開発者ガイド

AWS WAF、 AWS Firewall Manager、および AWS Shield Advanced

AWS WAF、 AWS Firewall Manager、および AWS Shield Advanced: 開発者ガイド

Table of Contents

AWS WAF、 AWS Shield Advanced、および とは AWS Firewall Manager

AWS WAF、、AWS Shieldおよび をAWS Firewall Manager一緒に使用して包括的なセキュリティソリューションを作成できます。 AWS WAF は、エンドユーザーがアプリケーションに送信するウェブリクエストをモニタリングし、コンテンツへのアクセスを制御するために使用できるウェブアプリケーションファイアウォールです。Shield Advanced は、ネットワークレイヤーとトランスポートレイヤー (レイヤー 3 と 4)、およびアプリケーションレイヤー (レイヤー 7) で、 AWS リソースに対する分散型サービス拒否 (DDoS) 攻撃に対する保護を提供します。 AWS Firewall Manager は、新しいリソースが追加されても、アカウントとリソース全体で AWS WAF や Shield Advanced などの保護を管理します。

トピック

- とは AWS WAF
- とは AWS Shield Advanced
- とは AWS Firewall Manager

とは AWS WAF

AWS WAF は、保護されたウェブアプリケーションリソースに転送される HTTP および HTTPS リクエストをモニタリングできるウェブアプリケーションファイアウォールです。以下のリソースタイプを保護できます。

- Amazon CloudFront ディストリビューション
- Amazon API Gateway REST API
- Application Load Balancer
- AWS AppSync GraphQL API
- Amazon Cognito ユーザープール
- AWS App Runner サービス
- AWS Verified Access インスタンス

AWS WAF では、コンテンツへのアクセスを制御できます。リクエストの発生元の IP アドレスまたはクエリ文字列の値など、指定した条件に基づいて、保護されたリソースはリクエストに対し、リク

エストされたコンテンツ、HTTP 403 ステータスコード (禁止)、カスタム応答のリクエストのいずれかで応答します。

最も単純なレベルで AWS WAF は、次のいずれかの動作を選択します。

- 指定したリクエストを除くすべてのリクエストを許可する – これは、Amazon CloudFront、Amazon API Gateway 、Application Load Balancer 、 AWS AppSync、Amazon Cognito 、または AWS Verified Access でパブリックウェブサイトのコンテンツを配信したいが、攻撃者からのリクエストをブロックしたい場合に便利です。 AWS App Runner
- 指定したリクエスト以外のすべてのリクエストをブロックする - これは、制限されたウェブサイトのコンテンツを提供する場合に便利です。ユーザーは、ウェブサイトをウェブリクエストのプロパティ (ウェブサイトを参照するために使用する IP アドレスなど) によって簡単に識別できるようになります。
- 条件に一致するリクエストをカウントする - Count アクションを使用して、処理方法を変更せずにウェブトラフィックを追跡できます。これは、一般的なモニタリングのほか、新しいウェブリクエスト処理ルールをテストするためにも使用できます。ウェブリクエストの新しいプロパティに基づいてリクエストを許可またはブロックする場合は、まず、それらのプロパティに一致するリクエストをカウント AWS WAF するように を設定できます。これにより、ルールを切り替えて一致するリクエストを許可またはブロックする前に、新しい構成設定を確認できます。
- 基準に一致するリクエストに対して CAPTCHA または、チャレンジチェックを実行する – リクエストに対する CAPTCHA とサイレントチャレンジコントロールを実装して、保護されたリソースへのボットトラフィックを削減することができます。

の使用 AWS WAF にはいくつかの利点があります。

- 指定した基準を使用した、ウェブ攻撃に対する追加の保護。基準を定義するには、次のようなウェブリクエストの特性を使用します。
  - リクエストの発生元の IP アドレス。
  - リクエスト送信元の国。
  - リクエストヘッダーの値。
  - リクエストに含まれる文字列 (正規表現パターンと一致する特定の 1 つ以上の文字列)。
  - リクエストの長さ。
  - 悪意のある可能性がある SQL コード (SQL インジェクション) の有無。
  - 悪意のある可能性があるスクリプト (クロスサイトスクリプティング) の有無。

- 指定された基準を満たすウェブリクエストを許可、ブロック、またはカウントするルール。または、ルールは、指定された条件を満たすだけでなく、1 分または 5 分で指定されたリクエスト数を超えるウェブリクエストをブロックまたはカウントできます。
- 複数のウェブアプリケーションで再利用できるルール。
- AWS および AWS Marketplace 販売者のマネージドルールグループ。
- リアルタイムのメトリクスとサンプリングされたウェブリクエスト。
- AWS WAF API を使用した自動管理。

リソースに追加する保護をよりきめ細かに制御したい場合は、 AWS WAF のみを使用することが適切な選択である場合があります。の詳細については、 AWS WAF「」を参照してくださいAWS WAF。

とは AWS Shield Advanced

AWS WAF ウェブアクセスコントロールリスト (ウェブ ACLs) を使用すると、分散型サービス拒否 (DDoS) 攻撃の影響を最小限に抑えることができます。DDoS 攻撃に対する保護を強化するために、は AWS Shield Standard と AWS も提供します AWS Shield Advanced。 AWS Shield Standard は、すでに およびその他の AWS サービスに対して支払っている料金を超える追加コストなしで自動的に含まれ AWS WAF ます。

Shield Advanced は、Amazon EC2 インスタンス、Elastic Load Balancing ロードバランサー、CloudFront ディストリビューション、Route 53 ホストゾーン、および AWS Global Accelerator 標準アクセラレーターに対して拡張された DDoS 攻撃保護を提供します。Shield Advanced には追加料金が発生します。Shield Advanced のオプションと機能には、アプリケーションレイヤー DDoS 自動緩和、高度なイベント可視性、および Shield Response Team (SRT) からの専用サポートが含まれます。認知度の高いウェブサイトを所有している場合、または頻繁な DDoS 攻撃を受けやすいという場合は、Shield Advanced が提供する追加保護の購入を検討してください。詳細については、「AWS Shield Advanced 機能とオプション」および「AWS Shield Advanced をサブスクライブして追加の保護を適用するか否かの判断」を参照してください。

とは AWS Firewall Manager

AWS Firewall Manager は、、、 AWS WAF AWS Shield Advanced Amazon VPC セキュリティグループとネットワーク ACLs、Amazon Route 53 Resolver DNS Firewall など AWS Network Firewall、さまざまな保護のために複数のアカウントとリソースにわたる管理およびメンテナンスタ

スクを簡素化します。Firewall Manager を使用すると、保護を 1 回設定するだけで、アカウントとリソースに (追加する新しいアカウントとリソースにも) その保護が自動的に適用されます。

Firewall Manager の詳細については、「AWS Firewall Manager」を参照してください。

サービスを使用するためのアカウントのセットアップ

このトピックでは、、、 AWS WAF AWS Firewall Managerおよび を使用する準備をするための アカウントの作成などの準備手順について説明します AWS Shield Advanced。これらの暫定項目は請求されません。使用した AWS サービスに対してのみ課金されます。

トピック

にサインアップする AWS アカウント

がない場合は AWS アカウント、次の手順を実行して作成します。

にサインアップするには AWS アカウント

1. https://portal.aws.amazon.com/billing/signup を開きます。
2. オンラインの手順に従います。

   サインアップ手順の一環として、通話呼び出しを受け取り、電話キーパッドで検証コードを入力するように求められます。

   にサインアップすると AWS アカウント、 AWS アカウントのルートユーザーが作成されます。ルートユーザーには、アカウントのすべての AWS のサービス とリソースへのアクセス権があります。セキュリティのベストプラクティスとして、ユーザーに管理アクセスを割り当て、ルートユーザーのみを使用してルートユーザーアクセスが必要なタスクを実行してください。

AWS サインアッププロセスが完了すると、 から確認メールが送信されます。https://aws.amazon.com/ の [マイアカウント] を選んで、いつでもアカウントの現在のアクティビティを表示し、アカウントを管理できます。

管理アクセスを持つユーザーを作成する

にサインアップしたら AWS アカウント、 を保護し AWS アカウントのルートユーザー、 を有効にして AWS IAM Identity Center、日常的なタスクにルートユーザーを使用しないように管理ユーザーを作成します。

のセキュリティ保護 AWS アカウントのルートユーザー

1. ルートユーザーを選択し、 AWS アカウント E メールアドレスを入力して、アカウント所有者AWS Management Consoleとして にサインインします。次のページでパスワードを入力します。

   ルートユーザーを使用してサインインする方法については、AWS サインイン ユーザーガイドの「ルートユーザーとしてサインインする」を参照してください。
2. ルートユーザーの多要素認証 (MFA) を有効にします。

   手順については、「IAM ユーザーガイド」の AWS アカウント 「ルートユーザーの仮想 MFA デバイスを有効にする (コンソール)」を参照してください。

管理アクセスを持つユーザーを作成する

1. IAM アイデンティティセンターを有効にします。

   手順については、「AWS IAM Identity Center ユーザーガイド」の「AWS IAM Identity Centerの有効化」を参照してください。
2. IAM アイデンティティセンターで、ユーザーに管理アクセスを付与します。

   を ID ソース IAM アイデンティティセンターディレクトリ として使用する方法のチュートリアルについては、「 ユーザーガイド」の「デフォルト でユーザーアクセス IAM アイデンティティセンターディレクトリを設定するAWS IAM Identity Center 」を参照してください。

管理アクセス権を持つユーザーとしてサインインする

- IAM アイデンティティセンターのユーザーとしてサインインするには、IAM アイデンティティセンターのユーザーの作成時に E メールアドレスに送信されたサインイン URL を使用します。

  IAM Identity Center ユーザーを使用してサインインする方法については、「 AWS サインイン ユーザーガイド」の AWS 「 アクセスポータルにサインインする」を参照してください。

追加のユーザーにアクセス権を割り当てる

1. IAM アイデンティティセンターで、最小特権のアクセス許可を適用するというベストプラクティスに従ったアクセス許可セットを作成します。

   手順については、「AWS IAM Identity Center ユーザーガイド」の「権限設定を作成する」を参照してください。

2. グループにユーザーを割り当て、そのグループにシングルサインオンアクセス権を割り当てます。

   手順については、「AWS IAM Identity Center ユーザーガイド」の「グループの参加」を参照してください。

ツールをダウンロード

AWS Management Console には、 AWS WAF、 AWS Shield Advanced、および のコンソールが含まれていますが AWS Firewall Manager、プログラムでサービスにアクセスする場合は、以下を参照してください。

- API ガイドには、サービスがサポートする操作が記載されており、関連する SDK および CLI のドキュメントへのリンクも確認できます。
  - AWS WAF API リファレンス
  - AWS Shield Advanced API リファレンス
  - AWS Firewall Manager API リファレンス
- raw HTTP リクエストの組み立てなどの低レベルの詳細を処理せずに API を呼び出すには、 AWS SDK を使用できます。 AWS SDKs AWS サービスの機能をカプセル化する関数とデータ型を提供します。 AWS SDK をダウンロードしてインストール手順にアクセスするには、該当するページを参照してください。
  - Java
  - JavaScript
  - .NET
  - Node.js
  - PHP
  - Python
  - Ruby

AWS SDKs「Amazon Web Services のツール」を参照してください。

- AWS Command Line Interface（ AWS CLI) を使用して、コマンドラインから複数の AWS サービスを制御できます。スクリプトを使用してコマンドを自動化することもできます。詳細については、「AWS Command Line Interface」を参照してください。
- AWS Tools for Windows PowerShell は、これらの AWS サービスをサポートしています。詳細については、「AWS Tools for PowerShell Cmdlet Reference」(Cmdlet リファレンス) を参照してください。

AWS WAF

AWS WAF は、保護されている Web アプリケーションリソースに転送される HTTP (S) リクエストを監視できる Web アプリケーションファイアウォールです。以下のリソースタイプを保護できます。

- Amazon CloudFront ディストリビューション
- Amazon API Gateway REST API
- Application Load Balancer
- AWS AppSync GraphQL API
- Amazon Cognito ユーザープール
- AWS App Runner サービス
- AWS 検証済みアクセスインスタンス

AWS WAF コンテンツへのアクセスを制御できます。リクエストの発生元の IP アドレスまたはクエリ文字列の値など、指定した条件に基づいて、保護されたリソースに関連付けられたサービスはリクエストに対し、リクエストされたコンテンツ、HTTP 403 ステータスコード (禁止)、カスタム応答のいずれかで応答します。

> **Note**
>
> Amazon Elastic Container Service (Amazon ECS) AWS WAF コンテナでホストされているアプリケーションを保護するためにも使用できます。Amazon ECS は、クラスターで Docker コンテナを簡単に実行、停止、管理できる非常にスケーラブルで高速なコンテナ管理サービスです。このオプションを使用するには、サービス内のタスク全体で HTTP (S) レイヤー 7 トラフィックをルーティングおよび保護できるApplication Load Balancer を使用するように Amazon ECS を設定します。 AWS WAF 詳細については、「Amazon Elastic Container Service デベロッパーガイド」の「Service load balancing」を参照してください。

トピック

- AWS WAF ルールグループ
- AWS WAF 規則
- でのオーバーサイズリクエストコンポーネントの処理 AWS WAF
- での正規表現パターンマッチング AWS WAF
- の IP セットと正規表現パターンセット AWS WAF
- AWS WAFのカスタマイズされたウェブリクエストとレスポンス
- AWS WAF ウェブリクエストの ラベル
- AWS WAF インテリジェントな脅威軽減
- AWS WAF ウェブ ACL トラフィックのログ記録
- AWS WAF 保護機能のテストと調整
- Amazon AWS WAF CloudFront の機能との連携方法
- AWS WAF サービスの利用におけるセキュリティ
- AWS WAF クォータ
- AWS WAF クラシックリソースをに移行する AWS WAF

AWS WAF 仕組み

AWS WAF を使用して、保護対象リソースが HTTP (S) ウェブリクエストにどのように応答するかを制御します。これを行うには、ウェブアクセスコントロールリスト (ACL) を定義し、保護する 1 つ以上のウェブアプリケーションリソースと関連付けます。関連リソースは、 AWS WAF 受信したリクエストをウェブ ACL による検査に転送します。

ウェブ ACL では、リクエスト内で検索するトラフィックパターンを定義し、一致するリクエストに対して実行するアクションを指定するルールを作成します。アクションの選択肢は次のとおりです。

- 処理と応答のために、リクエストを保護されたリソースに送信することを許可する。
- リクエストをブロックする。
- リクエストをカウントする。
- リクエストに対して CAPTCHA またはチャレンジチェックを実行して、人間のユーザーと標準的なブラウザの使用を確認します。

AWS WAF コンポーネント

以下が主な構成要素です AWS WAF。

- ウェブ ACL — ウェブアクセスコントロールリスト (ACL) AWS を使用して一連のリソースを保護します。ウェブ ACL を作成し、ルールを追加してその保護戦略を定義します。ルールは、ウェブリクエストを検査する基準を定義し、条件に一致するリクエストに対して取る行動を指定します。また、ルールによってまだブロックまたは許可されていないすべてのリクエストをブロックするか、許可するかを示すウェブ ACL に対してデフォルトのアクションをセットします。ウェブ ACL の詳細については、「AWS WAF ウェブアクセスコントロールリスト (ウェブ ACLs)」を参照してください。

  ウェブ ACL AWS WAF はリソースです。
- ルール - 各ルールには、検査基準を定義するステートメントと、ウェブリクエストがその基準を満たす場合に実行するアクションが含まれます。ウェブリクエストが条件を満たしている場合、それは一致となります。CAPTCHA パズルまたはサイレントクライアントブラウザのチャレンジを使用する一致リクエストをブロック、許可、カウント、ボットコントロールを実行するルールを設定できます。ルールの詳細については、「AWS WAF 規則」を参照してください。

  AWS WAF ルールはリソースではありません。ルールはウェブ ACL またはルールグループのコンテキストでのみ定義されます。
- ルールグループ — ウェブ ACL 内で直接ルールを定義することも、再利用可能なルールグループ内でルールを定義することもできます。 AWS AWS Marketplace マネージドルールとセラーは、お客様が使用できるマネージドルールグループを提供しています。また、独自のルールグループを定義することもできます。ルールグループの詳細については、「AWS WAF ルールグループ」を参照してください。

  AWS WAF ルールグループはリソースです。

トピック

- AWS WAF ウェブ ACL キャパシティーユニット (WCUs)
- 保護できるリソース AWS WAF

AWS WAF ウェブ ACL キャパシティーユニット (WCUs)

AWS WAF は、ウェブ ACL キャパシティーユニット (WCU) を使用して、ルール、ルールグループ、およびウェブ ACLs の実行に必要な運用リソースを計算および制御します。 は、ルールグループとウェブ ACL を設定するときに AWS WAF WCU ACLs 制限を適用します。WCUs がウェブトラフィックを AWS WAF 検査する方法には影響しません。

AWS WAF は、ルール、ルールグループ、およびウェブ ACLs容量を管理します。

ルール WCU

AWS WAF は、ルールを作成または更新するときにルール容量を計算します。 は、各ルールの相対コストを反映するために、ルールタイプごとに異なる容量を AWS WAF 計算します。実行コストがほとんどない単純なルールでは、処理能力が大きい複雑なルールよりも使用される WCU が少なくなります。例えば、サイズ制約ルールステートメントでは、正規表現パターンセットを使用して検査するステートメントよりも使用する WCU が少なくなります。

ルール容量要件は通常、ルールタイプの基本コストに始まり、検査前にテキスト変換を追加する場合や JSON 本文を検査する場合など、複雑になるほど増えていきます。ルール容量要件については、「ルールステートメントの基本」にあるルールステートメントのリストを参照してください。

ルールグループ WCU

ルールグループの WCU 要件は、ルールグループ内で定義したルール数によって決まります。ルールグループの最大容量は 5,000 WCU です。

各ルールグループには、所有者が作成時に割り当てるイミュータブルな容量設定があります。これは、 で作成したマネージドルールグループとルールグループで当てはまります AWS WAF。ルールグループを変更する場合、それらの変更に伴うルールグループの WCU を容量内に収める必要があります。そうすることで、ルールグループを使用しているウェブ ACL が確実に容量要件内にとどまります。

ルールグループで使用されている WCUs は、ルールの WCUs の合計から、ルールの動作を組み合わせることで取得 AWS WAF できる処理の最適化を引いたものです。例えば、同じウェブリクエストコンポーネントを調べるために 2 つのルールを定義し、各ルールが検査する前にコンポーネントに特定の変換を適用する場合、変換の適用に対して 1 回だけ課金できる AWS WAF 可能性があります。ウェブ ACL のルールグループを使用するための WCU コストは、常にルールグループ作成時に定義した固定の WCU 設定です。

ルールグループを作成するときは、ルールグループの有効期間中に使用するルール数に対応できる十分な容量を設定するように注意してください。

ウェブ ACL WCU

ウェブ ACL の WCU 要件は、ウェブ ACL 内で使用するルールとルールグループの数によって決まります。

- ウェブ ACL のルールグループの使用コストは、ルールグループの容量設定に基づきます。
- ルールの使用コストは、ルールの計算された WCUs から、ウェブ ACL AWS WAF のルールの組み合わせから取得できる処理の最適化を引いたものです。例えば、同じウェブリクエストコンポーネ

ントを調べるために 2 つのルールを定義し、各ルールが検査する前にコンポーネントに特定の変換を適用する場合、変換の適用に対して 1 回だけ課金できる AWS WAF 可能性があります。

ウェブ ACL の基本料金には、最大 1,500 WCU の容量が含まれます。階層型料金モデルに従って、1,500 を超える WCUs を使用すると、追加料金が発生します。 は、ウェブ ACL WCU の使用状況の変化に応じて、ウェブ ACL の料金 AWS WAF を自動的に調整します。料金の詳細については、「AWS WAF の料金」を参照してください。

ウェブ ACL の最大容量は 5,000 WCU です。

ルールグループまたはウェブ ACL WCUs の確認

前のセクションで説明したように、ルールグループまたはウェブ ACL で使用される WCU の合計は、ルールグループまたはウェブ ACL で定義されているすべてのルールの WCU の合計と同じかそれ以下になります。

AWS WAF コンソールでは、ウェブ ACL またはルールグループにルールを追加するときに消費される容量を確認できます。ルールを追加するときに使用された現在のキャパシティユニットがコンソールに表示されます。

API を介して、ウェブ ACL またはルールグループで使用するルールの最大容量要件を確認できます。確認する場合は、チェックキャパシティコールでルールの JSON リストを指定します。詳細については、AWS WAF「 V2 API リファレンスCheckCapacity」の「」を参照してください。

保護できるリソース AWS WAF

AWS WAF ウェブ ACL を使用して、グローバルまたはリージョンのリソースタイプを保護できます。ウェブ ACL を保護するリソースに関連付けることにより、これを実行できます。ウェブ ACL とウェブ ACL AWS WAF が使用するリソースは、関連するリソースがあるリージョンに配置する必要があります。Amazon CloudFront ディストリビューションの場合、これは米国東部 (バージニア北部) に設定されます。

Amazon CloudFront ディストリビューション

AWS WAF コンソールまたは API を使用して、 AWS WAF ウェブ ACL CloudFront をディストリビューションに関連付けることができます。 CloudFront ディストリビューション自体を作成または更新するときに、ウェブ ACL をディストリビューションに関連付けることもできます。でアソシエーションを設定するには AWS CloudFormation、 CloudFront ディストリビューション設定を使

用する必要があります。Amazon について詳しくは CloudFront、Amazon CloudFront 開発者ガイドの「コンテンツへのアクセスの制御方法」を参照してください。 AWS WAF

AWS WAF CloudFront は世界中でディストリビューションで利用できますが、ウェブ ACL およびウェブ ACL で使用されるリソース (ルールグループ、IP セット、正規表現パターンセットなど) を作成するには、米国東部 (バージニア北部) リージョンを使用する必要があります。一部のインターフェースでは、「Global ()」のリージョンを選択できます。CloudFrontこれを選択することは、米国東部 (バージニア北部) リージョンまたは us-east-1 を選択することと同じです。

地域リソース

利用可能なすべてのリージョンのリージョナルリソースを保護できます。 AWS WAF 「Amazon Web Services 全般のリファレンス」の「AWS WAF エンドポイントとクォータ」でリストを確認できます。

を使用すると AWS WAF 、以下の種類の地域資源を保護できます。

- Amazon API Gateway REST API
- Application Load Balancer
- AWS AppSync GraphQL API
- Amazon Cognito ユーザープール
- AWS App Runner サービス
- AWS 検証済みアクセスインスタンス

ウェブ ACL を AWS リージョン内にある Application Load Balancer にのみ関連付けることができます。例えば、ウェブ ACL を AWS Outposts上にある Application Load Balancer に関連付けることはできません。

ウェブ ACL AWS WAF とそれが使用するその他のリソースは、保護対象リソースと同じリージョンにある必要があります。保護対象地域のリソースに対するウェブリクエストを監視および管理する場合、 AWS WAF すべてのデータを保護対象リソースと同じリージョンに保持します。

複数リソースの関連付けにおける制限

1 つのウェブ ACL を 1 AWS つ以上のリソースに関連付けることができますが、以下の制限があります。

- AWS 各リソースは 1 つのウェブ ACL にのみ関連付けることができます。ウェブ ACL AWS とリソースの関係はです one-to-many。

- ウェブ ACL は 1 CloudFront つ以上のディストリビューションに関連付けることができます。CloudFront ディストリビューションに関連付けたウェブ ACL AWS を他のリソースタイプに関連付けることはできません。

はじめるには AWS WAF

このチュートリアルでは、 AWS WAF を使用して次のタスクを実行する方法を説明します。

- セットアップ AWS WAF。
- AWS WAF コンソールのウィザードを使用して Web アクセスコントロールリスト (ウェブ ACL) を作成します。
- AWS AWS WAF ウェブリクエストを検査したいリソースを選択します。このチュートリアルでは、Amazon の手順について説明します CloudFront。プロセスは、Amazon API ゲートウェイ REST API、Application Load Balancer、 AWS AppSync GraphQL API、Amazon Cognito ユーザープール、 AWS App Runner サービス、 AWS または検証済みアクセスインスタンスと基本的に同じです。
- ウェブリクエストのフィルタリングに使用するルールおよびルールグループを追加します。例えば、リクエストの発生元の IP アドレスと、攻撃者によってのみ使用されるリクエスト内の値を指定できます。各ルールについて、一致するウェブリクエストの処理方法を指定します。ウェブリクエストをブロックしたりカウントしたり、CAPTCHA のようなボットチャレンジを実行することもできます。ウェブ ACL 内で定義する各ルールと、ルールグループ内で定義する各ルール用に、アクションを定義します。
- ウェブ ACL のデフォルトのアクション (Block または Allow) を指定します。これは、ウェブ ACL のルールがリクエストを明示的に許可またはブロックしていない場合に実行されるアクションです。 AWS WAF

> **Note**
>
> AWS 通常、このチュートリアルで作成したリソースに対して請求されるのは 1 日あたり 0.25 USD 未満です。チュートリアルを終了したら、不要な料金が発生しないようにリソースを削除することをお勧めします。

トピック

- ステップ 1: セットアップ AWS WAF

ステップ 1: セットアップ AWS WAF

サービスを使用するためのアカウントのセットアップ の一般的なセットアップ手順をまだ実行していない場合、今すぐ実行してください。

ステップ 2: ウェブ ACL を作成する

AWS WAF コンソールは、リクエストの発信元の IP アドレスやリクエストの値など、 AWS WAF 指定した条件に基づいてウェブリクエストをブロックまたは許可するように設定するプロセスを順を追って説明します。このステップでは、ウェブ ACL を作成します。 AWS WAF ウェブ ACL の詳細については、を参照してくださいAWS WAF ウェブアクセスコントロールリスト (ウェブ ACLs)。

ウェブ ACL を作成するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。
2. AWS WAF ホームページから [ウェブ ACL の作成] を選択します。
3. [Name] (名前) で、このウェブ ACL の識別に使用する名前を入力します。

   > **Note**
   >
   > ウェブ ACL の作成後は、名前を変更することはできません。

4. (オプション) 必要に応じて、[Description - optional] (説明 - オプション) に、ウェブ ACL の詳しい説明を入力します。
5. CloudWatch メトリクス名については、該当する場合はデフォルト名を変更します。有効な文字については、コンソールのガイダンスに従ってください。名前には、特殊文字、空白や、「All」および「Default_Action」などの AWS WAF用に予約されたメトリクス名を使用できません。

> **Note**
>
> ウェブ ACL CloudWatch を作成した後でメトリックス名を変更することはできません。

6. [リソースタイプ] には、[CloudFrontディストリビューション] を選択します。ディストリビューションの場合 CloudFront 、リージョンは自動的に Global (CloudFront) に設定されます。
7. (オプション) [AWS 関連リソース]-[オプション] では、[リソースの追加 AWS] を選択します。ダイアログボックスで、関連付けるリソースを選択し、[Add] (追加) を選択します。 AWS WAF は [Describe web ACL and associated AWS resources] (ウェブ ACL と関連付けられた リソースの説明) ページに戻します。
8. [Next] (次へ) を選択します。

ステップ 3: 文字列一致ルールを追加する

このステップでは、文字列一致ステートメントを使用してルールを作成し、一致リクエストの処理方法を指定します。文字列一致ルールステートメントは、 AWS WAF がリクエストで検索する文字列を識別します。通常、文字列は印刷可能な ASCII 文字で構成されますが、16 進数 0x00 ～ 0xFF (10 進数 0 ～ 255) の任意の文字を指定できます。検索する文字列を指定するだけでなく、ヘッダー、クエリ文字列、リクエストボディなど、検索するウェブリクエストコンポーネントを指定します。

このステートメントタイプは、ウェブリクエストコンポーネントで動作し、次のリクエストコンポーネント設定が必要です。

- [リクエストコンポーネント] — ウェブリクエストの検査対象部分 (クエリ文字列や本文など)。

> **Warning**
>
> リクエストコンポーネントのボディ、JSON ボディ、ヘッダー、または Cookie を調べる場合は、 AWS WAF 検査できるコンテンツの量の制限についてお読みください。でのオーバーサイズリクエストコンポーネントの処理 AWS WAF

  ウェブリクエストコンポーネントの詳細については、「ウェブリクエストコンポーネントの仕様と処理」を参照してください。
- オプションのテキスト変換 — AWS WAF リクエストコンポーネントを検査する前に実行したい変換。例えば、小文字に変換したり、空白を正規化したりできます。複数の変換を指定すると、

AWS WAF 記載されている順序で処理されます。詳細については、テキスト変換オプション を参照してください。

AWS WAF ルールに関する追加情報については、を参照してくださいAWS WAF 規則。

文字列一致ルールステートメントを作成するには

1. [Add rules and rule groups] (ルールとルールグループの追加) ページで、[Add rules] (ルールの追加)、[Add my own rules and rule groups] (独自のルールとルールグループの追加)、[Rule builder] (ルールビルダー)、[Rule visual editor] (ルールビジュアルエディタ) の順に選択します。

   > **Note**
   >
   > コンソールには、ルールビジュアルエディタとルール JSON エディタが用意されています。JSON エディタを使用すると、ウェブ ACL 間で設定を簡単にコピーできます。これは、ネストのレベルが複数あるルールセットなど、より複雑なルールセットに必要です。
   > この手順では、ルールビジュアルエディタを使用します。

2. [Name] (名前) で、このルールの識別に使用する名前を入力します。
3. [Type] (タイプ) で、[Regular rule] (通常のルール) を選択します。
4. [If a request] (リクエストの状態) で、[matches the statement] (ステートメントに一致) を選択します。

   その他のオプションは、論理ルールステートメントタイプ用です。これらを使用して、他のルールステートメントの結果を組み合わせたり、否定したりできます。
5. 「Statement」 の「Inspect」で、ドロップダウンを開き、検査するウェブリクエストコンポーネントを選択します。 AWS WAF この例では、[Header] (ヘッダー) を選択します。

   [Header] (ヘッダー) を選択した場合は、 AWS WAF で検査するヘッダーも指定します。**`User-Agent`** と入力します。この値では大文字と小文字は区別されません。
6. [Match type] (一致タイプ) で、指定した文字列が `User-Agent` ヘッダーに表示される場所を選択します。

   この例では、[Exactly matches string] (文字列に完全一致) を選択します。これは、各ウェブリクエストのユーザーエージェントヘッダーに、 AWS WAF 指定した文字列と同じ文字列がないかを検査することを示しています。

7. [String to match] (照合する文字列) で、 AWS WAF で検索する文字列を指定します。[String to match] (照合する文字列) は最大 200 文字です。base64 でエンコードされた値を指定する場合、エンコード前の長さで最大 200 文字指定できます。

   この例では、と入力します。MyAgent AWS WAF `User-Agent`MyAgentウェブリクエストのヘッダーに値があるかどうかを調べます。

8. [Text transformation] (テキスト変換) を [None] (なし) のままにします。
9. [Action] (アクション) で、ウェブリクエストに一致したときにルールによって実行されるアクションを選択します。この例では、[Count] (カウント) を選択し、他の選択肢はそのままにしておきます。カウントアクションにより、ルールに一致するウェブリクエストのメトリクスが作成されますが、リクエストが許可またはブロックされるかどうかには影響しません。アクションの選択の詳細については、「ルールアクション」および「ウェブ ACL ルールおよびルールグループの評価」を参照してください。
10. [Add rule] (ルールの追加) を選択します。

ステップ 4: AWS マネージドルールルールグループを追加する

AWS Managed Rulesでは、お客様が使用できる一連のマネージドルールグループが提供され、AWS WAF そのほとんどはお客様に無料で提供されます。ルールグループの詳細については、「AWS WAF ルールグループ」を参照してください。 AWS マネージドルールルールグループをこのウェブ ACL に追加します。

AWS マネージドルールルールグループを追加するには

1. [Add rules and rule groups] (ルールとルールグループの追加) ページで、[Add rules] (ルールの追加) を選択し、[Add managed rule groups] (マネージドルールグループの追加) を選択します。
2. [マネージドルールグループを追加] ページで、[AWS マネージドルールグループ] のリストを展開します。(AWS Marketplace 出品者向けの出品情報も表示されます。 それらの商品を購読すると、 AWS マネージドルールルールグループと同じ方法で使用できます。)
3. 追加するルールグループについて、次を実行します。

   a. [Action] (アクション) 列で、[Add to web ACL] (ウェブ ACL に追加) 切り替えボタンをオンにします。

   b. [Edit] (編集) を選択し、ルールグループの [Rules] (ルール) リストで [Override all rule actions] (すべてのルールアクションをオーバーライド) ドロップダウンを開いて [Count] を選択します。これにより、ルールグループ内のすべてのルールのアクションがカウントのみ

に設定されます。これにより、ルールグループのルールを使用する前に、ルールグループのすべてのルールがウェブリクエストでどのように動作するかを確認できます。

c. [Save rule] (ルールを保存) を選択します。

4. [Add managed rule groups] (マネージドルールグループを追加) ページで、[Add rules] (ルールを追加) を選択します。これにより、[Add rules and rule groups] (ルールとルールグループを追加) ページに戻ります。

ステップ 5: ウェブ ACL の設定を完了する

ルールとルールグループをウェブ ACL 設定に追加したら、ウェブ ACL 内のルールの優先順位を管理し、メトリクス、タグ付け、ログ記録などの設定を行うことで完了します。

ウェブ ACL の設定を完了するには

1. [Add rules and rule groups] (ルールとルールグループの追加) ページで、[Next] (次へ) を選択します。
2. 「ルール優先度の設定」ページでは、ウェブ ACL 内のルールとルールグループの処理順序を確認できます。 AWS WAF リストの一番上から処理します。処理順序は、ルールを上下に移動することで変更できます。これを行うには、リストで 1 つを選択し、[Move up] (上へ移動) または [Move down] (下へ移動) を選択します。ルーティングの優先度の詳細については、「ウェブ ACL でのルールおよびルールグループの処理順序」をご覧ください。
3. [次へ] を選択します。
4. Amazon CloudWatchメトリクスのConfigure metricsページでは、ルールとルールグループの計画メトリックスと、ウェブリクエストのサンプリングオプションを確認できます。サンプリングされたリクエストの表示方法については、「ウェブリクエストのサンプルの表示」を参照してください。Amazon CloudWatch メトリックスについて詳しくは、を参照してくださいAmazon によるモニタリング CloudWatch。

   ウェブトラフィックメトリックスの概要には、 AWS WAF コンソールのウェブ ACL ページの「トラフィック概要」タブでアクセスできます。コンソールダッシュボードには、ウェブ ACL の Amazon CloudWatch メトリックスの概要がほぼリアルタイムで表示されます。詳細については、「ウェブ ACL トラフィック概要ダッシュボード」を参照してください。
5. [次へ] を選択します。
6. [Review and create web ACL] (ウェブ ACL の確認と作成) ページで、設定を確認し、[Create web ACL] (ウェブ ACL の作成) を選択します。

ウィザードによって [Web ACL] (ウェブ ACL) ページに戻ります。このページには、新しいウェブ ACL が一覧表示されます。

ステップ 6: リソースをクリーンアップする

これでチュートリアルは完了です。 AWS WAF アカウントに追加料金が発生しないようにするには、 AWS WAF 作成したオブジェクトをクリーンアップしてください。または、を使用して本当に管理したいウェブリクエストに合わせて設定を変更することもできます。 AWS WAF

> **Note**
>
> AWS 通常、このチュートリアルで作成したリソースについて、1 日あたり 0.25 USD 未満で請求されます。終了したら、不要な料金が発生しないようにリソースを削除することをお勧めします。

AWS WAF 料金が発生するオブジェクトを削除するには

1. [Web ACL] (ウェブ ACL) ページで、リストからウェブ ACL を選択し、[Edit] (編集) を選択します。
2. 「 AWS 関連リソース」タブでは、関連する各リソースについて、リソース名の横にあるラジオボタンを選択し、「関連付け解除」を選択します。これにより、ウェブ ACL とリソースの関連付けが解除されます。 AWS
3. 次の各画面で、[Web ACL] (ウェブ ACL) ページに戻るまで [Next] (次へ) を選択します。

   [Web ACL] (ウェブ ACL) ページで、リストからウェブ ACL を選択し、[Delete] (削除) を選択します。

ルールおよびルールステートメントは、ルールグループおよびウェブ ACL 定義の外部には存在しません。ウェブ ACL を削除すると、ウェブ ACL で定義した個々のルールがすべて削除されます。ウェブ ACL からルールグループを削除する場合は、そのグループへの参照を削除するだけです。

AWS WAF ウェブアクセスコントロールリスト (ウェブ ACLs)

ウェブアクセスコントロールリスト (ウェブ ACL) を使用すると、保護されたリソースが応答するすべての HTTP(S) ウェブリクエストをきめ細かく制御できます。Amazon CloudFront、Amazon API GatewayApplication Load Balancer AWS AppSync、Amazon Cognito AWS App Runner、および AWS Verified Access リソースを保護できます。

次のような基準を使用すると、リクエストを許可またはブロックできます。

- リクエストの IP アドレスの送信元
- リクエストの送信元の国
- リクエストの一部に含まれる文字列一致または正規表現 (regex) 一致
- リクエストの特定の部分のサイズ
- 悪意のある SQL コードまたはスクリプトの検出

これらの条件の任意の組み合わせをテストすることもできます。指定された条件を満たすだけでなく、1 分間に指定されたリクエスト数を超えるウェブリクエストをブロックまたはカウントできます。論理演算子を使用して条件を組み合わせることができます。リクエストに対して CAPTCHA パズルやサイレントクライアントセッションのチャレンジを実行することもできます。

一致基準と、 AWS WAF ルールステートメントで一致に対して実行するアクションを指定します。ルールステートメントは、ウェブ ACL 内、およびウェブ ACL で使用する再利用可能なルールグループで直接定義できます。オプションの詳細なリストについては、「ルールステートメントの基本」および「ルールアクション」を参照してください。

ウェブリクエストの検査および処理基準を指定するには、次のタスクを実行します。

1. 指定したルールのいずれにも一致しないウェブリクエストのウェブ ACL デフォルトアクション (Allow または Block) を選択します。詳細については、「ウェブ ACL のデフォルトアクション」を参照してください。
2. ウェブ ACL で使用するルールグループを追加します。マネージドルールグループには通常、ウェブリクエストをブロックするルールが含まれます。ルールグループについては、「AWS WAF ルールグループ」を参照してください。
3. 1 つ以上のルールで、追加の一致基準と処理手順を指定します。複数のルールを追加するには、AND または OR ルールステートメントをまず使用し、結合するルールをそれらの下にネストします。ルールオプションを否定する場合は、NOT ステートメントでルールをネストします。必要に応じて、通常のルールの代わりにレートベースのルールを使用して、条件を満たす単一の IP アドレスからのリクエストの数を制限できます。ルールについては、「AWS WAF 規則」を参照してください。

ウェブ ACL に複数のルールを追加すると、 はウェブ ACL にリストされている順序でルール AWS WAF を評価します。詳細については、「ウェブ ACL ルールおよびルールグループの評価」を参照してください。

ウェブ ACL を作成するときに、その ACL を使用するリソースのタイプを指定します。詳細については、「ウェブ ACL の作成」を参照してください。ウェブ ACL を定義した後、その ACL をリソースに関連付けて、リソースの保護を開始できます。詳細については、「ウェブ ACL とリソースの関連付けまたは関連付け解除 AWS」を参照してください。

AWS リソースが からの応答遅延を処理する方法 AWS WAF

場合によっては、リクエストを許可またはブロックするかどうかについて、関連する AWS リソースへの応答を遅らせる内部エラーが発生する AWS WAF ことがあります。このような場合、CloudFront は通常、リクエストを許可またはコンテンツを提供しますが、リージョンサービスは通常、リクエストを拒否し、コンテンツを提供しません。

トピック

- ウェブ ACL ルールおよびルールグループの評価
- ウェブ ACL のデフォルトアクション
- 本文検査のサイズ制限の管理
- CAPTCHA、チャレンジ、トークンの設定
- ウェブ ACL の使用

ウェブ ACL ルールおよびルールグループの評価

ウェブ ACL がウェブリクエストを処理する方法は、次に応じて異なります。

- ウェブ ACL およびルールグループ内のルールの優先順位の数値設定
- ルールおよびウェブ ACL のアクション設定
- 追加したルールグループ内のルールに設定した上書き

ルールアクション設定のリストについては、「ルールアクション」を参照してください。

ルールアクション設定とデフォルトのウェブ ACL アクション設定で、リクエストと応答の処理をカスタマイズできます。詳細については、「AWS WAFのカスタマイズされたウェブリクエストとレスポンス」を参照してください。

トピック

- ウェブ ACL でのルールおよびルールグループの処理順序
- がウェブ ACL でルールとルールグループのアクション AWS WAF を処理する方法

- ルールグループのアクションオーバーライドオプション

ウェブ ACL でのルールおよびルールグループの処理順序

ウェブ ACL および任意のルールグループ内では、優先順位の数値設定を使用してルールの評価順序を決定します。ウェブ ACL 内の各ルールには、そのウェブ ACL 内で一意の優先順位を設定する必要があります。また、ルールグループ内の各ルールには、そのルールグループ内で一意の優先順位を設定する必要があります。

> **Note**
>
> コンソールからルールグループとウェブ ACL を管理する場合、 AWS WAF リスト内のルールの順序に基づいて固有の数値優先順位設定が自動的に割り当てられます。 AWS WAF リストの一番上にあるルールには最小の数値優先度を割り当て、一番下のルールには数値的に最も高い優先度を割り当てます。

ウェブリクエストと照合してウェブ ACL AWS WAF またはルールグループを評価する場合、優先順位が最も低いものから順に、一致するものが見つかって評価を終了するか、すべてのルールを使い果たすまで、ルールを評価します。

例えば、ウェブ ACL に次のルールとルールグループがあり、次のように優先順位付けされているとします。

- Rule1 — 優先度 0
- RuleGroupA — 優先度 100
  - RuleA1 – 優先度 10,000
  - RuleA2 – 優先度 20,000
- Rule2 — 優先度 200
- RuleGroupB — プライオリティ 300
  - RuleB1 – 優先度 0
  - RuleB2 – 優先度 1

AWS WAF このウェブ ACL のルールを次の順序で評価します。

- Rule1

- RuleGroupルール A1
- RuleGroupルール A2
- Rule2
- RuleGroupB ルール B1
- RuleGroupB ルール B2

がウェブ ACL でルールとルールグループのアクション AWS WAF を処理する方法

ルールとルールグループを設定するときは、一致するウェブリクエスト AWS WAF の処理方法を選択します。

- Allow および Block は終了アクションです – Allow および Block アクションは、一致するウェブリクエストにおけるウェブ ACL のその他の処理をすべて停止されます。ウェブ ACL のルールがリクエストの一致を検出し、そのルールアクションが Allowまたは の場合Block、その一致によってウェブ ACL のウェブリクエストの最終処理が決まります。一致するルールの後に来るウェブ ACL 内の他のルールは処理 AWS WAF されません。これに該当するのは、ウェブ ACL に直接追加するルールや、追加されたルールグループに属するルールです。Block アクションでは、保護されたリソースはウェブリクエストを受信または処理しません。
- Count は非終了アクションです – Count アクションのあるルールがリクエストと一致すると、AWS WAF はリクエストをカウントし、その後にウェブ ACL ルールセットに従うルールの処理を続行します。
- CAPTCHA および は、非終了アクションまたは終了アクションChallengeにすることができます – これらのアクションのいずれかを持つルールがリクエストに一致すると、 はトークンのステータス AWS WAF をチェックします。リクエストに有効なトークンがある場合、 は一致をCount一致と同様に AWS WAF 処理し、ウェブ ACL ルールセットに続くルールの処理を続行します。リクエストに有効なトークンがない場合、 は評価 AWS WAF を終了し、解決する CAPTCHA パズルまたはサイレントバックグラウンドクライアントセッションチャレンジをクライアントに送信します。

ルール評価によって終了アクションが実行されない場合、 はウェブ ACL のデフォルトアクションをリクエスト AWS WAF に適用します。詳細については、「ウェブ ACL のデフォルトアクション」を参照してください。

ウェブ ACL では、ルールグループ内のルールのアクション設定を上書きしたり、ルールグループによって返されるアクションを上書きしたりできます。詳細については、「ルールグループのアクションオーバーライドオプション」を参照してください。

アクションと優先度設定の相互作用

ウェブリクエスト AWS WAF に適用されるアクションは、ウェブ ACL のルールの優先順位の数値設定の影響を受けます。たとえば、ウェブ ACL に Allow アクションと 50 の優先順位の数値を持つルール、ならび Count アクションと 100 の優先順位の数値を持つ別のルールがあるとします。 AWS WAF は優先順位に応じて最小のものからウェブ ACL 内のルールが評価するため、許可ルールをカウントルールより先に評価します。両方のルールに一致するウェブリクエストは、最初に許可ルールに一致します。Allow は終了アクションであるため、 はこの一致で評価 AWS WAF を停止し、カウントルールに対してリクエストを評価しません。

- 許可ルールに一致しないリクエストのみをカウントルールメトリクスに含める場合は、ルールの優先度設定が便利です。
- 一方、許可ルールに一致するリクエストに対してもカウントルールのカウントメトリクスを取得する場合、カウントルールには許可ルールより小さい優先順位の数値を設定し、先に実行されるようにする必要があります。

優先順位の設定の詳細については、「ウェブ ACL でのルールおよびルールグループの処理順序」を参照してください。

ルールグループのアクションオーバーライドオプション

ルールグループをウェブ ACL に追加するとき、一致するウェブリクエストに対して実行されるアクションをオーバーライドできます。ウェブ ACL 設定内のルールグループのアクションをオーバーライドしても、ルールグループ自体は変更されません。ウェブ ACL のコンテキストで がルールグループ AWS WAF を使用する方法のみを変更します。

ルールグループのルールアクションの上書き

ルールグループ内のルールのアクションは、任意の有効なルールアクションにオーバーライドきできます。これを実行すると、一致するリクエストは、設定されたルールのアクションがオーバーライド設定である場合とまったく同様に処理されます。

> **Note**
>
> ルールアクションは、終了アクションまたは非終了アクションである場合があります。終了アクションは、リクエストのウェブ ACL 評価を停止し、保護されたアプリケーションへのリクエストの継続を許可またはブロックします。

ルールアクションのオプションは以下のとおりです。

- Allow – AWS WAF リクエストを保護された AWS リソースに転送して処理と応答を許可します。これは終了アクションです。定義したルールでは、リクエストを保護されたリソースに転送する前に、カスタムヘッダーを挿入できます。
- Block – リクエストを AWS WAF ブロックします。これは終了アクションです。デフォルトでは、保護された AWS リソースは HTTP `403 (Forbidden)`ステータスコードで応答します。定義したルールでは、応答をカスタマイズできます。がリクエストを AWS WAF ブロックすると、Blockアクション設定によって、保護されたリソースがクライアントに送り返すレスポンスが決まります。
- Count – リクエストを AWS WAF カウントしますが、許可するかブロックするかは決定しません。これは非終了アクションです。 AWS WAF がウェブ ACL の残りのルールの処理を継続します。定義したルールでは、リクエストにカスタムヘッダーを挿入し、他のルールで一致するラベルを追加できます。
- CAPTCHA および Challenge - CAPTCHA パズルとサイレントチャレンジ AWS WAF を使用して、リクエストがボットから送信されていないことを確認し、トークン AWS WAF を使用して最近成功したクライアントレスポンスを追跡します。

  CAPTCHA パズルとサイレントチャレンジは、ブラウザが HTTPS エンドポイントにアクセスしている場合にのみ実行できます。トークンを取得するには、ブラウザクライアントが安全なコンテキストで実行されている必要があります。

  > **Note**
  >
  > CAPTCHA または Challenge ルールアクションを 1 つのルールで使用、あるいはルールグループでルールアクションのオーバーライドとして使用すると、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

  これらのルールアクションは、リクエスト内のトークンの状態に応じて、終了アクションまたは非終了アクションである場合があります。

  - 有効で有効期限が切れていないトークンの非終了 – トークンが有効で、設定された CAPTCHA またはチャレンジイミュニティ時間に従って有効期限が切れていない場合、 は Count action. AWS WAF continues のようなリクエスト AWS WAF を処理し、ウェブ ACL の残りのルールに基づいてウェブリクエストを検査します。Count 設定と同様に、定義したルールでは、リクエストに挿入するカスタムヘッダーを使用してこれらのアクションを設定したり (オプション)、他のルールが照合できるラベルを追加したりできます。

- 無効または期限切れのトークンのリクエストがブロックされた状態で終了する – トークンが無効であるか、指定されたタイムスタンプの有効期限が切れている場合、 は Blockアクションと同様にウェブリクエストの検査 AWS WAF を終了し、リクエストをブロックします。 AWS WAF その後、 はカスタムレスポンスコードでクライアントに応答します。の場合CAPTCHA、リクエストの内容がクライアントブラウザが処理できることを示している場合、 AWS WAF は人間のクライアントをボットと区別するように設計された JavaScript インタースティシャルで CAPTCHA パズルを送信します。Challenge アクションでは、 は、通常のブラウザをボットによって実行されているセッションと区別するように設計されたサイレントチャレンジで JavaScript インタースティシャル AWS WAF を送信します。

詳細については、「CAPTCHAChallengeの および AWS WAF」を参照してください。

このオプションの使用方法については、「ルールグループ内のルールアクションのオーバーライド」を参照してください。

ルールアクションを Count にオーバーライド

ルールアクションオーバーライドの最も一般的な使用例は、ルールアクションの一部またはすべてを Count にオーバーライドして、ルールグループの動作を本番稼働に移行する前にテストおよびモニタリングすることです。

これを使用して誤検出を生成しているルールグループをトラブルシューティングすることもできます。誤検出は、ブロックすると想定していないトラフィックをルールグループがブロックするときに発生します。ルールグループ内で、許可したいリクエストをブロックするルールを特定した場合、そのルールに対するこのカウントアクションのオーバーライドを保持し、リクエストに対するアクションを除外できます。

テストでルールアクションのオーバーライドを使用する詳細については、「AWS WAF 保護機能のテストと調整」を参照してください。

JSON リスト: **`RuleActionOverrides`** を **`ExcludedRules`** に置き換えます

2022 年 10 月 27 日より前にウェブ ACL 設定Countでルールグループのルールアクションを に設定した場合、 はウェブ ACL JSON のオーバーライドを として AWS WAF 保存しましたExcludedRules。これで、ルールを Count にオーバーライドする JSON 設定が RuleActionOverrides 設定に追加されました。

AWS WAF コンソールを使用して既存のルールグループ設定を編集すると、コンソールは JSON のすべてのRuleActionOverrides設定を ExcludedRules 設定に自動的に変換し、オーバーライドアクションを に設定しますCount。

- 現在の設定例:

```
       "ManagedRuleGroupStatement": {
          "VendorName": "AWS",
          "Name": "AWSManagedRulesAdminProtectionRuleSet",
          "RuleActionOverrides": [
            {
              "Name": "AdminProtection_URIPATH",
              "ActionToUse": {
                "Count": {}
              }
            }
          ]
```

- 古い設定例:

```
OLD SETTING
       "ManagedRuleGroupStatement": {
          "VendorName": "AWS",
          "Name": "AWSManagedRulesAdminProtectionRuleSet",
          "ExcludedRules": [
            {
              "Name": "AdminProtection_URIPATH"
            }
          ]
OLD SETTING
```

JSON リストですべての `ExcludedRules` 設定は、アクションを Count に設定した `RuleActionOverrides` 設定に更新することをお勧めします。API はどちらの設定も受け付けますが、新しい `RuleActionOverrides` 設定のみを使用した場合、コンソール作業と API 作業の間で JSON リストの一貫性が保たれます。

ルールグループがアクションの上書きを に返す Count

ルールグループが返すアクションをオーバーライドして、Count に設定できます。

> **Note**
>
> これは、 がルールグループ自体 AWS WAF を評価する方法を変更しないため、ルールグループのルールをテストするには適していません。これは、 がルールグループ評価からウェブ ACL に返された結果 AWS WAF を処理する方法にのみ影響します。ルールグループ内の

ルールをテストする場合は、前述のセクションで説明したオプション ルールグループのルールアクションの上書き を使用します。

ルールグループアクションを にオーバーライドするとCount、 はルールグループの評価を正常に AWS WAF 処理します。

ルールグループ内のルールが一致しない、あるいはすべての一致するルールに Count アクションがある場合、このオーバーライドはルールグループまたはウェブ ACL の処理に影響を与えません。

ウェブリクエストに一致し、終了ルールアクションを持つルールグループ内の最初のルールは、AWS WAF がルールグループの評価を停止し、終了アクションの結果をウェブ ACL 評価レベルに返します。この時点で、ウェブ ACL 評価では、このオーバーライドが effect. AWS WAF overrides the terminationing action so that the result of the rule group evaluation is only a Count action. AWS WAF は、ウェブ ACL 内の残りのルールの処理を続行します。

このオプションの使用方法については、「ルールグループの評価結果を Count にオーバーライド」を参照してください。

ウェブ ACL のデフォルトアクション

ウェブ ACL を作成および設定するときに、ウェブ ACL のデフォルトアクションを設定する必要があります。 AWS WAF は、終了アクションが適用されることなく、ウェブ ACL のルール評価をすべて通過したウェブリクエストすべてに、このアクションを適用します。終了アクションは、リクエストのウェブ ACL 評価を停止し、保護されたアプリケーションへのリクエストの継続を許可またはブロックします。ルールアクションについては、「ルールアクション」を参照してください。

ウェブ ACL のデフォルトアクションがウェブリクエストの最終的な処理を決定する必要があります。したがって、これは終了アクションです。

- Allow – ほとんどのユーザーに対してウェブサイトへのアクセスを許可する一方、指定した IP アドレスからのリクエストまたは悪意のある SQL コードや指定した値が含まれている可能性があるリクエストを行う攻撃者に対してアクセスを拒否する場合、デフォルトアクションとして Allow を選択します。その後、ブロックする特定のリクエストを識別してブロックするルールを、ウェブ ACL に追加します。このアクションを使用すると、保護されたリソースに転送する前に、カスタムヘッダーをリクエストに挿入できます。
- Block – ほとんどのユーザーに対してはウェブサイトへのアクセスを拒否する一方、指定した IP アドレスからのリクエストまたは指定した値が含まれているリクエストを行うユーザーに対してアク

セスを許可する場合、デフォルトアクションとして Block を選択します。その後、許可する特定のリクエストを識別して許可するルールを、ウェブ ACL に追加します。デフォルトでは、Blockアクションの場合、 AWS リソースは HTTP 403 (Forbidden) ステータスコードで応答しますが、応答はカスタマイズできます。

リクエストとレスポンスをカスタマイズする方法については、「AWS WAFのカスタマイズされたウェブリクエストとレスポンス」を参照してください。

独自のルールとルールグループの設定は、ほとんどのウェブリクエストを許可するかブロックするかに応じて、一部が異なります。たとえば、ほとんどのリクエストを許可する場合、ウェブ ACL のデフォルトアクションを Allow に設定し、その後にブロックするウェブリクエストを識別するルールを追加します。これには次のようなリクエストが該当します。

- リクエスト数が不当に多い IP アドレスからのリクエスト
- お客様がビジネスを行っていない国、または頻繁に攻撃元になっている国からのリクエスト
- User-agent ヘッダーに不正な値が含まれているリクエスト
- 悪意のある SQL コードが含まれている可能性があるリクエスト

マネージドルールグループのルールは通常、Block アクションを使用しますが、すべての場合に限りません。例えば、Bot Control に使用される一部のルールでは、CAPTCHA および Challenge アクション設定を使用します。マネージドルールグループの詳細については、「マネージドルールグループ」を参照してください。

本文検査のサイズ制限の管理

本文検査サイズの制限は、 が検査 AWS WAF できるリクエスト本文の最大サイズです。ウェブリクエスト本文が制限を超えると、基盤となるホストサービスは、制限内のコンテンツを検査 AWS WAF のために に転送するだけです。

- Application Load Balancer と の場合 AWS AppSync、制限は 8 KB (8,192 バイト) に固定されます。
- CloudFront、API Gateway、Amazon Cognito 、App Runner、Verified Access の場合、デフォルトの制限は 16 KB (16,384 バイト) で、どのリソースタイプの制限も 16 KB 単位で最大 64 KB まで増やすことができます。設定オプションは 16 KB、32 KB、48 KB、および 64 KB です。

オーバーサイズ本文の処理

ウェブトラフィックに制限を超える本文が含まれている場合、設定されたオーバーサイズ処理が適用されます。オーバーサイズ処理のオプションについては、「」を参照してくださいでのオーバーサイズリクエストコンポーネントの処理 AWS WAF。

制限設定を増やす際の料金に関する考慮事項

AWS WAF は、リソースタイプのデフォルト制限内のトラフィックを検査するための基本レートを課金します。

CloudFront、API Gateway、Amazon CognitoApp Runner、Verified Access リソースの場合、制限設定を増やすと、検査 AWS WAF できるトラフィックには新しい制限までの本文サイズが含まれます。本文サイズがデフォルトの 16 KB を超えるリクエストの検査に対してのみ、追加料金がかかります。料金の詳細については、「AWS WAF 料金」を参照してください。

本文検査のサイズ制限を変更するためのオプション

、API Gateway CloudFront、Amazon Cognito、App Runner、または Verified Access リソースの本文検査サイズ制限を設定できます。

ウェブ ACL を作成または編集するときに、リソースの関連付け設定で本文検査サイズの制限を変更できます。API については、 のウェブ ACL の関連付け設定を参照してくださいAssociationConfig。コンソールについては、ウェブ ACL の関連リソースを指定するページの設定を参照してください。コンソール設定に関するガイダンスについては、「ウェブ ACL の使用」を参照してください。

CAPTCHA、チャレンジ、トークンの設定

ウェブ ACL で、 CAPTCHAまたは ルールアクションを使用するChallengeルールと、 AWS WAF マネージド保護のサイレントクライアントチャレンジを管理するアプリケーション統合 SDKs のオプションを設定できます。

これらの機能は、エンドユーザーに CAPTCHA パズルで挑戦させて、クライアントセッションにサイレントチャレンジを提供することにより、ボットの活動を軽減します。クライアントが応答に成功すると、 AWS WAF はクライアントがウェブリクエストで使用するトークンを提供します。このトークンは最後に成功したパズルおよびチャレンジレス応答のタイムスタンプが付いています。詳細については、「AWS WAF インテリジェントな脅威軽減」を参照してください。

ウェブ ACL 設定では、 がこれらのトークン AWS WAF を管理する方法を設定できます。

- CAPTCHA およびチャレンジイミュニティ時間 – CAPTCHA またはチャレンジのタイムスタンプの有効期間を指定します。ウェブ ACL 設定は、独自のイミュニティ時間設定が設定されていない

すべてのルール、ならびにアプリケーション統合 SDK にも継承されます。詳細については、「タイムスタンプの有効期限： AWS WAF トークンのイミュニティ時間」を参照してください。

- トークンドメイン – デフォルトでは、 はウェブ ACL が関連付けられているリソースのドメインに対してのみトークン AWS WAF を受け入れます。トークンドメインリストを設定すると、 はリスト内のすべてのドメインと、関連付けられたリソースのドメインのトークン AWS WAF を受け入れます。詳細については、「AWS WAF ウェブ ACL トークンドメインリストの設定」を参照してください。

ウェブ ACL の使用

このセクションでは、 AWS コンソールからウェブ ACL を作成、管理、使用する手順について説明します。

使用しているウェブ ACL のウェブトラフィックメトリクスの概要には、 AWS WAF コンソールのウェブ ACL ページの [トラフィック概要] タブからアクセスできます。コンソールダッシュボードには、 AWS WAF アプリケーションのウェブトラフィックを評価する際に収集される Amazon CloudWatch メトリックスの概要がほぼリアルタイムで表示されます。ダッシュボードのページの詳細については、「ウェブ ACL トラフィック概要ダッシュボード」を参照してください。ウェブ ACL のトラフィックのモニタリングに関する追加情報については、「モニタリングとチューニング」を参照してください。

> **本番稼働トラフィックのリスク**
>
> 本番稼働トラフィックのウェブ ACL に変更をデプロイする前に、ステージング環境またはテスト環境でテストおよびチューニングしてトラフィックへの潜在的な影響を確認します。その後、更新したルールを有効にする前に、本番稼働用トラフィックでカウントモードでルールをテストしてチューニングします。ガイダンスについては、「AWS WAF 保護機能のテストと調整」を参照してください。

> **Note**
>
> ウェブ ACL で 1,500 WCU を超える容量を使用すると、ウェブ ACL の基本料金を超えるコストが発生します。詳細については、「AWS WAF ウェブ ACL キャパシティーユニット (WCUs)」と「AWS WAF 料金表」を参照してください。

更新中の一時的な不一致

ウェブ ACL AWS WAF やその他のリソースを作成または変更した場合、その変更がリソースが保存されているすべての領域に反映されるまでに少し時間がかかります。伝播時間は、数秒から数分までかかります。

次の内容では、変更伝播中に直面する一時的な不整合性の例を紹介します。

- ウェブ ACL を作成した後、それをリソースに関連付けようとすると、ウェブ ACL が利用できないことを示す例外が表示される場合があります。
- ルールグループをウェブ ACL に追加した後、新しいルールグループのルールは、ウェブ ACL が使用される工リアで有効になり、別の工リアでは有効にならない場合があります。
- ルールのアクション設定を変更した後、古いアクションを一部の工リアで確認され、新しいアクションを別の工リアで確認される場合があります。
- ブロックルールで使用されている IP セットに IP アドレスを追加した後、新しいアドレスはある工リアではブロックされ、別の工リアでは許可される場合があります。

トピック

ウェブ ACL の作成

新しいウェブ ACL を作成するには、このページの手順に従ってウェブ ACL 作成ウィザードを使用します。

> ⚠ 本番稼働トラフィックのリスク
>
> 本番稼働トラフィックのウェブ ACL に変更をデプロイする前に、ステージング環境またはテスト環境でテストおよびチューニングしてトラフィックへの潜在的な影響を確認します。その後、更新したルールを有効にする前に、本番稼働用トラフィックでカウントモードでルールをテストしてチューニングします。ガイダンスについては、「AWS WAF 保護機能のテストと調整」を参照してください。

> **Note**
>
> ウェブ ACL で 1,500 WCU を超える容量を使用すると、ウェブ ACL の基本料金を超えるコストが発生します。詳細については、「AWS WAF ウェブ ACL キャパシティーユニット (WCUs)」と「AWS WAF 料金表」を参照してください。

ウェブ ACL を作成するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF でコンソールを開きます。
2. ナビゲーションペインの [Web ACLs] (ウェブ ACL) を選択してから、[Create web ACL] (ウェブ ACL を作成) を選択します。
3. [Name] (名前) で、このウェブ ACL の識別に使用する名前を入力します。

   > **Note**
   >
   > ウェブ ACL の作成後は、名前を変更することはできません。

4. (オプション) 必要に応じて、[Description - optional] (説明 - オプション) に、ウェブ ACL の詳しい説明を入力します。
5. CloudWatch メトリック名については、該当する場合はデフォルト名を変更します。有効な文字については、コンソールのガイダンスに従ってください。名前には、特殊文字、空白、または「All」や「Default_Action」などの専用メトリック名を含めることはできません。 AWS WAF

   > **Note**
   >
   > ウェブ ACL CloudWatch を作成した後でメトリクス名を変更することはできません。

6. [リソースタイプ] で、このウェブ ACL AWS に関連付けるリソースのカテゴリ ([Amazon CloudFront ディストリビューション] または [リージョンリソース]) を選択します。詳細については、「ウェブ ACL とリソースの関連付けまたは関連付け解除 AWS」を参照してください。
7. [リージョン] で [リージョン] リソースタイプを選択した場合は、ウェブ ACL AWS WAF を保存するリージョンを選択します。

   このオプションは、リージョン別リソースタイプの場合にのみ選択する必要があります。CloudFront ディストリビューションの場合、リージョンは米国東部 (バージニア北部) リー

ジョンにus-east-1、グローバル () アプリケーションの場合はハードコーディングされます。CloudFront

8. (CloudFront、API Gateway、Amazon Cognito、App Runner、検証済みアクセス) ウェブリクエストインスペクションのサイズ制限-オプション。別のボディインスペクションサイズ制限を指定する場合は、制限を選択します。デフォルトの 16 KB を超えるボディサイズを検査すると、追加費用が発生する可能性があります。このオプションについては、「本文検査のサイズ制限の管理」を参照してください。
9. (オプション) AWS 関連リソース-オプション。リソースを今すぐ指定する場合は、[リソースの追加 AWS] を選択します。ダイアログボックスで、関連付けるリソースを選択し、「追加」を選択します。 AWS WAF 「ウェブ ACL AWS と関連するリソースの説明」ページに戻ります。
10. [Next] (次へ) を選択します。
11. (オプション) マネージドルールグループを追加する場合は、[Add rules and rule groups] (ルールとルールグループの追加) ページで、[Add rules] (ルールの追加) を選択し、[Add managed rule groups] (マネージドルールグループの追加) を選択します。追加するマネージドルールグループごとに次を実行します。

    a. 「マネージドルールグループの追加」ページで、 AWS AWS Marketplace マネージドルールグループまたは選択した出品者のリストを展開します。

    b. 追加するルールグループでは、[Action] (アクション) 列で [Add to web ACL] (ウェブ ACL に追加) 切り替えボタンをオンにします。

    ウェブ ACL がルールグループを使用する方法をカスタマイズするには、[Edit] (編集) を選択します。一般的なカスタマイズ設定は次のとおりです。

    - 一部またはすべてのルールのルールアクションをオーバーライドします。ルールにオーバーライドアクションを定義しない場合、評価にはルールグループ内で定義されているルールアクションが使用されます。このオプションについては、「ルールグループのアクションオーバーライドオプション」を参照してください。
    - スコープダウンステートメントを追加することで、ルールグループが検査するウェブリクエストのスコープを縮小します。このオプションについては、「スコープダウンステートメント」を参照してください。
    - 一部のマネージドルールグループは追加の設定が必要です。マネージドルールグループのプロバイダーのドキュメントを参照してください。 AWS マネージドルールルールグループに固有の情報については、を参照してくださいAWS のマネージドルール AWS WAF。

    設定が完了したら、[Save rule] (ルールを保存) を選択します。

[Add rules] (ルールの追加) を選択してマネージドルールの追加を終了し、[Add rules and rule groups] (ルールとルールグループの追加) ページに戻ります。

12. (オプション) 独自のルールグループを追加する場合は、[Add rules and rule groups] (ルールとルールグループの追加) ページで、[Add rules] (ルールの追加) を選択し、[Add my own rules and rule groups] (独自のルールとルールグループの追加) を選択します。追加するルールグループごとに次を実行します。

    a. [Add my own rules and rule groups] (独自のルールとルールグループの追加) ページで、[Rule group] (ルールグループ) を選択します。

    b. [Name] (名前) で、このウェブ ACL のルールグループのルールに使用する名前を入力します。`AWS`、`Shield`、`PreFM`、または `PostFM` で始まる名前は使用しないでください。これらの文字列は、予約されているか、他のサービスが管理するルールグループと混同される可能性があります。他のサービスによって提供されるルールグループ を参照してください。

    c. リストからルールグループを選択します。

    > **Note**
    >
    > 独自のルールグループのルールアクションを上書きする場合は、まずそのルールをウェブ ACL に保存し、ウェブ ACL とウェブ ACL のルールリストにあるルールグループ参照ステートメントを編集します。マネージドルールグループの場合と同様に、ルールアクションは任意の有効なアクション設定に上書きできます。

    d. [ルールを追加] を選択します。

13. (オプション) 独自のルールを追加する場合は、[Add rules and rule groups] (ルールとルールグループの追加) ページで、[Add rules] (ルールの追加)、[Add my own rules and rule groups] (独自のルールとルールグループの追加)、[Rule builder] (ルールビルダー)、[Rule visual editor] (ルールビジュアルエディタ) の順に選択します。

    > **Note**
    >
    > コンソールの [Rule visual editor] (ルールビジュアルエディタ) は、1 レベルのネストをサポートします。例えば、単一の論理 `AND` または `OR` ステートメントを使用して、その中に 1 レベルの他のステートメントをネストすることはできますが、論理ステートメントの中に論理ステートメントをネストすることはできません。より複雑なルールステートメントを管理するには、[Rule JSON editor] (ルール JSON エディタ) を使用します。ルールのすべてのオプションについては、「AWS WAF 規則」を参照してください

