",
            "guardduty:getDetector",
            "guardduty:getFindings"
         ],
         "Resource":"*"
      }
   ]
}
```

共有サブネット

自分と共有されているサブネットで VPC エンドポイントを作成、説明、変更、または削除することはできません。ただし、自分と共有されているサブネットでVPC エンドポイントを使用することはできます。VPC 共有の詳細については、「Amazon VPC ユーザーガイド」の「VPC を他のアカウントと共有する」を参照してください。

GuardDuty と AWS セキュリティサービスとの統合

GuardDuty は他の AWS セキュリティサービスと統合できます。これらのサービスは GuardDuty からデータを取り込み、新しい方法で検出結果を表示できるようにします。次の統合オプションを参照して、そのサービスが GuardDuty でどのように動作するように設定されているかについての詳細を確認してください。

GuardDuty と の統合 AWS Security Hub CSPM

AWS Security Hub CSPM は、 AWS アカウント、サービス、およびサポートされているサードパーティーパートナー製品全体からセキュリティデータを収集し、業界標準とベストプラクティスに従って環境のセキュリティ状態を評価します。セキュリティ体制の評価に加えて、Security Hub CSPM は、統合されたすべての AWS サービスと AWS パートナー製品にわたる検出結果のための一元的な場所を作成します。GuardDuty で Security Hub CSPM を有効にすると、GuardDuty の検出結果データが Security Hub CSPM によって自動的に取り込まれます。

GuardDuty で Security Hub CSPM を使用する方法の詳細については、「」を参照してくださいとの統合 AWS Security Hub CSPM。

GuardDuty と Amazon Detective の統合

Amazon Detective は、 AWS アカウント全体のログデータを使用して、環境とやり取りするリソースと IP アドレスのデータ可視化を作成します。Detective のビジュアライゼーションは、セキュリティ問題をすばやく簡単に調査するのに役立ちます。両方のサービスが有効化されると、GuardDuty の検出結果の詳細から Detective コンソール内の情報にピボットできます。

GuardDuty で Detective を使用する方法については、「Amazon Detective との統合」を参照してください。

との統合 AWS Security Hub CSPM

AWS Security Hub CSPM では、 AWS のセキュリティ状態を総合的に把握することができ、セキュリティ業界標準およびベストプラクティスに照らし合わせて環境をチェックすることができます。Security Hub CSPM は、複数の AWS アカウント、サービス、サポートされているサードパーティーパートナー製品からセキュリティデータを収集し、セキュリティの傾向を分析し、最も優先度の高いセキュリティ問題を特定するのに役立ちます。

Amazon GuardDuty と Security Hub CSPM の統合により、GuardDuty から Security Hub CSPM に検出結果を送信できます。Security Hub CSPM は、これらの検出結果をセキュリティ体制の分析に含めることができます。

目次

- Amazon GuardDuty が検出結果を に送信する方法 AWS Security Hub CSPM
  - GuardDuty が Security Hub CSPM に送信する検出結果のタイプ
    - 新しい検出結果が送信されるまでのレイテンシー
    - Security Hub CSPM が使用できない場合の再試行
    - Security Hub CSPM の既存の検出結果を更新する
- での GuardDuty の検出結果の表示 AWS Security Hub CSPM
  - での GuardDuty の検出結果名の解釈 AWS Security Hub CSPM
  - GuardDuty からの一般的な検出結果
- 統合の有効化と構成
- Security Hub CSPM での GuardDuty コントロールの使用
- 検出結果の Security Hub CSPM への公開の停止

Amazon GuardDuty が検出結果を に送信する方法 AWS Security Hub CSPM

では AWS Security Hub CSPM、セキュリティの問題は検出結果として追跡されます。一部の検出結果は、他の AWS サービスまたはサードパーティーパートナーによって検出された問題から発生します。Security Hub CSPM には、セキュリティの問題を検出し、検出結果を生成するために使用する一連のルールもあります。

Security Hub CSPM には、これらすべてのソースからの検出結果を管理するためのツールが用意されています。検出結果の一覧を表示およびフィルタリングして、検出結果の詳細を表示できます。詳細については、AWS Security Hub ユーザーガイドの「検出結果の表示」を参照してください。検出結果の調査状況を追跡することもできます 詳細については、AWS Security Hub ユーザーガイドの「検出結果に対するアクションの実行」を参照してください。

Security Hub CSPM のすべての検出結果は、 AWS Security Finding Format (ASFF) と呼ばれる標準 JSON 形式を使用します。ASFF には、問題のソース、影響を受けるリソース、および検出結果の現在のステータスに関する詳細が含まれます。 AWS ユーザーガイド の「AWS Security Hub Security Finding 形式 (ASFF)」を参照してください 。

Amazon GuardDuty は、Security Hub CSPM に結果を送信する AWS サービスの 1 つです。

GuardDuty が Security Hub CSPM に送信する検出結果のタイプ

同じアカウント内で同じアカウントで GuardDuty と Security Hub CSPM を有効にすると AWS リージョン、GuardDuty は生成されたすべての検出結果を Security Hub CSPM に送信し始めます。これらの検出結果は、Security AWS Finding 形式 (ASFF) を使用して Security Hub CSPM に送信されます。ASFF では、Types フィールドが検出結果タイプを提供します。

新しい検出結果が送信されるまでのレイテンシー

GuardDuty が新しい検出結果を作成すると、通常は 5 分以内に Security Hub CSPM に送信されます。

Security Hub CSPM が使用できない場合の再試行

Security Hub CSPM が使用できない場合、GuardDuty は結果を受信するまで結果の送信を再試行します。

Security Hub CSPM の既存の検出結果を更新する

Security Hub CSPM に検出結果を送信すると、GuardDuty は検出結果アクティビティの追加の観測を反映する更新を Security Hub CSPM に送信します。これらの検出結果の新しい観測結果は、ステップ 5 – 検出結果をエクスポートする頻度の設定に基づいて Security Hub CSPM に送信されます AWS アカウント。

検出結果をアーカイブまたはアーカイブ解除すると、GuardDuty はその検出結果を Security Hub CSPM に送信しません。GuardDuty で後でアクティブになる手動でアーカイブ解除された検出結果は、Security Hub CSPM に送信されません。

での GuardDuty の検出結果の表示 AWS Security Hub CSPM

にサインイン AWS マネジメントコンソール し、https://console.aws.amazon.com/securityhub/ で AWS Security Hub CSPM コンソールを開きます。

次のいずれかの方法を使用して、Security Hub CSPM コンソールで GuardDuty の検出結果を表示できるようになりました。

オプション 1: Security Hub CSPM で統合を使用する

1. 左側のナビゲーションペインから、[統合] を選択します。
2. [統合] ページで、Amazon: GuardDuty の ステータス を確認します。

- ステータスが [検出結果を受け入る] の場合は、[検出結果を受け入れる] の横にある [検出結果を参照] を選択します。
- そうでない場合は、統合の仕組みの詳細については、AWS Security Hub 「 ユーザーガイド」の「Security Hub CSPM 統合」を参照してください。

オプション 2: Security Hub CSPM での結果の使用

1. 左のナビゲーションペインで [検出結果] を選択します。
2. [検出結果] ページで、[製品名] のフィルターを追加し、**`GuardDuty`** と入力して GuardDuty の検出結果のみを表示します。

での GuardDuty の検出結果名の解釈 AWS Security Hub CSPM

GuardDuty は、Security AWS Finding 形式 (ASFF) を使用して Security Hub CSPM に検出結果を送信します。ASFF では、`Types` フィールドが検出結果タイプを提供します。ASFF タイプは、GuardDuty のタイプとは異なる命名規則を使用します。次の表は、Security Hub CSPM に表示されるすべての GuardDuty 検出結果タイプと ASFF の対応するタイプの詳細を示しています。

> **Note**
>
> 一部の GuardDuty 検出結果タイプでは、Security Hub CSPM は、検出結果の詳細のリソースロールが ACTOR か TARGET かに応じて、異なる ASFF 検出結果名を割り当てます。詳細については、「検出結果の詳細」を参照してください。

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
AttackSequence:IAM/CompromisedCredentials	TTPs/AttackSequence:IAM/CompromisedCredentials
AttackSequence:S3/CompromisedData	TTPs/AttackSequence:S3/CompromisedData
Backdoor:EC2/C&CActivity.B	TTPs/Command and Control/Backdoor:EC2-C&CActivity.B
Backdoor:EC2/C&CActivity.B!DNS	TTPs/Command and Control/Backdoor:EC2-C&CActivity.B!DNS

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
Backdoor:EC2/DenialOfService.Dns	TTPs/Command and Control/Backdoor:EC2-DenialOfService.Dns
Backdoor:EC2/DenialOfService.Tcp	TTPs/Command and Control/Backdoor:EC2-DenialOfService.Tcp
Backdoor:EC2/DenialOfService.Udp	TTPs/Command and Control/Backdoor:EC2-DenialOfService.Udp
Backdoor:EC2/DenialOfService.UdpOnTcpPorts	TTPs/Command and Control/Backdoor:EC2-DenialOfService.UdpOnTcpPorts
Backdoor:EC2/DenialOfService.UnusualProtocol	TTPs/Command and Control/Backdoor:EC2-DenialOfService.UnusualProtocol
Backdoor:EC2/Spambot	TTPs/Command and Control/Backdoor:EC2-Spambot
Behavior:EC2/NetworkPortUnusual	Unusual Behaviors/VM/Behavior:EC2-NetworkPortUnusual
Behavior:EC2/TrafficVolumeUnusual	Unusual Behaviors/VM/Behavior:EC2-TrafficVolumeUnusual
Backdoor:Lambda/C&CActivity.B	TTPs/Command and Control/Backdoor:Lambda-C&CActivity.B
Backdoor:Runtime/C&CActivity.B	TTPs/Command and Control/Backdoor:Runtime-C&CActivity.B
Backdoor:Runtime/C&CActivity.B!DNS	TTPs/Command and Control/Backdoor:Runtime-C&CActivity.B!DNS
CredentialAccess:IAMUser/AnomalousBehavior	TTPs/Credential Access/IAMUser-AnomalousBehavior
CredentialAccess:Kubernetes/AnomalousBehavior.SecretsAccessed	TTPs/AnomalousBehavior/CredentialAccess:Kubernetes-SecretsAccessed

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
CredentialAccess:Kubernetes/MaliciousIPCaller	TTPs/CredentialAccess/CredentialAccess:Kubernetes-MaliciousIPCaller
CredentialAccess:Kubernetes/MaliciousIPCaller.Custom	TTPs/CredentialAccess/CredentialAccess:Kubernetes-MaliciousIPCaller.Custom
CredentialAccess:Kubernetes/SuccessfulAnonymousAccess	TTPs/CredentialAccess/CredentialAccess:Kubernetes-SuccessfulAnonymousAccess
CredentialAccess:Kubernetes/TorIPCaller	TTPs/CredentialAccess/CredentialAccess:Kubernetes-TorIPCaller
CredentialAccess:RDS/AnomalousBehavior.FailedLogin	TTPs/Credential Access/CredentialAccess:RDS-AnomalousBehavior.FailedLogin
CredentialAccess:RDS/AnomalousBehavior.SuccessfulBruteForce	TTPs/Credential Access/CredentialAccess:RDS-AnomalousBehavior.SuccessfulBruteForce
CredentialAccess:RDS/AnomalousBehavior.SuccessfulLogin	TTPs/Credential Access/CredentialAccess:RDS-AnomalousBehavior.SuccessfulLogin
CredentialAccess:RDS/MaliciousIPCaller.FailedLogin	TTPs/Credential Access/CredentialAccess:RDS-MaliciousIPCaller.FailedLogin
CredentialAccess:RDS/MaliciousIPCaller.SuccessfulLogin	TTPs/Credential Access/CredentialAccess:RDS-MaliciousIPCaller.SuccessfulLogin
CredentialAccess:RDS/TorIPCaller.FailedLogin	TTPs/Credential Access/CredentialAccess:RDS-TorIPCaller.FailedLogin
CredentialAccess:RDS/TorIPCaller.SuccessfulLogin	TTPs/Credential Access/CredentialAccess:RDS-TorIPCaller.SuccessfulLogin
CryptoCurrency:EC2/BitcoinTool.B	TTPs/Command and Control/CryptoCurrency:EC2-BitcoinTool.B
CryptoCurrency:EC2/BitcoinTool.B!DNS	TTPs/Command and Control/CryptoCurrency:EC2-BitcoinTool.B!DNS

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
CryptoCurrency:Lambda/BitcoinTool.B	TTPs/Command and Control/CryptoCurrency:Lambda-BitcoinTool.B Effects/Resource Consumption/CryptoCurrency:Lambda-BitcoinTool.B
CommandAndControl:Runtime/SuspiciousCommand	TTPs/Command and Control/CommandAndControl:Runtime-SuspiciousCommand
CredentialAccess:Runtime/SensitiveFileRead	TTPs/Credential Access/CredentialAccess:Runtime-SensitiveFileRead
CryptoCurrency:Runtime/BitcoinTool.B	TTPs/Command and Control/CryptoCurrency:Runtime-BitcoinTool.B
CryptoCurrency:Runtime/BitcoinTool.B!DNS	TTPs/Command and Control/CryptoCurrency:Runtime-BitcoinTool.B!DNS
DefenseEvasion:EC2/UnusualDNSResolver	TTPs/DefenseEvasion/EC2:Unusual-DNS-Resolver
DefenseEvasion:EC2/UnusualDoHActivity	TTPs/DefenseEvasion/EC2:Unusual-DoH-Activity
DefenseEvasion:EC2/UnusualDoTActivity	TTPs/DefenseEvasion/EC2:Unusual-DoT-Activity
DefenseEvasion:IAMUser/AnomalousBehavior	TTPs/Defense Evasion/IAMUser-AnomalousBehavior
DefenseEvasion:IAMUser/BedrockLoggingDisabled	TTPs/Defense Evasion/DefenseEvasion:IAMUser-BedrockLoggingDisabled
DefenseEvasion:Kubernetes/MaliciousIPCaller	TTPs/DefenseEvasion/DefenseEvasion:Kubernetes-MaliciousIPCaller
DefenseEvasion:Kubernetes/MaliciousIPCaller.Custom	TTPs/DefenseEvasion/DefenseEvasion:Kubernetes-MaliciousIPCaller.Custom

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
DefenseEvasion:Kubernetes/SuccessfulAnonymousAccess	TTPs/DefenseEvasion/DefenseEvasion:Kubernetes-SuccessfulAnonymousAccess
DefenseEvasion:Kubernetes/TorIPCaller	TTPs/DefenseEvasion/DefenseEvasion:Kubernetes-TorIPCaller
DefenseEvasion:Runtime/FilelessExecution	TTPs/Defense Evasion/DefenseEvasion:Runtime-FilelessExecution
DefenseEvasion:Runtime/KernelModuleLoaded	TTPs/Defense Evasion/DefenseEvasion:Runtime-KernelModuleLoaded
DefenseEvasion:Runtime/SensitiveFileModified	TTPs/Defense Evasion/DefenseEvasion:Runtime-SensitiveFileModified
DefenseEvasion:Runtime/ProcessInjection.Proc	TTPs/Defense Evasion/DefenseEvasion:Runtime-ProcessInjection.Proc
DefenseEvasion:Runtime/ProcessInjection.Ptrace	TTPs/Defense Evasion/DefenseEvasion:Runtime-ProcessInjection.Ptrace
DefenseEvasion:Runtime/ProcessInjection.VirtualMemoryWrite	TTPs/Defense Evasion/DefenseEvasion:Runtime-ProcessInjection.VirtualMemoryWrite
DefenseEvasion:Runtime/PtraceAntiDebugging	TTPs/DefenseEvasion/DefenseEvasion:Runtime-PtraceAntiDebugging
DefenseEvasion:Runtime/SuspiciousCommand	TTPs/DefenseEvasion/DefenseEvasion:Runtime-SuspiciousCommand
Discovery:IAMUser/AnomalousBehavior	TTPs/Discovery/IAMUser-AnomalousBehavior
Discovery:Kubernetes/AnomalousBehavior.PermissionChecked	TTPs/AnomalousBehavior/Discovery:Kubernetes-PermissionChecked
Discovery:Kubernetes/MaliciousIPCaller	TTPs/Discovery/Discovery:Kubernetes-MaliciousIPCaller

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
Discovery:Kubernetes/MaliciousIPCaller.Custom	TTPs/Discovery/Discovery:Kubernetes-MaliciousIPCaller.Custom
Discovery:Kubernetes/SuccessfulAnonymousAccess	TTPs/Discovery/Discovery:Kubernetes-SuccessfulAnonymousAccess
Discovery:Kubernetes/TorIPCaller	TTPs/Discovery/Discovery:Kubernetes-TorIPCaller
Discovery:RDS/MaliciousIPCaller	TTPs/Discovery/RDS-MaliciousIPCaller
Discovery:RDS/TorIPCaller	TTPs/Discovery/RDS-TorIPCaller
Discovery:Runtime/SensitiveFileRead	TTPs/Discovery/Discovery:Runtime-SensitiveFileRead
Discovery:Runtime/SuspiciousCommand	TTPs/Discovery/Discovery:Runtime-SuspiciousCommand
Discovery:S3/AnomalousBehavior	TTPs/Discovery:S3-AnomalousBehavior
Discovery:S3/BucketEnumeration.Unusual	TTPs/Discovery:S3-BucketEnumeration.Unusual
Discovery:S3/MaliciousIPCaller.Custom	TTPs/Discovery:S3-MaliciousIPCaller.Custom
Discovery:S3/TorIPCaller	TTPs/Discovery:S3-TorIPCaller
Discovery:S3/MaliciousIPCaller	TTPs/Discovery:S3-MaliciousIPCaller
Exfiltration:IAMUser/AnomalousBehavior	TTPs/Exfiltration/IAMUser-AnomalousBehavior
Execution:Kubernetes/ExecInKubeSystemPod	TTPs/Execution/Execution:Kubernetes-ExecInKubeSystemPod
Execution:Kubernetes/AnomalousBehavior.ExecInPod	TTPs/AnomalousBehavior/Execution:Kubernetes-ExecInPod

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
Execution:Kubernetes/AnomalousBehavior.WorkloadDeployed	TTPs/AnomalousBehavior/Execution:Kubernetes-WorkloadDeployed
Impact:EC2/MaliciousDomainRequest.Custom	TTPs/Impact/Impact:EC2-MaliciousDomainRequest.Custom
Impact:Kubernetes/MaliciousIPCaller	TTPs/Impact/Impact:Kubernetes-MaliciousIPCaller
Impact:Kubernetes/MaliciousIPCaller.Custom	TTPs/Impact/Impact:Kubernetes-MaliciousIPCaller.Custom
Impact:Kubernetes/SuccessfulAnonymousAccess	TTPs/Impact/Impact:Kubernetes-SuccessfulAnonymousAccess
Impact:Kubernetes/TorIPCaller	TTPs/Impact/Impact:Kubernetes-TorIPCaller
Persistence:Kubernetes/ContainerWithSensitiveMount	TTPs/Persistence/Persistence:Kubernetes-ContainerWithSensitiveMount
Persistence:Kubernetes/AnomalousBehavior.WorkloadDeployed!ContainerWithSensitiveMount	TTPs/AnomalousBehavior/Persistence:Kubernetes-WorkloadDeployed!ContainerWithSensitiveMount
PrivilegeEscalation:Kubernetes/AnomalousBehavior.WorkloadDeployed!PrivilegedContainer	TTPs/AnomalousBehavior/PrivilegeEscalation:Kubernetes-WorkloadDeployed!PrivilegedContainer
Persistence:Kubernetes/MaliciousIPCaller	TTPs/Persistence/Persistence:Kubernetes-MaliciousIPCaller
Persistence:Kubernetes/MaliciousIPCaller.Custom	TTPs/Persistence/Persistence:Kubernetes-MaliciousIPCaller.Custom
Persistence:Kubernetes/SuccessfulAnonymousAccess	TTPs/Persistence/Persistence:Kubernetes-SuccessfulAnonymousAccess

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
Persistence:Kubernetes/TorIPCaller	TTPs/Persistence/Persistence:Kubernetes-TorIPCaller
Execution:EC2/MaliciousFile	TTPs/Execution/Execution:EC2-MaliciousFile
Execution:ECS/MaliciousFile	TTPs/Execution/Execution:ECS-MaliciousFile
Execution:Kubernetes/MaliciousFile	TTPs/Execution/Execution:Kubernetes-MaliciousFile
Execution:Container/MaliciousFile	TTPs/Execution/Execution:Container-MaliciousFile
Execution:EC2/SuspiciousFile	TTPs/Execution/Execution:EC2-SuspiciousFile
Execution:ECS/SuspiciousFile	TTPs/Execution/Execution:ECS-SuspiciousFile
Execution:Kubernetes/SuspiciousFile	TTPs/Execution/Execution:Kubernetes-SuspiciousFile
Execution:Container/SuspiciousFile	TTPs/Execution/Execution:Container-SuspiciousFile
Execution:EC2/MaliciousFile!Snapshot	TTPs/Execution/Execution:EC2-MaliciousFile!Snapshot
Execution:EC2/MaliciousFile!AMI	TTPs/Execution/Execution:EC2-MaliciousFile!AMI
Execution:EC2/MaliciousFile!RecoveryPoint	TTPs/Execution/Execution:EC2-MaliciousFile!RecoveryPoint
Execution:S3/MaliciousFile!RecoveryPoint	TTPs/Execution/Execution:S3-MaliciousFile!RecoveryPoint
Execution:Runtime/MaliciousFileExecuted	TTPs/Execution/Execution:Runtime-MaliciousFileExecuted

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
Execution:Runtime/MaliciousFileExecuted.Custom	TTPs/Execution/Execution:Runtime-MaliciousFileExecuted.Custom
Execution:Runtime/NewBinaryExecuted	TTPs/Execution/Execution:Runtime-NewBinaryExecuted
Execution:Runtime/NewLibraryLoaded	TTPs/Execution/Execution:Runtime-NewLibraryLoaded
Execution:Runtime/ReverseShell	TTPs/Execution/Execution:Runtime-ReverseShell
Execution:Runtime/SuspiciousCommand	TTPs/Execution/Execution:Runtime-SuspiciousCommand
Execution:Runtime/SuspiciousShellCreated	TTPs/Execution/Execution:Runtime-SuspiciousShellCreated
Execution:Runtime/SuspiciousTool	TTPs/Execution/Execution:Runtime-SuspiciousTool
Exfiltration:S3/AnomalousBehavior	TTPs/Exfiltration:S3-AnomalousBehavior
Exfiltration:S3/ObjectRead.Unusual	TTPs/Exfiltration:S3-ObjectRead.Unusual
Exfiltration:S3/MaliciousIPCaller	TTPs/Exfiltration:S3-MaliciousIPCaller
Impact:EC2/AbusedDomainRequest.Reputation	TTPs/Impact:EC2-AbusedDomainRequest.Reputation
Impact:EC2/BitcoinDomainRequest.Reputation	TTPs/Impact:EC2-BitcoinDomainRequest.Reputation
Impact:EC2/MaliciousDomainRequest.Reputation	TTPs/Impact:EC2-MaliciousDomainRequest.Reputation
Impact:EC2/PortSweep	TTPs/Impact/Impact:EC2-PortSweep

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
Impact:EC2/SuspiciousDomainRequest.Reputation	TTPs/Impact:EC2-SuspiciousDomainRequest.Reputation
Impact:EC2/WinRMBruteForce	TTPs/Impact/Impact:EC2-WinRMBruteForce
Impact:IAMUser/AnomalousBehavior	TTPs/Impact/IAMUser-AnomalousBehavior
Exfiltration:Runtime/SuspiciousCommand	TTPs/Exfiltration/Exfiltration:Runtime-SuspiciousCommand
Impact:Runtime/AbusedDomainRequest.Reputation	TTPs/Impact/Impact:Runtime-AbusedDomainRequest.Reputation
Impact:Runtime/BitcoinDomainRequest.Reputation	TTPs/Impact/Impact:Runtime-BitcoinDomainRequest.Reputation
Impact:Runtime/CryptoMinerExecuted	TTPs/Impact/Impact:Runtime-CryptoMinerExecuted
Impact:Runtime/MaliciousDomainRequest.Reputation	TTPs/Impact/Impact:Runtime-MaliciousDomainRequest.Reputation
Impact:Runtime/SuspiciousDomainRequest.Reputation	TTPs/Impact/Impact:Runtime-SuspiciousDomainRequest.Reputatio
Impact:S3/AnomalousBehavior.Delete	TTPs/Impact:S3-AnomalousBehavior.Delete
Impact:S3/AnomalousBehavior.Permission	TTPs/Impact:S3-AnomalousBehavior.Permission
Impact:S3/AnomalousBehavior.Write	TTPs/Impact:S3-AnomalousBehavior.Write
Impact:S3/ObjectDelete.Unusual	TTPs/Impact:S3-ObjectDelete.Unusual
Impact:S3/PermissionsModification.Unusual	TTPs/Impact:S3-PermissionsModification.Unusual
Impact:S3/MaliciousIPCaller	TTPs/Impact:S3-MaliciousIPCaller

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
InitialAccess:IAMUser/AnomalousBehavior	TTPs/Initial Access/IAMUser-AnomalousBehavior
Object:S3/MaliciousFile	TTPs/Object/Object:S3-MaliciousFile
PenTest:IAMUser/KaliLinux	TTPs/PenTest:IAMUser/KaliLinux
PenTest:IAMUser/ParrotLinux	TTPs/PenTest:IAMUser/ParrotLinux
PenTest:IAMUser/PentooLinux	TTPs/PenTest:IAMUser/PentooLinux
PenTest:S3/KaliLinux	TTPs/PenTest:S3-KaliLinux
PenTest:S3/ParrotLinux	TTPs/PenTest:S3-ParrotLinux
PenTest:S3/PentooLinux	TTPs/PenTest:S3-PentooLinux
Persistence:IAMUser/AnomalousBehavior	TTPs/Persistence/IAMUser-AnomalousBehavior
Persistence:IAMUser/NetworkPermissions	TTPs/Persistence/Persistence:IAMUser-NetworkPermissions
Persistence:IAMUser/ResourcePermissions	TTPs/Persistence/Persistence:IAMUser-ResourcePermissions
Persistence:IAMUser/UserPermissions	TTPs/Persistence/Persistence:IAMUser-UserPermissions
Persistence:Runtime/SuspiciousCommand	TTPs/Persistence/Persistence:Runtime-SuspiciousCommand
Persistence:Runtime/SensitiveFileModified	TTPs/Persistence/Persistence:Runtime-SensitiveFileModified
Policy:IAMUser/RootCredentialUsage	TTPs/Policy:IAMUser-RootCredentialUsage
Policy:IAMUser/ShortTermRootCredentialUsage	TTPs/Policy:IAMUser-ShortTermRootCredentialUsage

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
Policy:Kubernetes/AdminAccessToDefaultServiceAccount	Software and Configuration Checks/AWS Security Best Practices/Policy:Kubernetes-AdminAccessToDefaultServiceAccount
Policy:Kubernetes/AnonymousAccessGranted	Software and Configuration Checks/AWS Security Best Practices/Policy:Kubernetes-AnonymousAccessGranted
Policy:Kubernetes/ExposedDashboard	Software and Configuration Checks/AWS Security Best Practices/Policy:Kubernetes-ExposedDashboard
Policy:Kubernetes/KubeflowDashboardExposed	Software and Configuration Checks/AWS Security Best Practices/Policy:Kubernetes-KubeflowDashboardExposed
Policy:S3/AccountBlockPublicAccessDisabled	TTPs/Policy:S3-AccountBlockPublicAccessDisabled
Policy:S3/BucketAnonymousAccessGranted	TTPs/Policy:S3-BucketAnonymousAccessGranted
Policy:S3/BucketBlockPublicAccessDisabled	Effects/Data Exposure/Policy:S3-BucketBlockPublicAccessDisabled
Policy:S3/BucketPublicAccessGranted	TTPs/Policy:S3-BucketPublicAccessGranted
PrivilegeEscalation:IAMUser/AnomalousBehavior	TTPs/Privilege Escalation/IAMUser-AnomalousBehavior
PrivilegeEscalation:IAMUser/AdministrativePermissions	TTPs/Privilege Escalation/PrivilegeEscalation:IAMUser-AdministrativePermissions
PrivilegeEscalation:Kubernetes/AnomalousBehavior.RoleBindingCreated	TTPs/AnomalousBehavior/PrivilegeEscalation:Kubernetes-RoleBindingCreated

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
PrivilegeEscalation:Kubernetes/AnomalousBehavior.RoleCreated	TTPs/AnomalousBehavior/PrivilegeEscalation:Kubernetes-RoleCreated
PrivilegeEscalation:Kubernetes/PrivilegedContainer	TTPs/PrivilegeEscalation/PrivilegeEscalation:Kubernetes-PrivilegedContainer
PrivilegeEscalation:Runtime/ContainerMountsHostDirectory	TTPs/Privilege Escalation/PrivilegeEscalation:Runtime-ContainerMountsHostDirectory
PrivilegeEscalation:Runtime/CGroupsReleaseAgentModified	TTPs/Privilege Escalation/PrivilegeEscalation:Runtime-CGroupsReleaseAgentModified
PrivilegeEscalation:Runtime/DockerSocketAccessed	TTPs/Privilege Escalation/PrivilegeEscalation:Runtime-DockerSocketAccessed
PrivilegeEscalation:Runtime/ElevationToRoot	TTPs/Privilege Escalation/PrivilegeEscalation:Runtime-ElevationToRoot
PrivilegeEscalation:Runtime/RuncContainerEscape	TTPs/Privilege Escalation/PrivilegeEscalation:Runtime-RuncContainerEscape
PrivilegeEscalation:Runtime/SuspiciousCommand	Software and Configuration Checks/PrivilegeEscalation:Runtime-SuspiciousCommand
PrivilegeEscalation:Runtime/SensitiveFileModified	TTPs/Privilege Escalation/PrivilegeEscalation:Runtime-SensitiveFileModified
PrivilegeEscalation:Runtime/UserfaultfdUsage	TTPs/Privilege Escalation/PrivilegeEscalation:Runtime-UserfaultfdUsage
Recon:EC2/PortProbeEMRUnprotectedPort	TTPs/Discovery/Recon:EC2-PortProbeEMRUnprotectedPort
Recon:EC2/PortProbeUnprotectedPort	TTPs/Discovery/Recon:EC2-PortProbeUnprotectedPort
Recon:EC2/Portscan	TTPs/Discovery/Recon:EC2-Portscan

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
Recon:IAMUser/MaliciousIPCaller	TTPs/Discovery/Recon:IAMUser-MaliciousIPCaller
Recon:IAMUser/MaliciousIPCaller.Custom	TTPs/Discovery/Recon:IAMUser-MaliciousIPCaller.Custom
Recon:IAMUser/NetworkPermissions	TTPs/Discovery/Recon:IAMUser-NetworkPermissions
Recon:IAMUser/ResourcePermissions	TTPs/Discovery/Recon:IAMUser-ResourcePermissions
Recon:IAMUser/TorIPCaller	TTPs/Discovery/Recon:IAMUser-TorIPCaller
Recon:IAMUser/UserPermissions	TTPs/Discovery/Recon:IAMUser-UserPermissions
ResourceConsumption:IAMUser/ComputeResources	Unusual Behaviors/User/ResourceConsumption:IAMUser-ComputeResources
Stealth:IAMUser/CloudTrailLoggingDisabled	TTPs/Defense Evasion/Stealth:IAMUser-CloudTrailLoggingDisabled
Stealth:IAMUser/LoggingConfigurationModified	TTPs/Defense Evasion/Stealth:IAMUser-LoggingConfigurationModified
Stealth:IAMUser/PasswordPolicyChange	TTPs/Defense Evasion/Stealth:IAMUser-PasswordPolicyChange
Stealth:S3/ServerAccessLoggingDisabled	TTPs/Defense Evasion/Stealth:S3-ServerAccessLoggingDisabled
Trojan:EC2/BlackholeTraffic	TTPs/Command and Control/Trojan:EC2-BlackholeTraffic
Trojan:EC2/BlackholeTraffic!DNS	TTPs/Command and Control/Trojan:EC2-BlackholeTraffic!DNS

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
Trojan:EC2/DGADomainRequest.B	TTPs/Command and Control/Trojan:EC2-DGADomainRequest.B
Trojan:EC2/DGADomainRequest.C!DNS	TTPs/Command and Control/Trojan:EC2-DGADomainRequest.C!DNS
Trojan:EC2/DNSDataExfiltration	TTPs/Command and Control/Trojan:EC2-DNSDataExfiltration
Trojan:EC2/DriveBySourceTraffic!DNS	TTPs/Initial Access/Trojan:EC2-DriveBySourceTraffic!DNS
Trojan:EC2/DropPoint	Effects/Data Exfiltration/Trojan:EC2-DropPoint
Trojan:EC2/DropPoint!DNS	Effects/Data Exfiltration/Trojan:EC2-DropPoint!DNS
Trojan:EC2/PhishingDomainRequest!DNS	TTPs/Command and Control/Trojan:EC2-PhishingDomainRequest!DNS
Trojan:Lambda/BlackholeTraffic	TTPs/Command and Control/Trojan:Lambda-BlackholeTraffic
Trojan:Lambda/DropPoint	Effects/Data Exfiltration/Trojan:Lambda-DropPoint
Trojan:Runtime/BlackholeTraffic	TTPs/Command and Control/Trojan:Runtime-BlackholeTraffic
Trojan:Runtime/BlackholeTraffic!DNS	TTPs/Command and Control/Trojan:Runtime-BlackholeTraffic!DNS
Trojan:Runtime/DGADomainRequest.C!DNS	TTPs/Command and Control/Trojan:Runtime-DGADomainRequest.C!DNS
Trojan:Runtime/DriveBySourceTraffic!DNS	TTPs/Initial Access/Trojan:Runtime-DriveBySourceTraffic!DNS

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
Trojan:Runtime/DropPoint	Effects/Data Exfiltration/Trojan:Runtime-DropPoint
Trojan:Runtime/DropPoint!DNS	Effects/Data Exfiltration/Trojan:Runtime-DropPoint!DNS
Trojan:Runtime/PhishingDomainRequest!DNS	TTPs/Command and Control/Trojan:Runtime-PhishingDomainRequest!DNS
UnauthorizedAccess:EC2/MaliciousIPCaller.Custom	TTPs/Command and Control/UnauthorizedAccess:EC2-MaliciousIPCaller.Custom
UnauthorizedAccess:EC2/MetadataDNSRebind	TTPs/UnauthorizedAccess:EC2-MetadataDNSRebind
UnauthorizedAccess:EC2/RDPBruteForce	TTPs/Initial Access/UnauthorizedAccess:EC2-RDPBruteForce
UnauthorizedAccess:EC2/SSHBruteForce	TTPs/Initial Access/UnauthorizedAccess:EC2-SSHBruteForce
UnauthorizedAccess:EC2/TorClient	Effects/Resource Consumption/UnauthorizedAccess:EC2-TorClient
UnauthorizedAccess:EC2/TorRelay	Effects/Resource Consumption/UnauthorizedAccess:EC2-TorRelay
UnauthorizedAccess:IAMUser/ConsoleLogin	Unusual Behaviors/User/UnauthorizedAccess:IAMUser-ConsoleLogin
UnauthorizedAccess:IAMUser/ConsoleLoginSuccess.B	TTPs/UnauthorizedAccess:IAMUser-ConsoleLoginSuccess.B
UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.InsideAWS	Effects/Data Exfiltration/UnauthorizedAccess:IAMUser-InstanceCredentialExfiltration.InsideAWS

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.OutsideAWS	Effects/Data Exfiltration/UnauthorizedAccess:IAMUser-InstanceCredentialExfiltration.OutsideAWS
UnauthorizedAccess:IAMUser/MaliciousIPCaller	TTPs/UnauthorizedAccess:IAMUser-MaliciousIPCaller
UnauthorizedAccess:IAMUser/MaliciousIPCaller.Custom	TTPs/UnauthorizedAccess:IAMUser-MaliciousIPCaller.Custom
UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.InsideAWS	Effects/Data Exfiltration/UnauthorizedAccess:IAMUser-ResourceCredentialExfiltration.InsideAWS
UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.OutsideAWS	Effects/Data Exfiltration/UnauthorizedAccess:IAMUser-ResourceCredentialExfiltration.OutsideAWS
UnauthorizedAccess:IAMUser/TorIPCaller	TTPs/Command and Control/UnauthorizedAccess:IAMUser-TorIPCaller
UnauthorizedAccess:Lambda/MaliciousIPCaller.Custom	TTPs/Command and Control/UnauthorizedAccess:Lambda-MaliciousIPCaller.Custom
UnauthorizedAccess:Lambda/TorClient	Effects/Resource Consumption/UnauthorizedAccess:Lambda-TorClient
UnauthorizedAccess:Lambda/TorRelay	Effects/Resource Consumption/UnauthorizedAccess:Lambda-TorRelay
UnauthorizedAccess:Runtime/MetadataDNSRebind	TTPs/UnauthorizedAccess:Runtime-MetadataDNSRebind
UnauthorizedAccess:Runtime/TorRelay	Effects/Resource Consumption/UnauthorizedAccess:Runtime-TorRelay

GuardDuty の検出結果タイプ	ASFF 検出結果タイプ
UnauthorizedAccess:Runtime/TorClient	Effects/Resource Consumption/UnauthorizedAccess:Runtime-TorClient
UnauthorizedAccess:S3/MaliciousIPCaller.Custom	TTPs/UnauthorizedAccess:S3-MaliciousIPCaller.Custom
UnauthorizedAccess:S3/TorIPCaller	TTPs/UnauthorizedAccess:S3-TorIPCaller

GuardDuty からの一般的な検出結果

GuardDuty は、Security AWS Finding 形式 (ASFF) を使用して Security Hub CSPM に結果を送信します。

次に、GuardDuty からの一般的な検出結果の例を示します。


```
  {
  "SchemaVersion": "2018-10-08",
  "Id": "arn:aws:guardduty:us-east-1:193043430472:detector/
d4b040365221be2b54a6264dc9a4bc64/finding/46ba0ac2845071e23ccdeb2ae03bfdea",
  "ProductArn": "arn:aws:securityhub:us-east-1:product/aws/guardduty",
  "GeneratorId": "arn:aws:guardduty:us-east-1:193043430472:detector/
d4b040365221be2b54a6264dc9a4bc64",
  "AwsAccountId": "193043430472",
  "Types": [
    "TTPs/Initial Access/UnauthorizedAccess:EC2-SSHBruteForce"
  ],
  "FirstObservedAt": "2020-08-22T09:15:57Z",
  "LastObservedAt": "2020-09-30T11:56:49Z",
  "CreatedAt": "2020-08-22T09:34:34.146Z",
  "UpdatedAt": "2020-09-30T12:14:00.206Z",
  "Severity": {
    "Product": 2,
    "Label": "MEDIUM",
    "Normalized": 40
  },
  "Title": "199.241.229.197 is performing SSH brute force attacks against
 i-0c10c2c7863d1a356.",
```

```
  "Description": "199.241.229.197 is performing SSH brute force attacks against
 i-0c10c2c7863d1a356. Brute force attacks are used to gain unauthorized access to your
 instance by guessing the SSH password.",
  "SourceUrl": "https://us-east-1.console.aws.amazon.com/guardduty/home?region=us-
east-1#/findings?macros=current&fId=46ba0ac2845071e23ccdeb2ae03bfdea",
  "ProductFields": {
    "aws/guardduty/service/action/networkConnectionAction/remotePortDetails/portName":
 "Unknown",
    "aws/guardduty/service/archived": "false",
    "aws/guardduty/service/action/networkConnectionAction/remoteIpDetails/organization/
asnOrg": "CENTURYLINK-US-LEGACY-QWEST",
    "aws/guardduty/service/action/networkConnectionAction/remoteIpDetails/geoLocation/
lat": "42.5122",
    "aws/guardduty/service/action/networkConnectionAction/remoteIpDetails/ipAddressV4":
 "199.241.229.197",
    "aws/guardduty/service/action/networkConnectionAction/remoteIpDetails/geoLocation/
lon": "-90.7384",
    "aws/guardduty/service/action/networkConnectionAction/blocked": "false",
    "aws/guardduty/service/action/networkConnectionAction/remotePortDetails/port":
 "46717",
    "aws/guardduty/service/action/networkConnectionAction/remoteIpDetails/country/
countryName": "United States",
    "aws/guardduty/service/serviceName": "guardduty",
    "aws/guardduty/service/evidence": "",
    "aws/guardduty/service/action/networkConnectionAction/localIpDetails/ipAddressV4":
 "172.31.43.6",
    "aws/guardduty/service/detectorId": "d4b040365221be2b54a6264dc9a4bc64",
    "aws/guardduty/service/action/networkConnectionAction/remoteIpDetails/organization/
org": "CenturyLink",
    "aws/guardduty/service/action/networkConnectionAction/connectionDirection":
 "INBOUND",
    "aws/guardduty/service/eventFirstSeen": "2020-08-22T09:15:57Z",
    "aws/guardduty/service/eventLastSeen": "2020-09-30T11:56:49Z",
    "aws/guardduty/service/action/networkConnectionAction/localPortDetails/portName":
 "SSH",
    "aws/guardduty/service/action/actionType": "NETWORK_CONNECTION",
    "aws/guardduty/service/action/networkConnectionAction/remoteIpDetails/city/
cityName": "Dubuque",
    "aws/guardduty/service/additionalInfo": "",
    "aws/guardduty/service/resourceRole": "TARGET",
    "aws/guardduty/service/action/networkConnectionAction/localPortDetails/port": "22",
    "aws/guardduty/service/action/networkConnectionAction/protocol": "TCP",
    "aws/guardduty/service/count": "74",
```