この手順では、[Rule visual editor] (ルールビジュアルエディタ) について説明します。

a. [Name] (名前) で、このルールの識別に使用する名前を入力します。`AWS`、`Shield`、`PreFM`、または `PostFM` で始まる名前は使用しないでください。これらの文字列は、予約されているか、他のサービスが管理するルールグループと混同される可能性があります。

b. 必要に応じて、ルールの定義を入力します。論理 `AND` および `OR` ルールステートメントの中でルールを組み合わせることができます。ウィザードに、コンテキストに応じた各ルールのオプションが表示されます。ルールのオプションについては、「AWS WAF 規則」を参照してください。

c. [Action] (アクション) で、ウェブリクエストに一致したときにルールによって実行されるアクションを選択します。選択の詳細については、「ルールアクション」と「ウェブ ACL ルールおよびルールグループの評価」を参照してください。

   [CAPTCHA] または [Challenge] アクションを使用している場合、このルールの必要に応じて [Immunity time] (イミュニティ時間) の設定を調整します。設定を指定しない場合、ルールはウェブ ACL から設定を継承します。ウェブ ACL のイミュニティ時間設定を変更するには、ウェブ ACL の作成後にウェブ ACL を編集します。イミュニティ時間の詳細については、「タイムスタンプの有効期限： AWS WAF トークンのイミュニティ時間」を参照してください。

   > **Note**
   >
   > CAPTCHA または Challenge ルールアクションを 1 つのルールで使用、あるいはルールグループでルールアクションのオーバーライドとして使用すると、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

   リクエストまたはレスポンスをカスタマイズする場合は、そのオプションを選択し、カスタマイズの詳細を入力します。詳細については、「AWS WAFのカスタマイズされたウェブリクエストとレスポンス」を参照してください。

   一致するウェブリクエストにルールがラベルを追加するようにする場合は、そのオプションを選択し、ラベルの詳細を入力します。詳細については、「AWS WAF ウェブリクエストの ラベル」を参照してください。

d. [Add Rule] (ルールの追加) を選択します。

14. ウェブ ACL のデフォルトアクションに Block または Allow を選択します。これは、ウェブ ACL のルールがリクエストを明示的に許可または拒否していない場合にリクエストに対して実行されるアクションです。 AWS WAF 詳細については、「ウェブ ACL のデフォルトアクション」を参照してください。

    デフォルトのアクションをカスタマイズする場合は、そのオプションを選択し、カスタマイズの詳細を入力します。詳細については、「AWS WAFのカスタマイズされたウェブリクエストとレスポンス」を参照してください。

15. [Token domain list] (トークンドメインリスト) を定義して、保護されたアプリケーション間でトークンの共有を有効にできます。トークンは、CAPTCHAChallengeアドアクションと、 AWS AWS WAF マネージドルールルールグループを不正防止アカウント作成詐欺防止 (ACFP)、不正防止アカウント乗っ取り防止 (ATP)、 AWS WAF ボットコントロールに使用する際に実装するアプリケーション統合 SDK によって使用されます。 AWS WAF

    パブリックサフィックスは許可されません。たとえば、gov.au または co.uk をトークンドメインとして使用することはできません。

    デフォルトでは、 AWS WAF 保護対象リソースのドメインのトークンのみを受け付けます。このリストにトークンドメインを追加すると、 AWS WAF リスト内のすべてのドメインと関連するリソースのドメインのトークンを受け入れます。詳細については、「AWS WAF ウェブ ACL トークンドメインリストの設定」を参照してください。

16. [次へ] を選択します。

17. 「ルール優先度の設定」ページで、ルールとルールグループを選択し、 AWS WAF 処理したい順序に移動します。 AWS WAF ルールをリストの一番上から処理します。ウェブ ACL を保存すると、 AWS WAF では、リストされている順に、優先順位の数値設定がルールに割り当てられます。詳細については、「ウェブ ACL でのルールおよびルールグループの処理順序」を参照してください。

18. [Next] (次へ) を選択します。

19. [Configure metrics] (メトリクスを設定) ページで、オプションを確認し、必要な更新を適用します。同じメトリクス名を指定することで、CloudWatch 複数のソースのメトリクスを組み合わせることができます。

20. [Next] (次へ) を選択します。

21. [Review and create web ACL] (ウェブ ACL の確認と作成) ページで定義を確認します。エリアを変更する場合は、エリアの [Edit] (編集) を選択します。これにより、ウェブ ACL ウィザードのページに戻ります。変更を加えてから、[Review and create web ACL] (確認してウェブ ACL を作成する) ページに戻るまで、[Next] (次へ) を選択してページを進みます。

22. [Create web ACL] (ウェブ ACL の作成) を選択します。新しいウェブ ACL は、[Web ACLs] (ウェブ ACL) ページにリストされます。

ウェブ ACL の編集

ウェブ ACL のルールを追加、削除、あるいは設定を変更するには、このページの手順を使用してウェブ ACL にアクセスします。ウェブ ACL を更新している間、 AWS WAF ウェブ ACL に関連付けたリソースは継続的に適用されます。

> **本番稼働トラフィックのリスク**
>
> 本番稼働トラフィックのウェブ ACL に変更をデプロイする前に、ステージング環境またはテスト環境でテストおよびチューニングしてトラフィックへの潜在的な影響を確認します。その後、更新したルールを有効にする前に、本番稼働用トラフィックでカウントモードでルールをテストしてチューニングします。ガイダンスについては、「AWS WAF 保護機能のテストと調整」を参照してください。

> **Note**
>
> ウェブ ACL で 1,500 WCU を超える容量を使用すると、ウェブ ACL の基本料金を超えるコストが発生します。詳細については、「AWS WAF ウェブ ACL キャパシティーユニット (WCUs)」と「AWS WAF 料金表」を参照してください。

ウェブ ACL を編集するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。
2. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。
3. 編集するウェブ ACL の名前を選択します。コンソールでウェブ ACL の説明が表示されます。

> **Note**
>
> によって管理されている Web ACL AWS Firewall Manager `FMManagedWebACLV2-` の名前はで始まります。Firewall Manager 管理者は、Firewall Manager AWS WAF ポリシーでこれらを管理します。これらのウェブ ACL の最初と最後には、追加して管理するルー

ルまたはルールグループのいずれかの側で実行するように指定されたルールグループのセットが含まれる場合があります。これらの最初と最後のルールグループの指定はいずれも変更できません。最初と最後のルールグループには、それぞれ PREFMManaged- と POSTFMManaged- で始まる名前が付いています。これらのポリシーの詳細については、「AWS WAF ポリシー」を参照してください。

4. 必要に応じてウェブ ACL を編集します。関心のある設定領域のタブを選択し、ミュータブルな設定を編集します。編集する設定ごとに [保存] を選択してウェブ ACL の説明ページに戻ると、コンソールではウェブ ACL に変更が保存されます。

   ウェブ ACL 設定コンポーネントを含むタブを以下に示します。

   - [ルール] タブ
     - ウェブ ACL で定義したルール – ウェブ ACL で定義したルールは、ウェブ ACL の作成時と同様に編集および管理できます。

       Note

       ウェブ ACL に手動で追加していないルールの名前は変更しないでください。他のサービスを使用してルールを管理している場合、名前を変更すると、そのサービスが意図した保護を提供できなくなったり、機能が低下したりする可能性があります。 AWS Shield Advanced また、 AWS Firewall Manager どちらもウェブ ACL にルールを作成します。詳細については、他のサービスによって提供されるルールグループ を参照してください。

       Note

       ルールの名前を変更し、その変更をルールのメトリクス名に反映させたい場合は、メトリクス名も更新する必要があります。 AWS WAF ルール名を変更しても、ルールのメトリック名は自動的に更新されません。ルールの JSON エディターを使用して、コンソールでルールを編集するときに、メトリック名を変更できます。API や、ウェブ ACL またはルールグループの定義に使用する JSON リストを使用して、両方の名前を変更することもできます。

ルールおよびルールグループの設定については、「AWS WAF 規則」と「AWS WAF ルールグループ」を参照してください。

- [使用するウェブ ACL ルールキャパシティーユニット] – ウェブ ACL の現在のキャパシティ使用量。これは表示のみです。
- [どのルールにも一致しないリクエストに対するデフォルトのウェブ ACL アクション] – この設定の詳細については、「ウェブ ACL のデフォルトアクション」を参照してください。
- [ウェブ ACL CAPTCHA およびチャレンジ設定] – これらのイミュニティ時間によって、CAPTCHA またはチャレンジトークンの取得後の有効期間が決まります。ウェブ ACL の作成後、この設定は、このタブでしか変更できません。これらの設定については、「タイムスタンプの有効期限： AWS WAF トークンのイミュニティ時間」を参照してください。
- トークンドメインリスト — AWS WAF リスト内のすべてのドメインと関連するリソースのドメインのトークンを受け入れます。詳細については、「AWS WAF ウェブ ACL トークンドメインリストの設定」を参照してください。

- 「 AWS 関連リソース」タブ
  - Web リクエストインスペクションのサイズ制限 — CloudFront ディストリビューションを保護する Web ACL にのみ含まれています。ボディインスペクションのサイズ制限によって、 AWS WAF ボディコンポーネントのどれだけの量を検査に送るかが決まります。この設定の詳細については、「本文検査のサイズ制限の管理」を参照してください。
  - AWS 関連リソース — ウェブ ACL が現在関連付けられ保護されているリソースのリスト。ウェブ ACL と同じリージョン内にあるリソースを見つけて、ウェブ ACL に関連付けることができます。詳細については、「ウェブ ACL とリソースの関連付けまたは関連付け解除 AWS」を参照してください。
- [カスタムレスポンス本文] タブ
  - アクションが Block に設定されているウェブ ACL ルールで使用できるカスタムレスポンス本文。詳細については、「Block アクションのカスタムレスポンス」を参照してください。
- [ログ記録とメトリクス] タブ
  - ログ記録 – ウェブ ACL で評価されるトラフィックのログ記録。詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。
  - サンプリングされたリクエスト – ウェブリクエストに一致するルールに関する情報。サンプリングされたリクエストの表示方法については、「ウェブリクエストのサンプルの表示」を参照してください。

- CloudWatch メトリクス — ウェブ ACL 内のルールのメトリクス。Amazon CloudWatch メトリックスについて詳しくは、を参照してくださいAmazon によるモニタリング CloudWatch。

更新中の一時的な不一致

ウェブ ACL AWS WAF やその他のリソースを作成または変更した場合、その変更がリソースが保存されているすべての領域に反映されるまでに少し時間がかかります。伝播時間は、数秒から数分までかかります。

次の内容では、変更伝播中に直面する一時的な不整合性の例を紹介します。

- ウェブ ACL を作成した後、それをリソースに関連付けようとすると、ウェブ ACL が利用できないことを示す例外が表示される場合があります。
- ルールグループをウェブ ACL に追加した後、新しいルールグループのルールは、ウェブ ACL が使用されるエリアで有効になり、別のエリアでは有効にならない場合があります。
- ルールのアクション設定を変更した後、古いアクションを一部のエリアで確認され、新しいアクションを別のエリアで確認される場合があります。
- ブロックルールで使用されている IP セットに IP アドレスを追加した後、新しいアドレスはあるエリアではブロックされ、別のエリアでは許可される場合があります。

ウェブ ACL でのルールグループの動作の管理

このセクションでは、ウェブ ACL でルールグループを使用する方法を変更するオプションについて説明します。この情報は、すべてのルールグループタイプに適用されます。ルールグループをウェブ ACL に追加すると、ルールグループ内の個々ルールのアクションを Count またはその他の有効なルールアクション設定にオーバーライドできます。ルールグループの結果として生じるアクションを Count にオーバーライドすることもできます。これは、ルールグループ内でルールがどのように評価されるかについて影響されません。

これらのオプションについては、「ルールグループのアクションオーバーライドオプション」を参照してください。

ルールグループ内のルールアクションのオーバーライド

ウェブ ACL の各ルールグループにおいて、一部またはすべてのルールに含まれているルールのアクションをオーバーライドできます。

この場合の最も一般的な使用例は、ルールアクションを Count にオーバーライドし、新しいまたは更新されたルールをテストすることです。メトリクスを有効にしている場合は、オーバーライドしたルールごとにメトリクスを受け取ります。テストの詳細については、「AWS WAF 保護機能のテストと調整」を参照してください。

ルールグループのルールアクションのオーバーライド方法

これらの変更は、マネージドルールグループをウェブ ACL に追加するときに行うことができ、ウェブ ACL を編集するときにはどのタイプのルールグループにも変更できます。この手順は、ウェブ ACL にすでに追加されているルールグループを対象としています。このオプションに関する追加情報については、を参照してくださいルールグループのルールアクションの上書き。

1. ウェブ ACL を編集します。
2. ウェブ ACL ページの [Rules] (ルール) タブで、ルールグループを選択し、[Edit] (編集) を選択します。
3. ルールグループの [Rules] (ルール) セクションで、必要に応じてアクション設定を管理します。
   - すべてのルール – ルールグループ内のすべてのルールにオーバーライドアクションを設定するには、[Override all rule actions] (すべてのルールアクションをオーバーライド) ドロップダウンを開いてオーバーライドアクションを選択します。すべてのルールのオーバーライドを削除するには、[Remove all overrides] (すべてのオーバーライドを削除) を選択します。
   - 単一ルール – 単一ルールにオーバーライドアクションを設定するには、ルールのドロップダウンを開いてオーバーライドアクションを選択します。ルールのオーバーライドを削除するには、ルールのドロップダウンを開いて [Remove override] (オーバーライドを削除) を選択します。
4. 変更が完了したら、[Save Rule] (ルールを保存) を選択します。ルールアクションおよびオーバーライドアクション設定は、ルールグループページに一覧表示されます。

次の JSON リストの例は、ルール `CategoryVerifiedSearchEngine` および `CategoryVerifiedSocialMedia` に対してルールアクションを Count にオーバーライドするウェブ ACL 内のルールグループ宣言を示しています。JSON では、個々ルールごとに `RuleActionOverrides` エントリを指定することで、すべてのルールアクションをオーバーライドします。

```
{
    "Name": "AWS-AWSBotControl-Example",
   "Priority": 5,
   "Statement": {
```

```
    "ManagedRuleGroupStatement": {
        "VendorName": "AWS",
        "Name": "AWSManagedRulesBotControlRuleSet",
        "RuleActionOverrides": [
          {
            "ActionToUse": {
              "Count": {}
            },
            "Name": "CategoryVerifiedSearchEngine"
          },
          {
            "ActionToUse": {
              "Count": {}
            },
            "Name": "CategoryVerifiedSocialMedia"
          }
        ],
        "ExcludedRules": []
    },
   "VisibilityConfig": {
       "SampledRequestsEnabled": true,
       "CloudWatchMetricsEnabled": true,
       "MetricName": "AWS-AWSBotControl-Example"
   }
}
```

ルールグループの評価結果を Count にオーバーライド

ルールグループ内のルールの設定または評価方法を変更せずに、ルールグループ評価の結果によって発生するアクションをオーバーライドできます。このオプションは一般的に使用されません。ルールグループ内のいずれかのルールが一致した場合、このオーバーライドはルールグループの結果として生じるアクションを Count に設定します。

> **Note**
>
> これは珍しいユースケースです。ほとんどのアクションオーバーライドは、で説明されているように、ルールグループ内のルールレベルで行われます。ルールグループ内のルールアクションのオーバーライド

ルールグループを追加または編集するとき、ウェブ ACL 内でルールグループの結果として生じるのアクションをオーバーライドできます。コンソールで、ルールグループの [Override rule group

action - optional] (ルールグループアクションのオーバーライド - オプション) ペインを開いてオーバーライドを有効にします。次のリストの例に示すように、JSON セットで `OverrideAction` をルールグループステートメントに設定します。

```
{
   "Name": "AWS-AWSBotControl-Example",
   "Priority": 5,
   "Statement": {
    "ManagedRuleGroupStatement": {
     "VendorName": "AWS",
     "Name": "AWSManagedRulesBotControlRuleSet"
     }
   },
    "OverrideAction": {
       "Count": {}
    },
   "VisibilityConfig": {
        "SampledRequestsEnabled": true,
        "CloudWatchMetricsEnabled": true,
        "MetricName": "AWS-AWSBotControl-Example"
   }
}
```

ウェブ ACL とリソースの関連付けまたは関連付け解除 AWS

を使用すると AWS WAF 、ウェブ ACL とリソースとの間に次のような関連付けを作成できます。

- 地域のウェブ ACL を以下のリージョナルリソースのいずれかに関連付けます。このオプションでは、ウェブ ACL はリソースと同じ地域にある必要があります。
  - Amazon API Gateway REST API
  - Application Load Balancer
  - AWS AppSync GraphQL API
  - Amazon Cognito ユーザープール
  - AWS App Runner サービス
  - AWS 検証済みアクセスインスタンス
- グローバルウェブ ACL を Amazon CloudFront ディストリビューションに関連付けます。グローバルウェブ ACL には、米国東部 (バージニア北部) リージョンのハードコードリージョンを持ちます。

CloudFront ディストリビューション自体を作成または更新するときに、ウェブ ACL をディストリビューションに関連付けることもできます。詳細については、Amazon CloudFront 開発者ガイドの「AWS WAF コンテンツへのアクセスを制御するための使用」を参照してください。

複数の関連付けに関する制限

以下の制限に従って、1 つのウェブ ACL を 1 AWS つ以上のリソースに関連付けることができます。

- AWS 各リソースは 1 つのウェブ ACL にのみ関連付けることができます。ウェブ ACL AWS とリソースの関係はです one-to-many。
- ウェブ ACL は 1 CloudFront つ以上のディストリビューションに関連付けることができます。CloudFront ディストリビューションに関連付けたウェブ ACL AWS を他のリソースタイプに関連付けることはできません。

追加の制限

ウェブ ACL の関連付けについて、次の追加制限が適用されます。

- ウェブ ACL は、 AWS リージョン内の Application Load Balancer にのみ関連付けることができます。例えば、ウェブ ACL を AWS Outpostsにある Application Load Balancer に関連付けることはできません。
- Amazon Cognito ユーザープールを、 AWS WAF 不正防止アカウント作成詐欺防止 (ACFP) `AWSManagedRulesACFPRuleSet` マネージドルールグループまたは不正防止アカウント乗っ取り防止 (ATP) マネージドルールグループを使用するウェブ ACL に関連付けることはできません。AWS WAF `AWSManagedRulesATPRuleSet`Account Creation Fraud Prevention については、「AWS WAF 不正防止アカウント作成詐欺防止 (ACFP)」を参照してください。アカウント乗っ取り防止の情報については、「AWS WAF 不正防止アカウント乗っ取り防止 (ATP)」を参照してください。

> **本番稼働トラフィックのリスク**
>
> 本番稼働トラフィックにウェブ ACL をデプロイする前に、トラフィックへの潜在的な影響に慣れるまで、ステージング環境またはテスト環境でテストおよびチューニングします。その後、ルールを有効にする前に、本番稼働用トラフィックでカウントモードでルールをテストしてチューニングします。ガイダンスについては、「AWS WAF 保護機能のテストと調整」を参照してください。

ウェブ ACL をリソースに関連付けるには AWS

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。
2. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。
3. リソースに関連付けるウェブ ACL の名前を選択します。コンソールでウェブ ACL の説明が表示され、そこで編集できます。
4. [AWS 関連リソース] タブで [AWS リソースの追加] を選択します。
5. プロンプトが表示されたら、リソースの種類を選択し、関連付けるリソースの横にあるラジオボタンを選択してから、[Add] (追加) を選択します。

ウェブ ACL とリソースの関連付けを解除するには AWS

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。
2. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。
3. リソースとの関連付けを解除するウェブ ACL の名前を選択します。コンソールでウェブ ACL の説明が表示され、そこで編集できます。
4. 「 AWS 関連リソース」タブで、このウェブ ACL の関連付けを解除したいリソースを選択します。

   > **Note**
   >
   > 一度に 1 つのリソースの関連付けを解除する必要があります。リソースを選択する際に複数選択しないでください。

5. [Disassociate] (関連付け解除) を選択します。コンソールに確認ダイアログが表示されます。ウェブ ACL とリソースの関連付けを解除する選択を確認します。 AWS

ウェブ ACL の削除

ウェブ ACL を削除するには、まずウェブ ACL AWS からすべてのリソースの関連付けを解除します。次の手順を実行します。

ウェブ ACL を削除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。
2. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。
3. 削除するウェブ ACL の名前を選択します。コンソールでウェブ ACL の説明が表示され、そこで編集できます。
4. 「 AWS 関連リソース」タブでは、関連する各リソースについて、リソース名の横にあるラジオボタンを選択し、「関連付け解除」を選択します。これにより、ウェブ ACL とリソースの関連付けが解除されます。 AWS
5. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。
6. 削除するウェブ ACL の横にあるラジオボタンを選択し、[Delete] (削除) を選択します。

AWS WAF ルールグループ

ルールグループは、ウェブ ACL に追加できる再利用可能なルールのセットです。ウェブ ACL の詳細については、「AWS WAF ウェブアクセスコントロールリスト (ウェブ ACLs)」を参照してください。

ルールグループは、主に次のカテゴリに分類されます。

- ユーザー独自のルールグループは、ユーザーが作成して管理します。
- マネージドルールチームが作成および管理する AWS マネージドルールグループ。
- AWS Marketplace 販売者が作成および管理するマネージドルールグループ。
- や Shield Advanced などの他のサービスによって所有 AWS Firewall Manager および管理されるルールグループ。

ルールグループとウェブ ACL の相違点

ルールグループとウェブ ACL には、どちらもルールが含まれています。ルールは、両方の場所で同じ方法で定義されます。ルールグループは、次の点でウェブ ACL と異なります。

- ルールグループには、ルールグループ参照ステートメントを含めることはできません。
- 各ウェブ ACL にルールグループリファレンスステートメントを追加することで、複数のウェブ ACL で 1 つのルールグループを再利用できます。ウェブ ACL を再利用することはできません。

- ルールグループにはデフォルトのアクションがありません。ウェブ ACL では、含めるルールまたはルールグループごとにデフォルトのアクションを設定します。ルールグループまたはウェブ ACL 内の個々のルールには、アクションが定義されています。
- ルールグループを AWS リソースに直接関連付けることはありません。ルールグループを使用してリソースを保護するには、ウェブ ACL でルールグループを使用します。
- ウェブ ACL のシステム定義の最大容量は、5,000 WCU です。各ルールグループには、作成時に設定する必要がある WCU 設定があります。この設定を使用して、ルールグループを使用してウェブ ACL に追加される追加の容量要件を計算できます。WCU の詳細については、「AWS WAF ウェブ ACL キャパシティーユニット (WCUs)」を参照してください。

ルールについては、「AWS WAF 規則」を参照してください。

このセクションでは、独自のルールグループを作成および管理するためのガイダンス、使用できるマネージドルールグループの説明、マネージドルールグループの使用に関するガイダンスを提供します。

トピック

- マネージドルールグループ
- 独自のルールグループの管理
- 他のサービスによって提供されるルールグループ

マネージドルールグループ

マネージドルールグループは、 AWS と AWS Marketplace 販売者がユーザーに代わって記述および維持する、事前定義された ready-to-use ルールのコレクションです。マネージドルールグループの使用には基本 AWS WAF 料金が適用されます。 AWS WAF 料金情報については、「 AWS WAF の料金」を参照してください。

- AWS WAF Bot Control、 AWS WAF Fraud Control アカウント乗っ取り防止 (ATP)、および AWS WAF Fraud Control アカウント作成不正防止 (ACFP) の AWS マネージドルールルールグループは、基本 AWS WAF 料金以外の追加料金で利用できます。料金の詳細については、「AWS WAF の料金」を参照してください。
- 他のすべての AWS マネージドルールルールグループは、追加料金なしで AWS WAF お客様にご利用いただけます。
- AWS Marketplace マネージドルールグループは、 を通じてサブスクリプションで使用できます AWS Marketplace。これらのルールグループはそれぞれ、 AWS Marketplace 販売者が所有および

管理します。 AWS Marketplace マネージドルールグループを使用するための料金情報については、 AWS Marketplace 販売者にお問い合わせください。

一部のマネージドルールグループは、、JoomlaWordPress、PHP などの特定のタイプのウェブアプリケーションを保護するように設計されています。「OWASP Top 10」にリストされているものを含め、既知の脅威や一般的なウェブアプリケーションの脆弱性に対する幅広い保護を提供するものもあります。PCI や HIPAA などの規制の遵守が必要な場合は、マネージドルールグループを使用してウェブアプリケーションファイアウォールの要件を満たすことができます。

自動更新

絶えず変化する脅威の状況に遅れずについていくには、時間とコストがかかることがあります。マネージドルールグループを使用すると、 AWS WAFを実装して使用する際の時間を節約できます。多くの AWS および AWS Marketplace 販売者は、新しい脆弱性や脅威が発生したときに、マネージドルールグループを自動的に更新し、新しいバージョンのルールグループを提供します。

場合によっては、 AWS は多くのプライベート開示コミュニティに参加しているため、公開前に新しい脆弱性が通知されます。このような場合は、新しい脅威が広く知られる前でも、 AWS マネージドルールのルールグループを更新してデプロイ AWS できます。

マネージドルールグループのルールへの制限付きアクセス

各マネージドルールグループには、どのようなタイプの攻撃や脆弱性に対して保護するように設計されているかが包括的に定義されています。ルールグループプロバイダーの知的財産を保護するために、ルールグループ内の個々のルールのすべての詳細を表示することはできません。この制限は、悪意のあるユーザーが公開されたルールを特に回避する脅威を設計するのを防ぐのにも役立ちます。

トピック

- バージョニングされたマネージドルールグループ
- マネージドルールグループの使用
- AWS のマネージドルール AWS WAF
- AWS Marketplace マネージドルールグループ

バージョニングされたマネージドルールグループ

多くのマネージドルールグループプロバイダーは、バージョニングを使用してルールグループのオプションと機能を更新します。通常、マネージドルールグループの特定のバージョンは静的です。場

合によっては、プロバイダーがセキュリティ上の新たな脅威に対応するために、マネージドルールグループの静的バージョンの一部またはすべてを更新する必要があることがあります。

ウェブ ACL でバージョン管理されたマネージドルールグループを使用する場合、デフォルトバージョンを選択し、使用する静的バージョンをプロバイダーに管理させるか、特定の静的バージョンを選択できます。

必要なバージョンが見つかりませんか?

ルールグループのバージョン一覧にバージョンが表示されない場合は、そのバージョンの有効期限の失効が予定されているか、すでに期限切れになっている可能性があります。バージョンの有効期限がスケジュールされると、 ではルールグループに対してバージョンを選択 AWS WAF できなくなります。

AWS マネージドルールのルールグループの SNS 通知

AWS マネージドルールのルールグループはすべて、IP 評価ルールグループを除き、バージョニングと SNS 更新の通知を提供します。通知を提供する AWS マネージドルールのルールグループは、すべて同じ SNS トピックの Amazon リソースネーム (ARN) を使用します。SNS 通知にサインアップするには、「」を参照してください新しいバージョンとアップデートの通知を受け取る。

トピック

- マネージドルールグループのバージョンライフサイクル
- マネージドルールグループのバージョンの有効期限
- マネージドルールグループのバージョン処理に関するベストプラクティス

マネージドルールグループのバージョンライフサイクル

プロバイダーは、マネージドルールグループの静的バージョンの次のライフサイクルステージを処理します。

- リリースとアップデート – マネージドルールグループのプロバイダーは、Amazon Simple Notification Service (Amazon SNS) トピックに対する通知を通じて、マネージドルールグループの今後のバージョンと新しい静的バージョンを知らせます。また、プロバイダーは、緊急時の必須の更新など、ルールグループに関するその他の重要な情報を伝達するためにトピックを使用する場合もあります。

  ルールグループのトピックをサブスクライブし、通知の受信方法を設定できます。詳細については、「新しいバージョンとアップデートの通知を受け取る」を参照してください。

- 有効期限のスケジュール – マネージドルールグループのプロバイダーは、古いバージョンのルールグループの有効期限をスケジュールします。失効予定のバージョンは、ウェブ ACL ルールに追加できません。バージョンの有効期限がスケジュールされると、 は Amazon のカウントダウンメトリクスを使用して有効期限 AWS WAF を追跡します CloudWatch。
- バージョンの有効期限 – マネージドルールグループの期限切れバージョンを使用するようにウェブ ACL を設定している場合、ウェブ ACL の評価中に、 はルールグループのデフォルトバージョン AWS WAF を使用します。さらに、 は、ルールグループを削除したり、そのバージョンを有効期限が切れていないバージョンに変更したりしないウェブ ACL の更新を AWS WAF ブロックします。

AWS Marketplace マネージドルールグループを使用する場合は、バージョンライフサイクルに関する追加情報をプロバイダーに依頼してください。

マネージドルールグループのバージョンの有効期限

ルールグループの特定のバージョンを使用する場合は、有効期限切れのバージョンを使用し続けないようにしてください。バージョンの有効期限は、ルールグループの SNS 通知と Amazon CloudWatch メトリクスを通じてモニタリングできます。

ウェブ ACL で使用しているバージョンの有効期限が切れている場合、 は、ルールグループを有効期限が切れていないバージョンに移動するなど、ウェブ ACL の更新を AWS WAF ブロックします。ルールグループを利用可能なバージョンに更新することも、ウェブ ACL から削除することもできます。

マネージドルールグループの有効期限の処理は、ルールグループプロバイダーによって異なります。AWS マネージドルールのルールグループの場合、期限切れのバージョンは自動的にルールグループのデフォルトバージョンに変更されます。 AWS Marketplace ルールグループについては、有効期限の処理方法をプロバイダーに依頼してください。

プロバイダーは、ルールグループの新しいバージョンを作成するときに、バージョンの予測される有効期間を設定します。バージョンの有効期限が切れるようにスケジュールされていない間、Amazon CloudWatch メトリクス値は予測された有効期間設定に設定され、 では CloudWatchメトリクスのフラットな値が表示されます。プロバイダーがメトリクスの有効期限をスケジュールすると、メトリクス値は有効期限の到来時にゼロになるまで、毎日減少します。有効期限切れのモニタリングについては、「」を参照してくださいバージョンの有効期限の追跡。

マネージドルールグループのバージョン処理に関するベストプラクティス

バージョン管理されたマネージドルールグループを使用する場合は、このベストプラクティスのガイダンスに従ってバージョニングを行ってください。

ウェブ ACL でマネージドルールグループを使用する場合は、ルールグループの特定の静的バージョンを使用するか、デフォルトバージョンを使用するように選択できます。

- デフォルトバージョン — AWS WAF 常にデフォルトバージョンをプロバイダーが現在推奨している静的バージョンに設定します。推奨される静的バージョンをプロバイダーが更新すると、 AWS WAF は、ウェブ ACL のルールグループのデフォルトバージョンの設定を自動的に更新します。

  マネージドルールグループのデフォルトバージョンを使用する場合は、ベストプラクティスとして次の手順を実行します。

  - 通知をサブスクライブする – ルールグループへの変更に関する通知をサブスクライブし、それらを監視します。ほとんどのプロバイダーは、新しい静的バージョンとデフォルトバージョンの変更について事前通知を送信します。これにより、 がデフォルトバージョンを AWS そのバージョンに切り替える前に、新しい静的バージョンの影響を確認できます。詳細については、「新しいバージョンとアップデートの通知を受け取る」を参照してください。
  - デフォルトを新しい静的バージョンに設定する前に、その影響を確認し、必要に応じて調整する – デフォルトを新しい静的バージョンに設定する前に、ウェブリクエストのモニタリングと管理に対する静的バージョンの影響を確認します。新しい静的バージョンには、確認する新しいルールが含まれている可能性があります。ルールグループの使用方法を変更する必要がある場合に備えて、誤検出やその他の予期しない動作を見つけます。例えば、新しい動作の処理方法を把握しながら、トラフィックをブロックしないようにするために、ルールをカウントに設定できます。詳細については、「AWS WAF 保護機能のテストと調整」を参照してください。
- 静的バージョン – 静的バージョンを使用する場合は、ルールグループの新しいバージョンを採用する準備ができたら、バージョン設定を手動で更新する必要があります。

  マネージドルールグループの静的バージョンを使用する場合は、ベストプラクティスとして次の手順を実行します。

  - バージョンを最新の状態に保つ – マネージドルールグループを可能な限り最新バージョンに近いものにします。新しいバージョンがリリースされたら、それをテストし、必要に応じて設定を調整し、適時に実装してください。テストについては、「AWS WAF 保護機能のテストと調整」を参照してください。
  - 通知をサブスクライブする – ルールグループに対する変更に関する通知をサブスクライブして、プロバイダーが新しい静的バージョンをいつリリースするかを把握します。ほとんどのプロ

バイダーは、バージョン変更に関する事前の通知を提供します。さらに、セキュリティホールをふさぐため、またはその他の緊急の理由で、使用している静的バージョンをプロバイダーが更新する必要がある場合があります。プロバイダーの通知をサブスクライブすると、状況について知ることができます。詳細については、「新しいバージョンとアップデートの通知を受け取る」を参照してください。

- [Avoid version expiration] (バージョンの有効期限切れを回避する) - これ使用している間は、静的バージョンの有効期限が切れないようにします。期限切れのバージョンのプロバイダーによる処理はさまざまであり、使用可能なバージョンへのアップグレードの強制や、予期しない結果をもたらす可能性のあるその他の変更が含まれる場合があります。 AWS WAF 有効期限メトリクスを追跡し、サポートされているバージョンに正常にアップグレードするのに十分な日数を示すアラームを設定します。詳細については、「バージョンの有効期限の追跡」を参照してください。

マネージドルールグループの使用

このセクションでは、マネージドルールグループにアクセスして管理するためのガイダンスを提供します。

マネージドルールグループをウェブ ACL に追加すると、独自のルールグループと同じ設定オプションに加えて、追加設定を選択できます。

コンソールで、ウェブ ACL でルールを追加および編集するプロセス中に、マネージドルールグループの情報にアクセスします。API とコマンドラインインターフェイス (CLI) を通じて、マネージドルールグループの情報を直接リクエストできます。

ウェブ ACL でマネージドルールグループを使用する場合、次の設定を編集できます。

- [Version] (バージョン) – これは、ルールグループがバージョン管理されている場合にのみ使用できます。詳細については、「バージョニングされたマネージドルールグループ」を参照してください。
- ルールアクションのオーバーライド – ルールグループ内のルールのアクションを任意のアクションにオーバーライドできます。Count に設定すると、ルールグループを使用してウェブリクエストを管理する前に、そのルールグループをテストするときに便利です。詳細については、「ルールグループのルールアクションの上書き」を参照してください。
- [Scope-down statement] (スコープダウンステートメント) – スコープダウンステートメントを追加して、ルールグループで評価したくないウェブリクエストを除外できます。詳細については、「スコープダウンステートメント」を参照してください。

- [Override rule group action] (ルールグループアクションを上書き) – ルールグループ評価の結果として生じるアクションを上書きして、Count のみに設定できます。このオプションは、一般的に使用されません。 AWS WAF ルールグループ内のルールを評価する方法は変わりません。詳細については、「ルールグループがアクションの上書きを に返す Count」を参照してください。

ウェブ ACL でマネージドルールグループの設定を編集するには

- コンソール
  - (オプション) マネージドルールグループをウェブ ACL に追加する場合、[Edit] (編集) を選択して、設定を表示および編集できます。
  - (オプション) マネージドルールグループをウェブ ACL に追加したら、[Web ACLs] (ウェブ ACL) ページから、作成したばかりのウェブ ACL を選択します。これにより、ウェブ ACL 編集ページが表示されます。
    - [Rules] (ルール) を選択します。
    - ルールグループを選択し、[Edit] (編集) を選択して、設定を表示および編集します。
- API および CLI - コンソールの外部でウェブ ACL を作成および更新するときに、マネージドルールグループの設定を管理できます。

マネージドルールグループのリストの取得

ウェブ ACL で使用可能なマネージドルールグループのリストを取得できます。このリストには、以下が含まれます。

- AWS すべてのマネージドルールルールグループ。
- AWS Marketplace 購読しているルールグループ。

> **Note**
>
> AWS Marketplace ルールグループへの登録については、を参照してください。AWS Marketplace マネージドルールグループ

マネージドルールグループのリストを取得すると、返されるリストは、使用しているインターフェイスによって異なります。

- コンソール — コンソールには、まだ登録していないルールグループを含め、 AWS Marketplace すべてのマネージドルールグループが表示されます。まだサブスクライブしていないものについては、インターフェイスにサブスクライブするためのリンクが用意されています。
- API および CLI - コンソールの外部では、リクエストは使用可能なルールグループのみを返します。

マネージドルールグループのリストを取得するには

- コンソール - ウェブ ACL の作成プロセス 中に、[Add rules and rule groups] (ルールとルールグループの追加) ページで [Add managed rule groups] (マネージドルールグループの追加) を選択します。最上位レベルには、プロバイダー名が一覧表示されます。各プロバイダーリストを展開して、マネージドルールグループのリストを表示します。バージョン対応ルールグループでは、このレベルで表示される情報はデフォルトバージョンの情報です。マネージドルールグループをウェブ ACL に追加すると、コンソールに命名スキーム `<Vendor Name>-<Managed Rule Group Name>` に基づいて一覧表示されます。
- API –
  - `ListAvailableManagedRuleGroups`
- CLI –
  - `aws wafv2 list-available-managed-rule-groups --scope=<CLOUDFRONT|REGIONAL>`

マネージドルールグループのルールの取得

マネージドルールグループ内のルールのリストを取得できます。API および CLI 呼び出しは、JSON モデル内またはそれを介して参照できるルール仕様を返します AWS CloudFormation。

マネージドルールグループ内のルールの一覧を取得するには

- コンソール
  - (オプション) マネージドルールグループをウェブ ACL に追加する場合、[Edit] (編集) を選択してルールを表示できます。
  - (オプション) マネージドルールグループをウェブ ACL に追加したら、[Web ACLs] (ウェブ ACL) ページから、作成したばかりのウェブ ACL を選択します。これにより、ウェブ ACL 編集ページが表示されます。
    - [Rules] (ルール) を選択します。

    - ルールリストを確認したいルールグループを選択し、[Edit] を選択します。 AWS WAF ルールグループ内のルールの一覧が表示されます。
- API – `DescribeManagedRuleGroup`
- CLI – `aws wafv2 describe-managed-rule-group --scope=<CLOUDFRONT|REGIONAL> --vendor-name <vendor> --name <managedrule_name>`

マネージドルールグループで使用可能なバージョンの取得

マネージドルールグループの利用可能なバージョンは、まだ期限切れになる予定がないバージョンです。このリストは、ルールグループの現在のデフォルトバージョンであるバージョンを示します。

マネージドルールグループの使用可能なバージョンのリストを取得するには

- コンソール
    - (オプション) マネージドルールグループをウェブ ACL に追加する場合は、[Edit] (編集) を選択して、ルールグループの情報を表示します。[Version] (バージョン) ドロップダウンを展開して、使用可能なバージョンのリストを表示します。
    - (オプション) マネージドルールグループをウェブ ACLに追加したら、ウェブ ACL で [Edit] (編集) を選択し、ルールグループルールを選択して編集します。[Version] (バージョン) ドロップダウンを展開して、使用可能なバージョンのリストを表示します。
- API –
    - `ListAvailableManagedRuleGroupVersions`
- CLI –
    - `aws wafv2 list-available-managed-rule-group-versions --scope=<CLOUDFRONT|REGIONAL> --vendor-name <vendor> --name <managedrule_name>`

コンソールを通じたウェブ ACL へのマネージドルールグループの追加

このガイダンスは、 AWS AWS Marketplace すべてのマネージドルールルールグループと、登録しているルールグループに適用されます。

> ⚠ 本番稼働トラフィックのリスク
>
> 本番稼働トラフィックのウェブ ACL に変更をデプロイする前に、ステージング環境またはテスト環境でテストおよびチューニングしてトラフィックへの潜在的な影響を確認します。そ

の後、更新したルールを有効にする前に、本番稼働用トラフィックでカウントモードでルールをテストしてチューニングします。ガイダンスについては、「AWS WAF 保護機能のテストと調整」を参照してください。

> **Note**
>
> ウェブ ACL で 1,500 WCU を超える容量を使用すると、ウェブ ACL の基本料金を超えるコストが発生します。詳細については、「AWS WAF ウェブ ACL キャパシティーユニット (WCUs)」と「AWS WAF 料金表」を参照してください。

コンソールでウェブ ACL にマネージドルールグループを追加するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。
2. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。
3. [Web ACLs] (ウェブ ACL) ページで、ウェブ ACL のリストから、ルールグループを追加するウェブ ACL を選択します。これにより、単一のウェブ ACL のページが表示されます。
4. ウェブ ACL ページで、[Rules] (ルール) タブを選択します。
5. [Rules] (ルール) ペインで、[Add rules] (ルールを追加) を選択してから、[Add managed rule groups] (マネージドルールグループを追加) を選択します。
6. [Add managed rule groups] (マネージドルールグループを追加) ページで、ルールグループのベンダーの選択を展開して、使用可能なルールグループのリストを表示します。
7. 追加するルールグループごとに、[Add to web ACL] (ウェブ ACL に追加) を選択します。ルールグループのウェブ ACL の設定を変更する場合は、[Edit] (編集) を選択し、変更を加えてから、[Save rule] (ルールを保存) を選択します。オプションの詳細については、バージョニングされたマネージドルールグループ のバージョニングに関するガイダンス、および マネージドルールグループステートメント のウェブ ACL でのマネージドルールグループの使用に関するガイダンスを参照してください。
8. [Add managed rule groups] (マネージドルールグループを追加) ページの下部で、[Add rules] (ルールを追加) を選択します。
9. [Set rule priority] (ルールの優先度を設定) ページで、必要に応じてルールが実行される順序を調整し、[Save] (保存) を選択します。詳細については、「ウェブ ACL でのルールおよびルールグループの処理順序」を参照してください。

ウェブ ACL のページで、追加したマネージドルールグループが [Rules] (ルール) タブに一覧表示されます。

AWS WAF 保護機能の変更は、本番環境のトラフィックに使用する前にテストして調整してください。詳細については、AWS WAF 保護機能のテストと調整 を参照してください。

更新中の一時的な不一致

ウェブ ACL AWS WAF やその他のリソースを作成または変更した場合、その変更がリソースが保存されているすべての領域に反映されるまでに少し時間がかかります。伝播時間は、数秒から数分までかかります。

次の内容では、変更伝播中に直面する一時的な不整合性の例を紹介します。

- ウェブ ACL を作成した後、それをリソースに関連付けようとすると、ウェブ ACL が利用できないことを示す例外が表示される場合があります。
- ルールグループをウェブ ACL に追加した後、新しいルールグループのルールは、ウェブ ACL が使用されるエリアで有効になり、別のエリアでは有効にならない場合があります。
- ルールのアクション設定を変更した後、古いアクションを一部のエリアで確認され、新しいアクションを別のエリアで確認される場合があります。
- ブロックルールで使用されている IP セットに IP アドレスを追加した後、新しいアドレスはあるエリアではブロックされ、別のエリアでは許可される場合があります。

マネージドルールグループに対する新しいバージョンと更新の通知を受け取る

マネージドルールグループプロバイダーは、SNS 通知を使用して、今後の新しいバージョンや緊急のセキュリティアップデートなどのルールグループの変更を知らせます。

SNS 通知をサブスクライブするには

ルールグループの通知をサブスクライブするには、米国東部 (バージニア北部) リージョン us-east-1 のルールグループの Amazon SNS トピック ARN の Amazon SNS サブスクリプションを作成します。

サブスクライブする方法については、「Amazon Simple Notification Service デベロッパーガイド」を参照してください。

> **Note**
>
> SNS トピックのサブスクリプションを、us-east-1 リージョンでのみ作成します。

バージョニングされた AWS マネージドルールのルールグループはすべて、同じ SNS トピックの Amazon リソースネーム (ARN) を使用します。 AWS マネージドルールのルールグループ通知の詳細については、「」を参照してくださいデプロイ通知。

マネージドルールグループの Amazon SNS トピック ARN を確認できる場所

AWS マネージドルールのルールグループは 1 つの SNS トピック ARN を使用するため、ルールグループの 1 つからトピック ARN を取得し、サブスクライブして、SNS 通知を提供するすべての AWS マネージドルールのルールグループの通知を取得できます。

- コンソール
  - (オプション) マネージドルールグループをウェブ ACL に追加する場合は、[Edit] (編集) を選択して、ルールグループの Amazon SNS トピック ARN を含むルールグループの情報を表示します。
  - (オプション) マネージドルールグループをウェブ ACL に追加したら、ウェブ ACL で [Edit] (編集) を選択し、ルールグループのルールを選択して編集し、ルールグループの Amazon SNS トピック ARN を表示します。
- API – `DescribeManagedRuleGroup`
- CLI – `aws wafv2 describe-managed-rule-group --scope=<CLOUDFRONT|REGIONAL> --vendor-name <vendor> --name <managedrule_name>`

Amazon SNS 通知形式に関する一般的な情報と、受信する通知をフィルタリングする方法については、「Amazon Simple Notification Service デベロッパーガイド」の「メッセージ形式を解析する」と「Amazon SNS サブスクリプションフィルターポリシー」を参照してください。

ルールグループのバージョンの有効期限の追跡

ルールグループの特定のバージョンを使用する場合は、有効期限切れのバージョンを使用し続けないようにしてください。

> **Tip**
>
> マネージドルールグループの Amazon SNS 通知にサインアップし、マネージドルールグループバージョンで最新の状態を維持します。ルールグループからのほとんどの up-to-date 保護の恩恵を受け、有効期限を先取りできます。詳細については、「新しいバージョンとアップデートの通知を受け取る」を参照してください。

Amazon を介してマネージドルールグループの有効期限スケジューリングをモニタリングするには CloudWatch

1. で CloudWatch、マネージドルールグループの からの有効期限メトリクスを見つけ AWS WAF ます。メトリクスには、次のメトリクス名とディメンションがあります。
   - メトリクス名: DaysToExpiry
   - メトリクスのディメンション: Region、ManagedRuleGroup、Vendor、および Version

   トラフィックを評価するウェブ ACL にマネージドルールグループがある場合、そのメトリクスが取得されます。このメトリクスは、使用しないルールグループでは使用できません。
2. 関心のあるメトリクスにアラームを設定して、新しいバージョンのルールグループに切り替えるよう時間内に通知されるようにします。

Amazon CloudWatch メトリクスの使用とアラームの設定については、「Amazon ユーザーガイド CloudWatch 」を参照してください。

JSON および YAML でのマネージドルールグループ設定の例

API と CLI の呼び出しでは、JSON AWS CloudFormationモデル内またはそれを介して参照できるマネージドルールグループのすべてのルールのリストが返されます。

JSON

JSON を使用して、ルールステートメント内でマネージドルールグループを参照および変更できます。次のリストは`AWSManagedRulesCommonRuleSet`、JSON AWS 形式のマネージドルールルールグループを示しています。RuleActionOverrides 仕様には、アクションが Count にオーバーライドされたルールが一覧表示されています。

```
{
    "Name": "AWS-AWSManagedRulesCommonRuleSet",
```

```
    "Priority": 0,
    "Statement": {
      "ManagedRuleGroupStatement": {
        "VendorName": "AWS",
        "Name": "AWSManagedRulesCommonRuleSet",
        "RuleActionOverrides": [

          {

            "ActionToUse": {

              "Count": {}

            },

            "Name": "NoUserAgent_HEADER"

          }

        ],
        "ExcludedRules": []
      }
    },
    "OverrideAction": {
      "None": {}
    },
    "VisibilityConfig": {
      "SampledRequestsEnabled": true,
      "CloudWatchMetricsEnabled": true,
      "MetricName": "AWS-AWSManagedRulesCommonRuleSet"
    }
}
```

YAML

AWS CloudFormation YAML テンプレートを使用して、ルールステートメント内のマネージドルールグループを参照および変更できます。次のリストは、 AWS AWS CloudFormation テンプレート内のマネージドルールルールグループを示しています。AWSManagedRulesCommonRuleSetRuleActionOverrides 仕様には、アクションが Count にオーバーライドされたルールが一覧表示されています。

```
Name: AWS-AWSManagedRulesCommonRuleSet
Priority: 0
```

```
Statement:
  ManagedRuleGroupStatement:
    VendorName: AWS
    Name: AWSManagedRulesCommonRuleSet
    RuleActionOverrides:
    - ActionToUse:
        Count: {}
      Name: NoUserAgent_HEADER
    ExcludedRules: []
OverrideAction:
  None: {}
VisibilityConfig:
  SampledRequestsEnabled: true
  CloudWatchMetricsEnabled: true
  MetricName: AWS-AWSManagedRulesCommonRuleSet
```

AWS のマネージドルール AWS WAF

AWS Managed Rules for AWS WAF は、一般的なアプリケーションの脆弱性やその他の不要なトラフィックから保護するマネージドサービスです。ウェブ ACL の最大キャパシティユニット (WCU) の上限まで、各ウェブ ACL AWS のマネージドルールから 1 つ以上のルールグループを選択できます。

誤検出の軽減とルールグループの変更のテスト

本番稼働でルールグループを使用する前に、AWS WAF 保護機能のテストと調整 にあるガイダンスに従って、非本番稼働環境でテストします。ウェブ ACL にルールグループを追加する場合、新しいバージョンのルールグループをテストする場合、およびルールグループが必要に応じてウェブトラフィックを処理していないときは、テストとチューニングのガイダンスに従ってください。

セキュリティ責任の共有

AWS マネージドルールは、一般的な Web の脅威からユーザーを保護するように設計されています。マニュアルに従って使用すると、 AWS マネージドルールルールグループはアプリケーションのセキュリティをさらに強化します。ただし、 AWS マネージドルールルールグループは、 AWS 選択したリソースによって決定されるセキュリティ責任の代わりとなるものではありません。責任共有モデルを参照して、 AWS 内のリソースが適切に保護されていることを確認してください。

AWS マネージドルールルールグループリスト

AWS マネージドルールルールグループのルールについて公開する情報は、ルールを使用するための十分な情報を提供することを目的としており、悪意のある人物がルールを回避するために悪用する

可能性のある情報は提供していません。このドキュメントに記載されている以上の情報が必要な場合は、AWS Support センター にお問い合わせください。

このセクションでは、 AWS マネージドルールルールグループの最新バージョンについて説明します。これらの情報は、ウェブ ACL にマネージドルールグループを追加するときにコンソールに表示されます。API を使用すると、 AWS Marketplace `ListAvailableManagedRuleGroups`呼び出しによって登録しているマネージドルールグループとともにこのリストを取得できます。

> **Note**
>
> AWS マネージドルールルールグループのバージョンを取得する方法については、を参照してください。マネージドルールグループで使用可能なバージョンの取得

AWS マネージドルールルールグループはすべてラベル付けをサポートしており、このセクションのルール一覧にはラベル仕様が含まれています。`DescribeManagedRuleGroup` を呼び出すことにより、API を介してマネージドルールグループのラベルを取得できます。ラベルは、応答の AvailableLabels プロパティにリストされています。ラベル付けの詳細については、「AWS WAF ウェブリクエストの ラベル」を参照してください。

AWS WAF 保護への変更は、本番環境のトラフィックに使用する前にテストして調整してください。詳細については、AWS WAF 保護機能のテストと調整 を参照してください。

AWS マネージドルールルールグループ

- ベースラインルールグループ
  - コアルールセット (CRS) マネージドルールグループ
  - 管理者保護マネージドルールグループ
  - 既知の不正な入力マネージドルールグループ
- ユースケース固有のルールグループ
  - SQL データベースマネージドルールグループ
  - Linux オペレーティングシステムマネージドルールグループ
  - POSIX オペレーティングシステムマネージドルールグループ
  - Windows オペレーティングシステムマネージドルールグループ
  - PHP アプリケーションマネージドルールグループ
  - WordPress アプリケーションマネージドルールグループ
- IP 評価ルールグループ

- Amazon IP 評価リストマネージドルールグループ
- 匿名 IP リストマネージドルールグループ
- AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) ルールグループ
  - このルールグループを使用する際の考慮事項
  - このルールグループによって追加されるラベル
    - トークンラベル
    - ACFP ラベル
  - Account Creation Fraud Prevention ルールリスト
- AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループ
  - このルールグループを使用する際の考慮事項
  - このルールグループによって追加されるラベル
    - トークンラベル
    - ATP ラベル
  - アカウント乗っ取り防止のルールリスト
- AWS WAF Bot Control ルールグループ
  - 保護レベル
  - このルールグループを使用する際の考慮事項
  - このルールグループによって追加されるラベル
    - トークンラベル
    - Bot Control ラベル
  - Bot Control のルールリスト

ベースラインルールグループ

ベースラインマネージドルールグループは、さまざまな一般的な脅威に対する一般的な保護を提供します。これらのルールグループを 1 つ以上選択して、リソースのベースライン保護を確立します。

> **Note**
>
> AWS マネージドルールルールグループ内のルールについて公開する情報は、ルールを使用するのに十分な情報を提供することを目的としており、悪意のある人物がルールを回避する

ために悪用する可能性のある情報を提供するものではありません。このドキュメントに記載されている以上の情報が必要な場合は、AWS Support センター にお問い合わせください。

コアルールセット (CRS) マネージドルールグループ

VendorName: AWS、名前:AWSManagedRulesCommonRuleSet、WCU: 700

コアルールセット (CRS) ルールグループには、ウェブアプリケーションに一般的に適用可能なルールが含まれています。これにより、OWASP Top 10 などの OWASP の出版物に記載されている、リスクが高く一般的に発生するいくつかの脆弱性を含む、さまざまな脆弱性の悪用に対する保護が提供されます。 AWS WAF どのようなユースケースにもこのルールグループを使用することを検討してください。

このマネージドルールグループは、評価対象のウェブリクエストにラベルを追加します。このラベルは、ウェブ ACL 内のこのルールグループの後に実行されるルールでも使用できます。 AWS WAF また、ラベルを Amazon CloudWatch メトリックスに記録します。ラベルとラベルメトリクスに関する一般的な情報については、「ウェブリクエストのラベル」および「ラベルメトリクスとディメンション」を参照してください。

Note

この表には、このルールグループの最新の静的バージョンが示されています。他のバージョンでは API コマンドを使用してくださいDescribeManagedRuleGroup。

ルール名	説明とラベル
NoUserAgent_HEADER	HTTP User-Agent ヘッダーが欠落しているリクエストを検査します。 ルールアクション: Block ラベル: awswaf:managed:aws:core-rule-set:NoUserAgent_Header
UserAgent_BadBots_HEADER	リクエストが不正なボットであることを示す一般的な User-Agent ヘッダー値を検査しま

ルール名	説明とラベル
	す。パターンの例には、nessus、nmap などがあります。ボット管理については、「AWS WAF Bot Control ルールグループ」も参照してください。 ルールアクション: Block ラベル: awswaf:managed:aws:core-rule-set:BadBots_Header
SizeRestrictions_QUERYSTRING	2,048 バイトを超える URI クエリ文字列を検査します。 ルールアクション: Block ラベル: awswaf:managed:aws:core-rule-set:SizeRestrictions_QueryString
SizeRestrictions_Cookie_HEADER	10,240 バイトを超える cookie ヘッダーを検査します。 ルールアクション: Block ラベル: awswaf:managed:aws:core-rule-set:SizeRestrictions_Cookie_Header
SizeRestrictions_BODY	8 KB (8,192 バイト) を超えるリクエストボディを検査します。 ルールアクション: Block ラベル: awswaf:managed:aws:core-rule-set:SizeRestrictions_Body

ルール名	説明とラベル
`SizeRestrictions_URIPATH`	1,024 バイトを超える URI パスを検査します。 ルールアクション: Block ラベル: `awswaf:managed:aws:core-rule-set:SizeRestrictions_URIPath`
`EC2MetaDataSSRF_BODY`	リクエストボディから Amazon EC2 メタデータを盗み出す試みがないかを検査します。 **Warning** このルールは、ウェブ ACL とリソースタイプの本文サイズ制限までリクエスト本文を検査するだけです。Application Load Balancer およびの場合 AWS AppSync、制限は 8 KB に固定されています。API Gateway、Amazon Cognito、アプリケーションランナー、検証済みアクセスの場合、デフォルトの制限は 16 KB ですが、ウェブ ACL 設定で制限を最大 64 KB まで増やすことができます。 CloudFrontこのルールは、オーバーサイズコンテンツの処理に `Continue` オプションを使用します。詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。 ルールアクション: Block ラベル: `awswaf:managed:aws:core-rule-set:EC2MetaDataSSRF_Body`

ルール名	説明とラベル
EC2MetaDataSSRF_COOKIE	リクエスト cookie から Amazon EC2 メタデータを盗み出す試みがないかを検査します。 ルールアクション: Block ラベル: awswaf:managed:aws:core-rule-set:EC2MetaDataSSRF_Cookie
EC2MetaDataSSRF_URIPATH	リクエスト URI パスから Amazon EC2 メタデータを盗み出す試みがないかを検査します。 ルールアクション: Block ラベル: awswaf:managed:aws:core-rule-set:EC2MetaDataSSRF_URIPath
EC2MetaDataSSRF_QUERYARGUMENTS	リクエストクエリ引数から Amazon EC2 メタデータを盗み出す試みがないかを検査します。 ルールアクション: Block ラベル: awswaf:managed:aws:core-rule-set:EC2MetaDataSSRF_QueryArguments
GenericLFI_QUERYARGUMENTS	クエリ引数に、ローカルファイルインクルージョン (LFI) を悪用する形跡がないかを検査します。例には、../../ などの手法を使用したパストラバーサルの試行があります。 ルールアクション: Block ラベル: awswaf:managed:aws:core-rule-set:GenericLFI_QueryArguments

ルール名	説明とラベル
GenericLFI_URIPATH	URI パスに、ローカルファイルインクルージョン (LFI) を悪用する形跡がないかを検査します。例には、`../../` などの手法を使用したパストラバーサルの試行があります。 ルールアクション: Block ラベル: `awswaf:managed:aws:core-rule-set:GenericLFI_URIPath`

ルール名	説明とラベル
`GenericLFI_BODY`	リクエストボディに、ローカルファイルインクルージョン (LFI) を悪用する形跡がないかを検査します。例には、`../../` などの手法を使用したパストラバーサルの試行があります。 **Warning** このルールは、ウェブ ACL とリソースタイプの本文サイズ制限までのリクエスト本文のみを検査します。Application Load Balancer およびの場合 AWS AppSync、制限は 8 KB に固定されています。API Gateway、Amazon Cognito、アプリケーションランナー、検証済みアクセスの場合、デフォルトの制限は 16 KB ですが、ウェブ ACL 設定で制限を最大 64 KB まで増やすことができます。 CloudFrontこのルールは、オーバーサイズコンテンツの処理に `Continue` オプションを使用します。詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。 ルールアクション: Block ラベル: `awswaf:managed:aws:core-rule-set:GenericLFI_Body`

ルール名	説明とラベル
RestrictedExtensions_URIPATH	読み取りや実行が安全でないシステムファイルの拡張子が URI パスに含まれているリクエストを検査します。パターンの例には、.log や .ini などの拡張子があります。 ルールアクション: Block ラベル: awswaf:managed:aws:core-rule-set:RestrictedExtensions_URIPath
RestrictedExtensions_QUERYARGUMENTS	クエリ引数に、読み取りや実行が安全でないシステムファイル拡張子が含まれているリクエストを検査します。パターンの例には、.log や .ini などの拡張子があります。 ルールアクション: Block ラベル: awswaf:managed:aws:core-rule-set:RestrictedExtensions_QueryArguments
GenericRFI_QUERYARGUMENTS	IPv4 アドレスを含む URL を埋め込むことにより、Web アプリケーションでRFI (リモートファイルインクルード) を悪用しようとする試みに対して、すべてのクエリパラメーターの値を検査します。例としては、http://、https://、ftp://、ftps://、file などのパターンがあり、悪用の試みに IPv4 ホストヘッダーが含まれています。 ルールアクション: Block ラベル: awswaf:managed:aws:core-rule-set:GenericRFI_QueryArguments

ルール名	説明とラベル
`GenericRFI_BODY`	IPv4 アドレスを含む URL を埋め込むことにより、ウェブアプリケーションの RFI (リモートファイルインクルージョン) を悪用しようとする試行に対してリクエストボディを検査します。例としては、`http://`、`https://`、`ftp://`、`ftps://`、`file://` などのパターンがあり、悪用の試みに IPv4 ホストヘッダーが含まれています。 **Warning** このルールは、ウェブ ACL とリソースタイプの本文サイズ制限までのリクエスト本文のみを検査します。Application Load Balancer および AWS AppSync、制限は 8 KB に固定されています。API Gateway、Amazon Cognito、アプリケーションランナー、検証済みアクセスの場合、デフォルトの制限は 16 KB ですが、ウェブ ACL 設定で制限を最大 64 KB まで増やすことができます。 CloudFrontこのルールは、オーバーサイズコンテンツの処理に `Continue` オプションを使用します。詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。 ルールアクション: Block ラベル: `awswaf:managed:aws:core-rule-set:GenericRFI_Body`

ルール名	説明とラベル
GenericRFI_URIPATH	IPv4 アドレスを含む URL を埋め込むことにより、ウェブアプリケーションの RFI (リモートファイルインクルージョン) を悪用しようとする試行に対してURI パスを検査します。例としては、http://、https://、ftp://、ftps://、file などのパターンがあり、悪用の試みに IPv4 ホストヘッダーが含まれています。 ルールアクション: Block ラベル: awswaf:managed:aws:core-rule-set:GenericRFI_URIPath
CrossSiteScripting_COOKIE	ビルトインを使用して、一般的なクロスサイトスクリプティング (XSS) パターンの Cookie ヘッダーの値を検査します。 AWS WAF クロスサイトスクリプティング攻撃ルールステートメントパターンの例には、<script>alert("hello")</script> などのスクリプトあります。 Note このルールグループのバージョン 2.0 では、 AWS WAF ログ内のルールマッチの詳細は入力されません。 ルールアクション: Block ラベル: awswaf:managed:aws:core-rule-set:CrossSiteScripting_Cookie

ルール名	説明とラベル
`CrossSiteScripting_QUERYARGUMENTS`	ビルトインを使用して、一般的なクロスサイトスクリプティング (XSS) パターンのクエリ引数の値を検査します。 AWS WAF クロスサイトスクリプティング攻撃ルールステートメントパターンの例には、`<script>alert("hello")</script>` などのスクリプトあります。 **Note** このルールグループのバージョン 2.0 では、 AWS WAF ログ内のルールマッチの詳細は入力されません。 ルールアクション: Block ラベル: `awswaf:managed:aws:core-rule-set:CrossSiteScripting_QueryArguments`

ルール名	説明とラベル
`CrossSiteScripting_BODY`	ビルトインを使用してリクエスト本文を検査し、一般的なクロスサイトスクリプティング (XSS) パターンがないか調べます。 AWS WAF クロスサイトスクリプティング攻撃ルールステートメントパターンの例には、`<script>alert("hello")</script>` などのスクリプトあります。 **Note** このルールグループのバージョン 2.0 では、 AWS WAF ログ内のルールマッチの詳細は入力されません。 **Warning** このルールは、ウェブ ACL とリソースタイプの本文サイズ制限までリクエスト本文を検査するだけです。Application Load Balancer およびの場合 AWS AppSync、制限は 8 KB に固定されています。API Gateway、Amazon Cognito、アプリケーションランナー、検証済みアクセスの場合、デフォルトの制限は 16 KB ですが、ウェブ ACL 設定で制限を最大 64 KB まで増やすことができます。 CloudFrontこのルールは、オーバーサイズコンテンツの処理に `Continue` オプションを使用します。詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。

ルール名	説明とラベル
	ルールアクション: Block ラベル: `awswaf:managed:aws:core-rule-set:CrossSiteScripting_Body`
`CrossSiteScripting_URIPATH`	組み込みを使用して URI パスの値を検査し、一般的なクロスサイトスクリプティング (XSS) パターンがないか調べます。 AWS WAF クロスサイトスクリプティング攻撃ルールステートメントパターンの例には、`<script>alert("hello")</script>` などのスクリプトあります。 **Note** このルールグループのバージョン 2.0 では、 AWS WAF ログ内のルールマッチの詳細は入力されません。 ルールアクション: Block ラベル: `awswaf:managed:aws:core-rule-set:CrossSiteScripting_URIPath`

管理者保護マネージドルールグループ

VendorName: AWS、名前:AWSManagedRulesAdminProtectionRuleSet、WCU: 100

管理者保護ルールグループには、公開されている管理ページへの外部アクセスをブロックするためのルールが含まれています。これは、サードパーティーのソフトウェアを実行している場合や、悪意のあるアクターがアプリケーションへの管理アクセスを得るリスクを緩和したい場合に便利です。

このマネージドルールグループは、評価対象のウェブリクエストにラベルを追加します。このラベルは、ウェブ ACL 内のこのルールグループの後に実行されるルールで使用できます。 AWS WAF また、ラベルを Amazon CloudWatch メトリックスに記録します。ラベルとラベルメトリクスに関する一般的な情報については、「ウェブリクエストのラベル」および「ラベルメトリクスとディメンション」を参照してください。

> **Note**
>
> この表には、このルールグループの最新の静的バージョンが示されています。他のバージョンでは API コマンドを使用してくださいDescribeManagedRuleGroup。

ルール名	説明とラベル
`AdminProtection_URIPATH`	一般的にウェブサーバーまたはアプリケーションの管理用に確保されている URI パスの有無を検査します。パターンの例には、`sqlmanager` などがあります。 ルールアクション: Block ラベル: `awswaf:managed:aws:admin-protection:AdminProtection_URIPath`

既知の不正な入力マネージドルールグループ

VendorName: `AWS`、名前:`AWSManagedRulesKnownBadInputsRuleSet`、WCU: 200

既知の不正な入力ルールグループには、無効であることがわかっており脆弱性の悪用または発見に関連するリクエストパターンをブロックするルールが含まれています。これにより、悪意のあるアクターが脆弱なアプリケーションを発見するリスクを緩和できます。

このマネージドルールグループは、評価対象のウェブリクエストにラベルを追加します。このラベルは、ウェブ ACL 内のこのルールグループの後に実行されるルールで使用できます。 AWS WAF また、ラベルを Amazon CloudWatch メトリックスに記録します。ラベルとラベルメトリクスに関する

一般的な情報については、「ウェブリクエストのラベル」および「ラベルメトリクスとディメンション」を参照してください。

> **Note**
>
> この表には、このルールグループの最新の静的バージョンが示されています。他のバージョンでは API コマンドを使用してくださいDescribeManagedRuleGroup。

ルール名	説明とラベル
`JavaDeserializationRCE_HEADER`	HTTP リクエストヘッダーのキーと値に、Spring Core および Cloud Function RCE の脆弱性 (CVE-2022-22963、CVE-2022-22965) などの Java デシリアライゼーション Remote Command Execution (RCE) の試行を示すパターンがないかどうかを検査します。パターンの例には、`(java.lang.Runtime).getRuntime().exec("whoami")` などがあります。 **Warning** このルールは、リクエストヘッダーの最初の 8 KB または最初の 200 個のヘッダーのうち、いずれかの制限に先に達した方のみを検査し、オーバーサイズコンテンツの処理に `Continue` オプションを使用します。詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。 ルールアクション: Block

ルール名	説明とラベル
	ラベル: `awswaf:managed:aws:known-bad-inputs:JavaDeserializationRCE_Header`

ルール名	説明とラベル
`JavaDeserializationRCE_BODY`	リクエスト本文で、Spring Core および Cloud Function RCE の脆弱性 (CVE-2022-22963、CVE-2022-22965) などの Java デシリアライゼーション Remote Command Execution (RCE) の試行を示すパターンがないかどうかを検査します。パターンの例には、`(java.lang.Runtime).getRuntime().exec("whoami")` などがあります。 **Warning** このルールは、ウェブ ACL とリソースタイプの本文サイズ制限までリクエスト本文を検査するだけです。Application Load Balancer および AWS AppSync、制限は 8 KB に固定されています。API Gateway、Amazon Cognito、アプリケーションランナー、検証済みアクセスの場合、デフォルトの制限は 16 KB ですが、ウェブ ACL 設定で制限を最大 64 KB まで増やすことができます。 CloudFrontこのルールは、オーバーサイズコンテンツの処理に `Continue` オプションを使用します。詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。 ルールアクション: Block

ルール名	説明とラベル
	ラベル: awswaf:managed:aws:known-bad-inputs:JavaDeserializationRCE_Body
JavaDeserializationRCE_URIPATH	リクエスト URI で、Spring Core および Cloud Function RCE の脆弱性 (CVE-2022-22963、CVE-2022-22965) などの Java デシリアライゼーション Remote Command Execution (RCE) の試行を示すパターンがないかどうかを検査します。パターンの例には、(java.lang.Runtime).getRuntime().exec("whoami") などがあります。 ルールアクション: Block ラベル: awswaf:managed:aws:known-bad-inputs:JavaDeserializationRCE_URIPath
JavaDeserializationRCE_QUERYSTRING	リクエストクエリ文字列で、Spring Core および Cloud Function RCE の脆弱性 (CVE-2022-22963、CVE-2022-22965) などの Java デシリアライゼーション Remote Command Execution (RCE) の試行を示すパターンがないかどうかを検査します。パターンの例には、(java.lang.Runtime).getRuntime().exec("whoami") などがあります。 ルールアクション: Block ラベル: awswaf:managed:aws:known-bad-inputs:JavaDeserializationRCE_QueryString

ルール名	説明とラベル
Host_localhost_HEADER	リクエストのホストヘッダーに、localhost を示すパターンがないかを検査します。パターンの例には、localhost などがあります。 ルールアクション: Block ラベル: awswaf:managed:aws:known-bad-inputs:Host_Localhost_Header
PROPFIND_METHOD	リクエストの HTTP メソッドに、PROPFIND がないかを検査します。このメソッドは HEAD と同様ですが、XML オブジェクトを抽出しようとする点が異なります。 ルールアクション: Block ラベル: awswaf:managed:aws:known-bad-inputs:Propfind_Method
ExploitablePaths_URIPATH	URI パスに、悪用可能なウェブアプリケーションパスにアクセスする試みがないかを検査します。パターンの例には、web-inf などのパスがあります。 ルールアクション: Block ラベル: awswaf:managed:aws:known-bad-inputs:ExploitablePaths_URIPath

ルール名	説明とラベル
Log4JRCE_HEADER	リクエストヘッダーのキーと値に Log4j の脆弱性 (CVE-2021-44228、CVE-2021-45046、CVE-2021-45105) の存在の有無を検査し、Remote Code Execution (RCE) の試行から保護します。パターンの例には、`${jndi:ldap://example.com/}` などがあります。 **Warning** このルールは、リクエストヘッダーの最初の 8 KB または最初の 200 個のヘッダーのうち、いずれかの制限に先に達した方のみを検査し、オーバーサイズコンテンツの処理に `Continue` オプションを使用します。詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。 ルールアクション: Block ラベル: `awswaf:managed:aws:known-bad-inputs:Log4JRCE_Header`

ルール名	説明とラベル
`Log4JRCE_QUERYSTRING`	クエリ文字列に Log4j の脆弱性 (CVE-2021-44228、CVE-2021-45046、CVE-2021-45105) の存在の有無を検査し、Remote Code Execution (RCE) の試行から保護します。パターンの例には、`${jndi:ldap://example.com/}` などがあります。 ルールアクション: Block ラベル: `awswaf:managed:aws:known-bad-inputs:Log4JRCE_QueryString`

ルール名	説明とラベル
Log4JRCE_BODY	本文に Log4j の脆弱性 (CVE-2021-44228、CVE-2021-45046、CVE-2021-45105) の存在の有無を検査し、Remote Code Execution (RCE) の試行から保護します。パターンの例には、`${jndi:ldap://example.com/}` などがあります。 **Warning** このルールは、ウェブ ACL とリソースタイプの本文サイズ制限までのリクエスト本文のみを検査します。Application Load Balancer および AWS AppSync、制限は 8 KB に固定されています。API Gateway、Amazon Cognito、アプリケーションランナー、検証済みアクセスの場合、デフォルトの制限は 16 KB ですが、ウェブ ACL 設定で制限を最大 64 KB まで増やすことができます。 CloudFrontこのルールは、オーバーサイズコンテンツの処理に `Continue` オプションを使用します。詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。 ルールアクション: Block ラベル: `awswaf:managed:aws:known-bad-inputs:Log4JRCE_Body`

ルール名	説明とラベル
`Log4JRCE_URIPATH`	URI パスに Log4j の脆弱性 (CVE-2021-44228、CVE-2021-45046、CVE-2021-45105) の存在の有無を検査し、Remote Code Execution (RCE) の試行から保護します。パターンの例には、`${jndi:ldap://example.com/}` などがあります。 ルールアクション: Block ラベル: `awswaf:managed:aws:known-bad-inputs:Log4JRCE_URIPath`

ユースケース固有のルールグループ

ユースケース固有のルールグループは、さまざまな AWS WAF ユースケースに対して段階的な保護を提供します。アプリケーションに適用するルールグループを選択します。

> **Note**
>
> AWS マネージドルールのルールグループでルールに対して公開する情報は、不正な攻撃者がルールを回避するために使用できる情報を提供せずに、ルールを使用するのに十分な情報を提供することを目的としています。このドキュメントに記載されている以上の情報が必要な場合は、AWS Support センター にお問い合わせください。

SQL データベースマネージドルールグループ

VendorName: `AWS`、名前: `AWSManagedRulesSQLiRuleSet`、WCU: 200

SQL Database ルールグループには、SQL インジェクション攻撃などの SQL データベースの悪用に関連するリクエストパターンをブロックするルールが含まれています。これにより、不正なクエリのリモートインジェクションを防ぐことができます。アプリケーションが SQL データベースと連結している場合は、このルールグループを評価します。

このマネージドルールグループは、評価対象のウェブリクエストにラベルを追加します。このラベルは、ウェブ ACL のこのルールグループの後に実行されるルールで使用できます。 AWS WAF は、ラ

ベルを Amazon CloudWatch メトリクスにも記録します。ラベルとラベルメトリクスに関する一般的な情報については、「ウェブリクエストのラベル」および「ラベルメトリクスとディメンション」を参照してください。

> **Note**
>
> この表には、このルールグループの最新の静的バージョンが示されています。他のバージョンの場合は、 API コマンド を使用しますDescribeManagedRuleGroup。

ルール名	説明とラベル
SQLi_QUERYARGUMENTS	組み込みの を使用し AWS WAF SQL インジェクション攻撃ルールステートメント、機密性レベルを に設定してLow、悪意のある SQL コードに一致するパターンがないか、すべてのクエリパラメータの値を検査します。 ルールアクション: Block ラベル: awswaf:managed:aws:sql-database:SQLi_QueryArguments
SQLiExtendedPatterns_QUERYARGUMENTS	すべてのクエリパラメータの値に、悪意のある SQL コードに一致するパターンがないかを検査します。このルールが検査するパターンは、ルール SQLi_QUERYARGUMENTS の対象外です。 ルールアクション: Block ラベル: awswaf:managed:aws:sql-database:SQLiExtendedPatterns_QueryArguments
SQLi_BODY	組み込みの を使用し AWS WAF SQL インジェクション攻撃ルールステートメント、機密性レ

ルール名	説明とラベル
	ベルを に設定してLow、悪意のある SQL コードに一致するパターンがないかリクエストボディを検査します。 **Warning** このルールは、ウェブ ACL とリソースタイプの本文サイズ制限までのリクエスト本文のみを検査します。Application Load Balancer と の場合 AWS AppSync、制限は 8 KB に固定されます。 CloudFront、API Gateway、Amazon Cognito 、App Runner、Verified Access の場合、デフォルトの制限は 16 KB で、ウェブ ACL 設定で制限を最大 64 KB に増やすことができます。このルールは、オーバーサイズコンテンツの処理に `Continue` オプションを使用します。詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。 ルールアクション: Block ラベル: `awswaf:managed:aws:sql-database:SQLi_Body`

ルール名	説明とラベル
SQLiExtendedPatterns_BODY	リクエストボディに、悪意のある SQL コードに一致するパターンがないかを検査します。このルールが検査するパターンは、ルール SQLi_BODY の対象外です。 **Warning** このルールは、ウェブ ACL とリソースタイプの本文サイズ制限までのリクエスト本文のみを検査します。Application Load Balancer と の場合 AWS AppSync、制限は 8 KB に固定されます。 CloudFront、API Gateway、Amazon Cognito 、App Runner、Verified Access の場合、デフォルトの制限は 16 KB で、ウェブ ACL 設定で制限を最大 64 KB に増やすことができます。このルールは、オーバーサイズコンテンツの処理に Continue オプションを使用します。詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。 ルールアクション: Block ラベル: awswaf:managed:aws:sql-database:SQLiExtendedPatterns_Body

ルール名	説明とラベル
SQLi_COOKIE	機密レベルを に設定して組み込みの を使用して AWS WAF SQL インジェクション攻撃ルールステートメント、悪意のある SQL コードに一致するパターンがないかリクエスト Cookie ヘッダーLowを検査します。 ルールアクション: Block ラベル: awswaf:managed:aws:sql-database:SQLi_Cookie

Linux オペレーティングシステムマネージドルールグループ

VendorName: AWS、名前: AWSManagedRulesLinuxRuleSet、WCU: 200

Linux オペレーティングシステムルールグループには、Linux 固有のローカルファイルインクルージョン (LFI) 攻撃など、Linux 固有の脆弱性の悪用に関連するリクエストパターンをブロックするルールが含まれています。これにより、攻撃者がアクセスしてはならないファイルの内容を公開したり、コードを実行したりする攻撃を防ぐことができます。アプリケーションの一部が Linux で実行されている場合は、このルールグループを評価する必要があります。このルールグループは、POSIX オペレーティングシステム ルールグループと組み合わせて使用する必要があります。

このマネージドルールグループは、評価対象のウェブリクエストにラベルを追加します。このラベルは、ウェブ ACL のこのルールグループの後に実行されるルールで使用できます。 AWS WAF は、ラベルを Amazon CloudWatch メトリクスにも記録します。ラベルとラベルメトリクスに関する一般的な情報については、「ウェブリクエストのラベル」および「ラベルメトリクスとディメンション」を参照してください。

> **Note**
>
> この表には、このルールグループの最新の静的バージョンが示されています。他のバージョンの場合は、 API コマンド を使用しますDescribeManagedRuleGroup。

ルール名	説明とラベル
LFI_URIPATH	リクエストパスに、ウェブアプリケーションのローカルファイルインクルージョン (LFI) の脆弱性を悪用する試みがないかを検査します。パターンの例には、攻撃者にオペレーティングシステムの情報を提供できる /proc/version などのファイルがあります。 ルールアクション: Block ラベル: awswaf:managed:aws:linux-os:LFI_URIPath
LFI_QUERYSTRING	クエリ文字列の値に、ウェブアプリケーションのローカルファイルインクルージョン (LFI) の脆弱性を悪用する試みがないかを検査します。パターンの例には、攻撃者にオペレーティングシステムの情報を提供できる /proc/version などのファイルがあります。 ルールアクション: Block ラベル: awswaf:managed:aws:linux-os:LFI_QueryString
LFI_HEADER	リクエストヘッダーに、ウェブアプリケーションのローカルファイルインクルージョン (LFI) の脆弱性を悪用する試みの有無を検査します。パターンの例には、攻撃者にオペレーティングシステムの情報を提供できる /proc/version などのファイルがあります。

ルール名	説明とラベル
	Warning このルールは、リクエストヘッダーの最初の 8 KB または最初の 200 個のヘッダーのうち、いずれかの制限に先に達した方のみを検査し、オーバーサイズコンテンツの処理に Continue オプションを使用します。詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。 ルールアクション: Block ラベル: `awswaf:managed:aws:linux-os:LFI_Header`

POSIX オペレーティングシステムマネージドルールグループ

VendorName: AWS、名前: AWSManagedRulesUnixRuleSet、WCU: 100

POSIX オペレーティングシステムルールグループには、POSIX および POSIX と同等のオペレーティングシステムに固有の脆弱性の悪用 (ローカルファイルインクルージョン (LFI) 攻撃など) に関連するリクエストパターンをブロックするルールが含まれています。これにより、攻撃者がアクセスしてはならないファイルの内容を公開したり、コードを実行したりする攻撃を防ぐことができます。アプリケーションの一部が POSIX または POSIX と同等のオペレーティングシステム (Linux、AIX、HP-UX、macOS、Solaris、FreeBSD、OpenBSD など) で実行されている場合は、このルールグループを評価する必要があります。

このマネージドルールグループは、評価対象のウェブリクエストにラベルを追加します。このラベルは、ウェブ ACL のこのルールグループの後に実行されるルールで使用できます。 AWS WAF は、ラベルを Amazon CloudWatch メトリクスにも記録します。ラベルとラベルメトリクスに関する一般的な情報については、「ウェブリクエストのラベル」および「ラベルメトリクスとディメンション」を参照してください。

> **Note**
>
> この表には、このルールグループの最新の静的バージョンが示されています。他のバージョンの場合は、 API コマンド を使用しますDescribeManagedRuleGroup。

ルール名	説明とラベル
`UNIXShellCommandsVariables_QUERYSTRING`	Unix システムで実行されるウェブアプリケーションのコマンドインジェクション、LFI、パストラバーサルの脆弱性を悪用しようとする試みについて、クエリ文字列の値を検査します。パターンの例には、`echo $HOME` や `echo $PATH` などがあります。 ルールアクション: Block ラベル: `awswaf:managed:aws:posix-os:UNIXShellCommandsVariables_QueryString`
`UNIXShellCommandsVariables_BODY`	リクエストボディに、Unix システムで実行されるウェブアプリケーションのコマンドインジェクション、LFI、パストラバーサルの脆弱性を悪用する試みがないかを検査します。パターンの例には、`echo $HOME` や `echo $PATH` などがあります。 **Warning** このルールは、ウェブ ACL とリソースタイプの本文サイズ制限までのリクエスト本文のみを検査します。Application Load Balancer と の場合 AWS AppSync、制限は 8 KB に固定されます。 CloudFront、API Gateway、A

ルール名	説明とラベル
	mazon Cognito 、App Runner、Verified Access の場合、デフォルトの制限は 16 KB で、ウェブ ACL 設定で制限を最大 64 KB に増やすことができます。このルールは、オーバーサイズコンテンツの処理に Continue オプションを使用します。詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。 ルールアクション: Block ラベル: awswaf:managed:aws:posix-os:UNIXShellCommandsVariables_Body

ルール名	説明とラベル
`UNIXShellCommandsVariables_HEADER`	Unix システムで実行されるウェブアプリケーションのコマンドインジェクション、LFI、パストラバーサルの脆弱性を悪用する試みがないか、すべてのリクエストヘッダーを検査します。パターンの例には、`echo $HOME` や `echo $PATH` などがあります。 **Warning** このルールは、リクエストヘッダーの最初の 8 KB または最初の 200 個のヘッダーのうち、いずれかの制限に先に達した方のみを検査し、オーバーサイズコンテンツの処理に `Continue` オプションを使用します。詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。 ルールアクション: Block ラベル: `awswaf:managed:aws:posix-os:UNIXShellCommandsVariables_Header`

Windows オペレーティングシステムマネージドルールグループ

VendorName: AWS、名前: AWSManagedRulesWindowsRuleSet、WCU: 200

Windows オペレーティングシステムのルールグループには、 PowerShell コマンドのリモート実行など、Windows 固有の脆弱性の悪用に関連するリクエストパターンをブロックするルールが含まれています。これにより、攻撃者が不正なコマンドまたは悪意のあるコードを実行できる脆弱性の悪用を

防ぐことができます。アプリケーションの一部が Windows オペレーティングシステムで実行されている場合は、このルールグループを評価します。

このマネージドルールグループは、評価対象のウェブリクエストにラベルを追加します。このラベルは、ウェブ ACL のこのルールグループの後に実行されるルールで使用できます。 AWS WAF は、ラベルを Amazon CloudWatch メトリクスにも記録します。ラベルとラベルメトリクスに関する一般的な情報については、「ウェブリクエストのラベル」および「ラベルメトリクスとディメンション」を参照してください。

> **Note**
>
> この表には、このルールグループの最新の静的バージョンが示されています。他のバージョンの場合は、 API コマンド を使用しますDescribeManagedRuleGroup。

ルール名	説明とラベル
`WindowsShellCommands_COOKIE`	ウェブアプリケーションでの WindowsShell コマンドインジェクションの試行について、リクエスト Cookie ヘッダーを検査します。一致パターンはWindowsShell コマンドを表します。パターンの例には、`||nslookup` 、`;cmd` などがあります。 ルールアクション: Block ラベル: `awswaf:managed:aws:windows-os:WindowsShellCommands_Cookie`
`WindowsShellCommands_QUERYARGUMENTS`	ウェブアプリケーションでのWindowsShell コマンドインジェクションの試行について、すべてのクエリパラメータの値を検査します。一致パターンはWindowsShell コマンドを表します。パターンの例には、`||nslookup` 、`;cmd` などがあります。 ルールアクション: Block

ルール名	説明とラベル
	ラベル: `awswaf:managed:aws:windows-os:WindowsShellCommands_QueryArguments`

ルール名	説明とラベル
`WindowsShellCommands_BODY`	ウェブアプリケーションでの WindowsShell コマンドインジェクションの試行について、リクエストボディを検査します。一致パターンは WindowsShell コマンドを表します。パターンの例には、`||nslookup` 、`;cmd` などがあります。 **Warning** このルールは、ウェブ ACL とリソースタイプの本文サイズ制限までのリクエスト本文のみを検査します。Application Load Balancer と の場合 AWS AppSync、制限は 8 KB に固定されます。 CloudFront、API Gateway、Amazon Cognito 、App Runner、Verified Access の場合、デフォルトの制限は 16 KB で、ウェブ ACL 設定で制限を最大 64 KB に増やすことができます。このルールは、オーバーサイズコンテンツの処理に `Continue` オプションを使用します。詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。 ルールアクション: Block ラベル: `awswaf:managed:aws:windows-os:WindowsShellCommands_Body`

ルール名	説明とラベル
PowerShellCommands_COOKIE	ウェブアプリケーションでの PowerShell コマンドインジェクションの試行について、リクエスト Cookie ヘッダーを検査します。一致パターンはPowerShell コマンドを表します。例えば Invoke-Expression です。 ルールアクション: Block ラベル: awswaf:managed:aws:windows-os:PowerShellCommands_Cookie
PowerShellCommands_QUERYARGUMENTS	ウェブアプリケーションでのPowerShell コマンドインジェクションの試行について、すべてのクエリパラメータの値を検査します。一致パターンはPowerShell コマンドを表します。例えば Invoke-Expression です。 ルールアクション: Block ラベル: awswaf:managed:aws:windows-os:PowerShellCommands_QueryArguments

ルール名	説明とラベル
PowerShellCommands_BODY	ウェブアプリケーションでの PowerShell コマンドインジェクションの試行について、リクエストボディを検査します。一致パターンは PowerShell コマンドを表します。例えば `Invoke-Expression` です。 **Warning** このルールは、ウェブ ACL とリソースタイプの本文サイズ制限までのリクエスト本文のみを検査します。Application Load Balancer と の場合 AWS AppSync、制限は 8 KB に固定されます。 CloudFront、API Gateway、Amazon Cognito 、App Runner、Verified Access の場合、デフォルトの制限は 16 KB で、ウェブ ACL 設定で制限を最大 64 KB に増やすことができます。このルールは、オーバーサイズコンテンツの処理に `Continue` オプションを使用します。詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。 ルールアクション: Block ラベル: `awswaf:managed:aws:windows-os:PowerShellCommands_Body`

PHP アプリケーションマネージドルールグループ

VendorName: `AWS`、名前: `AWSManagedRulesPHPRuleSet`、WCU: 100

PHP アプリケーションルールグループには、安全でない PHP 関数のインジェクションなど、PHP プログラミング言語の使用に固有の脆弱性の悪用に関連するリクエストパターンをブロックするルールが含まれています。これにより、攻撃者が許可されていないコードまたはコマンドを遠隔で実行できる脆弱性の悪用を防ぐことができます。アプリケーションが連結するサーバーに PHP がインストールされている場合は、このルールグループを評価します。

このマネージドルールグループは、評価対象のウェブリクエストにラベルを追加します。このラベルは、ウェブ ACL のこのルールグループの後に実行されるルールで使用できます。 AWS WAF は、ラベルを Amazon CloudWatch メトリクスにも記録します。ラベルとラベルメトリクスに関する一般的な情報については、「ウェブリクエストのラベル」および「ラベルメトリクスとディメンション」を参照してください。

> **Note**
>
> この表には、このルールグループの最新の静的バージョンが示されています。他のバージョンの場合は、 API コマンド を使用しますDescribeManagedRuleGroup。

ルール名	説明とラベル
PHPHighRiskMethodsVariables_HEADER	PHP スクリプトコードインジェクションの試行について、すべてのヘッダーを検査します。パターンの例には、fsockopen や $_GET スーパーグローバル変数などの関数があります。 **Warning** このルールは、リクエストヘッダーの最初の 8 KB または最初の 200 個のヘッダーのうち、いずれかの制限に先に達した方のみを検査し、オーバーサイズコンテンツの処理に Continue オプションを使用します。詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。

ルール名	説明とラベル
	ルールアクション: Block ラベル: awswaf:managed:aws:php-app:PHPHighRiskMethodsVariables_Header
PHPHighRiskMethodsVariables_QUERYSTRING	リクエスト URL の最初の ? 以降をすべて検査し、PHP スクリプトコードインジェクションの試行がないかを調べます。パターンの例には、fsockopen や $_GET スーパーグローバル変数などの関数があります。 ルールアクション: Block ラベル: awswaf:managed:aws:php-app:PHPHighRiskMethodsVariables_QueryString

ルール名	説明とラベル
PHPHighRiskMethodsVariables_BODY	リクエストボディの値に、PHP スクリプトコードインジェクションがないかを検査します。パターンの例には、fsockopen や $_GET スーパーグローバル変数などの関数があります。 **Warning** このルールは、ウェブ ACL とリソースタイプの本文サイズ制限までのリクエスト本文のみを検査します。Application Load Balancer と の場合 AWS AppSync、制限は 8 KB に固定されます。 CloudFront、API Gateway、Amazon Cognito 、App Runner、Verified Access の場合、デフォルトの制限は 16 KB で、ウェブ ACL 設定で制限を最大 64 KB に増やすことができます。このルールは、オーバーサイズコンテンツの処理に Continue オプションを使用します。詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。 ルールアクション: Block ラベル: awswaf:managed:aws:php-app:PHPHighRiskMethodsVariables_Body

WordPress アプリケーションマネージドルールグループ

VendorName: AWS、名前: AWSManagedRulesWordPressRuleSet、WCU: 100

WordPress アプリケーションルールグループには、WordPress サイト固有の脆弱性の悪用に関連するリクエストパターンをブロックするルールが含まれています。を実行している場合は、このルールグループを評価する必要がありますWordPress。このルールグループは、SQL データベース および PHP アプリケーション ルールグループと組み合わせて使用する必要があります。

このマネージドルールグループは、評価対象のウェブリクエストにラベルを追加します。このラベルは、ウェブ ACL のこのルールグループの後に実行されるルールで使用できます。 AWS WAF は、ラベルを Amazon CloudWatch メトリクスにも記録します。ラベルとラベルメトリクスに関する一般的な情報については、「ウェブリクエストのラベル」および「ラベルメトリクスとディメンション」を参照してください。

> **Note**
>
> この表には、このルールグループの最新の静的バージョンが示されています。他のバージョンの場合は、 API コマンド を使用しますDescribeManagedRuleGroup。

ルール名	説明とラベル
WordPressExploitableCommands_QUERYSTRING	リクエストクエリ文字列に、脆弱なインストールやプラグインで悪用される可能性のある高リスクWordPress コマンドがないかを検査します。パターンの例には、do-reset-wordpress などのコマンドがあります。 ルールアクション: Block ラベル: awswaf:managed:aws:wordpress-app:WordPressExploitableCommands_QUERYSTRING
WordPressExploitablePaths_URIPATH	リクエスト URI パスでxmlrpc.php 、悪用しやすい脆弱性があることがわかっている な

ルール名	説明とラベル
	どの WordPress ファイルがないかを検査します。 ルールアクション: Block ラベル: `awswaf:managed:aws:wordpress-app:WordPressExploitablePaths_URIPATH`

IP 評価ルールグループ

IP 評価ルールグループはソース IP アドレスに基づいてリクエストをブロックします。

> **Note**
>
> これらのルールは、ウェブリクエストの発信元のソース IP アドレスを使用します。トラフィックが 1 つ以上のプロキシまたはロードバランサーを通過する場合、ウェブリクエストの発信元には、クライアントの発信アドレスではなく、最後のプロキシのアドレスが含まれます。

ボットトラフィックや悪用の試みを緩和する場合、またはコンテンツに地理的制限を適用する場合は、これらのルールグループを 1 つ以上選択します。ボット管理については、「AWS WAF Bot Control ルールグループ」も参照してください。

このカテゴリのルールグループは、バージョニングや SNS 更新通知を提供しません。

> **Note**
>
> AWS マネージドルールのルールグループでルールに対して公開する情報は、不正な攻撃者がルールを回避するために使用できる情報を提供せずに、ルールを使用するのに十分な情報を提供することを目的としています。このドキュメントに記載されている以上の情報が必要な場合は、AWS Support センター にお問い合わせください。

Amazon IP 評価リストマネージドルールグループ

VendorName: AWS、名前: AWSManagedRulesAmazonIpReputationList、WCU: 25

Amazon IP 評価リストルールグループには、Amazon 内部脅威インテリジェンスに基づくルールが含まれています。これは、通常、ボットやその他の脅威に関連付けられている IP アドレスをブロックする場合に便利です。これらの IP アドレスをブロックすることで、ボットを緩和し、悪意のあるアクターが脆弱なアプリケーションを発見するリスクを緩和できます。

このマネージドルールグループは、評価対象のウェブリクエストにラベルを追加します。このラベルは、ウェブ ACL のこのルールグループの後に実行されるルールで使用できます。 AWS WAF は、ラベルを Amazon CloudWatch メトリクスにも記録します。ラベルとラベルメトリクスに関する一般的な情報については、「ウェブリクエストのラベル」および「ラベルメトリクスとディメンション」を参照してください。

ルール名	説明とラベル
AWSManagedIPReputationList	悪意のあるアクティビティに積極的に関与していると特定された IP アドレスを検査します。AWS WAF は、Amazon が顧客をサイバー犯罪から保護するために使用する脅威インテリジェンスツールである など MadPot、さまざまなソースから IP アドレスリストを収集します。の詳細については、 MadPot「」を参照してくださいhttps://www.aboutamazon.com/news/aws/amazon-madpot-stops-cybersecurity-crime。 ルールアクション: Block ラベル: awswaf:managed:aws:amazon-ip-list:AWSManagedIPReputationList
AWSManagedReconnaissanceList	AWS リソースに対して偵察を実行している IP アドレスからの接続を検査します。 ルールアクション: Block

ルール名	説明とラベル
	ラベル: awswaf:managed:aws:amazon-ip-list:AWSManagedReconnaissanceList
AWSManagedIPDDoSList	DDoS アクティビティにアクティブに関与していると識別された IP アドレスを検査します。 ルールアクション: Count ラベル: awswaf:managed:aws:amazon-ip-list:AWSManagedIPDDoSList

匿名 IP リストマネージドルールグループ

VendorName: AWS、名前: AWSManagedRulesAnonymousIpList、WCU: 50

匿名 IP リストのルールグループには、ビューワー ID の難読化を許可するサービスからのリクエストをブロックするルールが含まれています。これには、VPN、プロキシ、Tor ノード、ウェブホスティングプロバイダーなどからのリクエストが含まれます。このルールグループは、アプリケーションから ID を隠そうとするビューワーを除外する場合に便利です。これらのサービスの IP アドレスをブロックすると、ボットの緩和や地理的制限の回避に役立ちます。

このマネージドルールグループは、評価対象のウェブリクエストにラベルを追加します。このラベルは、ウェブ ACL のこのルールグループの後に実行されるルールで使用できます。 AWS WAF は、ラベルを Amazon CloudWatch メトリクスにも記録します。ラベルとラベルメトリクスに関する一般的な情報については、「ウェブリクエストのラベル」および「ラベルメトリクスとディメンション」を参照してください。

ルール名	説明とラベル
AnonymousIPList	クライアントの情報を匿名化することがわかっているソース (TOR ノード、一時プロキシ、その他のマスキングサービスなど) の IP アドレスのリストを検査します。 ルールアクション: Block

ルール名	説明とラベル
	ラベル: awswaf:managed:aws:anonymous-ip-list:AnonymousIPList
HostingProviderIPList	エンドユーザートラフィックのソースになる可能性が低いウェブホスティングプロバイダーとクラウドプロバイダーの IP アドレスのリストを検査します。IP リストには AWS IP アドレスは含まれません。 ルールアクション: Block ラベル: awswaf:managed:aws:anonymous-ip-list:HostingProviderIPList

AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) ルールグループ

VendorName: AWS、名前: AWSManagedRulesACFPRuleSet、WCU: 50

AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) マネージドルールグループは、不正なアカウント作成の試みの一部である可能性のあるリクエストにラベルを付けて管理します。ルールグループは、クライアントがアプリケーションの登録エンドポイントとアカウント作成エンドポイントに送信するアカウント作成リクエストを検査することでこれを行います。

ACFP ルールグループは、さまざまな方法でアカウント作成の試みを検査し、悪意のある可能性のあるインタラクションを可視化し、コントロールできるようにします。ルールグループは、リクエストトークンを使用して、クライアントブラウザに関する情報と、アカウント作成リクエストの作成における人間のインタラクティビティのレベルに関する情報を収集します。ルールグループは、IP アドレスとクライアントセッションごとにリクエストを集計し、物理アドレスや電話番号などの提供されたアカウント情報ごとに集計することで、一括アカウント作成の試みを検出および管理します。さらに、ルールグループは、侵害された認証情報を使用した新しいアカウントの作成を検出してブロックします。これは、アプリケーションと新しいユーザーのセキュリティ体制の保護に役立ちます。

このルールグループを使用する際の考慮事項

このルールグループには、アプリケーションのアカウント登録パスとアカウント作成パスの仕様を含むカスタム設定が必要です。特に明記されている場合を除き、このルールグループのルールは、クライアントがこれらの 2 つのエンドポイントに送信するすべてのリクエストを検査します。このルー

ルグループを設定および実装するには、「AWS WAF 不正防止アカウント作成詐欺防止 (ACFP)」のガイダンスを参照してください。

> **Note**
>
> このマネージドルールグループを使用する場合、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

このルールグループは、 AWS WAFでのインテリジェントな脅威の軽減保護の一部です。詳細については、「AWS WAF インテリジェントな脅威軽減」を参照してください。

コストを抑え、ウェブトラフィックを希望どおりに管理していることを確実にするには、インテリジェントな脅威の軽減のためのベストプラクティス のガイダンスに従ってこのルールグループを使用してください。

このルールグループは、Amazon Cognito ユーザープールでは使用できません。このルールグループを使用するウェブ ACL をユーザープールに関連付けることはできません。また、このルールグループをユーザープールに既に関連付けられたウェブ ACL に追加することはできません。

このルールグループによって追加されるラベル

このマネージドルールグループは、評価対象のウェブリクエストにラベルを追加します。このラベルは、ウェブ ACL のこのルールグループの後に実行されるルールで使用できます。 AWS WAF は、ラベルを Amazon CloudWatch メトリクスにも記録します。ラベルとラベルメトリクスに関する一般的な情報については、「ウェブリクエストのラベル」および「ラベルメトリクスとディメンション」を参照してください。

トークンラベル

このルールグループは、 AWS WAF トークン管理を使用して、 AWS WAF トークンのステータスに従ってウェブリクエストを検査し、ラベル付けします。 は、クライアントセッションの追跡と検証にトークン AWS WAF を使用します。

トークンおよびトークンの管理の詳細については、「AWS WAF ウェブリクエストトークン」を参照してください。

ここで説明するラベルコンポーネントについては、「AWS WAF ラベル構文と命名要件」を参照してください。

クライアントセッションラベル

ラベルには、 AWS WAF トークン管理がクライアントセッションを識別するために使用する一意の識別子`awswaf:managed:token:id:`*`identifier`*が含まれています。この識別子は、クライアントが使用していたトークンを破棄した後など、新しいトークンを取得すると変わる可能性があります。

> **Note**
>
> AWS WAF は、このラベルの Amazon CloudWatch メトリクスを報告しません。

トークンステータスラベル: ラベル名前空間プレフィックス

トークンステータスラベルは、トークン、チャレンジのステータス、およびそれに含まれる CAPTCHA 情報を報告します。

各トークンステータスラベルは、次のプレフィクスの 1 つで始まります。

- `awswaf:managed:token:`— トークンの一般的なステータスを報告したり、トークンのチャレンジ情報のステータスを報告したりするために使用されます。
- `awswaf:managed:captcha:`— トークンの CAPTCHA 情報のステータスを報告するために使用されます。

トークンステータスラベル: ラベル名

プレフィックスに続いて、ラベルの残りの部分には詳細なトークンステータス情報が表示されます。

- `accepted` - リクエストトークンが存在し、以下の内容が含まれています。
  - 有効なチャレンジまたは CAPTCHA ソリューション。
  - 有効期限が切れていないチャレンジまたは CAPTCHA タイムスタンプ。
  - ウェブ ACL に有効なドメイン仕様。

  例: ラベル `awswaf:managed:token:accepted` には、ウェブリクエストのトークンに有効なチャレンジソリューション、有効期限が切れていないチャレンジタイムスタンプ、および有効なドメインがあることが示されています。
- `rejected` - リクエストトークンは存在するが、承認基準を満たしていない。

  トークン管理では、拒否されたラベルに加えて、理由を示すカスタムラベル名前空間と名前が追加されます。

- `rejected:not_solved` — トークンにチャレンジまたは CAPTCHA ソリューションがない。
- `rejected:expired` — ウェブ ACL に設定されているトークンイミュニティ時間によると、トークンのチャレンジまたは CAPTCHA タイムスタンプの有効期限が切れている。
- `rejected:domain_mismatch` — トークンのドメインが、ウェブ ACL のトークンドメイン設定と一致しない。
- `rejected:invalid` - 指定されたトークンを読み AWS WAF 取れませんでした。

  例: ラベル `awswaf:managed:captcha:rejected` と `awswaf:managed:captcha:rejected:expired` には、トークンの CAPTCHA タイムスタンプがウェブ ACL で設定されている CAPTCHA トークンのイミュニティ時間を超えたためにリクエストが拒否されたことが示されています。

- `absent` — リクエストにトークンがないか、トークンマネージャーがそれを読み取れなかった。

  例: ラベル `awswaf:managed:captcha:absent` には、リクエストにトークンがないことが示されています。

ACFP ラベル

このルールグループは、名前空間プレフィックス `awswaf:managed:aws:acfp:` が付いたラベルを生成し、その後にカスタム名前空間およびラベル名が付いたラベルを生成します。ルールグループは、リクエストに複数のラベルを追加する場合があります。

`DescribeManagedRuleGroup` を呼び出すことにより、API を介してルールグループのすべてのラベルを取得できます。ラベルは、応答の `AvailableLabels` プロパティにリストされています。

Account Creation Fraud Prevention ルールリスト

次のセクションには、`AWSManagedRulesACFPRuleSet` の ACFP ルールとルールグループがウェブリクエストに追加するラベルが示されています。

> **Note**
>
> AWS マネージドルールのルールグループでルールに対して公開する情報は、不正な攻撃者がルールを回避するために使用できる情報を提供せずに、ルールを使用するのに十分な情報を提供することを目的としています。このドキュメントに記載されている以上の情報が必要な場合は、AWS Support センター にお問い合わせください。

このルールグループ内のすべてのルールでは、最初の 2 つの `UnsupportedCognitoIDP` と `AllRequests` を除き、ウェブリクエストトークンが必要です。トークンが提供する情報の説明については、「AWS WAF トークンの特性」を参照してください。

特に明記されていない限り、このルールグループのルールは、ルールグループの設定で指定したアカウント登録ページのパスとアカウント作成ページのパスにクライアントが送信するすべてのリクエストを検査します。このルールグループの設定の詳細については、「AWS WAF 不正防止アカウント作成詐欺防止 (ACFP)」を参照してください。

ルール名	説明とラベル
`UnsupportedCognitoIDP`	Amazon Cognito ユーザープールに向かうウェブトラフィックの有無を検査します。ACFP は Amazon Cognito ユーザープールでは使用できません。このルールは、他の ACFP ルールグループのルールがユーザープールのトラフィックの評価に使用されないようにすることに役立ちます。 ルールアクション: Block ラベル: `awswaf:managed:aws:acfp:unsupported:cognito_idp`
`AllRequests`	登録ページのパスにアクセスするリクエストにルールアクションを適用します。ルールグループを設定するときに、登録ページのパスを設定します。 デフォルトでは、このルールは Challenge をリクエストに適用します。このアクションを適用することにより、ルールは、ルールグループ内の残りのルールによってリクエストが評価される前に、クライアントがチャレンジトークンを取得するようにします。

ルール名	説明とラベル
	エンドユーザーがアカウント作成リクエストを送信する前に、登録ページのパスをロードするようにします。 トークンは、クライアントアプリケーション統合 SDK、ならびに CAPTCHA および Challenge のルールアクションによってリクエストに追加されます。最も効率的にトークンを取得するために、アプリケーション統合 SDK を使用することを強くお勧めします。詳細については、「AWS WAF クライアントアプリケーション統合」を参照してください。 ルールアクション: Challenge ラベル: なし

ルール名	説明とラベル
RiskScoreHigh	IP アドレスやその他の非常に疑わしい要素が含まれるアカウント作成リクエストを検査します。この評価は通常、複数の寄与要因に基づいており、ルールグループがリクエストに追加する risk_score ラベルで確認できます。 ルールアクション: Block ラベル: awswaf:managed:aws:acfp:risk_score:high このルールは、medium または low のリスクスコアラベルをリクエストに適用することもできます。 AWS WAF がウェブリクエストのリスクスコアの評価に成功しない場合、ルールは ラベルを追加します。 awswaf:managed:aws:acfp:risk_score:evaluation_failed さらに、このルールは、IP レピュテーションや盗難された認証情報の評価など、特定のリスクスコアの寄与要因のリスクスコアの評価ステータスと結果を含む名前空間 awswaf:managed:aws:acfp:risk_score:contributor: のラベルを追加します。

ルール名	説明とラベル
`SignalCredentialCompromised`	盗まれた認証情報データベースで、アカウント作成リクエストで送信された認証情報を検索します。 このルールにより、新しいクライアントは、好ましいセキュリティ体制でアカウントを初期化するようになります。 **Note** カスタムブロックレスポンスを追加して、エンドユーザーに問題を説明し、続行方法を伝えることができます。詳細については、「ACFP の例: 侵害された認証情報についてのカスタムレスポンス」を参照してください。 ルールアクション: Block ラベル: `awswaf:managed:aws:acfp:signal:credential_compromised` ルールグループは次の関連ラベルを適用しますが、アカウント作成のすべてのリクエストに認証情報があるわけではないため、それに対するアクションは実行されません: `awswaf:managed:aws:acfp:signal:missing_credential` 。

ルール名	説明とラベル
`SignalClientHumanInteractivityAbsentLow`	アカウント作成リクエストのトークンで、人間によるアプリケーションとの異常なインタラクティブ性を示すデータがないかどうかを検査します。人間のインタラクティビティは、マウスの動きやキーの押下などのインタラクションを通じて検出されます。ページに HTML フォームがある場合、人間によるインタラクションにはフォームとのインタラクションが含まれます。 **Note** このルールは、アカウント作成パスに対するリクエストのみを検査し、アプリケーション統合 SDK を実装している場合にのみ評価されます。SDK を実装することで、人間のインタラクティブ性を受動的にキャプチャし、その情報をリクエストトークンに保存できます。詳細については、「AWS WAF トークンの特性」および「AWS WAF クライアントアプリケーション統合」を参照してください。 ルールアクション: CAPTCHA ラベル: なし このルールはさまざまな要因に基づいて一致を決定するため、考えられるすべての一致シナリオに適用される個別のラベルはありません。 ルールグループは、次の 1 つ以上のラベルをリクエストに適用できます。

ルール名	説明とラベル
	awswaf:managed:aws:acfp:signal:client:human_interactivity:low/medium/high awswaf:managed:aws:acfp:signal:client:human_interactivity:insufficient_data awswaf:managed:aws:acfp:signal:form_detected .
SignalAutomatedBrowser	クライアントブラウザが自動化されている可能性があることを示す要素がないか、リクエストを検査します。 ルールアクション: Block ラベル: awswaf:managed:aws:acfp:signal:automated_browser
SignalBrowserInconsistency	ブラウザ調査のデータに一貫性がないかどうかを確認するために、リクエストのトークンを検査します。詳細については、「AWS WAF トークンの特性」を参照してください。 ルールアクション: CAPTCHA ラベル: awswaf:managed:aws:acfp:signal:browser_inconsistency

ルール名	説明とラベル
VolumetricIpHigh	個々の IP アドレスから送信された大量のアカウント作成リクエストを検査します。大量とは、10 分ウィンドウにリクエストが 20 件を超えることです。 **Note** このルールが適用するしきい値は、レイテンシーによって若干異なる場合があります。大量の場合、ルールアクションが適用される前に、いくつかのリクエストが制限を超える可能性があります。 ルールアクション: CAPTCHA ラベル: awswaf:managed:aws:acfp:aggregate:volumetric:ip:creation:high このルールは、中程度のボリューム (10 分ウィンドウあたり 15 件を超えるリクエスト) と少量 (10 分ウィンドウあたり 10 件を超えるリクエスト) のリクエストに次のラベルを適用しますが、 awswaf:managed:aws:acfp:aggregate:volumetric:ip:creation:medium および のアクションは実行しませんawswaf:managed:aws:acfp:aggregate:volumetric:ip:creation:low 。

ルール名	説明とラベル
`VolumetricSessionHigh`	個々のクライアントセッションから送信された大量のアカウントリクエストを検査します。大量とは、30 分間の時間枠にリクエストが 10 件を超えることです。 **Note** このルールが適用するしきい値は、レイテンシーによって若干異なる場合があります。ルールアクションが適用される前に、いくつかのリクエストが制限を超えることがあります。 ルールアクション: Block ラベル: `awswaf:managed:aws:acfp:aggregate:volumetric:session:creation:high` ルールグループは、中程度のボリューム (30 分ウィンドウあたり 5 件を超えるリクエスト) と少量 (30 分ウィンドウあたり 1 件を超えるリクエスト) のリクエストに次のラベルを適用しますが、 `awswaf:managed:aws:acfp:aggregate:volumetric:session:creation:medium` および のアクションは実行しません`awswaf:managed:aws:acfp:aggregate:volumetric:session:creation:low` 。

ルール名	説明とラベル
AttributeUsernameTraversalHigh	異なるユーザー名を使用する単一のクライアントセッションからアカウント作成リクエストが高頻度で発生していないかどうかを検査します。30 分間のリクエスト数が 10 件を超えている場合は、大量であると評価されます。 **Note** このルールが適用するしきい値は、レイテンシーによって若干異なる場合があります。ルールアクションが適用される前に、いくつかのリクエストが制限を超えることがあります。 ルールアクション: Block ラベル: awswaf:managed:aws:acfp:aggregate:attribute:username_traversal:creation:high ルールグループは、中程度のボリューム (30 分ウィンドウあたり 5 件を超えるリクエスト) および少量 (30 分ウィンドウあたり 1 件を超えるリクエスト) のユーザー名トラバーサルリクエストを含むリクエストに次のラベルを適用しますが、 awswaf:managed:aws:acfp:aggregate:attribute:username_traversal:creation:medium および のアクションは実行しませんawswaf:managed:aws:acfp:aggregate:attribute:username_traversal:creation:low 。

ルール名	説明とラベル
VolumetricPhoneNumberHigh	同じ電話番号を使用する大量のアカウント作成リクエストが発生していないかを検査します。30 分間のリクエスト数が 10 件を超えている場合は、大量であると評価されます。 **Note** このルールが適用するしきい値は、レイテンシーによって若干異なる場合があります。ルールアクションが適用される前に、いくつかのリクエストが制限を超えることがあります。 ルールアクション: Block ラベル: `awswaf:managed:aws:acfp:aggregate:volumetric:phone_number:high` ルールグループは、中程度のボリューム (30 分ウィンドウあたり 5 件を超えるリクエスト) と少量 (30 分ウィンドウあたり 1 件を超えるリクエスト) のリクエストに次のラベルを適用しますが、 `awswaf:managed:aws:acfp:aggregate:volumetric:phone_number:medium` および のアクションは実行しませんawswaf:managed:aws:acfp:aggregate:volumetric:phone_number:low 。

ルール名	説明とラベル
`VolumetricAddressHigh`	同じ物理的な住所を使用する大量のアカウント作成リクエストが発生していないかを検査します。30 分間の時間枠あたりのリクエスト数が 100 件を超えている場合は、大量であると評価されます。 **Note** このルールが適用するしきい値は、レイテンシーによって若干異なる場合があります。ルールアクションが適用される前に、いくつかのリクエストが制限を超えることがあります。 ルールアクション: Block ラベル: `awswaf:managed:aws:acfp:aggregate:volumetric:address:high`

ルール名	説明とラベル
`VolumetricAddressLow`	同じ物理的な住所を使用する少量または中程度の量のアカウント作成リクエストが発生していないかを検査します。中程度の評価のしきい値は 30 分ウィンドウあたり 50 リクエストを超え、低評価の場合、30 分ウィンドウあたり 10 リクエストを超えます。 このルールは、中程度の量または少量のいずれかの場合にアクションを適用します。 **Note** このルールが適用するしきい値は、レイテンシーによって若干異なる場合があります。ルールアクションが適用される前に、いくつかのリクエストが制限を超えることがあります。 ルールアクション: CAPTCHA ラベル: `awswaf:managed:aws:acfp:aggregate:volumetric:address:low` または `awswaf:managed:aws:acfp:aggregate:volumetric:address:medium`

ルール名	説明とラベル
`VolumetricIPSuccessfulResponse`	単一の IP アドレスに対する大量の正常なアカウント作成リクエストを検査します。このルールは、保護されたリソースからのアカウント作成リクエストに対する成功レスポンスを集計します。10 分間の時間枠あたりのリクエスト数が 10 件を超えている場合は、大量であると評価されます。 このルールは、アカウントの一括作成の試みからの保護に役立ちます。リクエストのみをカウントするルール `VolumetricIpHigh` よりもしきい値が低くなります。 レスポンス本文または JSON コンポーネントを検査するようにルールグループを設定している場合、 はこれらのコンポーネントタイプの最初の 65,536 バイト (64 KB) で成功または失敗のインジケータを検査 AWS WAF できます。 このルールは、同じ IP アドレスからの最新のログイン試行に対する保護されたリソースからの成功応答と失敗応答に基づいて、IP アドレスからの新しいウェブリクエストにルールアクションとラベリングを適用します。ルールグループを設定するときに、成功数と失敗数のカウント方法を定義します。 **Note** AWS WAF は、Amazon CloudFront ディストリビューションを保護するウェブ ACLs でのみこのルールを評価します。

ルール名	説明とラベル
	Note このルールが適用するしきい値は、レイテンシーによって若干異なる場合があります。ルールがその後の試みに対して一致処理を開始する前に、正常なアカウント作成の試みが、許可されているよりも多くクライアントから送信される可能性があります。 ルールアクション: Block ラベル: `awswaf:managed:aws:acfp:aggregate:volumetric:ip:successful_creation_response:high` ルールグループは、次の関連ラベルもリクエストに適用します。関連するアクションはありません。すべてのカウントは 10 分間の時間枠のものです。5 件以上の成功したリクエストには `awswaf:managed:aws:acfp:aggregate:volumetric:ip:successful_creation_response:medium` 、1 件以上の成功したリクエストには `awswaf:managed:aws:acfp:aggregate:volumetric:ip:successful_creation_response:low` 、10 件以上の失敗したリクエストには `awswaf:managed:aws:acfp:aggregate:volumetric:ip:failed_creation_response:high` 、5 件以上の失敗したリクエストには `awswaf:managed:aws:acfp:aggregate:volumetric:`

ルール名	説明とラベル
	ip:failed_creation_response :medium 、1 件以上の失敗したリクエストには awswaf:managed:aws:acfp:aggregate:volumetric:ip:failed_creation_response:low です。

ルール名	説明とラベル
`VolumetricSessionSuccessful Response`	単一のクライアントセッションから送信されるアカウント作成リクエストに対する、保護されたリソースからの少量の成功したレスポンスを検査します。これは、アカウントの一括作成の試みからの保護に役立ちます。30 分間の時間枠あたりのリクエスト数が 1 件を超えている場合は、少量であると評価されます。 これは、アカウントの一括作成の試みからの保護に役立ちます。このルールは、リクエストのみをカウントするルール `VolumetricSessionHigh` よりも低いしきい値を使用します。 レスポンス本文または JSON コンポーネントを検査するようにルールグループを設定している場合、 はこれらのコンポーネントタイプの最初の 65,536 バイト (64 KB) で成功または失敗のインジケータを検査 AWS WAF できます。 このルールは、同じクライアントセッションからの最新のログイン試行に対する保護されたリソースからの成功応答と失敗応答に基づいて、クライアントセッションからの新しいウェブリクエストにルールアクションとラベリングを適用します。ルールグループを設定するときに、成功数と失敗数のカウント方法を定義します。 **Note** AWS WAF は、Amazon CloudFront ディストリビューションを保護する

ルール名	説明とラベル
	ウェブ ACLs でのみこのルールを評価します。 **Note** このルールが適用するしきい値は、レイテンシーによって若干異なる場合があります。ルールがその後の試みに対して一致処理を開始する前に、失敗したアカウント作成の試みが、許可されているよりも多くクライアントから送信される可能性があります。 ルールアクション: Block ラベル: `awswaf:managed:aws:acfp:aggregate:volumetric:session:successful_creation_response:low` ルールグループは、次の関連ラベルもリクエストに適用します。すべてのカウントは 30 分間の時間枠のものです。10 件以上の成功したリクエストには `awswaf:managed:aws:acfp:aggregate:volumetric:session:successful_creation_response:high` 、5 件以上の成功したリクエストには `awswaf:managed:aws:acfp:aggregate:volumetric:session:successful_creation_response:medium` 、10 件以上の失敗したリクエストには `awswaf:managed:aws:acfp:aggregate:volumetric:`

ルール名	説明とラベル
	session:failed_creation_response:high 、5 件以上の失敗したリクエストには awswaf:managed:aws:acfp:aggregate:volumetric:session:failed_creation_response:medium 、1 件以上の失敗したリクエストには awswaf:managed:aws:acfp:aggregate:volumetric:session:failed_creation_response:low です。
VolumetricSessionTokenReuseIp	アカウント作成リクエストを検査して、5 つを超える異なる IP アドレス間で単一のトークンが使用されていないかどうかをチェックします。 **Note** このルールが適用するしきい値は、レイテンシーによって若干異なる場合があります。ルールアクションが適用される前に、いくつかのリクエストが制限を超えることがあります。 ルールアクション: Block ラベル: awswaf:managed:aws:acfp:aggregate:volumetric:session:creation:token_reuse:ip

AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループ

VendorName: AWS、名前: AWSManagedRulesATPRuleSet、WCU: 50

AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) マネージドルールグループは、悪意のあるアカウント乗っ取りの試みの一部である可能性のあるリクエストにラベルを付けて管理します。ルールグループは、クライアントでアプリケーションのログインエンドポイントに送信するログイン試行を検査することでこれを行います。

- リクエスト検査 – ATP を使用すると、異常なログイン試行や盗まれた認証情報を使用するログイン試行を可視化して制御できるため、不正行為につながる可能性のあるアカウントの乗っ取りを防ぐことができます。ATP は、盗まれた認証情報のデータベースに照らして E メールとパスワードの組み合わせをチェックします。このデータベースは、漏洩された認証情報がダークウェブ上で新しく見つかると定期的に更新されます。ATP は、IP アドレスやクライアントセッションごとにデータを集約し、不審なリクエストを大量に送信するクライアントを検出してブロックします。
- レスポンス検査 — CloudFront ディストリビューションの場合、ATP ルールグループは、受信ログインリクエストの検査に加えて、ログイン試行に対するアプリケーションのレスポンスを検査し、成功率と失敗率を追跡します。この情報を使用して、ATP はログイン失敗の回数が過度に多いクライアントセッションまたは IP アドレスを一時的にブロックできます。 AWS WAF は、レスポンス検査を非同期で実行するため、ウェブトラフィックのレイテンシーが大きくなることはありません。

このルールグループを使用する際の考慮事項

このルールグループには特定の設定が必要です。このルールグループを設定および実装するには、「AWS WAF 不正防止アカウント乗っ取り防止 (ATP)」のガイダンスを参照してください。

このルールグループは、 AWS WAFでのインテリジェントな脅威の軽減保護の一部です。詳細については、「AWS WAF インテリジェントな脅威軽減」を参照してください。

> **Note**
>
> このマネージドルールグループを使用する場合、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

コストを抑え、ウェブトラフィックを希望どおりに管理していることを確実にするには、インテリジェントな脅威の軽減のためのベストプラクティス のガイダンスに従ってこのルールグループを使用してください。

このルールグループは、Amazon Cognito ユーザープールでは使用できません。このルールグループを使用するウェブ ACL をユーザープールに関連付けることはできません。また、このルールグループをユーザープールに既に関連付けられたウェブ ACL に追加することはできません。

このルールグループによって追加されるラベル

このマネージドルールグループは、評価対象のウェブリクエストにラベルを追加します。このラベルは、ウェブ ACL のこのルールグループの後に実行されるルールで使用できます。 AWS WAF は、ラベルを Amazon CloudWatch メトリクスにも記録します。ラベルとラベルメトリクスに関する一般的な情報については、「ウェブリクエストのラベル」および「ラベルメトリクスとディメンション」を参照してください。

トークンラベル

このルールグループは、 AWS WAF トークン管理を使用して、 AWS WAF トークンのステータスに従ってウェブリクエストを検査およびラベル付けします。 は、クライアントセッションの追跡と検証にトークン AWS WAF を使用します。

トークンおよびトークンの管理の詳細については、「AWS WAF ウェブリクエストトークン」を参照してください。

ここで説明するラベルコンポーネントについては、「AWS WAF ラベル構文と命名要件」を参照してください。

クライアントセッションラベル

ラベルには、 AWS WAF トークン管理がクライアントセッションを識別するために使用する一意の識別子`awswaf:managed:token:id:`*`identifier`*が含まれています。この識別子は、クライアントが使用していたトークンを破棄した後など、新しいトークンを取得すると変わる可能性があります。

> **Note**
>
> AWS WAF は、このラベルの Amazon CloudWatch メトリクスを報告しません。

トークンステータスラベル: ラベル名前空間プレフィックス

トークンステータスラベルは、トークン、チャレンジのステータス、およびそれに含まれる CAPTCHA 情報を報告します。

各トークンステータスラベルは、次のプレフィクスの 1 つで始まります。

- awswaf:managed:token:— トークンの一般的なステータスを報告したり、トークンのチャレンジ情報のステータスを報告したりするために使用されます。
- awswaf:managed:captcha:— トークンの CAPTCHA 情報のステータスを報告するために使用されます。

トークンステータスラベル: ラベル名

プレフィックスに続いて、ラベルの残りの部分には詳細なトークンステータス情報が表示されます。

- accepted - リクエストトークンが存在し、以下の内容が含まれています。
  - 有効なチャレンジまたは CAPTCHA ソリューション。
  - 有効期限が切れていないチャレンジまたは CAPTCHA タイムスタンプ。
  - ウェブ ACL に有効なドメイン仕様。

  例: ラベル awswaf:managed:token:accepted には、ウェブリクエストのトークンに有効なチャレンジソリューション、有効期限が切れていないチャレンジタイムスタンプ、および有効なドメインがあることが示されています。
- rejected - リクエストトークンは存在するが、承認基準を満たしていない。

  トークン管理では、拒否されたラベルに加えて、理由を示すカスタムラベル名前空間と名前が追加されます。
  - rejected:not_solved — トークンにチャレンジまたは CAPTCHA ソリューションがない。
  - rejected:expired — ウェブ ACL に設定されているトークンイミュニティ時間によると、トークンのチャレンジまたは CAPTCHA タイムスタンプの有効期限が切れている。
  - rejected:domain_mismatch — トークンのドメインが、ウェブ ACL のトークンドメイン設定と一致しない。
  - rejected:invalid – 指定されたトークンを読み AWS WAF 取れませんでした。

  例: ラベル awswaf:managed:captcha:rejected と awswaf:managed:captcha:rejected:expired には、トークンの CAPTCHA タイムスタンプがウェブ ACL で設定されている CAPTCHA トークンのイミュニティ時間を超えたためにリクエストが拒否されたことが示されています。
- absent — リクエストにトークンがないか、トークンマネージャーがそれを読み取れなかった。

  例: ラベル awswaf:managed:captcha:absent には、リクエストにトークンがないことが示されています。

ATP ラベル

ATP マネージドルールグループは、名前空間プレフィックス `awswaf:managed:aws:atp:` が付いたラベルを生成し、その後にカスタム名前空間およびラベル名が付いたラベルを生成します。

ルールグループは、ルールリストに記載されているラベルに加えて、次のラベルのいずれかを追加する場合があります。

- `awswaf:managed:aws:atp:signal:credential_compromised` – リクエストで送信された認証情報が、盗まれた認証情報データベースに含まれていることを示します。
- `awswaf:managed:aws:atp:aggregate:attribute:suspicious_tls_fingerprint` – 保護された Amazon CloudFront ディストリビューションでのみ使用できます。クライアントセッションが、疑わしい TLS フィンガープリントを使用した複数のリクエストを送信したことを示します。
- `awswaf:managed:aws:atp:aggregate:volumetric:session:token_reuse:ip` - 5 つを超える異なる IP アドレス間で単一のトークンが使用されていることを示します。このルールが適用するしきい値は、レイテンシーによって若干異なる場合があります。ラベルが適用される前に、いくつかのリクエストが制限を超えることがあります。

`DescribeManagedRuleGroup` を呼び出すことにより、API を介してルールグループのすべてのラベルを取得できます。ラベルは、応答の `AvailableLabels` プロパティにリストされています。

アカウント乗っ取り防止のルールリスト

次のセクションには、`AWSManagedRulesATPRuleSet` の ATP ルールとルールグループがウェブリクエストに追加するラベルが示されています。

> **Note**
>
> AWS マネージドルールのルールグループでルールに対して公開する情報は、不正な攻撃者がルールを回避するために使用できる情報を提供せずに、ルールを使用するのに十分な情報を提供することを目的としています。このドキュメントに記載されている以上の情報が必要な場合は、AWS Support センター にお問い合わせください。

ルール名	説明とラベル
UnsupportedCognitoIDP	Amazon Cognito ユーザープールに向かうウェブトラフィックの有無を検査します。ATP は Amazon Cognito ユーザープールでは使用できません。このルールは、他の ATP ルールグループのルールがユーザープールのトラフィックの評価に使用されないようにすることに役立ちます。 ルールアクション: Block ラベル: awswaf:managed:aws:atp:unsupported:cognito_idp
VolumetricIpHigh	個々の IP アドレスから送信された大量のリクエストを検査します。大量とは、10 分ウィンドウにリクエストが 20 件を超えることです。 **Note** このルールが適用するしきい値は、レイテンシーによって若干異なる場合があります。大量の場合、ルールアクションが適用される前に、いくつかのリクエストが制限を超える可能性があります。 ルールアクション: Block ラベル: awswaf:managed:aws:atp:aggregate:volumetric:ip:high ルールグループは、中程度のボリューム (10 分ウィンドウあたり 15 件を超えるリクエスト) と少量 (10 分ウィンドウあたり 10 件を超

ルール名	説明とラベル
	えるリクエスト) のリクエストに次のラベルを適用しますが、 `awswaf:managed:aws:atp:aggregate:volumetric:ip:medium` および のアクションは実行しませんawswaf:managed:aws:atp:aggregate:volumetric:ip:low 。
`VolumetricSession`	個々のクライアントセッションから送信された大量のリクエストを検査します。しきい値は、30 分ウィンドウあたり 20 を超えるリクエスト数に設定します。 この検査は、ウェブリクエストにトークンがある場合にのみ適用されます。トークンは、アプリケーション統合 SDK、ならびに CAPTCHA および Challenge のルールアクションによってリクエストに追加されます。詳細については、「AWS WAF ウェブリクエストトークン」を参照してください。 **Note** このルールが適用するしきい値は、レイテンシーによって若干異なる場合があります。ルールアクションが適用される前に、いくつかのリクエストが制限を超えることがあります。 ルールアクション: Block ラベル: `awswaf:managed:aws:atp:aggregate:volumetric:session`

ルール名	説明とラベル
`AttributeCompromisedCredentials`	盗まれた認証情報を使用する同じクライアントセッションからの複数リクエストを検査します。 ルールアクション: Block ラベル: `awswaf:managed:aws:atp:aggregate:attribute:compromised_credentials`
`AttributeUsernameTraversal`	ユーザー名トラバーサルを使用する同じクライアントセッションからの複数リクエストを検査します。 ルールアクション: Block ラベル: `awswaf:managed:aws:atp:aggregate:attribute:username_traversal`
`AttributePasswordTraversal`	パスワードトラバーサルを使用する同じユーザー名の複数のリクエストを検査します。 ルールアクション: Block ラベル: `awswaf:managed:aws:atp:aggregate:attribute:password_traversal`

ルール名	説明とラベル
`AttributeLongSession`	長期に継続するセッションを使用する同じクライアントセッションからの複数リクエストを検査します。しきい値は、30 分ごとに少なくとも 1 つのログインリクエストがある 6 時間を超えるトラフィックです。 この検査は、ウェブリクエストにトークンがある場合にのみ適用されます。トークンは、アプリケーション統合 SDK、ならびに CAPTCHA および Challenge のルールアクションによってリクエストに追加されます。詳細については、「AWS WAF ウェブリクエストトークン」を参照してください。 ルールアクション: Block ラベル: `awswaf:managed:aws:atp:aggregate:attribute:long_session`

ルール名	説明とラベル
`TokenRejected`	トークン管理によって拒否された AWS WAF トークンを含むリクエストを検査します。 この検査は、ウェブリクエストにトークンがある場合にのみ適用されます。トークンは、アプリケーション統合 SDK、ならびに CAPTCHA および Challenge のルールアクションによってリクエストに追加されます。詳細については、「AWS WAF ウェブリクエストトークン」を参照してください。 ルールアクション: Block ラベル: なし トークンが拒否されたかどうか確認するには、ラベル一致ルールを使用してラベルを照合します。`awswaf:managed:token:rejected`
`SignalMissingCredential`	認証情報を含むリクエストに、ユーザー名またはパスワードが不足しているかどうか検査します。 ルールアクション: Block ラベル: `awswaf:managed:aws:atp:signal:missing_credential`

ルール名	説明とラベル
`VolumetricIpFailedLoginResponseHigh`	最近のログイン試行の失敗率が過度に高い IP アドレスを検査します。大量とは、10 分間の時間枠に 1 つの IP アドレスからの失敗したログインリクエストが 10 件を超えることです。 レスポンス本文または JSON コンポーネントを検査するようにルールグループを設定している場合、 はこれらのコンポーネントタイプの最初の 65,536 バイト (64 KB) で成功または失敗のインジケータを検査 AWS WAF できます。 このルールは、同じ IP アドレスからの最新のログイン試行に対する保護されたリソースからの成功応答と失敗応答に基づいて、IP アドレスからの新しいウェブリクエストにルールアクションとラベリングを適用します。ルールグループを設定するときに、成功数と失敗数のカウント方法を定義します。 **Note** AWS WAF は、Amazon CloudFront ディストリビューションを保護するウェブ ACLs でのみこのルールを評価します。 **Note** このルールが適用するしきい値は、レイテンシーによって若干異なる場合があります。ルールがその後のログイン試行に対して一致処理を開始する前

ルール名	説明とラベル
	に、許可されているよりも多い回数の失敗したログイン試行がクライアントから送信される可能性があります。 ルールアクション: Block ラベル: awswaf:managed:aws:atp:aggregate:volumetric:ip:failed_login_response:high ルールグループは、次の関連ラベルもリクエストに適用します。関連するアクションはありません。すべてのカウントは 10 分間の時間枠のものです。5 件以上の失敗したリクエストには awswaf:managed:aws:atp:aggregate:volumetric:ip:failed_login_response:medium 、1 件以上の失敗したリクエストには awswaf:managed:aws:atp:aggregate:volumetric:ip:failed_login_response:low 、10 件以上の成功したリクエスト には awswaf:managed:aws:atp:aggregate:volumetric:ip:successful_login_response:high 、5 件以上の成功したリクエストには awswaf:managed:aws:atp:aggregate:volumetric:ip:successful_login_response:medium 、1 件以上の成功したリクエストには awswaf:managed:aws:atp:aggregate:volumetric:ip:successful_login_response:low です。

ルール名	説明とラベル
`VolumetricSessionFailedLoginResponseHigh`	最近のログイン試行の失敗率が過度に高いクライアントセッションを検査します。大量とは、30 分間の時間枠にクライアントセッションからの失敗したログインリクエストが 10 件を超えることです。 レスポンス本文または JSON コンポーネントを検査するようにルールグループを設定している場合、 はこれらのコンポーネントタイプの最初の 65,536 バイト (64 KB) で成功または失敗のインジケータを検査 AWS WAF できます。 このルールは、同じクライアントセッションからの最新のログイン試行に対する保護されたリソースからの成功応答と失敗応答に基づいて、クライアントセッションからの新しいウェブリクエストにルールアクションとラベリングを適用します。ルールグループを設定するときに、成功数と失敗数のカウント方法を定義します。 **Note** AWS WAF は、Amazon CloudFront ディストリビューションを保護するウェブ ACLs でのみこのルールを評価します。 **Note** このルールが適用するしきい値は、レイテンシーによって若干異なる場合があります。ルールがその後のログイ

ルール名	説明とラベル
	ン試行に対して一致処理を開始する前に、許可されているよりも多い回数の失敗したログイン試行がクライアントから送信される可能性があります。 この検査は、ウェブリクエストにトークンがある場合にのみ適用されます。トークンは、アプリケーション統合 SDK、ならびに CAPTCHA および Challenge のルールアクションによってリクエストに追加されます。詳細については、「AWS WAF ウェブリクエストトークン」を参照してください。 ルールアクション: Block ラベル: `awswaf:managed:aws:atp:aggregate:volumetric:session:failed_login_response:high` ルールグループは、次の関連ラベルもリクエストに適用します。関連するアクションはありません。すべてのカウントは 30 分間の時間枠のものです。5 件以上の失敗したリクエストには `awswaf:managed:aws:atp:aggregate:volumetric:session:failed_login_response:medium` 、1 件以上の失敗したリクエストには `awswaf:managed:aws:atp:aggregate:volumetric:session:failed_login_response:low` 、10 件以上の成功したリクエスト には `awswaf:managed:aws:atp:aggregate:volumetric:session:successful_login_response:high` 、5 件以上の成功したリクエス

ルール名	説明とラベル
	トには awswaf:managed:aws:atp:aggregate:volumetric:session:successful_login_response:medium 、1 件以上の成功したリクエストには awswaf:managed:aws:atp:aggregate:volumetric:session:successful_login_response:low です。

AWS WAF Bot Control ルールグループ

VendorName: AWS、名前: AWSManagedRulesBotControlRuleSet、WCU: 50

Bot Control マネージドルールグループは、ボットからのリクエストを管理するルールを提供します。ボットは過剰なリソースを消費し、ビジネスメトリクスを歪め、ダウンタイムを引き起こし、悪意のあるアクティビティを実行する可能性があります。