```
    "aws/guardduty/service/action/networkConnectionAction/remoteIpDetails/organization/
asn": "209",
    "aws/guardduty/service/action/networkConnectionAction/remoteIpDetails/organization/
isp": "CenturyLink",
    "aws/securityhub/FindingId": "arn:aws:securityhub:us-east-1::product/aws/guardduty/
arn:aws:guardduty:us-east-1:193043430472:detector/d4b040365221be2b54a6264dc9a4bc64/
finding/46ba0ac2845071e23ccdeb2ae03bfdea",
    "aws/securityhub/ProductName": "GuardDuty",
    "aws/securityhub/CompanyName": "Amazon"
  },
  "Resources": [
    {
      "Type": "AwsEc2Instance",
      "Id": "arn:aws:ec2:us-east-1:193043430472:instance/i-0c10c2c7863d1a356",
      "Partition": "aws",
      "Region": "us-east-1",
      "Tags": {
        "Name": "kubectl"
      },
      "Details": {
        "AwsEc2Instance": {
          "Type": "t2.micro",
          "ImageId": "ami-02354e95b39ca8dec",
          "IpV4Addresses": [
            "18.234.130.16",
            "172.31.43.6"
          ],
          "VpcId": "vpc-a0c2d7c7",
          "SubnetId": "subnet-4975b475",
          "LaunchedAt": "2020-08-03T23:21:57Z"
        }
      }
    }
  ],
  "WorkflowState": "NEW",
  "Workflow": {
    "Status": "NEW"
  },
  "RecordState": "ACTIVE"
}
```

統合の有効化と構成

との統合を使用するには AWS Security Hub CSPM、Security Hub CSPM を有効にする必要があります。Security Hub CSPM を有効にする方法については、「AWS Security Hub ユーザーガイド」の「Setting up Security Hub」(Security Hub の設定) を参照してください。

GuardDuty と Security Hub CSPM の両方を有効にすると、統合は自動的に有効になります。GuardDuty は、直ちに検出結果を Security Hub CSPM に送信し始めます。

Security Hub CSPM での GuardDuty コントロールの使用

AWS Security Hub CSPM はセキュリティコントロールを使用して AWS リソースを評価し、セキュリティ業界標準とベストプラクティスに照らしてコンプライアンスをチェックします。GuardDuty リソースおよび選択されている保護プランに関連するコントロールを使用できます。詳細については、「AWS Security Hub ユーザーガイド」の「Amazon GuardDuty のコントロール」を参照してください。

AWS サービスおよびリソース全体のすべてのコントロールのリストについては、「 AWS Security Hub ユーザーガイド」の「Security Hub CSPM コントロールリファレンス」を参照してください。

検出結果の Security Hub CSPM への公開の停止

Security Hub CSPM への結果の送信を停止するには、Security Hub CSPM コンソールまたは API を使用できます。

AWS Security Hub 「 ユーザーガイド」の「統合からの検出結果フローの無効化と有効化 (コンソール)」または「統合からの検出結果フローの無効化 (Security Hub API、 AWS CLI)」を参照してください。

Amazon Detective との統合

Amazon Detective は、リソースが時間の経過とともにどのように動作し、やり取りするかを表すデータ視覚化を生成することで、1 つ以上の AWS アカウントのセキュリティイベントを迅速に分析および調査するのに役立ちます。Detective は、GuardDuty の検出結果の可視化を作成します。

Detective は、すべての検出結果タイプの検出結果の詳細を取り込み、検出結果に関連するさまざまなエンティティを調査するためにエンティティプロファイルへのアクセスを提供します。エンティティは AWS アカウント、、 アカウント内の AWS リソース、または リソースとやり取

りした外部 IP アドレスです。GuardDuty コンソールは、検出結果のタイプ、IAM ロール、ユーザー、またはロールセッション AWS アカウント、ユーザーエージェント、フェデレーティッドユーザー、Amazon EC2 インスタンス、または IP アドレスに応じて、次のエンティティから Amazon Detective へのピボットをサポートします。

目次

- 統合の有効化
- GuardDuty の検出結果から Amazon Detective へのピボット
- GuardDuty マルチアカウント環境との統合を使用します。

統合の有効化

GuardDuty で Amazon Detective を使用するには、まず Amazon Detective を有効にする必要があります。Detective を有効にする方法については、「Amazon Detective ユーザーガイド」の「Amazon Detective の開始方法」を参照してください。

GuardDuty と Detective の両方を有効にすると、統合は自動的に有効になります。有効にすると、Detective はすぐに GuardDuty の検出結果データを取り込みます。

> **Note**
>
> GuardDuty は、GuardDuty の検出結果エクスポート頻度に基づいて Detective に検出結果を送信します。デフォルトで、既存の検出結果の更新用のエクスポート頻度は 6 時間です。Detective が最新の更新検出結果を受信できるように、GuardDuty で Detective を使用する各リージョンで、エクスポート頻度を 15 分に変更することをお勧めします。詳細については、「ステップ 5 – 更新されたアクティブな検出結果をエクスポートする頻度を設定する」を参照してください。

GuardDuty の検出結果から Amazon Detective へのピボット

1. https://console.aws.amazon.com/guardduty/ コンソールにログインします。
2. 検出結果テーブルから 1 つの検出結果を選択します。
3. 検出結果詳細ペインで、[Investigate with Detective] (Detective で調査する) を選択します。
4. Amazon Detective で検出結果のアスペクトを選択します。これにより、その検出結果または エンティティの Detective コンソールが開きます。

ピボットが正常に動作しない場合は、「Amazon Detective ユーザーガイド」の「ピボットのトラブルシューティング」を参照してください。

> **Note**
>
> Detective コンソールで GuardDuty の検出結果をアーカイブすると、その検出結果は GuardDuty コンソールでもアーカイブされます。

GuardDuty マルチアカウント環境との統合を使用します。

GuardDuty でマルチアカウント環境を管理している場合、アカウントの検出結果とエンティティの Detective データ可視化を表示するために、メンバーアカウントを Amazon Detective に追加する必要があります。

Detective の管理者アカウントと同じ GuardDuty 管理者アカウントを使用することをお勧めします。Detective でメンバーアカウントを追加する方法の詳細については、「Amazon Detective ユーザーガイド」の「アカウントの管理」を参照してください。

> **Note**
>
> Detective はリージョンレベルのサービスなので、Detective を有効にして、統合を使用したいリージョンごとにメンバーアカウントを追加する必要があります。

GuardDuty の停止または無効化

GuardDuty コンソールを使用して GuardDuty のサービスを停止または無効化できます。サービスの一時停止中は、GuardDuty の使用に対して課金されません。

- GuardDuty を停止または無効化するには、すべてのメンバーアカウントの関連付けを解除するか、これらを削除する必要があります。
- GuardDuty を停止すると、 AWS 環境のセキュリティをモニタリングしたり、新しい検出結果を生成したりしなくなります。既存の検出結果は変更されず、GuardDuty の停止によって影響を受けることはありません。GuardDuty は後で再度有効にすることができます。
- アカウントで GuardDuty を無効にした場合、現在選択されている AWS リージョンでのみ無効になります。GuardDuty を完全に無効にする場合は、有効になっている各リージョンで無効にする必要があります。
- GuardDuty を無効にすると、既存の検出結果と GuardDuty の設定は失われ、復旧できなくなります。既存の検出結果を保存する場合は、それをエクスポートしてから GuardDuty を無効にする必要があります。検出結果のエクスポート方法の詳細については、「生成された検出結果を Amazon S3 にエクスポートする」を参照してください。
- アカウント内の 1 つ以上の保護されたバケットで Malware Protection for S3 を有効にしている場合は、GuardDuty を停止または無効にしても、Malware Protection for S3 の保護されたバケットの状態には影響しません。GuardDuty を停止または無効にした後も、アカウントでは引き続き Malware Protection for S3 機能に伴う使用コストが発生します。Malware Protection for S3 の無効化については、「保護されたバケットの Malware Protection for S3 の無効化」を参照してください。

GuardDuty を停止または無効化するには

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。
2. ナビゲーションペインで [設定] を選択します。
3. [Suspend GuardDuty] (GuardDuty の停止) セクションで、[Suspend GuardDuty] (GuardDuty の停止) または [Disable GuardDuty] (GuardDuty の無効化) を選択し、アクションを [Confirm] (確認) します。

停止した後、GuardDuty を再度有効化するには

1. https://console.aws.amazon.com/guardduty/ で GuardDuty コンソールを開きます。

2. ナビゲーションペインで [設定] を選択します。
3. [Re-enable GuardDuty] (GuardDuty の再有効化) を選択します。

Amazon SNS GuardDuty のお知らせへのサブスクライブ

このセクションでは、GuardDuty のお知らせ用 Amazon SNS (Simple Notification Service) をサブスクライブして、新しくリリースされた検出結果タイプ、既存の検出結果タイプの更新、およびその他の機能性の変更に関する通知を受け取る方法について説明します。Amazon SNS がサポートするすべての形式で通知を使用できます。

GuardDuty SNS は、GuardDuty サービスの更新に関する通知 AWS を 経由でサブスクライブしているアカウントに送信します。アカウント内の検出結果に関する通知を受け取るには、「Amazon EventBridge を使用した GuardDuty 検出結果の処理」を参照してください。

> **Note**
>
> SNS をサブスクライブする場合、IAM ユーザーに `sns::subscribe` アクセス許可が必要です。

この通知トピックへの Amazon SQS キューをサブスクライブできますが、同じリージョンのトピックの ARN を使用する必要があります。詳細については、「Amazon Simple Queue Service デベロッパーガイド」の「チュートリアル: Amazon SNS トピックへの Amazon SQS キューのサブスクライブ」を参照してください。

AWS Lambda 関数を使用して、通知を受信したときにイベントをトリガーすることもできます。詳細については、「Amazon Simple Queue Service デベロッパーガイド」の「Amazon SNS 通知を使用した Lambda 関数の呼び出し」を参照してください。

各リージョンの Amazon SNS トピックの ARN は次のとおりです。

AWS リージョン	Amazon SNS トピックの ARN
米国東部 (バージニア北部) - `us-east-1`	`arn:aws:sns:us-east-1:242987662583:GuardDutyAnnouncements`
米国東部 (オハイオ) — `us-east-2`	`arn:aws:sns:us-east-2:118283430703:G`

AWS リージョン	Amazon SNS トピックの ARN
	uardDutyAnnounceme nts
米国西部 (北カリフォルニア) — us-west-1	arn:aws:sns:us-wes t-1:144182107116:G uardDutyAnnounceme nts
米国西部 (オレゴン) - us-west-2	arn:aws:sns:us-wes t-2:934957504740:G uardDutyAnnounceme nts
カナダ (中部) — ca-central-1	arn:aws:sns:ca-cen tral-1:10743005193 3:GuardDutyAnnounc ements
カナダ西部 (カルガリー) — ca-west-1	arn:aws:sns:ca-wes t-1:440427180217:G uardDutyAnnounceme nts
欧州 (ストックホルム) — eu-north-1	arn:aws:sns:eu-nor th-1:973841112453: GuardDutyAnnouncem ents
欧州 (アイルランド) - eu-west-1	arn:aws:sns:eu-wes t-1:965013871422:G uardDutyAnnounceme nts

AWS リージョン	Amazon SNS トピックの ARN
欧州 (ロンドン) — eu-west-2	arn:aws:sns:eu-west-2:506403581195:GuardDutyAnnouncements
欧州 (パリ) – eu-west-3	arn:aws:sns:eu-west-3:436163563069:GuardDutyAnnouncements
欧州 (フランクフルト) — eu-central-1	arn:aws:sns:eu-central-1:378365507264:GuardDutyAnnouncements
欧州 (チューリッヒ) — eu-central-2	arn:aws:sns:eu-central-2:383009515534:GuardDutyAnnouncements
アジアパシフィック (香港) – ap-east-1	arn:aws:sns:ap-east-1:646602203151:GuardDutyAnnouncements
アジアパシフィック (東京) - ap-northeast-1	arn:aws:sns:ap-northeast-1:741172661024:GuardDutyAnnouncements
アジアパシフィック (ソウル) – ap-northeast-2	arn:aws:sns:ap-northeast-2:464168911255:GuardDutyAnnouncements

AWS リージョン	Amazon SNS トピックの ARN
アジアパシフィック (シンガポール) — ap-southeast-1	arn:aws:sns:ap-southeast-1:476419727788:GuardDutyAnnouncements
アジアパシフィック (シドニー) — ap-southeast-2	arn:aws:sns:ap-southeast-2:457615622431:GuardDutyAnnouncements
アジアパシフィック (ムンバイ) – ap-south-1	arn:aws:sns:ap-south-1:926826061926:GuardDutyAnnouncements
南米 (サンパウロ) – sa-east-1	arn:aws:sns:sa-east-1:955633302743:GuardDutyAnnouncements
AWS GovCloud (US-West) - us-gov-west-1	arn:aws-us-gov:sns:us-gov-west-1:430639793359:GuardDutyAnnouncements
中国 (北京) – cn-north-1	arn:aws-cn:sns:cn-north-1:002991280229:GuardDutyAnnouncements
中国 (寧夏) – cn-northwest-1	arn:aws-cn:sns:cn-northwest-1:003033775354:GuardDutyAnnouncements

AWS リージョン	Amazon SNS トピックの ARN
中東 (バーレーン) — me-south-1	arn:aws:sns:me-south-1:552740612889:GuardDutyAnnouncements
中東 (アラブ首長国連邦) – me-central-1	arn:aws:sns:me-central-1:030935290150:GuardDutyAnnouncements
欧州 (ミラノ) — eu-south-1	arn:aws:sns:eu-south-1:188461706213:GuardDutyAnnouncements
欧州 (スペイン) — eu-south-2	arn:aws:sns:eu-south-2:445632894446:GuardDutyAnnouncements
AWS GovCloud (US-East) - us-gov-east-1	arn:aws:sns:us-gov-east-1:143972945659:GuardDutyAnnouncements
アジアパシフィック (大阪) - ap-northeast-3	arn:aws:sns:ap-northeast-3:129086577509:GuardDutyAnnouncements
アジアパシフィック (ジャカルタ): — ap-southeast-3	arn:aws:sns:ap-southeast-3:225965583551:GuardDutyAnnouncements

AWS リージョン	Amazon SNS トピックの ARN
アジアパシフィック (ハイデラバード) — ap-south-2	arn:aws:sns:ap-south-2:595653072700:GuardDutyAnnouncements
アジアパシフィック (メルボルン) — ap-southeast-4	arn:aws:sns:ap-southeast-4:529900636122:GuardDutyAnnouncements
アジアパシフィック (マレーシア) — ap-southeast-5	arn:aws:sns:ap-southeast-5:343218181797:GuardDutyAnnouncements
イスラエル (テルアビブ) — il-central-1	arn:aws:sns:il-central-1:847886274986:GuardDutyAnnouncements
アジアパシフィック (タイ) - ap-southeast-7	arn:aws:sns:ap-southeast-7:863518448376:GuardDutyAnnouncements
メキシコ (中部) - mx-central-1	arn:aws:sns:mx-central-1:060795916546:GuardDutyAnnouncements
アジアパシフィック (台北) - ap-east-2	arn:aws:sns:ap-east-2:604225987917:GuardDutyAnnouncements

で GuardDuty 更新通知 E メールをサブスクライブするには AWS マネジメントコンソール

1. https://console.aws.amazon.com/sns/v3/home で Amazon SNS コンソールを開きます。
2. リージョンのリストで、サブスクライブするトピックの ARN として同じリージョンを選択します。この例では、us-west-2 リージョンを使用します。
3. 左のナビゲーションペインで、[Subscriptions] (サブスクリプション)、[Create subscription] (サブスクリプションの作成) の順に選択します。
4. [Create Subscription] (サブスクリプションの作成) ダイアログボックスの [Topic ARN] (トピック ARN) で、トピック ARN `arn:aws:sns:us-west-2:934957504740:GuardDutyAnnouncements` を貼り付けます。
5. [Protocol] (プロトコル) で [Email] (E メール) を選択します。[Endpoint] (エンドポイント) で、通知を受信するために使用できる E メールアドレスを入力します。
6. [Create subscription] (サブスクリプションの作成) を選択します。
7. E メールアプリケーションで、 AWS 通知からのメッセージを開き、リンクを開いてサブスクリプションを確認します。

   ウェブブラウザに Amazon SNS の確認画面が表示されます。

を使用して GuardDuty 更新通知 E メールをサブスクライブするには AWS CLI

1. AWS CLIを使用して次のコマンドを実行します。

   ```
   aws  sns --region us-west-2 subscribe --topic-arn arn:aws:sns:us-west-2:934957504740:GuardDutyAnnouncements --protocol email --notification-endpoint your_email@your_domain.com
   ```

2. E メールアプリケーションで、 AWS 通知からのメッセージを開き、リンクを開いてサブスクリプションを確認します。

   ウェブブラウザに Amazon SNS の確認画面が表示されます。

Amazon SNS メッセージ形式

GuardDuty の一般的な通知メッセージの例:

```
{
    "Type" : "Notification",
```

```
    "MessageId" : "9101dc6b-726f-4df0-8646-ec2f94e674bc",
    "TopicArn" : "arn:aws:sns:us-west-2:934957504740:GuardDutyAnnouncements",
    "Message" : "{\"version\":\"1\",\"type\":\"GENERAL\",\"message\":[{\"title\":
\"Updated AmazonGuardDutyFullAccess_v2 policy\",\"body\":\"Added permission that
 allows you to pass an IAM role to GuardDuty when you enable Malware Protection for
 S3.\",\"links\":[\"https://docs.aws.amazon.com//guardduty/latest/ug/security-iam-
awsmanpol.html#security-iam-awsmanpol-AmazonGuardDutyFullAccess-v2\"]}]}",
    "Timestamp" : "2018-03-09T00:25:43.483Z",
    "SignatureVersion" : "1",
    "Signature" : "XWox8GDGLRiCgDOXlo/
fG9Lu/88P8S0FL6M6oQYOmUFzkucuhoblsdea3BjqdCHcWR7qdhMPQnLpN7y9iBrWVUqdAGJrukAI8athvAS
+4AQD/V/QjrhsEnlj+GaiW
+ozAu006X6GopOzFGnCtPMROjCMrMonjz7Hpv/8KRuMZR3pyQYm5d4wWB7xBPYhUMuLoZ1V8YFs55FMtgQV/
YLhSYuEu0BP1GMtLQauxDkscOtPP/vjhGQLFx1Q9LTadcQiRHtNIBxWL87PSI
+BVvkin6AL7PhksvdQ7FAgHfXsit+6p8GyOvKCqaeBG7HZhR1AbpyVka7JSNRO/6ssyrlj1g==",
    "SigningCertURL" : "https://sns.us-west-2.amazonaws.com/
SimpleNotificationService-433026a4050d206028891664da859041.pem",
    "UnsubscribeURL" : "https://sns.us-west-2.amazonaws.com/?
Action=Unsubscribe&SubscriptionArn=arn:aws:sns:us-
west-2:934957504740:GuardDutyAnnouncements:9225ed2b-7228-4665-8a01-c8a5db6859f4"
}
```

解析された [Message] (メッセージ) の値 (エスケープした引用符は削除) は次のように表示されます。

```
{
        "version": "1",
        "type": "GENERAL",
        "message": [
            {
                "title": "Updated AmazonGuardDutyFullAccess policy",
                "body": "Added permission that allows you to pass an IAM role to
 GuardDuty when you enable Malware Protection for S3.",
                "links": [
                    "https://docs.aws.amazon.com//guardduty/latest/ug/security-iam-
awsmanpol.html#security-iam-awsmanpol-AmazonGuardDutyFullAccess-v2"
                ]
             }
        ]
}
```

新しい検出結果に関する GuardDuty 更新通知メッセージの例を以下に示します。

```
{
    "Type" : "Notification",
    "MessageId" : "9101dc6b-726f-4df0-8646-ec2f94e674bc",
    "TopicArn" : "arn:aws:sns:us-west-2:934957504740:GuardDutyAnnouncements",
    "Message" : "{\"version\":\"1\",\"type\":\"NEW_FINDINGS\",\"findingDetails
\":[{\"link\":\"https://docs.aws.amazon.com//guardduty/latest/ug/
guardduty_unauthorized.html\",\"findingType\":\"UnauthorizedAccess:EC2/TorClient\",
\"findingDescription\":\"This finding informs you that an EC2 instance in your AWS
 environment is making connections to a Tor Guard or an Authority node. Tor is software
 for enabling anonymous communication. Tor Guards and Authority nodes act as initial
 gateways into a Tor network. This traffic can indicate that this EC2 instance is
 acting as a client on a Tor network. A common use for a Tor client is to circumvent
 network monitoring and filter for access to unauthorized or illicit content. Tor
 clients can also generate nefarious Internet traffic, including attacking SSH servers.
 This activity can indicate that your EC2 instance is compromised.\"}]}",
    "Timestamp" : "2018-03-09T00:25:43.483Z",
    "SignatureVersion" : "1",
    "Signature" : "XWox8GDGLRiCgDOXlo/
fG9Lu/88P8S0FL6M6oQYOmUFzkucuhoblsdea3BjqdCHcWR7qdhMPQnLpN7y9iBrWVUqdAGJrukAI8athvAS
+4AQD/V/QjrhsEnlj+GaiW
+ozAu006X6GopOzFGnCtPMROjCMrMonjz7Hpv/8KRuMZR3pyQYm5d4wWB7xBPYhUMuLoZ1V8YFs55FMtgQV/
YLhSYuEu0BP1GMtLQauxDkscOtPP/vjhGQLFx1Q9LTadcQiRHtNIBxWL87PSI
+BVvkin6AL7PhksvdQ7FAgHfXsit+6p8GyOvKCqaeBG7HZhR1AbpyVka7JSNRO/6ssyrlj1g==",
    "SigningCertURL" : "https://sns.us-west-2.amazonaws.com/
SimpleNotificationService-433026a4050d206028891664da859041.pem",
    "UnsubscribeURL" : "https://sns.us-west-2.amazonaws.com/?
Action=Unsubscribe&SubscriptionArn=arn:aws:sns:us-
west-2:934957504740:GuardDutyAnnouncements:9225ed2b-7228-4665-8a01-c8a5db6859f4"
}
```


解析された [Message] (メッセージ) の値 (エスケープした引用符は削除) は次のように表示されます。


```
{
    "version": "1",
    "type": "NEW_FINDINGS",
    "findingDetails": [{
        "link": "https://docs.aws.amazon.com//guardduty/latest/ug/
guardduty_unauthorized.html",
        "findingType": "UnauthorizedAccess:EC2/TorClient",
        "findingDescription": "This finding informs you that an EC2 instance in your
 AWS environment is making connections to a Tor Guard or an Authority node. Tor is
 software for enabling anonymous communication. Tor Guards and Authority nodes act as
```

```
 initial gateways into a Tor network. This traffic can indicate that this EC2 instance
 is acting as a client on a Tor network. A common use for a Tor client is to circumvent
 network monitoring and filter for access to unauthorized or illicit content. Tor
 clients can also generate nefarious Internet traffic, including attacking SSH servers.
 This activity can indicate that your EC2 instance is compromised."
    }]
}
```

GuardDuty の機能更新に関する GuardDuty 更新通知メッセージの例を、以下に示します。

```
{
    "Type" : "Notification",
    "MessageId" : "9101dc6b-726f-4df0-8646-ec2f94e674bc",
    "TopicArn" : "arn:aws:sns:us-west-2:934957504740:GuardDutyAnnouncements",
    "Message" : "{\"version\":\"1\",\"type\":\"NEW_FEATURES\",\"featureDetails
\":[{\"featureDescription\":\"Customers with high-volumes of global CloudTrail
 events should see a net positive impact on their GuardDuty costs.\",\"featureLink
\":\"https://docs.aws.amazon.com//guardduty/latest/ug/guardduty_data-
sources.html#guardduty_controlplane\"}]}",
    "Timestamp" : "2018-03-09T00:25:43.483Z",
    "SignatureVersion" : "1",
    "Signature" : "XWox8GDGLRiCgDOXlo/
fG9Lu/88P8S0FL6M6oQYOmUFzkucuhoblsdea3BjqdCHcWR7qdhMPQnLpN7y9iBrWVUqdAGJrukAI8athvAS
+4AQD/V/QjrhsEnlj+GaiW
+ozAu006X6GopOzFGnCtPMROjCMrMonjz7Hpv/8KRuMZR3pyQYm5d4wWB7xBPYhUMuLoZ1V8YFs55FMtgQV/
YLhSYuEu0BP1GMtLQauxDkscOtPP/vjhGQLFx1Q9LTadcQiRHtNIBxWL87PSI
+BVvkin6AL7PhksvdQ7FAgHfXsit+6p8GyOvKCqaeBG7HZhR1AbpyVka7JSNRO/6ssyrlj1g==",
    "SigningCertURL" : "https://sns.us-west-2.amazonaws.com/
SimpleNotificationService-433026a4050d206028891664da859041.pem",
    "UnsubscribeURL" : "https://sns.us-west-2.amazonaws.com/?
Action=Unsubscribe&SubscriptionArn=arn:aws:sns:us-
west-2:934957504740:GuardDutyAnnouncements:9225ed2b-7228-4665-8a01-c8a5db6859f4"
}
```

解析された [Message] (メッセージ) の値 (エスケープした引用符は削除) は次のように表示されます。

```
{
    "version": "1",
    "type": "NEW_FEATURES",
    "featureDetails": [{
        "featureDescription": "Customers with high-volumes of global CloudTrail events
 should see a net positive impact on their GuardDuty costs.",
```

```
        "featureLink": "https://docs.aws.amazon.com//guardduty/latest/ug/
guardduty_data-sources.html#guardduty_controlplane"
    }]
}
```

更新された検出結果に関する GuardDuty 更新通知メッセージの例を以下に示します。

```
{
    "Type": "Notification",
    "MessageId": "9101dc6b-726f-4df0-8646-ec2f94e674bc",
    "TopicArn": "arn:aws:sns:us-west-2:934957504740:GuardDutyAnnouncements",
    "Message": "{\"version\":\"1\",\"type\":\"UPDATED_FINDINGS\",
\"findingDetails\":[{\"link\":\"https://docs.aws.amazon.com//guardduty/latest/ug/
guardduty_unauthorized.html\",\"findingType\":\"UnauthorizedAccess:EC2/TorClient\",
\"description\":\"Increased severity value from 5 to 8.\"}]}",
    "Timestamp": "2018-03-09T00:25:43.483Z",
    "SignatureVersion": "1",
    "Signature": "XWox8GDGLRiCgDOXlo/
fG9Lu/88P8S0FL6M6oQYOmUFzkucuhoblsdea3BjqdCHcWR7qdhMPQnLpN7y9iBrWVUqdAGJrukAI8athvAS
+4AQD/V/QjrhsEnlj+GaiW
+ozAu006X6GopOzFGnCtPMROjCMrMonjz7Hpv/8KRuMZR3pyQYm5d4wWB7xBPYhUMuLoZ1V8YFs55FMtgQV/
YLhSYuEu0BP1GMtLQauxDkscOtPP/vjhGQLFx1Q9LTadcQiRHtNIBxWL87PSI
+BVvkin6AL7PhksvdQ7FAgHfXsit+6p8GyOvKCqaeBG7HZhR1AbpyVka7JSNRO/6ssyrlj1g==",
    "SigningCertURL": "https://sns.us-west-2.amazonaws.com/
SimpleNotificationService-433026a4050d206028891664da859041.pem",
    "UnsubscribeURL": "https://sns.us-west-2.amazonaws.com/?
Action=Unsubscribe&SubscriptionArn=arn:aws:sns:us-
west-2:934957504740:GuardDutyAnnouncements:9225ed2b-7228-4665-8a01-c8a5db6859f4"
}
```

解析された [Message] (メッセージ) の値 (エスケープした引用符は削除) は次のように表示されます。

```
{
    "version": "1",
    "type": "UPDATED_FINDINGS",
    "findingDetails": [{
        "link": "https://docs.aws.amazon.com//guardduty/latest/ug/
guardduty_unauthorized.html",
        "findingType": "UnauthorizedAccess:EC2/TorClient",
        "description": "Increased severity value from 5 to 8."
    }]
```

```
}
```

Amazon GuardDuty クォータ

AWS アカウント には、各 の制限と呼ばれるデフォルトのクォータがあります AWS のサービス。特に明記していない限り、クォータはリージョン固有です。一部のクォータの増加を要求できますが、他のクォータは増加できません。

GuardDuty のクォータを表示するには、Service Quotas コンソールを開きます。ナビゲーションペインで、[AWS のサービス] を選択し、[Amazon GuardDuty] を選択します。

クォータの引き上げをリクエストするには、「Service Quotas ユーザーガイド」の「クォータ引き上げリクエスト を参照してください。

AWS アカウント には、リージョンごとの Amazon GuardDuty の次のクォータがあります。

> **Note**
>
> - GuardDuty Malware Protection for EC2 専用のクォータについては、「Malware Protection for EC2 のクォータ」を参照してください。
> - Malware Protection for S3 専用のクォータについては、「Malware Protection for S3 のクォータ」を参照してください。

リージョンごとの GuardDuty クォータ

[リソース]	デフォルト	コメント
ディテクター	1	各リージョンの AWS アカウントごとに作成できるディテクターリソースの最大数。 クォータの引き上げはリクエストできません。
フィルター	100	リージョンごとの AWS アカウントあた

[リソース]	デフォルト	コメント
		りの保存済みフィルターの最大数。 クォータの引き上げはリクエストできません。
検出結果の保持期間	90 日間	検出結果を保持する最大日数。 クォータの引き上げはリクエストできません。
信頼できる IP アドレスリストあたりの IP アドレスと CIDR 範囲	2,000	1 つの信頼できる IP アドレスリストに含めることができる IP アドレスと CIDR 範囲の最大数。 クォータの引き上げはリクエストできません。
脅威アドレスリストあたりの IP アドレスと CIDR 範囲	250,000	脅威 IP アドレスリストに含めることができる IP アドレスと CIDR 範囲の最大数。 クォータの引き上げはリクエストできません。

[リソース]	デフォルト	コメント
脅威エンティティリストごとのエンティティ	1,000	1 つの脅威エンティティリストに含めることができるエンティティの最大数。 クォータの引き上げはリクエストできません。
信頼できるエンティティリストごとのエンティティ	1,000	信頼できるエンティティリストに含めることができるエンティティの最大数。 クォータの引き上げはリクエストできません。
最大ファイルサイズ	35 MB	エンティティリストまたは IP アドレスリストのアップロードに使用されるファイルの最大ファイルサイズ。 クォータの引き上げはリクエストできません。

[リソース]	デフォルト	コメント
メンバーアカウント数 (招待による)	5000	管理者アカウントに関連付けられたメンバーアカウントの最大数。 クォータの引き上げはリクエストできません。

[リソース]	デフォルト	コメント
メンバーアカウント	50,000	AWS Organizations を介して管理者アカウントに関連付けられたメンバーアカウントの最大数。これには、招待によって組織に追加されたメンバーアカウントが含まれます。 このデフォルト値は、 AWS Organizationsのメンバーアカウントの現在のクォータによって異なります。を通じて追加された GuardDuty のメンバーアカウントの数は AWS Organizations 、組織内のメンバーアカウントの数を超えることはできません。組織 AWS アカウント 内の の数については、AWS Organizations 「 ユーザーガイド」の「最大値と最小値」を参照してください。

[リソース]	デフォルト	コメント
脅威インテリジェンスセット (IP アドレスリスト)	6	各リージョンの AWS アカウント ごとに追加できる脅威 IP アドレスの最大数。 クォータの引き上げはリクエストできません。
信頼できる IP セット (IP アドレスリスト)	1	リージョン AWS アカウント ごとに ごとにアップロードおよびアクティブ化できる信頼された IP アドレスリストの最大数。 クォータの引き上げはリクエストできません。
脅威エンティティリスト	6	各リージョンの AWS アカウント アカウントごとに追加できる脅威エンティティリストの最大数。 クォータの引き上げはリクエストできません。

[リソース]	デフォルト	コメント
信頼できるエンティティリスト	1	リージョン AWS アカウント ごとに ごとにアップロードおよびアクティブ化できる信頼されたエンティティリストの最大数。 クォータの引き上げはリクエストできません。

GuardDuty マネージドインスタンスのサポート

次の表は、GuardDuty のさまざまな機能がさまざまなタイプのマネージドインスタンスに対して持つサポートを示しています。

マネージド特徴量タイプ	Runtime Monitoring のサポート	Malware Protection for EC2 のサポート
Amazon EKS 自動モード	サポート対象	サポート
Amazon ECS AWS Fargate マネージドインスタンス	サポートされていません	サポート
Lambda マネージドインスタンス	サポートされていません	サポートされていません

Amazon GuardDuty のトラブルシューティング

GuardDuty 固有のアクション実行に関する問題が発生した場合は、このセクションのトピックを参照してください。

トピック

- Amazon S3 への検出結果のエクスポート - アクセスエラー
- Malware Protection for EC2 の問題
- Runtime Monitoring の問題
- その他の問題のトラブルシューティング

Amazon S3 への検出結果のエクスポート - アクセスエラー

GuardDuty の検出結果を Amazon S3 バケット (発行先) にエクスポートするときに、GuardDuty がこの発行先にアクセスできない場合、アクセスエラーが返されることがあります。

検出結果のエクスポートについて設定した後、GuardDuty が検出結果をエクスポートできない場合、GuardDuty コンソールの [設定] ページにエラーメッセージが表示されます。これが起きるのは、GuardDuty がターゲットリソースにアクセスできなくなったときなどです。例えば、Amazon S3 バケットが削除された場合や、バケットにアクセスするためのアクセス許可が変更された場合です。これは、GuardDuty が Amazon S3 バケット内のデータの暗号化に使用された AWS KMS キーにアクセスできなくなった場合にも発生する可能性があります。エクスポートできない場合、GuardDuty はアカウントに関連付けられている E メールに通知を送信して、この問題に関する情報を提供します。

アクセスエラーを解決する方法とは

この問題を解決するには、対応するリソースが存在し、必要なリソースにアクセスするためのアクセス許可が GuardDuty に付与されていることを確認します。

詳細については、「生成された検出結果を Amazon S3 にエクスポートする」を参照してください。

このエラーを解決しないとどうなるか

GuardDuty では検出結果の保持期間が 90 日で終了します。その前に問題を解決しないと、検出結果はエクスポートされません。問題が起きているリージョンのこのアカウントで検出結果のエクスポート設定が無効になります。

検出結果のエクスポートを再開するには、そのリージョンの構成設定を更新します。

Malware Protection for EC2 の問題

このセクションでは、Malware Protection for EC2 をセットアップまたは使用しているときに発生しうるエラーについて説明します。

GuardDuty 実行型マルウェアスキャンを有効にするときに必要な AWS Organizations 管理アクセス許可がない

を使用して複数のアカウントを管理し AWS Organizations 、このエラー が表示される場合`The request failed because you do not have the required AWS Organizations management account permission.`、組織内の複数のアカウントに対して GuardDuty 実行型マルウェアスキャンを有効にするアクセス許可がありません。

管理アカウントにアクセス許可を付与する方法については、「GuardDuty 実行型マルウェアスキャンを有効にするための信頼されたアクセスの確立」を参照してください。

オンデマンドのマルウェアスキャンを開始しようとしていますが、必要な許可がないというエラーが表示されます。

Amazon EC2 インスタンスでオンデマンドマルウェアスキャンを開始するのに必要な許可がないことを示すエラーが表示された場合は、AWS 管理ポリシー: AmazonGuardDutyFullAccess_v2 (推奨) ポリシーを IAM ロールにアタッチしていることを確認してください。

AWS 組織のメンバーでも同じエラーが表示される場合は、管理アカウントに接続します。詳細については、「AWS Organizations SCP – アクセス拒否」を参照してください。

Malware Protection for EC2 を使用しているときに **`iam:GetRole`** エラーが表示されます。

このエラー `Unable to get role: AWSServiceRoleForAmazonGuardDutyMalwareProtection` が表示される場合は、GuardDuty 実行型マルウェアスキャンを有効にするか、オンデマンドマルウェアスキャンを使用する許可がないことを意味します。AWS 管理ポリシー: AmazonGuardDutyFullAccess_v2 (推奨) ポリシーが IAM ロールにアタッチされていることを確認します。

私は、GuardDuty 実行型マルウェアスキャンを有効にする必要がある GuardDuty 管理者アカウントですが、GuardDuty の管理に AWS マネージドポリシー AmazonGuardDutyFullAccess を使用しません。

- GuardDuty で使用する IAM ロールに、GuardDuty 実行型マルウェアスキャンを有効にするために必要な許可を設定します。必要なアクセス許可の詳細については、「Creating a service-linked role for Malware Protection for EC2」を参照してください。
- AWS 管理ポリシー: AmazonGuardDutyFullAccess_v2 (推奨) を IAM ロールにアタッチします。これにより、メンバーアカウントに対して GuardDuty 実行型マルウェアスキャンを有効にできます。

Runtime Monitoring の問題

このセクションでは、Runtime Monitoring をセットアップまたは使用しているときに発生しうるエラーについて説明します。

ランタイムカバレッジの問題

保護されたリソースのランタイムカバレッジが [異常] になると、GuardDuty コンソールにその問題の正確なタイプが表示されます。問題のタイプを確認したら、次のドキュメントを使用して、サポートされているリソースタイプごとにトラブルシューティング手順を表示します。

- Amazon EC2 ランタイムカバレッジの問題のトラブルシューティング
- Amazon ECS-Fargate ランタイムカバレッジの問題のトラブルシューティング
- Amazon EKS ランタイムカバレッジの問題のトラブルシューティング

Runtime Monitoring のメモリ不足エラーのトラブルシューティング (Amazon EC2 サポートのみ)

このセクションでは、GuardDuty セキュリティエージェントを手動でデプロイしようとして、「CPU とメモリの制限」に説明するようにメモリ不足エラーが発生したときのトラブルシューティング手順を示します。

`out-of-memory` の問題が原因で GuardDuty `systemd` エージェントが終了し、GuardDuty エージェントにより多くのメモリを提供することが妥当であると判断した場合は、制限を更新できます。

1. root 権限で、/lib/systemd/system/amazon-guardduty-agent.service を開きます。
2. MemoryLimit と MemoryMaxを検索し、両方の値を更新します。

```
MemoryLimit=256M
MemoryMax=256M
```

3. 値を更新したら、以下のコマンドを使用して GuardDuty エージェントを再起動します。

```
sudo systemctl daemon-reload
sudo systemctl restart amazon-guardduty-agent
```

4. 以下のコマンドを実行して、ステータスを表示します。