保護レベル

Bot Control マネージドルールグループには、次の 2 レベルの保護から選択できます。

- 共通 – ウェブスクレイピングフレームワーク、検索エンジン、自動ブラウザなど、さまざまな自己識別ボットを検出します。このレベルの Bot Control 保護は、静的リクエストデータ分析など、従来のボット検出技術を使用して一般的なボットを識別します。ルールはこれらのボットからのトラフィックにラベルを付け、検証できないものはブロックします。
- ターゲットを絞った – 一般的な保護機能に加え、自己識別を行わない高度なボットに対するターゲットを絞った検出機能も追加されています。ターゲットを絞った保護は、レート制限と CAPTCHA およびバックグラウンドブラウザのチャレンジの組み合わせを使用してボットアクティビティを軽減します。
  - **TGT_** – ターゲットを絞った保護を提供するルールには、TGT_ で始まる名前が付いています。すべてのターゲットを絞った保護では、ブラウザ調査、フィンガープリント、行動ヒューリスティックなどの検出技術を使用して不正なボットトラフィックを識別します。
  - **TGT_ML_** – 機械学習を使用するターゲットを絞った保護のルールには、TGT_ML_ で始まる名前が付いています。これらのルールは、ウェブサイトトラフィック統計の自動機械学習分析を使用して、分散された調整されたボットアクティビティを示す異常な動作を検出します。 は、タ

イムスタンプ、ブラウザの特性、以前にアクセスした URL などのウェブサイトトラフィックに関する統計 AWS WAF を分析し、Bot Control 機械学習モデルを改善します。機械学習機能はデフォルトで有効になっていますが、ルールグループ設定で無効にすることができます。機械学習が無効になっている場合、 AWS WAF はこれらのルールを評価しません。

ターゲットを絞った保護レベルと AWS WAF レートベースのルールステートメントはどちらもレート制限を提供します。この 2 つのオプションの比較については、「レートベースのルールとターゲットを絞った Bot Control ルールにおけるレート制限のオプション」を参照してください。

このルールグループを使用する際の考慮事項

このルールグループは、 AWS WAFでのインテリジェントな脅威の軽減保護の一部です。詳細については、「AWS WAF インテリジェントな脅威軽減」を参照してください。

> **Note**
>
> このマネージドルールグループを使用する場合、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

コストを抑え、ウェブトラフィックを希望どおりに管理していることを確実にするには、インテリジェントな脅威の軽減のためのベストプラクティス のガイダンスに従ってこのルールグループを使用してください。

ボットの予測を改善するために、ターゲットを絞った保護レベルの ML ベースのルールの機械学習 (ML) モデルを定期的に更新しています。ML ベースのルールには、 で始まる名前がありますTGT_ML_。これらのルールによって行われたボット予測に突然かつ大幅な変更が見られる場合は、 アカウントマネージャーを通じてお問い合わせいただくか、 AWS Support センターでケースを作成してください。

このルールグループによって追加されるラベル

このマネージドルールグループは、評価対象のウェブリクエストにラベルを追加します。このラベルは、ウェブ ACL のこのルールグループの後に実行されるルールで使用できます。 AWS WAF は、ラベルを Amazon CloudWatch メトリクスにも記録します。ラベルとラベルメトリクスに関する一般的な情報については、「ウェブリクエストのラベル」および「ラベルメトリクスとディメンション」を参照してください。

トークンラベル

このルールグループは、 AWS WAF トークン管理を使用して、 AWS WAF トークンのステータスに従ってウェブリクエストを検査し、ラベル付けします。 は、クライアントセッションの追跡と検証にトークン AWS WAF を使用します。

トークンおよびトークンの管理の詳細については、「AWS WAF ウェブリクエストトークン」を参照してください。

ここで説明するラベルコンポーネントについては、「AWS WAF ラベル構文と命名要件」を参照してください。

クライアントセッションラベル

ラベルには、 AWS WAF トークン管理がクライアントセッションを識別するために使用する一意の識別子`awswaf:managed:token:id:`*`identifier`*が含まれています。この識別子は、クライアントが使用していたトークンを破棄した後など、新しいトークンを取得すると変わる可能性があります。

> **Note**
>
> AWS WAF は、このラベルの Amazon CloudWatch メトリクスを報告しません。

トークンステータスラベル: ラベル名前空間プレフィックス

トークンステータスラベルは、トークン、チャレンジのステータス、およびそれに含まれる CAPTCHA 情報を報告します。

各トークンステータスラベルは、次のプレフィクスの 1 つで始まります。

- `awswaf:managed:token:` — トークンの一般的なステータスを報告したり、トークンのチャレンジ情報のステータスを報告したりするために使用されます。
- `awswaf:managed:captcha:` — トークンの CAPTCHA 情報のステータスを報告するために使用されます。

トークンステータスラベル: ラベル名

プレフィックスに続いて、ラベルの残りの部分には詳細なトークンステータス情報が表示されます。

- `accepted` - リクエストトークンが存在し、以下の内容が含まれています。

  - 有効なチャレンジまたは CAPTCHA ソリューション。
  - 有効期限が切れていないチャレンジまたは CAPTCHA タイムスタンプ。
  - ウェブ ACL に有効なドメイン仕様。

  例: ラベル `awswaf:managed:token:accepted` には、ウェブリクエストのトークンに有効なチャレンジソリューション、有効期限が切れていないチャレンジタイムスタンプ、および有効なドメインがあることが示されています。
- `rejected` - リクエストトークンは存在するが、承認基準を満たしていない。

  トークン管理では、拒否されたラベルに加えて、理由を示すカスタムラベル名前空間と名前が追加されます。
  - `rejected:not_solved` — トークンにチャレンジまたは CAPTCHA ソリューションがない。
  - `rejected:expired` — ウェブ ACL に設定されているトークンイミュニティ時間によると、トークンのチャレンジまたは CAPTCHA タイムスタンプの有効期限が切れている。
  - `rejected:domain_mismatch` — トークンのドメインが、ウェブ ACL のトークンドメイン設定と一致しない。
  - `rejected:invalid` – 指定されたトークンを読み AWS WAF 取れませんでした。

  例: ラベル `awswaf:managed:captcha:rejected` と `awswaf:managed:captcha:rejected:expired` には、トークンの CAPTCHA タイムスタンプがウェブ ACL で設定されている CAPTCHA トークンのイミュニティ時間を超えたためにリクエストが拒否されたことが示されています。
- `absent` — リクエストにトークンがないか、トークンマネージャーがそれを読み取れなかった。

  例: ラベル `awswaf:managed:captcha:absent` には、リクエストにトークンがないことが示されています。

Bot Control ラベル

Bot Control マネージドルールグループは、名前空間プレフィックス `awswaf:managed:aws:bot-control:` の後にカスタム名前空間およびラベル名が続くラベルを生成します。ルールグループは、リクエストに複数のラベルを追加する場合があります。

各ラベルは、Bot Control ルールの検出結果を反映しています。

- `awswaf:managed:aws:bot-control:bot:` – リクエストに関連付けられたボットに関する情報。

- awswaf:managed:aws:bot-control:bot:name:*<name>*
  – ボット名は (利用可能な場合)、たとえばカスタム名前空間 bot:name:slurp、bot:name:googlebot、bot:name:pocket_parser。
- awswaf:managed:aws:bot-control:bot:category:*<category>* - ボットのカテゴリ。 で定義されます。 AWS WAF例えば、 bot:category:search_engineと ですbot:category:content_fetcher。
- awswaf:managed:aws:bot-control:bot:organization:*<organization>* – ボットのパブリッシャー (例: bot:organization:google)。
- awswaf:managed:aws:bot-control:bot:verified – 自己を識別し、Bot Control が検証できたボットを示すために使用されます。これは、一般的な望ましいボットに使用され、bot:category:search_engine のようなカテゴリラベルや bot:name:googlebot のような名前ラベルと組み合わせると便利です。

  > **Note**
  >
  > Bot Control は、ウェブリクエストの送信元の IP アドレスを使用して、ボットが検証されているかどうかを判断します。 AWS WAF 転送された IP 設定を使用して別の IP アドレスソースを検査するように設定することはできません。プロキシまたはロードバランサーを介してルーティングするボットを検証した場合、Bot Control ルールグループの前に実行するルールを追加してこの問題に対処します。転送された IP アドレスを使用し、検証済みのボットからのリクエストを明示的に許可するように新しいルールを設定します。転送した IP アドレスの詳細については、「転送された IP アドレス」を参照してください。

- awswaf:managed:aws:bot-control:bot:user_triggered:verified – 検証済みのボットに類似しているが、エンドユーザーによって直接呼び出される可能性のあるボットを示すために使用されます。このカテゴリのボットは、Bot Control のルールによって未検証のボットのように扱われます。
- awswaf:managed:aws:bot-control:bot:developer_platform:verified – 検証済みのボットに類似しているが、Google Apps Script などのデベロッパープラットフォームによってスクリプト作成のために使用されるボットを示すために使用されます。このカテゴリのボットは、Bot Control のルールによって未検証のボットのように扱われます。
- awswaf:managed:aws:bot-control:bot:unverified – 自己を識別するボットを示すために使用されるため、名前を付けて分類できます。ただし、そのボットのアイデンティティを個別に検証する場合に使用する情報は公開されていません。これらの種類のボットシグネチャは改ざんされる可能性があるため、未検証として扱われます。

- `awswaf:managed:aws:bot-control:targeted:<additional-details>` — Bot Control の対象となる保護に固有のラベルに使用されます。
- `awswaf:managed:aws:bot-control:signal:<signal-details>` および `awswaf:managed:aws:bot-control:targeted:signal:<signal-details>` — 一部の状況において、リクエストに関する追加情報を提供するために使用されます。

  シグナルラベルの例は、次のとおりです。これは網羅的なリストではありません。

  - `awswaf:managed:aws:bot-control:targeted:signal:browser_automation_extension` — Selenium IDE など、自動化をサポートするブラウザー拡張機能が検出されたことを示します。

    このラベルは、ユーザーがこのタイプの拡張をインストールすると、ユーザーが自発的に使用していない場合でも追加されます。このためのラベル照合ルールを実装する場合は、ルールロジックとアクション設定で誤検出が発生する可能性があることに注意してください。たとえば、オートメーションが使用されていることを確保するために、Block の代わりに CAPTCHA アクションを使用したり、このラベルマッチを他のラベルマッチと組み合わせたりすることができます。
  - `awswaf:managed:aws:bot-control:signal:automated_browser` — リクエストに、クライアントブラウザが自動化されている可能性があることを示す要素が含まれていることを示します。
  - `awswaf:managed:aws:bot-control:targeted:signal:automated_browser` – リクエストの AWS WAF トークンに、クライアントブラウザが自動化されている可能性があることを示すインジケータが含まれていることを示します。

`DescribeManagedRuleGroup` を呼び出すことにより、API を介してルールグループのすべてのラベルを取得できます。ラベルは、応答の `AvailableLabels` プロパティにリストされています。

Bot Control マネージドルールグループは、一般的に許可されている検証可能な一連のボットにラベルを適用します。ルールグループは、これらの検証済みボットをブロックしません。必要に応じて、Bot Control マネージドルールグループによって適用されたラベルを使用するカスタムルールを記述することで、それらのボットまたはそのサブセットをブロックできます。これと例の詳細については、「AWS WAF ボットコントロール」を参照してください。

Bot Control のルールリスト

このセクションには Bot Control ルールが表示されています。

> **Note**
>
> AWS マネージドルールのルールグループでルールに対して公開する情報は、不正な攻撃者がルールを回避するために使用できる情報を提供せずに、ルールを使用するのに十分な情報を提供することを目的としています。このドキュメントに記載されている以上の情報が必要な場合は、AWS Support センター にお問い合わせください。

ルール名	説明
`CategoryAdvertising`	広告目的で使用されるボットを検査します。例えば、プログラムによるウェブサイトへのアクセスを必要とするサードパーティーの広告サービスを使用する場合があります。 未検証のボットにのみ適用されるルールアクション: Block ラベル: `awswaf:managed:aws:bot-control:bot:category:advertising` 検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルとラベル `awswaf:managed:aws:bot-control:bot:verified` を追加します。
`CategoryArchiver`	アーカイブ目的で使用されるボットを検査します。これらのボットは、アーカイブを作成する目的でウェブをクロールし、コンテンツをキャプチャします。 未検証のボットにのみ適用されるルールアクション: Block ラベル: `awswaf:managed:aws:bot-control:bot:category:archiver`

ルール名	説明
	検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルとラベル `awswaf:managed:aws:bot-control:bot:verified` を追加します。
`CategoryContentFetcher`	ユーザーに代わってアプリケーションのウェブサイトにアクセスし、RSS フィードのようなコンテンツを取得したり、コンテンツを検証したりするボットを検査します。 未検証のボットにのみ適用されるルールアクション: Block ラベル: `awswaf:managed:aws:bot-control:bot:category:content_fetcher` 検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルとラベル `awswaf:managed:aws:bot-control:bot:verified` を追加します。

ルール名	説明
CategoryEmailClient	アプリケーションのウェブサイトを指し示すメール内のリンクをチェックするボットを検査します。これには、E メール内のリンクを確認したり、疑わしい電子メールにフラグを立てたりする企業や E メールプロバイダーによって実行されるボットが含まれます。 未検証のボットにのみ適用されるルールアクション: Block ラベル: awswaf:managed:aws:bot-control:bot:category:email_client 検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルとラベル awswaf:managed:aws:bot-control:bot:verified を追加します。
CategoryHttpLibrary	さまざまなプログラミング言語の HTTP ライブラリからボットによって生成されたリクエストを検査します。これらには、許可またはモニタリングする API リクエストが含まれる場合があります。 未検証のボットにのみ適用されるルールアクション: Block ラベル: awswaf:managed:aws:bot-control:bot:category:http_library 検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルとラベル awswaf:managed:aws:bot-control:bot:verified を追加します。

ルール名	説明
CategoryLinkChecker	壊れたリンクをチェックするボットを検査します。 未検証のボットにのみ適用されるルールアクション: Block ラベル: awswaf:managed:aws:bot-control:bot:category:link_checker 検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルとラベル awswaf:managed:aws:bot-control:bot:verified を追加します。
CategoryMiscellaneous	他のカテゴリに一致しないその他のボットを検査します。 未検証のボットにのみ適用されるルールアクション: Block ラベル: awswaf:managed:aws:bot-control:bot:category:miscellaneous 検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルとラベル awswaf:managed:aws:bot-control:bot:verified を追加します。

ルール名	説明
CategoryMonitoring	モニタリング目的で使用されるボットを検査します。例えば、パフォーマンスや稼働時間などをモニタリングするために、アプリケーションのウェブサイトに定期的に ping を送信するボットモニタリングサービスを使用することができます。 未検証のボットにのみ適用されるルールアクション: Block ラベル: awswaf:managed:aws:bot-control:bot:category:monitoring 検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルとラベル awswaf:managed:aws:bot-control:bot:verified を追加します。
CategoryScrapingFramework	ウェブサイトからのコンテンツのクロールと抽出を自動化するために使用される、ウェブスクレイピングフレームワークからのボットを検査します。 未検証のボットにのみ適用されるルールアクション: Block ラベル: awswaf:managed:aws:bot-control:bot:category:scraping_framework 検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルとラベル awswaf:managed:aws:bot-control:bot:verified を追加します。

ルール名	説明
CategorySearchEngine	ウェブサイトをクロールしてコンテンツをインデックス化し、その情報を検索エンジンの結果に利用できるようにする検索エンジンボットを検査します。 未検証のボットにのみ適用されるルールアクション: Block ラベル: awswaf:managed:aws:bot-control:bot:category:search_engine 検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルとラベル awswaf:managed:aws:bot-control:bot:verified を追加します。
CategorySecurity	ウェブアプリケーションの脆弱性をスキャンしたり、セキュリティ監査を実施したりするボットを検査します。例えば、ウェブアプリケーションのセキュリティをスキャン、モニタリング、または監査するサードパーティーのセキュリティベンダーを利用することができます。 未検証のボットにのみ適用されるルールアクション: Block ラベル: awswaf:managed:aws:bot-control:bot:category:security 検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルとラベル awswaf:managed:aws:bot-control:bot:verified を追加します。

ルール名	説明
CategorySeo	検索エンジンの最適化に使用されるボットを検査します。例えば、検索エンジンのランキングを向上させるために、サイトをクロールする検索エンジンツールを使用することができます。 未検証のボットにのみ適用されるルールアクション: Block ラベル: awswaf:managed:aws:bot-control:bot:category:seo 検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルとラベル awswaf:managed:aws:bot-control:bot:verified を追加します。
CategorySocialMedia	ユーザーがコンテンツを共有するときに、コンテンツの概要を提供するためにソーシャルメディアプラットフォームで使用されるボットを検査します。 未検証のボットにのみ適用されるルールアクション: Block ラベル: awswaf:managed:aws:bot-control:bot:category:social_media 検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルとラベル awswaf:managed:aws:bot-control:bot:verified を追加します。

ルール名	説明
CategoryAI	人工知能 (AI) ボットを検査します。 ルールアクション: Block ラベル: awswaf:managed:aws:bot-control:bot:category:ai
SignalAutomatedBrowser	クライアントブラウザが自動化されている可能性があることを示す要素がないか、リクエストを検査します。自動ブラウザはテストやスクレイピングに使用できます。例えば、以下のようなタイプのブラウザを使用して、アプリケーションウェブサイトのモニタリングや検証を行うことができます。 ルールアクション: Block ラベル: awswaf:managed:aws:bot-control:signal:automated_browser
SignalKnownBotDataCenter	ボットが通常使用するデータセンターのインジケーターがないかどうかを検査します。 ルールアクション: Block ラベル: awswaf:managed:aws:bot-control:signal:known_bot_data_center

ルール名	説明
SignalNonBrowserUserAgent	ウェブブラウザからではないと考えられるユーザーエージェント文字列を検査します。このカテゴリには API リクエストが含まれる場合があります。 ルールアクション: Block ラベル: awswaf:managed:aws:bot-control:signal:non_browser_user_agent

ルール名	説明
TGT_VolumetricIpTokenAbsent	過去 5 分間のクライアントからのリクエストで、有効なチャレンジトークンが含まれていないものが 5 つ以上あるかどうかを検査します。トークンの詳細については、「AWS WAF ウェブリクエストトークン」を参照してください。 **Note** 同じクライアントからのリクエストで、最近トークンの欠落が発生するようになったという場合は、このルールがトークンを有するリクエストに一致する可能性があります。 このルールが適用されるしきい値は、レイテンシーによって若干異なる場合があります。 このルールは、トークンラベル awswaf:managed:token:absent とは異なる方法で欠落したトークンを処理します。トークンラベルは、トークンがない個々のリクエストにラベルを付けます。このルールは、各クライアント IP のトークンが欠落しているリクエスト数を把握し、制限を超過するクライアントに一致させます。 検証済みボットではないクライアントにのみ適用されるルールアクション: Challenge ラベル: awswaf:managed:aws:bot-control:targeted:aggregate:volumetric:ip:token_absent 検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルと

ルール名	説明
	ラベル `awswaf:managed:aws:bot-control:bot:verified` を追加します。

ルール名	説明
TGT_VolumetricSession	5 分ウィンドウ内でクライアントセッションからのリクエスト数が異常に多いかどうかを検査します。この評価は、過去のトラフィックパターンを使用して が AWS WAF 維持する標準ボリューメトリックベースラインとの比較に基づいています。 この検査は、ウェブリクエストにトークンがある場合にのみ適用されます。トークンは、アプリケーション統合 SDK、ならびに CAPTCHA および Challenge のルールアクションによってリクエストに追加されます。詳細については、「AWS WAF ウェブリクエストトークン」を参照してください。 **Note** このルールは、有効にしてから有効になるまでに 5 分かかることがあります。Bot Control は、現在のトラフィックと が AWS WAF 計算するトラフィックベースラインを比較することで、ウェブトラフィックの異常な動作を識別します。 検証済みボットではないクライアントにのみ適用されるルールアクション: CAPTCHA ラベル: `awswaf:managed:aws:bot-control:targeted:aggregate:volumetric:session:high`

ルール名	説明
	ルールグループは、最小しきい値を超える中規模および低ボリュームのリクエストに次のラベルを適用します。これらのレベルでは、クライアントが検証されているかどうかにかかわらず、ルールは何も実行しません。すなわち、`awswaf:managed:aws:bot-control:targeted:aggregate:volumetric:session:medium` および `awswaf:managed:aws:bot-control:targeted:aggregate:volumetric:session:low` 。 検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルとラベル `awswaf:managed:aws:bot-control:bot:verified` を追加します。

ルール名	説明
TGT_SignalAutomatedBrowser	リクエストのトークンで、クライアントブラウザが自動化されている可能性があることを示す要素がないかを検査します。詳細については、「AWS WAF トークンの特性」を参照してください。 この検査は、ウェブリクエストにトークンがある場合にのみ適用されます。トークンは、アプリケーション統合 SDK、ならびに CAPTCHA および Challenge のルールアクションによってリクエストに追加されます。詳細については、「AWS WAF ウェブリクエストトークン」を参照してください。 検証済みボットではないクライアントにのみ適用されるルールアクション: CAPTCHA ラベル: awswaf:managed:aws:bot-control:targeted:signal:automated_browser 検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルとラベル awswaf:managed:aws:bot-control:bot:verified を追加します。

ルール名	説明
TGT_SignalBrowserInconsistency	ブラウザ調査のデータに一貫性がないかどうか検査します。詳細については、「AWS WAF トークンの特性」を参照してください。 この検査は、ウェブリクエストにトークンがある場合にのみ適用されます。トークンは、アプリケーション統合 SDK、ならびに CAPTCHA および Challenge のルールアクションによってリクエストに追加されます。詳細については、「AWS WAF ウェブリクエストトークン」を参照してください。 検証済みボットではないクライアントにのみ適用されるルールアクション: CAPTCHA ラベル: awswaf:managed:aws:bot-control:targeted:signal:browser_inconsistency 検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルとラベル awswaf:managed:aws:bot-control:bot:verified を追加します。

ルール名	説明
TGT_TokenReuseIp	5 つを超える異なる IP アドレス間で単一のトークンが使用されているかどうかを検査します。 **Note** このルールが適用するしきい値は、レイテンシーによって若干異なる場合があります。ルールアクションが適用される前に、いくつかのリクエストが制限を超えることがあります。 ルールアクション: Count ラベル: awswaf:managed:aws:bot-control:targeted:aggregate:volumetric:session:token_reuse:ip

ルール名	説明
`TGT_ML_CoordinatedActivityMedium` および `TGT_ML_CoordinatedActivityHigh`	分散または協調ボットのアクティビティと一致する異常な動作がないか検査します。ルールレベルは、リクエストのグループが協調攻撃に参加しているかどうかの信頼度のレベルを示します。 **Note** これらのルールは、ルールグループが機械学習 (ML) を使用するように設定されている場合にのみ実行されます。この場合の設定については、「AWS WAF ボットコントロールマネージドルールグループをウェブ ACL に追加する」を参照してください。 AWS WAF は、ウェブサイトトラフィック統計の機械学習分析を通じてこの検査を実行します。 は、数分ごとにウェブトラフィック AWS WAF を分析し、多くの IP アドレスに分散されている低強度で長時間のボットの検出のために分析を最適化します。 これらのルールは、協調攻撃が進行中ではないと判断される前に、ごく少数のリクエストに一致する場合があります。そのため、表示された一致が 1 つか 2 つしかない場合は、結果が誤検出である可能性があります。ただし、これらのルールからの一致が多数表示されている場合は、協調攻撃を受けていると考えられます。

ルール名	説明
	Note ML オプションで Bot Control ターゲットルールを有効にしてから、これらのルールが有効になるまでに最大 24 時間かかることがあります。Bot Control は、現在のトラフィックを が計算した AWS WAF トラフィックベースラインと比較することで、ウェブトラフィックの異常な動作を識別します。 は、Bot Control のターゲットルールを ML オプションで使用している間 AWS WAF のみベースラインを計算し、意味のあるベースラインを確立するまでに最大 24 時間かかる場合があります。 ボットの予測を改善するために、これらのルールの機械学習モデルを定期的に更新しています。これらのルールによってボット予測が突然大幅に変化した場合は、アカウントマネージャーに連絡するか、AWS Support センターでケースを開いてください。 検証済みボットではないクライアントにのみ適用されるルールアクション: • ミディアム: Count • 高: Count ラベル: `awswaf:managed:aws:bot-control:targeted:aggregate:coordinated_activity:medium` および `awswaf:managed:aws:bot-cont`

ルール名	説明
	`rol:targeted:aggregate:coordinated_activity:high` 検証済みボットについては、ルールグループはアクションを実行しませんが、ルールラベルとラベル `awswaf:managed:aws:bot-control:bot:verified` を追加します。 ルールグループには信頼度が低いことを示すラベル `awswaf:managed:aws:bot-control:targeted:aggregate:coordinated_activity:low` も追加されますが、これらのリクエストに対してルールは適用されず、アクションも実行されません。

バージョニングされた AWS マネージドルールのルールグループのデプロイ

AWS は、リリース候補、静的バージョン、デフォルトバージョンの 3 つの標準デプロイで、バージョン管理された AWS マネージドルールのルールグループに変更をデプロイします。さらに、例外デプロイをリリースしたり、デフォルトバージョンのデプロイをロールバックしたりする必要がある AWS 場合があります。

> **Note**
>
> このセクションは、バージョニングされた AWS マネージドルールのルールグループにのみ適用されます。バージョニングされていない唯一のルールグループは、IP 評価ルールグループです。

トピック

- AWS マネージドルールのルールグループのデプロイの通知
- AWS マネージドルールの標準展開の概要
- AWS マネージドルールの一般的なバージョン状態
- AWS マネージドルールのリリース候補デプロイ
- AWS マネージドルールの静的バージョンのデプロイ

- AWS マネージドルールのデフォルトバージョンデプロイ
- AWS マネージドルールの例外のデプロイ
- AWS マネージドルールのデフォルトデプロイメントロールバック

AWS マネージドルールのルールグループのデプロイの通知

バージョニングされた AWS マネージドルールのルールグループはすべて、デプロイの SNS 更新通知を提供し、すべて同じ SNS トピックの Amazon リソースネーム (ARN) を使用します。バージョニングされていない唯一のルールグループは、IP 評価ルールグループです。

保護に影響するデプロイ (デフォルトバージョンへの変更など) の場合、 AWS は SNS 通知を提供して、計画されたデプロイについて通知し、デプロイが開始されるタイミングを知らせます。保護に影響しないデプロイ (リリース候補や静的バージョンのデプロイなど) の場合、 AWS は、デプロイが開始された後や完了した後でも通知を行う場合があります。新しい静的バージョンのデプロイが完了すると、 は、 の Changelog AWS マネージドルールの変更ログと のドキュメント履歴ページでこのガイド AWS を更新しますドキュメント履歴。

が AWS マネージドルールのルールグループ AWS に提供するすべての更新を受信するには、このガイドの任意の HTML ページから RSS フィードをサブスクライブし、 AWS マネージドルールのルールグループの SNS トピックをサブスクライブします。SNS 通知のサブスクライブについては、「」を参照してくださいマネージドルールグループに対する新しいバージョンと更新の通知を受け取る。

SNS 通知の内容

Amazon SNS 通知のフィールドには、常に件名、メッセージ、および が含まれます MessageAttributes。追加のフィールドは、メッセージのタイプと通知対象のマネージドルールグループによって異なります。AWSManagedRulesCommonRuleSet の通知リストの例を次に示します。

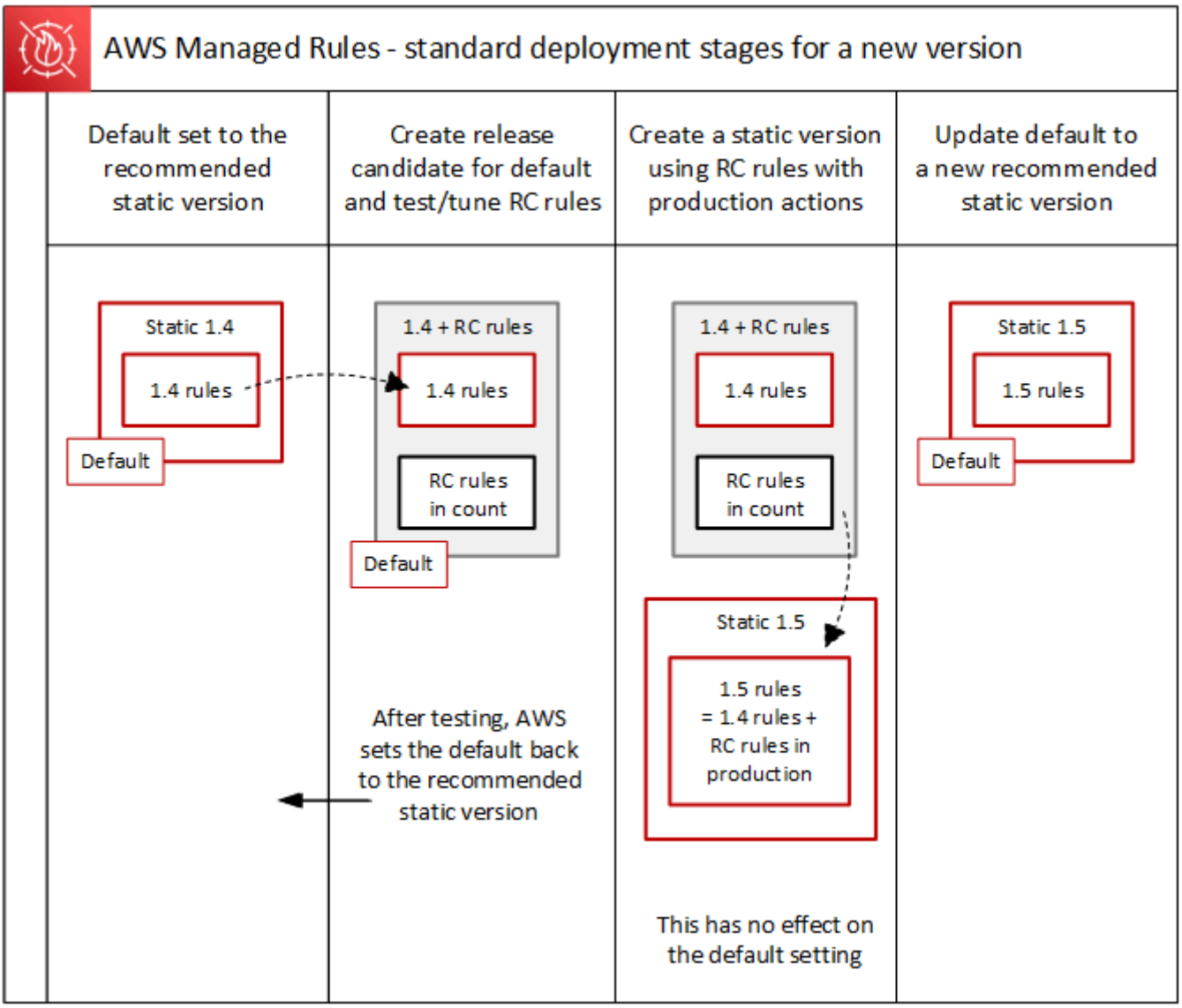
```
{
    "Type": "Notification",
    "MessageId": "4286b830-a463-5e61-bd15-e1ae72303868",
    "TopicArn": "arn:aws:sns:us-west-2:123456789012:MyTopic",
    "Subject": "New version available for rule group AWSManagedRulesCommonRuleSet",
    "Message": "Welcome to AWSManagedRulesCommonRuleSet version 1.5! We've updated
 the regex specification in this version to improve protection coverage, adding
 protections against insecure deserialization. For details about this change, see
 http://updatedPublicDocs.html. Look for more exciting updates in the future! ",
    "Timestamp": "2021-08-24T11:12:19.810Z",
```

```
    "SignatureVersion": "1",
    "Signature": "EXAMPLEHXgJm...",
    "SigningCertURL": "https://sns.us-west-2.amazonaws.com/SimpleNotificationService-
f3ecfb7224c7233fe7bb5f59f96de52f.pem",
    "SubscribeURL": "https://sns.us-west-2.amazonaws.com/?
Action=ConfirmSubscription&TopicArn=arn:aws:sns:us-
west-2:123456789012:MyTopic&Token=2336412f37...",
    "MessageAttributes": {
        "major_version": {
            "Type": "String",
            "Value": "v1"
        },
        "managed_rule_group": {
            "Type": "String",
            "Value": "AWSManagedRulesCommonRuleSet"
        }
    }
}
```

AWS マネージドルールの標準展開の概要

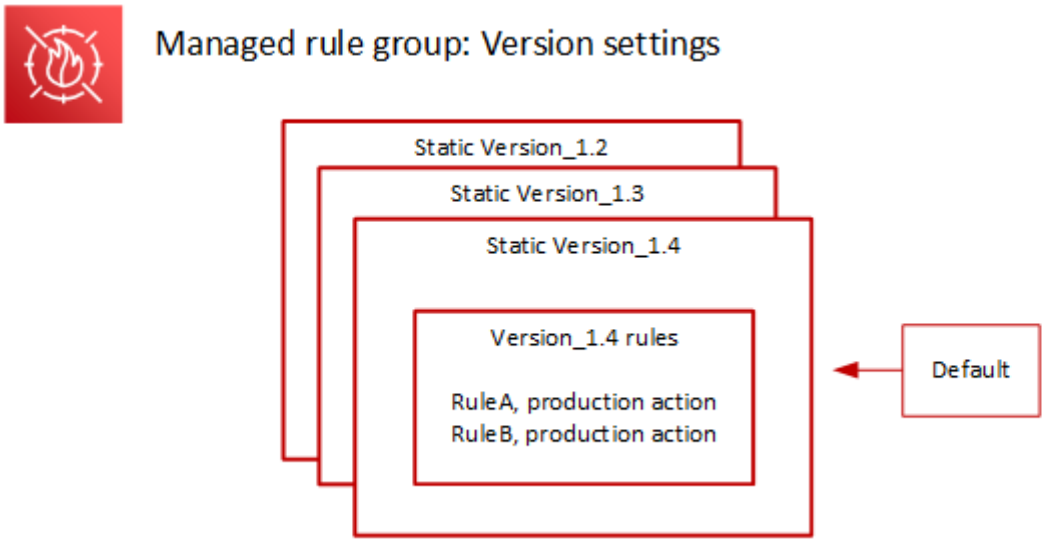
AWS リリース候補、静的バージョン、デフォルトバージョンの 3 つの標準デプロイステージを使用して、 AWS 新しいマネージドルール機能を展開します。

次の図は、これらの標準的なデプロイを示しています。それぞれについて、以降のセクションで詳しく説明します。



AWS マネージドルールの一般的なバージョン状態

通常、バージョン管理されたマネージドルールグループには有効期限が切れていない静的バージョンがいくつかあり、デフォルトバージョンは推奨されている静的バージョンを指します。 AWS 次の図は、典型的な一連の静的バージョンとデフォルトバージョンの設定における例を示しています。



静的バージョンにおけるほとんどのルールの稼働アクションは Block ですが、別のアクションにセットされる場合があります。ルールアクション設定の詳細については、「AWS マネージドルールルールグループリスト」で各ルールグループに関するルールのリストを参照してください。

AWS マネージドルールのリリース候補デプロイ

AWS マネージドルールグループのルール変更候補セットがある場合は、一時的なリリース候補デプロイメントでそれらをテストします。 AWS 本番環境のトラフィックに対してカウントモードで候補ルールを評価し、誤検出の軽減を含む最終的な調整作業を行います。 AWS テストでは、デフォルトバージョンのルールグループを使用するすべての顧客を対象に、この方法で候補ルールをリリースします。リリース候補のデプロイは、ルールグループの静的バージョンを使用するお客様には適用されません。

デフォルトバージョンを使用する場合、リリース候補のデプロイは、ルールグループによるウェブトラフィックの管理方法を変更しません。候補ルールがテストされている間、次のことに気づくかもしれません。

- デフォルトバージョン名が `Default (using Version_X.Y)` から `Default (using Version_X.Y_PLUS_RC_COUNT)` に変更された。
- CloudWatch RC_COUNT名前にが付いているAmazonのその他のカウントメトリックス。これらはリリース候補ルールによって生成されます。

AWS リリース候補を約 1 週間テストしてから削除し、デフォルトバージョンを現在推奨されている静的バージョンにリセットします。

AWS リリース候補のデプロイメントに対して以下のステップを実行します。

1. リリース候補の作成 — 現在推奨されている静的バージョン、 AWS つまりデフォルトが指しているバージョンに基づいてリリース候補を追加します。

   リリース候補の名前は、静的バージョン名に _PLUS_RC_COUNT が付加されたものです。例えば、現在推奨されている静的バージョンが Version_2.1 である場合、リリース候補の名前は Version_2.1_PLUS_RC_COUNT になります。

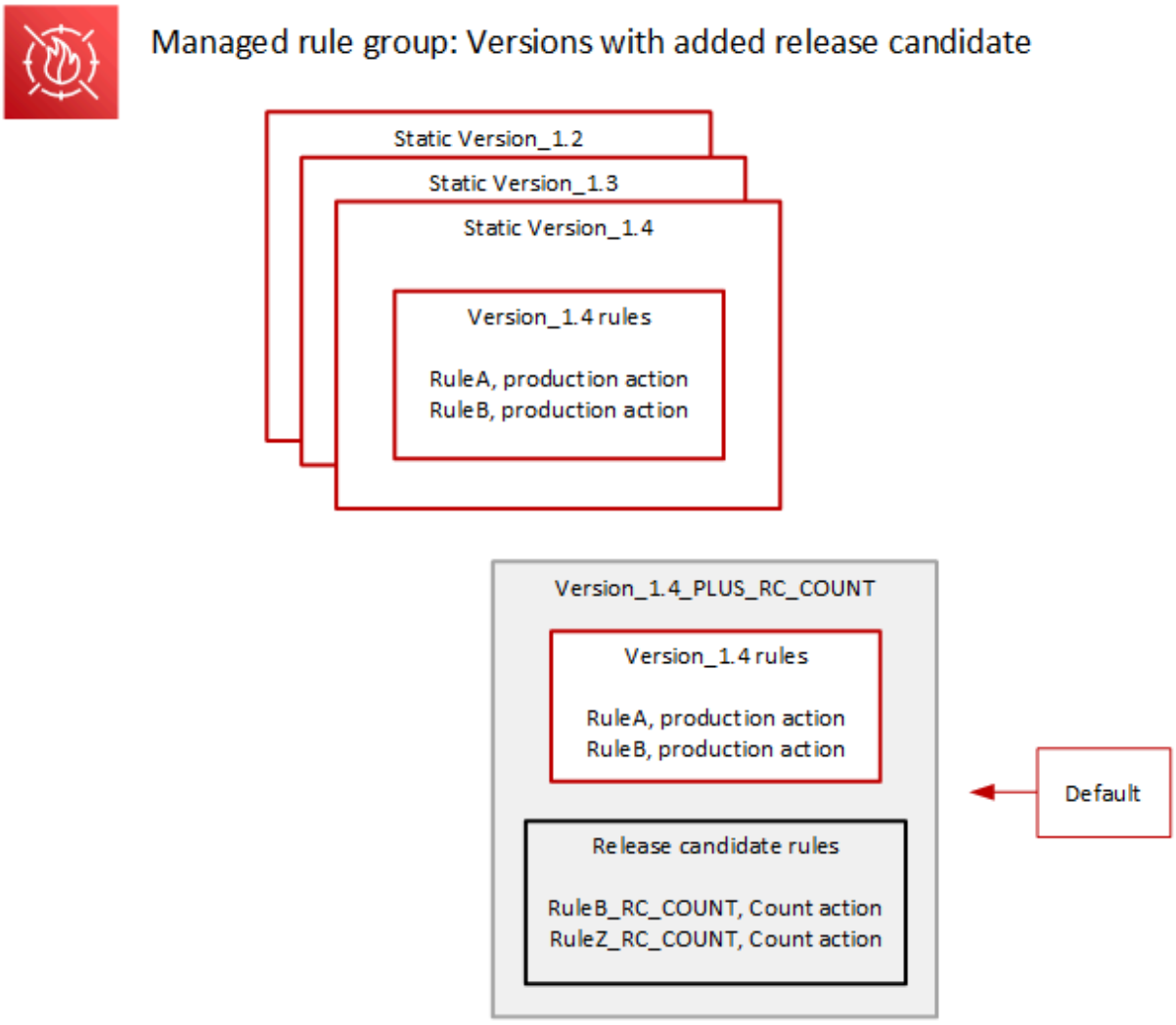
   リリース候補には次のルールが含まれています。

   - ルール設定を変更せずに、現在推奨されている静的バージョンから正確にコピーされたルール。
   - ルールアクションが Count に設定され、名前が _RC_COUNT で終わる候補の新しいルール。

     ほとんどの候補ルールは、ルールグループに既に存在するルールに対して提案された改善を提供します。これらの各ルールの名前は、既存のルールの名前に _RC_COUNT が付加されたものです。
2. デフォルトバージョンをリリース候補に設定してテスト — AWS 新しいリリース候補を参照するようにデフォルトバージョンを設定し、本番環境のトラフィックに対してテストを実行します。通常、テストには約 1 週間かかります。

   デフォルトバージョンの名前が、静的バージョンのみを示すもの (Default（using Version_1.4）など) から、静的バージョンとリリース候補ルールを示すもの (Default (using Version_1.4_PLUS_RC_COUNT) など) に変更されます。この命名スキームにより、ウェブトラフィックの管理に使用している静的バージョンを特定できます。

   次の図は、この時点でのサンプルルールグループバージョンの状態を示しています。



リリース候補ルールは常に Count アクションで設定されているため、ルールグループがウェブトラフィックを管理する方法が変更されることはありません。

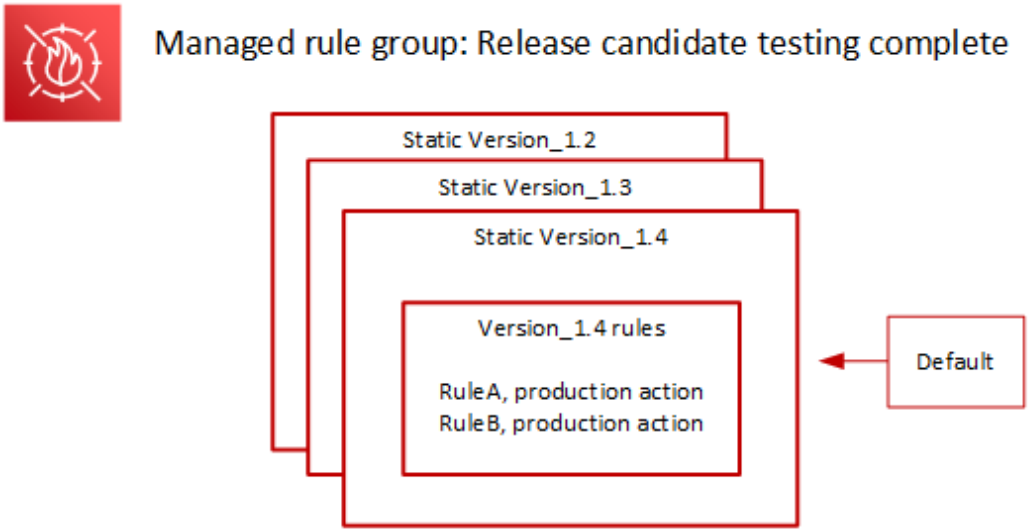
リリース候補ルールは、 AWS 動作の検証と誤検出の特定に使用する Amazon CloudWatch カウントメトリクスを生成します。 AWS 必要に応じて調整を行い、リリース候補のカウントルールの動作を調整します。

リリース候補バージョンは静的バージョンではないため、静的ルールグループバージョンのリストから選択することはできません。デフォルトバージョンの仕様では、リリース候補バージョンの名前のみが表示されます。

3. デフォルトバージョンを推奨の静的バージョンに戻す — リリース候補ルールをテストした後、AWS デフォルトバージョンを現在の推奨静的バージョンに戻します。_PLUS_RC_COUNTデフォルトのバージョン名設定では末尾が削除され、 CloudWatch ルールグループはリリース候補ルー

ルのカウントメトリックの生成を停止します。これはサイレント変更であり、デフォルトバージョンロールバックのデプロイとは異なります。

次の図は、リリース候補のテストが完了した後のサンプルルールグループのバージョンの状態を示しています。



タイミングと通知

AWS ルールグループの改善をテストするために、必要に応じてリリース候補バージョンをデプロイします。

- SNS — AWS デプロイの開始時に SNS 通知を送信します。通知には、リリース候補がテストされる推定時間が表示されます。テストが完了すると、2 AWS 回目の通知なしで、デフォルトを静的なバージョン設定に戻します。
- 変更ログ — このタイプのデプロイでは、 AWS 変更ログや本ガイドの他の部分は更新されません。

AWS マネージドルールの静的バージョンのデプロイ

AWS は、リリース候補がルールグループに貴重な変更を提供すると判断した場合、リリース候補に基づいてルールグループの新しい静的バージョンを AWS デプロイします。このデプロイでは、ルールグループのデフォルトバージョンは変更されません。

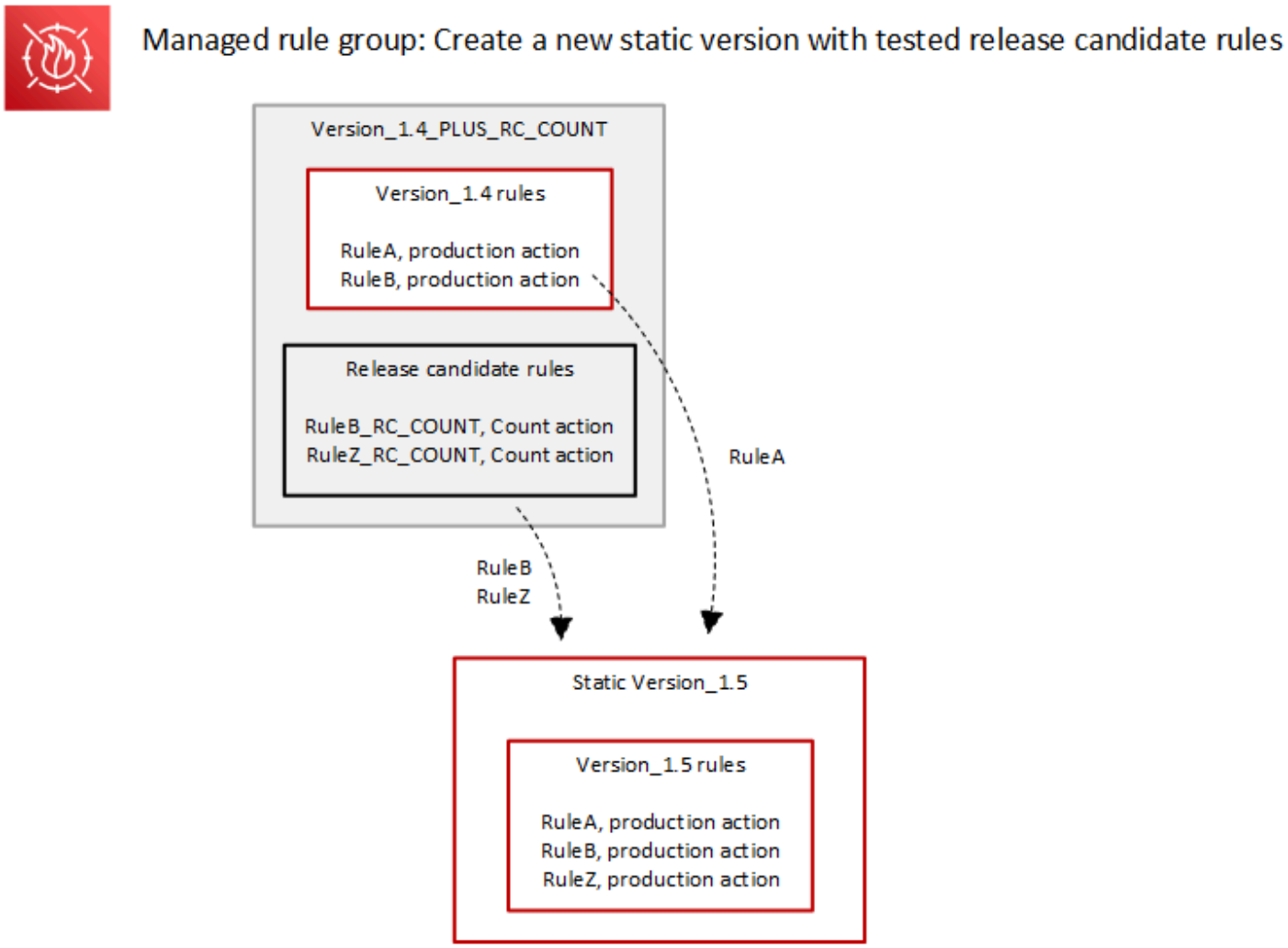
新しい静的バージョンには、リリース候補からの次のルールが含まれています。

- リリース候補ルールの中に置換候補がない、以前の静的バージョンのルール。

- 次の変更を加えて、候補ルールをリリースします。
  - AWS は、リリース候補のサフィックス を削除してルール名を変更します_RC_COUNT。
  - AWS は、ルールアクションを から本番稼働用ルールアクションCountに変更します。

  以前の既存のルールを置き換えるリリース候補ルールの場合、これは新しい静的バージョンの以前のルールの機能を置き換えます。

次の図は、リリース候補から新しい静的バージョンを作成する方法を示しています。



デプロイ後、新しい静的バージョンをテストして、必要に応じて保護に使用できます。AWS マネージドルールルールグループリスト のルールグループのルールリストで、新規および更新されたルールアクションと説明を確認できます。

静的バージョンはデプロイ後にイミュータブルであり、 AWS が期限切れになったときにのみ変更されます。バージョンのライフサイクルについては、「バージョニングされたマネージドルールグループ」を参照してください。

タイミングと通知

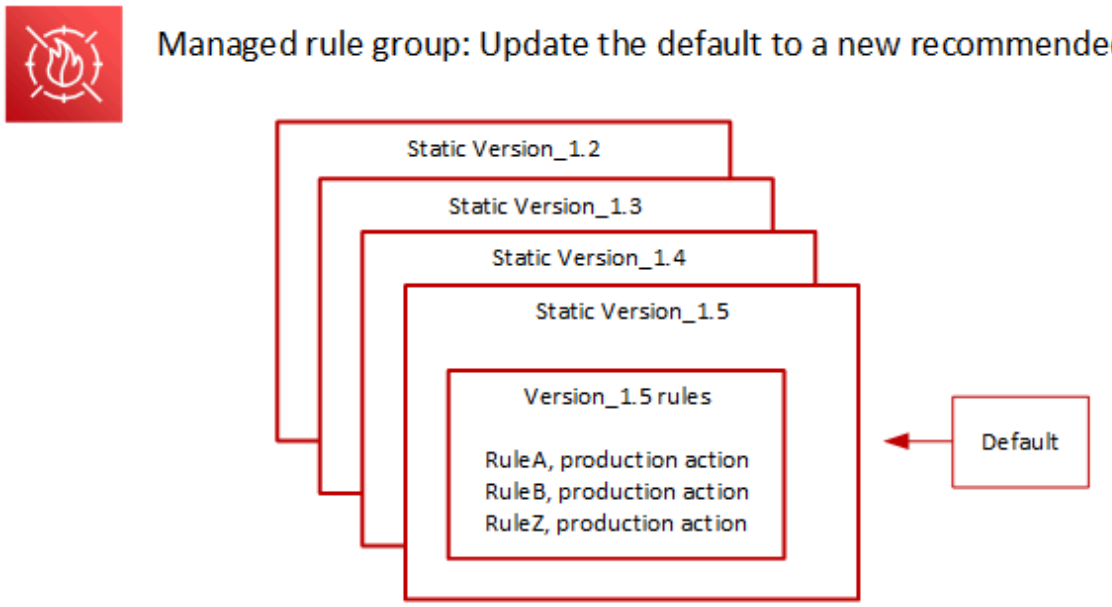
AWS は、ルールグループ機能の改善をデプロイするために、必要に応じて新しい静的バージョンをデプロイします。静的バージョンのデプロイは、デフォルトのバージョン設定には影響しません。

- SNS – デプロイが完了すると SNS 通知 AWS を送信します。
- 変更ログ – 利用可能なすべての場所でデプロイが完了すると、 AWS WAF は必要に応じてこのガイドのルールグループ定義 AWS を更新し、 AWS マネージドルールのルールグループ変更ログとドキュメント履歴ページでリリースを通知します。

AWS マネージドルールのデフォルトバージョンデプロイ

AWS 新しい静的バージョンでは現在のデフォルトよりもルールグループの保護が強化されていると判断された場合は、 AWS デフォルトバージョンを新しい静的バージョンに更新します。 AWS 1 つの静的バージョンをルールグループのデフォルトバージョンに昇格する前に、複数の静的バージョンをリリースする可能性があります。

次の図は、 AWS デフォルトバージョン設定を新しい静的バージョンに移行した後のルールグループバージョンの例の状態を示しています。



この変更をデフォルトバージョンにデプロイする前に、 AWS 今後の変更をテストして準備できるように通知します。デフォルトバージョンを使用する場合は、何も実行せずに、更新後もそのバージョンに留まることができます。デフォルトバージョンのデプロイの計画された開始の前に、代わりに新しいバージョンへの切り替えを遅らせたい場合、デフォルトが設定されている静的バージョンを使用するように、ルールグループを明示的に設定できます。

タイミングと通知

AWS 現在使用中のものとは異なる静的バージョンをルールグループに推奨する場合、デフォルトバージョンを更新します。

- SNS — 対象となるデプロイ日の少なくとも 1 週間前に SNS AWS 通知を送信し、デプロイ日のデプロイ開始時に別の通知を送信します。各通知には、ルールグループ名、デフォルトバージョンの更新先の静的バージョン、デプロイ日、 AWS および更新が実行される各リージョンのデプロイ予定タイミングが含まれます。
- 変更ログ — AWS このタイプのデプロイに関する変更ログや本ガイドの他の部分は更新されません。

AWS マネージドルールの例外のデプロイ

AWS 重大なセキュリティリスクに対処する更新を迅速に展開するために、標準の展開段階を迂回する可能性があります。例外デプロイには、標準デプロイタイプのいずれかが含まれる場合があり、AWS リージョン全体で迅速に展開される可能性があります。

AWS 例外デプロイメントについては、できる限り事前に通知します。

タイミングと通知

AWS 例外デプロイメントは必要な場合にのみ実行されます。

- SNS — 対象となる展開日のできるだけ前に SNS AWS 通知を送信し、展開の開始時に別の SNS 通知を送信します。各通知には、ルールグループ名、行われる変更、およびデプロイ日が含まれます。
- 変更ログ — 静的バージョンのデプロイの場合、利用可能なすべての場所でデプロイが完了した後、 AWS 必要に応じてこのガイドのルールグループ定義を更新し、 AWS Managed Rulesルールグループの変更ログとドキュメント履歴ページでリリースを通知します。 AWS WAF

AWS マネージドルールのデフォルトデプロイメントロールバック

特定の条件下では、 AWS デフォルトバージョンを以前の設定にロールバックすることがあります。ロールバックには通常、 AWS すべてのリージョンで 10 分もかかりません。

AWS ロールバックは、許容できないほど高いレベルの誤検出など、静的バージョンの重大な問題を軽減するためだけに実行されます。

デフォルトバージョン設定のロールバック後、問題のある静的バージョンの有効期限切れと、 AWS 問題に対処するための新しい静的バージョンのリリースの両方を早めます。

タイミングと通知

AWS 必要な場合にのみデフォルトバージョンのロールバックを実行します。

- SNS — AWS ロールバック時に 1 つの SNS 通知を送信します。通知には、ルールグループ名、デフォルトバージョンが設定されるバージョン、およびデプロイ日が含まれます。このデプロイタイプは非常に高速なので、通知はリージョンのタイミング情報を提供しません。
- 変更ログ — この種のデプロイでは、 AWS 変更ログや本ガイドの他の部分は更新されません。

AWS マネージドルールに関する免責事項

AWS マネージドルールは、一般的な Web の脅威からユーザーを保護するように設計されています。マニュアルに従って使用すると、 AWS マネージドルールルールグループはアプリケーションのセキュリティをさらに強化します。ただし、 AWS マネージドルールルールグループは、 AWS 選択したリソースによって決定されるセキュリティ責任の代わりとなるものではありません。責任共有モデルを参照して、 AWS 内のリソースが適切に保護されていることを確認してください。

AWS マネージドルールの変更ログ

このセクションでは、2019 年 11 月のリリース AWS WAF 以降の の AWS マネージドルールに対する変更を一覧表示します。

> **Note**
>
> この変更ログは、 の AWS マネージドルールのルールとルールグループに対する変更を報告します AWS WAF。
> の場合IP 評価ルールグループ、この変更ログはルールとルールグループへの変更を報告し、ルールが使用する IP アドレスリストのソースへの重大な変更を報告します。これらのリスト

は動的であるため、IP アドレスリスト自体の変更はレポートされません。IP アドレスリストについてご質問がある場合は、 アカウントマネージャーに問い合わせるか、 AWS Support センターでケースを開いてください。

ルールグループおよびルール	説明	日付
Linux オペレーティングシステムマネージドルールグループ すべてのルール	このルールグループの静的バージョン 2.3 をリリースしました。これにより、デフォルトのバージョン設定は変更されません。 検出の向上を図るために署名を追加しました。	2024-06-06
AWS WAF Bot Control ルールグループ AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループ AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) ルールグループ	これで、ボットと不正ルールグループがバージョニングされます。これらのルールグループのいずれかを使用している場合、この更新によってウェブトラフィックの処理方法が変更されることはありません。 この更新では、現在のルールグループバージョンを静的バージョン 1.0 に設定し、それを指すようにデフォルトバージョンを設定します。 バージョニングされたマネージドルールの詳細については、以下を参照してください。 • バージョニングされたマネージドルールグループ	2024-05-29

ルールグループおよびルール	説明	日付
	• バージョニングされた AWS マネージドルールのルールグループのデプロイ • マネージドルールグループに対する新しいバージョンと更新の通知を受け取る	

ルールグループおよびルール	説明	日付
POSIX オペレーティングシステムマネージドルールグループ • UNIXShellCommandsVariables_QUERYARGUMENTS • UNIXShellCommandsVariables_QUERYSTRING • UNIXShellCommandsVariables_HEADER • UNIXShellCommandsVariables_BODY	このルールグループの静的バージョン 3.0 をリリースしました。これにより、デフォルトのバージョン設定は変更されません。 削除しUNIXShellCommandsVariables_QUERYARGUMENTS 、 に置き換えましたUNIXShellCommandsVariables_QUERYSTRING 。のラベルに一致するルールがある場合はUNIXShellCommandsVariables_QUERYARGUMENTS 、このバージョンを使用するときに、 のラベルと一致するようにルールを切り替えますUNIXShellCommandsVariables_QUERYSTRING 。新しいラベルは ですawswaf:managed:aws:posix-os:UNIXShellCommandsVariables_QueryString 。 すべてのヘッダーUNIXShellCommandsVariables_HEADER に一致するルール を追加しました。 マネージドルールグループ内のすべてのルールを更新し、	2024-05-28

ルールグループおよびルール	説明	日付
	検出ロジックを改善しました。 のラベルの大文字と小文字の区別を修正しましたUNIXShellCommandsVariables_BODY 。	
コアルールセット (CRS) マネージドルールグループ • CrossSiteScripting*	このルールグループの静的バージョン 1.12 をリリースしました。 検出の向上と誤検出の低減のため、クロスサイトのスクリプティングルール全体にシグネチャーを追加しました。	2024-05-21
SQL データベースマネージドルールグループ • SQLi_BODY • SQLi_QUERYARGUMENTS • SQLiExtendedPatterns_QUERYARGUMENTS	このルールグループの静的バージョン 1.2 をリリースしました。 リストされたルールにJS_DECODE テキスト変換を追加しました。	2024-05-14

ルールグループおよびルール	説明	日付
既知の不正な入力マネージドルールグループ • JavaDeserializationRCE_BODY • JavaDeserializationRCE_QUERYSTRING • Log4JRCE_QUERYSTRING • Log4JRCE_BODY • Log4JRCE_HEADER	このルールグループの静的バージョン 1.22 をリリースしました。 リストされたルールにJS_DECODE テキスト変換を追加しました。	2024-05-08
POSIX オペレーティングシステムマネージドルールグループ	このルールグループの静的バージョン 2.2 をリリースしました。 両方のルールにJS_DECODE テキスト変換を追加しました。	2024-05-08
Windows オペレーティングシステムマネージドルールグループ • PowerShellCommands_BODY	このルールグループの静的バージョン 2.1 をリリースしました。 検出を改善するPowerShellCommands_BODY ために、に署名を追加しました。	2024-05-03

ルールグループおよびルール	説明	日付
Amazon IP 評価リストマネージドルールグループ • AWSManagedIPReputationList	IP 評価リストのソースを更新して、悪意のあるアクティビティに積極的に関与しているアドレスの識別を改善し、誤検出を減らしました。 このルールグループはバージョニングされていないため、この更新には新しいバージョンは含まれません。	2024-03-13
既知の不正な入力マネージドルールグループ	このルールグループの静的バージョン 1.21 をリリースしました。 検出を改善し、誤検出を減らすためにシグネチャを追加しました。	2023-12-16
既知の不正な入力マネージドルールグループ • ExploitablePaths_URIPATH	このルールグループの静的バージョン 1.20 をリリースしました。 Atlassian Confluence CVE-2023-22518 の不適切な認証の脆弱性に一致するリクエストの検出を追加するように ExploitablePaths_URIPATH ルールを更新しました。この脆弱性は Confluence Data Center および Server のすべてのバージョンに影響します。詳細については、「NIST: National Vulnerability Database: CVE-2023-22518 Detail」を参照してください。	2023-12-14

ルールグループおよびルール	説明	日付
コアルールセット (CRS) マネージドルールグループ • CrossSiteScripting*	このルールグループの静的バージョン 1.11 をリリースしました。 検出の向上と誤検出の低減のため、クロスサイトのスクリプティングルール全体にシグネチャーを追加しました。	2023-12-06
AWS WAF Bot Control ルールグループ • 新しいラベル: awswaf:managed:aws:bot-control:targeted:aggregate:coordinated_activity:low	ルールグループの対象保護レベルラベルに調整されたアクティビティの下限ラベルを追加しました。このラベルはいずれのルールにも関連付けられていません。このラベルは、中レベルおよび高レベルのルールとラベルに追加されるものです。	2023-12-05
Bot Control ラベル • ラベル: awswaf:managed:aws:bot-control:targeted:signal:browser_automation_extension	自動化を支援するブラウザ拡張機能が検出されたことを示すシグナルラベルをルールグループに追加しました。このラベルは個々のルールに固有のものではありません。	2023-11-14
コアルールセット (CRS) マネージドルールグループ • EC2MetaDataSSRF_QUERYARGUMENTS	このルールグループの静的バージョン 1.10 をリリースしました。 1 つのルールを更新し、検出の向上と誤検出の低減を実現しました。	2023-11-02

ルールグループおよびルール	説明	日付
コアルールセット (CRS) マネージドルールグループ • EC2MetaDataSSRF_BODY • EC2MetaDataSSRF_COOKIE • EC2MetaDataSSRF_URIPATH • EC2MetaDataSSRF_QUERYARGUMENTS	このルールグループの静的バージョン 1.9 をリリースしました。 ルールが更新され、検出の向上と誤検出の低減を実現しました。	2023-10-30
POSIX オペレーティングシステムマネージドルールグループ • UNIXShellCommandsVariables_QUERYARGUMENTS	このルールグループの静的バージョン 2.1 をリリースしました。 クエリ引数ルールを更新して、検出を改善しました。	2023-10-12

ルールグループおよびルール	説明	日付
コアルールセット (CRS) マネージドルールグループ • GenericLFI_QUERYARGUMENTS • GenericLFI_URIPATH • RestrictedExtensions_URIPATH • RestrictedExtensions_QUERYARGUMENTS	このルールグループの静的バージョン 1.8 をリリースしました。 ルールを更新して検出を改善しました。	2023-10-11

ルールグループおよびルール	説明	日付
既知の不正な入力マネージドルールグループ • `ExploitablePaths_URIPATH`	例外のデプロイ: このルールグループの静的バージョン 1.19 をリリースしました。バージョン 1.19 を使用するようにデフォルトバージョンを更新しました。 Atlassian Confluence CVE-2023-22515 の権限昇格の脆弱性に一致するリクエストの検出を追加するように `ExploitablePaths_URIPATH` ルールを更新しました。この脆弱性は、Atlassian Confluence の一部のバージョンに影響を及ぼします。詳細については、「NIST: National Vulnerability Database: CVE-2023-22515 Detail」および「Atlassian Support: FAQ for CVE-2023-22515」を参照してください。 デプロイタイプの詳細については、「AWS マネージドルールの例外のデプロイ」を参照してください。	2023-10-04

ルールグループおよびルール	説明	日付
既知の不正な入力マネージドルールグループ • Host_localhost_HEADER • Log4J* • JavaDeserialization*	例外のデプロイ: このルールグループの静的バージョン 1.18 をリリースしました。これは、この静的バージョンの迅速なロールアウトであり、バージョン 1.19 の作成とロールアウトに対応するためのものです。 検出を改善するため、Host_localhost_HEADER ルールとすべての Log4J および Java 逆シリアル化ルールを更新しました。 デプロイタイプの詳細については、「AWS マネージドルールの例外のデプロイ」を参照してください。	2023-10-04

ルールグループおよびルール	説明	日付
AWS WAF Bot Control ルールグループ • TGT_TokenReuseIp • TGT_ML_CoordinatedActivityMedium • TGT_ML_CoordinatedActivityHigh	Count アクションでルールグループにルールを追加しました。 トークン再利用 IP ルールは、IP アドレス間でのトークンの共有を検出してカウントします。 調整されたアクティビティルールは、ウェブサイトのトラフィックを機械学習 (ML) で自動的に分析し、ボット関連のアクティビティを検出します。ルールグループ設定で、ML の使用をオプトアウトできます。このリリースでは、ターゲットを絞った保護レベルを現在使用しているユーザーが ML の使用にオプトインされます。オプトアウトすると、調整されたアクティビティルールが無効になります。	2023-09-06
AWS WAF Bot Control ルールグループ • CategoryAI	ルール CategoryAI をルールグループに追加しました。	2023-08-30

ルールグループおよびルール	説明	日付
コアルールセット (CRS) マネージドルールグループ • RestrictedExtensions_URIPATH • RestrictedExtensions_QUERYARGUMENTS • EC2MetaDataSSRF_COOKIE • EC2MetaDataSSRF_QUERYARGUMENTS • EC2MetaDataSSRF_BODY • EC2MetaDataSSRF_URIPATH	このルールグループの静的バージョン 1.7 をリリースしました。 制限付き拡張ルールおよび EC2 メタデータ SSRF ルールを更新し、検出の向上と誤検出の低減を実現しました。	2023-07-26
AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) ルールグループ 新しいルールグループ内のすべてのルール	ルールグループ AWSManagedRulesACFPRuleSet を追加しました。	2023-06-13

ルールグループおよびルール	説明	日付
Linux オペレーティングシステムマネージドルールグループ • LFI_HEADER • LFI_URIPATH • LFI_QUERYSTRING	このルールグループの静的バージョン 2.2 をリリースしました。 検出の向上を図るために署名を追加しました。	2023-05-22
コアルールセット (CRS) マネージドルールグループ • RestrictedExtensions_URIPATH • RestrictedExtensions_QUERYARGUMENTS • CrossSiteScripting_COOKIE • CrossSiteScripting_QUERYARGUMENTS • CrossSiteScripting_BODY • CrossSiteScripting_URIPATH	このルールグループの静的バージョン 1.6 をリリースしました。 クロスサイトスクリプティング (XSS) および制限付き拡張ルールを更新し、検出の向上と誤検出の低減を実現しました。	2023-04-28

ルールグループおよびルール	説明	日付
PHP アプリケーションマネージドルールグループ • 「PHPHighRiskMethodsVariables_BODY 」を更新 • 「PHPHighRiskMethodsVariables_QUERYARGUMENTS 」を削除 • 「PHPHighRiskMethodsVariables_QUERYSTRING 」を追加 • 「PHPHighRiskMethodsVariables_HEADER 」を追加	このルールグループの静的バージョン 2.0 をリリースしました。 すべてのルールで検出を改善するために署名を追加しました。 ルール PHPHighRiskMethodsVariables_QUERYARGUMENTS を、クエリ引数だけでなくクエリ文字列全体を検査する PHPHighRiskMethodsVariables_QUERYSTRING に置き換えました。 ルール PHPHighRiskMethodsVariables_HEADER を追加し、すべてのヘッダーを含むようにカバレッジを拡張しました。 標準の AWS マネージドルールのラベル付けに合わせて、次のラベルを更新しました。 • 現在の名前: PHPHighRiskMethodsVariables_BODY 新しい名前: PHPHighRiskMethodsVariables_Body • 現在の名前: PHPHighRiskMethodsVariables_QUERYARGUMENTS 新しい名前: PHPHighRi	2023-02-27