```
sudo systemctl status amazon-guardduty-agent
```

予想される出力には、新しいメモリ制限が表示されます。

```
Main PID: 2540 (amazon-guardduty)
Tasks: 16
Memory: 21.9M (limit: 256.0M)
```

AWS Step Functions ワークフローが予期せず失敗している

GuardDuty コンテナが原因でワークフローが失敗する場合は、「Amazon ECS-Fargate ランタイムカバレッジの問題のトラブルシューティング」を参照してください。問題が解決しない場合は、GuardDuty コンテナが原因でワークフローが失敗しないように、次のいずれかの手順を実行します。

- 関連付けられた Amazon ECS クラスターに GuardDutyManaged:false タグを追加します。
- (AWS Fargate ECS のみ) の自動エージェント設定をアカウントレベルで無効にします。関連付けられた Amazon ECS クラスターに、今後も GuardDuty 自動エージェントで継続してモニタリングできるように包含タグ GuardDutyManaged:true を追加します。

その他の問題のトラブルシューティング

問題に適したシナリオが見つからない場合は、次のトラブルシューティングもご参照ください。

- https://console.aws.amazon.com/guardduty/ へのアクセス時の IAM に関する一般的な問題については、「Amazon GuardDuty アイデンティティとアクセスのトラブルシューティング」を参照してください。
- アクセス時の認証と認可の問題については AWS AWS Console Home、「IAM のトラブルシューティング」を参照してください。

Amazon GuardDuty のリージョンとエンドポイント

AWS リージョン Amazon GuardDuty が利用可能な を表示するには、 のAmazon GuardDuty エンドポイント」を参照してくださいAmazon Web Services 全般のリファレンス。

GuardDuty は、サポートされているすべての AWS リージョンで有効にすることをお勧めします。このように設定することで、GuardDuty はアクティブに使用されていないリージョンでも、許可されていないアクティビティや異常なアクティビティに関する検出結果を生成できます。これにより、GuardDuty はサポートされている の AWS CloudTrail イベントをモニタリングでき AWS リージョン、グローバルサービスに関連するアクティビティを検出する機能が低下します。

リージョン固有機能の可用性

GuardDuty の機能の可用性は、以下のようにリージョンによって異なります。

CreateFilter および UpdateFilter のフィルター可能なフィールドを拡張

抑制ルールとフィルターの追加のフィルタリング可能なフィールドは、 AWS パーティション () でのみ使用できますaws。他のパーティションでは、コンソールでサポートされているフィールドを引き続き使用できます。

ListFindings API と GetFindingsStatistics API

GetFindingsStatistics API と ListFindings API には、一時的に利用できる `consoleOnly` フラグがあります。これらの API のいずれかまたは両方を使用した場合、`consoleOnly` フラグにより、最大 1,000 件の結果を取得できます。

Malware Protection for EC2

GuardDuty は、AWS 専有ローカルゾーンで Malware Protection for EC2 機能をサポートしています。

RDS Protection

RDS Protection は、アジアパシフィック (台北) (`ap-east-2`) リージョンではサポートされていません。

RDS Protection は Dedicated Local Zones AWS ではサポートされていません。

IAM 検出結果タイプ – CredentialAccess:IAMUser/CompromisedCredentials

CredentialAccess:IAMUser/CompromisedCredentials の検出結果タイプは、次のリージョンではサポートされていません。

AWS リージョン	リージョンコード
AWS GovCloud (US-West)	us-gov-west-1
AWS GovCloud (US-East)	us-gov-east-1
中国 (北京)	cn-north-1
中国 (寧夏)	cn-northwest-1

IAM 検出結果タイプ – DefenseEvasion:IAMUser/BedrockLoggingDisabled

DefenseEvasion:IAMUser/BedrockLoggingDisabled 結果タイプは、アジアパシフィック (香港) (ap-east-1) リージョンではサポートされていません。

AI Protection の検出結果タイプ

Impact:IAMUser/AnomalousModelInvocation および Impact:IAMUser/CostHarvesting検出結果タイプには、Amazon Bedrock または Amazon SageMaker AI が必要です。Amazon Bedrock AWS リージョン が利用できない では、GuardDuty は Amazon SageMaker AI モデルの呼び出しからのみこれらの検出結果タイプを生成します。Amazon Bedrock が利用可能なリージョンについては、「Amazon Bedrock ユーザーガイド AWS リージョン」の「Amazon Bedrock supported」を参照してください。

Impact:IAMUser/PromptInjection.Direct 検出結果のタイプは Amazon Bedrock ガードレールによって異なりますが、すべてで利用できるわけではありません AWS リージョン。この検出結果タイプは、Amazon Bedrock ガードレールが利用可能なリージョンでのみサポートされています。現在のリストについては、「Amazon Bedrock ユーザーガイド」の「Amazon Bedrock ガードレールでサポートされているリージョン」を参照してください。

一般的な API サポート

Amazon GuardDuty APIs リファレンスの次の API は、以前に指定した の一部のデータソースまたは機能が使用できないため、リージョンによって異なる場合があります AWS リージョン。Amazon GuardDuty

- CreateDetector
- UpdateDetector
- UpdateMemberDetectors
- UpdateOrganizationConfiguration

- GetDetector
- GetMemberDetectors
- DescribeOrganizationConfiguration

Amazon EC2 検出結果 - DefenseEvasion:EC2/UnusualDoHActivity および DefenseEvasion:EC2/UnusualDoTActivity

次の表は、GuardDuty を使用できるが、これら 2 つの Amazon EC2 検出結果タイプがまだサポートされていない AWS リージョン を示しています。

AWS リージョン	リージョンコード
アジアパシフィック (ソウル)	ap-northeast-2
アジアパシフィック (大阪)	ap-northeast-3
アジアパシフィック (ジャカルタ)	ap-southeast-3

AWS GovCloud (US) リージョン

最新情報については、「AWS GovCloud (US) ユーザーガイド」の「Amazon GuardDuty」を参照してください。

中国リージョン

最新情報については、「機能の可用性と実装の違い」を参照してください。

GuardDuty の従来のアクションとパラメータ

Amazon GuardDuty では、一部の API アクションとパラメータが非推奨になりましたが、引き続きサポートされています。従来のオプションに置き換わる新しい API アクションとパラメータを使用するのがベストプラクティスです。次の表では、従来のアクションとパラメータを、新しいアクションとパラメータと比較しています。

従来のアクション/パラメータ	新しいアクション/パラメータ	比較
DisassociateFromMasterAccount	DisassociateFromAdministratorAccount	両方のアクションで同じ実装が行われるため、GuardDuty は DisassociateFromAdministratorAccount の Administrator という用語を使用します。
DescribeOrganizationConfiguration および UpdateOrganizationConfiguration の autoEnable パラメータ	autoEnableOrganizationMembers	autoEnableOrganizationMembers を使用すると、GuardDuty 管理者アカウントは、すべてのメンバーアカウントについて GuardDuty を監査し、いずれかの値に設定するよう強制できます。API を使用すると、すべてのメンバーアカウントの設定を更新するのに最大 24 時間かかる場合があります。autoEnableOrganizationMembers フィールドの可能な値の詳細については、「autoEnableOrganizationMembers」を参照してください
2023 年 3 月の GuardDuty API の変更 にリストされている API の dataSourc	features	2023 年 3 月以降、features を使用して GuardDuty Malware Protection for EC2 と新しい GuardDuty 保護プランを設定できるようになります。2023 年 3 月より前にリ

従来のアクション/パラメータ	新しいアクション/パラメータ	比較
es パラメータ。		リースされた保護プラン (Malware Protection for EC2 を含む) は、引き続き dataSources を使用した設定をサポートしています。API を使用して保護プランを設定する場合、各 API リクエストには dataSources または features のいずれかを含めることができます。両方を含めることはできません。

Amazon GuardDuty のドキュメント履歴

次の表に、「Amazon GuardDuty ユーザーガイド」の前回のリリース以後にドキュメントに加えられた重要な変更を示します。このドキュメントの更新に関する通知を受け取るには、RSS フィードにサブスクライブできます。

変更	説明	日付
新機能 - ランタイムモニタリング用のカスタム検出ルール	GuardDuty のカスタム検出ルールは、Runtime Monitoring をサポートし、権限のエスカレーション、認証情報の盗難、防御の障害、Fargate の Amazon EKS、Amazon EC2、Amazon ECS でのコンテナエスケープなどのホストとコンテナのアクティビティを検出できるようになりました。詳細については、「ランタイムモニタリングのカスタム検出ルール」を参照してください。これらのルールは、4 つの新しい Runtime Monitoring 結果タイプも生成します。詳細については、「GuardDuty Runtime Monitoring の検出結果タイプでカスタム検出ルールを必要とする検出結果タイプ GuardDuty」を参照してください。	2026 年 10 月 15 日
更新された検出結果タイプ - UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.InsideAWS	既存のResourceCredentialExfiltration.InsideAWS検出結果は、Amazon ECS に加えて Lambda 認証情報の流出を検出するようにな	2026 年 9 月 28 日

	りました。詳細については、「UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.InsideAWS」を参照してください。	
更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring は、Amazon EKS、Amazon EC2、Amazon ECS AWS Fargate リソース用の新しいセキュリティエージェントバージョン 1.17.1 をリリースします。新しいエージェントバージョンとセキュリティエージェントを更新するための追加リソースのリストの詳細については、「GuardDuty security agent release versions」を参照してください。	2026 年 9 月 18 日

ドキュメントの更新 - Runtime Monitoring でサポートされているプラットフォーム	GuardDuty Runtime Monitoring に、サポートされている CPU アーキテクチャ、オペレーティングシステム、カーネルバージョンのページが追加されました。Amazon EC2 インスタンス、 AWS Fargate (Amazon ECS のみ）、および Amazon EKS クラスターの前提条件ページでは、検証済みの CPU アーキテクチャ、OS ディストリビューション、カーネルバージョンについてこのページを参照するようになりました。このページには Bottlerocket カーネルバージョン 6.18 が含まれるようになりました。	2026 年 9 月 18 日
カスタム検出ルールのドキュメント化された編集済み AWS CloudTrail イベントフィールド	AWS サービスが発行する AWS CloudTrail イベントのフィールドの値を秘匿化できること、および秘匿化されたフィールドに依存する検出ロジックを持つカスタム検出ルールが一致しないことを説明するガイダンスを追加しました。詳細については、「編集されたイベントフィールド」を参照してください。	2026 年 9 月 18 日

AI Protection が攻撃シーケンスの検出に貢献	GuardDuty 拡張脅威検出では、攻撃シーケンスを検出するために相関するシグナルに GuardDuty AI Protection の検出結果が含まれるようになりました。詳細については、GuardDuty Extended Threat Detection」および「GuardDuty AI Protection」を参照してください。	2026 年 9 月 17 日
GuardDuty サービスにリンクされたロール (SLR) の更新	GuardDuty は、 `vpc-lattice:CreateServiceNetworkVpcAssociation` および アクセス`vpc-lattice:AssociateViaAWSService` 許可を含めるようにサービスにリンクされたロール (SLR) を更新しました。これらのアクセス許可により、GuardDuty は Amazon VPC Lattice アクションを使用して Runtime Monitoring エージェントの接続を有効にすることができます。詳細については、「GuardDuty のためのサービスにリンクされたロールの許可」を参照してください。	2026 年 9 月 9 日

新機能 - カスタム検出ルール	GuardDuty は、脅威アクターの手法に沿った検出ルールの厳選されたライブラリであるカスタム検出ルールを導入しました。ルールを参照、有効化、管理して、環境内の特定の手法を検出できます。詳細については、「カスタム検出ルール」を参照してください。	2026 年 9 月 1 日
更新された機能 – GuardDuty 調査 (プレビュー)	組織レベルの調査で 100 アカウントの制限を削除しました。組織調査で、組織内のすべてのアカウントが分析されるようになりました。詳細については、GuardDuty の調査」を参照してください。	2026 年 8 月 24 日
更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring は、Amazon EKS、Amazon EC2、Amazon ECS AWS Fargate リソース用の新しいセキュリティエージェントバージョン 1.16.0 をリリースします。新しいエージェントバージョンとセキュリティエージェントを更新するための追加リソースのリストの詳細については、「GuardDuty security agent release versions」を参照してください。	2026 年 7 月 17 日

新しい検出結果タイプ - UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.InsideAWSおよび更新された検出結果タイプ - UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.OutsideAWS	GuardDuty は、Amazon ECS タスク用に作成された一時的な AWS 認証情報が別のからいつ使用されるかを検出する新しい検出結果タイプを導入しました AWS アカウント。さらに、既存のResourceCredentialExfiltration.OutsideAWS検出結果は、Lambda に加えて Amazon ECS タスク認証情報の流出を検出するようになりました。詳細については、「UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.InsideAWS」および「UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.OutsideAWS」を参照してください。	2026 年 7 月 15 日
新しい保護プラン - GuardDuty AI Protection	GuardDuty AI Protection は、Amazon Bedrock と Amazon SageMaker AI を使用する AI ワークロードに対する脅威を検出します。AI Protection を有効にすると、GuardDuty はこれらのサービスからの AWS CloudTrail データイベントを分析して、異常なモデル呼び出し、コスト収集攻撃、直接的なプロンプト挿入の試行を検出します。詳細については、GuardDuty AI Protection」を参照してください。	2026 年 7 月 13 日

Malware Protection for S3 の Amazon EventBridge イベントスキーマのその他の脅威の詳細	Malware Protection for S3 オブジェクトスキャン結果の Amazon EventBridge イベントで、追加の脅威の詳細が提供されるようになりました。新しいフィールドには、`source`、`itemDetails` 、`hash`、および が含まれます`itemPath`。詳細については、S3 オブジェクトのスキャン結果」を参照してください。	2026 年 7 月 7 日

GuardDuty 調査 (プレビュー)	Amazon GuardDuty は、パブリックプレビューで AI を活用した調査を提供するようになりました。調査を作成して GuardDuty の検出結果とアカウントを自動的に分析し、信頼度スコア処理、MITRE ATT&CK® 手法分類、実用的な推奨事項を使用して構造化された概要を数分で受け取ることができます。GuardDuty コンソール、CLI、API、または AWS MCP サーバーから調査を開始できます。プレビューは、 AWS リージョン米国東部 (バージニア北部) 、米国東部 (オハイオ) 、米国西部 (オレゴン) 、カナダ (中部) 、欧州 (フランクフルト) 、欧州 (アイルランド) 、欧州 (ロンドン) 、欧州 (パリ) 、欧州 (ストックホルム) 、アジアパシフィック (東京) の 10 で利用できます。詳細については、GuardDuty の調査」を参照してください。	2026 年 6 月 22 日
Amazon RDS for MariaDB と MySQL	Database Insights が利用可能になりました。	2026 年 6 月 11 日

更新された機能 - Malware Protection for S3	GuardDuty は、アーカイブから抽出および分析できるファイルの最大数と、抽出できるネストされたアーカイブの最大レベルの Malware Protection for S3 クォータを増やしました。詳細については、「Quotas in Malware Protection for S3」を参照してください。	2026 年 6 月 9 日
新しい Runtime Monitoring の検出結果タイプ - Sensitive FileModified	GuardDuty Runtime Monitoring には、Amazon EC2 インスタンスまたはコンテナのセキュリティ重視のシステムファイルが変更されたときに通知する 3 つの新しい検出結果タイプが導入されており、攻撃者が永続性を取得したり、権限を昇格したり、検出を回避したりできる可能性があります。詳細については、「Persistence:Runtime/SensitiveFileModified」、「PrivilegeEscalation:Runtime/SensitiveFileModified」、および「DefenseEvasion:Runtime/SensitiveFileModified」を参照してください。	2026 年 6 月 2 日

更新された機能 - 検出結果フィルター条件	結果フィルター条件リストからサポートされていないフィールドを削除しました。詳細については、GuardDuty での検出結果のフィルタリング」を参照してください。	2026 年 5 月 29 日
廃止された 2 つの検出結果タイプ	PrivilegeEscalation:Kubernetes/PrivilegedContainer および Persistence:Kubernetes/ContainerWithSensitiveMountの検出結果タイプは廃止されました。	2026 年 5 月 29 日
更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring Kubernetes バージョン 1.36 のサポートが追加されました。詳細については、「ランタイムモニタリングの前提条件」を参照してください。	2026 年 5 月 28 日
更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring は、Amazon ECS AWS Fargate リソース用の新しいセキュリティエージェントバージョン 1.15.0 をリリースします。新しいエージェントバージョンとセキュリティエージェントを更新するための追加リソースのリストの詳細については、「GuardDuty security agent release versions」を参照してください。	2026 年 5 月 28 日

更新された機能 - Extended Threat Detection	GuardDuty 拡張脅威検出は、MALICIOUS_PACKAGE、MISCONFIGURATION、REACHABILITY、SENSITIVE_DATA、および脆弱性の 5 つの新しいインジケータタイプを攻撃シーケンスの検出結果の詳細に追加します。詳細については、「攻撃シーケンスの検出結果の詳細」を参照してください。	2026 年 5 月 20 日
更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring は、Amazon EKS および Amazon EC2 リソース用の新しいセキュリティエージェントバージョン 1.15.0 をリリースします。新しいエージェントバージョンとセキュリティエージェントを更新するための追加リソースのリストの詳細については、「GuardDuty security agent release versions」を参照してください。	2026 年 5 月 14 日

AmazonGuardDutyServiceRolePolicy を更新	s3:ListBucket 、ecs:DescribeTasks 、および アクセスecs:DescribeTaskDefinition 許可を GuardDuty サービスにリンクされたロールに追加しました。詳細については、GuardDuty updates to AWS managed policies」を参照してください。	2026 年 4 月 23 日
更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring は、Amazon EKS リソース用の新しいセキュリティエージェントバージョン 1.12.2 をリリースします。新しいエージェントバージョンとセキュリティエージェントを更新するための追加リソースのリストの詳細については、「GuardDuty security agent release versions」を参照してください。	2026 年 4 月 22 日
機能の更新 - フィルタリング可能なフィールドの拡張	GuardDuty は、抑制ルールを作成または更新するときに、検出結果フィールドのフィルタリングをサポートするようになりました。タイムスタンプフィールドは、Unix エポックミリ秒形式の値を受け入れます。詳細については、GuardDuty での検出結果のフィルタリング」を参照してください。	2026 年 4 月 20 日

更新された機能 - Malware Protection for S3	S3 オブジェクトスキャン結果の EventBridge 通知に、スキャンがスキップされた`scanResultDetails` ときに に `statusReasons` フィールドが含まれるようになりました。このフィールドには、スキャンがスキップされた具体的な理由が表示されます。詳細については、「S3 object potential scan status and result status」を参照してください。	2026 年 4 月 10 日
更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring が Amazon EC2 リソース用の新しいセキュリティエージェントバージョン 1.9.2 をリリースしました。新しいエージェントバージョンとセキュリティエージェントを更新するための追加リソースのリストの詳細については、「GuardDuty security agent release versions」を参照してください。	2026 年 4 月 3 日

AmazonGuardDutyServiceRolePolicy を更新	AWS CloudTrail イベントを消費するための追加のメカニズムを有効にするアクセス`cloudtrail:CreateServiceLinkedChannel` 許可を追加しました。詳細については、GuardDuty updates to AWS managed policies」を参照してください。	2026 年 3 月 25 日
更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring Kubernetes バージョン 1.35 のサポートが追加されました。詳細については、「ランタイムモニタリングの前提条件」を参照してください。	2026 年 3 月 20 日
新しいスコープダウンフルアクセスポリシー `AmazonGuardDutyFullAccess_v2` で `AmazonGuardDutyFullAccess` を置き換えます。	2026 年 3 月 13 日をもって、GuardDuty は`AmazonGuardDutyFullAccess` ポリシーを廃止し、 という名前のスコープダウンポリシーに置き換えました`AmazonGuardDutyFullAccess_v2` 。詳細については、「AWS マネージドポリシー: AmazonGuardDutyFullAccess_v2」を参照してください。	2026 年 3 月 13 日

新しい検出結果タイプ - CredentialAccess:IAMUser/CompromisedCredentials	GuardDuty 侵害されている可能性のある IAM アクセスキーが API オペレーションの呼び出しにいつ使用されたかを検出する新しい検出結果タイプを紹介します AWS。詳細については、「CredentialAccess:IAMUser/CompromisedCredentials」を参照してください。	2026 年 3 月 6 日
新しい検出結果タイプ - UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.OutsideAWS	GuardDuty は、 以外のホストが AWS 環境内の AWS Lambda リソースで作成された一時的な AWS 認証情報を使用して AWS API オペレーションを実行 AWS しようとしたときを検出する新しい検出結果タイプを導入しました。詳細については、「UnauthorizedAccess:IAMUser/ResourceCredentialExfiltration.OutsideAWS」を参照してください。	2025 年 12 月 16 日

更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring が Amazon EC2 リソース用の新しいセキュリティエージェントバージョン 1.9.1 をリリースしました。新しいエージェントバージョンとセキュリティエージェントを更新するための追加リソースのリストの詳細については、「GuardDuty security agent release versions」を参照してください。	2025 年 12 月 2 日
更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring は、Amazon EKS リソース用の新しいセキュリティエージェントバージョン 1.12.1 をリリースします。新しいエージェントバージョンとセキュリティエージェントを更新するための追加リソースのリストの詳細については、「GuardDuty security agent release versions」を参照してください。	2025 年 12 月 2 日

更新された機能 - Extended Threat Detection	GuardDuty 拡張脅威検出は、AttackSequence:EC2/CompromisedInstanceGroup と AttackSequence:ECS/CompromisedCluster の 2 つの新しい攻撃シーケンス検出タイプを導入し、侵害された Amazon EC2 インスタンスと Amazon ECS クラスターに関連するマルチステージ攻撃を検出するのに役立ちます。Runtime Monitoring を有効にして脅威の検出を最大化します。詳細については、「攻撃シーケンスの検出結果タイプ」を参照してください。	2025 年 12 月 2 日
新機能 - 抑制ルールのワイルドカード	GuardDuty は、ワイルドカードをサポートする Matches および NotMatches 条件の導入により、抑制機能を強化しました。お客様は * (任意の文字数に一致) と を使用できるようになりました。 (最大 1 文字一致) 抑制ルールを作成するときにワイルドカードとして指定します。抑制ルール。	2025 年 12 月 2 日

新機能 - Amazon CloudWatch 使用状況メトリクスのサポート	GuardDuty は、すべての保護プランの使用状況メトリクスを Amazon CloudWatch に公開し、お客様が使用状況をモニタリングできるようにします。詳細については、GuardDuty の使用状況のモニタリングとコストの見積もり」を参照してください。	2025 年 11 月 25 日
新しい検出結果タイプ - DefenseEvasion:IAMUser/BedrockLoggingDisabled	GuardDuty は、Amazon Bedrock モデルの呼び出しログ記録が無効になっているタイミングを検出する新しい検出結果タイプを導入しました。これは、AI ワークロードアクティビティの検出を回避しようとしていることを示している可能性があります。詳細については、「DefenseEvasion:IAMUser/BedrockLoggingDisabled」を参照してください。	2025 年 11 月 21 日

新機能 - Malware Protection for Backup	GuardDuty Malware Protection for Backup が、GuardDuty が利用可能なすべての商用リージョンで利用可能になりました。この機能は、Amazon EBS スナップショット、EC2 AMIs、リカバリポイントの完全スキャンと増分スキャンを通じて、バックアップリソースにマルウェアが存在する可能性を検出するのに役立ちます。この機能の詳細については、「Malware Protection for Backup」を参照してください。	2025 年 11 月 19 日
AmazonGuardDutyFullAccess と AmazonGuardDutyFullAccess_V2 への変更	Malware Protection for Backup を有効にするときに IAM ロールを GuardDuty に渡すことができるアクセス許可を追加しました。	2025 年 11 月 19 日
新機能 - Malware Protection for S3 オンデマンドスキャン。	GuardDuty が Malware Protection for S3 のオンデマンドスキャンを発表しました。この機能を使用すると、新しい SendObjectMalwareScan API を使用して、S3 バケットに保存されている既存のオブジェクトのスキャンをトリガーできます。詳細については、「オンデマンド S3 マルウェアスキャン」を参照してください。	2025 年 11 月 17 日

更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring が Amazon ECS AWS Fargate リソース用の新しいセキュリティエージェントバージョン 1.9.0 をリリースしました。新しいエージェントバージョンやセキュリティエージェントを更新するための追加リソースのリストについては、「GuardDuty セキュリティエージェントのリリースバージョン」を参照してください。	2025 年 10 月 28 日
更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring が Amazon EC2 リソース用の新しいセキュリティエージェントバージョン 1.9.0 をリリースしました。新しいエージェントバージョンやセキュリティエージェントを更新するための追加リソースのリストについては、「GuardDuty セキュリティエージェントのリリースバージョン」を参照してください。	2025 年 10 月 23 日

新しい Runtime Monitoring の検出結果タイプ - DefenseEvasion:Runtime/KernelModuleLoaded	GuardDuty Runtime Monitoring には、脅威アクターがカーネルモジュールを使用してセキュリティコントロールをバイパスし、システムオペレーションを変更し、永続的なシステムアクセスを取得する防御回避手法を特定するのに役立つ新しい検出結果タイプが導入されています。詳細については、「DefenseEvasion:Runtime/KernelModuleLoaded」を参照してください。	2025 年 10 月 15 日
更新された機能 - Runtime Monitoring	Amazon EKS リソース用の GuardDuty SageMaker Role Manager の OS ディストリビューションとセキュリティエージェントバージョンの Kubernetes バージョン 1.34 のサポートが追加されました。詳細については、「Validate architectural requirements for Amazon EKS」を参照してください。	2025 年 10 月 3 日

Amazon RDS データベースの問題に関するトラブルシューティングガイダンスを追加しました	GuardDuty RDS Protection のモニタリング機能に影響を与える可能性のある RDS データベースの問題を解決するためのガイダンスを追加しました。これには、`storage-full` ステータスや RDS for PostgreSQL リードレプリカの要件が含まれます。詳細については、「Troubleshooting RDS Protection monitoring issues」を参照してください。	2025 年 9 月 15 日
更新された機能 - Runtime Monitoring	Amazon EC2 の GuardDuty Runtime Monitoring のカーネルバージョン 6.13 および 6.14 をサポートする Ubuntu Noble OS ディストリビューションを追加しました。この OS ディストリビューションでサポートされている EC2 エージェントバージョンについては、「Validate architectural requirements for Amazon EC2」を参照してください。	2025 年 9 月 13 日
更新された機能 - RDS Protection	GuardDuty RDS Protection によってモニタリングされる RDS for PostgreSQL データベースの PostgreSQL 17 サポートが追加されました。詳細については、RDS Protection の「サポートされているデータベース」を参照してください。	2025 年 9 月 13 日

更新された機能 - Malware Protection for S3	GuardDuty Malware Protection for S3 は、抽出されたアーカイブファイルのデフォルトクォータを 1,000 から 10,000 ファイルに引き上げます。詳細については、「Quotas in Malware Protection for S3」を参照してください。	2025 年 9 月 3 日
更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring が Amazon EKS リソース用の新しいセキュリティエージェントバージョン 1.11 をリリースしました。新しいエージェントバージョンとセキュリティエージェントを更新するための追加リソースのリストの詳細については、「GuardDuty security agent release versions」を参照してください。	2025 年 8 月 29 日
更新された機能 - リストの操作	GuardDuty では、IP アドレスとドメイン名の両方をサポートするカスタムの信頼できるエンティティリストと脅威エンティティリストが導入されています。GuardDuty は IP アドレスリストのサポートを継続し、カスタム脅威検出にエンティティリストを使用することをお勧めします。詳細については、「Customizing threat detection with entity lists and IP address lists」を参照してください。	2025 年 8 月 15 日

更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring は、Amazon EC2 リソースと Amazon ECS AWS Fargate リソースの両方で新しいセキュリティエージェントバージョン 1.8.0 をリリースします。新しいエージェントバージョンとセキュリティエージェントを更新するための追加リソースのリストの詳細については、「GuardDuty security agent release versions」を参照してください。	2025 年 8 月 12 日
アジアパシフィック (台北) リージョンのサポート	Amazon GuardDuty がアジアパシフィック (台北) (ap-east-2) リージョンで利用可能になりました。このリージョンで GuardDuty を有効にするには、「開始方法」を参照してください。このリージョンで Amazon SNS GuardDuty の発表をサブスクライブすることで、GuardDuty の機能の更新と脅威検出に関する通知を受け取ることができます。	2025 年 7 月 31 日

更新された機能 - Malware Protection for S3	Malware Protection for S3 で、5 GB から最大 100 GB のオブジェクトのスキャンがサポートされるようになりました。これには、個々のオブジェクトと抽出されたアーカイブファイルの両方が含まれます。詳細については、「Malware Protection for S3 quotas」を参照してください。	2025 年 7 月 23 日
更新された機能 - 信頼 IP および脅威リストに期待されるバケット所有者を追加	GuardDuty は、信頼できる IP および脅威 IP リストの[期待されるバケット所有者]フィールドを追加しました。このオプションのフィールドでは、GuardDuty が Amazon S3 バケットの所有権を検証するために使用する AWS アカウント ID を指定できます。Amazon S3 詳細については、「Working with trusted IP lists and threat lists」を参照してください。	2025 年 7 月 16 日

更新された機能 - Extended Threat Detection	GuardDuty 拡張脅威検出は、EKS 監査ログ、プロセスのランタイム動作、 AWS API アクティビティ間で複数のセキュリティシグナルを相関させることで、Amazon EKS クラスターのサポートを拡張するようになりました。EKS Protection、Runtime Monitoring (EKS アドオンを使用)、またはその両方を有効にして、脅威の検出を最大化します。潜在的な脅威を特定するため、GuardDuty は新しい検出結果タイプである AttackSequence:EKS/CompromisedCluster を導入します。詳細については、「Extended Threat Detection」を参照してください。	2025 年 6 月 17 日
更新された機能 - Malware Protection for S3	Malware Protection for S3 は、非常に高い圧縮率のアーカイブのスキャンをサポートしていません。これらのファイルはスキャン中にスキップされ、`UNSUPPORTED` のスキャン結果でマークされます。詳細については、「S3 object potential scan status and result status」を参照してください。	2025 年 6 月 13 日

更新された機能 - Malware Protection for EC2	Malware Protection for EC2 では、`productCode` を `marketplace` として使用することで、ほとんどのインスタンスをスキャンするサポートが追加されました。これは、GuardDuty 実行型マルウェアスキャンとオンデマンドマルウェアスキャンの両方に適用されます。詳細については、「Reasons for skipping resource during malware scan」を参照してください。	2025 年 6 月 13 日
新しい AmazonGuardDutyFullAccess_v2 ポリシー	GuardDuty サービスプリンシパルに管理アクションを制限することでセキュリティを強化するためのアクセス許可を持つ新しい AmazonGuardDutyFullAccess_v2 ポリシーを追加しました。この推奨ポリシーの詳細については、「AWS マネージドポリシー: AmazonGuardDutyFullAccess_v2」を参照してください。	2025 年 6 月 4 日

RDS Protection のリージョンサポートの拡張	GuardDuty RDS Protection が、メキシコ (中部) 、アジアパシフィック (タイ)、アジアパシフィック (マレーシア) の各リージョンで利用可能になりました。RDS Protection は、サポートされている Aurora MySQL、Aurora PostgreSQL (Limitless Database を含む)、RDS for PostgreSQL で疑わしいログイン動作を検出するのに役立ちます。脅威が検出された場合、GuardDuty は RDS Protection の検出結果を生成します。サポートされているデータベースと、新しくサポートされているリージョンでこの保護プランを有効にする方法の詳細については、「RDS Protection」を参照してください。	2025 年 6 月 4 日
更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring が Amazon EC2 リソース用の新しいセキュリティエージェントバージョン 1.7.1 をリリースしました。新しいエージェントバージョンとセキュリティエージェントを更新するための追加リソースのリストの詳細については、「GuardDuty security agent release versions」を参照してください。	2025 年 6 月 3 日

Extended Threat Detection のサポート	GuardDuty Extended Threat Detection がアジアパシフィック (タイ) (ap-southeast-7) で利用可能になりました。アクティベーションを必要とせずに、 内のデータソース、複数のタイプの AWS リソース、および時間にわたるマルチステージ攻撃を検出します AWS アカウント。潜在的な脅威が検出されると、攻撃シーケンスの検出結果が生成されます。詳細については、「Extended Threat Detection」を参照してください。	2025 年 5 月 12 日
メキシコ (中部) リージョンのサポート	Amazon GuardDuty がメキシコ (中部) (`mx-central-1`) リージョンで利用可能になりました。このリージョンで GuardDuty を有効にするには、「開始方法」を参照してください。このリージョンで Amazon SNS GuardDuty の発表をサブスクライブすることで、GuardDuty の機能の更新と脅威検出に関する通知を受け取ることができます。	2025 年 5 月 7 日

更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring が Amazon EKS リソース用の新しいセキュリティエージェントバージョン 1.10.0 をリリースしました。新しいエージェントバージョンとセキュリティエージェントを更新するための追加リソースのリストの詳細については、「GuardDuty security agent release versions」を参照してください。	2025 年 4 月 4 日
更新された機能 - Runtime Monitoring	Runtime Monitoring が Amazon ECS-Fargate リソース用の新しいエージェントバージョン 1.7.0 をリリースしました。新しいエージェントバージョンとセキュリティエージェントを更新するための追加リソースのリストの詳細については、「GuardDuty security agent release versions」を参照してください。	2025 年 4 月 4 日

更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring が Amazon EC2 リソース用の新しいセキュリティエージェントバージョン 1.7.0 をリリースしました。新しいエージェントバージョンとセキュリティエージェントを更新するための追加リソースのリストの詳細については、「GuardDuty security agent release versions」を参照してください。	2025 年 4 月 3 日
アジアパシフィック (タイ) リージョンのサポート	Amazon GuardDuty が、アジアパシフィック (タイ) リージョンで利用可能になりました。このリージョンでサポートされている機能の詳細については、「Region-specific feature availability」を参照してください。このリージョンで GuardDuty を有効にするには、「開始方法」を参照してください。Amazon SNS GuardDuty の発表をサブスクライブすることで、GuardDuty の機能の更新と脅威の検出に関する通知を受け取ることができます。	2025 年 4 月 1 日

更新された機能	概要ダッシュボードに、生成されたすべてのセキュリティ検出結果に基づくインサイトが表示され、以前の 5,000 件の検出結果の制約が削除されます。これらのインサイトの詳細については、「GuardDuty Summary dashboard」を参照してください。	2025 年 3 月 17 日
更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring が Amazon EKS リソース用の新しいエージェントバージョン 1.9.0 をリリースしました。新しいエージェントバージョンとセキュリティエージェントを更新するための追加リソースのリストの詳細については、「GuardDuty security agent release versions」を参照してください。	2025 年 3 月 2 日
更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring に、Amazon EC2 リソースの新しいカバレッジ問題タイプ (Agent Not Provisioned) が追加されました。この問題のトラブルシューティングについては、「Troubleshooting Amazon EC2 runtime coverage issues」を参照してください。	2025 年 2 月 21 日

更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring は、Amazon EC2 および Amazon ECS-Fargate リソース用の新しいセキュリティエージェントをリリースします。新しいエージェントバージョンとセキュリティエージェントを更新するための追加リソースのリストの詳細については、「GuardDuty security agent release versions」を参照してください。	2025 年 2 月 6 日
既存のアジアパシフィック (マレーシア) リージョンでの GuardDuty サポート	GuardDuty Extended Threat Detection がアジアパシフィック (マレーシア) リージョンで利用可能になりました。詳細については、「Extended Threat Detection」を参照してください。	2025 年 1月 28 日

アジアパシフィック (マレーシア) リージョンのサポート	Amazon GuardDuty がアジアパシフィック (マレーシア) リージョンで使用可能になりました。このリージョンでサポートされている機能の詳細については、「Region-specific feature availability」を参照してください。このリージョンで GuardDuty を有効にするには、「開始方法」を参照してください。Amazon SNS GuardDuty の発表をサブスクライブすることで、GuardDuty の機能の更新と脅威の検出に関する通知を受け取ることができます。	2025 年 1 月 16 日
更新された機能 - Runtime Monitoring	GuardDuty Runtime Monitoring は、プロビジョニングされていないエージェントに関連する Amazon ECS-Fargate カバレッジの問題に関する追加情報とトラブルシューティング手順を更新しました。エージェントがプロビジョニングされていないという問題タイプの詳細については、「Amazon ECS-Fargate ランタイムカバレッジの問題のトラブルシューティング」を参照してください。	2025 年 1 月 8 日

新しい検出結果タイプ - Policy:IAMUser/ShortTermRootCredentialUsage	GuardDuty は、 AWS アカウント 環境にリストされている 用に作成された制限付きユーザー認証情報がリクエストの実行に使用されている場合に警告する新しい検出結果タイプを導入しました AWS のサービス。詳細については、「Policy:IAMUser/ShortTermRootCredentialUsage」を参照してください。	2025 年 1 月 8 日
新機能 - GuardDuty Extended Threat Detection	GuardDuty は、拡張脅威検出を発表して AWS アカウント、特定の期間に GuardDuty の基本的なデータソースと AWS リソースにまたがるマルチステージ攻撃シーケンスを検出します。追加コストなしで、この機能は GuardDuty を有効にしたすべてのアカウントで自動的に有効になります。この機能は、攻撃シーケンス検出結果タイプと呼ばれる 2 つの新しい GuardDuty 検出結果タイプを発表しました。詳細については、「Extended Threat Detection」を参照してください。	2024 年 12 月 1 日

強化されたクロスサービス機能 - EC2 の Runtime Monitoring と Malware Protection	Amazon GuardDuty 機能に対する新しい Amazon Elastic Kubernetes Service (Amazon EKS) 機能の影響: • Amazon EKS Auto Mode – Amazon EKS の Runtime Monitoring と EC2 の Malware Protection の両方がこれをサポートしています。 • Amazon EKS Hybrid Nodes – Amazon EKS の Runtime Monitoring と EC2 の Malware Protection の両方がこれをサポートしていません。 詳細については、「How Runtime Monitoring works with Amazon EKS clusters」と「Malware Protection for EC2」を参照してください。	2024 年 12 月 1 日

Runtime Monitoring の更新された機能 - Amazon EKS	Runtime Monitoring が Amazon EKS リソース用の新しいエージェントバージョン 1.8.1 (v1.8.1-eks-build.2) をリリースしました。この新しいエージェントバージョンでは、GuardDuty は RedHat、CentOS、および Fedora で実行される Amazon EKS リソースの Runtime Monitoring サポートを拡張します。詳細については、「Validating architectural requirements」を参照してください。リリースノートの詳細については、「Amazon EKS リソース用の GuardDuty セキュリティエージェント」を参照してください。	2024 年 11 月 23 日

Runtime Monitoring の更新された機能 - Amazon EC2	Runtime Monitoring が Amazon EC2 リソース用の新しいエージェントバージョン 1.5.0 をリリースしました。この新しいエージェントバージョンでは、GuardDuty は RedHat、CentOS、および Fedora で実行される Amazon EC2 リソースの Runtime Monitoring サポートを拡張します。詳細については、「Validating architectural requirements」を参照してください。リリースノートの詳細については、「Amazon EC2 リソース用の GuardDuty セキュリティエージェント」を参照してください。	2024 年 11 月 20 日
Runtime Monitoring の更新された機能 - Amazon ECS-Fargate	Runtime Monitoring は、Amazon ECS-Fargate リソース用の新しいエージェントバージョン 1.5.0 をリリースしました。リリースノートの詳細については、「AWS Fargate (Amazon ECS のみ) の GuardDuty セキュリティエージェント」を参照してください。	2024 年 11 月 14 日

Malware Protection for EC2 の更新された機能	GuardDuty Malware Protection for EC2 は、Amazon EC2 インスタンスで GuardDuty 実行型マルウェアスキャンを呼び出す検出結果のリストに 3 つの Runtime Monitoring 検出結果タイプを追加しました。Malware Protection for EC2 を有効にしたアカウントは、GuardDuty が以下のいずれかの検出結果を生成すると、GuardDuty が開始したマルウェアスキャンを監視します。 • Execution:Runtime/MaliciousFileExecuted • Execution:Runtime/SuspiciousShellCreated • PrivilegeEscalation:Runtime/ElevationToRoot	2024 年 11 月 7 日

RDS Protection の更新された機能	GuardDuty RDS Protection は、新しくリリースされた Aurora PostgreSQL Limitless Database エンジンバージョン `16.4-limitless` を、サポートされているデータベースのリストに追加します。RDS Protection を既に有効に AWS アカウント している の場合、GuardDuty は Limitless Database のログイン動作のモニタリングを自動的に開始します。RDS Protection の 30 日間無料トライアルを既に利用しているアカウントでは、Limitless Database およびモニタリング対象の他のサポートされているデータベースに関連する使用コストが発生します。詳細については、「RDS Protection」を参照してください。	2024 年 11 月 6 日

リージョンの拡張 - GuardDuty と AWS PrivateLink 統合	GuardDuty は、Amazon GuardDuty とインターフェイス VPC エンドポイント (AWS PrivateLink)のリージョンサポートを拡張するようになりました。以前は、リージョンのサポートが米国東部 (バージニア北部)、欧州 (アイルランド)、イスラエル (テルアビブ) で利用可能でした。このサポートは、GuardDuty AWS リージョン が利用可能なすべての に拡張されました。リージョンの違いの詳細については、「Region-specific feature availability」を参照してください。	2024 年 11 月 6 日
Runtime Monitoring の更新された機能 - Amazon ECS-Fargate	Runtime Monitoring が Amazon ECS-Fargate リソース用の新しいエージェントバージョン 1.4.1 をリリースしました。リリースノートの詳細については、「AWS Fargate (Amazon ECS のみ) の GuardDuty セキュリティエージェント」を参照してください。	2024 年 10 月 24 日

GuardDuty CloudFormation タグオペレーションのサポートの追加	GuardDuty で、タグキーと値の更新、およびスタックレベルのタグがサポートされるようになりました。これを行うには、IAM ロールに `guardduty:tagResource` アクセス許可を追加します。GuardDuty CloudFormation の詳細については、「AWS CloudFormation ユーザーガイド」の「Amazon GuardDuty リソースタイプのリファレンス」を参照してください。	2024 年 10 月 24 日
GuardDuty Malware Protection for S3 の更新された機能	Malware Protection for S3 を有効にするとき、ユーザーに代わってマルウェアスキャンアクションを実行するために必要なアクセス許可を持つサービスロールを選択できます。Malware Protection for S3 を有効にする詳細については、「バケットの Malware Protection for S3 の設定」を参照してください。	2024 年 10 月 22 日

更新された機能	GuardDuty は検出UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.InsideAWS結果タイプを強化して、Amazon EC2 インスタンスロールに関連付けられていない の VPC AWS アカウント エンドポイント (AWS PrivateLink) からの Amazon EC2 インスタンス AWS 認証情報の使用を検出します。この新しい GuardDuty 機能は、Amazon EC2 インスタンス認証情報の誤用の可能性を検出し、セッション認証情報を抽出 AWS アカウント してリモートのコンテキストを提供します。この新しい検出でサポートされる AWS サービスエンドポイントの詳細については、「 AWS CloudTrail ユーザーガイド」の「ネットワークアクティビティイベントのログ記録」を参照してください。	2024 年 10 月 21 日

更新された機能 - GuardDuty Runtime Monitoring	GuardDuty Runtime Monitoring で、 AWS 環境内の Amazon EC2 インスタンスまたはコンテナワークロードで疑わしいコマンドが実行されたときに通知する検出結果タイプに次の 3 つが追加されました。 • Discovery:Runtime/SuspiciousCommand • Persistence:Runtime/SuspiciousCommand • PrivilegeEscalation:Runtime/SuspiciousCommand	2024 年 10 月 10 日
新機能 - VPC エンドポイントのサポートの追加	GuardDuty が と統合 AWS PrivateLink され、VPC エンドポイントがサポートされるようになりました。 AWS PrivateLink 統合の詳細については、Amazon GuardDuty」と「インターフェイス VPC エンドポイント (AWS PrivateLink)」を参照してください。	2024 年 9 月 17 日
Runtime Monitoring の更新された機能 - Amazon EKS	Runtime Monitoring が Amazon EKS リソース用の新しいエージェントバージョン 1.7.1 をリリースしました。リリースノートの詳細については、「Amazon EKS クラスター用の GuardDuty セキュリティエージェント」を参照してください。	2024 年 9 月 13 日

Malware Protection for S3 の更新された機能	Malware Protection for S3 で、S3 オブジェクトスキャンの結果 Amazon EventBridge (EventBridge) スキーマに新しいフィールド `s3Throttled` が追加されました。`s3Throttled` フィールドは、Amazon Simple Storage Service (Amazon S3) バケットからストレージをアップロードまたは取得する際に遅延があったかどうかを示します。詳細については、「Amazon EventBridge による S3 オブジェクトスキャンのモニタリング」を参照してください。	2024 年 9 月 13 日
Runtime Monitoring の更新された機能 - Amazon EC2	Runtime Monitoring が Amazon EC2 リソース用の新しいエージェントバージョン 1.3.1 をリリースしました。リリースノートの詳細については、「Amazon EC2 インスタンス用の GuardDuty セキュリティエージェント」を参照してください。	2024 年 9 月 12 日

Runtime Monitoring の更新された機能 - Amazon ECS-Fargate	Runtime Monitoring が Amazon ECS-Fargate リソース用の新しいエージェントバージョン 1.3.1 をリリースしました。リリースノートの詳細については、「AWS Fargate (Amazon ECS のみ) の GuardDuty セキュリティエージェント」を参照してください。	2024 年 9 月 11 日
GuardDuty サービスにリンクされたロール (SLR) の更新	GuardDuty は SLR を更新し、Amazon EC2 アクションに `ec2:Describe:Vpcs` 許可を含めました。詳細については、「GuardDuty のためのサービスにリンクされたロールの許可」を参照してください。	2024 年 8 月 22 日

コンテンツの大幅な追加	GuardDuty は、Malware Protection for S3 機能にコンテンツの大幅な更新を追加しました。 • 「Malware Protection プランのリソースステータス」と「S3 オブジェクトスキャンの結果」に関連する通知を受信する Amazon EventBridge ルールを設定するための通知スキーマの例を新たに追加しました。詳細については、「Amazon EventBridge による S3 オブジェクトスキャンのモニタリング」を参照してください。 • 「S3 オブジェクトスキャン後のタグ失敗のトラブルシューティング」に関する情報を追加しました。	2024 年 8 月 20 日
GuardDuty Runtime Monitoring の更新された機能 - Amazon EC2	Runtime Monitoring が Amazon EC2 リソース用の新しいエージェントバージョン 1.3.0 をリリースしました。リリースノートの詳細については、「Amazon EC2 インスタンス用の GuardDuty セキュリティエージェント」を参照してください。	2024 年 8 月 19 日

GuardDuty Runtime Monitoring の更新された機能 - Amazon EKS	Runtime Monitoring が Amazon EKS リソース用の新しいエージェントバージョン 1.7.0 をリリースしました。リリースノートの詳細については、「Amazon EKS クラスター用の GuardDuty セキュリティエージェント」を参照してください。	2024 年 8 月 17 日
コンテンツの大幅な追加	GuardDuty は、Malware Protection for S3 および Malware Protection for EC2 機能に使用するマルウェア検出方法とスキャンエンジンに関する新しい情報を追加しました。詳細については、「GuardDuty マルウェア検出のスキャンエンジン」を参照してください。	2024 年 8 月 15 日
新機能 - AI ワークロードの保護	GuardDuty の基本的な脅威検出と Lambda Protection は、AWS上に構築された AI ワークロードのセキュリティを強化し、脅威を検出するのに役立ちます。詳細については、「GuardDuty による AI ワークロードの保護」を参照してください。	2024 年 8 月 14 日

GuardDuty Runtime Monitoring の更新された機能 - Fargate (Amazon ECS のみ)	Runtime Monitoring は、 AWS Fargate (Amazon ECS のみ) リソース用の新しいエージェントバージョン 1.3.0 をリリースしました。リリースノートの詳細については、「GuardDuty security agent for Fargate-ECS」を参照してください。	2024 年 8 月 9 日
更新された機能 - Malware Protection for S3	GuardDuty Malware Protection for S3 は、S3 バケットクォータの最大数を 10 から 25 バケットに増やします。このクォータは、各 AWS アカウント ごとに に適用されます AWS リージョン。詳細については、「Malware Protection for S3」を参照してください。	2024 年 8 月 8 日
更新 - Runtime Monitoring の新しい検出結果タイプ	GuardDuty は、モニタリング対象リソースでの疑わしいシェルの作成、およびプロセスが不審にその特権をルートに昇格させる特権エスカレーションを含む脅威を検出するのに役立つ新しい Runtime Monitoring の検出結果タイプを 2 つ追加しました。 • Execution:Runtime/SuspiciousShellCreated • PrivilegeEscalation:Runtime/ElevationToRoot	2024 年 8 月 6 日

更新 - との統合 AWS Security Hub CSPM	AWS Security Hub CSPM は、リソースを評価し、セキュリティ業界標準とベストプラクティスに照らしてコンプライアンスをチェックするための GuardDuty セキュリティコントロールのリストを提供します。詳細については、「Security Hub CSPM での GuardDuty コントロールの使用」を参照してください。	2024 年 7 月 11 日
検出結果の GuardDuty テスタースクリプトの更新	GuardDuty は、専用アカウントのさまざまな AWS リソースで 100 を超える検出結果をサポートするようになりました。詳細については、「専用アカウントでの GuardDuty の検出結果のテスト」を参照してください。	2024 年 6 月 28 日

Runtime Monitoring の更新された機能	Runtime Monitoring が Amazon EC2 リソース用の新しいセキュリティエージェントバージョン 1.2.0 をリリースしました。リリースノートの詳細については、「Amazon EC2 インスタンス用の GuardDuty セキュリティエージェント」を参照してください。セキュリティエージェントをこのリリースバージョンに手動で更新する詳細については、「Amazon EC2 リソースのセキュリティエージェントの手動管理」を参照してください。	2024 年 6 月 13 日
新機能 - Malware Protection for S3 が利用可能なリージョン	GuardDuty を利用可能なすべての商用リージョンで、GuardDuty Malware Protection for S3 を利用できるようになりました。この機能により、Amazon S3 バケットに新しくアップロードされたオブジェクトをスキャンしてマルウェアの可能性や疑わしいアップロードを調べ、ダウンストリームプロセスに取り込まれる前にそれらを分離するアクションを実行することができます。Malware Protection for S3 を有効にする詳細については、「GuardDuty Malware Protection for S3」を参照してください。	2024 年 6 月 12 日

新機能 - Malware Protection for S3	GuardDuty は、Malware Protection for S3 の一般提供を発表しました。これにより、Amazon S3 バケットに新しくアップロードされたオブジェクトをスキャンしてマルウェアの可能性や疑わしいアップロードを調べ、ダウンストリームプロセスに取り込まれる前にそれらを分離するアクションを実行することができます。この機能はAWSによって完全に管理されます。GuardDuty は、S3 オブジェクトスキャン結果を EventBridge のデフォルトのイベントバスに発行します。GuardDuty がスキャンされた S3 オブジェクトにタグを追加できるようにすることができます。隔離バケットへの分離などの下流のワークフローを構築したり、ユーザーやアプリケーションが特定のオブジェクトにアクセスできないようにタグを使用してバケットポリシーを定義したりできます。詳細については、「GuardDuty Malware Protection for S3」を参照してください。現在、以下のリージョンで利用できます。 • 米国東部 (バージニア北部) • 米国東部 (オハイオ) • 米国西部 (オレゴン)	2024 年 6 月 11 日