ルールグループおよびルール	説明	日付
	skMethodsVariables_QueryString	
AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループ • VolumetricIpFailedLoginResponseHigh • VolumetricSessionFailedLoginResponseHigh	保護された Amazon CloudFront ディストリビューションで使用するログインレスポンス検査ルールを追加しました。これらのルールは、最近のログイン試行の失敗回数が過度に多い IP アドレスやクライアントセッションからの新たなログイン試行をブロックします。	2023-02-15
コアルールセット (CRS) マネージドルールグループ • NoUserAgent_HEADER • CrossSiteScripting_COOKIE • CrossSiteScripting_QUERYARGUMENTS • CrossSiteScripting_BODY • CrossSiteScripting_URIPATH	このルールグループの静的バージョン 1.5 をリリースしました。 検出を改善するため、クロスサイトスクリプティング (XSS) フィルターを更新しました。	2023-01-25

ルールグループおよびルール	説明	日付
Linux オペレーティングシステムマネージドルールグループ • LFI_COOKIE - 削除済み • LFI_HEADER - 追加済み • LFI_URIPATH • LFI_QUERYSTRING	このルールグループの静的バージョン 2.1 をリリースしました。 ルール LFI_COOKIE およびそのラベル awswaf:managed:aws:linux-os:LFI_Cookie を削除し、新しいルール LFI_HEADER およびそのラベル awswaf:managed:aws:linux-os:LFI_Header に置き換えました。この変更により、検査が複数のヘッダーに拡張されます。 検出を改善するため、すべてのルールにテキスト変換および署名を追加しました。	2022-12-15

ルールグループおよびルール	説明	日付
コアルールセット (CRS) マネージドルールグループ • NoUserAgent_HEADER • CrossSiteScripting_COOKIE • CrossSiteScripting_QUERYARGUMENTS • CrossSiteScripting_BODY • CrossSiteScripting_URIPATH	このルールグループの静的バージョン 1.4 をリリースしました。 すべての NULL バイトを削除するため、テキスト変換を NoUserAgent_HEADER に追加しました。検出を改善するため、クロスサイトのスクリプティングルールのフィルターを更新しました。	2022-12-05

ルールグループおよびルール	説明	日付
既知の不正な入力マネージドルールグループ • JavaDeserializationRCE_BODY • JavaDeserializationRCE_URIPATH • JavaDeserializationRCE_HEADER • JavaDeserializationRCE_QUERYSTRING • Host_localhost_HEADER	このルールグループの静的バージョン 1.17 をリリースしました。 Java 逆シリアル化ルールを更新し、1.10.0 以前の Apache Commons Text バージョンのリモートコード実行 (RCE) 脆弱性である Apache CVE-2022-42889 に一致するリクエストの検出を追加しました。詳細については、「NIST: National Vulnerability Database: CVE-2022-42889 Detail」(NIST: 国家脆弱性データベース: CVE-2022-42889 詳細) および「CVE-2022-42889: Apache Commons Text prior to 1.10.0 allows RCE when applied to untrusted input due to insecure interpolation defaults」(CVE-2022-42889: 信頼されない入力に適用された場合にセキュアでない補間デフォルトが原因で 1.10.0 以前の Apache Commons Text で RCE が許可される) を参照してください。 Host_localhost_HEADER での検出を改善しました。	2022-10-20

ルールグループおよびルール	説明	日付
既知の不正な入力マネージドルールグループ • Log4JRCE_HEADER • Log4JRCE_QUERYSTRING • Log4JRCE_URIPATH • Log4JRCE_BODY	このルールグループの静的バージョン 1.16 をリリースしました。 バージョン 1.15 で AWS 識別された誤検出を削除しました。	2022-10-05
POSIX オペレーティングシステムマネージドルールグループ PHP アプリケーションマネージドルールグループ WordPress アプリケーションマネージドルールグループ	記載されているラベル名を修正しました。	2022-09-19
IP 評価ルールグループ • AWSManagedIPDDoSList	この変更により、ルールグループがウェブトラフィックを処理する方法が変更されることはありません。 Amazon 脅威インテリジェンスに従い、DDoS アクティビティに積極的に関与している IP アドレスを検査する Count アクションを含む新しいルールが追加されました。	2022-08-30

ルールグループおよびルール	説明	日付
既知の不正な入力マネージドルールグループ • Log4JRCE • Log4JRCE_HEADER • Log4JRCE_QUERYSTRING • Log4JRCE_URIPATH • Log4JRCE_BODY • JavaDeserializationRCE_HEADER • JavaDeserializationRCE_BODY • JavaDeserializationRCE_URIPATH • JavaDeserializationRCE_QUERYSTRING • Host_localhost_HEADER • PROPFIND_METHOD	このルールグループの静的バージョン 1.15 をリリースしました。 誤検出のよりきめ細かにモニタリングと管理を行うため、Log4JRCE を削除して Log4JRCE_HEADER 、Log4JRCE_QUERYSTRING 、Log4JRCE_URI 、Log4JRCE_BODY に置き換えました。 PROPFIND_METHOD とすべての JavaDeserializationRCE* と Log4JRCE* ルールに対する検出とブロックを改善するため、署名を追加しました。 Host_localhost_HEADER と全ての JavaDeserializationRCE* ルールにおける大文字化の修正をするためにラベルを更新しました。 JavaDeserializationRCE_HEADER の記述を修正しました。	2022-08-22

ルールグループおよびルール	説明	日付
AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループ • UnsupportedCognitoIDP	Amazon Cognito ユーザープールのウェブトラフィック用のアカウント乗っ取り防止マネージドルールグループの使用を防止するルールを追加しました。	2022-08-11
コアルールセット (CRS) マネージドルールグループ	AWS には、ルールグループのバージョン Version_1.2 と Version_2.0 の有効期限がスケジュールされています。このバージョンの有効期限は 2022 年 9 月 9 日に失効します。バージョンの有効期限の詳細については、「バージョニングされたマネージドルールグループ」を参照してください。	2022-06-09
コアルールセット (CRS) マネージドルールグループ • GenericLFI_URIPATH GenericRFI_URIPATH	このルールグループのバージョン 1.3 をリリースしました。このリリースでは、検出を改善するため、GenericLFI_URIPATH および GenericRFI_URIPATH のルールにある一致する署名が更新されます。	2022-05-24
AWS WAF Bot Control ルールグループ • CategoryEmailClient	ルール CategoryEmailClient をルールグループに追加しました。	2022-04-06

ルールグループおよびルール	説明	日付
既知の不正な入力マネージドルールグループ • JavaDeserializationRCE_HEADER • JavaDeserializationRCE_BODY • JavaDeserializationRCE_URI • JavaDeserializationRCE_QUERYSTRING	このルールグループのバージョン 1.14 をリリースしました。4 つの JavaDeserializtionRCE ルールが Block モードに移行します。	2022-03-31
既知の不正な入力マネージドルールグループ • JavaDeserializationRCE_HEADER_RC_COUNT • JavaDeserializationRCE_BODY_RC_COUNT • JavaDeserializationRCE_URI_RC_COUNT • JavaDeserializationRCE_QUERYSTRING_RC_COUNT	このルールグループのバージョン 1.13 をリリースしました。Spring Core および Cloud Function RCE 脆弱性のテキスト変換を更新しました。これらのルールは、メトリクスを収集して一致したパターンを評価するため、カウントモードになっています。ラベルは、カスタムルール内のリクエストをブロックするために使用できます。後続のバージョンは、これらのルールがブロックモードになった状態でデプロイされます。	2022-03-31

ルールグループおよびルール	説明	日付
既知の不正な入力マネージドルールグループ • JavaDeserializationRCE_HEADER_RC_COUNT • JavaDeserializationRCE_BODY_RC_COUNT • JavaDeserializationRCE_URI_RC_COUNT • JavaDeserializationRCE_QUERYSTRING_RC_COUNT • Log4JRCE_HEADER • Log4JRCE_QUERYSTRING • Log4JRCE_URI • Log4JRCE_BODY • Log4JRCE	このルールグループのバージョン 1.12 をリリースしました。Spring Core および Cloud Function RCE の脆弱性の署名を追加しました。これらのルールは、メトリクスを収集して一致したパターンを評価するため、カウントモードになっています。ラベルは、カスタムルール内のリクエストをブロックするために使用できます。後続のバージョンは、これらのルールがブロックモードになった状態でデプロイされます。 ルール Log4JRCE_HEADER 、Log4JRCE_QUERYSTRING 、Log4JRCE_URI 、Log4JRCE_BODY を削除してルール Log4JRCE に置き換えました。	2022-03-30
IP 評価ルールグループ • AWSManagedReconnaissanceList	アクションをカウントからブロックに変更するように AWSManagedReconnaissanceList ルールを更新しました。	2022-02-15

ルールグループおよびルール	説明	日付
AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループ 新しいルールグループ内のすべてのルール	ルールグループ AWSManagedRulesATPRuleSet を追加しました。	2022-02-11
既知の不正な入力マネージドルールグループ • Log4JRCE • Log4JRCE_HEADER • Log4JRCE_QUERYSTRING • Log4JRCE_URI • Log4JRCE_BODY	このルールグループのバージョン 1.9 をリリースしました。この機能を柔軟に使用できるように、ルール Log4JRCE を削除し、ルール Log4JRCE_HEADER 、Log4JRCE_QUERYSTRING 、Log4JRCE_URI 、および Log4JRCE_BODY に置き換えました。検出とブロックを改善するために署名を追加しました。	2022-01-28
コアルールセット (CRS) • CrossSiteScripting_URIPATH • CrossSiteScripting_BODY • CrossSiteScripting_QUERYARGUMENTS • CrossSiteScripting_COOKIE	このルールグループのバージョン 2.0 をリリースしました。これらのルールでは、誤検出を減らすために検出シグネチャをチューニングしました。URL_DECODE テキスト変換をダブル URL_DECODE_UNI テキスト変換に置き換えました。HTML_ENTITY_DECODE テキスト変換を追加しました。	2022-01-10

ルールグループおよびルール	説明	日付
コアルールセット (CRS) • RestrictedExtensions_URIPATH • RestrictedExtensions_QUERYARGUMENTS	このルールグループのバージョン 2.0 のリリースの一部として、URL_DECODE_UNI テキスト変換が追加されました。RestrictedExtensions_URIPATH から URL_DECODE テキスト変換を削除しました。	2022-01-10
SQL データベース • SQLi_BODY • SQLi_QUERYARGUMENTS • SQLi_COOKIE • SQLi_URIPATH • SQLiExtendedPatterns_BODY • SQLiExtendedPatterns_QUERYARGUMENTS	このルールグループのバージョン 2.0 をリリースしました。URL_DECODE テキスト変換をダブル URL_DECODE_UNI テキスト変換に置き換え、COMPRESS_WHITE_SPACE テキスト変換を追加しました。 検出シグネチャを SQLiExtendedPatterns_QUERYARGUMENTS に追加しました。 SQLi_BODY に JSON 検査を追加しました。 ルール SQLiExtendedPatterns_BODY を追加しました。 ルール SQLi_URIPATH を削除しました。	2022-01-10

ルールグループおよびルール	説明	日付
既知の不正な入力 • Log4JRCE	ヘッダーの検査と一致基準を改善するために、ルール Log4JRCE のバージョン 1.8 をリリースしました。	2021-12-17
既知の不正な入力 • Log4JRCE	一致基準をチューニングし、追加のヘッダーを検査するために、ルール Log4JRCE のバージョン 1.4 をリリースしました。バージョン 1.5 をリリースし、一致条件をチューニングしました。	2021-12-11
既知の不正な入力 • Log4JRCE • BadAuthToken_COOKIE_AUTHORIZATION	Log4j 内で最近公開されたセキュリティ問題に対応して、ルール Log4JRCE バージョン 1.2 を追加しました。詳細については、「CVE-2021-44228」を参照してください。このルールは、共通の URI パス、クエリ文字列、リクエストボディの最初の 8 KB、および共通ヘッダーを検査します。ルールはダブル URL_DECODE_UNI テキスト変換を使用します。一致基準をチューニングし、追加のヘッダーを検査するために、Log4JRCE のバージョン 1.3 をリリースしました。 ルール BadAuthToken_COOKIE_AUTHORIZATION を削除しました。	2021-12-10

次の表には、2021 年 12 月よりも前の変更が記載されています。

ルールグループおよびルール	説明	Date	
Amazon IP 評価リスト	`AWSManagedReconnaissanceList`	モニタリング/カウントモードの `AWSManagedReconnaissanceList` ルールを追加しました。このルールには、AWS リソースに対して偵察を実行している IP アドレスが含まれています。	2021-11-23

Windows オペレーティングシステム	`WindowsShellCommands`
	
	`PowerShellCommands`
		WindowsShell コマンドに `WindowsShellCommands_COOKIE` 、、 の 3 つの新しいルールが追加され`WindowsShellCommands_QUERYARGUMENTS` ましたWindowsShellCommands_BODY 。
		
		新しい PowerShell ルール を追加しましたPowerShellCommands_COOKIE 。
			2021-11-23

ルールグループおよびルール	説明	Date	
		文字列 _Set1 と _Set2 を削除して、PowerShellComands ルールの命名を再構築しました。 より包括的な検出シグネチャを PowerShellRules に追加しました。 すべての Windows オペレーティングシステムのルールに URL_DECODE_UNI テキスト変換を追加しました。	

ルールグループおよびルール	説明	Date	
Linux オペレーティングシステム	LFI_URIPATH LFI_QUERYSTRING LFI_BODY LFI_COOKIE	ダブル URL_DECODE テキスト変換をダブル URL_DECODE_UNI に置き換えました。 2 つ目のテキスト変換として NORMALIZE_PATH_WIN を追加しました。 LFI_BODY ルールを LFI_COOKIE ルールに置き換えました。 すべての LFI ルールに、より包括的な検出シグネチャを追加しました。	2021-11-23
コアルールセット (CRS)	SizeRestrictions_BODY	8 KB を超える本文ペイロードを持つウェブリクエストをブロックするためのサイズ制限を縮小しました。これまでは、制限は 10 KB でした。	2021-10-27

ルールグループおよびルール	説明	Date	

コアルールセット (CRS)	`EC2MetaDataSSRF_BODY`
	`EC2MetaDataSSRF_COOKIE`
	`EC2MetaDataSSRF_URIPATH`
	`EC2MetaDataSSRF_QUERYARGUMENTS`
		検出シグネチャを追加しました。ブロックを改善するためにダブルユニコード URL デコードを追加しました。	2021-10-27

コアルールセット (CRS)	`GenericLFI_QUERYARGUMENTS`
	`GenericLFI_URIPATH`
	`RestrictedExtensions_URIPATH`
	`RestrictedExtensions_QUERYARGUMENTS`
		ブロックを改善するためにダブルユニコード URL デコードを追加しました。	2021-10-27

ルールグループおよびルール	説明	Date	
コアルールセット (CRS)	GenericRFI_QUERYARGUMENTS GenericRFI_BODY GenericRFI_URIPATH	お客様からのフィードバックに基づいて、誤検出を減らすためにルールシグネチャを更新しました。ブロックを改善するためにダブルユニコード URL デコードを追加しました。	2021-10-27
すべて	すべてのルール	AWS WAF ラベル付けをまだサポートしていないすべてのルールにラベルのサポートを追加しました。	2021-10-25
Amazon IP 評価リスト	AWSManagedIPReputationList_xxxx	IP 評価リストを再構築し、ルール名からサフィックスを削除し、 AWS WAF ラベルのサポートを追加しました。	2021-05-04
匿名 IP リスト	AnonymousIPList HostingProviderList	AWS WAF ラベルのサポートが追加されました。	2021-05-04
Bot Control	すべて	Bot Control ルールセットを追加しました。	2021-04-01

ルールグループおよびルール	説明	Date	
コアルールセット (CRS)	GenericRFI_QUERYARGUMENTS	二重 URL デコードを追加しました。	2021-03-03
コアルールセット (CRS)	RestrictedExtensions_URIPATH	ルールの設定を改善し、追加の URL デコードを追加しました。	2021-03-03
管理者保護	AdminProtection_URIPATH	二重 URL デコードを追加しました。	2021-03-03
既知の不正な入力	ExploitablePaths_URIPATH	ルールの設定を改善し、追加の URL デコードを追加しました。	2021-03-03
Linux オペレーティングシステム	LFI_QUERYARGUMENTS	ルールの設定を改善し、追加の URL デコードを追加しました。	2021-03-03
Windows オペレーティングシステム	すべて	ルールの設定を改善しました。	2020-09-23
PHP アプリケーション	PHPHighRiskMethodsVariables_QUERYARGUMENTS PHPHighRiskMethodsVariables_BODY	ブロックを改善するために、テキスト変換を HTML デコードから URL デコードに変更しました。	2020-09-16

ルールグループおよびルール	説明	Date	

POSIX オペレーティングシステム	`UNIXShellCommandsVariables_QUERYARGUMENTS`
	
	`UNIXShellCommandsVariables_BODY`
		ブロックを改善するために、テキスト変換を HTML デコードから URL デコードに変更しました。	2020-09-16

コアルールセット	`GenericLFI_QUERYARGUMENTS`
	
	`GenericLFI_URIPATH`
	
	GenericLFI_BODY
		ブロックを改善するために、テキスト変換を HTML デコードから URL デコードに変更しました。	2020-08-07

Linux オペレーティングシステム	`LFI_URIPATH`
	
	`LFI_QUERYARGUMENTS`
	
	`LFI_BODY`
		検出とブロックを改良するために、テキスト変換を HTML エンティティデコードから URL デコードに変更しました。	2020-05-19

匿名 IP リスト	すべて	ビューワー ID の難読化を許可するサービスからのリクエストをブロックする IP 評価ルールグループ の新しいルールグループは、ボットや地理的制限の回避に対する軽減に役立ちます。	2020-03-06

ルールグループおよびルール	説明	Date	
WordPress アプリケーション	`WordPressExploitableCommands_QUERYSTRING`	クエリ文字列内の悪用の対象となるコマンドをチェックする新しいルール。	2020-03-03

コアルールセット (CRS)	`SizeRestrictions_QUERYSTRING`
	
	`SizeRestrictions_Cookie_HEADER`
	
	`SizeRestrictions_BODY`
	
	`SizeRestrictions_URIPATH`
		精度を向上させるために、サイズ値の制約を調整しました。	2020-03-03

SQL データベース	`SQLi_URIPATH`	ルールは、メッセージ URI をチェックするようになりました。	2020-01-23

SQL データベース	`SQLi_BODY`
	
	`SQLi_QUERYARGUMENTS`
	
	`SQLi_COOKIE`
		テキスト変換を更新しました。	2019-12-20

ルールグループおよびルール	説明	Date	

コアルールセット (CRS)	`CrossSite Scripting _URIPATH`
	
	`CrossSite Scripting_BODY`
	
	`CrossSite Scripting _QUERYARG UMENTS`
	
	`CrossSite Scripting _COOKIE`
		テキスト変換を更新しました。	2019-12-20

AWS Marketplace マネージドルールグループ

AWS Marketplace マネージドルールグループは、 の AWS Marketplace コンソールからサブスクリプションで使用できますAWS Marketplace。 AWS Marketplace マネージドルールグループをサブスクライブしたら、 で使用できます AWS WAF。 AWS Firewall Manager AWS WAF ポリシーで AWS Marketplace ルールグループを使用するには、組織内の各アカウントがルールグループをサブスクライブする必要があります。

本番トラフィックに使用する前に、 AWS WAF 保護の変更をテストして調整します。詳細については、「AWS WAF 保護機能のテストと調整」を参照してください。

AWS Marketplace ルールグループの料金

AWS Marketplace ルールグループは、長期契約や最低契約金なしで利用できます。ルールグループをサブスクライブすると、月額料金 (時間数で按分) およびボリュームに基づく継続中のリクエスト料金が課金されます。詳細については、「」の「 のAWS WAF 料金」と「各 AWS Marketplace ルールグループの説明」を参照してくださいAWS Marketplace。

AWS Marketplace ルールグループについて質問がありますか？

AWS Marketplace 販売者が管理するルールグループに関する質問や機能の変更をリクエストするには、プロバイダーのカスタマーサポートチームにお問い合わせください。連絡先情報を検索するには、「AWS Marketplace」のプロバイダーのリストを参照してください。

AWS Marketplace ルールグループプロバイダーは、ルールグループを更新する方法や、ルールグループがバージョニングされているかどうかなど、ルールグループを管理する方法を決定します。プロバイダーは、ルール、ルールアクション、ルールが一致するウェブリクエストに追加するラベルなど、ルールグループの詳細も決定します。

AWS Marketplace マネージドルールグループのサブスクライブ

AWS WAF コンソールで AWS Marketplace ルールグループをサブスクライブおよびサブスクライブ解除できます。

> **Important**
>
> AWS Firewall Manager ポリシーで AWS Marketplace ルールグループを使用するには、まず組織内の各アカウントがそのルールグループにサブスクライブする必要があります。

AWS Marketplace マネージドルールグループにサブスクライブするには

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/wafv2/ で AWS WAF コンソールを開きます。
2. ナビゲーションペインで、[AWS Marketplace] を選択します。
3. [Available marketplace products] (利用可能な Marketplace 製品) セクションで、ルールグループの名前を選択して、詳細と料金情報を表示します。
4. ルールグループにサブスクライブする場合は、[Continue] (続行) を選択します。

   > **Note**
   >
   > このルールグループをサブスクライブしたくない場合は、ブラウザでこのページを閉じるだけです。

5. [Set up your account] (アカウントをセットアップ) を選択します。
6. 個々のルールを追加する場合と同様に、ルールグループをウェブ ACL に追加します。詳細については、「ウェブ ACL の作成」または「ウェブ ACL の編集」を参照してください。

> **Note**
>
> ルールグループをウェブ ACL に追加する場合、ルールグループ内のルールおよびルールグループの結果のアクションを上書きできます。詳細については、「ルールグループのアクションオーバーライドオプション」を参照してください。

AWS Marketplace ルールグループをサブスクライブしたら、他のマネージドルールグループと同様に、ウェブ ACLs でルールグループを使用します。詳細については、「ウェブ ACL の作成」を参照してください。

AWS Marketplace マネージドルールグループからのサブスクリプションの解除

コンソールで AWS Marketplace AWS WAF ルールグループのサブスクリプションを解除できます。

> **Important**
>
> AWS Marketplace マネージドルールグループのサブスクリプション料金を停止するには、サブスクリプションを解除するだけでなく、Firewall Manager AWS WAF ポリシー AWS WAF のすべてのウェブ ACLs から削除する必要があります。 AWS Marketplace マネージドルールグループのサブスクリプションを解除しても、ウェブ ACLs から削除しない場合は、サブスクリプションに対して引き続き課金されます。

AWS Marketplace マネージドルールグループのサブスクリプションを解除するには

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/wafv2/ で AWS WAF コンソールを開きます。
2. すべてのウェブ ACL からルールグループを削除します。詳細については、「ウェブ ACL の編集」を参照してください。
3. ナビゲーションペインで、[AWS Marketplace] を選択します。
4. [Manage your subscriptions] (サブスクリプションを管理) を選択します。
5. サブスクリプションを解除するルールグループの名前の横にある [Cancel subscription] (サブスクリプションをキャンセル) を選択します。
6. [Yes, cancel subscription] (はい、サブスクリプションをキャンセルします) を選択します。

AWS Marketplace ルールグループのトラブルシューティング

AWS Marketplace ルールグループが正当なトラフィックをブロックしていることがわかった場合は、次の手順を実行して問題をトラブルシューティングできます。

AWS Marketplace ルールグループをトラブルシューティングするには

1. アクションをオーバーライドして、正当なトラフィックをブロックしているルールをカウントします。 AWS WAF サンプリングされたリクエストまたは AWS WAF ログを使用して、特定のリクエストをブロックしているルールを特定できます。ログの `ruleGroupId` フィールドまたはサンプリングされたリクエストの `RuleWithinRuleGroup` フィールドを調べることによって、ルールを識別できます。パターン `<Seller Name>#<RuleGroup Name>#<Rule Name>` 内のルールを識別できます。
2. リクエストのみをカウントするように特定のルールを設定しても問題が解決しない場合は、すべてのルールアクションを上書きするか、 AWS Marketplace ルールグループ自体のアクションを「上書きなし」から「カウントに上書き」に変更できます。これにより、ルールグループ内の個々のルールアクションに関係なく、ウェブリクエストが通過します。
3. 個々のルールアクションまたは AWS Marketplace ルールグループアクション全体を上書きしたら、ルールグループプロバイダーのカスタマーサポートチームに連絡して、問題のトラブルシューティングをさらに行ってください。連絡先については、「 AWS Marketplace」の製品リストページのルールグループリストを参照してください。

AWS サポートへのお問い合わせ

AWS WAF または によって管理されるルールグループに関する問題については AWS、 にお問い合わせください AWS Support。 AWS Marketplace 販売者が管理するルールグループに関する問題については、プロバイダーのカスタマーサポートチームにお問い合わせください。連絡先情報を確認するには、「」のプロバイダーのリストを参照してください AWS Marketplace。

独自のルールグループの管理

独自のルールグループを作成して、マネージドルールグループサービスに見つからないルールのコレクションや、自分で処理したいルールのコレクションを再利用できます。

作成したルールグループは、ウェブ ACL と同じようにルールを保持し、ウェブ ACL と同じようにルールグループにルールを追加します。独自のルールグループを作成する場合は、そのルールにイミュータブルな最大容量を設定する必要があります。

トピック

- ルールグループの作成
- ルールグループの編集
- ウェブ ACL でのルールグループの使用
- 別のアカウントとのルールグループの共有
- ルールグループの削除

ルールグループの作成

新しいルールグループを作成するには、このページの手順に従います。

ルールグループを作成するには

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/wafv2/ で AWS WAF コンソールを開きます。
2. ナビゲーションペインで、[Rule groups] (ルールグループ)、[Create rule group] (ルールグループの作成) の順に選択します。
3. ルールの名前と説明を入力します。これらを使用して、ルールセットを識別して管理し、使用します。

   `AWS`、`Shield`、`PreFM`、または `PostFM` で始まる名前は使用しないでください。これらの文字列は、予約されているか、他のサービスが管理するルールグループと混同される可能性があります。他のサービスによって提供されるルールグループ を参照してください。

   > **Note**
   >
   > ウェブ ACL の作成後は、名前を変更できません。

4. [Region] (リージョン) で、ルールグループを保存するリージョンを選択します。Amazon CloudFront ディストリビューションを保護するウェブ ACLs でルールグループを使用するには、 グローバル設定を使用する必要があります。リージョン別アプリケーションにもグローバル設定を使用できます。
5. [Next] (次へ) を選択します。
6. ウェブ ACL 管理の場合と同様に、[Rule builder (ルールビルダー)] ウィザードを使用してルールグループにルールを追加します。唯一の違いは、ルールグループを別のルールグループに追加できないことです。

7. [Capacity] (容量) で、ルールグループによるウェブ ACL 容量ユニット (WCU) の使用の最大値を設定します。この設定はイミュータブルです。WCU の詳細については、「AWS WAF ウェブ ACL キャパシティーユニット (WCUs)」を参照してください。

   ルールグループにルールを追加すると、[Add rules and set capacity] (ルールの追加と容量の設定) ペインに、追加済みのルールに基づいて、必要な最小容量が表示されます。これとルールグループの将来の計画を使用して、ルールグループに必要な容量を見積もることができます。

8. ルールグループの設定を確認し、[Create] (作成) を選択します。

ルールグループの編集

ルールグループを追加、削除、あるいは設定を変更するには、このページの手順を使用してルールグループにアクセスします。

> ⚠ 本番稼働トラフィックのリスク
>
> ウェブ ACL で現在使用しているルールグループを変更すると、その変更は、使用されている場所に関係なくウェブ ACL の動作に影響します。トラフィックへの潜在的な影響に納得がいくまで、すべての変更をステージング環境またはテスト環境でテストし、調整するようにしてください。その後、更新したルールを有効にする前に、本番稼働用トラフィックでカウントモードでルールをテストしてチューニングします。ガイダンスについては、「AWS WAF 保護機能のテストと調整」を参照してください。

ルールグループを編集するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF でコンソールを開きます。
2. ナビゲーションペインで、[Rule groups] (ルールグループ) を選択します。
3. 編集するルールグループ名を選択します。コンソールにルールグループのページが表示されます。
4. 必要に応じてルールグループを編集します。ルールグループの変更可能なプロパティは、作成時と同様に編集できます。変更内容は、実行中にコンソールに保存されます。

> **Note**
>
> ルールの名前を変更し、その変更をルールのメトリック名に反映させたい場合は、メトリック名も更新する必要があります。 AWS WAF ルール名を変更しても、ルールのメトリック名は自動的に更新されません。ルールの JSON エディターを使用して、コンソールでルールを編集するときに、メトリック名を変更できます。API や、ウェブ ACL またはルールグループの定義に使用する JSON リストを使用して、両方の名前を変更することもできます。

更新中の一時的な不一致

ウェブ ACL AWS WAF やその他のリソースを作成または変更した場合、その変更がリソースが保存されているすべての領域に反映されるまでに少し時間がかかります。伝播時間は、数秒から数分までかかります。

次の内容では、変更伝播中に直面する一時的な不整合性の例を紹介します。

- ウェブ ACL を作成した後、それをリソースに関連付けようとすると、ウェブ ACL が利用できないことを示す例外が表示される場合があります。
- ルールグループをウェブ ACL に追加した後、新しいルールグループのルールは、ウェブ ACL が使用される一部のエリアで有効になり、別のエリアでは有効にならない場合があります。
- ルールのアクション設定を変更した後、古いアクションを一部のエリアで確認され、新しいアクションを別のエリアで確認される場合があります。
- ブロックルールで使用されている IP セットに IP アドレスを追加した後、新しいアドレスはあるエリアではブロックされ、別のエリアでは許可される場合があります。

ウェブ ACL でのルールグループの使用

ウェブ ACL でルールグループを使用するには、ルールグループリファレンスステートメントのウェブ ACL にルールグループを追加します。

> **本番稼働トラフィックのリスク**
>
> 本番稼働トラフィックのウェブ ACL に変更をデプロイする前に、ステージング環境またはテスト環境でテストおよびチューニングしてトラフィックへの潜在的な影響を確認します。そ

の後、更新したルールを有効にする前に、本番稼働用トラフィックでカウントモードでルールをテストしてチューニングします。ガイダンスについては、「AWS WAF 保護機能のテストと調整」を参照してください。

> **Note**
>
> ウェブ ACL で 1,500 WCU を超える容量を使用すると、ウェブ ACL の基本料金を超えるコストが発生します。詳細については、「AWS WAF ウェブ ACL キャパシティーユニット (WCUs)」と「AWS WAF 料金表」を参照してください。

コンソールで、ウェブ ACL のルールを追加または更新するときに、[Add rules and rule groups] (ルールとルールグループの追加) ページで、[Add rules] (ルールの追加) を選択し、[Add my own rules and rule groups] (独自のルールとルールグループの追加) を選択します。その後、[Rule group] (ルールグループ) を選択し、リストからルールグループを選択します。

ウェブ ACL では、個々のルールアクションが Count またはその他のアクションを起こすように設定することで、ルールグループおよびそのルールの動作を変更できます。これは、ルールグループのテスト、ルールグループ内のルールからの誤検出の特定、マネージドルールグループによるリクエストの処理方法のカスタマイズなどを行うのに役立ちます。詳細については、「ルールグループのアクションオーバーライドオプション」を参照してください。

ルールグループにレートベースのステートメントが含まれている場合、ルールグループを使用する各ウェブ ACL は、ルールグループを使用する他のウェブ ACL とは無関係に、レートベースのルールについて独自のレートトラッキングと管理を行います。詳細については、「レートベースのルールステートメント」を参照してください。

更新中の一時的な不一致

ウェブ ACL AWS WAF やその他のリソースを作成または変更すると、その変更がリソースが保存されているすべての領域に反映されるまでに少し時間がかかります。伝播時間は、数秒から数分までかかります。

次の内容では、変更伝播中に直面する一時的な不整合性の例を紹介します。

- ウェブ ACL を作成した後、それをリソースに関連付けようとすると、ウェブ ACL が利用できないことを示す例外が表示される場合があります。

- ルールグループをウェブ ACL に追加した後、新しいルールグループのルールは、ウェブ ACL が使用されるエリアで有効になり、別のエリアでは有効にならない場合があります。
- ルールのアクション設定を変更した後、古いアクションを一部のエリアで確認され、新しいアクションを別のエリアで確認される場合があります。
- ブロックルールで使用されている IP セットに IP アドレスを追加した後、新しいアドレスはあるエリアではブロックされ、別のエリアでは許可される場合があります。

別のアカウントとのルールグループの共有

ルールグループを他のアカウントと共有して、それらのアカウントで使用できます。1 つ以上の特定のアカウントと共有でき、組織内のすべてのアカウントと共有できます。

これを行うには、 AWS WAF API を使用して、必要なルールグループ共有のポリシーを作成します。詳細については、 API リファレンスPutPermissionPolicyの AWS WAF 「」を参照してください。

ルールグループの削除

ルールグループを削除するには、このセクションのガイダンスに従います。

参照セットとルールグループの削除

IP セット、正規表現パターンセット、ルールグループなど、ウェブ ACL で使用できるエンティティを削除すると、 はエンティティがウェブ ACL で現在使用されている AWS WAF かどうかを確認します。使用中であることがわかった場合、 は AWS WAF ユーザーに警告します。 AWS WAF は、ほとんどの場合、エンティティがウェブ ACL によって参照されているかどうかを判断できます。ただし、まれに判別できないことがあります。エンティティが現在使用中でないことを確認する必要がある場合は、削除する前にウェブ ACL でそのエンティティを確認してください。エンティティが参照されているセットである場合は、ルールグループでエンティティが使用されていないことも確認してください。

ルールグループを削除するには

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/wafv2/ で AWS WAF コンソールを開きます。
2. ナビゲーションペインで、[Rule groups] (ルールグループ) を選択します。
3. 削除するスナップショットを選択し、[Delete] (削除) を選択します。

他のサービスによって提供されるルールグループ

AWS Firewall Manager AWS Shield Advanced ユーザーまたは組織の管理者がを使用してリソース保護を使用または管理している場合 AWS WAF、アカウントのウェブ ACL にルールグループ参照ステートメントが追加されていることがあります。

これらのルールグループの名前は、次の文字列で始まります。

- **ShieldMitigationRuleGroup**— これらのルールグループは、保護対象のアプリケーション層 (レイヤー 7) リソースに対するアプリケーション層の DDoS AWS Shield Advanced 対策を自動的に行うために管理され、使用されます。

  保護されたリソースでアプリケーションレイヤー DDoS 自動緩和を有効にすると、Shield Advanced は、リソースに関連付けたウェブ ACL に、これらのルールグループの 1 つを追加します。Shield Advanced は、ルールグループ参照ステートメントに優先順位の設定として 10,000,000 を割り当て、ユーザーがウェブ ACL で設定したルールの後に実行されるようにします。これらのルールタイプの詳細については、「Shield Advanced アプリケーションレイヤー DDoS 自動緩和」を参照してください。

  > **Warning**
  >
  > ウェブ ACL 内のこのルールグループを手動で管理しないでください。特に、ShieldMitigationRuleGroup ルールグループ参照ステートメントをウェブ ACL から手動で削除しないでください。これにより、ウェブ ACL に関連付けられているすべてのリソースに意図しない結果が生じていた可能性もあります。代わりに、Shield Advanced を使用して、ウェブ ACL に関連付けられているリソースの自動緩和を無効にします。Shield Advanced は、ルールグループが自動緩和に必要でない場合に、ユーザーに代わって削除します。

- **PREFMManaged**そして **POSTFMManaged** — これらのルールグループはによって管理されます。AWS Firewall Manager Firewall Manager は、Firewall Manager が作成および管理するウェブ ACL 内にそれらを提供します。ウェブ ACL の名前は FMManagedWebACLV2 で始まります。これらのウェブ ACL およびルールグループの詳細については、「AWS WAF ポリシー」を参照してください。

AWS WAF 規則

AWS WAF ルールは、HTTP (S) Web リクエストを検査する方法と、リクエストが検査基準に一致したときに実行するアクションを定義します。ルールは、ルールグループまたはウェブ ACL のコンテキストでのみ定義されます。

AWS WAF ルールはそれ自体では存在しません。 AWS これらはリソースではなく、Amazon リソースネーム (ARN) もありません。ルールが定義されているルールグループまたはウェブ ACL に含まれるルールにアクセスするには、名前を使用します。ルールを管理し、他のウェブ ACL にコピーするには、そのルールが含まれているルールグループまたはウェブ ACL の JSON ビューを使用します。また、ウェブ ACL AWS WAF とルールグループで使用できるコンソールルールビルダーを使用して管理することもできます。

ルール名

各ルールには名前が必要です。AWS で始まる名前や、他のサービスによって管理されているルールグループまたはルールに使用されている名前は避けてください。他のサービスによって提供されるルールグループ を参照してください。

> **Note**
>
> ルールの名前を変更し、その変更をルールのメトリック名に反映させたい場合は、メトリック名も更新する必要があります。 AWS WAF ルール名を変更しても、ルールのメトリック名は自動的に更新されません。ルールの JSON エディターを使用して、コンソールでルールを編集するときに、メトリック名を変更できます。API や、ウェブ ACL またはルールグループの定義に使用する JSON リストを使用して、両方の名前を変更することもできます。

ルールステートメント

各ルールには、ルールがウェブリクエストを検査する方法を定義するルールステートメントも必要です。ルールステートメントには、ルールとステートメントのタイプに応じて、ネストされたステートメントを任意の深さに含めることができます。一部のルールステートメントは、条件のセットを採用します。例えば、IP セット一致ルールに最大 10,000 個の IP アドレスまたは IP アドレス範囲を指定できます。

次のような基準を検査するルールを定義できます。

- 悪意のある可能性が高いスクリプト。攻撃者は、ウェブアプリケーションの脆弱性を悪用できるスクリプトを埋め込みます。これはクロスサイトスクリプティング (XSS) と呼ばれます。

- リクエストの発生元の IP アドレスまたはアドレス範囲。
- リクエスト送信元の国または地理的場所。
- クエリ文字列など、リクエストの指定した部分の長さ。
- 悪意のある可能性が高い SQL コード。攻撃者は、ウェブリクエストに悪意のある SQL コードを埋め込むことで、データベースからデータを抽出しようとします。これは SQL インジェクションと呼ばれます。
- リクエストに表示される文字列。例えば、User-Agent ヘッダーに表示される値、またはクエリ文字列に表示されるテキスト文字列です。正規表現を使用してこれらの文字列を指定することもできます。
- ウェブ ACL の以前のルールがリクエストに追加したラベル。

前述のリストにあるような Web リクエスト検査基準のあるステートメントに加えて、、、 AWS WAF の論理ステートメントもサポートしています。これらのステートメントはANDOR、ルール内のステートメントを組み合わせるために使用します。NOT

例えば、攻撃者からの最近のリクエストに基づいて、次のネストされたステートメントを組み合わせた論理 AND ステートメントを使用してルールを作成できます。

- リクエストが 192.0.2.44 から発生した。
- リクエストの User-Agent ヘッダーに BadBot 値が含まれる。
- それらのクエリ文字列には、SQL などのコードが含まれる。

この場合、上位のレベルの AND に一致するようにするには、ウェブリクエストはすべてのステートメントに一致する必要があります。

トピック

- ルールアクション
- ルールステートメントの基本
- 一致ルールステートメント
- 論理ルールステートメント
- レートベースのルールステートメント
- ルールグループのルールステートメント

ルールアクション

ルールアクションは、ウェブリクエストがルールで定義された条件に一致する場合に AWS WAF 、ウェブリクエストの処理方法を指示します。オプションで、各ルールアクションにカスタム動作を追加できます。

> **Note**
>
> ルールアクションは、終了アクションまたは非終了アクションである場合があります。終了アクションは、リクエストのウェブ ACL 評価を停止し、保護されたアプリケーションへのリクエストの継続を許可またはブロックします。

ルールアクションのオプションは以下のとおりです。

- Allow – AWS WAF リクエストを保護された AWS リソースに転送して処理と応答を許可します。これは終了アクションです。定義したルールでは、リクエストを保護されたリソースに転送する前に、カスタムヘッダーを挿入できます。
- Block – リクエストを AWS WAF ブロックします。これは終了アクションです。デフォルトでは、保護された AWS リソースは HTTP `403 (Forbidden)`ステータスコードで応答します。定義したルールでは、応答をカスタマイズできます。がリクエストを AWS WAF ブロックすると、Blockアクション設定によって、保護されたリソースがクライアントに送り返すレスポンスが決まります。
- Count – リクエストを AWS WAF カウントしますが、許可するかブロックするかは決定しません。これは非終了アクションです。 AWS WAF がウェブ ACL の残りのルールの処理を継続します。定義したルールでは、リクエストにカスタムヘッダーを挿入し、他のルールで一致するラベルを追加できます。
- CAPTCHA および Challenge — CAPTCHA パズルとサイレントチャレンジ AWS WAF を使用して、リクエストがボットから送信されていないことを確認し、トークン AWS WAF を使用して最近成功したクライアントレスポンスを追跡します。

  CAPTCHA パズルとサイレントチャレンジは、ブラウザが HTTPS エンドポイントにアクセスしている場合にのみ実行できます。トークンを取得するには、ブラウザクライアントが安全なコンテキストで実行されている必要があります。

> **Note**
>
> CAPTCHA または Challenge ルールアクションを 1 つのルールで使用、あるいはルールグループでルールアクションのオーバーライドとして使用すると、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

これらのルールアクションは、リクエスト内のトークンの状態に応じて、終了アクションまたは非終了アクションである場合があります。

- 有効で有効期限が切れていないトークンの非終了 – トークンが有効で、設定された CAPTCHA またはチャレンジイミュニティ時間に従って有効期限が切れていない場合、 は Count action. AWS WAF continues のようなリクエスト AWS WAF を処理し、ウェブ ACL の残りのルールに基づいてウェブリクエストを検査します。Count 設定と同様に、定義したルールでは、リクエストに挿入するカスタムヘッダーを使用してこれらのアクションを設定したり (オプション)、他のルールが照合できるラベルを追加したりできます。
- 無効または期限切れのトークンのリクエストがブロックされた状態で終了する – トークンが無効であるか、指定されたタイムスタンプの有効期限が切れている場合、 は Blockアクションと同様にウェブリクエストの検査 AWS WAF を終了し、リクエストをブロックします。 AWS WAF その後、 はカスタムレスポンスコードでクライアントに応答します。の場合CAPTCHA、リクエストの内容がクライアントブラウザが処理できることを示している場合、 AWS WAF は人間のクライアントをボットと区別するように設計された JavaScript インタースティシャルで CAPTCHA パズルを送信します。Challenge アクションの場合、 は、通常のブラウザをボットによって実行されているセッションと区別するように設計されたサイレントチャレンジで JavaScript インタースティシャル AWS WAF を送信します。

詳細については、「CAPTCHAChallengeの および AWS WAF」を参照してください。

リクエストとレスポンスをカスタマイズする方法については、「AWS WAFのカスタマイズされたウェブリクエストとレスポンス」を参照してください。

一致するリクエストへのラベルの追加については、「AWS WAF ウェブリクエストの ラベル」を参照してください。

ウェブ ACL とルール設定の相互作用の詳細については、「ウェブ ACL ルールおよびルールグループの評価」を参照してください。

ルールステートメントの基本

ルールステートメントは、ウェブリクエストの検査 AWS WAF 方法を に指示するルールの一部です。がウェブリクエストで検査基準 AWS WAF を見つけると、ウェブリクエストは ステートメントと一致すると言います。すべてのルールステートメントは、ステートメントのタイプに応じて、何をどのように検索するかを指定します。

のすべてのルール AWS WAF には、他のステートメントを含めることができる 1 つの最上位ルールステートメントがあります。ルールステートメントは非常にシンプルにすることができます。例えば、ウェブリクエストを検査するための送信国のセットを提供するステートメントを作成したり、ウェブ ACL でルールグループを参照するだけのルールステートメントを保持したりできます。ルールステートメントはまた、非常に複雑にすることもできます。例えば、他の多くのステートメントを論理 AND、OR、および NOT ステートメントと組み合わせたステートメントを作成できます。

ほとんどのルールでは、一致するリクエストにカスタム AWS WAF ラベルを追加できます。 AWS マネージドルールのルールグループのルールは、一致するリクエストにラベルを追加します。ルールが追加するラベルは、ウェブ ACL の後で評価されるルール、および AWS WAF ログとメトリクスにリクエストに関する情報を提供します。ラベル付けの詳細については、AWS WAF ウェブリクエストの ラベル「」および「」を参照してくださいラベル一致ルールステートメント。

ルールステートメントのネスト

AWS WAF は、多くのルールステートメントのネストをサポートしていますが、すべてではありません。例えば、ルールグループステートメントを別のステートメント内にネストすることはできません。スコープダウンステートメントや論理ステートメントなど、一部のシナリオではネストを使用する必要があります。ルールステートメントのリストとそれに続くルールの詳細では、各カテゴリとルールのネスト機能と要件について説明されています。

コンソール内のルールのビジュアルエディタでは、ルールステートメントの 1 レベルのネストしかサポートされません。例えば、論理 AND または OR ルール内に多くのタイプのステートメントをネストできますが、他の AND または OR ルールをネストすることはできません。2 レベルのネストが必要になるからです。複数のレベルのネストを実装するには、コンソールの JSON ルールエディタまたは API を使用して、JSON でルール定義を指定します。

トピック

- ウェブリクエストコンポーネントの仕様と処理
- スコープダウンステートメント
- セットまたはルールグループを参照するステートメント

ウェブリクエストコンポーネントの仕様と処理

このセクションでは、ウェブリクエストのコンポーネントを検査するルールステートメントで指定できる設定について説明します。使用の詳細については、「一致ルールステートメント」で個別のルールステートメントを参照してください。

これらのウェブリクエストコンポーネントのサブセットは、カスタムリクエスト集約キーとしてレートベースのルールでも使用できます。詳細については、「レートベースのルール集約オプションとキー」を参照してください。

リクエストコンポーネントの設定では、コンポーネントタイプ自体と、コンポーネントタイプに応じて追加のオプションを指定します。例えば、テキストを含むコンポーネントタイプを検査する場合、検査する前にテキスト変換を適用できます。

> **Note**
>
> 特に明記されていない限り、ウェブリクエストにルールステートメントで指定されたリクエストコンポーネントがない場合、 はリクエストをルール基準に一致しないものとして AWS WAF 評価します。

目次

- 転送された IP アドレス
- HTTP/2 擬似ヘッダーを検査するためのオプション
- テキスト変換オプション

リクエストコンポーネントオプション

このセクションでは、検査のために指定できるウェブリクエストのコンポーネントについて説明します。ウェブリクエスト内のパターンを検索する一致ルールステートメントのリクエストコンポーネントを指定します。これらのステートメントのタイプには、文字列一致、正規表現一致、サイズ制約、SQL インジェクション攻撃などのステートメントがあります。リクエストコンポーネント設定の使用方法については、「一致ルールステートメント」で個々のルールステートメントを参照してください。

特に明記されていない限り、ウェブリクエストにルールステートメントで指定されたリクエストコンポーネントがない場合、 はリクエストをルール基準に一致しないものとして AWS WAF 評価します。

> **Note**
>
> リクエストコンポーネントは、それを必要とするルールステートメントごとに 1 つずつ指定します。リクエストの複数のコンポーネントを検査するには、コンポーネントごとにルールステートメントを作成します。

AWS WAF コンソールと API のドキュメントには、以下の場所にあるリクエストコンポーネント設定に関するガイダンスが記載されています。

- コンソールのルールビルダー – 通常のルールタイプの [Statement] (ステートメント) 設定で、[Request components] (コンポーネントをリクエスト) の下の [Inspect] (検査) ダイアログで検査するコンポーネントを選択します。
- API ステートメントのコンテンツ – `FieldToMatch`

このセクションの残りの部分では、ウェブリクエストの検査対象部分のオプションについて説明します。

トピック

- HTTP メソッド

- 単一ヘッダー
- すべてのヘッダー
- ヘッダーの順序
- cookie
- URI パス
- JA3 フィンガープリント
- クエリ文字列
- Single query parameter (単一クエリパラメータ)
- All query parameters (すべてのクエリパラメータ)
- [Body] (本文)
- JSON 本文

HTTP メソッド

リクエストの HTTP メソッドが検査されます。HTTP メソッドは、ウェブリクエストが保護対象リソースに対して実行を求めている操作のタイプ (POST または GET など) を示しています。

単一ヘッダー

リクエスト内の単一の名前付きヘッダーが検査されます。

このオプションでは、User-Agent や Referer などのヘッダー名を指定します。名前と一致する文字列は、大文字と小文字を区別しません。

すべてのヘッダー

すべてのリクエストヘッダー (cookie を含む) を検査します。フィルターを適用して、すべてのヘッダーのサブセットを検査できます。

このオプションでは、次の仕様を指定します。

- 一致パターン – inspection 用のヘッダーのサブセットを取得するために使用するフィルター。は、ヘッダーキーでこれらのパターン AWS WAF を探します。

  一致パターン設定は、次のいずれかになります。

  - [All] (すべて) – すべてのキーに一致します。すべてのヘッダーのルール検査基準を評価します。

  - [Excluded headers] (除外されるヘッダー) – ここで指定した文字列のいずれとも一致しないキーを持つヘッダーのみを検査します。キーと一致する文字列は大文字と小文字に区別されません。
  - [Included headers] (含まれるヘッダー) – ここで指定した文字列のいずれかに一致するキーを持つヘッダーのみを検査します。キーと一致する文字列は大文字と小文字に区別されません。
- 一致範囲 – がルール検査基準で検査 AWS WAF するヘッダーの部分。[キー]、[値]、または [すべて] を指定して、キーと値の両方で一致するものがあるかどうかを検査することができます。

  [すべて] では、キーで一致するもの、および値で一致するものを見つける必要はありません。キー、値、またはその両方で一致するものを見つける必要があります。キーと値で一致するものを見つけるようにするには、論理 AND ステートメントを使用して、キーを検査する一致ルールと値を検査する一致ルールの 2 つを組み合わせます。
- オーバーサイズ処理 — が検査 AWS WAF できるよりも大きいヘッダーデータを持つリクエストを が処理 AWS WAF する方法。 は、リクエストヘッダーの最初の 8 KB (8,192 バイト) まで、および最初の 200 個のヘッダーまで検査 AWS WAF できます。コンテンツは、最初の制限に達する AWS WAF まで検査できます。検査を続行するか、検査をスキップするかを選択できます。検査をスキップする場合、リクエストがルールに一致するとマークするか一致しないとマークするかを選択できます。オーバーサイズコンテンツの処理の詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。

ヘッダーの順序

が検査のために AWS WAF 受け取るウェブリクエストに表示される順序で、リクエストのヘッダー名のリストを含む文字列を検査します。 AWS WAF は文字列を生成し、それをフィールドとして使用して、検査のコンポーネントを照合します。 は、文字列内のヘッダー名をコロンで AWS WAF 区切り、スペースを追加しません。例えば、 ですhost:user-agent:accept:authorization:referer。

このオプションでは、次の仕様を指定します。

- オーバーサイズ処理 — が検査 AWS WAF できる数よりも多い、または大きいヘッダーデータを持つリクエストを が処理 AWS WAF する方法。 は、リクエストヘッダーの最初の 8 KB (8,192 バイト) まで、および最初の 200 個のヘッダーまで検査 AWS WAF できます。コンテンツは、最初の制限に達する AWS WAF まで検査できます。使用可能なヘッダーの検査を続行するか、検査をスキップするかを選択できます。検査をスキップする場合、リクエストがルールに一致するか一致しないかをマークします。オーバーサイズコンテンツの処理の詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。

cookie

すべてのリクエスト cookie を検査します。フィルターを適用して、すべての cookie のサブセットを検査できます。

このオプションでは、次の仕様を指定します。

- [Match patterns] (一致パターン) – 検査用の cookie のサブセットを取得するために使用するフィルター。 AWS WAF は、cookie キーでこれらのパターンを検索します。

  一致パターン設定は、次のいずれかになります。

  - [All] (すべて) – すべてのキーに一致します。すべての cookie のルール検査基準を評価します。
  - [Excluded cookies] (除外される cookie) – ここで指定した文字列のいずれとも一致しないキーを持つ cookie のみを検査します。キーの文字列一致は大文字と小文字が区別され、完全に一致する必要があります。
  - [Included cookies] (含まれる cookie) – ここで指定した文字列のいずれかに一致するキーを持つ cookie のみを検査します。キーの文字列一致は大文字と小文字が区別され、完全に一致する必要があります。
- 一致範囲 – がルール検査基準で検査 AWS WAF する必要がある Cookie の部分。キーと値の両方に、[Keys] (キー)、[Values] (値)、または [All] (すべて) を指定できます。

  [すべて] では、キーで一致するもの、および値で一致するものを見つける必要はありません。キー、値、またはその両方で一致するものを見つける必要があります。キーと値で一致するものを見つけるようにするには、論理 AND ステートメントを使用して、キーを検査する一致ルールと値を検査する一致ルールの 2 つを組み合わせます。
- オーバーサイズ処理 — が検査 AWS WAF できるサイズよりも大きい Cookie データを含むリクエストを が処理 AWS WAF する方法。 は、リクエスト Cookie の最初の 8 KB (8,192 バイト) まで、および最初の 200 個の Cookie まで検査 AWS WAF できます。コンテンツは、最初の制限に達する AWS WAF まで検査できます。検査を続行するか、検査をスキップするかを選択できます。検査をスキップする場合、リクエストがルールに一致するとマークするか一致しないとマークするかを選択できます。オーバーサイズコンテンツの処理の詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。

URI パス

URL 内でリソースを識別する部分 (`/images/daily-ad.jpg` など) が検査されます。詳細については、「Uniform Resource Identifier (URI): 一般的な構文」を参照してください。

このオプションでテキスト変換を使用しない場合、 は URI AWS WAF を正規化せず、リクエストでクライアントから受信したとおりに検査します。テキスト変換については、「テキスト変換オプション」を参照してください。

JA3 フィンガープリント

リクエストの JA3 フィンガープリントを検査します。

> **Note**
>
> JA3 フィンガープリント検査は、Amazon CloudFront ディストリビューションと Application Load Balancer でのみ使用できます。

JA3 フィンガープリントは、受信リクエストの TLS Client Hello から生成される 32 文字のハッシュです。このフィンガープリントは、クライアントの TLS 設定の一意の識別子として機能します。AWS WAF は、計算に十分な TLS Client Hello 情報を持つ各リクエストについて、このフィンガープリントを計算してログに記録します。この情報は、ほとんどすべてのウェブリクエストに含まれています。

クライアントの JA3 フィンガープリントを取得する方法

クライアントリクエストの JA3 フィンガープリントは、ウェブ ACL ログからを取得できます。AWS WAF がフィンガープリントを計算できる場合は、それをログに含めます。フィールドのログ記録については、「ログフィールド」を参照してください。

ルールステートメントの要件

JA3 フィンガープリントは、指定した文字列と完全に一致するように設定されている文字列一致ステートメント内のみで検査することができます。同じ TLS 設定を持つ将来のリクエストと一致させるために、文字列一致ステートメントの仕様のログから JA3 フィンガープリント文字列を指定します。文字列一致ルールステートメントの詳細については、「文字列一致ルールステートメント」を参照してください。

このルールステートメントにはフォールバック動作を指定する必要があります。フォールバック動作は、 が JA3 フィンガープリントを計算できない場合にウェブリクエスト AWS WAF AWS WAF に割り当てる一致ステータスです。一致を選択した場合、 AWS WAF はリクエストをルールステートメントに一致するものとして処理し、ルールアクションをリクエストに適用します。一致しないことを選択した場合、 はリクエストをルールステートメントと一致しないものとして AWS WAF 処理します。

この一致オプションを使用するには、ウェブ ACL トラフィックをログに記録する必要があります。詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。

クエリ文字列

URL 内で ? 文字の後に続く部分 (ある場合) が検査されます。

> **Note**
>
> クロスサイトスクリプティングの一致ステートメントについては、[Query string] (クエリ文字列) ではなく、[All query parameters] (すべてのクエリパラメータ) を選択することをお勧めします。[All query parameters] (すべてのクエリパラメータ) を選択すると、基本コストに 10 WCU が追加されます。

Single query parameter (単一クエリパラメータ)

クエリ文字列の一部として定義した単一のクエリパラメータを検査します。指定したパラメータの値を AWS WAF 検査します。

このオプションでは、[Query argument] (クエリ引数) も指定します。例えば、URL が `www.xyz.com?UserName=abc&SalesRegion=seattle` である場合は、クエリ引数として `UserName` または `SalesRegion` を指定できます。引数の名前は最大 30 文字です。名前では大文字と小文字が区別されないため、`UserName` と指定すると、 AWS WAF では `UserName` のすべてのバリエーション (`username`、`UsERName` など) と一致します。

クエリ文字列に、指定したクエリ引数の複数のインスタンスが含まれている場合、 はORロジックを使用して、一致のすべての値を AWS WAF 検査します。例えば、URL `www.xyz.com?SalesRegion=boston&SalesRegion=seattle` では、 AWS WAF は、指定された名前を `boston` および `seattle` に対して評価します。いずれかが一致する場合、検査結果は一致となります。

All query parameters (すべてのクエリパラメータ)

リクエスト内のすべてのクエリパラメータが検査されます。これは、単一のクエリパラメータコンポーネントの選択に似ていますが、クエリ文字列内のすべての引数の値を AWS WAF 検査します。例えば、URL が `www.xyz.com?UserName=abc&SalesRegion=seattle` である場合は、`UserName` または `SalesRegion` の値が検査基準に一致すると、 AWS WAF は一致をトリガーします。

このオプションを選択すると、基本コストに 10 WCU が追加されます。

[Body] (本文)

プレーンテキストとして評価されて、リクエストボディが検査されます。また、JSON コンテンツタイプを使用して、本文を JSON として評価することもできます。

リクエストボディは、リクエストの一部で、リクエストヘッダーの直後に続く部分です。これには、フォームからのデータなど、ウェブリクエストに必要な追加データが含まれます。

- コンソールで、[Content type] (コンテンツタイプ) の [Plain text] (プレーンテキスト) を選択して、[Request option] (リクエストオプション) の [Body] (本文) でこれを選択します。
- API では、ルールの `FieldToMatch` の指定で、リクエストボディをプレーンテキストとして検査するように `Body` を指定します。

Application Load Balancer と の場合 AWS AppSync、 はリクエストの本文の最初の 8 KB を検査 AWS WAF できます。 CloudFront、API Gateway、Amazon CognitoApp Runner、Verified Access の場合、デフォルトでは最初の 16 KB を検査 AWS WAF でき、ウェブ ACL 設定で制限を最大 64 KB まで増やすことができます。詳細については、「本文検査のサイズ制限の管理」を参照してください。

このコンポーネントタイプには、オーバーサイズの処理を指定する必要があります。オーバーサイズ処理は、 が検査 AWS WAF できるよりも大きい本文データを持つリクエストを が AWS WAF 処理する方法を定義します。検査を続行するか、検査をスキップするかを選択できます。検査をスキップする場合、リクエストがルールに一致するとマークするか一致しないとマークするかを選択できます。オーバーサイズコンテンツの処理の詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。

本文を解析された JSON として評価することもできます。これに関する詳細については、次のセクションを参照してください。

JSON 本文

JSON として評価されて、リクエストボディが検査されます。本文をプレーンテキストとして評価することもできます。

リクエストボディは、リクエストの一部で、リクエストヘッダーの直後に続く部分です。これには、フォームからのデータなど、ウェブリクエストに必要な追加データが含まれます。

- コンソールで、[Content type] (コンテンツタイプ) の [JSON] を選択して、[Request option] (リクエストオプション) の [Body] (本文) でこれを選択します。
- API で、ルールの `FieldToMatch` の指定で `JsonBody` を指定します。

Application Load Balancer と の場合 AWS AppSync、 はリクエストの本文の最初の 8 KB を検査 AWS WAF できます。 CloudFront、API Gateway、Amazon CognitoApp Runner、Verified Access の場合、デフォルトでは最初の 16 KB を検査 AWS WAF でき、ウェブ ACL 設定で制限を最大 64 KB まで増やすことができます。詳細については、「本文検査のサイズ制限の管理」を参照してください。

このコンポーネントタイプには、オーバーサイズの処理を指定する必要があります。オーバーサイズ処理は、 が検査 AWS WAF できるよりも大きい本文データを持つリクエストを が AWS WAF 処理する方法を定義します。検査を続行するか、検査をスキップするかを選択できます。検査をスキップする場合、リクエストがルールに一致するとマークするか一致しないとマークするかを選択できます。オーバーサイズコンテンツの処理の詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。

このオプションを選択すると、一致ステートメントの基本コスト WCU が 2 倍になります。例えば、一致ステートメントのベースコストが JSON 解析なしで 5 WCU の場合、JSON 解析を使用すると、コストが 10 WCU に倍増します。

JSON 本文検査のステップとオプション

は、ウェブリクエスト本文を JSON として AWS WAF 検査する場合、本文を解析し、検査のために JSON 要素を抽出するステップを実行します。以下に、このリクエストコンポーネントタイプのステップと追加の設定オプションを示します。

1. 本文の内容を解析する - ウェブリクエスト本文の内容を AWS WAF 解析して、インスペクション用の JSON 要素を抽出します。本文の内容を解析するには最善を AWS WAF 尽くしますが、コンテンツのさまざまなエラー状態で解析が失敗する可能性があります。例としては、無効な文字、重複するキー、切り捨て、ルートノードがオブジェクトまたは配列ではないコンテンツなどがあります。

   オプションの本文解析フォールバック動作 AWS WAF は、JSON 本文を完全に解析できなかった場合の動作を決定します。

   - なし (デフォルトの動作) - 解析エラーが発生した時点までのみコンテンツ AWS WAF を評価します。
   - 文字列として評価 - 本文をプレーンテキストとして検査します。 は、JSON 検査用に定義したテキスト変換と検査基準を本文テキスト文字列 AWS WAF に適用します。
   - 一致 - ウェブリクエストをルールステートメントに一致するものとして扱います。 AWS WAF は、ルールアクションをリクエストに適用します。
   - 一致なし - ウェブリクエストをルールステートメントと一致しないものとして処理します。

> **Note**
>
> このフォールバック動作は、 が JSON 文字列の解析中にエラー AWS WAF を検出したときにのみトリガーされます。

解析では JSON が完全に検証されない

AWS WAF 解析では入力 JSON 文字列が完全に検証されないため、無効な JSON であっても解析が成功する可能性があります。

例えば、 は、次の無効な JSON をエラーなしで AWS WAF 解析します。

- カンマ不足: `{"key1":"value1""key2":"value2"}`
- コロン不足: `{"key1":"value1","key2""value2"}`
- 余分なコロン: `{"key1"::"value1","key2""value2"}`

解析は成功したが、結果が完全に有効な JSON ではない場合など、評価の後続のステップの結果は異なる場合があります。抽出で一部の要素が欠落したり、ルール評価で予期しない結果が生じる可能性があります。アプリケーションで受信した JSON を検証し、必要に応じて無効な JSON を処理することをお勧めします。

2. JSON 要素を抽出する – 設定に従って検査する JSON 要素のサブセット AWS WAF を識別します。

   - オプション JSON 一致スコープは、 が検査 AWS WAF する JSON 内の要素のタイプを指定します。

     キーと値の両方に、[Keys] (キー)、[Values] (値)、または [All] (すべて) を指定できます。

     [すべて] では、キーで一致するもの、および値で一致するものを見つける必要はありません。キー、値、またはその両方で一致するものを見つける必要があります。キーと値で一致するものを見つけるようにするには、論理 AND ステートメントを使用して、キーを検査する一致ルールと値を検査する一致ルールの 2 つを組み合わせます。

   - 検査するコンテンツオプションは、 AWS WAF 検査するサブセットに要素セットをフィルタリングする方法を指定します。

     いずれかを指定する必要があります。

     - 完全な JSON コンテンツ - すべての要素を評価します。

- 含まれている要素のみ - 指定した JSON ポインター基準に一致するパスを持つ要素のみを評価します。このオプションを使用して JSON 内のすべてのパスを指定しないでください。代わりに、完全な JSON コンテンツ を使用します。

  JSON ポインタ構文の詳細については、「Internet Engineering Task Force (IETF)」ドキュメントJavaScript 「Object Notation (JSON) Pointer」を参照してください。

  例えば、コンソールで次の内容を指定できます。

  ```
  /dogs/0/name
  /dogs/1/name
  ```

  API または CLI では、次を指定できます。

  ```
  "IncludedPaths": ["/dogs/0/name", "/dogs/1/name"]
  ```

例えば、検査するコンテンツ設定が含められる要素のみ で、含められる要素設定が であるとします/a/b。

次の JSON 本文の例：

```
{
  "a":{
    "c":"d",
    "b":{
      "e":{
        "f":"g"
      }
    }
  }
}
```

が各 JSON 一致スコープ設定を検査 AWS WAF する要素セットを以下に示します。含まれbている要素パスの一部であるキー は評価されないことに注意してください。

- すべての：e、f,、および g。
- キー: eおよび f。
- 値：g。

3. JSON 要素セットの検査 – 抽出された JSON 要素に指定したテキスト変換 AWS WAF を適用し、結果の要素セットをルールステートメントの一致基準と照合します。これは、他のウェブリクエストコンポーネントと同じ変換および評価動作です。抽出された JSON 要素のいずれかが一致した場合、ウェブリクエストはルールと一致します。

転送された IP アドレス

このセクションは、ウェブリクエストの IP アドレスを使用するルールステートメントに適用されます。デフォルトでは、ウェブリクエストオリジンの IP AWS WAF アドレスを使用します。ただし、ウェブリクエストが 1 つ以上のプロキシまたはロードバランサーを通過する場合、ウェブリクエストの発信元には、クライアントの発信アドレスではなく、最後のプロキシのアドレスが含まれます。この場合、通常、発信元のクライアントアドレスは別の HTTP ヘッダーに転送されます。このヘッダーは通常 `X-Forwarded-For` (XFF) ですが、別のヘッダーにすることもできます。

IP アドレスを使用するルールステートメント

IP アドレスを使用するルールステートメントは次のとおりです。

- IP セット一致 - IP セットで定義されているアドレスと一致する IP アドレスを検査します。
- 地理的一致 - IP アドレスを使用して発信元の国と地域を特定し、それを国のリストと照合します。
- レートベースのルールステートメント - IP アドレスでリクエストを集約して、個々の IP アドレスがリクエストを過度に高いレートで送信しないようにできます。IP アドレスの集約は、単独で使用することも、他の集約キーと組み合わせて使用することもできます。

これらのルールステートメントには、ウェブリクエストのオリジンを使用する代わりに、`X-Forwarded-For`ヘッダーまたは別の HTTP ヘッダーから転送された IP AWS WAF アドレスを使用するように指示できます。仕様を指定する方法の詳細については、個別のルールステートメントタイプのガイダンスを参照してください。

> **Note**
>
> 指定したヘッダーがリクエストに含まれていない場合、 AWS WAF ルールはウェブリクエストにまったく適用されません。

フォールバック動作

転送された IP アドレスを使用するときに、指定された位置に有効な IP AWS WAF アドレスがない場合にウェブリクエストに割り当てるマッチステータスを指定します。

- 一致-ウェブリクエストをルールステートメントと一致するものとして扱います。 AWS WAF ルールアクションをリクエストに適用します。
- 一致なし - ウェブリクエストをルールステートメントと一致しないものとして処理します。