	• 欧州 (アイルランド) • 欧州 (フランクフルト) • 欧州 (ストックホルム) • アジアパシフィック (シドニー) • アジアパシフィック (東京) • アジアパシフィック (シンガポール)	
AmazonGuardDutyFullAccess ポリシーの更新	Malware Protection for S3 を有効にするときに IAM ロールを GuardDuty に渡すことができるアクセス許可を追加しました。このポリシー更新の詳細については、「AWS マネージドポリシーAmazonGuardDutyFullAccess」を参照してください。	2024 年 6 月 10 日
GuardDuty RDS Protection の更新された機能	RDS Protection は、RDS for PostgreSQL データベースのログインアクティビティをモニタリングするようサポートが拡張されます。この拡張の一環として、GuardDuty は、GuardDuty RDS Protection を既に有効にしているアカウントで RDS for PostgreSQL データベースからのログインデータのモニタリングを自動的に開始します。詳細については、「RDS Protection」を参照してください。	2024 年 6 月 6 日

GuardDuty Runtime Monitoring の更新された機能 - Fargate (Amazon ECS のみ)	Runtime Monitoring は、 AWS Fargate (Amazon ECS のみ) リソース用の新しいエージェントバージョン 1.2.0 をリリースしました。リリースノートの詳細については、「GuardDuty security agent for Fargate-ECS」を参照してください。	2024 年 5 月 31 日
GuardDuty Malware Protection for EC2 の更新された機能	Amazon EC2 インスタンスおよびコンテナワークロードにアタッチされている各 Amazon EBS ボリュームについて、GuardDuty Malware Protection for EC2 はスキャンする EBS ボリュームのサイズを最大 2048 GB に引き上げました。インスタンスにアタッチされた Amazon EBS ボリュームのスキャンについては、「GuardDuty Malware Protection for EC2」を参照してください。	2024 年 5 月 29 日
Runtime Monitoring の更新された機能	Amazon ECS-Fargate リソースのランタイムモニタリングは、 AWS Batch および によって起動されたタスクに対する潜在的な脅威の検出をサポートするようになりました AWS CodePipeline。詳細については、「Fargate での Runtime Monitoring の仕組み (Amazon ECS のみ)」を参照してください。	2024 年 5 月 28 日

Runtime Monitoring の更新された機能	Runtime Monitoring が Amazon EKS リソース用の新しいエージェントバージョン 1.6.1 をリリースしました。リリースノートの詳細については、「EKS アドオンエージェントのリリース履歴」を参照してください。	2024 年 5 月 14 日
Runtime Monitoring のリージョンサポートの拡張	GuardDuty は、Runtime Monitoring のサポートをカナダ西部 (カルガリー) リージョンに拡張しました。Runtime Monitoring の開始方法については、「Runtime Monitoring の有効化」を参照してください。	2024 年 5 月 7 日
RDS Protection のリージョンサポートの拡張	GuardDuty は RDS Protection のサポートを次のように拡張します AWS リージョン。 • カナダ西部 (カルガリー) • アジアパシフィック (ハイデラバード) • 欧州 (スペイン) • 欧州 (チューリッヒ) • 中東 (アラブ首長国連邦) • イスラエル (テルアビブ) • アジアパシフィック (メルボルン) この機能を有効にする方法については、「RDS Protection」を参照してください。	2024 年 5 月 3 日

Runtime Monitoring の更新された機能	Runtime Monitoring は、 AWS Fargate (Amazon ECS のみ) リソース用の新しいエージェントバージョン 1.1.0 をリリースしました。リリースノートの詳細については、「GuardDuty security agent for Fargate-ECS」を参照してください。	2024 年 5 月 1 日
Runtime Monitoring の更新された機能	Runtime Monitoring が Amazon EKS リソース用の新しいエージェントバージョン 1.6.0 をリリースしました。リリースノートの詳細については、「EKS アドオンエージェントのリリース履歴」を参照してください。	2024 年 4 月 29 日
IPAddressv6 のサポート	GuardDuty は、ローカル IP とリモート IP の両方の詳細に対する IPAddressv6 のサポートを追加しました。関連付けられたフィルター属性を使用して、GuardDuty の検出結果をフィルタリングしたり、抑制ルールを作成したりできます。	2024 年 4 月 18 日
検出結果のエクスポートを設定するコンソールエクスペリエンスの更新	GuardDuty は、 AWS アカウントで生成された検出結果を Amazon S3 バケットにエクスポートするコンソールエクスペリエンスを更新しました。詳細については、GuardDuty の検出結果のエクスポートを参照してください。	2024 年 4 月 1 日

Runtime Monitoring の更新された機能	Runtime Monitoring が Amazon EC2 リソース用の新しいセキュリティエージェントバージョン 1.1.0 をリリースしました。このバージョンは、Amazon EC2 インスタンスの Runtime Monitoring で GuardDuty 自動エージェント設定をサポートしています。リリースノートの詳細については、「Amazon EC2 インスタンス用の GuardDuty セキュリティエージェント」を参照してください。	2024 年 3 月 28 日

Amazon EC2 インスタンスの Runtime Monitoring の一般提供	GuardDuty は、Amazon EC2 インスタンスの Runtime Monitoring の一般提供 (GA) を発表しました。これで、GuardDuty がユーザーに代わって Amazon EC2 インスタンスのセキュリティエージェントをインストールおよび管理できるようにする自動エージェント設定を有効にすることができます。GuardDuty 自動エージェントでは、包含タグまたは除外タグを使用して、選択した Amazon EC2 インスタンスにのみセキュリティエージェントをインストールおよび管理するよう GuardDuty に通知することもできます。詳細については「How Runtime Monitoring works with Amazon EC2 instances」を参照してください。 この GA と共にリリースされた新しい検出結果タイプのリスト • Execution:Runtime/SuspiciousTool • Execution:Runtime/SuspiciousCommand • DefenseEvasion:Runtime/SuspiciousCommand • DefenseEvasion:Runtime/PtraceAntiDebugging	2024 年 3 月 28 日

- Execution:Runtime/MaliciousFileExecuted

Amazon GuardDuty によりサービスにリンクされたロール (SLR) が更新されました	Amazon EC2 の自動エージェントで GuardDuty Runtime Monitoring を有効にする場合は、 AWS Systems Manager アクションを使用して Amazon EC2 インスタンスの SSM 関連付けを管理します。GuardDuty 自動エージェント設定が無効になっている場合、GuardDuty は包含タグ (`GuardDutyManaged :true`) を持つ EC2 インスタンスのみを考慮します。 • 次のリストに、新しいアクセス許可を示します。 `"ssm:DescribeAssociation", "ssm:DeleteAssociation", "ssm:UpdateAssociation", "ssm:CreateAssociation", "ssm:StartAssociationsOnce", "ssm:AddTagsToResource", "ssm:CreateAssociation", "ssm:UpdateAssociation", "ssm:SendCommand", "ssm:GetCommandInvocation"`	2024 年 3 月 26 日

Runtime Monitoring の更新された機能	Amazon EKS 用の最新の GuardDuty セキュリティエージェント (アドオン) v1.5.0 リリースにより、Runtime Monitoring では、CPU とメモリの設定、`PriorityClass` 設定、DNS ポリシー設定など、GuardDuty セキュリティエージェントの特定のパラメータの設定がサポートされるようになりました。詳細については、「Amazon EKS の GuardDuty セキュリティエージェント (アドオン) パラメータを設定する」を参照してください。	2024 年 3 月 7 日
Runtime Monitoring の更新された機能	Runtime Monitoring が Amazon EKS リソース用の新しいエージェントバージョン 1.5.0 をリリースしました。リリースノートの詳細については、「EKS アドオンエージェントのリリース履歴」を参照してください。	2024 年 3 月 7 日
カナダ西部 (カルガリー) のサポート	Amazon GuardDuty がカナダ西部 (カルガリー) リージョンで使用可能になりました。GuardDuty 内の保護プランの一部はこのリージョンで使用できない場合があります。最新情報については、リージョンとエンドポイントを参照してください。	2024 年 3 月 6 日

Runtime Monitoring の更新された機能	Amazon EKS クラスターの GuardDuty セキュリティエージェントバージョン 1.0.0 および 1.1.0 は、2024 年 5 月 14 日以降サポートされなくなります。標準サポートが終了する前に実行できる手順については、「Amazon EKS クラスター用の GuardDuty セキュリティエージェント」を参照してください。	2024 年 2 月 16 日
Runtime Monitoring の更新された機能	Runtime Monitoring は、既存のセキュリティエージェントバージョン 1.4.1 で最新の Kubernetes バージョン 1.29 をサポートしています。この Kubernetes バージョンのローンチからサポートが提供されています。サポートされている Kubernetes バージョンの詳細については、「GuardDuty セキュリティエージェントでサポートされている Kubernetes のバージョン」を参照してください。	2024 年 2 月 16 日

Runtime Monitoring の更新された機能 - リージョナルな可用性	GuardDuty Runtime Monitoring で、同じ AWS Organizations内の共有 Amazon VPC がサポートされるようになりました。GuardDuty サービスにリンクされたロール (SLR) の新しいアクセス許可 `organizations:DescribeOrganization` では、共有 Amazon VPC アカウントの組織 ID を取得してエンドポイントポリシーを設定できます。Runtime Monitoring で共有 Amazon VPC エンドポイントを使用するための前提条件については、共有 Amazon VPC のサポートを参照してください。この機能は、GuardDuty が Runtime Monitoring をサポートするすべてのリージョンで使用できます。	2024 年 2 月 12 日

Runtime Monitoring の更新された機能 - リージョナルな可用性	GuardDuty Runtime Monitoring で、同じ AWS Organizations内の共有 Amazon VPC がサポートされるようになりました。GuardDuty サービスにリンクされたロール (SLR) の新しいアクセス許可 `organizations:DescribeOrganization` では、共有 Amazon VPC アカウントの組織 ID を取得してエンドポイントポリシーを設定できます。Runtime Monitoring で共有 Amazon VPC エンドポイントを使用するための前提条件については、共有 Amazon VPC のサポートを参照してください。現在、この機能は AWS リージョンの一部で利用できます。詳細については、「リージョンとエンドポイント」を参照してください。	2024 年 2 月 9 日
新しい のサポートで機能を更新 AWS リージョン – Malware Protection for EC2	Malware Protection for EC2 は、米国西部 (オレゴン) リージョン AWS マネージドキーで で暗号化された EBS ボリュームのスキャンをサポートするようになりました。	2024 年 2 月 6 日

新しい のサポートで機能を更新 AWS リージョン – Malware Protection for EC2	Malware Protection for EC2 は、次の AWS マネージドキー で で暗号化された EBS ボリュームのスキャンをサポートするようになりましたAWS リージョン。 • アジアパシフィック (シンガポール) (ap-southeast-1) • 欧州 (フランクフルト) (eu-central-1) • アジアパシフィック (大阪) (ap-northeast-3) • 米国東部 (オハイオ) (us-east-2) • 欧州 (ミラノ) (eu-south-1) • アジアパシフィック (東京) (ap-northeast-1) • アジアパシフィック (ソウル) (ap-northeast-2) • カナダ (中部) (ca-central-1) • 欧州 (アイルランド) (eu-west-1) • 米国東部 (バージニア北部) (us-east-1)	2024 年 2 月 5 日

Runtime Monitoring の更新された機能	GuardDuty Runtime Monitoring が Amazon EC2 インスタンス用の新しい GuardDuty セキュリティエージェントバージョン (v1.0.2) をリリースしました。このエージェントバージョンには、最新の Amazon ECS AMI のサポートが含まれています。エージェントリリース履歴の詳細については、「Amazon EC2 インスタンス用の GuardDuty セキュリティエージェント」を参照してください。	2024 年 2 月 2 日

新しい のサポートで機能を更新 AWS リージョン – Malware Protection for EC2	Malware Protection for EC2 は、次の AWS マネージドキー で で暗号化された Amazon EBS ボリュームのスキャンをサポートするようになりました AWS リージョン。 • 欧州 (ロンドン) (eu-west-2) • 欧州 (ストックホルム) (eu-north-1) • アジアパシフィック (香港) (ap-east-1) • アフリカ (ケープタウン) (af-south-1) • 中東 (バーレーン) (me-south-1) • アジアパシフィック (ハイデラバード) (ap-south-2) • 欧州 (スペイン) (eu-south-2) • アジアパシフィック (メルボルン) (ap-southeast-4) • アジアパシフィック (シドニー) (ap-southeast-2) • イスラエル (テルアビブ) (il-central-1)	2024 年 1 月 31 日

でアカウントの管理を更新 AWS Organizations	「アカウントを管理する AWS Organizations」のコンテンツを再編成し、委任 GuardDuty 管理者アカウントを変更する手順を追加し、GuardDuty 管理者アカウントとメンバーアカウントの関係について」を更新しました。	2024 年 1 月 30 日
新しい のサポートで機能を更新 AWS リージョン	Malware Protection for EC2 は、次の AWS マネージドキー で で暗号化された EBS ボリュームのスキャンをサポートするようになりました AWS リージョン。 • アジアパシフィック (ジャカルタ) (ap-southeast-3) • 米国西部 (北カリフォルニア) (us-west-1) • 中東 (アラブ首長国連邦) (me-central-1) • 欧州 (チューリッヒ) (eu-central-2) • アジアパシフィック (ムンバイ) (ap-south-1) • 南米 (サンパウロ) (sa-east-1)	2024 年 1 月 29 日

Malware Protection for EC2 の更新された機能	Malware Protection for EC2 で、 AWS マネージドキーを使用して暗号化された EBS ボリュームのスキャンがサポートされるようになりました。Malware Protection for EC2 のサービスにリンクされたロール (SLR) に、GetSnapshotBlock と ListSnapshotBlocks の 2 つの新しいアクセス許可があります。これらのアクセス許可は、GuardDuty が AWS マネージドキーから EBS ボリュームのスナップショット (を使用して暗号化) を取得し AWS アカウント 、マルウェアスキャンを開始する前に GuardDuty サービスアカウントにコピーするのに役立ちます。現在、この機能は欧州 (パリ) (eu-west-3) でのみ利用できます。詳細については、「マルウェアスキャンでサポートされている Amazon EBS ボリューム」を参照してください。	2024 年 1 月 25 日

Runtime Monitoring の更新された機能	GuardDuty Runtime Monitoring が、一般的なパフォーマンスの調整と強化を備えた新しい GuardDuty セキュリティエージェントバージョン (v1.0.1) をリリースしました。エージェントリリース履歴の詳細については、「Amazon EC2 インスタンス用の GuardDuty セキュリティエージェント」を参照してください。	2024 年 1 月 23 日
Runtime Monitoring の更新された機能	Runtime Monitoring が Amazon EKS リソース用の新しいエージェントバージョン 1.4.1 をリリースしました。詳細については、「EKS アドオンエージェントのリリース履歴」を参照してください。	2024 年 1 月 16 日
Runtime Monitoring が Amazon EKS リソース用の新しいエージェント v1.4.0 をリリース	Runtime Monitoring が Amazon EKS リソース用の新しいエージェントバージョン 1.4.0 をリリースしました。詳細については、「EKS アドオンエージェントのリリース履歴」を参照してください。	2023 年 12 月 21 日

欧州 (チューリッヒ) 、欧州 (スペイン) 、アジアパシフィック (ハイデラバード) 、アジアパシフィック (メルボルン) 、イスラエル (テルアビブ) に S3 および AWS CloudTrail 機械学習 (ML) ベースの検出結果タイプを追加しました。	GuardDuty の異常検出機械学習 (ML) モデルを使用して異常な動作を特定する次の S3 および CloudTrail の検出結果が、欧州 (チューリッヒ)、欧州 (スペイン)、アジアパシフィック (ハイデラバード)、アジアパシフィック (メルボルン)、イスラエル (テルアビブ) の各リージョンで利用可能になりました。 • Discovery:S3/AnomalousBehavior • Impact:S3/AnomalousBehavior.Write • Impact:S3/AnomalousBehavior.Delete • Impact:S3/AnomalousBehavior.Permission • Exfiltration:S3/AnomalousBehavior • Exfiltration:IAMUser/AnomalousBehavior • Impact:IAMUser/AnomalousBehavior • CredentialAccess:IAMUser/AnomalousBehavior • DefenseEvasion:IAMUser/AnomalousBehavior • InitialAccess:IAMUser/AnomalousBehavior • Persistence:IAMUser/AnomalousBehavior	2023 年 12 月 21 日

	• PrivilegeEscalation:IAMUser/AnomalousBehavior • Discovery:IAMUser/AnomalousBehavior	
GuardDuty は、 を通じて 50,000 個のメンバーアカウントをサポートします。 AWS Organizations	委任 GuardDuty 管理者は、 を通じて最大 50,000 個のメンバーアカウントを管理できるようになりました AWS Organizations。これには、招待によって GuardDuty 管理者アカウントに関連付けた最大 5000 件のメンバーアカウントも含まれます。	2023 年 12 月 20 日
GuardDuty Runtime Monitoring のサポートが 19 に拡張されました AWS リージョン	Runtime Monitoring が利用可能な地域は次の通りです: アジアパシフィック (ジャカルタ)、欧州 (パリ)、アジアパシフィック (大阪)、アジアパシフィック (ソウル)、中東 (バーレーン)、欧州 (スペイン)、アジアパシフィック (ハイデラバード)、アジアパシフィック (メルボルン)、イスラエル (テルアビブ)、米国西部 (北カリフォルニア)、欧州 (ロンドン)、アジアパシフィック (香港)、欧州 (ミラノ)、中東 (UAE)、南米 (サンパウロ)、アジアパシフィック (ムンバイ)、カナダ (中部)、アフリカ (ケープタウン)、欧州 (チューリッヒ)。	2023 年 12 月 6 日

GuardDuty の Runtime Monitoring 機能を拡張	Amazon EKS クラスターの脅威検出に加え、GuardDuty は、Amazon ECS ワークロードの脅威検出を可能にする Runtime Monitoring の一般提供、および Amazon EC2 インスタンスの脅威検出を可能にするプレビューリリースを発表します。現在 Runtime Monitoring がサポートされている AWS リージョン の詳細については、「リージョンとエンドポイント」をご確認ください。	2023 年 11 月 26 日

Amazon GuardDuty によりサービスにリンクされたロール (SLR) が更新されました	GuardDuty では新たな許可が追加されたことで、Amazon ECS アクションを使用した Amazon ECS クラスターの管理や情報の取得と、`guarddutyActivate` を使用した Amazon ECS アカウント設定の管理ができるようになりました。Amazon ECS に関するアクションでも、GuardDuty に関連するタグの情報も取得します。 • GuardDuty の Runtime Monitoring 機能拡張の一環として、以下の権限が追加されました。 `"ecs:ListClusters", "ecs:DescribeClusters", "ecs:PutAccountSettingDefault"`	2023 年 11 月 26 日
AWS 管理ポリシーを更新しました	GuardDuty は新しいアクセス許可である `organizations:ListAccounts` を AWS マネージドポリシー: AmazonGuardDutyFullAccess と AmazonGuardDutyReadOnlyAccess に追加しました。	2023 年 11 月 16 日

GuardDuty では、EKS 監査ログモニタリングを使用する新たな検出結果タイプがリリースされました。	EKS 監査ログモニタリングでは、アジアパシフィック (メルボルン) (ap-southeast-4) で以下の検出結果タイプがサポートされるようになりました。 • CredentialAccess:Kubernetes/AnomalousBehavior.SecretsAccessed • PrivilegeEscalation:Kubernetes/AnomalousBehavior.RoleBindingCreated • Execution:Kubernetes/AnomalousBehavior.ExecInPod • PrivilegeEscalation:Kubernetes/AnomalousBehavior.WorkloadDeployed!PrivilegedContainer • PrivilegeEscalation:Kubernetes/AnomalousBehavior.WorkloadDeployed!ContainerWithSensitiveMount • Execution:Kubernetes/AnomalousBehavior.WorkloadDeployed • PrivilegeEscalation:Kubernetes/AnomalousBehavior.RoleCreated • Discovery:Kubernetes/AnomalousBehavior.PermissionChecked	2023 年 11 月 11 日

GuardDuty では、EKS 監査ログモニタリングを使用する新たな検出結果タイプがリリースされました。	EKS 監査ログモニタリングでは、アジアパシフィック (ハイデラバード) (ap-south-2)、欧州 (チューリッヒ) (eu-central-2)、欧州 (スペイン) (eu-south-2)で以下の検出結果タイプがサポートされるようになりました。 • CredentialAccess:Kubernetes/AnomalousBehavior.SecretsAccessed • PrivilegeEscalation:Kubernetes/AnomalousBehavior.RoleBindingCreated • Execution:Kubernetes/AnomalousBehavior.ExecInPod • PrivilegeEscalation:Kubernetes/AnomalousBehavior.WorkloadDeployed!PrivilegedContainer • PrivilegeEscalation:Kubernetes/AnomalousBehavior.WorkloadDeployed!ContainerWithSensitiveMount • Execution:Kubernetes/AnomalousBehavior.WorkloadDeployed • PrivilegeEscalation:Kubernetes/AnomalousBehavior.RoleCreated	2023 年 11 月 10 日

- Discovery:Kubernetes/AnomalousBehavior.PermissionChecked

GuardDuty では、EKS 監査ログモニタリングを使用する新たな検出結果タイプがリリースされました。	EKS 監査ログモニタリングでは、以下の検出結果タイプがサポートされるようになりました。次の検出結果タイプは、アジアパシフィック (ハイデラバード) (`ap-south-2`)、欧州 (チューリッヒ) (`eu-central-2`)、欧州 (スペイン) (`eu-south-2`)、アジアパシフィック (メルボルン)(`ap-southeast-4`)では現在ご利用いただけません。 • CredentialAccess:Kubernetes/AnomalousBehavior.SecretsAccessed • PrivilegeEscalation:Kubernetes/AnomalousBehavior.RoleBindingCreated • Execution:Kubernetes/AnomalousBehavior.ExecInPod • PrivilegeEscalation:Kubernetes/AnomalousBehavior.WorkloadDeployed!PrivilegedContainer • PrivilegeEscalation:Kubernetes/AnomalousBehavior.WorkloadDeployed!ContainerWithSensitiveMount • Execution:Kubernetes/AnomalousBehavior.WorkloadDeployed	2023 年 11 月 8 日

	• PrivilegeEscalation:Kubernetes/AnomalousBehavior.RoleCreated • Discovery:Kubernetes/AnomalousBehavior.PermissionChecked	
EKS Runtime Monitoring が新しいエージェント v1.3.1 をリリース	EKS Runtime Monitoring が重要なセキュリティパッチと更新を含む新しいエージェントバージョン 1.3.1 をリリースしました。	2023 年 10 月 23 日
検出結果の新しいフィルター属性	GuardDuty に、生成された検出結果をフィルタリングする新しい基準が追加されました。DNS リクエストドメインサフィックスは、GuardDuty がその検出結果を生成する原因となったアクティビティに関連する 2 番目および最上位レベルのドメインを示します。	2023 年 10 月 17 日
EKS Runtime Monitoring が Kubernetes バージョン 1.28 をサポートする新しいエージェント v1.3.0 をリリース	EKS Runtime Monitoring が、Kubernetes バージョン 1.28 をサポートする新しいエージェントバージョン 1.3.0 をリリースしました。Ubuntu のサポートを追加しました。詳細については、「EKS アドオンエージェントのリリース履歴」を参照してください。	2023 年 10 月 5 日

アジアパシフィック (ジャカルタ) および中東 (アラブ首長国連邦) リージョンに S3 および AWS CloudTrail 機械学習 (ML) ベースの検出結果タイプを追加	アジアパシフィック (ジャカルタ) および中東 (アラブ首長国連邦) のリージョンで、GuardDuty の異常検出の機械学習 (ML) モデルを使用して異常動作を特定する、以下の S3 および CloudTrail の検出結果が利用可能になりました。 • Discovery:S3/AnomalousBehavior • Impact:S3/AnomalousBehavior.Write • Impact:S3/AnomalousBehavior.Delete • Impact:S3/AnomalousBehavior.Permission • Exfiltration:S3/AnomalousBehavior • Exfiltration:IAMUser/AnomalousBehavior • Impact:IAMUser/AnomalousBehavior • CredentialAccess:IAMUser/AnomalousBehavior • DefenseEvasion:IAMUser/AnomalousBehavior • InitialAccess:IAMUser/AnomalousBehavior • Persistence:IAMUser/AnomalousBehavior • PrivilegeEscalation:IAMUser/AnomalousBehavior	2023 年 9 月 20 日

	• Discovery:IAMUser/AnomalousBehavior	
GuardDuty EKS Runtime Monitoring にクラスターレベルでの GuardDuty セキュリティエージェントの管理を導入	個々の EKS クラスターの GuardDuty セキュリティエージェントを管理して、これらの選択的なクラスターのみのランタイムイベントをモニタリングするためのサポートが、EKS Runtime Monitoring に追加されました。EKS Runtime Monitoring は、タグのサポートによりこの機能を拡張します。	2023 年 9 月 13 日
GuardDuty Malware Protection for EC2 のサポートを延長 AWS リージョン	Malware Protection for EC2 は現在、アジアパシフィック (ハイデラバード)、アジアパシフィック (メルボルン)、欧州 (チューリッヒ)、および欧州 (スペイン) で利用可能です。	2023 年 9 月 11 日

GuardDuty がイスラエル (テルアビブ) リージョンで利用可能に	GuardDuty AWS リージョンが利用可能になった のリストにイスラエル (テルアビブ) リージョンを追加しました。イスラエル (テルアビブ) リージョンでは、次の保護プランも利用できます。 • EKS Protection には、EKS 監査ログのモニタリングと EKS ランタイムモニタリングの両方が含まれます。 • Lambda Protection. • Malware Protection for EC2. • S3 Protection. イスラエル (テルアビブ) リージョンで利用可能な保護プランの詳細については、「リージョンとエンドポイント」を参照してください。	2023 年 8 月 24 日
GuardDuty が保護プランレベルで組織の自動有効化の設定を追加	リージョンの保護プランの組織設定を更新します。可能な設定オプションは、すべてのアカウントのために有効にする、新しいアカウントのために自動的に有効にする、組織内のいずれのアカウントのためにも自動的に有効にしない、のいずれかです。	2023 年 8 月 16 日

GuardDuty の異常検出機械学習 (ML) モデルを使用して異常な動作を特定する S3 検出結果タイプがアジアパシフィック (大阪) で利用可能に	次の検出結果タイプがアジアパシフィック (大阪) リージョンで利用できるようになりました。 • Discovery:S3/AnomalousBehavior • Impact:S3/AnomalousBehavior.Write • Impact:S3/AnomalousBehavior.Delete • Impact:S3/AnomalousBehavior.Permission • Exfiltration:S3/AnomalousBehavior	2023 年 8 月 10 日
EKS ランタイムモニタリングがアジアパシフィック (メルボルン) で利用可能に	GuardDuty EKS Protection 内の EKS Runtime Monitoring は、 AWS 環境内の Amazon EKS クラスターのランタイム脅威検出を提供します。現在、アジアパシフィック (メルボルン) リージョンでサポートされています。	2023 年 8 月 8 日