AWS WAF ボットコントロールで使用される IP アドレス

Bot Control が管理するルールグループは、からの IP アドレスを使用してボットを検証します。AWS WAF Bot Control を使用し、プロキシまたはロードバランサーを介してルーティングするボットを検証した場合は、カスタムルールを使用して明示的に許可する必要があります。例えば、転送された IP アドレスを使用して検証済みボットを検出および許可するカスタム IP セット一致ルールを設定できます。ルールを使用して、さまざまな方法でボット管理をカスタマイズできます。説明と例については、「AWS WAF ボットコントロール」を参照してください。

転送された IP アドレスの使用に関する一般的な考慮事項

転送された IP アドレスを使用する前に、次の一般的な注意事項に留意してください。

- ヘッダーは途中でプロキシによって変更でき、プロキシはヘッダーをさまざまな方法で処理することがあります。
- 攻撃者は、 AWS WAF 検査をバイパスしようとしてヘッダーの内容を変更する可能性があります。
- ヘッダー内の IP アドレスは、形式が正しくないか、無効である可能性があります。
- 指定したヘッダーは、リクエストにまったく存在しない可能性があります。

転送された IP アドレスをと一緒に使用する場合の考慮事項 AWS WAF

次のリストでは、で転送 IP アドレスを使用する際の要件と注意事項について説明します。 AWS WAF

- 単一のルールでは、転送された IP アドレス用に 1 つのヘッダーを指定できます。ヘッダーの仕様では、大文字と小文字は区別されません。
- レートベースのルールステートメントでは、ネストされたスコープステートメントは、転送された IP 設定を継承しません。転送された IP アドレスを使用する各ステートメントの設定を指定します。

- AWS WAF ジオマッチとレートベースのルールでは、ヘッダーの最初のアドレスを使用します。たとえば、ヘッダーに uses が含まれているとします。`10.1.1.1, 127.0.0.0, 10.10.10.10` AWS WAF `10.1.1.1`
- IP セットの一致では、ヘッダーの最初のアドレス、最後のアドレス、または任意のアドレスのいずれと照合するかを指定します。いずれかを指定すると、ヘッダー内のすべてのアドレスが一致するか、最大 10 AWS WAF 個のアドレスが検査されます。ヘッダーに 10 個を超えるアドレスが含まれている場合は、最後の 10 AWS WAF 個を検査します。
- 複数のアドレスを含むヘッダーでは、アドレスの間にカンマ区切り文字を使用する必要があります。リクエストでカンマ以外の区切り文字が使用されている場合、 AWS WAF はヘッダーの IP アドレスの形式が正しくないとみなします。
- ヘッダー内の IP アドレスが不正な形式であるか、または無効である場合、 AWS WAF は、転送された IP 設定で指定したフォールバック動作に従って、ウェブリクエストをルールに一致するか一致しないかを指定します。
- 指定したヘッダーがリクエストに含まれていない場合、 AWS WAF ルールはリクエストにまったく適用されません。つまり、 AWS WAF ルールアクションは適用されず、フォールバック動作も適用されません。
- IP アドレス用に転送された IP ヘッダーを使用するルールステートメントでは、ウェブリクエストの発信元によって報告された IP アドレスは使用されません。

転送 IP アドレスを使用する際のベストプラクティスは次のとおりです。 AWS WAF

転送された IP アドレスを使用する場合は、次のベストプラクティスを使用します。

- 転送された IP 設定を有効にする前に、リクエストヘッダーの可能な状態をすべて慎重に検討してください。目的の動作を実現するには、複数のルールを使用する必要がある場合があります。
- 複数の転送された IP ヘッダーを検査したり、ウェブリクエストの発信元と転送された IP ヘッダーを検査したりするには、IP アドレスのソースごとに 1 つのルールを使用します。
- 無効なヘッダーを持つウェブリクエストをブロックするには、ブロックするようにルールアクションを設定し、転送された IP 設定のフォールバック動作を一致するように設定します。

転送された IP アドレスの JSON の例

次の地理一致ステートメントは、発信元の国が US である IP が `X-Forwarded-For` ヘッダーに含まれている場合にのみ一致します。

```
{
```

```
  "Name": "XFFTestGeo",
  "Priority": 0,
  "Action": {
    "Block": {}
  },
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "XFFTestGeo"
  },
  "Statement": {
    "GeoMatchStatement": {
      "CountryCodes": [
        "US"
      ],
      "ForwardedIPConfig": {
        "HeaderName": "x-forwarded-for",
        "FallbackBehavior": "MATCH"
      }
    }
  }
}
```

次のレートベースのルールは、X-Forwarded-For ヘッダーの最初の IP に基づいてリクエストを集約します。ルールは、ネストされた地理一致ステートメントに一致するリクエストのみをカウントし、地理一致ステートメントに一致するリクエストのみをブロックします。ネストされた地理一致ステートメントは、X-Forwarded-For ヘッダーを使用して、IP アドレスが US の発信元の国を示しているかどうかも判断します。示している場合、またはヘッダーが存在していても形式が間違っている場合、地理一致ステートメントは一致を返します。

```
{
  "Name": "XFFTestRateGeo",
  "Priority": 0,
  "Action": {
    "Block": {}
  },
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "XFFTestRateGeo"
  },
  "Statement": {
```

```
    "RateBasedStatement": {
      "Limit": "100",
      "AggregateKeyType": "FORWARDED_IP",
      "ScopeDownStatement": {
        "GeoMatchStatement": {
          "CountryCodes": [
            "US"
          ],
          "ForwardedIPConfig": {
            "HeaderName": "x-forwarded-for",
            "FallbackBehavior": "MATCH"
          }
        }
      },
      "ForwardedIPConfig": {
        "HeaderName": "x-forwarded-for",
        "FallbackBehavior": "MATCH"
      }
    }
  }
}
```

HTTP/2 擬似ヘッダーを検査するためのオプション

HTTP/2 トラフィックをサポートする保護された AWS リソースは、検査 AWS WAF のために HTTP/2 疑似ヘッダーを に転送しませんが、 AWS WAF 検査するウェブリクエストコンポーネントに疑似ヘッダーの内容を提供します。

を使用して AWS WAF 、次の表に示す疑似ヘッダーのみを検査できます。

ウェブリクエストコンポーネントにマップされた HTTP/2 疑似ヘッダーの内容

HTTP/2 擬似ヘッダー	検査するウェブリクエストコンポーネント	ドキュメント
`:method`	HTTP メソッド	HTTP メソッド
`:authority`	Host ヘッダー	単一ヘッダー すべてのヘッダー
	URI パス	URI パス

HTTP/2 擬似ヘッダー	検査するウェブリクエストコンポーネント	ドキュメント
`:path` URI パス		
`:path` クエリ	クエリ文字列	クエリ文字列 Single query parameter (単一クエリパラメータ) All query parameters (すべてのクエリパラメータ)

テキスト変換オプション

パターンを検索したり、制約を設定したりするステートメントでは、リクエストを検査する前に が適用 AWS WAF する変換を指定できます。変換では、 AWS WAFをバイパスするために攻撃者が使用する異常なフォーマットの一部を削除するために、ウェブリクエストが再フォーマットされます。

これを JSON 本文リクエストコンポーネントの選択で使用する場合、 AWS WAF は JSON から検査する要素を解析および抽出した後、変換を適用します。詳細については、「JSON 本文」を参照してください。

複数の変換が指定された場合、 AWS WAF は変換の適用順序も設定します。

WCU - テキスト変換ごとには 10 個の WCU。

AWS WAF コンソールと API のドキュメントには、以下の場所でのこれらの設定に関するガイダンスも記載されています。

- コンソールのルールビルダー - [Text transformation] (テキスト変換)。このオプションは、リクエストコンポーネントの使用時に選択できます。
- API ステートメントのコンテンツ – `TextTransformations`

テキスト変換のオプション

各変換リストには、コンソールと API の仕様が表示され、その後に説明が続きます。

Base64 decode – `BASE64_DECODE`

AWS WAF は BaseBase64-encoded文字列をデコードします。

Base64 decode extension – `BASE64_DECODE_EXT`

AWS WAF は Base64-encodedされた文字列をデコードしますが、無効な文字を無視する寛容な実装を使用します。

Command line – `CMD_LINE`

このオプションは、攻撃者がオペレーティングシステムのコマンドラインコマンドを挿入し、異常な形式を使用してコマンドの一部またはすべてを偽装する状況を軽減します。

このオプションを使用して、次の変換を実行します。

- 次の文字を削除します: `\ " ' ^`
- 次の文字の前にあるスペースを削除します: `/ (`
- 次の文字をスペースに置き換えます: `, ;`
- 複数のスペースを 1 つのスペースに置き換えます。
- 大文字 `A-Z` を小文字 `a-z` に変換します。

Compress whitespace – `COMPRESS_WHITE_SPACE`

AWS WAF は、複数のスペースを 1 つのスペースに置き換え、次の文字をスペース文字 (ASCII 32) に置き換えることで空白を圧縮します。

- フォームフィード (ASCII 12)
- タブ (ASCII 9)
- 改行 (ASCII 10)
- キャリッジリターン (ASCII 13)
- 垂直タブ (ASCII 11)
- 改行なしスペース (ASCII 160)

CSS decode – `CSS_DECODE`

AWS WAF は、CSS 2.x エスケープルール を使用してエンコードされた文字をデコードしますsyndata.html#characters。この関数は、デコード処理で最大 2 バイトを使用するため、通常はエンコードされない CSS エンコーディングを使用してエンコードされた ASCII 文字を発見するのに役立ちます。また、バックスラッシュと 16 進数以外の文字の組み合わせである回避対策にも役立ちます。たとえば、`javascript` の `ja\vascript` を設定します。

Escape sequences decode – ESCAPE_SEQ_DECODE

AWS WAF は、次の ANSI C エスケープシーケンスをデコードします: \a、\b\f\n\r、\t、、、\v、\\、\?、、\xHH、(16 進数) \"、 \0OOO (8 \'進数) 。有効でないエンコーディングは出力に残ります。

Hex decode – HEX_DECODE

AWS WAF は 16 進数の文字の文字列をバイナリにデコードします。

HTML entity decode – HTML_ENTITY_DECODE

AWS WAF は、16 進形式&#xhhhh;または 10 進形式で表される文字を、対応する文字&#nnnn;に置き換えます。

AWS WAF は、次の HTML エンコード文字をエンコードされていない文字に置き換えます。このリストでは小文字の HTML エンコーディングを使用していますが、処理では大文字と小文字は区別されません。例えば、 &QuOt; と " は同じように扱われます。

HTML でエンコードされた文字	以下に置き換えます
"	"
&	&
<	<
>	>
または	改行なしスペース、10 進数 160
\n、10 進数 10
	\t、10 進数 9
{ または {	{
|, |, または		|
} または }	}
!	!

HTML でエンコードされた文字	以下に置き換えます
#	#
$	$
&percent; または %	%
'	\
(	(
)	)
* または *	*
+	+
,	,
.	.
/	/
:	:
;	;
=	=
?	?
˜ または ˜	~
−	-
[または [	[
\	\\
] または]	]

HTML でエンコードされた文字	以下に置き換えます
&hat;	^
_ または &underbar;	_
` または `	`

JS decode – JS_DECODE

AWS WAF はエス JavaScript ケープシーケンスをデコードします。\uHHHH コードが FF01-FF5E の全角 ASCII コード範囲内にある場合、高位バイトを使用して下位バイトが検出され、調整されます。そうでない場合は、下位バイトのみが使用され、上位バイトはゼロになり、情報が失われる可能性があります。

Lowercase – LOWERCASE

AWS WAF は、大文字 (A ~ Z) を小文字 (a ~ z) に変換します。

MD5 – MD5

AWS WAF は、入力内のデータから MD5 ハッシュを計算します。計算されたハッシュは生のバイナリ形式です。

None – NONE

AWS WAF は、テキスト変換なしで、受信したウェブリクエストを検査します。

Normalize path – NORMALIZE_PATH

AWS WAF は、入力の先頭にない複数のスラッシュ、ディレクトリの自己参照、およびディレクトリのバックリファレンスを削除することで、入力文字列を正規化します。

Normalize path Windows – NORMALIZE_PATH_WIN

AWS WAF はバックスラッシュ文字をスラッシュに変換し、NORMALIZE_PATH変換を使用して結果の文字列を処理します。

Remove nulls – REMOVE_NULLS

AWS WAF は入力からすべてのNULLバイトを削除します。

Replace comments – REPLACE_COMMENTS

AWS WAF は、C 形式のコメント (/* ... */) の各出現を 1 つのスペースに置き換えます。コメントが複数連続しているときは、圧縮しません。コメントの終端がないときもスペース (ASCII 0x20) に置き換えられます。コメントの終端 (*/) のみがあるときは変更されません。

Replace nulls – REPLACE_NULLS

AWS WAF は、入力の各バイトをスペース文字 (ASCII NULL 0x20) に置き換えます。

SQL hex decode – SQL_HEX_DECODE

AWS WAF は SQL 16 進数のデータをデコードします。例えば、 (0x414243) を () に AWS WAF デコードしますABC。

URL decode – URL_DECODE

AWS WAF は URL エンコードされた値をデコードします。

URL decode Unicode – URL_DECODE_UNI

URL_DECODE と同様ですが、Microsoft 固有の %u エンコーディングをサポートしています。コードが FF01-FF5E の全角 ASCII コード範囲内にある場合、高位バイトを使用して下位バイトが検出され、調整されます。それ以外の場合は、下位バイトのみが使用され、高位バイトはゼロになります。

UTF8 to Unicode – UTF8_TO_UNICODE

AWS WAF は、すべての UTF-8 文字シーケンスを Unicode に変換します。これにより、入力を正規化し、英語以外の言語の偽陽性と偽陰性を最小限に抑えることができます。

スコープダウンステートメント

スコープダウンステートメントは、マネージドルールグループステートメントまたはレートベースのステートメント内に追加できるネスト可能なルールステートメントで、包含ルールが評価するリクエストのセットを絞り込むことができます。包含ルールは、スコープダウンステートメントに最初に一致するリクエストのみを評価します。

- マネージドルールグループステートメント — マネージドルールグループステートメントにスコープダウンステートメントを追加すると、 AWS WAF スコープダウンステートメントと一致しないリクエストはすべてルールグループと一致しないと評価されます。スコープダウンステートメントに一致するリクエストのみがルールグループに対して評価されます。評価されたリクエスト数に基

づいて料金が発生するマネージドルールグループでは、スコープダウンステートメントはコストを抑えるのに役立ちます。

マネージドルールグループステートメントの詳細については、「マネージドルールグループステートメント」を参照してください。

- レートベースのルールステートメント – スコープダウンステートメントのレートを含まないレートベースのルールステートメントは、評価するすべてのリクエストのレートを制限します。特定のカテゴリのリクエストのレートのみを制御する場合は、レートベースのルールにスコープダウンステートメントを追加します。例えば、特定の地理的エリアから送信されたリクエストのレートだけを追跡および制御するには、地理的照合ステートメントでその地理的エリアを指定することで、スコープダウンステートメントとしてレートベースのルールに追加できます。

  レートベースのルールステートメントの詳細については、「レートベースのルールステートメント」を参照してください。

スコープダウンステートメントでは、任意のネスト可能なルールを使用できます。利用可能なステートメントについては、「一致ルールステートメント」および「論理ルールステートメント」を参照してください。スコープダウンステートメントの WCU は、その中で定義するルールステートメントに必要な WCU です。スコープダウンステートメントの使用に追加コストは発生しません

スコープダウンステートメントは、通常のルールでステートメントを使用する場合と同じ方法で設定できます。例えば、検査対象のウェブリクエストコンポーネントにテキスト変換を適用したり、IP アドレスとして使用するために転送された IP アドレスを指定したりできます。これらの設定はスコープダウンステートメントにのみ適用され、包含マネージドルールグループやレートベースのルールステートメントには継承されません。

例えば、スコープダウンステートメントのクエリ文字列にテキスト変換を適用すると、スコープダウンステートメントは変換を適用後、クエリ文字列を検査します。リクエストがスコープダウンステートメントの基準に一致する場合は、 AWS WAF スコープダウンステートメントの変換を行わずに、元の状態でウェブリクエストを包含ルールに渡します。スコープダウンステートメントを含むルールは、独自のテキスト変換を適用する場合がありますが、スコープダウンステートメントから継承されるものはありません。

スコープダウンステートメントを使用して、包含ルールステートメントのリクエスト検査設定を指定することはできません。スコープダウンステートメントを、包含ルールステートメントのウェブリクエストプリプロセッサとして使用することはできません。スコープダウンステートメントの唯一の役割は、どのリクエストを包含ルールステートメントに渡して検査するかを決定することです。

セットまたはルールグループを参照するステートメント

一部のルールでは、再利用可能なエンティティが使用され、ウェブ ACL の外部で、ユーザー自身または販売者が管理します。 AWS AWS Marketplace 再利用可能なエンティティが更新されると、AWS WAF は更新をルールに伝達します。たとえば、ウェブ ACL AWS でマネージドルールルールグループを使用する場合、 AWS ルールグループを更新すると、その変更がウェブ ACL AWS に反映され、動作が更新されます。ルールで IP セットステートメントを使用する場合、そのセットを更新すると、 AWS WAF その変更を参照するすべてのルールに変更が反映されるため、そのルールを使用するウェブ ACL up-to-date はすべて変更と共に保持されます。

ルールステートメントで使用できる再利用可能なエンティティを次に示します。

- IP セット - 独自の IP セットを作成および管理します。コンソールでは、ナビゲーションペインからこれらにアクセスできます。IP セットの管理については、「の IP セットと正規表現パターンセット AWS WAF」を参照してください。
- 正規表現一致セット - 独自の正規表現一致セットを作成および管理します。コンソールでは、ナビゲーションペインからこれらにアクセスできます。正規表現パターンセットの管理については、「の IP セットと正規表現パターンセット AWS WAF」を参照してください。
- AWS マネージドルールルールグループ — AWS これらのルールグループを管理します。コンソールでは、マネージドルールグループをウェブ ACL に追加する際にこれらを使用できます。これらの詳細については、「AWS マネージドルールルールグループリスト」を参照してください。
- AWS Marketplace マネージドルールグループ — AWS Marketplace 販売者がこれらのルールグループを管理し、ユーザーはそれらを購読して使用することができます。サブスクリプションを管理するには、コンソールのナビゲーションペインで [AWS Marketplace] を選択します。 AWS Marketplace マネージドルールグループをウェブ ACL に追加すると、マネージドルールグループが一覧表示されます。まだ登録していないルールグループについては、 AWS Marketplace そのページにもリンクがあります。 AWS Marketplace セラー管理ルールグループの詳細については、「」を参照してくださいAWS Marketplace マネージドルールグループ。
- 独自のルールグループ - マネージドルールグループでは利用できない動作が必要な場合は通常、独自のルールグループを管理します。コンソールでは、ナビゲーションペインからこれらにアクセスできます。詳細については、「独自のルールグループの管理」を参照してください。

参照先のセットまたはルールグループの削除

参照先エンティティを削除すると、そのエンティティがウェブ ACL AWS WAF で現在使用されているかどうかを確認します。 AWS WAF 使用中であることが判明すると、警告が表示されます。 AWS WAF エンティティーがウェブ ACL によって参照されているかどうかは、ほとんどいつでも判断でき

ます。ただし、まれに判別できないことがあります。削除するエンティティが使用中でないことを確認する必要がある場合は、削除する前にウェブ ACL でエンティティを確認してください。

一致ルールステートメント

一致ステートメントは、ウェブリクエストまたはその送信元を、指定された基準と比較します。このタイプの多くのステートメントでは、 AWS WAF リクエストの特定のコンポーネントを比較してコンテンツのマッチングを行います。

一致ステートメントはネスト可能です。これらのステートメントはいずれも論理ルールステートメント内にネストできる他、スコープダウンステートメントで使用できます。倫理ルールステートメントの詳細については、「論理ルールステートメント」を参照してください。スコープダウンステートメントの詳細については、「スコープダウンステートメント」を参照してください。

この表では、ルールに追加できる標準の一致ステートメントについて説明し、それぞれのウェブ ACL キャパシティーユニット (WCU) 使用量を計算するためのガイドラインを提供します。WCU の詳細については、「AWS WAF ウェブ ACL キャパシティーユニット (WCUs)」を参照してください。

一致ステートメント	説明	WCU
地理的一致	リクエストの送信元の国を検査し、その国および地域のラベルを適用します。	1
IP セット一致	リクエストを一連の IP アドレスおよびアドレス範囲と照合します。	ほとんどの場合 1。転送された IP アドレスを持つヘッダーを使用するようにステートメントを設定し、Any のヘッダー内の位置を指定すると、WCU が 4 増えます。
ラベル一致ルールステートメント	同じウェブ ACL 内の他のルールによって追加されたラベルのリクエストを検査します。	1

一致ステートメント	説明	WCU
正規表現一致ルールステートメント	正規表現パターンを指定されたリクエストコンポーネントと比較します。	3 (基本コストとして)。 [All query parameters] (すべてのクエリパラメータ) のリクエストコンポーネントを使用する場合、10 WCU を追加します。[JSON body] (JSON 本文) のリクエストコンポーネントを使用する場合、基本コストの WCU を倍増させます。適用する各テキスト変換について、10 WCU を追加します。
正規表現パターンセット	正規表現パターンを指定されたリクエストコンポーネントと比較します。	パターンセットあたり 25 (基本コストとして)。 [All query parameters] (すべてのクエリパラメータ) のリクエストコンポーネントを使用する場合、10 WCU を追加します。[JSON body] (JSON 本文) のリクエストコンポーネントを使用する場合、基本コストの WCU を倍増させます。適用する各テキスト変換について、10 WCU を追加します。

一致ステートメント	説明	WCU
サイズ制約	指定されたリクエストコンポーネントに対してサイズ制約をチェックします。	1 (基本コストとして)。 [All query parameters] (すべてのクエリパラメータ) のリクエストコンポーネントを使用する場合、10 WCU を追加します。[JSON body] (JSON 本文) のリクエストコンポーネントを使用する場合、基本コストの WCU を倍増させます。適用する各テキスト変換について、10 WCU を追加します。
SQLi 攻撃	指定されたリクエストコンポーネント内の悪意のある SQL コードを検査します。	20 (基本コストとして)。 [All query parameters] (すべてのクエリパラメータ) のリクエストコンポーネントを使用する場合、10 WCU を追加します。[JSON body] (JSON 本文) のリクエストコンポーネントを使用する場合、基本コストの WCU を倍増させます。適用する各テキスト変換について、10 WCU を追加します。

一致ステートメント	説明	WCU
文字列一致	指定されたリクエストコンポーネントと文字列を比較します。	基本コストは、文字列の一致のタイプによって異なり、1 ~ 10 の範囲です。 [All query parameters] (すべてのクエリパラメータ) のリクエストコンポーネントを使用する場合、10 WCU を追加します。[JSON body] (JSON 本文) のリクエストコンポーネントを使用する場合、基本コストの WCU を倍増させます。適用する各テキスト変換について、10 WCU を追加します。
XSS スクリプティング攻撃	指定されたリクエストコンポーネントでのクロスサイトスクリプティング攻撃を検査します。	40 (基本コストとして)。 [All query parameters] (すべてのクエリパラメータ) のリクエストコンポーネントを使用する場合、10 WCU を追加します。[JSON body] (JSON 本文) のリクエストコンポーネントを使用する場合、基本コストの WCU を倍増させます。適用する各テキスト変換について、10 WCU を追加します。

地理的一致ルールステートメント

地理的または地理照合ステートメントを使用して、発信元の国や地域に基づいてウェブリクエストを管理します。地理的照合ステートメントは、ウェブリクエストに発信国や発信地域を示すラベルを追加します。これらのラベルは、ステートメントの条件がリクエストと一致するかどうかに関係なく追加されます。また、地理的照合ステートメントは、リクエストの発信国に対する一致も実行します。

地理的一致ステートメントの使用方法

地理的一致ステートメントは、次のような国や地域のマッチングに使用できます。

- 国 — 地理的一致ルールを単独で使用し、発信国のみに基づいてリクエストを管理できます。ルールステートメントは国コードに対する一致を実行します。また、発信元のラベルに一致するラベル一致ルールが適用されている地域的一致ルールに従うこともできます。
- 地域 — 地理的一致ルールに続いてラベル一致ルールを使用し、発信地域に基づいてリクエストを管理します。地理的一致ルールだけを使用して地域コードとの一致を実行することはできません。

ラベルマッチルールの使用方法については、「ラベル一致ルールステートメント」および「AWS WAF ウェブリクエストの ラベル」を参照してください。

地理的一致ステートメントのしくみ

geo match ステートメントを使用して、 AWS WAF 各ウェブリクエストを次のように管理します。

1. AWS WAF リクエストの国と地域コードの決定 — IP アドレスに基づいてリクエストの国と地域を決定します。デフォルトでは、ウェブリクエストの発信元の IP AWS WAF アドレスを使用します。ルールステートメント設定で転送 IP AWS WAF 設定を有効にするなどして、代替リクエストヘッダーの IP `X-Forwarded-For` アドレスを使用するように指示できます。

   AWS WAF MaxMind GeoIP データベースを使用してリクエストの場所を決定します。MaxMind 国や IP の種類などの要因によって精度は異なりますが、国レベルでのデータの精度は非常に高いと報告します。詳細については MaxMind、「MaxMind IP ジオロケーション」を参照してください。GeoIPデータのいずれかが正しくないと思われる場合は、「GeoIP2データの修正」MaxMindでMaxmindに修正リクエストを送信できます。

   AWS WAF 国際標準化機構 (ISO) 3166 規格の alpha-2 国および地域コードを使用します。コードは次の場所にあります。

   - ISO ウェブサイトでは、「ISO Online Browsing Platform (OBP)」(ISO オンラインブラウジングプラットフォーム (OBP)) で国コードを検索できます。
   - ウィキペディアでは、国コードは「ISO 3166-2」に一覧表示されています。

     国の地域コードは `https://en.wikipedia.org/wiki/ISO_3166-2:<ISO country code>` の URL に一覧表示されています。たとえば、米国の地域は「ISO 3166-2:US」にあり、ウクライナの地域は「ISO 3166-2:UA」にあります。

2. リクエストに追加する国ラベルおよび地域ラベルを決定します — ラベルは、地理一致ステートメントが発信元 IP 設定を使用するか、転送された IP 設定を使用するかを示します。

   - 発信元 IP

     国ラベルは awswaf:clientip:geo:country:*<ISO country code>* です。米国の例は awswaf:clientip:geo:country:US です。

     地域ラベルは awswaf:clientip:geo:region:*<ISO country code>*-*<ISO region code>* です。米国のオレゴン州の例は awswaf:clientip:geo:region:US-OR です。

   - 転送された IP

     国ラベルは awswaf:forwardedip:geo:country:*<ISO country code>* です。米国の例は awswaf:forwardedip:geo:country:US です。

     地域ラベルは awswaf:forwardedip:geo:region:*<ISO country code>*-*<ISO region code>* です。米国のオレゴン州の例は awswaf:forwardedip:geo:region:US-OR です。

   リクエストが指定した IP アドレスの国または地域コードがない場合、 AWS WAF はラベル内で値の代わりに XX を使用します。たとえば、次のラベルは国コードがないクライアント IP 用です。awswaf:clientip:geo:country:XX および次のラベルは、国が米国であるが地域コードがない転送先 IP 用です。awswaf:forwardedip:geo:region:US-XX。

3. リクエストの国コードをルール基準に参照して評価

地理一致ステートメントは、一致するものを見つけるかどうかにかかわらず、検査するすべてのリクエストに国と地域のラベルを追加します。

> **Note**
>
> AWS WAF ルールの Web リクエスト評価の最後に任意のラベルを追加します。そのため、地理的一致ステートメントのラベルに対して使用するラベル一致は、地理的一致ステートメントを含むルールとは別のルールで定義する必要があります。

地域値のみを調べる場合、Count アクションと単一の国コード一致で地域的一致ルールを記述し、その後に地域ラベルのラベル一致ルールを作成することができます。この方法でも、地域一致ルールを評価するには国コードを入力する必要があります。サイトへのトラフィック元になる可能性が非常に低い国を指定することで、ログ記録およびカウントメトリクスを減らすことができます。

CloudFront CloudFront ディストリビューションと地域制限機能

CloudFront ディストリビューションでは、 CloudFront 地域制限機能を使用する場合、ブロックされたリクエストはに転送されないことに注意してください。 AWS WAF許可されたリクエストはに転送されます。 AWS WAF地域やその他の指定可能な条件に基づいてリクエストをブロックしたい場合は AWS WAF、地域制限機能を使わずに AWS WAF Geo match ステートメントを使用してください。 CloudFront

地理的一致ステートメントの特徴

ネスト可能 - このステートメントタイプはネスト可能です。

WCU - 1 つの WCU。

設定 – このステートメントは次の設定を使用します。

- 国コード – 地理一致のために比較する国コードの配列。これらは、ISO 3166 国際規格の alpha-2 の国 ISO コード (たとえば、`["US","CN"]` など) を基準とした 2 文字の国コードである必要があります。
- (オプション) 転送 IP 設定 — デフォルトでは、ウェブリクエストの送信元の IP AWS WAF アドレスを使用して送信元国を決定します。代わりに、HTTP ヘッダーで転送された IP `X-Forwarded-For` を使用するようにルールを設定することもできます。 AWS WAF ヘッダーの最初の IP アドレスを使用します。この設定では、ヘッダーに不正な形式の IP アドレスを持つウェブリクエストに適用するフォールバック動作も指定します。フォールバック動作は、リクエストの一致結果を、一致または不一致のいずれにするかを設定します。詳細については、「転送された IP アドレス」を参照してください。

このルールステートメントの場所

- コンソールのルールビルダー - [Request option] (リクエストオプション) で [Originates from a country in] (次の国からの送信) を選択します。
- API — GeoMatchStatement

例

地域一致ステートメントを使用して、特定の国または地域からのリクエストを管理できます。たとえば、特定の国からのリクエストをブロックしても、それらの国に属する一連の IP アドレスからのリクエストを許可するには、Block に設定されたアクションと次のネストされたステートメント (次の疑似コードで表示) を含めたルール作成できます。

- AND ステートメント
  - ブロックする国をリストした 地理一致ステートメント
  - NOT ステートメント
    - 許可する IP アドレスを指定する IP セットステートメント

または、特定の国の一部地域をブロックしても、それらの国における他の地域からのリクエストを許可するには、まずアクションセットを持った地理一致リールを Count に定義できます。その後、追加された地理一致ラベルと照合し、必要に応じてリクエストを処理するラベルマッチルールを定義します。

次の擬似コードは、このアプローチの例について説明しています。

1. 地理一致ステートメントはブロックする地域がある国をリストしますが、アクションを Count に設定された状態で実行します。これにより、マッチステータスに関係なくすべてのウェブリクエストにラベルが付けられ、対象国のカウントメトリクスも表示されます。
2. Block アクションを含む AND ステートメント
   - ブロックする国のラベルを指定するラベル一致ステートメント
   - NOT ステートメント
     - 許可する国の地域のラベルを指定するラベル一致ステートメント

次の JSON リストは、前述の擬似コードで説明した 2 つのルールの実装を示しています。これらのルールは、オレゴン州とワシントン州からのトラフィックを除く米国からのトラフィックをすべてブロックします。地理一致ステートメントは、検査するすべてのリクエストに国と地域のラベルを追加します。ラベル一致ルールは地理一致ルールの後に実行されるため、地理一致ルールが追加したばかりの国や地域のラベルと照合できます。地理一致ステートメントは転送された IP アドレスを使用するため、ラベル一致は転送された IP ラベルも指定します。

```
{
   "Name": "geoMatchForLabels",
   "Priority": 10,
   "Statement": {
     "GeoMatchStatement": {
       "CountryCodes": [
         "US"
       ],
       "ForwardedIPConfig": {
           "HeaderName": "X-Forwarded-For",
```

```
            "FallbackBehavior": "MATCH"
          }
        }
    },
    "Action": {
      "Count": {}
    },
    "VisibilityConfig": {
      "SampledRequestsEnabled": true,
      "CloudWatchMetricsEnabled": true,
      "MetricName": "geoMatchForLabels"
    }
},
{
    "Name": "blockUSButNotOROrWA",
    "Priority": 11,
    "Statement": {
      "AndStatement": {
        "Statements": [
          {
            "LabelMatchStatement": {
              "Scope": "LABEL",
              "Key": "awswaf:forwardedip:geo:country:US"
            }
          },
          {
            "NotStatement": {
              "Statement": {
                 "OrStatement": {
                   "Statements": [
                     {
                        "LabelMatchStatement": {
                          "Scope": "LABEL",
                          "Key": "awswaf:forwardedip:geo:region:US-OR"
                        }
                     },
                     {
                        "LabelMatchStatement": {
                          "Scope": "LABEL",
                          "Key": "awswaf:forwardedip:geo:region:US-WA"
                        }
                     }
                   ]
                }
```

```
              }
            }
          }
        ]
      }
    },
    "Action": {
      "Block": {}
    },
    "VisibilityConfig": {
      "SampledRequestsEnabled": true,
      "CloudWatchMetricsEnabled": true,
      "MetricName": "blockUSButNotOROrWA"
    }
}
```

別の例として、地理一致とレートベースのルールを組み合わせて、特定の国または地域のユーザーのリソースに優先順位を付けることができます。ユーザーを区別するために使用する地理一致またはラベル一致のステートメントごとに、異なるレートベースのステートメントを作成します。優先させる国または地域のユーザーのレート制限をより高く設定し、他のユーザーのレート制限をより低く設定します。

次の JSON リストは、米国からのトラフィック量を制限する地理一致ルールの次にレートベースのルールを示しています。このルールにより、オレゴン州からのトラフィックは、同国の他の地域からのトラフィックよりも高いレートで許可されます。

```
{
  "Name": "geoMatchForLabels",
  "Priority": 190,
  "Statement": {
    "GeoMatchStatement": {
      "CountryCodes": [
        "US"
      ]
    }
  },
  "Action": {
    "Count": {}
  },
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
```

```
    "MetricName": "geoMatchForLabels"
  }
},
{
  "Name": "rateLimitOregon",
  "Priority": 195,
  "Statement": {
    "RateBasedStatement": {
      "Limit": 3000,
      "AggregateKeyType": "IP",
      "ScopeDownStatement": {
        "LabelMatchStatement": {
          "Scope": "LABEL",
          "Key": "awswaf:clientip:geo:region:US-OR"
        }
      }
    }
  },
  "Action": {
    "Block": {}
  },
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "rateLimitOregon"
  }
},
{
  "Name": "rateLimitUSNotOR",
  "Priority": 200,
  "Statement": {
    "RateBasedStatement": {
      "Limit": 100,
      "AggregateKeyType": "IP",
      "ScopeDownStatement": {
        "AndStatement": {
          "Statements": [
            {
              "LabelMatchStatement": {
                "Scope": "LABEL",
                "Key": "awswaf:clientip:geo:country:US"
              }
            },
            {
```

```
              "NotStatement": {
                "Statement": {
                  "LabelMatchStatement": {
                    "Scope": "LABEL",
                    "Key": "awswaf:clientip:geo:region:US-OR"
                  }
                }
              }
            }
          ]
        }
      }
    }
  },
  "Action": {
    "Block": {}
  },
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "rateLimitUSNotOR"
  }
}
```

IP セット一致ルールステートメント

IP セット一致ステートメントは、一連の IP アドレスおよびアドレス範囲に対してウェブリクエストの IP アドレスを検査します。これを使用して、リクエストの送信元の IP アドレスに基づいてウェブリクエストを許可またはブロックします。デフォルトでは、 AWS WAF はウェブリクエストの発信元からの IP アドレスを使用しますが、代わりに X-Forwarded-For などの HTTP ヘッダーを使用するようにルールを設定できます。

AWS WAF を除くすべての IPv4 および IPv6 CIDR 範囲をサポートします。/0CIDR 表記の詳細については、Wikipedia の「Classless Inter-Domain Routing」(クラスレスドメイン間ルーティング) を参照してください。IP セットには、チェック対象として最大 10,000 個の IP アドレスまたは IP アドレス範囲を保持できます。

> **Note**
>
> 各 IP セット一致ルールは IP セットを参照します。このセットは、ルールとは無関係に作成し、維持します。1 つの IP セットを複数のルールで使用でき、参照先セットを更新すると、AWS WAF そのセットを参照するすべてのルールが自動的に更新されます。
> IP セットの作成および管理については、「IP セットの作成と管理」を参照してください。

ルールグループまたはウェブ ACL でルールを追加または更新する場合は、[IP set] (IP セット) オプションを選択し、使用する IP セットの名前を選択します。

ネスト可能 - このステートメントタイプはネスト可能です。

WCU- ほとんどの場合 1 WCU。転送された IP アドレスを使用するようにステートメントを設定し、ANY の位置を指定すると、WCU の使用量が 4 増えます。

このステートメントには、次の設定を使用します。

- IP セットの指定 - 使用する IP セットをリストから選択するか、新しい IP セットを作成します。
- (オプション) 転送された IP 設定 – リクエストの発信元の代わりに使用する代替の転送された IP ヘッダー名。ヘッダーの最初のアドレス、最後のアドレス、または任意のアドレスを照合するかどうかを指定します。指定したヘッダーに不正な形式の IP アドレスを持つウェブリクエストに適用するフォールバック動作も指定します。フォールバック動作は、リクエストの一致結果を、一致または不一致のいずれにするかを設定します。詳細については、「転送された IP アドレス」を参照してください。

このルールステートメントの場所

- コンソールのルールビルダー - [Request option] (リクエストオプション) で [Originates from an IP address in] (次の IP アドレスからの送信) を選択します。
- コンソールの [Add my own rules and rule groups] (独自のルールとルールグループの追加) ページ - [IP set] (IP セット) オプションを選択します。
- API — IP SetReferenceStatement

ラベル一致ルールステートメント

ラベル一致ステートメントは、ウェブリクエストにあるラベルを文字列指定に照らして検査します。検査用のルールで使用できるラベルは、同じウェブ ACL 評価内の他のルールによってウェブリクエストに既に追加されているラベルです。

ラベルはウェブ ACL CloudWatch 評価以外では保持されませんが、ラベルメトリックスにアクセスしたり、任意のウェブ ACL のラベル情報の概要をコンソールで確認したりできます。 AWS WAF 詳細については、ラベルメトリクスとディメンションおよびモニタリングとチューニングを参照してください。ログにはラベルも表示されます。詳細については、ログフィールド を参照してください。

> **Note**
>
> ラベル一致ステートメントは、ウェブ ACL で以前に評価されたルールのラベルのみを表示できます。ウェブ ACL AWS WAF 内のルールとルールグループを評価する方法については、を参照してください。ウェブ ACL でのルールおよびルールグループの処理順序

ラベルの追加と一致の詳細については、「AWS WAF ウェブリクエストの ラベル」を参照してください。

ネスト可能 - このステートメントタイプはネスト可能です。

WCU - 1 つの WCU

このステートメントには、次の設定を使用します。

- [Match scope] (一致範囲) – これを [Label] (ラベル) に設定して、ラベル名と、ならびにオプションで、先行する名前空間およびプレフィックスと照合します。これを [Namespace] (名前空間) に設定して、名前空間の指定の一部または全部、およびオプションで、先行するプレフィックスと照合します。
- [Key] (キー) – 照合する文字列。名前空間一致の範囲を指定する場合、これは、名前空間と、オプションでプレフィックスのみを指定する必要があり、末尾にコロンを付けます。ラベル一致の範囲を指定する場合、これはラベル名を含む必要があり、オプションで前述の名前空間とプレフィックスを含めることができます。

これらの設定の詳細については、「AWS WAF ラベルに一致する ルール」および「AWS WAF ラベル一致の例」を参照してください。

このルールステートメントの場所

- コンソールのルールビルダー - [Request option] (リクエストオプション) で [Has label] (ラベルあり) を選択します。
- API — LabelMatchStatement

正規表現一致ルールステートメント

regex match ステートメントは、リクエストコンポーネントを単一の正規表現 (regex) AWS WAF と照合するように指示します。リクエストコンポーネントが指定した正規表現と一致する場合、ウェブリクエストはステートメントと一致します。

このステートメントタイプは、数理論理を使用して一致基準を組み合わせることを希望する状況において、正規表現パターンセット一致ルールステートメント に代わる優れた方法です。例えば、リクエストコンポーネントを一部の正規表現パターンと照合し、他の正規表現パターンと照合しないようにする場合は、AND ルールステートメント と NOT ルールステートメント を使用して正規表現一致ステートメントを組み合わせることができます。

AWS WAF PCRE ライブラリが使用するパターン構文をサポートしますが、一部例外があります。`libpcre`ライブラリは、「PCRE - Perl Compatible Regular Expressions」で文書化されています。 AWS WAF サポートについては、を参照してくださいでの正規表現パターンマッチング AWS WAF。

ネスト可能 - このステートメントタイプはネスト可能です。

WCU- 3 WCU (基本コストとして)。[All query parameters] (すべてのクエリパラメータ) のリクエストコンポーネントを使用する場合、10 WCU を追加します。[JSON body] (JSON 本文) のリクエストコンポーネントを使用する場合、基本コストの WCU を倍増させます。適用する各テキスト変換について、10 WCU を追加します。

このステートメントタイプは、ウェブリクエストコンポーネントで動作し、次のリクエストコンポーネント設定が必要です。

- [リクエストコンポーネント] — ウェブリクエストの検査対象部分 (クエリ文字列や本文など)。

> **Warning**
>
> リクエストコンポーネントのボディ、JSON ボディ、ヘッダー、または Cookie を調べる場合は、 AWS WAF で検査できるコンテンツの量の制限についてお読みください。での オーバーサイズリクエストコンポーネントの処理 AWS WAF

ウェブリクエストコンポーネントの詳細については、「ウェブリクエストコンポーネントの仕様と処理」を参照してください。

- オプションのテキスト変換 — AWS WAF リクエストコンポーネントを検査する前に実行したい変換。例えば、小文字に変換したり、空白を正規化したりできます。複数の変換を指定すると、AWS WAF リストされている順序で処理されます。詳細については、テキスト変換オプション を参照してください。

このルールステートメントの場所

- コンソールのルールビルダー – [Match type] (一致タイプ) で、Matches regular expression] (正規表現に一致) を選択します。
- API — RegexMatchStatement

正規表現パターンセット一致ルールステートメント

正規表現パターンセット一致は、ウェブリクエストの指定した部分を、正規表現パターンセット内の指定した正規表現パターンに照らして検査します。

AWS WAF PCRE `libpcre` ライブラリで使用されるパターン構文をサポートします。ただし、いくつかの例外があります。ライブラリは、「PCRE - Perl Compatible Regular Expressions」で文書化されています。 AWS WAF サポートについては、を参照してくださいでの正規表現パターンマッチング AWS WAF。

> **Note**
>
> 各正規表現パターンセット一致ルールは、正規表現パターンセットを参照します。このパターンセットは、ルールとは無関係に作成し、維持します。1 つの正規表現パターンセットを複数のルールで使用でき、参照セットを更新すると、 AWS WAF それを参照するすべてのルールが自動的に更新されます。

正規表現パターンセットの作成および管理については、「正規表現パターンセットの作成と管理」を参照してください。

正規表現パターンセットマッチステートメントは、 AWS WAF 選択したリクエストコンポーネント内のセット内のパターンを検索するように指示します。リクエストコンポーネントがセット内のいずれかのパターンに一致する場合、ウェブリクエストはパターンセットルールステートメントと一致します。

論理を使用して正規表現パターンの一致を組み合わせる場合、例えば、一部の正規表現と照合し、他の正規表現とは照合しない場合は、正規表現一致ルールステートメント を使用することを検討してください。

ネスト可能 - このステートメントタイプはネスト可能です。

WCU - 基本コストとして 25 WCU。[All query parameters] (すべてのクエリパラメータ) のリクエストコンポーネントを使用する場合、10 WCU を追加します。[JSON body] (JSON 本文) のリクエストコンポーネントを使用する場合、基本コストの WCU を倍増させます。適用する各テキスト変換について、10 WCU を追加します。

このステートメントタイプは、ウェブリクエストコンポーネントで動作し、次のリクエストコンポーネント設定が必要です。

- [リクエストコンポーネント] — ウェブリクエストの検査対象部分 (クエリ文字列や本文など)。

  > **Warning**
  >
  > リクエストコンポーネントのボディ、JSON ボディ、ヘッダー、または Cookie を調べる場合は、 AWS WAF コンテンツを検査できる量の制限についてお読みください。でのオーバーサイズリクエストコンポーネントの処理 AWS WAF

  ウェブリクエストコンポーネントの詳細については、「ウェブリクエストコンポーネントの仕様と処理」を参照してください。
- オプションのテキスト変換 — AWS WAF リクエストコンポーネントを検査する前に実行したい変換。例えば、小文字に変換したり、空白を正規化したりできます。複数の変換を指定すると、AWS WAF 記載されている順序で処理されます。詳細については、「テキスト変換オプション」を参照してください。

このステートメントには、次の設定が必要です。

- 正規表現パターンセットの指定 - 使用する正規表現パターンセットをリストから選択するか、新しい IP セットを作成します。

このルールステートメントの場所

- コンソールのルールビルダー - [Match type] (一致タイプ) で [String match condition] (文字列一致条件) > [Matches pattern from regular expression set] (正規表現セットのパターンに一致) を選択します。
- API — RegexPatternSetReferenceStatement

サイズ制約ルールステートメント

サイズ制約ステートメントは、ウェブリクエストコンポーネントのバイト数とユーザーが指定したバイト数を比較し、比較基準に従って一致を実行します。比較基準は、「より大きい (>)」や「より小さい (<)」などの演算子です。例えば、100 バイトを超えるサイズのクエリ文字列を含むリクエストの一致を実行できます。

> **Note**
>
> このステートメントはウェブリクエストコンポーネントのサイズのみを検査します。コンポーネントのコンテンツは検査されません。

URI パスを検査する場合、パス内の / は 1 文字としてカウントされます。例えば、URI パスの /logo.jpg は 9 文字の長さになります。

ネスト可能 - このステートメントタイプはネスト可能です。

WCU- 1 WCU (基本コストとして)。[All query parameters] (すべてのクエリパラメータ) のリクエストコンポーネントを使用する場合、10 WCU を追加します。[JSON body] (JSON 本文) のリクエストコンポーネントを使用する場合、基本コストの WCU を倍増させます。適用する各テキスト変換について、10 WCU を追加します。

このステートメントタイプは、ウェブリクエストコンポーネントで動作し、次のリクエストコンポーネント設定が必要です。

- [リクエストコンポーネント] — ウェブリクエストの検査対象部分 (クエリ文字列や本文など)。ウェブリクエストコンポーネントの詳細については、「ウェブリクエストコンポーネントの仕様と処理」を参照してください。

  サイズ制約ステートメントは、何らかの変換が適用された後のコンポーネントのサイズのみを検査します。コンポーネントのコンテンツは検査されません。
- オプションのテキスト変換 — AWS WAF サイズを調べる前にリクエストコンポーネントに対して実行したい変換。例えば、空白を圧縮したり、HTML エンティティをデコードしたりすることができます。複数の変換を指定すると、 AWS WAF リストされている順序で処理されます。詳細については、「テキスト変換オプション」を参照してください。

さらに、このステートメントには、次の設定が必要です。

- Size match condition (サイズ一致条件) - これは、提供するサイズと選択したリクエストコンポーネントを比較するために使用する数値比較演算子を示します。リストから演算子を選択します。
- Size (サイズ) - 比較で使用するサイズ設定 (バイト単位)。

このルールステートメントの場所

- コンソールのルールビルダー - [Match type] (一致タイプ) の [Size match condition] (サイズ一致条件) で、使用する条件を選択します。
- API — SizeConstraintStatement

SQL インジェクション攻撃ルールステートメント

SQL インジェクションルールステートメントは、悪意のある SQL コードを検査します。攻撃者は、データベースを変更したり、データベースからデータを抽出したりするために、悪意のある SQL コードをウェブリクエストに挿入します。

ネスト可能 - このステートメントタイプはネスト可能です。

WCU – 基本コストは、ルールステートメントの感度レベルの設定によって異なります。Low のコストは 20 で、High のコストは 30 です。

[All query parameters] (すべてのクエリパラメータ) のリクエストコンポーネントを使用する場合、10 WCU を追加します。[JSON body] (JSON 本文) のリクエストコンポーネントを使用する場合、基本コストの WCU を倍増させます。適用する各テキスト変換について、10 WCU を追加します。

このステートメントタイプは、ウェブリクエストコンポーネントで動作し、次のリクエストコンポーネント設定が必要です。

- [リクエストコンポーネント] — ウェブリクエストの検査対象部分 (クエリ文字列や本文など)。

  > **Warning**
  >
  > リクエストコンポーネントのボディ、JSON ボディ、ヘッダー、または Cookie を調べる場合は、 AWS WAF コンテンツを検査できる量の制限についてお読みください。[でのオーバーサイズリクエストコンポーネントの処理 AWS WAF]

  ウェブリクエストコンポーネントの詳細については、「[ウェブリクエストコンポーネントの仕様と処理]」を参照してください。
- オプションのテキスト変換 — AWS WAF リクエストコンポーネントを検査する前に実行したい変換。例えば、小文字に変換したり、空白を正規化したりできます。複数の変換を指定すると、AWS WAF リストされている順序で処理されます。詳細については、[テキスト変換オプション] を参照してください。

さらに、このステートメントには、次の設定が必要です。

- 感度レベル – この設定は、SQL インジェクション一致基準の感度を調整します。オプションは LOW と HIGH です。デフォルトの設定は、LOW です。

  HIGH 設定は、より多くの SQL インジェクション攻撃を検出するため、推奨される設定です。感度が高いため、この設定では、特にウェブリクエストに通常とは異なる文字列が一般的に含まれている場合に、より多くの誤検知が生成されます。ウェブ ACL のテストとチューニング中に、誤検知を軽減するためにさらに多くの作業が必要になる場合があります。詳細については、[AWS WAF 保護機能のテストと調整] を参照してください。

  設定が低いほど、SQL インジェクションの検出の厳格度も緩くなり、誤検知も少なくなります。LOW は、SQL インジェクション攻撃に対する他の保護を備えているリソースや、誤検知に対する許容度が低いリソースにとって、より適切な選択肢である可能性があります。

このルールステートメントの場所

- コンソールのルールビルダー - [Match type] (一致タイプ) で [Attack match condition] (攻撃一致条件) > [Contains SQL injection attacks] (SQL インジェクション攻撃を含む) を選択します。

- API — SqliMatchStatement

文字列一致ルールステートメント

文字列一致ステートメントは、リクエストで AWS WAF 検索する文字列、検索するリクエスト内の場所、および方法を示します。例えば、リクエストに含まれるクエリ文字列の先頭にある特定の文字列、またはリクエストの `User-agent` ヘッダーと完全に一致する特定の文字列を検索できます。通常、文字列は印刷可能な ASCII 文字で構成されますが、16 進数 0x00 ～ 0xFF (10 進数 0 ～ 255) の任意の文字を使用できます。

ネスト可能 - このステートメントタイプはネスト可能です。

WCU – 基本コストは、使用する一致のタイプによって異なります。

- 次の文字列に完全一致 - 2
- 文字列で始まる - 2
- 文字列で終わる - 2
- 文字列を含む - 10
- 単語を含む – 10

[All query parameters] (すべてのクエリパラメータ) のリクエストコンポーネントを使用する場合、10 WCU を追加します。[JSON body] (JSON 本文) のリクエストコンポーネントを使用する場合、基本コストの WCU を倍増させます。適用する各テキスト変換について、10 WCU を追加します。

このステートメントタイプは、ウェブリクエストコンポーネントで動作し、次のリクエストコンポーネント設定が必要です。

- [リクエストコンポーネント] — ウェブリクエストの検査対象部分 (クエリ文字列や本文など)。

> **Warning**
>
> リクエストコンポーネント本文 、JSON 本文 、ヘッダー 、または Cookie を検査する場合は、 で検査 AWS WAF できるコンテンツの量に関する制限についてお読みくださいでのオーバーサイズリクエストコンポーネントの処理 AWS WAF。

ウェブリクエストコンポーネントの詳細については、「ウェブリクエストコンポーネントの仕様と処理」を参照してください。

- オプションのテキスト変換 – 検査する前にリクエストコンポーネントで AWS WAF 実行する変換。例えば、小文字に変換したり、空白を正規化したりできます。複数の変換を指定すると、 はリストされた順序で AWS WAF 変換を処理します。詳細については、「テキスト変換オプション」を参照してください。

さらに、このステートメントには、次の設定が必要です。

- 照合する文字列 – これは、指定されたリクエストコンポーネント AWS WAF と比較する文字列です。通常、文字列は印刷可能な ASCII 文字で構成されますが、16 進数 0x00 ～ 0xFF (10 進数 0 ～ 255) の任意の文字を使用できます。
- 文字列一致条件 — AWS WAF 実行する検索タイプを示します。
  - Exactly matches string (次の文字列に完全一致) - リクエストコンポーネントの文字列と値が同一です。
  - Starts with string (次の文字列で始まる) - この文字列は、リクエストコンポーネントの先頭に出現します。
  - Ends with string (次の文字列で終わる) - この文字列は、リクエストコンポーネントの末尾に出現します。
  - Contains string (次の文字列を含む) - この文字列は、リクエストコンポーネント内の任意の場所に出現します。
  - Contains word (次の文字列を含む) - 指定した文字列がリクエストコンポーネントに出現する必要があります。

    このオプションの場合、指定する文字列には英数字またはアンダースコア (A～Z、a～z、0～9、または_) のみを使用できます。

    リクエストが一致するには、次のいずれかに当てはまる必要があります。
    - 文字列が、ヘッダーの値など、リクエストコンポーネントの値と正確に一致する。
    - 文字列が、リクエストコンポーネントの先頭にあり、英数字または下線 (_) 以外の文字が続く (例: `BadBot;`)。
    - 文字列が、リクエストコンポーネントの末尾にあり、英数字または下線 (_) 以外の文字が先行する (例: `;BadBot`)。

- 文字列が、リクエストコンポーネントの中央にあり、英数字または下線 (_) 以外の文字が前後にある (例: `-BadBot;`)。

このルールステートメントの場所

- コンソールのルールビルダー - [Match type] (一致タイプ) で [String match condition] (文字列一致条件) を選択し、一致させる文字列を入力します。
- API – ByteMatchStatement

クロスサイトスクリプティング攻撃ルールステートメント

XSS (クロスサイトスクリプティング) 攻撃ステートメントは、ウェブリクエストコンポーネント内の悪意のあるスクリプトを検査します。XSS 攻撃では、攻撃者は、悪意のあるクライアントサイトスクリプトを他の正当なウェブブラウザに挿入するための手段として、悪意のないウェブサイトの脆弱性を利用します。

ネスト可能 - このステートメントタイプはネスト可能です。

WCU- 40 WCU (基本コストとして)。[All query parameters] (すべてのクエリパラメータ) のリクエストコンポーネントを使用する場合、10 WCU を追加します。[JSON body] (JSON 本文) のリクエストコンポーネントを使用する場合、基本コストの WCU を倍増させます。適用する各テキスト変換について、10 WCU を追加します。

このステートメントタイプは、ウェブリクエストコンポーネントで動作し、次のリクエストコンポーネント設定が必要です。

- [リクエストコンポーネント] — ウェブリクエストの検査対象部分 (クエリ文字列や本文など)。

> **Warning**
>
> リクエストコンポーネントのボディ、JSON ボディ、ヘッダー、または Cookie を調べる場合は、 AWS WAF コンテンツを検査できる量の制限についてお読みください。でのオーバーサイズリクエストコンポーネントの処理 AWS WAF

ウェブリクエストコンポーネントの詳細については、「ウェブリクエストコンポーネントの仕様と処理」を参照してください。

- オプションのテキスト変換 — AWS WAF リクエストコンポーネントを検査する前に実行したい変換。例えば、小文字に変換したり、空白を正規化したりできます。複数の変換を指定すると、AWS WAF リストされている順序で処理されます。詳細については、テキスト変換オプション を参照してください。

このルールステートメントの場所

- コンソールのルールビルダー - [Match type] (一致タイプ) で [Attack match condition] (攻撃一致条件) > [Contains XSS injection attacks] (SQL インジェクション攻撃を含む) を選択します。
- API — XssMatchStatement

論理ルールステートメント

論理ルールステートメントを使用して他のステートメントを組み合わせたり、結果を否定したりします。すべての論理ルールステートメントに、1 つ以上のネストされたステートメントが必要です。

ルールステートメントの結果を論理的に結合または否定するには、ステートメントを論理ルールステートメントの下にネストします。

論理ルールステートメントはネスト可能です。他の論理ルールステートメント内にネストして、スコープダウンステートメントで使用できます。スコープダウンステートメントの詳細については、「スコープダウンステートメント」を参照してください。

> **Note**
>
> コンソールのビジュアルエディタは、1 レベルのルールステートメントのネストをサポートしており、多くのニーズに対応しています。より多くのレベルをネストするには、コンソールでルールの JSON 表現を編集するか、API を使用します。

この表では、論理ルールステートメントについて説明し、それぞれのウェブ ACL キャパシティーユニット (WCU) 使用量を計算するためのガイドラインを提供します。WCU の詳細については、「AWS WAF ウェブ ACL キャパシティーユニット (WCUs)」を参照してください。

論理ステートメント	説明	WCU
AND のロジック	ネストされたステートメントを AND ロジックと組み合わせます。	ネストされたステートメントに基づく
NOT のロジック	ネストされたステートメントの結果を否定します。	ネストされたステートメントに基づく
OR のロジック	ネストされたステートメントを OR ロジックと組み合わせます。	ネストされたステートメントに基づく

AND ルールステートメント

AND ルールステートメントは、ネストされたステートメントを論理 AND 演算と組み合わせるため、AND ステートメントが一致するには、ネストされたステートメントがすべて一致する必要があります。これには、少なくとも 2 つのネストされたステートメントが必要です。

ネスト可能 - このステートメントタイプはネスト可能です。

WCU - ネストされたステートメントに応じて異なります。

このルールステートメントの場所

- コンソールのルールビルダー - [If a request] (リクエストの状態) で [matches all the statements (AND)] (すべてのステートメントに一致する場合 (AND)) を選択してから、ネストされたステートメントに入力します。
- API – AndStatement

例

次のリストは、AND および NOT 論理ルールステートメントを使用して、SQL インジェクション攻撃ステートメントの一致から誤検知を排除する方法を示しています。この例では、誤検知につながるリクエストと一致する 1 バイトの一致ステートメントを記述できるとします。

AND ステートメントは、バイト一致ステートメントと一致せず、SQL インジェクション攻撃ステートメントと一致するリクエストと一致します。

```
{
      "Name": "SQLiExcludeFalsePositives",
      "Priority": 0,
      "Statement": {
        "AndStatement": {
          "Statements": [
            {
              "NotStatement": {
                "Statement": {
                  "ByteMatchStatement": {
                    "SearchString": "string identifying a false positive",
                    "FieldToMatch": {
                      "Body": {
                        "OversizeHandling": "MATCH"
                      }
                    },
                    "TextTransformations": [
                      {
                        "Priority": 0,
                        "Type": "NONE"
                      }
                    ],
                    "PositionalConstraint": "CONTAINS"
                  }
                }
              }
            },
            {
              "SqliMatchStatement": {
                "FieldToMatch": {
                  "Body": {
                    "OversizeHandling": "MATCH"
                  }
                },
                "TextTransformations": [
                  {
                    "Priority": 0,
                    "Type": "NONE"
                  }
                ]
              }
            }
          ]
```

```
        }
      },
      "Action": {
        "Block": {}
      },
      "VisibilityConfig": {
        "SampledRequestsEnabled": true,
        "CloudWatchMetricsEnabled": true,
        "MetricName": "SQLiExcludeFalsePositives"
      }
    }
```

コンソールルールビジュアルエディタを使用して、非論理ステートメントまたは NOT ステートメントを OR または AND ステートメントの下にネストできます。NOT ステートメントのネストは、前の例に示されています。

コンソールルールビジュアルエディタを使用すると、ほとんどのネスト可能なステートメントを、前の例に示したような論理ルールステートメントの下にネストできます。ビジュアルエディタを使用して OR または AND ステートメントをネストすることはできません。このタイプのネストを設定するには、JSON でルールステートメントを指定する必要があります。例えば、次の JSON ルールリストには、AND ステートメント内にネストされた OR ステートメントが含まれています。

```
{
  "Name": "match_rule",
  "Priority": 0,
  "Statement": {
    "AndStatement": {
      "Statements": [
        {
          "LabelMatchStatement": {
            "Scope": "LABEL",
            "Key": "awswaf:managed:aws:bot-control:bot:category:monitoring"
          }
        },
        {
          "NotStatement": {
            "Statement": {
              "LabelMatchStatement": {
                "Scope": "LABEL",
                "Key": "awswaf:managed:aws:bot-control:bot:name:pingdom"
              }
            }
```

```
          }
        },
        {
          "OrStatement": {
            "Statements": [
              {
                "GeoMatchStatement": {
                  "CountryCodes": [
                    "JM",
                    "JP"
                  ]
                }
              },
              {
                "ByteMatchStatement": {
                  "SearchString": "JCountryString",
                  "FieldToMatch": {
                    "Body": {}
                  },
                  "TextTransformations": [
                    {
                      "Priority": 0,
                      "Type": "NONE"
                    }
                  ],
                  "PositionalConstraint": "CONTAINS"
                }
              }
            ]
          }
        }
      ]
    }
  },
  "Action": {
    "Block": {}
  },
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "match_rule"
  }
}
```

NOT ルールステートメント

NOT ルールステートメントは、単一のネストされたステートメントの結果を論理的に否定するため、NOT ステートメントが一致するには、ネストされたステートメントが一致してはならず、その逆も同様です。これには、ネストされたステートメントが 1 つ必要です。

例えば、特定の国を送信元としないリクエストをブロックする場合は、アクションをブロックに設定した NOT ステートメントを作成し、国を指定する地理的一致ステートメントをネストします。

ネスト可能 - このステートメントタイプはネスト可能です。

WCU - ネストされたステートメントに応じて異なります。

このルールステートメントの場所

- コンソールのルールビルダー - [If a request] (リクエストの状態) で [doesn't match the statement (NOT)] (すべてのステートメントに一致しない場合 (NOT)) を選択してから、ネストされたステートメントに入力します。
- API — NotStatement

OR ルールステートメント

OR ルールステートメントは、ネストされたステートメントを OR ロジックと組み合わせるため、OR ステートメントが一致するには、ネストされたステートメントのいずれか 1 つが一致する必要があります。これには、少なくとも 2 つのネストされたステートメントが必要です。

例えば、特定の国から送信されたリクエストや特定のクエリ文字列を含むリクエストをブロックする場合は、OR ステートメントを作成し、その国の地理的照合ステートメントとクエリ文字列の文字列照合ステートメントをネストします。

代わりに、特定の国から送信されていないリクエストや特定のクエリ文字列を含むリクエストをブロックする場合は、前述の OR ステートメントを変更して、NOT ステートメント内の 1 レベル下に地理的照合ステートメントをネストします。コンソールでは 1 レベルのネストしかサポートされないため、このレベルのネストでは JSON 形式を使用する必要があります。

ネスト可能 - このステートメントタイプはネスト可能です。

WCU - ネストされたステートメントに応じて異なります。

このルールステートメントの場所

- コンソールのルールビルダー - [If a request] (リクエストの状態) で [matches at least one of the statements (OR)] (1 つ以上のステートメントに一致する場合 (OR)) を選択してから、ネストされたステートメントに入力します。
- API – OrStatement

例

次のリストは、OR を使用して他の 2 つのステートメントを結合する方法を示しています。ネストされたステートメントのいずれかが一致する場合、OR ステートメントは一致します。

```
{
  "Name": "neitherOfTwo",
  "Priority": 1,
  "Action": {
    "Block": {}
  },
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "neitherOfTwo"
  },
  "Statement": {
    "OrStatement": {
      "Statements": [
        {
          "GeoMatchStatement": {
            "CountryCodes": [
              "CA"
            ]
          }
        },
        {
          "IPSetReferenceStatement": {
            "ARN": "arn:aws:wafv2:us-east-1:111111111111:regional/ipset/test-ip-
set-22222222/33333333-4444-5555-6666-777777777777"
          }
        }
      ]
    }
  }
```

```
}
```

コンソールルールビジュアルエディタを使用すると、ネスト可能なステートメントのほとんどを論理ルールステートメントの下にネストできますが、ビジュアルエディタを使用して OR または AND ステートメントをネストすることはできません。このタイプのネストを設定するには、JSON でルールステートメントを指定する必要があります。例えば、次の JSON ルールリストには、AND ステートメント内にネストされた OR ステートメントが含まれています。

```
{
  "Name": "match_rule",
  "Priority": 0,
  "Statement": {
    "AndStatement": {
      "Statements": [
        {
          "LabelMatchStatement": {
            "Scope": "LABEL",
            "Key": "awswaf:managed:aws:bot-control:bot:category:monitoring"
          }
        },
        {
          "NotStatement": {
            "Statement": {
              "LabelMatchStatement": {
                "Scope": "LABEL",
                "Key": "awswaf:managed:aws:bot-control:bot:name:pingdom"
              }
            }
          }
        },
        {
          "OrStatement": {
            "Statements": [
              {
                "GeoMatchStatement": {
                  "CountryCodes": [
                    "JM",
                    "JP"
                  ]
                }
              },
              {
                "ByteMatchStatement": {
```

```
                "SearchString": "JCountryString",
                "FieldToMatch": {
                  "Body": {}
                },
                "TextTransformations": [
                  {
                    "Priority": 0,
                    "Type": "NONE"
                  }
                ],
                "PositionalConstraint": "CONTAINS"
              }
            }
          ]
        }
      }
    ]
  }
},
"Action": {
  "Block": {}
},
"VisibilityConfig": {
  "SampledRequestsEnabled": true,
  "CloudWatchMetricsEnabled": true,
  "MetricName": "match_rule"
}
}
```

レートベースのルールステートメント

レートベースのルールでは、受信リクエストをカウントし、レートが速すぎる場合にはリクエストを制限します。ルールは条件に従ってリクエストを集約し、ルールの評価ウィンドウ、リクエスト制限、アクション設定に基づいて集計されたグループをカウントし、レート制限します。

> **Note**
>
> また、Bot Control AWS Managed Rules ルールグループのターゲット保護レベルを使用してウェブリクエストをレート制限することもできます。このマネージドルールグループを使用すると、追加料金がかかります。詳細については、「レートベースのルールとターゲットを絞った Bot Control ルールにおけるレート制限のオプション」を参照してください。

AWS WAF 使用するレートベースのルールのインスタンスごとにウェブリクエストを個別に追跡および管理します。たとえば、2 つのウェブ ACL に同じレートベースのルール設定を指定した場合、2 つのルールステートメントはそれぞれレートベースのルールの個別のインスタンスを表し、それぞれが独自の追跡と管理を行います。 AWS WAFルールグループ内でレートベースのルールを定義し、そのルールグループを複数の場所で使用すると、使用するたびにレートベースのルールのインスタンスが個別に作成され、独自の追跡と管理が行われます。 AWS WAF

ネスト不可 - このステートメントタイプを他のステートメント内にネストすることはできません。このタイプは、ウェブ ACL およびルールグループに直接含めることができます。

スコープダウンステートメント — このルールタイプではスコープダウンステートメントを使用して、ルールが追跡するリクエストの範囲とレート制限を絞り込むことができます。スコープダウンステートメントは、他のルール構成設定に応じて、オプションでも必須でもかまいません。詳細はこのセクションで説明されています。スコープダウンステートメントに関する一般的な情報については、を参照してください。スコープダウンステートメント

WCU - 2 個の WCU (基本コストとして)。指定するカスタム集約キーごとに、30 個の WCU を追加します。ルールでスコープダウンステートメントを使用する場合は、その分の WCU を計算して追加します。

このルールステートメントの場所

- ウェブ ACL のコンソールのルールビルダー – [Rule] (ルール) の [Type] (タイプ) で、[Rate-based rule] (レートベースのルール) を選択します。
- API — RateBasedStatement

トピック

- レートベースのルールの概要レベル設定
- レートベースのルールに関する注意事項
- レートベースのルール集約オプションとキー
- レートベースのルール集約インスタンスとカウント
- レートベースのルールリクエストレート制限の動作
- レートベースのルールの例
- レートベースのルールによってレート制限されている IP アドレスの一覧表示

レートベースのルールの概要レベル設定

レートベースのルールステートメントでは、以下の高レベル設定を使用します。

- 評価ウィンドウ — 現在時刻から振り返って、 AWS WAF リクエスト数に含める必要のある時間 (秒単位)。たとえば、120 に設定した場合、 AWS WAF レートを確認すると、現在の時刻の直前の 2 分間のリクエストがカウントされます。有効な設定は 60 (1 分)、120 (2 分)、300 (5 分)、600 (10 分) で、300 (5 分) がデフォルトです。

  この設定によってレートをチェックする頻度は決まりませんが、 AWS WAF 毎回チェックするたびにどれくらい前に戻るかが決まります。 AWS WAF 評価ウィンドウの設定とは独立したタイミングで、レートを頻繁にチェックします。
- レート制限 — AWS WAF 指定した評価期間中に追跡すべき、条件に一致するリクエストの最大数です。許容される最小制限設定は 100 です。この制限を超えると、 AWS WAF 条件に一致する追加のリクエストにルールアクション設定が適用されます。

  AWS WAF 設定した制限付近にレート制限を適用しますが、制限値と完全に一致することを保証するものではありません。詳細については、「レートベースのルールに関する注意事項」を参照してください。
- リクエスト集約 – レートベースのルールがカウントおよびレート制限するウェブリクエストで使用する集約条件です。設定したレート制限は、各アグリゲーションインスタンスに適用されます。詳細については、「集約オプションおよびキー」および「集約インスタンスおよびカウント」を参照してください。
- アクション – ルールによってレート制限されているリクエストに対して実行するアクションです。Allow 以外のルールアクションを使用できます。これは通常どおりルールレベルで設定されますが、レートベースのルールに固有の制限や動作がいくつかあります。ルールアクションの一般情報については、「ルールアクション」を参照してください。レート制限に固有の情報については、このセクションのを参照してくださいレートベースのルールリクエストレート制限の動作。
- 検査の範囲とレート制限 – スコープダウンステートメントを追加して、レートベースのステートメントが追跡およびレート制限するリクエストの範囲を絞り込むことができます。スコープダウンステートメントを指定すると、ルールはスコープダウンステートメントに一致するリクエストのみを集約、カウント、およびリスト制限します。リクエスト集約オプションの [すべてをカウント] を選択する場合は、スコープダウンステートメントが必要です。スコープダウンステートメントの詳細については、「スコープダウンステートメント」を参照してください。
- (オプション) 転送された IP 設定 – 単独で、またはカスタムキー設定の一部としてリクエスト集約で [ヘッダーの IP アドレス] を指定する場合にのみ使用されます。 AWS WAF は指定されたヘッダーの最初の IP アドレスを取得し、それを集約した値として使用します。この用途によく使用さ

れるヘッダーは X-Forwarded-For ですが、任意のヘッダーを指定できます。詳細については、「転送された IP アドレス」を参照してください。

レートベースのルールに関する注意事項

AWS WAF レート制限は、高いリクエストレートを制御し、アプリケーションの可用性を可能な限り最も効率的かつ効果的に保護するように設計されています。リクエストレートを正確に制限することを意図したものではありません。

- AWS WAF 最近のリクエストを重視するアルゴリズムを使用して、現在のリクエストレートを推定します。そのため、 AWS WAF 設定した制限に近いレート制限を適用しますが、制限値と完全に一致することを保証するものではありません。
- AWS WAF リクエストのレートを見積もるたびに、 AWS WAF 設定した評価期間中に受信したリクエストの数を振り返ります。このほか、伝播遅延などの他の要因により、 AWS WAF リクエストを検出してレート制限する前に、最大数分間、リクエストが高すぎるレートで受信する可能性があります。同様に、 AWS WAF 減少を検出してレート制限アクションを中止するまでの間、リクエストレートが制限値を下回っている可能性があります。通常、この遅延は 30 秒未満です。
- 使用中のルールのレート制限設定を変更すると、その変更によってルールのレート制限カウントがリセットされます。これにより、ルールのレート制限アクティビティが最大 1 分間一時停止する可能性があります。レート制限設定は、評価ウィンドウ、レート制限、リクエストアグリゲーション設定、転送された IP 設定、検査範囲です。

レートベースのルール集約オプションとキー

デフォルトでは、レートベースのルールはリクエスト IP アドレスに基づき、リクエストを集約してレート制限します。他のさまざまな集約キーやキーの組み合わせを使用するようにルールを設定できます。例えば、転送された IP アドレス、HTTP メソッド、またはクエリ引数に基づいて集計できます。IP アドレスや HTTP メソッドなどの集計キーの組み合わせ、または 2 つの異なる Cookie の値を指定することもできます。

> **Note**
>
> リクエストを評価したり、ルールによってレート制限したりするには、集約キーで指定するすべてのリクエストコンポーネントがウェブリクエストに含まれている必要があります。

レートベースのルールは、次の集約オプションを使用して設定できます。

- 送信元 IP アドレス – ウェブリクエストの発信元 IP アドレスのみを使用して集約します。

  送信元 IP アドレスには、発信元クライアントのアドレスが含まれていない場合があります。ウェブリクエストが 1 つ以上のプロキシまたはロードバランサーを経由する場合、これには最後のプロキシのアドレスが含まれます。

- ヘッダーの IP アドレス – HTTP ヘッダー内のクライアントアドレスのみを使用して集約します。これは転送された IP アドレスとも呼ばれます。

  この設定では、ヘッダーに不正な形式の IP アドレスを持つウェブリクエストに適用するフォールバック動作も指定します。フォールバック動作は、リクエストの一致結果を、一致または不一致のいずれにするかを設定します。不一致の場合、レートベースのルールは、リクエストをカウントまたはレート制限しません。一致の場合、レートベースのルールは、指定されたヘッダーに不正な形式の IP アドレスを持つ他のリクエストとともに、リクエストをグループ化します。

  ヘッダーはプロキシによって一貫性なく処理され、検査をバイパスするように変更される可能性があるため、このオプションには注意が必要です。追加情報とベストプラクティスについては、「転送された IP アドレス」を参照してください。

- すべてをカウント – ルールのスコープダウンステートメントに一致するすべてのリクエストをカウントおよびレート制限します。このオプションには、スコープダウンステートメントが必要です。これは通常、特定のラベルが付いた全リクエストや特定の地域からの全リクエストなど、特定のリクエストセットをレート制限するために使用されます。

- カスタムキー – 1 つ以上のカスタム集約キーを使用して集約します。いずれかの IP アドレスオプションを他の集約キーと組み合わせるには、それらをカスタムキーで定義します。

  カスタム集約キーは、「リクエストコンポーネントオプション」で説明されているウェブリクエストコンポーネントオプションのサブセットです。

  キーオプションは次のとおりです。特に明記されていない限り、オプションは複数回使用できます。例えば、2 つのヘッダーや 3 つのラベル名前空間などが使用可能です。

  - ラベル名前空間 – ラベル名前空間を集約キーとして使用します。指定されたラベル名前空間を持つ個別の完全修飾ラベル名はそれぞれ、集約インスタンスに影響します。カスタムキーとしてラベル名前空間を 1 つだけ使用する場合、各ラベル名は集約インスタンスを完全に定義します。

    レートベースのルールが使用するラベルは、ウェブ ACL であらかじめ評価されたルールによってリクエストに追加されたものに限ります。

ラベル名前空間とラベル名の詳細については、「AWS WAF ラベル構文と命名要件」を参照してください。

- ヘッダー – 名前付きヘッダーを集約キーとして使用します。ヘッダー内の個別の値はそれぞれ、集約インスタンスに影響します。

  ヘッダーはオプションでテキスト変換を実行します。テキスト変換オプション を参照してください。

- Cookie – 名前付き Cookie を集約キーとして使用します。Cookie 内の個別の値はそれぞれ、集約インスタンスに影響します。

  Cookie はオプションでテキスト変換を実行します。テキスト変換オプション を参照してください。

- クエリ引数 – リクエスト内で 1 つのクエリ引数を集約キーとして使用します。名前付きクエリ引数の個別の値はそれぞれ、集約インスタンスに影響します。

  クエリ引数はオプションでテキスト変換を実行します。テキスト変換オプション を参照してください。

- クエリ文字列 – リクエスト内のクエリ文字列全体を集約キーとして使用します。個別のクエリ文字列はそれぞれ、集約インスタンスに影響します。このキータイプは一度だけ使用できます。

  クエリ文字列はオプションでテキスト変換を実行します。テキスト変換オプション を参照してください。

- URI パス — リクエスト内の URI パスを集約キーとして使用します。個別の URI パスはそれぞれ、集約インスタンスに影響します。このキータイプは一度だけ使用できます。

  URI パスはオプションでテキスト変換を実行します。テキスト変換オプション を参照してください。

- HTTP メソッド – リクエストの HTTP メソッドを集約キーとして使用します。個別の HTTP メソッドはそれぞれ、集約インスタンスに影響します。このキータイプは一度だけ使用できます。
- IP アドレス – ウェブリクエストの発信元 IP アドレスを他のキーと組み合わせて集約します。

  これには、発信元クライアントのアドレスが含まれていない可能性があります。ウェブリクエストが 1 つ以上のプロキシまたはロードバランサーを経由する場合、これには最後のプロキシのアドレスが含まれます。

- ヘッダーの IP アドレス – HTTP ヘッダー内のクライアントアドレスを他のキーと組み合わせて集約します。これは転送された IP アドレスとも呼ばれます。

このオプションでは、プロキシによって一貫性のないヘッダー処理が行われ、検査を回避するように変更される可能性があるため、注意が必要です。追加情報とベストプラクティスについては、「転送された IP アドレス」を参照してください。

レートベースのルール集約インスタンスとカウント

レートベースのルールが集約条件を使用してウェブリクエストを評価する場合、指定された集約キーに対してルールが検出した一意の値セットはそれぞれ、一意の集約インスタンスを定義します。

- 複数キー - 複数のカスタムキーを定義した場合、各キーの値は集約インスタンスの定義に影響します。値の一意の組み合わせはそれぞれ、集約インスタンスを定義します。
- 単一キー - カスタムキーで単一キーを選択した場合、またはシングルトン IP アドレスの選択肢の中から単一キーを選択した場合、キーの一意の値はそれぞれ、集約インスタンスを定義します。
- すべてをカウント - キーなし - 集約オプションの [すべてをカウント] を選択した場合、ルールが評価するすべてのリクエストは、ルールの 1 つの集約インスタンスに属します。この選択には、スコープダウンステートメントが必要です。

レートベースのルールは、識別された集約インスタンスごとにウェブリクエストを個別にカウントします。

例えば、レートベースのルールが次の IP アドレスと HTTP メソッドの値を持つウェブリクエストを評価するとします。

- IP アドレス 10.1.1.1、HTTP メソッド POST
- IP アドレス 10.1.1.1、HTTP メソッド GET
- IP アドレス 127.0.0.0、HTTP メソッド POST
- IP アドレス 10.1.1.1、HTTP メソッド GET

このルールは、集約条件に従ってさまざまな集約インスタンスを作成します。

- 集約基準が IP アドレスにすぎない場合、個々の IP アドレスは集約インスタンスであり、 はリクエストを個別に AWS WAF カウントします。この例の集約インスタンスおよびリクエストカウントは次のようになります。
  - IP アドレス 10.1.1.1: カウント 3
  - IP アドレス 127.0.0.0: カウント 1

- 集約条件が HTTP メソッドの場合、個々の HTTP メソッドが集約インスタンスになります。この例の集約インスタンスおよびリクエストカウントは次のようになります。
  - HTTP メソッド POST: カウント 2
  - HTTP メソッド GET: カウント 2
- 集約条件が IP アドレスと HTTP メソッドの場合、各 IP アドレスと各 HTTP メソッドは、複合的な集約インスタンスに影響します。この例の集約インスタンスおよびリクエストカウントは次のようになります。
  - IP アドレス 10.1.1.1、HTTP メソッド POST: カウント 1
  - IP アドレス 10.1.1.1、HTTP メソッド GET: カウント 2
  - IP アドレス 127.0.0.0、HTTP メソッド POST: カウント 1

レートベースのルールリクエストレート制限の動作

AWS WAF がレートベースのルールのリクエストをレート制限するために使用する基準は、 AWS WAF がルールのリクエストを集約するために使用する基準と同じです。ルールのスコープダウンステートメントを定義すると、 AWS WAF はスコープダウンステートメントに一致するリクエストのみを集計、カウント、レート制限します。

レートベースのルールが特定のウェブリクエストにルールアクション設定を適用する一致条件は、次のとおりです。

- ウェブリクエストは、ルールのスコープダウンステートメントと一致している（定義されている場合）。
- ウェブリクエストは、現在リクエストカウントがルールの制限を超えている集約インスタンスに属している。

がルールアクション AWS WAF を適用する方法

レートベースのルールがリクエストにレート制限を適用すると、ルールアクションが適用され、アクション仕様でカスタム処理またはラベル付けを定義している場合、ルールはそれらを適用します。このリクエスト処理は、一致ルールが一致したウェブリクエストにアクション設定を適用する方法と同じです。レートベースのルールは、レート制限が積極的に行われているリクエストにのみラベルを適用したり、他のアクションを実行したりします。

Allow 以外のルールアクションを使用できます。ルールアクションの一般情報については、「ルールアクション」を参照してください。

次のリストは、各アクションのレート制限の仕組みを示しています。

- Block – リクエストを AWS WAF ブロックし、定義したカスタムブロック動作を適用します。
- Count – リクエストを AWS WAF カウントし、定義したカスタムヘッダーまたはラベルを適用し、リクエストのウェブ ACL 評価を続行します。

  このアクションはリクエストのレートを制限しません。制限を超えるリクエストのみをカウントします。
- CAPTCHA または Challenge - AWS WAF はリクエストのトークンの状態に応じて、Count または Block のいずれかのようにリクエストを処理します。

  このアクションは、有効なトークンを持つリクエストのレートを制限しません。これにより、制限を超え、有効なトークンも欠落しているリクエストのレートが制限されます。
  - リクエストに有効期限が切れていない有効なトークンがない場合、このアクションはリクエストをブロックし、CAPTCHA パズルまたはブラウザのチャレンジをクライアントに返送します。

    エンドユーザーまたはクライアントブラウザが正常に応答すると、クライアントは有効なトークンを受け取り、元のリクエストが自動的に再送信されます。集約インスタンスのレート制限がまだ有効である場合、有効期限が切れていない有効なトークンを含むこの新しいリクエストには、次の箇条書きで説明するアクションが適用されます。
  - リクエストに有効期限が切れていない有効なトークンがある場合、CAPTCHA また Challenge アクションはトークンを検証し、Count アクションと同様にリクエストに対してアクションを実行しません。レートベースのルールは、終了アクションを実行せずにリクエスト評価をウェブ ACL に返し、ウェブ ACL はリクエストの評価を続行します。

  詳細については、「CAPTCHAChallengeの および AWS WAF」を参照してください。

IP アドレスまたは転送された IP アドレスのみをレート制限する場合

転送された IP アドレスに対して、IP アドレスのみをレート制限するようにルールを設定した場合、ルールインスタンスは最大 10,000 個の IP アドレスをレート制限できます。ルールインスタンスで 10,000 個を超える IP アドレスがレート制限の対象として識別された場合、送信レートの速い上位 10,000 個のみが制限されます。

この設定では、レートベースのルールが現在レート制限している IP アドレスのリストを取得できます。スコープダウンステートメントを使用している場合、レート制限されているリクエストは、スコープダウンステートメントに一致する IP リスト内のリクエストのみです。IP アドレスリストの取

得の詳細については、「レートベースのルールによってレート制限されている IP アドレスの一覧表示」を参照してください。

レートベースのルールの例

このセクションでは、一般的なレートベースのルールのさまざまなユースケースにおける設定例について説明します。

各例は、ユースケースの説明を提供し、カスタム設定ルールの JSON リストにそのソリューションを示します。

> **Note**
>
> これらの例に示されている JSON リストは、ルールを設定し、Rule JSON エディタを使用して編集することにより、コンソールで作成されました。

トピック

- ログインページへのリクエストをレート制限する
- 任意の IP アドレス、ユーザーエージェントペアからのログインページへのリクエストをレート制限する
- 特定のヘッダーが欠落しているリクエストをレート制限する
- 特定のラベルを使用してリクエストをレート制限する
- ラベル名前空間が指定されたラベルのリクエストをレート制限する

ログインページへのリクエストをレート制限する

ウェブサイトのログインページへのリクエスト数を、サイトの他の部分へのトラフィックに影響を与えることなく制限するには、ログインページへのリクエストに一致するスコープダウンステートメントを含むレートベースのルールを作成し、リクエスト集約を [すべてをカウント] に設定します。

レートベースのルールは、ログインページへのリクエストすべてを 1 つの集約インスタンスでカウントし、リクエストが制限を超えるとルールアクションを適用します。

次の JSON リストは、このルール設定の例を示しています。集約をすべてカウントするオプションは、`CONSTANT` 設定として JSON に記載されています。この例は、`/login` で始まるログインページと一致します。

```
{
```

```
  "Name": "test-rbr",
  "Priority": 0,
  "Action": {
    "Block": {}
  },
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "test-rbr"
  },
  "Statement": {
    "RateBasedStatement": {
      "Limit": 1000,
      "EvaluationWindowSec": 300,
      "AggregateKeyType": "CONSTANT",
      "ScopeDownStatement": {
        "ByteMatchStatement": {
          "FieldToMatch": {
            "UriPath": {}
          },
          "PositionalConstraint": "STARTS_WITH",
          "SearchString": "/login",
          "TextTransformations": [
            {
              "Type": "NONE",
              "Priority": 0
            }
          ]
        }
      }
    }
  }
}
```

任意の IP アドレス、ユーザーエージェントペアからのログインページへのリクエストをレート制限する

IP アドレス、ユーザーエージェントペアが制限を超えた場合に、ウェブサイトのログインページへのリクエスト数を制限するには、リクエスト集約を [カスタムキー] に設定して集約条件を指定します。

次の JSON リストは、このルール設定の例を示しています。この例では、IP アドレス、ユーザーエージェントペアあたり、5 分間で 100 リクエストに制限を設定しています。

```
{
  "Name": "test-rbr",
  "Priority": 0,
  "Action": {
    "Block": {}
  },
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "test-rbr"
  },
  "Statement": {
    "RateBasedStatement": {
      "Limit": 100,
      "EvaluationWindowSec": 300,
      "AggregateKeyType": "CUSTOM_KEYS",
      "CustomKeys": [
        {
          "Header": {
            "Name": "User-Agent",
            "TextTransformations": [
              {
                "Priority": 0,
                "Type": "NONE"
              }
            ]
          }
        },
        {
          "IP": {}
        }
      ],
      "ScopeDownStatement": {
        "ByteMatchStatement": {
          "FieldToMatch": {
            "UriPath": {}
          },
          "PositionalConstraint": "STARTS_WITH",
          "SearchString": "/login",
          "TextTransformations": [
            {
              "Type": "NONE",
              "Priority": 0
```

```
            }
          ]
        }
      }
    }
  }
}
```

特定のヘッダーが欠落しているリクエストをレート制限する

特定のヘッダーが欠落しているリクエスト数を制限するには、スコープダウンステートメントで [すべてをカウント] 集約オプションを使用できます。ヘッダーが存在していて値がある場合にのみ true を返すステートメントを含む、論理 NOT ステートメントを使用してスコープダウンステートメントを設定します。

次の JSON リストは、このルール設定の例を示しています。

```
{
  "Name": "test-rbr",
  "Priority": 0,
  "Action": {
    "Block": {}
  },
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "test-rbr"
  },
  "Statement": {
    "RateBasedStatement": {
      "Limit": 1000,
      "AggregateKeyType": "CONSTANT",
      "EvaluationWindowSec": 300,
      "ScopeDownStatement": {
        "NotStatement": {
          "Statement": {
            "SizeConstraintStatement": {
              "FieldToMatch": {
                "SingleHeader": {
                  "Name": "user-agent"
                }
              },
              "ComparisonOperator": "GT",
```

```
              "Size": 0,
              "TextTransformations": [
                {
                  "Type": "NONE",
                  "Priority": 0
                }
              ]
            }
          }
        }
      }
    }
  }
}
```

特定のラベルを使用してリクエストをレート制限する

リクエストにラベルを追加する任意のルールまたはルールグループとレート制限を組み合わせると、さまざまなカテゴリのリクエスト数を制限できます。そのためには、ウェブ ACL を次のように設定します。

- ラベルを追加するルールまたはルールグループを追加し、レート制限するリクエストがブロックまたは許可されないように設定します。マネージドルールグループを使用する場合、この動作を実現するために一部のルールグループのルールアクションを Count にオーバーライドすることが必要になる場合があります。
- ラベル付けルールやルールグループよりも高い優先度番号を設定して、レートベースのルールをウェブ ACL に追加します。 AWS WAF ルールを番号の低いものから順に評価するので、レートベースのルールはラベリングルールの後に実行されます。ルールのスコープダウンステートメントのラベル一致とラベル集約を組み合わせて、ラベルのレート制限を設定します。

次の例では、Amazon IP AWS レピュテーションリストマネージドルールルールグループを使用しています。このルールグループのルールである `AWSManagedIPDDoSList` は、IP が DDoS 攻撃に積極的に関与していることがわかっているリクエストを検出し、ラベルを付けます。ルールのアクションは、ルールグループ定義で Count に設定されています。ルールグループの詳細については、「the section called “Amazon IP 評価リスト”」を参照してください。

次のウェブ ACL JSON リストでは、IP 評価ルールグループの後にラベル一致のレートベースルールが続いています。レートベースのルールでは、スコープダウンステートメントを使用して、ルールグループのルールでマークされたリクエストをフィルタリングします。レートベースのルールステートメントは、フィルタリングされたリクエストを IP アドレスで集約してレート制限します。

```
{
  "Name": "test-web-acl",
  "Id": ...
  "ARN": ...
  "DefaultAction": {
    "Allow": {}
  },
  "Description": "",
  "Rules": [
    {
      "Name": "AWS-AWSManagedRulesAmazonIpReputationList",
      "Priority": 0,
      "Statement": {
        "ManagedRuleGroupStatement": {
          "VendorName": "AWS",
          "Name": "AWSManagedRulesAmazonIpReputationList"
        }
      },
      "OverrideAction": {
        "None": {}
      },
      "VisibilityConfig": {
        "SampledRequestsEnabled": true,
        "CloudWatchMetricsEnabled": true,
        "MetricName": "AWS-AWSManagedRulesAmazonIpReputationList"
      }
    },
    {
      "Name": "test-rbr",
      "Priority": 1,
      "Statement": {
        "RateBasedStatement": {
          "Limit": 100,
          "EvaluationWindowSec": 300,
          "AggregateKeyType": "IP",
          "ScopeDownStatement": {
            "LabelMatchStatement": {
              "Scope": "LABEL",
              "Key": "awswaf:managed:aws:amazon-ip-list:AWSManagedIPDDoSList"
            }
          }
        }
      },
```

```
      "Action": {
        "Block": {}
      },
      "VisibilityConfig": {
        "SampledRequestsEnabled": true,
        "CloudWatchMetricsEnabled": true,
        "MetricName": "test-rbr"
      }
    }
  ],
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "test-web-acl"
  },
  "Capacity": 28,
  "ManagedByFirewallManager": false,
  "LabelNamespace": "awswaf:0000000000:webacl:test-web-acl:"
}
```

ラベル名前空間が指定されたラベルのリクエストをレート制限する

Bot Control マネージドルールグループの共通レベルルールは、さまざまなカテゴリのボットにラベルを追加しますが、ブロックするのは未検証ボットからのリクエストに限ります。これらのルールの詳細については、「Bot Control のルールリスト」を参照してください。

Bot Control マネージドルールグループを使用する場合、個々の検証済みボットからのリクエストにレート制限を追加できます。そのためには、Bot Control ルールグループの後に実行されて、リクエストをボット名のラベル別に集約するレートベースのルールを追加します。[ラベル名前空間] 集約キーを指定し、名前空間キーを `awswaf:managed:aws:bot-control:bot:name:` と設定します。指定された名前空間を持つ一意のラベルはそれぞれ、集約インスタンスを定義します。例えば、`awswaf:managed:aws:bot-control:bot:name:axios` と `awswaf:managed:aws:bot-control:bot:name:curl` というラベルは、それぞれに集約インスタンスを定義します。

次のウェブ ACL JSON リストは、この設定を示しています。この例のルールでは、1 つのボット集約インスタンスのリクエストを 2 分間で 1,000 に制限しています。

```
{
  "Name": "test-web-acl",
  "Id": ...
  "ARN": ...
  "DefaultAction": {
```

```
    "Allow": {}
  },
  "Description": "",
  "Rules": [
    {
      "Name": "AWS-AWSManagedRulesBotControlRuleSet",
      "Priority": 0,
      "Statement": {
        "ManagedRuleGroupStatement": {
          "VendorName": "AWS",
          "Name": "AWSManagedRulesBotControlRuleSet",
          "ManagedRuleGroupConfigs": [
            {
              "AWSManagedRulesBotControlRuleSet": {
                "InspectionLevel": "COMMON"
              }
            }
          ]
        }
      },
      "OverrideAction": {
        "None": {}
      },
      "VisibilityConfig": {
        "SampledRequestsEnabled": true,
        "CloudWatchMetricsEnabled": true,
        "MetricName": "AWS-AWSManagedRulesBotControlRuleSet"
      }
    },
    {
      "Name": "test-rbr",
      "Priority": 1,
      "Statement": {
        "RateBasedStatement": {
          "Limit": 1000,
          "EvaluationWindowSec": 120,
          "AggregateKeyType": "CUSTOM_KEYS",
          "CustomKeys": [
            {
              "LabelNamespace": {
                "Namespace": "awswaf:managed:aws:bot-control:bot:name:"
              }
            }
          ]
```

```
        }
      },
      "Action": {
        "Block": {}
      },
      "VisibilityConfig": {
        "SampledRequestsEnabled": true,
        "CloudWatchMetricsEnabled": true,
        "MetricName": "test-rbr"
      }
    }
  ],
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "test-web-acl"
  },
  "Capacity": 82,
  "ManagedByFirewallManager": false,
  "LabelNamespace": "awswaf:0000000000:webacl:test-web-acl:"
}
```

レートベースのルールによってレート制限されている IP アドレスの一覧表示

レートベースのルールが IP アドレスまたは転送された IP アドレスのみに基づいて集計される場合は、ルールが現在レート制限している IP アドレスのリストを取得できます。 AWS WAF これらの IP アドレスをルールの管理キーリストに保存します。

> **Note**
>
> このオプションは、IP アドレスのみ、またはヘッダーの IP アドレスのみを集約する場合に限り使用できます。カスタムキーのリクエスト集約を使用する場合、カスタムキーでいずれかの IP アドレス仕様を使用しても、レート制限されている IP アドレスのリストは取得できません。

レートベースのルールは、そのルールのスコープダウンステートメントと一致する、ルールのマネージドキーリストからのリクエストにルールアクションを適用します。ルールにスコープダウンステートメントがない場合は、リストに含まれている IP アドレスからのすべてのリクエストにアクションが適用されます。ルールアクションはデフォルトで Block ですが、Allow を除く有効なルールアクションであればどれでもかまいません。1 AWS WAF つのレートベースのルールインスタンスを使用

してレート制限できる IP アドレスの最大数は 10,000 です。10,000 個を超えるアドレスがレート制限を超える場合は、レートが最も高い IP AWS WAF アドレスを制限します。

CLI、API、または任意の SDK を使用して、レートベースのルールのマネージドキーリストにアクセスできます。このトピックでは、CLI および API を使用したアクセスについて説明します。現時点では、コンソールからリストにアクセスすることはできません。

AWS WAF API の場合、コマンドはですGetRateBasedStatementManagedKeys。

AWS WAF CLI の場合、コマンドは get-rate-based-statement-managed-keys です。

以下は、Amazon ディストリビューションのウェブ ACL で使用されているレートベースのルールのレート制限 IP アドレスのリストを取得するための構文を示しています。 CloudFront

```
aws wafv2 get-rate-based-statement-managed-keys --scope=CLOUDFRONT --region=us-east-1
 --web-acl-name=WebACLName --web-acl-id=WebACLId --rule-name=RuleName
```

以下は、リージョンアプリケーション、Amazon API ゲートウェイ REST API、Application Load Balancer、 AWS AppSync GraphQL API、Amazon Cognito ユーザープール、 AWS App Runner サービス、 AWS または検証済みアクセスインスタンスの構文を示しています。

```
aws wafv2 get-rate-based-statement-managed-keys --scope=REGIONAL --region=region --web-
acl-name=WebACLName --web-acl-id=WebACLId --rule-name=RuleName
```

AWS WAF ウェブリクエストをモニタリングし、ウェブ ACL、オプションのルールグループ、レートベースのルールの組み合わせごとに個別にキーを管理します。例えば、ルールグループ内でレートベースのルールを定義してから、ウェブ ACL でルールグループを使用すると、 AWS WAF はウェブリクエストをモニタリングし、そのウェブ ACL、ルールグループ参照ステートメント、およびレートベースのルールインスタンスのキーを管理できます。同じルールグループを 2 つ目のウェブ ACL で使用すると、 AWS WAF ウェブリクエストを監視し、この 2 回目の使用に関するキーを 1 回目のウェブ ACL とは完全に独立して管理します。

ルールグループ内で定義したレートベースのルールの場合は、ルールグループ内のウェブ ACL 名とレートベースのルール名に加えて、リクエストでルールグループ参照ステートメントの名前を指定する必要があります。レートベースのルールがルールグループ内で定義され、ウェブ ACL でルールグループが使用されるリージョンレベルのアプリケーションの構文を次に示します。

```
aws wafv2 get-rate-based-statement-managed-keys --scope=REGIONAL --region=region --web-
acl-name=WebACLName --web-acl-id=WebACLId --rule-group-rule-name=RuleGroupRuleName --
rule-name=RuleName
```

ルールグループのルールステートメント

ルールグループのルールステートメントはネストできません。

このセクションでは、ウェブ ACL で使用できるルールグループのルールステートメントについて説明します。ルールグループのウェブ ACL キャパシティーユニット (WCU) は、ルールグループの作成時にルールグループの所有者によって設定されます。WCU の詳細については、「AWS WAF ウェブ ACL キャパシティーユニット (WCUs)」を参照してください。

ルールグループステートメント	説明	WCU
マネージドルールグループ	指定されたマネージドルールグループで定義されているルールを実行します。 スコープダウンステートメントを追加することで、ルールグループで評価されるリクエストの範囲を絞り込むことができます。 マネージドルールグループステートメントは、他のステートメントタイプ内にネストできません。	ルールグループと、スコープダウンステートメント用の追加の WCU によって定義されます。
ルールグループ	管理するルールグループで定義されているルールを実行します。 独自のルールグループのルールグループ参照ステートメントにスコープダウンステートメントを追加することはできません。	WCU の制限は、ルールグループを作成するときに定義します。

ルールグループステートメント	説明	WCU
	ルールグループステートメントは、他のステートメントタイプ内にネストできません。	

マネージドルールグループステートメント

マネージドルールグループルールステートメントは、ウェブ ACL ルールリストに含まれる参照をマネージドルールグループに追加します。コンソールのルールステートメントにはこのオプションは表示されませんが、ウェブ ACL の JSON 形式を操作しているときには、追加したマネージドルールグループがこのタイプとしてウェブ ACL ルールに表示されます。

マネージドルールグループは、 AWS マネージドルールルールグループ（ AWS WAF そのほとんどは顧客向け）AWS Marketplace またはマネージドルールグループのいずれかです。 AWS 有料のマネージドルールルールグループをウェブ ACL に追加すると、自動的にそのルールグループに登録されます。 AWS Marketplace マネージドルールグループにはを通じて登録できます AWS Marketplace。詳細については、「マネージドルールグループ」を参照してください。

ルールグループをウェブ ACL に追加すると、グループ内のルールのアクションを Count またはその他のルールアクションにオーバーライドできます。詳細については、「ルールグループのアクションオーバーライドオプション」を参照してください。

AWS WAF ルールグループで評価されるリクエストの範囲を絞り込むことができます。これを行うには、ルールグループステートメント内にスコープダウンステートメントを追加します。スコープダウンステートメントの詳細については、「スコープダウンステートメント」を参照してください。これは、ルールグループがトラフィックに与える影響を管理し、ルールグループを使用するときにトラフィック量に関連するコストを抑えるのに役立ちます。 AWS WAF Bot Control が管理するルールグループで scope-down ステートメントを使用する方法の詳細と例については、を参照してください。AWS WAF ボットコントロール

ネスト不可 - このステートメントタイプを他のステートメント内にネストしたり、ルールグループに含めたりすることはできません。このタイプはウェブ ACL に直接含めることができます。

(オプション) スコープダウンステートメント – このルールタイプは、オプションのスコープダウンステートメントを使用して、ルールグループが評価するリクエストの範囲を絞り込みます。詳細については、「スコープダウンステートメント」を参照してください。

WCU - 作成時にルールグループに設定します。

このルールステートメントの場所

- コンソール - ウェブ ACL の作成プロセス中に、[Add rules and rule groups] (ルールとルールグループの追加) ページで [Add managed rule groups] (マネージドルールグループの追加) を選択し、使用するルールグループを見つけて選択します。
- API — ManagedRuleGroupStatement

ルールグループステートメント

ルールグループルールステートメントは、ウェブ ACL ルールリストへの参照を、管理されているルールグループに追加します。コンソールのルールステートメントにはこのオプションは表示されませんが、ウェブ ACL の JSON 形式を操作しているときには、追加した独自のルールグループがこのタイプとしてウェブ ACL ルールに表示されます。独自のルールグループの使用については、「独自のルールグループの管理」を参照してください。

ルールグループをウェブ ACL に追加すると、グループ内のルールのアクションを Count またはその他のルールアクションにオーバーライドできます。詳細については、「ルールグループのアクションオーバーライドオプション」を参照してください。

ネスト不可 - このステートメントタイプを他のステートメント内にネストしたり、ルールグループに含めたりすることはできません。このタイプはウェブ ACL に直接含めることができます。

WCU - 作成時にルールグループに設定します。

このルールステートメントの場所

- コンソール - ウェブ ACL の作成プロセス中に、[Add rules and rule groups] (ルールとルールグループの追加) ページで、[Add my own rules and rule groups] (独自のルールとルールグループの追加)、[Rule group] (ルールグループ) を選択し、使用するルールグループを追加します。
- API — RuleGroupReferenceStatement

でのオーバーサイズリクエストコンポーネントの処理 AWS WAF

AWS WAF は、ウェブリクエストコンポーネント本文、ヘッダー、または Cookie の非常に大きなコンテンツの検査をサポートしていません。基盤となるホストサービスには、検査 AWS WAF のために に転送されるものの数とサイズの制限があります。例えば、ホストサービスは に 200 個を超え

るヘッダーを送信しないため AWS WAF、205 個のヘッダーを持つウェブリクエストの場合、 AWS WAF は最後の 5 個のヘッダーを検査できません。

がウェブリクエストを保護されたリソースに進ませること AWS WAF を許可すると、検査できた力ウントとサイズの制限外のコンテンツを含むウェブリクエスト全体が送信 AWS WAF されます。

コンポーネント検査のサイズ制限

コンポーネント検査のサイズ制限は次のとおりです。

- **`Body`** および **`JSON Body`** – Application Load Balancer および の場合 AWS AppSync、 はリクエストの本文の最初の 8 KB を検査 AWS WAF できます。 CloudFront、API Gateway、Amazon CognitoApp Runner、Verified Access では、デフォルトで最初の 16 KB を検査 AWS WAF でき、ウェブ ACL 設定で制限を最大 64 KB まで増やすことができます。詳細については、「本文検査のサイズ制限の管理」を参照してください。
- **`Headers`** - リクエストヘッダーの最初の 8 KB (8,192 バイト) まで、および最初の 200 個のヘッダーまで検査 AWS WAF できます。コンテンツは、最初の制限に達する AWS WAF まで検査できます。
- **`Cookies`** - リクエスト Cookie の最初の 8 KB (8,192 バイト) まで、および最初の 200 個の Cookie まで検査 AWS WAF できます。コンテンツは、最初の制限に達する AWS WAF まで検査できます。

ルールステートメントのオーバーサイズの処理オプション

これらのリクエストコンポーネントタイプのいずれかを検査するルールステートメントを記述するときは、オーバーサイズコンポーネントの処理方法を指定します。オーバーサイズ処理は、ルールが検査するリクエストコンポーネントがサイズ制限を超えたときにウェブリクエストで AWS WAF 何をするかを指示します。

オーバーサイズコンポーネントを処理するためのオプションは次のとおりです。

- Continue – ルール検査基準に従って、リクエストコンポーネントを正常に検査します。 AWS WAF は、サイズ制限内にあるリクエストコンポーネントのコンテンツを検査します。
- Match – ウェブリクエストをルールステートメントに一致するものとして扱います。 AWS WAF は、ルールの検査基準に照らして評価することなく、ルールアクションをリクエストに適用します。

- No match – ウェブリクエストは、ルールの検査基準に照らして評価せずに、ルールステートメントと一致しないものとして扱います。 は、一致しないルールの場合と同様に、ウェブ ACL 内の残りのルールを使用して、ウェブリクエストの検査 AWS WAF を続行します。

AWS WAF コンソールでは、これらの処理オプションのいずれかを選択する必要があります。コンソールの外では、デフォルトのオプションは Continue です。

Block に設定されたアクションを含むルールで Match オプションを使用する場合、そのルールは、検査したコンポーネントがオーバーサイズであるリクエストをブロックします。その他の設定では、リクエストの最終的な処理は、ウェブ ACL 内の他のルールの設定や、ウェブ ACL のデフォルトのアクション設定など、さまざまな要因によって異なります。

ユーザーが所有していないルールグループでのオーバーサイズの処理

コンポーネントのサイズと数の制限は、ウェブ ACL で使用するすべてのルールに適用されます。これには、マネージドルールグループ内および別のアカウントと共有しているルールグループ内にある、使用しているが管理されていないルールが含まれます。

ユーザーが管理していないルールグループを使用する場合、ルールグループに制限されたリクエストコンポーネントを検査するルールがあっても、そのルールではオーバーサイズのコンテンツが必要な方法で処理されない場合があります。 AWS マネージドルールがオーバーサイズコンポーネントを管理する方法については、「」を参照してくださいAWS マネージドルールルールグループリスト。他のルールグループの詳細については、ルールグループプロバイダーにお問い合わせください。

ウェブ ACL 内のオーバーサイズコンポーネントを管理するためのガイドライン

ウェブ ACL でオーバーサイズのコンポーネントを処理する方法は、リクエストコンポーネントのコンテンツの予想サイズ、ウェブ ACL のデフォルトリクエスト処理、ウェブ ACL 内の他のルールがリクエストと一致して処理する方法など、さまざまな要因によって異なります。

オーバーサイズのウェブリクエストコンポーネントを管理するための一般的なガイドラインは、以下のとおりです。

- オーバーサイズのコンポーネントコンテンツを含む一部のリクエストを許可する必要がある場合は、可能な場合は、それらのリクエストのみを明示的に許可するルールを追加します。同じコンポーネントタイプを検査するウェブ ACL 内の他のルールより先に実行されるように、これらのルールの優先度を上げます。この方法では、 AWS WAF を使用して、保護されたリソースに渡すことを許可するオーバーサイズコンポーネントのコンテンツ全体を検査することはできません。
- 他のすべてのリクエストでは、次のように、制限を超えるリクエストをブロックすることで、余計なバイトが通過するのを防ぐことができます。

- ルールとルールグループ – サイズ制限のあるコンポーネントを検査するルールで、制限を超過するリクエストをブロックするようにオーバーサイズの処理を設定します。例えば、ルールで特定のヘッダーコンテンツを含むリクエストをブロックする場合、オーバーサイズのヘッダーコンテンツを持つリクエストと一致するようにオーバーサイズの処理を設定できます。別の例として、ウェブ ACL によりデフォルトでリクエストがブロックされ、かつルールで特定のヘッダーコンテンツが許可されている場合、オーバーサイズのヘッダーコンテンツを持つすべてのリクエストと一致しないように、ルールのオーバーサイズ処理を設定できます。
- 管理していないルールグループ – 管理していないルールグループでオーバーサイズのリクエストコンポーネントを許可しないようにするには、リクエストコンポーネントタイプを検査して制限を超えるリクエストをブロックする別のルールを追加します。ウェブ ACL でそのルールがルールグループより先に実行されるように、ルールの優先度を上げます。例えば、ウェブ ACL で本文検査ルールを実行する前に、オーバーサイズの本文コンテンツを含むリクエストをブロックできます。次の手順では、このタイプのルールを追加する方法について説明します。

オーバーサイズのウェブリクエストコンポーネントのブロック

オーバーサイズのコンポーネントを含むリクエストをブロックするルールをウェブ ACL に追加できます。

オーバーサイズのコンテンツをブロックするルールを追加するには

1. ウェブ ACL を作成または編集するときは、ルール設定で、[Add rules] (ルールを追加)、[Add my own rules and rule groups] (独自のルールとルールグループを追加)、[Rule builder] (ルールビルダー)、[Rule visual editor] (ルールビジュアルエディタ) の順に選択します。ウェブ ACL の作成または編集に関するガイダンスについては、「ウェブ ACL の使用」を参照してください。
2. ルールの名前を入力し、[Type] (タイプ) 設定を [Regular rule] (通常のルール) のままにします。
3. 次の一致設定をデフォルトから変更します。

    a. [Statement] (ステートメント) の [Inspect] (検査) で、ドロップダウンを開き、必要なウェブリクエストコンポーネント ([Body] (本文)、[Headers] (ヘッダー)、[Cookies] (cookie) のいずれか) を選択します。

    b. [Match type] (一致タイプ) で、[Size greater than] (次より大きいサイズ:) を選択します。

    c. [サイズ] で、コンポーネントタイプの最小サイズ以上の数値を入力します。ヘッダーと Cookie の場合は、 と入力します8192。Application Load Balancer または AWS AppSync ウェブ ACLs、本文に と入力します8192。 CloudFront、API Gateway、Amazon CognitoApp Runner、または Verified Access ウェブ ACLs の本文で、デフォルトの本文サ

イズ制限を使用している場合は、 と入力します16384。それ以外の場合は、ウェブ ACL に定義した本文のサイズ制限を入力します。

d. [Oversize handling] (オーバーサイズ処理) で、[Match] (一致) を選択します。

4. [Action] (アクション) で、[Block] (ブロック) を選択します。
5. [Add Rule] (ルールの追加) を選択します。
6. ルールを追加したら、[Set rule priority] (ルールの優先度の設定) ページで、同じコンポーネントタイプを検査するウェブ ACL 内のルールまたはルールグループよりも上に移動します。これにより、新しいルールの優先順位の数値が低くなり、 AWS WAF が最初に評価します。詳細については、「ウェブ ACL でのルールおよびルールグループの処理順序」を参照してください。

での正規表現パターンマッチング AWS WAF

AWS WAF PCRE `libpcre` ライブラリが使用するパターン構文をサポートします。ライブラリは、「PCRE - Perl Compatible Regular Expressions」で文書化されています。

AWS WAF ライブラリのすべての構成をサポートしているわけではありません。例えば、一部のゼロ幅アサーションをサポートしますが、すべてではありません。サポートされているコンストラクトの包括的なリストはありません。ただし、有効でない正規表現パターンを指定したり、サポートされていない構成を使用したりすると、API は失敗を報告します。 AWS WAF

AWS WAF 次の PCRE パターンはサポートされていません。

- 後方参照と部分式取得
- サブルーチン参照と再帰パターン
- 条件付きパターン
- バックトラック制御動詞
- \C シングルバイトディレクティブ
- \R 改行一致ディレクティブ
- \K 一致開始位置リセットディレクティブ
- コールアウトと埋め込みコード
- アトミックグループと所有格量指定子

の IP セットと正規表現パターンセット AWS WAF

AWS WAF は、より複雑な情報を、ルールで参照して使用するセットに保存します。これらのセットにはそれぞれ名前があり、作成時に Amazon リソースネーム (ARN) が割り当てられます。これらのセットは、ルールステートメント内から管理でき、コンソールのナビゲーションペインから単独でアクセスして管理できます。

マネージドセットは、ルールグループまたはウェブ ACL で使用できます。

- IP セットを使用するには、「」を参照してくださいIP セット一致ルールステートメント。
- 正規表現パターンセットを使用するには、「」を参照してください正規表現パターンセット一致ルールステートメント。

更新中の一時的な不一致

ウェブ ACL やその他の AWS WAF リソースを作成または変更する場合、リソースが保存されているすべての領域にその変更が反映されるまでに少し時間がかかります。伝播時間は、数秒から数分までかかります。

次の内容では、変更伝播中に直面する一時的な不整合性の例を紹介します。

- ウェブ ACL を作成した後、それをリソースに関連付けようとすると、ウェブ ACL が利用できないことを示す例外が表示される場合があります。
- ルールグループをウェブ ACL に追加した後、新しいルールグループのルールは、ウェブ ACL が使用されるエリアで有効になり、別のエリアでは有効にならない場合があります。
- ルールのアクション設定を変更した後、古いアクションを一部のエリアで確認され、新しいアクションを別のエリアで確認される場合があります。
- ブロックルールで使用されている IP セットに IP アドレスを追加した後、新しいアドレスはあるエリアではブロックされ、別のエリアでは許可される場合があります。

トピック

- IP セットの作成と管理
- 正規表現パターンセットの作成と管理

IP セットの作成と管理

IP セットは、ルールステートメントで一緒に使用する IP アドレスと IP アドレス範囲のコレクションを提供します。IP セットは AWS リソースです。

ウェブ ACL またはルールグループで IP セットを使用するには、まずアドレス仕様`IPSet`で AWS リソースを作成します。その後、IP セットルールステートメントをウェブ ACL またはルールグループに追加するときに、このセットを参照します。

トピック

IP セットの作成

新しい IP セットを作成するには、このセクションの手順に従います。

> **Note**
>
> このセクションの手順に加えて、IP 一致ルールをウェブ ACL またはルールグループに追加するときに、新しい IP セットを追加するオプションがあります。このオプションを選択する場合は、この手順で必要な設定と同じ設定を指定する必要があります。

IP セットを作成するには

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/wafv2/ で AWS WAF コンソールを開きます。
2. ナビゲーションペインで、[IP sets] (IP セット) を選択し、[Create IP set] (IP セットの作成) を選択します。
3. IP セットの名前と説明を入力します。これらを使用して、セットを使用するときにそのセットを識別します。

   > **Note**
   >
   > IP セットの作成後は、名前を変更できません。

4. リージョン で、グローバル (CloudFront) を選択するか、IP セットを保存するリージョンを選択します。リージョン IP セットは、リージョンのリソースを保護するウェブ ACL でのみ使用できます。Amazon CloudFront ディストリビューションを保護するウェブ ACLsで IP セットを使用するには、グローバル () を使用する必要がありますCloudFront。
5. [IP version] (IP バージョン) で、使用するバージョンを選択します。
6. IP アドレステキストボックスに、CIDR notation に 1 行あたり 1 つの IP アドレスまたは IP アドレス範囲を入力します。 は、 を除くすべての IPv4 および IPv6 CIDR 範囲 AWS WAF をサポートします`/0`。CIDR 表記の詳細については、Wikipedia の「Classless Inter-Domain Routing」(クラスレスドメイン間ルーティング) 記事を参照してください。

   次に例を示します。

   - IPv4 アドレス 192.0.2.44 を指定するには、192.0.2.44/32 と入力します。
   - IPv6 アドレス 2620:0:2d0:200:0:0:0:0 を指定するには、2620:0:2d0:200:0:0:0:0/128 と入力します。
   - IPv4 アドレス範囲 192.0.2.0 ~ 192.0.2.255 を指定するには、192.0.2.0/24 と入力します。
   - IPv6 アドレス範囲の 2620:0:2d0:200:0:0:0:0 ~ 2620:0:2d0:200:ffff:ffff:ffff:ffff を指定するには、2620:0:2d0:200::/64 と入力します。
7. IP セットの設定を確認し、[Create IP set] (IP セットの作成) を選択します。

IP セットの削除

参照セットを削除するには、このセクションのガイダンスに従います。

参照セットとルールグループの削除

IP セット、正規表現パターンセット、ルールグループなど、ウェブ ACL で使用できるエンティティを削除すると、 はエンティティがウェブ ACL で現在使用されている AWS WAF かどうかを確認します。使用中であることがわかった場合、 は AWS WAF ユーザーに警告します。 AWS WAF は、エンティティがウェブ ACL によって参照されているかどうかをほとんどの場合判断できます。ただし、まれに判別できないことがあります。エンティティが現在使用中でないことを確認する必要がある場合は、削除する前にウェブ ACL でそのエンティティを確認してください。エンティティが参照されているセットである場合は、ルールグループでエンティティが使用されていないことも確認してください。

IP セットを削除するには

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/wafv2/ で AWS WAF コンソールを開きます。
2. ナビゲーションペインで [IP sets] (IP セット) を選択します。
3. 削除する IP セットを選択し、[Delete] (削除) を選択します。

正規表現パターンセットの作成と管理

正規表現パターンセットは、ルールステートメントで一緒に使用する正規表現のコレクションを提供します。正規表現パターンセットは AWS リソースです。

ウェブ ACL またはルールグループで正規表現パターンセットを使用するには、まず正規表現パターン仕様`RegexPatternSet`を使用して AWS リソースを作成します。その後、正規表現パターンセットルールステートメントをウェブ ACL またはルールグループに追加するときに、このセットを参照します。正規表現パターンセットには、少なくとも1つの正規表現パターンが含まれている必要があります。

正規表現パターンセットに複数の正規表現パターンが含まれている場合、ルールで使用される場合、パターン一致は `OR` ロジックと組み合わされます。つまり、リクエストコンポーネントがセット内のいずれかのパターンに一致する場合、ウェブリクエストはパターンセットルールステートメントと一致します。

AWS WAF は、いくつかの例外`libpcre`を除いて、PCRE ライブラリで使用されるパターン構文をサポートします。ライブラリは、「PCRE - Perl Compatible Regular Expressions」で文書化されています。 AWS WAF サポートの詳細については、「」を参照してくださいでの正規表現パターンマッチング AWS WAF。

トピック

- 正規表現パターンセットの作成
- 正規表現パターンセットの削除

正規表現パターンセットの作成

新しい正規表現パターンセットを作成するには、このセクションの手順に従います。

正規表現パターンセットを作成するには

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/wafv2/ で AWS WAF コンソールを開きます。
2. ナビゲーションペインで、[Regex pattern sets] (正規表現パターンセット) を選択し、[Create regex pattern set] (正規表現パターンセットを作成) を選択します。
3. 正規表現パターンセットの名前と説明を入力します。これらを使用して、セットを使用するときに識別します。

   > **Note**
   >
   > 正規表現パターンセットの作成後は、名前を変更できません。

4. リージョン で、グローバル (CloudFront) を選択するか、正規表現パターンセットを保存するリージョンを選択します。正規表現パターンセットは、リージョンのリソースを保護するウェブ ACL でのみ使用できます。Amazon CloudFront ディストリビューションを保護するウェブ ACLs で正規表現パターンセットを使用するには、グローバル () を使用する必要がありますCloudFront。
5. [Regular expressions] (正規表現) テキストボックスに、1 行につき 1 つの正規表現パターンを入力します。

   例えば、正規表現 `I[a@]mAB[a@]dRequest` は、`IamABadRequest`、`IamAB@dRequest`、`I@mABadRequest`、および `I@mAB@dRequest` の文字列に一致します。

   AWS WAF は、いくつかの例外`libpcre`を除いて、PCRE ライブラリで使用されるパターン構文をサポートします。ライブラリは、「PCRE - Perl Compatible Regular Expressions」で文書化されています。 AWS WAF サポートの詳細については、「」を参照してくださいでの正規表現パターンマッチング AWS WAF。
6. 正規表現パターンセットの設定を確認し、[Create regex pattern set] (正規表現パターンセットを作成) を選択します。

正規表現パターンセットの削除

参照セットを削除するには、このセクションのガイダンスに従います。

参照セットとルールグループの削除

IP セット、正規表現パターンセット、ルールグループなど、ウェブ ACL で使用できるエンティティを削除すると、 はエンティティがウェブ ACL で現在使用されている AWS WAF かどうかを確認します。使用中であることがわかった場合、 は AWS WAF ユーザーに警告します。 AWS WAF は、ほとんどの場合、エンティティがウェブ ACL によって参照されているかどうかを判断できます。ただし、まれに判別できないことがあります。エンティティが現在使用中でないことを確認する必要がある場合は、削除する前にウェブ ACL でそのエンティティを確認してください。エンティティが参照されているセットである場合は、ルールグループでエンティティが使用されていないことも確認してください。

正規表現パターンセットを削除するには

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/wafv2/ で AWS WAF コンソールを開きます。
2. ナビゲーションペインで、[Regex pattern sets] (正規表現パターンセット) を選択します。
3. 削除する正規表現パターンセットを選択し、[Delete] (削除) を選択します。

AWS WAFのカスタマイズされたウェブリクエストとレスポンス

AWS WAF ルールアクションとデフォルトウェブ ACL アクションに、カスタムのウェブリクエストとレスポンスの処理動作を追加できます。カスタム設定は、アタッチ先のアクションが適用されるたびに適用されます。

ウェブリクエストとレスポンスは、次の方法でカスタマイズできます。

- Allow、Count、CAPTCHA、Challenge アクションを使用すると、カスタムヘッダーをウェブリクエストに挿入できます。 AWS WAF がウェブリクエストを保護されたリソースに転送する場合、リクエストには、元のリクエスト全体と、挿入したカスタムヘッダーが含まれます。CAPTCHA および Challenge アクションの場合、リクエストが CAPTCHA またはチャレンジトークン検査に合格した場合のみに、 AWS WAF がカスタマイズを適用します。

- Block アクションを使用すると、レスポンスコード、ヘッダー、本文を含めた完全なカスタムレスポンスを定義できます。保護対象リソースは、 AWS WAFが提供するカスタムレスポンスを使用してリクエストに応答します。カスタムレスポンスは、`403 (Forbidden)` のデフォルトの Block アクションレスポンスを置き換えます。

カスタマイズできるアクション設定

次のアクション設定を定義する際に、カスタムリクエストまたはレスポンスを指定できます。

- ルールアクション。詳細については、「ルールアクション」を参照してください。
- ウェブ ACL のデフォルトアクション。詳細については、「ウェブ ACL のデフォルトアクション」を参照してください。

カスタマイズできないアクション設定

ウェブ ACL で使用するルールグループについては、上書きアクションでカスタムリクエスト処理を指定することはできません。「ウェブ ACL ルールおよびルールグループの評価」を参照してください。「マネージドルールグループステートメント」および「ルールグループステートメント」も参照してください。

更新中の一時的な不一致

ウェブ ACL AWS WAF やその他のリソースを作成または変更した場合、その変更がリソースが保存されているすべての領域に反映されるまでに少し時間がかかります。伝播時間は、数秒から数分までかかります。

次の内容では、変更伝播中に直面する一時的な不整合性の例を紹介します。

- ウェブ ACL を作成した後、それをリソースに関連付けようとすると、ウェブ ACL が利用できないことを示す例外が表示される場合があります。
- ルールグループをウェブ ACL に追加した後、新しいルールグループのルールは、ウェブ ACL が使用されるエリアで有効になり、別のエリアでは有効にならない場合があります。
- ルールのアクション設定を変更した後、古いアクションを一部のエリアで確認され、新しいアクションを別のエリアで確認される場合があります。
- ブロックルールで使用されている IP セットに IP アドレスを追加した後、新しいアドレスはあるエリアではブロックされ、別のエリアでは許可される場合があります。

カスタムリクエストとレスポンスの使用制限

AWS WAF カスタムリクエストとカスタムレスポンスの使用に関する最大設定を定義します。ウェブ ACL またはルールグループあたりのリクエストヘッダーの最大数、および単一のカスタムレスポンス定義のカスタムヘッダーの最大数はその一例です。詳細については、「AWS WAF クォータ」を参照してください。

トピック

ノンブロッキングアクション用にカスタムリクエストヘッダーの挿入

ルールアクションによってリクエストがブロックされない場合は、元の HTTP AWS WAF リクエストにカスタムヘッダーを挿入するように指示できます。このオプションでは、リクエストにのみ追加します。元のリクエストの一部を変更したり、置き換えたりすることはできません。カスタムヘッダー挿入のユースケースには、挿入されたヘッダーに基づいてリクエストを異なる方法で処理するようにダウンストリームアプリケーションに通知し、分析のためにリクエストのフラグを立てることが含まれます。

このオプションは、ルールアクション Allow、Count、CAPTCHA、Challenge に適用され、Allow に設定されているウェブ ACL のデフォルトアクションにも適用されます。ルールアクションの詳細については、「ルールアクション」を参照してください。デフォルトのウェブ ACL アクションの詳細については、「ウェブ ACL のデフォルトアクション」を参照してください。

カスタムリクエストヘッダー名

AWS WAF リクエストに既に含まれているヘッダーと混同されないようにx-amzn-waf-、挿入されるすべてのリクエストヘッダーにプレフィックスを付けます。たとえば、sampleヘッダー名を指定すると、 AWS WAF ヘッダーが挿入されます。x-amzn-waf-sample

同じ名前のヘッダー

リクエストに、挿入されるのと同じ名前のヘッダーが既にある場合は、 AWS WAF AWS WAF そのヘッダーが上書きされます。したがって、同じ名前の複数のルールでヘッダーを定義すると、リクエストを検査して一致を見つける最後のルールにはヘッダーが追加され、それよりも前のルールには追加されません。

終了しないルールアクションを含むカスタムヘッダー

アクションとは異なり、AllowCountアクションはウェブ ACL AWS WAF の残りのルールを使用してウェブリクエストの処理を停止しません。同様に、Challengeリクエストトークンが有効であると判断されても、 AWS WAF これらのアクションはウェブリクエストの処理を停止しません。CAPTCHAしたがって、これらのアクションを使用するルールでカスタムヘッダーを挿入する場合、後続のルールもカスタムヘッダーを挿入することがあります。ルールアクションの動作については、「ルールアクション」を参照してください。

例えば、表示された順序で優先順位付けされた次のルールがあるとします。

1. Count アクションと RuleAHeader という名前のカスタマイズされたヘッダーを持つ RuleA。
2. Allow アクションと RuleBHeader という名前のカスタマイズされたヘッダーを持つ RuleB。

リクエストが RuleA と RuleB の両方に一致する場合、 AWS WAF x-amzn-waf-RuleAHeaderヘッダーとを挿入しx-amzn-waf-RuleBHeader、そのリクエストを保護対象リソースに転送します。

AWS WAF Web リクエストの検査が終了すると、カスタムヘッダーをウェブリクエストに挿入します。したがって、アクションが Count に設定されているルールでカスタムリクエスト処理を使用する場合、追加するカスタムヘッダーは後続のルールによって検査されません。

カスタムリクエスト処理の例

ルールのアクションまたはウェブ ACL のデフォルトアクション用に、カスタムリクエスト処理を定義します。次のリストは、ウェブ ACL のデフォルトアクションに追加されたカスタム処理用の JSON を示しています。

```
{
 "Name": "SampleWebACL",
 "Scope": "REGIONAL",
 "DefaultAction": {
  "Allow": {
   "CustomRequestHandling": {
    "InsertHeaders": [
     {
      "Name": "fruit",
      "Value": "watermelon"
     },
     {
      "Name": "pie",
      "Value": "apple"
     }
    ]
   }
  }
 },
 "Description": "Sample web ACL with custom request handling configured for default
 action.",
 "Rules": [],
```

```
  "VisibilityConfig": {
   "SampledRequestsEnabled": true,
   "CloudWatchMetricsEnabled": true,
   "MetricName": "SampleWebACL"
  }
}
```

Block アクションのカスタムレスポンス

に設定されているルールアクションまたはウェブ ACL デフォルトアクションについて、カスタム HTTP AWS WAF Block レスポンスをクライアントに送り返すように指示できます。ルールアクションの詳細については、「ルールアクション」を参照してください。デフォルトのウェブ ACL アクションの詳細については、「ウェブ ACL のデフォルトアクション」を参照してください。

Block アクションのカスタムレスポンス処理を定義すると、ステータスコード、ヘッダー、レスポンス本文を定義します。で使用できるステータスコードのリストについては AWS WAF、カスタムレスポンスでサポートされるステータスコード以下のセクションを参照してください。

ユースケース

カスタムレスポンスのユースケースには、次が含まれます。

- 非デフォルトのステータスコードをクライアントに送り返します。
- カスタムレスポンスヘッダーをクライアントに送り返します。content-type を除き、任意のヘッダー名を指定できます。
- 静的エラーページをクライアントに送り返します。
- クライアントを別の URL にリダイレクトします。これを行うには、301（Moved Permanently）または 302（Found）などの 3xx リダイレクトステータスコードのいずれかを指定してから、新しい URL で Location という名前が付けられた新しいヘッダーを指定します。

保護されたリソースで定義したレスポンスとのインタラクション

AWS WAF Blockアクションに指定するカスタムレスポンスは、保護対象リソースで定義したレスポンス仕様よりも優先されます。

AWS 保護対象のリソースのホストサービスでは、 AWS WAF ウェブリクエストのカスタムレスポンス処理が許可されている場合があります。次に例を示します。

- Amazon では CloudFront、ステータスコードに基づいてエラーページをカスタマイズできます。詳細については、Amazon CloudFront 開発者ガイドの「カスタムエラーレスポンスの生成」を参照してください。
- Amazon API Gateway では、ゲートウェイのレスポンスおよびステータスコードを定義できます。詳細については、「Amazon API Gateway デベロッパーガイド」の「API Gateway でのゲートウェイレスポンス」を参照してください。

AWS WAF AWS 保護対象リソースのカスタムレスポンス設定とカスタムレスポンス設定を組み合わせることはできません。個々のウェブリクエストの応答の仕様は、 AWS WAF または保護されたリソースから、そのすべてが取得されます。

AWS WAF ブロックするウェブリクエストの優先順位は次のとおりです。

1. AWS WAF カスタムレスポンス — AWS WAF Block アクションでカスタムレスポンスが有効になっている場合、保護対象リソースは設定したカスタムレスポンスをクライアントに送り返します。保護されたリソース自体で定義する応答設定は、効果がありません。
2. 保護されたリソースで定義されているカスタムレスポンス - それ以外の場合、保護されたリソースにカスタムレスポンス設定が指定されているときは、保護されたリソースはそれらの設定を使用してクライアントに応答します。
3. AWS WAF Blockデフォルトレスポンス — それ以外の場合、 AWS WAF Block`403 (Forbidden)`保護対象リソースはクライアントにデフォルトレスポンスで応答します。

Web AWS WAF リクエストが許可されている場合、保護対象リソースの設定によって、クライアントに送り返されるレスポンスが決まります。 AWS WAF 許可されたリクエストのレスポンス設定はできません。 AWS WAF 許可されたリクエストについて設定できる唯一のカスタマイズは、保護されたリソースにリクエストを転送する前に、元のリクエストにカスタムヘッダーを挿入することです。このオプションについては、前のセクション「ノンブロッキングアクション用にカスタムリクエストヘッダーの挿入」で説明しました。

カスタムレスポンスヘッダー

`content-type` を除き、任意のヘッダー名を指定できます。

カスタムレスポンス本文

カスタムレスポンスの本文は、それを使用するウェブ ACL またはルールグループのコンテキスト内で定義します。カスタムレスポンスボディの定義後、それを作成したウェブ ACL またはルールグ

ループの他の場所を参照して使用できます。個々の Block アクション設定では、使用するカスタム本文を参照し、カスタムレスポンスのステータスコードおよびヘッダーを定義します。

コンソールでカスタムレスポンスを作成するときは、既に定義したレスポンス本文から選択するか、新しい本文を作成できます。コンソールの外部では、ウェブ ACL またはルールグループレベルでカスタムレスポンス本文を定義し、ウェブ ACL またはルールグループ内のアクション設定から参照します。これは、次のセクションの JSON の例で示されます。

カスタムレスポンスの例

次の例は、カスタムレスポンス設定を持つルールグループの JSON をリストします。カスタムレスポンス本文は、ルールグループ全体のために定義され、ルールアクションでキーによって参照されます。

```
{
 "ARN": "test_rulegroup_arn",
 "Capacity": 1,

 "CustomResponseBodies": {
  "CustomResponseBodyKey1": {
   "Content": "This is a plain text response body.",
   "ContentType": "TEXT_PLAIN"
  }
 },

 "Description": "This is a test rule group.",
 "Id": "test_rulegroup_id",
 "Name": "TestRuleGroup",

 "Rules": [
  {
   "Action": {
    "Block": {
     "CustomResponse": {
      "CustomResponseBodyKey": "CustomResponseBodyKey1",
      "ResponseCode": 404,
      "ResponseHeaders": [
       {
        "Name": "BlockActionHeader1Name",
        "Value": "BlockActionHeader1Value"
       }
      ]
     }
```

```
    }
   },
   "Name": "GeoMatchRule",
   "Priority": 1,
   "Statement": {
    "GeoMatchStatement": {
     "CountryCodes": [
      "US"
     ]
    }
   },
   "VisibilityConfig": {
    "CloudWatchMetricsEnabled": true,
    "MetricName": "TestRuleGroupReferenceMetric",
    "SampledRequestsEnabled": true
   }
  }
 ],
 "VisibilityConfig": {
  "CloudWatchMetricsEnabled": true,
  "MetricName": "TestRuleGroupMetric",
  "SampledRequestsEnabled": true
 }
}
```

カスタムレスポンスでサポートされるステータスコード

HTTP ステータスコードの詳細については、Internet Engineering Task Force (IETF) による「Status Codes」(ステータスコード) およびウィキペディアの「List of HTTP status codes」(HTTP ステータスコードのリスト) を参照してください。

カスタムレスポンスで が AWS WAF サポートする HTTP ステータスコードは次のとおりです。

- 2xx Successful
  - 200 – OK
  - 201 – Created
  - 202 – Accepted
  - 204 – No Content
  - 206 – Partial Content
- 3xx Redirection

- 300 – Multiple Choices
- 301 – Moved Permanently
- 302 – Found
- 303 –See Other
- 304 – Not Modified
- 307 – Temporary Redirect
- 308 – Permanent Redirect

- 4xx Client Error
  - 400 – Bad Request
  - 401 – Unauthorized
  - 403 – Forbidden
  - 404 – Not Found
  - 405 – Method Not Allowed
  - 408 – Request Timeout
  - 409 – Conflict
  - 411 – Length Required
  - 412 – Precondition Failed
  - 413 – Request Entity Too Large
  - 414 – Request-URI Too Long
  - 415 – Unsupported Media Type
  - 416 – Requested Range Not Satisfiable
  - 421 – Misdirected Request
  - 429 – Too Many Requests
- 5xx Server Error
  - 500 – Internal Server Error
  - 501 – Not Implemented
  - 502 – Bad Gateway
  - 503 – Service Unavailable
  - 504 – Gateway Timeout
  - 505 – HTTP Version Not Supported

AWS WAF ウェブリクエストの ラベル

ラベルは、ルールがリクエストに一致するときにルールによってウェブリクエストに追加されるメタデータです。追加すると、ウェブ ACL 評価が終了するまで、リクエストでラベルを使用できます。ウェブ ACL の評価で後から実行されるルール内のラベルには、ラベル照合ステートメントを使用してアクセスできます。詳細については、「ラベル一致ルールステートメント」を参照してください。

ウェブリクエストのラベルは、Amazon CloudWatch ラベルメトリクスを生成します。メトリクスとディメンションのリストについては、「ラベルメトリクスとディメンション」を参照してください。CloudWatch および コンソールからメトリクスとメトリクスの概要にアクセスする方法については、AWS WAF 「」を参照してくださいモニタリングとチューニング。