GuardDuty によるマルウェアスキャンを呼び出す GuardDuty 検出結果のリストを更新	特定の EKS ランタイムモニタリングの検出結果タイプで、GuardDuty が開始するマルウェアスキャンを AWS アカウント で呼び出すことができるようになりました。	2023 年 7 月 19 日

GuardDuty は、 を通じて 10,000 個のメンバーアカウントをサポートします。 AWS Organizations	GuardDuty 管理者アカウントは、最大 10,000 個のメンバーアカウントを管理できるようになりました AWS Organizations。これには、招待によって GuardDuty 管理者アカウントに関連付けた最大 5000 件のメンバーアカウントも含まれます。	2023 年 6 月 29 日
EKS Runtime Monitoring が 3 つの新しい検出結果タイプを発表	EKS Runtime Monitoring は、プロセスインジェクション技術に基づく 3 つの新しい検出結果タイプをサポートしています。その新しい検出結果タイプとは、DefenseEvasion:Runtime/ProcessInjection.Proc、DefenseEvasion:Runtime/ProcessInjection.Ptrace、およびDefenseEvasion:Runtime/ProcessInjection.VirtualMemoryWrite です。	2023 年 6 月 22 日
EKS Runtime Monitoring が Kubernetes バージョン 1.27 をサポートする新しいエージェント v1.2.0 をリリース	EKS Runtime Monitoring が、ARM64 ベースのインスタンスもサポートする新しいエージェントバージョン 1.2.0 をリリースしました。Bottlerocket のサポートが追加されました。詳細については、「EKS アドオンエージェントのリリース履歴」を参照してください。	2023 年 6 月 16 日

GuardDuty コンソールに検出結果の概要を表示	GuardDuty コンソールの概要ダッシュボードには、GuardDuty の検出結果が集計されて表示されます。現在、ダッシュボードには、現在のリージョンのアカウント (または GuardDuty 管理者アカウントの場合はメンバーアカウント) に対して生成された過去 10,000 件の検出結果について、さまざまなウィジェットを使用してデータが表示されます。	2023 年 6 月 12 日
EKS Audit Log Monitoring は現在、アジアパシフィック (ハイデラバード)、アジアパシフィック (メルボルン)、欧州 (チューリッヒ) および欧州 (スペイン)リージョンでご利用可能	アカウントに対して (EKS Protection で) EKS 監査ログのモニタリングを有効にして、Amazon EKS クラスターからの EKS 監査ログをモニタリングし、悪意のあるアクティビティや疑わしいアクティビティがないか分析します。	2023 年 6 月 1 日
EKS 監査ログのモニタリングが中東 (UAE) で利用可能になりました	EKS 監査ログのモニタリングが中東 (UAE) で利用可能になりました。アカウントに対して EKS 監査ログのモニタリングを有効にして、Amazon EKS クラスターからの EKS 監査ログをモニタリングし、悪意のあるアクティビティや疑わしいアクティビティがないか分析します。	2023 年 5 月 3 日

GuardDuty Malware Protection for EC2 がオンデマンドマルウェアスキャンを発表しました	Malware Protection for EC2 は、Amazon EC2 インスタンスおよびコンテナワークロードにアタッチされた Amazon EBS ボリュームにマルウェアが存在する可能性があるかどうかを検出するのに役立ちます。GuardDuty によるスキャンとオンデマンドのスキャンの 2 つのスキャンタイプが提供されるようになりました。GuardDuty 実行型マルウェアスキャンでは、GuardDuty 実行型マルウェアスキャンを呼び出す検出結果の 1 つが GuardDuty によって生成された場合にのみ、Amazon EBS ボリュームで自動的にエージェントレススキャンが開始されます。Amazon EC2 インスタンスに関連付けられた Amazon リソースネーム (ARN) を指定することで、アカウントの Amazon EC2 インスタンスのオンデマンドマルウェアスキャンを開始できます。両方のスキャンタイプの違いについての詳細は、「Malware Protection for EC2」を参照してください。 • GuardDuty 実行型マルウェアスキャン • オンデマンドのマルウェアスキャン	2023 年 4 月 27 日

GuardDuty が Lambda Protection を発表します	Lambda Protection は、 AWS Lambda 関数の潜在的なセキュリティ脅威を特定するのに役立ちます。 • Lambda Protection の検出結果タイプ • 侵害された可能性のある Lambda 関数の修復	2023 年 4 月 20 日
GuardDuty が、アジアパシフィック (メルボルン) リージョンで利用可能になりました	GuardDuty AWS リージョンが利用可能な のリストにアジアパシフィック (メルボルン) を追加しました。このリージョンで利用可能な機能については、「リージョンとエンドポイント」を参照してください。	2023 年 4 月 19 日
GuardDuty に 3 つの新しい EC2 検出結果タイプが追加されました	GuardDuty では、外部 DNS リゾルバーや暗号化された DNS テクノロジーの使用を検出する新しい検出結果タイプが導入されています。これらの検出結果タイプがサポートされている AWS リージョン場所については、「リージョンとエンドポイント」を参照してください。 • DefenseEvasion:EC2/UnusualDNSResolver • DefenseEvasion:EC2/UnusualDoHActivity • DefenseEvasion:EC2/UnusualDoTActivity	2023 年 4 月 5 日

GuardDuty は EKS Protection の EKS Runtime Monitoring を発表します	EKS Protection 内の EKS Runtime Monitoring は、 AWS 環境内の Amazon EKS クラスターのランタイム脅威検出を提供します。Amazon EKS アドオンエージェント (`aws-guardduty-agent`) を使用して、EKS ワークロードからランタイムイベントを収集します。GuardDuty はこれらのランタイムイベントを受け取ると、それらをモニタリングおよび分析して、潜在的に疑わしいセキュリティ脅威を特定します。詳細については、「検出結果の詳細」と「EKS Runtime Monitoring の検出結果タイプ」を参照してください。	2023 年 3 月 30 日

GuardDuty に新機能が追加されます - autoEnableOrganizationMembers	Amazon GuardDuty には、GuardDuty 管理者アカウントが監査を行い、(必要に応じて) 組織の ALL メンバーに対して GuardDuty が有効になっていることを強制するのに役立つ新しい組織設定オプションが追加されています。現在のベストプラクティスは、autoEnable の代わりに autoEnableOrganizationMembers を使用することです。autoEnable は非推奨ですが、まだサポートされています。次の API はこの新機能の影響を受けます。 • DescribeOrganizationConfiguration • UpdateOrganizationConfiguration • DisassociateMembers • DeleteMembers • DisassociateFromAdministratorAccount • StopMonitoringMembers	2023 年 3 月 23 日

Amazon GuardDuty の RDS Protection 機能が一般に利用可能になりました	GuardDuty RDS Protection は、RDS ログインアクティビティをモニタリングおよびプロファイリングして、Amazon Aurora データベースインスタンスでの疑わしいログイン動作を特定します。RDS Protection をサポートする AWS リージョン については、「リージョンとエンドポイント」を参照してください。	2023 年 3 月 16 日
GuardDuty が機能のアクティベーションを発表しました	これまで GuardDuty API では、機能とデータソースの両方の設定が可能でしたが、今後はすべての新しい GuardDuty 保護タイプがデータソースではなく機能として設定されるようになります。GuardDuty は引き続き API 経由のデータソースをサポートしていますが、新しい API は追加しません。機能の有効化は、GuardDuty または GuardDuty 内の保護タイプを有効にするために使用される API の動作に影響します。API、SDK、または CFN テンプレートを使用して GuardDuty アカウントを管理している場合は、「2023 年 3 月の GuardDuty API の変更点」を参照してください。	2023 年 3 月 16 日

GuardDuty Malware Protection for EC2 が中東 (UAE) リージョンで利用可能になりました	GuardDuty の Malware Protection for EC2 機能は中東 (UAE) リージョンでサポートされています。詳細については、「リージョンとエンドポイント」を参照してください。	2023 年 3 月 13 日
Amazon GuardDuty によりサービスにリンクされたロール (SLR) が更新されました	GuardDuty は、今後予定されている GuardDuty EKS Runtime Monitoring 機能をサポートするために、次の新しい許可を追加しました。 • Amazon EKS アクションを使用して、EKS クラスターに関する情報を管理および取得し、EKS クラスター上の EKS アドオンを管理します。EKS アクションは、GuardDuty に関連するタグに関する情報も取得します。 `"eks:ListClusters", "eks:DescribeCluster", "ec2:DescribeVpcEndpointServices", "ec2:DescribeSecurityGroups"`	2023 年 3 月 8 日
Amazon GuardDuty によりサービスにリンクされたロール (SLR) が更新されました	GuardDuty SLR が更新され、Malware Protection for EC2 を有効にした後で Malware Protection for EC2 SLR を作成できるようになりました。	2023 年 2 月 21 日

GuardDuty には TLS v1.2 以降が必要です	AWS リソースと通信するために、GuardDuty は TLS v1.2 以降を必要とし、サポートしています。詳細については、「データの保護」および「インフラストラクチャのセキュリティ」を参照してください。	2023 年 2 月 14 日
GuardDuty が、アジアパシフィック (ハイデラバード) リージョンで利用可能になりました	GuardDuty AWS リージョンが利用可能な のリストにアジアパシフィック (ハイデラバード) リージョンを追加しました。詳細については、「リージョンとエンドポイント」を参照してください。	2023 年 2 月 14 日
Amazon GuardDuty ユーザーガイドは IAM のベストプラクティスに沿ったものです	IAM ベストプラクティスに沿ってガイドを更新しました。詳細については、「IAM でのセキュリティのベストプラクティス」を参照してください。	2023 年 2 月 10 日
GuardDuty が欧州 (スペイン) リージョンで利用可能になりました	GuardDuty AWS リージョンが利用可能な のリストに欧州 (スペイン) を追加しました。詳細については、「リージョンとエンドポイント」を参照してください。	2023 年 2 月 8 日
GuardDuty が欧州 (チューリッヒ) リージョンで利用可能になりました	GuardDuty AWS リージョンが利用可能な のリストに欧州 (チューリッヒ) を追加しました。詳細については、「リージョンとエンドポイント」を参照してください。	2022 年 12 月 12 日

新機能「GuardDuty RDS Protection」のプレビューリリース	GuardDuty RDS Protection は、RDS ログインアクティビティをモニタリングおよびプロファイリングして、Amazon Aurora データベースインスタンスでの疑わしいログイン動作を特定します。現在、5 つの AWS リージョンでプレビューリリースを利用できます。詳細については、「リージョンとエンドポイント」を参照してください。	2022 年 11 月 30 日
GuardDuty が中東 (UAE) リージョンで利用可能になりました	GuardDuty AWS リージョンが利用可能な のリストに中東 (UAE) を追加しました。詳細については、「リージョンとエンドポイント」を参照してください。	2022 年 10 月 6 日

新機能「GuardDuty Malware Protection for EC2」のコンテンツを追加しました	GuardDuty Malware Protection for EC2 は Amazon GuardDuty の拡張機能です (オプション)。GuardDuty はリスクのあるリソースを識別しますが、Malware Protection for EC2 は侵害の原因となる可能性のあるマルウェアを検出します。Malware Protection for EC2 を有効にすると、GuardDuty によって Amazon EC2 インスタンスまたはマルウェアを示すコンテナワークロードで疑わしい動作が検出されると、GuardDuty Malware Protection for EC2 が、影響を受ける EC2 インスタンスまたはコンテナワークロードにアタッチされた EBS ボリュームでエージェントレススキャンを開始し、マルウェアの存在を検出します。Malware Protection for EC2 の仕組みと、この機能の設定については、「GuardDuty Malware Protection for EC2」を参照してください。 • Malware Protection for EC2 の検出結果については、「検出結果の詳細」を参照してください。 • 侵害された EC2 インスタンスとスタンドアロンコンテナの修復については、	2022 年 7 月 26 日

	「GuardDuty によって検出されたセキュリティ問題の修復」を参照してください。 • マルウェアスキャンの CloudWatch ログの監査と、マルウェアスキャン中にリソースがスキップされる理由については、「CloudWatch のログとスキップの理由を理解する」を参照してください。 • 脅威検出の誤検出については、「Malware Protection for EC2 の誤検出の報告」を参照してください。	
廃止された検出結果のタイプの 1 つ	Exfiltration:S3/ObjectRead.Unusual は廃止されました。	2022 年 7 月 5 日

GuardDuty の異常検出機械学習 (ML) モデルを使用して、異常な動作を識別する新しい S3 の検出結果タイプを追加しました。	次の新しい S3 の検出結果タイプを追加しました。これらの検出結果タイプでは、API リクエストが IAM エンティティを異常な方法で呼び出したかどうかを識別します。機械学習モデルは、アカウント内のすべての API リクエストを評価し、攻撃者が使用するテクニックに関連する異常なイベントを特定します。これらの新しい検出結果の詳細については、「S3 の検出結果タイプ」を参照してください。 • Discovery:S3/AnomalousBehavior • Impact:S3/AnomalousBehavior.Write • Impact:S3/AnomalousBehavior.Delete • Impact:S3/AnomalousBehavior.Permission • Exfiltration:S3/AnomalousBehavior	2022 年 7 月 5 日

GuardDuty に GuardDuty EKS Protection に関する内容が追加されました	GuardDuty は EKS 監査ログのモニタリングを通じて Amazon EKS リソースの検出結果を生成できるようになりました。この機能を設定する方法については、「Amazon GuardDuty の EKS Protection」を参照してください。GuardDuty が Amazon EKS リソースに関して生成できる検出結果のリストについては、「GuardDuty Kubernetes 検出結果タイプ」を参照してください。「Kubernetes の検出結果の修復ガイド」でこれらの検出結果の修復をサポートするために、新しい修復ガイダンスが追加されました。	2022 年 1 月 25 日
1 つの新しい検出結果が追加されました	新しい検出結果 UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.InsideAWS が追加されました。この検出結果は、インスタンス認証情報が AWS 環境外の AWS アカウントによってアクセスされたときに通知します。	2022 年 1 月 20 日

log4j に関連する問題の特定に役立つ検出結果タイプを更新しました	Amazon GuardDuty は、CVE-2021-44228 および CVE-2021-45046 に関連する問題を特定して優先順位を付けるのに役立つように、次の検出結果タイプを更新しました: Backdoor:EC2/C&CActivity.B、Backdoor:EC2/C&CActivity.B!DNS、Behavior:EC2/NetworkPortUnusual。	2021 年 12 月 22 日
検出結果の変更	UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration は UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.OutsideAWS に変更されました。この検出結果の改善バージョンでは、オンプレミスネットワークでの送信済みトラフィックの検出結果を減らすため、認証情報を使用する典型的な場所を学習します。UnauthorizedAccess:IAMUser/InstanceCredentialExfiltration.OutsideAWS	2021 年 9 月 7 日
GuardDuty SLR にアップデートする	GuardDuty SLRは、検出結果の精度を向上させる新しいアクションで更新されました。	2021 年 8 月 3 日
各検出結果タイプのデータソース情報が追加されました。	GuardDuty がその検出結果の生成に使用するデータソースに関する情報が、検出結果の説明に含まれるようになりました。	2021 年 5 月 10 日

廃止された 13 の検出結果タイプ	13 の検出結果は廃止となり、新たに異常動作の検出結果に置き換えられました (Persistence:IAMUser/NetworkPermissions、Persistence:IAMUser/ResourcePermissions、Persistence:IAMUser/UserPermissions、PrivilegeEscalation:IAMUser/AdministrativePermissions、Recon:IAMUser/NetworkPermissions、Recon:IAMUser/ResourcePermissions、Recon:IAMUser/UserPermissions、ResourceConsumption:IAMUser/ComputeResources、Stealth:IAMUser/LoggingConfigurationModified、Discovery:S3/BucketEnumeration.Unusual、Impact:S3/ObjectDelete.Unusual、Impact:S3/PermissionsModification.Unusual、および UnauthorizedAccess:IAMUser/ConsoleLogin)。	2021 年 3 月 12 日

異常動作の 8 つの新しい検出結果タイプが追加されました。	IAM プリンシパルの異常動作に基づいて 8 つの新しい IAMUser 検出結果タイプが追加されました (CredentialAccess:IAMUser/AnomalousBehavior、DefenseEvasion:IAMUser/AnomalousBehavior、Discovery:IAMUser/AnomalousBehavior、Exfiltration:IAMUser/AnomalousBehavior、Impact:IAMUser/AnomalousBehavior、InitialAccess:IAMUser/AnomalousBehavior、Persistence:IAMUser/AnomalousBehavior、PrivilegeEscalation:IAMUser/AnomalousBehavior)。	2021 年 3 月 12 日
ドメインの評判に基づく EC2 の検出結果が追加されました。	ドメインの評判に基づく、新しい 4 つのインパクト検出結果タイプが追加されました。Impact:EC2/AbusedDomainRequest.Reputation 、Impact:EC2/BitcoinDomainRequest.Reputation、Impact:EC2/MaliciousDomainRequest.Reputation。また、新しい C&CAtivity の EC2 検出結果も追加しました。Impact:EC2/SuspiciousDomainRequest.Reputation	2021 年 1 月 27 日

4 つの新しい検出結果タイプが追加されました。	新しく 3 つの S3 MaliciousIPCaller の検出結果が追加されました。Discovery:S3/MaliciousIPCaller、Exfiltration:S3/MaliciousIPCaller、Impact:S3/MaliciousIPCaller。また、新しい C&CAtivity の EC2 検出結果も追加しました。Backdoor:EC2/C&CActivity.B	2020 年 12 月 21 日
UnauthorizedAccess:EC2/TorIPCaller 検出結果タイプは廃止されました。	UnauthorizedAccess:EC2/TorIPCaller 検出結果タイプは GuardDuty で廃止されました。詳細はこちら。	2020 年 10 月 1 日
Impact:EC2/WinRmBruteForce 検出結果タイプが追加されました。	新しい Impact 検出結果 Impact:EC2/WinRmBruteForce が追加されました。詳細はこちら。	2020 年 9 月 17 日
Impact:EC2/PortSweep 検出結果タイプが追加されました。	新しい Impact 検出結果 Impact:EC2/PortSweep が追加されました。詳細はこちら。	2020 年 9 月 17 日
GuardDuty がアフリカ (ケープタウン) と欧州 (ミラノ) リージョンでご利用いただけるようになりました。	GuardDuty が利用可能な AWS リージョンのリストにアフリカ (ケープタウン) と欧州 (ミラノ) を追加しました。詳細はこちら	2020 年 7 月 31 日

GuardDuty コストをモニタリングする新しい使用状況の詳細が追加されました。	新しいメトリクスを使用して、管理しているアカウントおよびアカウントの GuardDuty 使用コストデータをクエリできるようになりました。使用コストの新しい概要は、https://console.aws.amazon.com/guardduty/ のコンソールで入手できます。詳細な情報には API からアクセスできます。	2020 年 7 月 31 日
GuardDuty で S3 データイベントモニタリングによる S3 Protection をするコンテンツが追加されました。	GuardDuty S3 Protection は、新しいデータソースとして S3 データプレーンのイベントをモニタリングすることが可能になりました。新しいアカウントでは、この機能が自動的に有効になります。GuardDuty を既に使用している場合は、自分またはメンバーアカウントに対して新しいデータソースを有効にできます。	2020 年 7 月 31 日
14 の新しい S3 検出結果が追加されました。	S3 コントロールプレーンとデータプレーンのソースに、14 の新しい S3 検出結果タイプが追加されました。	2020 年 7 月 31 日

S3 検出結果のサポートを追加し、2 つの既存の検出結果タイプ名を変更しました。	GuardDuty の検出結果には、S3 バケット関連検出結果の詳細が含まれるようになりました。S3 アクティビティに関連する既存の検出結果タイプの名前が変更されました。Policy:IAMUser/S3BlockPublicAccessDisabled は Policy:S3/BucketBlockPublicAccessDisabled に変更されました。Stealth:IAMUser/S3ServerAccessLoggingDisabled は Stealth:S3/ServerAccessLoggingDisabled に変更されました。	2020 年 5 月 28 日
AWS Organizations 統合のコンテンツを追加しました。	GuardDuty が AWS Organizations 委任管理者と統合され、組織内の GuardDuty アカウントを管理できるようになりました。GuardDuty 管理者アカウントとして委任された管理者を設定すると、委任された管理者のアカウントが管理する組織メンバーの GuardDuty を自動的に有効にできます。新しい AWS Organizations メンバーアカウントで GuardDuty が自動的に有効にできます。詳細はこちら。	2020 年 4 月 20 日
エクスポート検出結果機能のコンテンツが追加されました。	GuardDuty の検出結果のエクスポート機能を説明するコンテンツが追加されました。	2019 年 11 月 14 日

UnauthorizedAccess:EC2/MetadataDNSRebind 検出結果タイプが追加されました。	新しい Unauthorized 検出結果 UnauthorizedAccess:EC2/MetadataDNSRebind が追加されました。詳細はこちら。	2019 年 10 月 10 日
Stealth:IAMUser/S3ServerAccessLoggingDisabled 検出結果タイプが追加されました。	新しい Stealth 検出結果 Stealth:IAMUser/S3ServerAccessLoggingDisabled が追加されました。詳細はこちら。	2019 年 10 月 10 日
Policy:IAMUser/S3BlockPublicAccessDisabled 検出結果タイプが追加されました。	新しい Policy 検出結果 Policy:IAMUser/S3BlockPublicAccessDisabled が追加されました。詳細はこちら。	2019 年 10 月 10 日
Backdoor:EC2/XORDDOS 検出結果タイプは廃止されました。	Backdoor:EC2/XORDDOS 検出結果タイプは GuardDuty で廃止されました。詳細はこちら	2019 年 6 月 12 日
PrivilegeEscalation 検出結果タイプが追加されました。	PrivilegeEscalation 検出結果タイプでは、ユーザーが自分のアカウントの権限を昇格して過剰な特権を割り当てようした場合に検出されます。詳細はこちら	2019 年 5 月 14 日
GuardDuty が欧州 (ストックホルム) リージョンで利用可能になりました。	GuardDuty が利用可能な AWS リージョンのリストに欧州 (ストックホルム) を追加しました。詳細はこちら	2019 年 5 月 9 日

新しい検出結果タイプ Recon:EC2/PortProbeEMRUnprotectedPort が追加されました。	この検出結果は、EC2 インスタンス上にある EMR 関連のセンシティブなポートがブロックされておらず、悪意のあるホストが探していることを知らせるものです。詳細はこちら	2019 年 5 月 8 日
EC2 インスタンスがサービス拒否 (DoS) 攻撃に使われる可能性がある場合に検出される 5 つの新しい検出結果タイプが追加されました。	これらの検出結果から、環境内の EC2 インスタンスがサービス拒否 (DoS) 攻撃の実行に利用されている可能性があることがわかります。詳細はこちら	2019 年 3 月 8 日
新しい検出結果タイプ Policy:IAMUser/RootCredentialUsage が追加されました。	Policy:IAMUser/RootCredentialUsage 検出結果タイプは、のルートユーザーのサインイン認証情報 AWS アカウント が AWS サービスへのプログラムによるリクエストに使用されていることを知らせます。詳細はこちら	2019 年 1 月 24 日
UnauthorizedAccess:IAMUser/UnusualASNCaller 検出結果タイプは廃止されました	UnauthorizedAccess:IAMUser/UnusualASNCaller 検出結果タイプは廃止されました。今後は、他のアクティブな GuardDuty の検出結果タイプを介して異常なネットワークから呼び出されたアクティビティについて通知されます。生成される検出結果タイプは、異常なネットワークから呼び出された API のカテゴリに基づきます。詳細はこちら	2018 年 12 月 21 日

2 つの新しい検出結果タイプ PenTest:IAMUser/ParrotLinux および PenTest:IAMUser/PentooLinux が追加されました。	PenTest:IAMUser/ParrotLinux の検出結果タイプは、Parrot Security Linux を実行しているコンピュータで、自身の AWS アカウントの認証情報を使用して API コールが行われていることを通知します。PenTest:IAMUser/PentooLinux 検出結果タイプは、Pentoo Linux を実行しているマシンで、自身の AWS アカウントの認証情報を使用して API コールが行われていることを通知します。詳細はこちら	2018 年 12 月 21 日
Amazon GuardDuty Announcements SNS トピックのサポートが追加されました	GuardDuty のお知らせ用 SNS トピックをサブスクライブして、新しくリリースされた検出結果タイプ、既存の検出結果タイプの更新情報、その他機能の変更点に関する通知を受け取ることができるようになりました。Amazon SNS がサポートするすべての形式で通知を使用できます。詳細はこちら	2018 年 11 月 21 日

2 つの新しい検出結果タイプ UnauthorizedAccess:EC2/TorClient および UnauthorizedAccess:EC2/TorRelay が追加されました。	UnauthorizedAccess:EC2/TorClient 検出結果タイプは、AWS 環境内の EC2 インスタンスが Tor Guard または Authority ノードに接続していることを通知します。検出UnauthorizedAccess:EC2/TorRelay結果タイプは、 AWS 環境内の EC2 インスタンスが Tor リレーとして動作していることを示す方法で Tor ネットワークに接続していることを通知します。詳細はこちら	2018 年 11 月 16 日
新しい検出結果タイプ CryptoCurrency:EC2/BitcoinTool.B が追加されました。	この検出結果は、 AWS 環境内の EC2 インスタンスが Bitcoin または他の暗号通貨関連のアクティビティに関連付けられているドメイン名をクエリしていることを知らせるものです。詳細はこちら	2018 年 11 月 9 日
CloudWatch Events への通知の送信頻度を更新するサポートが追加されました	既存の検出結果の後続の発生について、CloudWatch Events に送信される通知の頻度を更新できるようになりました。有効な値は、15 分、1 時間、またはデフォルトの 6 時間です。詳細はこちら	2018 年 10 月 9 日
リージョンサポートが追加されました	AWS GovCloud (米国西部) のリージョンサポートを追加 詳細	2018 年 7 月 25 日

GuardDuty で CloudFormation StackSets のサポートを追加	Enable Amazon GuardDuty テンプレートを使用すると、複数のアカウントで GuardDuty を同時に有効にできます。詳細はこちら	2018 年 6 月 25 日
GuardDuty 自動アーカイブルールのサポートが追加されました	お客様は、検出結果の数を抑えられるよう、詳細な自動アーカイブルールを構築できるようになりました。自動アーカイブルールに一致する検出結果の場合、GuardDuty はそれらを自動的にアーカイブ済みとしてマークします。これにより、GuardDuty をさらに細かく調整して、現在の検出結果の表での関連検出結果のみを保持できます。詳細はこちら	2018 年 5 月 4 日
GuardDuty は欧州 (パリ) リージョンで利用できます	GuardDuty がこのリージョンでの継続的なセキュリティのモニタリングと脅威検出を拡張するため、欧州 (パリ) で利用可能になりました。詳細はこちら	2018 年 3 月 29 日
による GuardDuty 管理者アカウントとメンバーアカウントの作成 CloudFormation がサポートされるようになりました。	詳細については、「AWS::GuardDuty::master」および「AWS::GuardDuty::member」を参照してください。	2018 年 3 月 6 日

新しい 9 個の CloudTrail ベースの異常検出が追加されました。	これらの新しい検出結果タイプは、サポートされたすべてのリージョンの GuardDuty で自動的に有効になります。詳細はこちら	2018 年 2 月 28 日
3 つの新しい脅威インテリジェンス検出 (検出タイプ) を追加しました。	これらの新しい検出結果タイプは、サポートされたすべてのリージョンの GuardDuty で自動的に有効になります。詳細はこちら	2018 年 2 月 5 日
GuardDuty メンバーアカウントの制限の引き上げ。	このリリースで、 AWS アカウント (GuardDuty 管理者アカウント) ごとに最大 1,000 件の GuardDuty メンバーアカウントを追加することができます。詳細はこちら	2018 年 1 月 25 日
GuardDuty 管理者アカウントとメンバーアカウントの信頼できる IP リストと脅威リストのアップロードとさらなる管理の変更。	このリリースで、GuardDuty 管理者アカウントのユーザーが信頼できる IP リストと脅威リストをアップロードして管理することができます。GuardDuty のメンバーアカウントのユーザーは、リストのアップロードや管理を行うことはできません。管理者アカウントからアップロードされた信頼できる IP リストや脅威リストで、そのメンバーアカウントの GuardDuty 機能を与えました。詳細はこちら	2018 年 1 月 25 日

以前の更新

変更	説明	日付
初版発行	「Amazon GuardDuty ユーザーガイド」の初回リリース。	2017 年 11 月 28 日

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。