ラベリングのユースケース

AWS WAF ラベルの一般的なユースケースは次のとおりです。

- リクエストに対してアクションを実行する前に、複数のルールステートメントに対してウェブリクエストを評価する – ウェブ ACL 内のルールとの一致が見つかった後、ルールアクションがウェブ ACL 評価を終了しない場合、 はウェブ ACL に対するリクエストの評価 AWS WAF を続行します。リクエストを許可または拒否するか判断する前、ラベルを使用して複数のルールから情報を評価および収集できます。これを行うには、既存のルールのアクションを Count に変更し、ラベルを一致リクエストに追加するように設定します。その後、他のルールの後に実行する新しいルールを 1 つ以上追加し、ラベルを評価してラベル一致の組み合わせに応じてリクエストを管理するように設定します。
- 地域別のウェブリクエストの管理 – 地理的一致ルールを単独で使用して、ウェブリクエストを発信国別に管理できます。地域レベルの精度で場所を微調整するには、地理一致ルールを Count アクションと一緒に使用し、それに続いてラベルマッチルールを使用します。地理一致ルールの情報については、「地理的一致ルールステートメント」を参照してください。
- 複数のルール間でロジックを再利用する – 複数のルールで同じロジックを再利用する必要がある場合は、ラベルを使用してそのロジックを単一のソースにして、結果をテストします。ネストされたルールステートメントの共通のサブセットを使用する複雑なルールが複数ある場合、複雑なルール間で共通ルールセットを複製すると、時間がかかり、エラーが発生しやすくなります。ラベルを使用すると、一致するリクエストをカウントし、それらにラベルを追加する共通ルールサブセットを使用して新しいルールを作成できます。新しいルールをウェブ ACL に追加して、元の複雑なルールの前に実行されるようにします。その後、元のルールで、共有ルールサブセットを、ラベルをチェックする単一のルールに置き換えます。

例えば、ログインパスにのみ適用する複数のルールがあるとします。各ルールで潜在的なログインパスと一致する同じロジックを指定するのではなく、そのロジックを含む 1 つの新しいルールを実装できます。新しいルールで、一致するリクエストにラベルを追加して、リクエストがログインパス上にあることを示します。ウェブ ACL で、この新しいルールの優先順位の数値設定を、元のルールの数値よりも小さく設定して、最初に実行されるようにします。その後、元のルールで、共有ロジックをラベルの存在のチェックに置き換えます。ジョブの優先順位の設定については、「ウェブ ACL でのルールおよびルールグループの処理順序」を参照してください。

- ルールグループ内のルールに対する例外を作成する – このオプションは、表示または変更できないマネージドルールグループに特に役立ちます。多くのマネージドルールグループのルールはウェブリクエストにラベルを追加して一致したルールを示し、場合によってはその一致に関する追加情報を提供します。リクエストにラベルを追加するルールグループを使用すると、ルールグループのルールが一致をカウントするようにオーバーライドし、その後にルールグループラベルに基づいたウェブリクエストを処理するルールグループの後にルールを実行できます。すべての AWS マネージドルールは、一致するウェブリクエストにラベルを追加します。詳細については、「AWS マネージドルールルールグループリスト」のルールの説明を参照してください。
- ラベルメトリクスの使用によるトラフィックパターンの監視 — ルールを使用して追加したラベルのメトリクスや、ウェブ ACL で使用するマネージドルールグループによって追加されたメトリクスのメトリクスにアクセスできます。 AWS マネージドルールのルールグループのすべては、評価するウェブリクエストにラベルを追加します。ラベルメトリクスとディメンションのリストについては、「ラベルメトリクスとディメンション」を参照してください。メトリクスとメトリクスの概要には、 CloudWatch および AWS WAF コンソールのウェブ ACL ページからアクセスできます。詳細については、「モニタリングとチューニング」を参照してください。

AWS WAF ラベル付けの仕組み

ルールがウェブリクエストに一致すると、ルールにラベルが定義されている場合、 はルール評価の最後にラベルをリクエスト AWS WAF に追加します。ウェブ ACL 内のルール一致後に評価されるルールは、ルールが追加したラベルと照合できます。

何によってリクエストにラベルが追加されるのか

リクエストを評価するウェブ ACL コンポーネントは、リクエストにラベルを追加できます。

- ルールグループ参照ステートメントではないルールは、一致するウェブリクエストにラベルを追加できます。ラベル付け基準はルール定義の一部であり、ウェブリクエストがルールと一致すると、

はルールのラベルをリクエスト AWS WAF に追加します。詳細については、「the section called “ラベルを追加するルール”」を参照してください。

- 地理照合ステートメントは、ステートメントの結果が一致するかどうかに関係なく、検査するすべてのリクエストに国と地域のラベルを追加します。詳細については、「the section called “地理的一致”」を参照してください。
- AWS WAF すべての の AWS マネージドルールは、検査するリクエストにラベルを追加します。ルールグループ内のルールの一致に基づいてラベルを追加します。また、インテリジェントな脅威軽減ルールグループを使用すると追加されるトークンラベルなど、マネージドルールグループが使用する AWS プロセスに基づいてラベルを追加します。各マネージドルールグループが追加するラベルの詳細については、「the section called “AWS マネージドルールルールグループリスト”」を参照してください。

がラベル AWS WAF を管理する方法

AWS WAF は、ルールによるリクエストの検査の最後に、ルールのラベルをリクエストに追加します。ラベル付けは、アクションと同様にルールの照合アクティビティの一部です。

ウェブ ACL 評価が終了した後、ラベルはウェブリクエストに保持されません。ルールが追加するラベルに照らして他のルールが照合するためには、ルールアクションがウェブ ACL によるウェブリクエストの評価を終了してはなりません。ルールアクションは Count、CAPTCHA、Challenge に設定する必要があります。ウェブ ACL の評価が終了しない場合、ウェブ ACL 内の後続のルールは、リクエストに対してラベル一致基準を実行できます。ルールアクションの詳細については、「ルールアクション」を参照してください。

ウェブ ACL 評価中のラベルへのアクセス

追加されると、 がウェブ ACL に対してリクエストを評価している限り、ラベル AWS WAF はリクエストで引き続き使用できます。ウェブ ACL 内のすべてのルールは、同じウェブ ACL ですでに実行されているルールによって追加されたラベルにアクセスできます。これには、ウェブ ACL 内で直接定義されたルールと、ウェブ ACL で使用されるルールグループ内の手以後されたルールが含まれます。

- ラベル一致ステートメントを使用してルールのリクエスト検査基準のラベルと照合できます。リクエストに添付されているどのラベルとも照合できます。ステートメントの詳細については、「ラベル一致ルールステートメント」を参照してください。
- 地理的照合ステートメントは、一致の有無にかかわらずラベルを追加しますが、ステートメントに含まれるウェブ ACL ルールがリクエストの評価を完了して初めて使用できるようになります。

- 論理 AND ステートメントなどの単一のルールを使用して、地理的ラベルに対して地域照合ステートメントの後にラベル照合チステートメントを実行することはできません。ラベル照合ステートメントは、地理的照合ステートメントを含むルールの後に実行される別のルールに記述する必要があります。
- 地理的照合ステートメントをレートベースのルールステートメント、またはマネージドルールグループ参照ステートメント内のスコープダウンステートメントとして使用する場合、地理的照合ステートメントによって追加されたラベルは、包含ルールステートメントでは検査用に使用できません。レートベースのルールステートメントまたはルールグループの地理的ラベルを調べる必要がある場合は、事前に実行される別のルールで地理的照合ステートメントを実行する必要があります。

ウェブ ACL 評価以外のラベル情報へのアクセス

ウェブ ACL 評価が終了した後、ラベルはウェブリクエストに保持されませんが、 AWS WAF はラベル情報をログとメトリクスに記録します。

- AWS WAF は、1 回のリクエストで最初の 100 個のラベルの Amazon CloudWatch メトリクスを保存します。ラベルメトリクスへのアクセスの詳細については、「Amazon によるモニタリング CloudWatch」および「ラベルメトリクスとディメンション」を参照してください。
- AWS WAF は、 AWS WAF コンソールのウェブ ACL トラフィック概要ダッシュボードの CloudWatch ラベルメトリクスを要約します。ダッシュボードにはどのウェブ ACL ページからでもアクセスできます。詳細については、「ウェブ ACL トラフィック概要ダッシュボード」を参照してください。
- AWS WAF は、リクエストの最初の 100 個のラベルのラベルをログに記録します。ルールアクションとともにラベルを使用して、 AWS WAF が記録するログをフィルタリングできます。詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。

ウェブ ACL 評価では、ウェブリクエストに 100 個を超えるラベルを適用して 100 個を超えるラベルと照合できますが、ログとメトリクスには最初の 100 個 AWS WAF のみが記録されます。

AWS WAF ラベル構文と命名要件

ラベルは、プレフィックス、オプションの名前空間、および名前で構成される文字列です。ラベルのコンポーネントはコロンで区切られます。ラベルには次の要件と特性があります。

- ラベルでは、大文字と小文字が区別されます。
- 各ラベル名前空間またはラベル名には、最大 128 文字を使用できます。

- ラベルには、最大 5 つの名前空間を指定できます。
- ラベルのコンポーネントはコロン (:) で区切られます。
- ラベルに指定する名前空間または名前で次の予約済み文字列を使用することはできません: `awswaf`、`aws`、`waf`、`rulegroup`、`webacl`、`regexpatternset`、`ipset`、および `managed`。

ラベル構文

完全修飾ラベルには、プレフィックス、オプションの名前空間、およびラベル名があります。プレフィクスは、ラベルを追加したルールのルールグループまたはウェブ ACL コンテキストを識別します。名前空間は、ラベルのコンテキストを追加するために使用されることがあります。ラベル名は、ラベルの詳細レベルが最も低いレベルになります。多くの場合、リクエストにラベルを追加した特定のルールを示します。

ラベルのプレフィックスは、そのオリジンによって異なります。

- ラベル – ウェブ ACL およびルールグループのルールで作成するラベルの完全なラベル構文を次に示します。エンティティタイプは `rulegroup` と `webacl` です。

```
awswaf:<entity owner account id>:<entity type>:<entity name>:<custom
 namespace>:...:<label name>
```

  - ラベル名前空間プレフィックス: `awswaf:<entity owner account id>:<entity type>:<entity name>:`
  - カスタム名前空間の追加: `<custom namespace>:…:`

  ルールグループまたはウェブ ACL でルールのラベルを定義する場合は、カスタム名前空間文字列とラベル名をコントロールします。残りは によって生成されます AWS WAF。 は、すべてのラベルに `awswaf`と、アカウントおよびウェブ ACL またはルールグループのエンティティ設定 AWS WAF を自動的にプレフィックスします。

- マネージドルールグループのラベル – マネージドルールグループのルールによって作成されるラベルの完全なラベル構文を次に示します。

```
awswaf:managed:<vendor>:<rule group name>:<custom namespace>:...:<label name>
```

  - ラベル名前空間プレフィックス: `awswaf:managed:<vendor>:<rule group name>:`
  - カスタム名前空間の追加: `<custom namespace>:…:`

すべての AWS マネージドルールのルールグループがラベルを追加します。マネージドルールグループの詳細については、「マネージドルールグループ」を参照してください。

- 他の AWS プロセスのラベル – これらのプロセスは AWS マネージドルールのルールグループによって使用されるため、マネージドルールグループを使用して評価するウェブリクエストに追加されます。マネージドルールグループによって呼び出されるプロセスが作成するラベルの完全なラベル構文を次に示します。

  ```
  awswaf:managed:<process>:<custom namespace>:...:<label name>
  ```

  - ラベル名前空間プレフィックス: `awswaf:managed:<process>:`
  - カスタム名前空間の追加: `<custom namespace>:…:`

  このタイプのラベルは、 AWS プロセスを呼び出すマネージドルールグループ用に一覧表示されます。マネージドルールグループの詳細については、「マネージドルールグループ」を参照してください。

ルールの例にラベルを付ける

次のラベルの例は、アカウント 111122223333 に属する `testRules` という名前のルールグループのルールによって定義されています。

```
awswaf:111122223333:rulegroup:testRules:testNS1:testNS2:LabelNameA
```

```
awswaf:111122223333:rulegroup:testRules:testNS1:LabelNameQ
```

```
awswaf:111122223333:rulegroup:testRules:LabelNameZ
```

次のリストは、JSON のラベル指定の例を示しています。これらのラベル名には、末尾のラベル名の前にカスタム名前空間文字列が含まれます。

```
Rule: {
    Name: "label_rule",
    Statement: {...}
    RuleLabels: [
        Name: "header:encoding:utf8",
        Name: "header:user_agent:firefox"
```

```
    ],
    Action: { Count: {} }
}
```

> **Note**
>
> このタイプのリストには、ルール JSON エディタを通じてコンソールでアクセスできます。

前述のラベルの例と同じルールグループおよびアカウントで前述のルールを実行すると、結果として生じる完全修飾ラベルは次のようになります。

```
awswaf:111122223333:rulegroup:testRules:header:encoding:utf8
```

```
awswaf:111122223333:rulegroup:testRules:header:user_agent:firefox
```

マネージドルールグループのラベル例

以下に、 AWS マネージドルールのルールグループとプロセスから呼び出すラベルの例を示します。

```
awswaf:managed:aws:core-rule-set:NoUserAgent_Header
```

```
awswaf:managed:aws:sql-database:SQLiExtendedPatterns_QueryArguments
```

```
awswaf:managed:aws:atp:aggregate:attribute:compromised_credentials
```

```
awswaf:managed:token:accepted
```

AWS WAF ラベルを追加する ルール

ほとんどのルールでは、ラベルを定義して AWS WAF 、一致するリクエストに適用できます。

次のルールタイプが唯一の例外です。

- レートベースのルールラベルはレート制限中のみ – レートベースのルールは、特定の集約インスタンスのウェブリクエストにラベルを追加するだけですが、そのインスタンスは によってレート

制限されています AWS WAF。レートベースルールの詳細については、「レートベースのルールステートメント」を参照してください。

- ルールグループ参照ステートメントではラベル付けは許可されません – コンソールでは、これらのルールタイプのラベルは使用できません。API を使用して、いずれかのステートメントタイプのラベルを指定すると、検証例外が発生します。これらのステートメントのタイプについては、「マネージドルールグループステートメント」および「ルールグループステートメント」を参照してください。

WCU – ウェブ ACL またはルールグループのルールで定義する 5 つのラベルごとに 1 つの WCU。

ステートメントの場所

- コンソールのルールビルダー – ルールの [Action] (アクション) 設定の [Label] (ラベル) の下。
- API データタイプ – `Rule RuleLabels`

ルールでラベルを定義するには、ラベル名前空間 prefix に追加するカスタム名前空間文字列と名前を指定します。 は、ルールを定義するコンテキストからプレフィックス AWS WAF を取得します。これについては、「AWS WAF ラベル構文と命名要件」の下のラベル構文情報を参照してください。

AWS WAF ラベルに一致する ルール

ラベル一致ステートメントを使用して、ウェブリクエストラベルを評価できます。ラベル名が必要な [Label] (ラベル)、または名前空間の指定が必要な [Namespace] (名前空間) と照合できます。ラベルまたは名前空間のいずれの場合も、オプションで、前述の名前空間とプレフィックスを指定に含めることができます。このステートメントタイプの一般的な情報については、「ラベル一致ルールステートメント」を参照してください。

ラベルのプレフィクスは、ラベルのルールが定義されているルールグループまたはウェブ ACL のコンテキストを定義します。ルールのラベル一致ステートメントで、ラベルまたは名前空間一致文字列でプレフィックスが指定されていない場合、 はラベル一致ルールのプレフィックス AWS WAF を使用します。

- ウェブ ACL 内で直接定義されたルールのラベルには、ウェブ ACL コンテキストを指定するプレフィックスが付いています。
- ルールグループ内にあるルールのラベルには、ルールグループコンテキストを指定するプレフィクスがあります。これは、独自のルールグループでも、マネージドルールグループでもかまいません。

これについては、「AWS WAF ラベル構文と命名要件」のラベル構文を参照してください。

> **Note**
>
> 一部のマネージドルールグループは、ラベルを追加します。`DescribeManagedRuleGroup` を呼び出すことにより、API を介してこれらを取得できます。ラベルは、応答の `AvailableLabels` プロパティにリストされています。

ルールのコンテキストとは異なるコンテキストにあるルールと照合する場合は、一致文字列にプレフィックスを指定する必要があります。例えば、マネージドルールグループのルールによって追加されたラベルと照合する場合は、一致文字列でルールグループのプレフィクスとその後に追加の一致基準が指定されるラベル一致ステートメントを使用して、ウェブ ACL にルールを追加できます。

ラベル一致ステートメントの一致文字列で、ラベルまたは名前空間のいずれかを指定します。

- ラベル – 一致のラベルの指定は、ラベルの終了部分で構成されます。ラベル名の直前に、連続するネームスペースをいくつでも含めて、その後に名前を含めることができます。指定の先頭をプレフィックスにして、完全修飾ラベルを指定することもできます。

  指定の例:

  - `testNS1:testNS2:LabelNameA`
  - `awswaf:managed:aws:managed-rule-set:testNS1:testNS2:LabelNameA`
- 名前空間 – 一致の名前空間の指定は、名前を除くラベルの指定の連続するサブセットで構成されます。プレフィックスを含めることができ、1 つ以上の名前空間文字列を含めることができます。

  指定の例:

  - `testNS1:testNS2:`
  - `awswaf:managed:aws:managed-rule-set:testNS1:`

AWS WAF ラベル一致の例

このセクションは、ラベル一致ルールステートメントの一致の指定の例を示します。

> **Note**
>
> これらの JSON リストは、ラベル一致指定を使用してウェブ ACL にルールを追加し、ルールを編集して Rule JSON エディタに切り替えることでコンソールで作成されました。API ま

たはコマンドラインインターフェイスを通じて、ルールグループまたはウェブ ACL の JSON を取得することもできます。

トピック

- ローカルラベルと照合する
- 別のコンテキストのラベルと照合する
- マネージドルールグループラベルと照合する
- ローカル名前空間と照合する
- マネージドルールグループ名前空間と照合する

ローカルラベルと照合する

次の JSON リストは、このルールと同じコンテキストで、ウェブリクエストにローカルに追加されたラベルのラベル一致ステートメントを示しています。

```
Rule: {
    Name: "match_rule",
    Statement: {
        LabelMatchStatement: {
            Scope: "LABEL",
            Key: "header:encoding:utf8"
        }
    },
    RuleLabels: [
        ...generate_more_labels...
    ],
    Action: { Block: {} }
}
```

アカウント 111122223333 でこの一致ステートメントを使用する場合、ウェブ ACL `testWebACL` のために定義するルールで、次のラベルと照合します。

```
awswaf:111122223333:webacl:testWebACL:header:encoding:utf8
```

```
awswaf:111122223333:webacl:testWebACL:testNS1:testNS2:header:encoding:utf8
```

ラベル文字列が完全に一致しないため、次のラベルには一致しません。

```
awswaf:111122223333:webacl:testWebACL:header:encoding2:utf8
```

コンテキストが同じではないため、次のラベルには一致せず、したがってプレフィックスは一致しません。これは、ルールが定義されているウェブ ACL testWebACL にルールグループ productionRules を追加した場合であっても当てはまります。

```
awswaf:111122223333:rulegroup:productionRules:header:encoding:utf8
```

別のコンテキストのラベルと照合する

次の JSON リストは、ユーザーが作成したルールグループ内のルールのラベルと照合するラベル一致ルールを示しています。名前が挙げられるルールグループに属さないウェブ ACL で実行されているすべてのルールの指定では、プレフィックスが必要です。このラベル指定の例では、正確なラベルのみと一致します。

```
Rule: {
    Name: "match_rule",
    Statement: {
        LabelMatchStatement: {
            Scope: "LABEL",
            Key: "awswaf:111122223333:rulegroup:testRules:header:encoding:utf8"
        }
    },
    RuleLabels: [
        ...generate_more_labels...
    ],
    Action: { Block: {} }
}
```

マネージドルールグループラベルと照合する

これは、一致ルールのコンテキストとは別のコンテキストからのラベルとの一致の特殊なケースです。次の JSON リストは、マネージドルールグループラベルのラベル一致ステートメントを示しています。これは、ラベル一致ステートメントのキー設定で指定された正確なラベルのみと一致します。

```
Rule: {
    Name: "match_rule",
    Statement: {
        LabelMatchStatement: {
```

```
            Scope: "LABEL",
            Key: "awswaf:managed:aws:managed-rule-set:header:encoding:utf8"
        }
    },
    RuleLabels: [
        ...generate_more_labels...
    ],
    Action: { Block: {} }
}
```

ローカル名前空間と照合する

次の JSON リストは、ローカル名前空間のラベル一致ステートメントを示しています。

```
Rule: {
    Name: "match_rule",
    Statement: {
        LabelMatchStatement: {
            Scope: "NAMESPACE",
            Key: "header:encoding:"
        }
    },
    Labels: [
        ...generate_more_labels...
    ],
    Action: { Block: {} }
}
```

ローカル Label 一致と同様に、アカウント 111122223333 でこのステートメントを使用する場合、ウェブ ACL testWebACL のために定義するルールで、次のラベルと照合します。

```
awswaf:111122223333:webacl:testWebACL:header:encoding:utf8
```

アカウントが同じではないため、次のラベルには一致せず、したがってプレフィックスは一致しません。

```
awswaf:444455556666:webacl:testWebACL:header:encoding:utf8
```

プレフィックスは、次のようなマネージドルールグループによって適用されるラベルにも一致しません。

```
awswaf:managed:aws:managed-rule-set:header:encoding:utf8
```

マネージドルールグループ名前空間と照合する

次の JSON リストは、マネージドルールグループ名前空間のラベル一致ステートメントを示しています。所有しているルールグループの場合、ルールのコンテキスト外にある名前空間と照合するために、プレフィックスを指定する必要もあります。

```
Rule: {
    Name: "match_rule",
    Statement: {
        LabelMatchStatement: {
            Scope: "NAMESPACE",
            Key: "awswaf:managed:aws:managed-rule-set:header:"
        }
    },
    RuleLabels: [
        ...generate_more_labels...
    ],
    Action: { Block: {} }
}
```

この指定は、次のサンプルラベルと照合します。

```
awswaf:managed:aws:managed-rule-set:header:encoding:utf8
```

```
awswaf:managed:aws:managed-rule-set:header:encoding:unicode
```

次のラベルとは一致しません。

```
awswaf:managed:aws:managed-rule-set:query:badstring
```

AWS WAF インテリジェントな脅威軽減

このセクションでは、が提供するマネージド型インテリジェント脅威軽減機能について説明します。AWS WAFこれらは、悪意のあるボットやアカウント乗っ取りの試みなどの脅威から保護するために実装できる、高度で特殊な保護機能です。

> **Note**
>
> ここで説明する機能には、基本使用料以外に追加料金がかかります。 AWS WAF詳細については、「AWS WAF の料金」を参照してください。

このセクションのガイダンスは、 AWS WAF ウェブ ACL、ルール、およびルールグループを作成および管理する方法を一般的に理解しているユーザーを対象としています。これらのトピックは、このガイドの前のセクションでカバーされています。

トピック

- インテリジェントな脅威の軽減のためのオプション
- インテリジェントな脅威の軽減のためのベストプラクティス
- AWS WAF ウェブリクエストトークン
- AWS WAF 不正防止アカウント作成詐欺防止 (ACFP)
- AWS WAF 不正防止アカウント乗っ取り防止 (ATP)
- AWS WAF ボットコントロール
- AWS WAF クライアントアプリケーション統合
- CAPTCHAChallengeの および AWS WAF

インテリジェントな脅威の軽減のためのオプション

このセクションでは、インテリジェントな脅威の軽減を実装するためのオプションを詳細に比較します。

AWS WAF には、インテリジェントな脅威軽減のための次の種類の保護機能があります。

- AWS WAF Fraud Control アカウント作成詐欺防止 (ACFP) — アプリケーションのサインアップページでの悪意のあるアカウント作成の試みを検出して管理します。コア機能は、ACFP マネージドルールグループによって提供されます。詳細については、AWS WAF 不正防止アカウント作成詐欺防止 (ACFP)およびAWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) ルールグループを参照してください。
- AWS WAF 不正防止アカウント乗っ取り防止 (ATP) — アプリケーションのログインページでの悪意のある乗っ取りの試みを検出して管理します。コア機能は、ATP マネージドルールグループによって提供されます。詳細については、AWS WAF 不正防止アカウント乗っ取り防止 (ATP)およびAWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループを参照してください。

- AWS WAF ボットコントロール — 友好的なボットと悪意のあるボットの両方を識別、ラベル付け、管理します。この機能により、アプリケーション間で一意のシグネチャを持つ一般的なボットや、アプリケーション固有のシグネチャを持つターゲットしたボットを管理できます。コア機能は、Bot Control マネージドルールグループによって提供されます。詳細については、AWS WAF ボットコントロールおよびAWS WAF Bot Control ルールグループを参照してください。
- クライアントアプリケーション統合 SDK — Web ページ上のクライアントセッションとエンドユーザーを検証し、 AWS WAF クライアントがウェブリクエストで使用するトークンを取得します。ACFP、ATP、または Bot Control を使用する場合、可能であればクライアントアプリケーションにアプリケーション統合 SDK を実装し、ルールグループのすべての機能を最大限に活用してください。重大なリソースを迅速に保護する必要があり、SDK 統合に十分な時間がないときにのみ、一時的な対策として SDK を統合せずにこれらのルールグループを使用することをお勧めします。SDK を実装する情報については、「AWS WAF クライアントアプリケーション統合」を参照してください。
- ChallengeCAPTCHAおよびルールアクション — クライアントセッションとエンドユーザーを検証し、 AWS WAF クライアントがウェブリクエストで使用するトークンを取得します。これらは、ルールアクションを指定する任意の場所、ルール内、使用するルールグループのオーバーライドとして実装できます。これらのアクションは、 AWS WAF JavaScript インタースティシャルを使用してクライアントまたはエンドユーザーに問い合わせますが、それらをサポートするクライアントアプリケーションが必要です。 JavaScript詳細については、「CAPTCHAChallengeの およびAWS WAF」を参照してください。

AWS インテリジェントな脅威軽減マネージドルールグループ (ACFP、ATP、Bot Control) は、トークンを使用して高度な検出を行います。トークンがルールグループで有効にする機能については、「ACFP でアプリケーション統合 SDK を使用する理由」、「ATP でアプリケーション統合 SDK を使用する理由」、「Bot Control でアプリケーション統合 SDK を使用する理由」を参照してください。

インテリジェントな脅威軽減を実装するための選択肢は、チャレンジを実行してトークン取得を強制するためのルールアクションの基本的な使い方から、インテリジェントな脅威軽減マネージドルールルールグループが提供する高度な機能まで、多岐にわたります。 AWS

次の表では、基本および高度な機能のオプションを詳細に比較します。

トピック

- チャレンジとトークン取得のオプション
- インテリジェントな脅威の軽減マネージドルールグループのオプション

- レートベースのルールとターゲットを絞った Bot Control ルールにおけるレート制限のオプション

チャレンジとトークン取得のオプション

AWS WAF アプリケーション統合 SDKsまたはルールアクション および Challenge を使用して、チャレンジを提供し、トークンを取得できますCAPTCHA。大まかに言うと、ルールアクションの実装は簡単ですが、追加コストが発生し、カスタマーエクスペリエンスにさらに影響し、 が必要です JavaScript。SDKs にはクライアントアプリケーションでのプログラミングが必要ですが、カスタマーエクスペリエンスが向上し、無料で使用でき、Android JavaScript または iOS アプリケーションでも使用できます。アプリケーション統合 SDK は、次のセクションで説明する有料のインテリジェント脅威軽減のマネージドルールグループをいずれか 1 つ使用するウェブ ACL でのみ使用できます。

チャレンジおよびトークン取得のオプション比較

	Challenge ルールアクション	CAPTCHA ルールアクション	JavaScript SDK チャレンジ	モバイル SDK チャレンジ
その内容とは	ブラウザクライアントにサイレントチャレンジインタースティシャルを提示することで AWS WAF トークンの取得を強制するルールアクション	クライアントエンドユーザーにビジュアルまたはオーディオチャレンジインタースティシャルを提示することで AWS WAF トークンの取得を強制するルールアクション	を実行するクライアントブラウザやその他のデバイス用のアプリケーション統合レイヤー JavaScript。サイレントチャレンジをレンダリングし、トークンを取得します。	Android および iOS アプリケーション向けのアプリケーション統合レイヤー。サイレントチャレンジをネイティブにレンダリングし、トークンを取得します
こんなことにお勧めします	ボットセッションに対するサイレント検証と、をサポートするクライアントのトークン取得の適用 JavaScript	をサポートするクライアントのボットセッションに対するエンドユーザーとサイレントの検証とトークン取得	ボットセッションに対するサイレント検証と、をサポートするクライアントのトークン取得の強制 JavaScript。	ボットセッションに対するサイレント検証および Android iOS のネイティブモバイルアプリケーションに

	Challenge ルールアクション	CAPTCHA ルールアクション	JavaScript SDK チャレンジ	モバイル SDK チャレンジ
		の強制 JavaScript	SDK はレイテンシーを最小限に抑え、アプリケーション内で実行されるチャレンジスクリプトの場所を最適に制御します。	トークン取得の強制。 SDK はレイテンシーを最小限に抑え、アプリケーション内で実行されるチャレンジスクリプトの場所を最適に制御します。
実装の考慮事項	ルールアクション設定として実装済	ルールアクション設定として実装済	ウェブ ACL の ACFP、ATP、または Bot Control の有料ルールグループの 1 つが必要です。 クライアントアプリケーションでのコーディングが必要です。	ウェブ ACL の ACFP、ATP、または Bot Control の有料ルールグループの 1 つが必要です。 クライアントアプリケーションでのコーディングが必要です。

	Challenge ルールアクション	CAPTCHA ルールアクション	JavaScript SDK チャレンジ	モバイル SDK チャレンジ
ランタイムの考慮事項	有効なトークンがないリクエストの侵入的なフロー。クライアントは AWS WAF チャレンジインタースティシャルにリダイレクトされます。ネットワークラウンドトリップを追加し、ウェブリクエストの 2 次評価が必要とします。	有効なトークンがないリクエストの侵入的なフロー。クライアントは AWS WAF CAPTCHA インタースティシャルにリダイレクトされます。ネットワークラウンドトリップを追加し、ウェブリクエストの 2 次評価が必要とします。	舞台裏で実行できます。チャレンジ体験をより広範囲に制御できます。	舞台裏で実行できます。チャレンジ体験をより広範囲に制御できます。
必須 JavaScript	はい	はい	はい	なし
サポートされているクライアント	Javascript を実行するブラウザおよびデバイス	Javascript を実行するブラウザおよびデバイス	Javascript を実行するブラウザおよびデバイス	Android および iOS デバイス

	Challenge ルールアクション	CAPTCHA ルールアクション	JavaScript SDK チャレンジ	モバイル SDK チャレンジ
単一ページアプリケーション (SPA) をサポート	強制適用のみ。 Challenge アクションを SDK と組み合わせて使用することで、リクエストが有効なチャレンジトークンを確実に持つようにできます。このルールアクションを使用してチャレンジスクリプトをページに配信することはできません。	強制適用のみ。 CAPTCHA アクションを SDK と組み合わせて使用することで、リクエストが有効な CAPTCHA トークンを確実に持つようにできます。このルールアクションを使用して CAPTCHA スクリプトをページに配信することはできません。	あり	該当なし
追加料金	はい。定義したルールまたは使用するルールグループのルールアクションのオーバーライドとして明示的に指定するアクション設定に適用されます。それ以外の場合は、いいえ。	はい。定義したルールまたは使用するルールグループのルールアクションのオーバーライドとして明示的に指定するアクション設定に適用されます。それ以外の場合は、いいえ。	いいえ。 ただし、ACFP、ATP、または Bot Control のいずれかの有料ルールグループが必要です。	いいえ。 ただし、ACFP、ATP、または Bot Control のいずれかの有料ルールグループが必要です。

これらのオプションに関連するコストの詳細については、「AWS WAF の料金表」のインテリジェントな脅威の軽減情報を参照してください。

Challenge または CAPTCHA アクションを含むルールを追加するだけで、チャレンジを実行して基本的なトークンの強制がより簡単にできます。アプリケーションコードにアクセスできない場合など、ルールアクションの使用が必要になる場合があります。

ただし、SDK を実装できれば、Challenge アクションを使用する場合と比較し、クライアントのウェブリクエストのウェブ ACL 評価におけるコストを節約し、レイテンシーを低減できます。

- アプリケーションの任意の時点でチャレンジを実行するように SDK 実装を記述できます。保護されたリソースにウェブリクエストを送信するカスタマーアクションの前、トークンをバックグラウンドで取得できます。これにより、クライアントの最初のリクエストでトークンを送信できるようになります。
- 代わりに、Challenge アクションを含むルールを実装してトークンを取得する場合、クライアントが最初にリクエストを送信、ならびにトークンの有効期限が切れるとき、ルールおよびアクションに追加のウェブリクエスト評価と処理が必要になります。Challenge アクションは、有効期限が切れていない有効なトークンがないリクエストをブロックし、チャレンジインタースティシャルをクライアントに送り返します。クライアントがチャレンジの応答に成功した後、インタースティシャルは元のウェブリクエストを有効なトークンで再送信し、そのトークンがウェブ ACL によって再度評価されます。

インテリジェントな脅威の軽減マネージドルールグループのオプション

インテリジェントな脅威の軽減 AWS マネージドルールのルールグループは、基本的なボットの管理、高度な悪意のあるボットの検出と軽減、アカウント乗っ取りの試みの検出と軽減、不正なアカウント作成の試みの検出と軽減を提供します。これらのルールグループは、前のセクションで説明したアプリケーション統合 SDK と組み合わせると、最も高度な保護およびクライアントアプリケーションとの安全な連携が可能になります。

マネージドルールのグループオプションの比較

	ACFP	ATP	Bot Control の共通レベル	Bot Control の目標レベル
その内容とは	アプリケーションの登録ページおよびサインアップページでのアカウントの不正な作成の試	アプリケーションのログインページで、悪意のある乗っ取りの試みの一部である可能性があ	アプリケーション間で一意のシグネチャで自己識別を行う一般的なボットを管理します。	アプリケーション固有のシグネチャで自己識別を行わないターゲットしたボッ

	ACFP	ATP	Bot Control の共通レベル	Bot Control の目標レベル
	みの一環である可能性のあるリクエストを管理します。 ボットを管理しません。 AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) ルールグループ を参照してください。	るリクエストを管理します。 ボットを管理しません。 AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループ を参照してください。	AWS WAF Bot Control ルールグループ を参照してください。	トを管理します。 AWS WAF Bot Control ルールグループ を参照してください。

	ACFP	ATP	Bot Control の共通レベル	Bot Control の目標レベル
こんなことにお勧めします	ユーザー名トラバーサルによる作成の試みや、単一の IP アドレスから作成された多数の新しいアカウントなど、不正なアカウント作成攻撃がないかを確認するための、アカウント作成トラフィックの検査。	パスワードトラバーサルによるログイン試行や同じ IP アドレスから多数のログイン試行など、アカウント乗っ取り攻撃におけるログイントラフィックの検査。トークンと併用すると、大量の失敗したログイン試行の IP およびクライアントセッションにおけるレート制限などの総合的な保護も提供されます。	一般的な自動ボットトラフィックの基本的なボット保護およびラベル付け。	クライアントセッションレベルでのレート制限、ならびに Selenium や Puppeteer などのブラウザ自動化ツールの検出と軽減を含め、高度なボットに対するターゲットを絞った保護。
評価結果を示すラベルを追加します	はい	はい	はい	あり
トークンラベルを追加します	はい	はい	はい	あり

	ACFP	ATP	Bot Control の共通レベル	Bot Control の目標レベル
有効なトークンがないリクエストのブロック	含まれません。 有効な AWS WAF トークンがないリクエストのブロック を参照してください。	含まれません。 有効な AWS WAF トークンがないリクエストのブロック を参照してください。	含まれません。 有効な AWS WAF トークンがないリクエストのブロック を参照してください。	トークンなしで 5 回のリクエストを送信するクライアントセッションをブロックします。
AWS WAF トークンが必要 `aws-waf-token`	すべてのルールで必須です。 ACFP でアプリケーション統合 SDK を使用する理由 を参照してください。	多くのルールに必要です。 ATP でアプリケーション統合 SDK を使用する理由 を参照してください。	なし	あり
AWS WAF トークンを取得します。 `aws-waf-token`	はい。ルール `AllRequests` によって強制されます	なし	なし	トークンを取得する Challenge または CAPTCHA ルールアクションを使用するルールがあります。

これらのオプションに関連するコストの詳細については、「AWS WAF の料金表」のインテリジェントな脅威の軽減情報を参照してください。

レートベースのルールとターゲットを絞った Bot Control ルールにおけるレート制限のオプション

AWS WAF Bot Control ルールグループのターゲットレベルと AWS WAF レートベースのルールステートメントは、どちらもウェブリクエストのレート制限を提供します。以下の表は、これら 2 つのオプションを比較したものです。

レートベースの検出と緩和のためのオプションの比較

	AWS WAF レートベースのルール	AWS WAF Bot Control のターゲットルール	
レート制限の適用方法	レートが高すぎるリクエストのグループに対して動作します。を除く任意のアクションを適用できますAllow。	リクエストトークンの使用を通じて、人間のようなアクセスパターンを実施し、動的なレート制限を適用します。	
履歴的なトラフィックベースラインに基づく	なし	あり	
履歴的なトラフィックベースラインを蓄積するために要する時間	該当なし	動的しきい値の場合は 5 分。トークンがない場合は該当なし。	
軽減までの時間差	通常は 30～50 秒。最大で数分かかる場合もあります。	通常は 10 秒未満。最大で数分かかる場合もあります。	
緩和の対象	設定可能。スコープダウンステートメントと、IP アドレス、HTTP メソッド、クエリ文字列などの 1 つ以上の集約キーを	IP アドレスとクライアントセッション	

	AWS WAF レートベースのルール	AWS WAF Bot Control のターゲットルール	
	使用してリクエストをグループ化できます。		
緩和をトリガーするために必要なトラフィック量のレベル	Medium - 指定した時間枠で 100 リクエストまで可能です	低 – 低速なスクレイパーなどのクライアントパターンを検出するためのもの	
カスタマイズ可能なしきい値	あり	なし	
デフォルトの緩和アクション	コンソールのデフォルトは Block です。API にデフォルト設定はなく、設定が必要です。 これは、 以外の任意のルールアクションに設定できます Allow。	ルールグループのルールアクション設定は、トークンがない場合には Challenge、単一のクライアントセッションからの大量のトラフィックの場合には CAPTCHA になります。 これらのルールのいずれかを、任意の有効なルールアクションに設定できます。	
高度に分散された攻撃に対する耐久性	中 - IP アドレス制限に単独で最大 10,000 IP アドレス	中程度 – IP アドレスとトークン間の合計で 50,000 個に制限されます	

	AWS WAF レートベースのルール	AWS WAF Bot Control のターゲットルール	
AWS WAF 料金表	の標準料金に含まれています AWS WAF。	Bot Control のインテリジェントな脅威の軽減のターゲットレベルの料金に含まれています。	
詳細情報	レートベースのルールステートメント	AWS WAF Bot Control ルールグループ	

インテリジェントな脅威の軽減のためのベストプラクティス

インテリジェントな脅威の軽減機能の最も効果的でコスト効率性に優れた実装については、このセクションのベストプラクティスに従ってください。

- JavaScript およびモバイルアプリケーション統合 SDK の実装 — アプリケーション統合を実装して、ACFP、ATP、またはボットコントロール機能のフルセットを可能な限り最も効果的な方法で有効にします。マネージドルールグループは、SDK が提供するトークンを使用して、セッションレベルで正規のクライアントトラフィックを望ましくないトラフィックから分離させます。アプリケーション統合 SDK は、これらのトークンが常に利用可能であることを確実にします。詳細については、以下を参照してください。
  - ACFP でアプリケーション統合 SDK を使用する理由
  - ATP でアプリケーション統合 SDK を使用する理由
  - Bot Control でアプリケーション統合 SDK を使用する理由

  インテグレーションを使用してクライアントに課題を実装したり、エンドユーザーへの JavaScript CAPTCHA パズルの表示方法をカスタマイズしたりできます。詳細については、「AWS WAF クライアントアプリケーション統合」を参照してください。

  JavaScript API を使用して CAPTCHA パズルをカスタマイズし、ウェブ ACL CAPTCHA の任意の場所でルールアクションを使用する場合は、クライアントの AWS WAF CAPTCHA レスポンスの処理に関するガイダンスに従ってください。からのキャプチャレスポンスを処理する AWS WAFこのガイダンスは、ACFP マネージドルールグループや Bot Control マネージドルールグルー

プのターゲットを絞った保護レベルなど、CAPTCHA アクションを使用するすべてのルールに適用されます。

- 送信するリクエストを ACFP、ATP、Bot Control ルールグループに限定する — インテリジェント脅威軽減マネージドルールルールグループの使用には追加料金がかかります。 AWS ACFP ルールグループは、指定したアカウント登録エンドポイントと作成エンドポイントに対するリクエストを検査します。ATP ルールグループは、指定したログインエンドポイントに対するリクエストを検査します。Bot Control ルールグループは、ウェブ ACL 評価で、到達するすべてのリクエストを検査します。

  これらのルールグループの使用を減らすため、以下のアプローチを検討してください。

  - マネージドルールグループステートメント内のスコープダウンステートメントを使用して、検査からリクエストを除外します。これは、ネスト可能なステートメントならどれでも実行できます。詳細については、スコープダウンステートメント を参照してください。
  - ルールグループの前にルールを追加することで、検査からリクエストを除外します。スコープダウンステートメントで使用できないルール、およびラベル付けの後にラベルの照合が行われるような複雑な状況については、ルールグループの前に実行されるルールを追加する必要があるかもしれません。詳細については、「スコープダウンステートメント」および「ルールステートメントの基本」を参照してください。
  - 低料金のルールを実行してからルールグループを実行します。 AWS WAF 何らかの理由でリクエストをブロックする他の標準ルールがある場合は、これらの有料ルールグループよりも先にそのルールを実行してください。ルールとルール管理の詳細については、「ルールステートメントの基本」を参照してください。
  - インテリジェントな脅威の軽減のためのマネージドルールグループを複数使用している場合は、Bot Control、ATP、ACFP の順に実行すると、コストを抑えることができます。

  料金の詳細については、「AWS WAF の料金表」を参照してください。

- 通常のウェブトラフィック中に Bot Control ルールグループのターゲットを絞った保護レベルを有効にする – ターゲットを絞った保護レベルのルールの一部では、通常のトラフィックパターンのベースラインを確立してからでないと、不規則なトラフィックパターンや悪意のあるトラフィックパターンに対する認識と対応が行えません。例えば、`TGT_ML_*` ルールのウォームアップには最大 24 時間かかります。

  攻撃を受けていないときにこれらの保護を追加し、ルールが攻撃に対して適切に対応することを期待する前に、トラフィックパターンのベースラインを確立するための猶予時間を与えます。攻撃中にこれらのルールを追加した場合、攻撃が収まった後、ベースラインを確立するまでにかかる時間は通常の 2 倍から 3 倍になります。これは、攻撃トラフィックによって歪みが生じるためです。

ルールとルールのウォームアップに必要な時間に関する詳細については、「ルールの一覧」を参照してください。

- 分散型サービス拒否 (DDoS) からの保護には、Shield Advanced のアプリケーションレイヤー DDoS 自動緩和を使用する – インテリジェントな脅威の軽減ルールグループは DDoS 保護を提供しません。ACFP は、アプリケーションのサインアップページに対するアカウントの不正な作成の試みから保護します。ATP は、ログインページに対するアカウント乗っ取りの試みを防ぎます。Bot Control は、トークンとクライアントセッションに対する動的なレート制限を使用して、人間のようなアクセスパターンを実施することに重点を置いています。

  アプリケーションレイヤーの自動DDoS軽減を有効にしてShield Advancedを使用すると、Shield Advanced はユーザーに代わってカスタム緩和策を作成、評価、展開することで、検出された DDoS 攻撃に自動的に対応します。 AWS WAF Shield Advanced の詳細については、「AWS Shield Advanced 概要」および「AWS Shield Advanced アプリケーション層 (レイヤー 7) 保護」を参照してください。

- トークン処理を調整および設定する – 最高のユーザーエクスペリエンスが得られるように、ウェブ ACL のトークン処理を調整します。

  - 運用コストを削減し、エンドユーザーエクスペリエンスを改善するには、トークン管理のイミュニティ時間をセキュリティ要件が許容する最大時間に調整します。これにより、CAPTCHA パズルやサイレントチャレンジの使用を最小限に抑えることができます。詳細については、タイムスタンプの有効期限： AWS WAF トークンのイミュニティ時間 を参照してください。
  - 保護されたアプリケーション間でトークン共有を有効にするには、ウェブ ACL のトークンドメインリストを設定します。詳細については、AWS WAF トークンドメインとドメインリスト を参照してください。

- 任意のホスト仕様を持つリクエストを拒否する – ウェブリクエストの `Host` ヘッダーがターゲットリソースに一致することを必須とするように保護対象リソースを設定します。ホストについて、1 つの値、または特定の値セット (`myExampleHost.com` および `www.myExampleHost.com` など) を受け入れることはできますが、任意の値は受け入れないでください。

- CloudFront ディストリビューションのオリジンであるApplication Load Balancer については、CloudFront 適切なトークン処理を行うように設定してください。ウェブ ACL をApplication Load Balancer に関連付けて、アプリケーションロードバランサーをディストリビューションのオリジンとしてデプロイする場合は、を参照してください。 AWS WAF CloudFront CloudFront オリジンである Application Load Balancer に必要な設定

- デプロイ前にテストして調整する – ウェブ ACL に変更を実装する前に、本ガイドのテストおよび調整手順に従って、期待通りの動作が得られることを確認してください。これらの有料機能を使用する場合は特に重要です。一般的なガイダンスについては、「AWS WAF 保護機能のテストと調

整」を参照してください。有料マネージドルールグループ固有の情報については、「ACFP のテストとデプロイ」、「ATP のテストとデプロイ」、「AWS WAF Bot Control のテストとデプロイ」を参照してください。

AWS WAF ウェブリクエストトークン

AWS WAF トークンは、 AWS WAF インテリジェントな脅威の軽減によって提供される強化された保護の不可欠な部分です。トークンはフィンガープリントとも呼ばれ、クライアントが保存し、送信するすべてのウェブリクエストに提供する単一のクライアントセッションに関する情報のコレクションです。 はトークン AWS WAF を使用して、悪意のあるクライアントセッションを 1 つの IP アドレスから発信された場合でも、正規のセッションから識別して分離します。トークンの使用によるコストは、正規ユーザーにとってはごくわずかですが、ボットネットにとってはかなり高額になります。

AWS WAF はトークンを使用して、アプリケーション統合 SDKs とルールアクション Challenge および によって提供されるブラウザおよびエンドユーザーチャレンジ機能をサポートします CAPTCHA。さらに、トークンは Bot Control AWS WAF およびアカウント乗っ取り防止マネージドルールグループの機能を有効にします。

AWS WAF は、サイレントチャレンジや CAPTCHA パズルに正常に応答するクライアントのトークンを作成、更新、暗号化します。トークンを持つクライアントがウェブリクエストを送信すると、暗号化されたトークンが含まれ、トークンを復 AWS WAF 号化してその内容を確認します。

トピック

- AWS WAF トークンの使用方法
- AWS WAF トークンの特性
- タイムスタンプの有効期限： AWS WAF トークンのイミュニティ時間
- AWS WAF トークンドメインとドメインリスト
- AWS WAF ボットおよび不正マネージドルールグループによるトークンのラベル付け
- 有効な AWS WAF トークンがないリクエストのブロック
- CloudFront オリジンである Application Load Balancer に必要な設定

AWS WAF トークンの使用方法

AWS WAF トークンを使用して、以下の種類のクライアントセッション検証を記録および検証します。

- CAPTCHA — CAPTCHA パズルは、ボットと人間のユーザーを区別するうえで役立ちます。CAPTCHA は CAPTCHA ルールアクションによってのみ実行されます。パズルの完了が成功すると、CAPTCHA スクリプトはトークンの CAPTCHA タイムスタンプを更新します。詳細については、「CAPTCHAChallengeの および AWS WAF」を参照してください。
- チャレンジ — 通常のクライアントセッションとボットセッションを区別しやすくし、ボットの運用コストを高めるために、チャレンジはサイレントで実行されます。チャレンジが正常に完了すると、 AWS WAF チャレンジスクリプトは必要に応じて新しいトークンを自動的に取得し、トークンのチャレンジタイムスタンプを更新します。

  AWS WAF 以下の状況でチャレンジを実行します。

  - アプリケーション統合 SDK — アプリケーション統合 SDK は、クライアントアプリケーションセッション内で実行され、クライアントがチャレンジの応答に成功した後にのみログイン試行が許可されるようにします。詳細については、「AWS WAF クライアントアプリケーション統合」を参照してください。
  - Challenge ルールアクション – 詳細については、「CAPTCHAChallengeの および AWS WAF」を参照してください。
  - CAPTCHA — CAPTCHA インタースティシャルを実行するときにクライアントがまだトークンを持っていない場合、スクリプトは最初に自動的にチャレンジを実行し、クライアントセッションを検証してトークンを初期化します。

トークンは、 AWS インテリジェント脅威管理ルールルールグループの多くのルールで必要とされます。このルールは、セッションレベルでのクライアントの区別、ブラウザの特性の判断、アプリケーションウェブページにおける人間のインタラクティビティのレベルの理解などを行うためにトークンを使用します。 AWS WAF これらのルールグループはトークン管理を呼び出し、トークンのラベル付けを適用してルールグループが検査します。

- AWS WAF 不正防止アカウント作成詐欺防止 (ACFP) — ACFP ルールでは、有効なトークンを使用したウェブリクエストが義務付けられています。ルールの詳細については、「AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) ルールグループ」を参照してください。
- AWS WAF 不正防止アカウント乗っ取り防止 (ATP) — 大量かつ長期にわたるクライアントセッションを防止する ATP ルールでは、チャレンジタイムスタンプが期限切れでない有効なトークンを含む Web リクエストが必要です。詳細については、「AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループ」を参照してください。
- AWS WAF ボットコントロール — このルールグループの対象ルールは、有効なトークンなしでクライアントが送信できる Web リクエストの数を制限し、トークン・セッション・トラッキングを使用してセッションレベルの監視と管理を行います。必要に応じて、ルールは Challenge および

CAPTCHA ルールアクションを適用して、トークン取得および有効なクライアント動作を強制します。詳細については、「AWS WAF Bot Control ルールグループ」を参照してください。

AWS WAF トークンの特性

各トークンには次の特徴があります。

- トークンは、aws-waf-token という名前のクッキーに保存されます。
- トークンは暗号化されます。
- トークンは、次の情報を含む精度の高い識別子でクライアントセッションをフィンガープリントします。
  - クライアントがサイレントチャレンジに対して最後に成功した応答のタイムスタンプ。
  - エンドユーザーが CAPTCHA に対して最後に成功した応答のタイムスタンプ。これは保護機能で CAPTCHA を使用している場合にのみ表示されます。
  - 正規のクライアントを迷惑なトラフィックから切り離すうえで役立つクライアントおよびクライアント行動に関する追加情報。この情報には、自動化されたアクティビティを検出するために使用可能なさまざまなクライアント識別子およびクライアント側の信号が含まれます。収集される情報は一意ではなく、個別の人間を特定することはできません。
    - すべてのトークンには、オートメーションやブラウザ設定の不整合を示唆する要素など、クライアントブラウザの問い合わせから得られたデータが含まれています。この情報は、Challenge アクションによって実行されるスクリプトおよびクライアントアプリケーション SDK によって取得されます。スクリプトはブラウザに積極的に問い合わせて、結果をトークンに含めます。
    - さらに、クライアントアプリケーション統合 SDK を実装すると、トークンには、アプリケーションページとのエンドユーザーのインタラクティビティについて受動的に収集された情報が含まれます。インタラクティビティには、マウスの動き、キーの押下、およびページ上に存在する HTML フォームとのインタラクションが含まれます。この情報は、 AWS WAF がクライアントにおける人間のインタラクティビティのレベルを検出し、人間ではないように見えるユーザーにチャレンジを提示するのに役立ちます。クライアント側の統合については、「AWS WAF クライアントアプリケーション統合」を参照してください。

セキュリティ上の理由から、 AWS WAF トークンの内容の完全な説明やトークン暗号化プロセスに関する詳細情報は提供 AWS しないでください。

タイムスタンプの有効期限： AWS WAF トークンのイミュニティ時間

AWS WAF は、チャレンジと CAPTCHA イミュニティ時間を使用して、1 つのクライアントセッションにチャレンジまたは CAPTCHA を提示できる頻度を制御します。エンドユーザーが CAPTCHA に応答に成功した後、そのエンドユーザーに対して別の CAPTCHA が表示されない期間は、CAPTCHA イミュニティ時間によって決まります。同様に、チャレンジのイミュニティ時間は、チャレンジへの応答に成功した後、クライアントセッションが再度チャレンジを受けない期間を決定します。

AWS WAF は、トークン内の対応するタイムスタンプを更新することで、チャレンジまたは CAPTCHA への正常な応答を記録します。がトークンにチャレンジまたは CAPTCHA がないか AWS WAF 検査すると、現在の時刻からタイムスタンプが減算されます。結果が設定されたイミュニティ時間よりも大きい場合、タイムスタンプは期限切れになります。

チャレンジおよび CAPTCHA のイミュニティ時間は、ウェブ ACL および CAPTCHA または Challenge ルールアクションを使用する任意のルールで設定できます。

- 両方のイミュニティ時間におけるデフォルトのウェブ ACL 設定は 300 秒です。
- CAPTCHA または Challenge アクションを使用するすべてのルールにイミュニティ時間を指定できます。ルールにイミュニティ時間を指定しない場合、ウェブ ACL の設定が継承されます。
- CAPTCHA または Challenge アクションを使用するルールグループ内のルールには、ルールのイミュニティ時間を定義しない場合、ルールグループを使用する各ウェブ ACL から設定が継承されます。
- アプリケーション統合 SDK は、ウェブ ACL のチャレンジイミュニティ時間を使用します。

チャレンジイミュニティ時間の最小値は 300 秒です。CAPTCHA イミュニティ時間の最小値は 60 秒です。両方のイミュニティ時間の最大値は 259,200 秒、または 3 日間です。

ウェブ ACL およびルールレベルのイミュニティ時間設定を使用して、CAPTCHA アクション、Challenge、SDK チャレンジ管理の動作を調整できます。たとえば、機密性の高いデータへのアクセスを制御するルールを低いイミュニティ時間で設定し、その後にウェブ ACL 内で他のルールや SDK が承継するより高いイミュニティ時間を設定できます。

特に CAPTCHA の場合、パズルを解くことは顧客のウェブサイトエクスペリエンスを低下させる恐れがあるため、CAPTCHA のイミュニティ時間を調整すると必要な保護を提供し続けながら、顧客エクスペリエンスへの影響を軽減することに役立ちます。

Challenge および CAPTCHA ルールアクションを使用する際にイミュニティ時間の調整に関する詳細については、「CAPTCHA および Challenge アクションを使用するベストプラクティス」を参照してください。

AWS WAF トークンのイミュニティ時間を設定する場所

イミュニティ時間は、ウェブ ACL および Challenge や CAPTCHA ルールアクションを使用するルールで設定できます。

ウェブ ACL およびそのルールの管理に関する一般情報については、「ウェブ ACL の使用」を参照してください。

ウェブ ACL のイミュニティ時間を設定する場所

- コンソール — ウェブ ACL を編集するとき、[Rules] (ルール) タブで、[Web ACL CAPTCHA configuration] (ウェブ ACL の CAPTCHA 設定) および [Web ACL Challenge configuration] (ウェブ ACL のチャレンジ設定) ペインの設定を編集して変更します。コンソールでは、ウェブ ACL を作成した後にのみ、ウェブ ACL の CAPTCHA およびチャレンジのイミュニティ時間を設定できます。
- コンソールの外部 – ウェブ ACL のデータタイプには CAPTCHA およびチャレンジの設定パラメータがあり、ウェブ ACL の作成および更新操作に設定して提供できます。

ルールのイミュニティ時間を設定する場所

- コンソール – ルールを作成または編集して CAPTCHA または Challenge アクションを指定すると、ルールのイミュニティ時間設定を変更できます。
- コンソールの外部 – ルールのデータタイプには CAPTCHA およびチャレンジの設定パラメータがあり、ルールを定義するときに設定できます。

AWS WAF トークンドメインとドメインリスト

AWS WAF がクライアントのトークンを作成すると、トークンドメインで設定されます。 AWS WAF がウェブリクエスト内のトークンを検査するとき、そのドメインがウェブ ACL で有効と見なされるドメインと一切一致しない場合、そのトークンは無効として拒否されます。

デフォルトでは、 は、ドメイン設定がウェブ ACL に関連付けられているリソースのホストドメインと完全に一致するトークン AWS WAF のみを受け入れます。これはウェブリクエスト内の `Host` ヘッダーの値です。ブラウザでは、このドメインは JavaScript `window.location.hostname`プロパティと、ユーザーがアドレスバーに表示するアドレスにあります。

次のセクションで説明するように、ウェブ ACL 設定で許容されるトークンドメインを指定することもできます。この場合、 はホストヘッダーと完全一致とトークンドメインリストのドメインと完全一致の両方 AWS WAF を受け入れます。

ドメインを設定するとき AWS WAF 、およびウェブ ACL でトークンを評価するときに使用する のトークンドメインを指定できます。指定するドメインには gov.au など、パブリックサフィックスを使用できません。使用できないドメインについては、「パブリックサフィックスリスト」のリスト (https://publicsuffix.org/list/public_suffix_list.dat) を参照してください。

AWS WAF ウェブ ACL トークンドメインリストの設定

トークンドメインリストに受け入れる追加のドメインを指定することで、複数の保護されたリソース間でトークンを共有する AWS WAF ようにウェブ ACL を設定できます。トークンドメインリストでは、 AWS WAF Seltitle はリソースのホストドメインを受け入れます。さらに、プレフィックス付きのサブドメインを含め、トークンドメインリスト内のすべてのドメインを受け入れます。

たとえば、トークンドメインリスト内のドメイン仕様 `example.com` は `example.com` (`http://example.com/` から)、`api.example.com` (`http://api.example.com/` から)、`www.example.com` (`http://www.example.com/` から) と一致します。`example.api.com` (`http://example.api.com/` から) または `apiexample.com` (`http://apiexample.com/` から) と一致しません。

トークンドメインリストは、作成または編集するときにウェブ ACL で設定できます。ウェブ ACL の管理に関する一般情報については、「ウェブ ACL の使用」を参照してください。

AWS WAF トークンドメイン設定

AWS WAF は、アプリケーション統合 SDKs と Challengeおよび CAPTCHAルールアクションによって実行されるチャレンジスクリプトのリクエスト時にトークンを作成します。

がトークン AWS WAF に設定するドメインは、トークンをリクエストするチャレンジスクリプトのタイプと、指定した追加のトークンドメイン設定によって決まります。 AWS WAF は、トークンのドメインを、設定で確認できる最も短く、最も一般的な設定に設定します。

- JavaScript SDK – JavaScript SDK は、1 つ以上のドメインを含めることができるトークンドメイン仕様で設定できます。設定するドメインは、保護されたホストドメインとウェブ ACL のトークンドメインリストに基づいて、 が AWS WAF 受け入れるドメインである必要があります。

  がクライアントのトークン AWS WAF を発行すると、ホストドメインと設定済みリスト内のドメインの中から、ホストドメインと一致する最短のトークンドメインが設定されます。例えば、ホス

トドメインが api.example.comで、トークンドメインリストに がある場合example.com、 はトークンexample.comで AWS WAF 使用されます。これは、ホストドメインと一致し、短いためです。 JavaScript API 設定でトークンドメインリストを指定しない場合、 はドメインを保護されたリソースのホストドメイン AWS WAF に設定します。

詳細については、「トークンで使用するドメインの提供」を参照してください。

- モバイル SDK — アプリケーションコードでは、トークンドメインプロパティでモバイル SDK を設定する必要があります。このプロパティは、保護されたホストドメインおよびウェブ ACL のトークンドメインリストに基づき、 AWS WAF が受け入れるドメインでなければなりません。

  がクライアントのトークン AWS WAF を発行する場合、このプロパティをトークンドメインとして使用します。モバイル SDK AWS WAF クライアントに対して発行するトークンではホストドメインを使用しません。

  詳細については、「AWS WAF モバイル SDK 仕様」で WAFConfiguration domainName 設定を参照してください。

- Challenge アクション – ウェブ ACL でトークンドメインリストを指定すると、 は、ホストドメインとリスト内のドメインの中から、ホストドメインと一致する最短のトークンドメイン AWS WAF に設定します。例えば、ホストドメインが api.example.comで、トークンドメインリストに がある場合example.com、ホストドメインと一致し、短いため、トークンexample.comで AWS WAF を使用します。ウェブ ACL でトークンドメインリストを指定しない場合、 はドメインを保護されたリソースのホストドメイン AWS WAF に設定します。

AWS WAF ボットおよび不正マネージドルールグループによるトークンのラベル付け

このセクションでは、 AWS WAF トークン管理がウェブリクエストに追加するラベルについて説明します。ラベルの一般的な情報については、「」を参照してくださいAWS WAF ウェブリクエストの ラベル。

AWS WAF ボットまたは不正コントロールマネージドルールグループのいずれかを使用する場合、ルールグループは AWS WAF トークン管理を使用してウェブリクエストトークンを検査し、リクエストにトークンラベルを適用します。マネージドルールグループの詳細については、「AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) ルールグループ」、「AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループ」、「AWS WAF Bot Control ルールグループ」を参照してください。

> **Note**
>
> AWS WAF は、これらのインテリジェントな脅威の軽減マネージドルールグループのいずれかを使用する場合にのみ、トークンラベルを適用します。

トークン管理では、以下のラベルをウェブリクエストに追加できます。

クライアントセッションラベル

ラベルには、 AWS WAF トークン管理がクライアントセッションを識別するために使用する一意の識別子`awswaf:managed:token:id:`*`identifier`*が含まれています。この識別子は、クライアントが使用していたトークンを破棄した後など、新しいトークンを取得すると変わる可能性があります。

> **Note**
>
> AWS WAF は、このラベルの Amazon CloudWatch メトリクスを報告しません。

トークンステータスラベル: ラベル名前空間プレフィックス

トークンステータスラベルは、トークン、チャレンジのステータス、およびそれに含まれる CAPTCHA 情報を報告します。

各トークンステータスラベルは、次のプレフィクスの 1 つで始まります。

- `awswaf:managed:token:`— トークンの一般的なステータスを報告したり、トークンのチャレンジ情報のステータスを報告したりするために使用されます。
- `awswaf:managed:captcha:`— トークンの CAPTCHA 情報のステータスを報告するために使用されます。

トークンステータスラベル: ラベル名

プレフィックスに続いて、ラベルの残りの部分には詳細なトークンステータス情報が表示されます。

- `accepted` - リクエストトークンが存在し、以下の内容が含まれています。
  - 有効なチャレンジまたは CAPTCHA ソリューション。

- 有効期限が切れていないチャレンジまたは CAPTCHA タイムスタンプ。
- ウェブ ACL に有効なドメイン仕様。

例: ラベル awswaf:managed:token:accepted には、ウェブリクエストのトークンに有効なチャレンジソリューション、有効期限が切れていないチャレンジタイムスタンプ、および有効なドメインがあることが示されています。

- rejected - リクエストトークンは存在するが、承認基準を満たしていない。

  トークン管理では、拒否されたラベルに加えて、理由を示すカスタムラベル名前空間と名前が追加されます。

  - rejected:not_solved — トークンにチャレンジまたは CAPTCHA ソリューションがない。
  - rejected:expired — ウェブ ACL に設定されているトークンイミュニティ時間によると、トークンのチャレンジまたは CAPTCHA タイムスタンプの有効期限が切れている。
  - rejected:domain_mismatch — トークンのドメインが、ウェブ ACL のトークンドメイン設定と一致しない。
  - rejected:invalid - 指定されたトークンを読み AWS WAF 取れませんでした。

  例: ラベル awswaf:managed:captcha:rejected と awswaf:managed:captcha:rejected:expired には、トークンの CAPTCHA タイムスタンプがウェブ ACL で設定されている CAPTCHA トークンのイミュニティ時間を超えたためにリクエストが拒否されたことが示されています。

- absent — リクエストにトークンがないか、トークンマネージャーがそれを読み取れなかった。

  例: ラベル awswaf:managed:captcha:absent には、リクエストにトークンがないことが示されています。

有効な AWS WAF トークンがないリクエストのブロック

インテリジェントな脅威に対応した AWS マネージドルールのルールグループ AWSManagedRulesACFPRuleSet、AWSManagedRulesATPRuleSet、および を使用するとAWSManagedRulesBotControlRuleSet、ルールグループは AWS WAF トークン管理を呼び出してウェブリクエストトークンのステータスを評価し、それに応じてリクエストにラベルを付けます。

> **Note**
>
> トークンのラベル付けは、これらのマネージドルールグループのいずれかを使用して評価したウェブリクエストにのみ適用されます。

適用されるトークン管理のラベル付けについては、前述の「AWS WAF ボットおよび不正マネージドルールグループによるトークンのラベル付け」セクションを参照してください。

その後、インテリジェントな脅威軽減マネージドルールグループは、トークンの要件を次のように処理します。

- `AWSManagedRulesACFPRuleSet` `AllRequests` ルールは、すべてのリクエストに対して Challenge アクションを実行するように設定されており、`accepted` トークンラベルのないリクエストは効果的にブロックされます。
- `AWSManagedRulesATPRuleSet` は、`rejected` トークンラベルを持つリクエストをブロックしますが、`absent` トークンラベルを持つリクエストはブロックしません。
- `accepted`トークンラベルなしでリクエストを 5 回送信すると、`AWSManagedRulesBotControlRuleSet` のターゲットを絞った保護レベルが、クライアントにチャレンジを送信します。有効なトークンを持たない個別のリクエストはブロックされません。ルールグループの共通の保護レベルでは、トークンの要件は管理されません。

インテリジェントな脅威ルールグループの詳細については、「AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) ルールグループ」、「AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループ」、および「AWS WAF Bot Control ルールグループ」を参照してください。

Bot Control または ATP マネージドルールグループの使用時にトークンを持たないリクエストをブロックするには

Bot Control と ATP ルールグループを使用すると、有効なトークンを持たないリクエストがルールグループの評価を終了し、ウェブ ACL によって引き続き評価される可能性があります。

トークンが不足しているリクエスト、あるいはトークンが拒否されたリクエストをすべてブロックするには、マネージドルールグループの直後に実行するルールを追加し、ルールグループがユーザーに代わって処理しないリクエストをキャプチャしてブロックします。

次の内容では、ATP マネージドルールグループを使用するウェブ ACL の JSON リストの例を示します。ウェブ ACL には、awswaf:managed:token:absent ラベルをキャプチャして処理するルールが追加されています。このルールは、ATP ルールグループの範囲に合わせて、ログインエンドポイントに送信されるウェブリクエストに評価を絞り込みます。追加されたルールは太字で表示されます。

```
{
  "Name": "exampleWebACL",
  "Id": "55555555-6666-7777-8888-999999999999",
  "ARN": "arn:aws:wafv2:us-east-1:111111111111:regional/webacl/
exampleWebACL/55555555-4444-3333-2222-111111111111",
  "DefaultAction": {
    "Allow": {}
  },
  "Description": "",
  "Rules": [
    {
      "Name": "AWS-AWSManagedRulesATPRuleSet",
      "Priority": 1,
      "Statement": {
        "ManagedRuleGroupStatement": {
          "VendorName": "AWS",
          "Name": "AWSManagedRulesATPRuleSet",
          "ManagedRuleGroupConfigs": [
            {
              "AWSManagedRulesATPRuleSet": {
                "LoginPath": "/web/login",
                "RequestInspection": {
                  "PayloadType": "JSON",
                  "UsernameField": {
                    "Identifier": "/form/username"
                  },
                  "PasswordField": {
                    "Identifier": "/form/password"
                  }
                },
                "ResponseInspection": {
                  "StatusCode": {
                    "SuccessCodes": [
                      200
                    ],
                    "FailureCodes": [
                      401,
```

```
                    403,
                    500
                  ]
                }
              }
            }
          }
        ]
      }
    },
    "OverrideAction": {
      "None": {}
    },
    "VisibilityConfig": {
      "SampledRequestsEnabled": true,
      "CloudWatchMetricsEnabled": true,
      "MetricName": "AWS-AWSManagedRulesATPRuleSet"
    }
  },
  {
    "Name": "RequireTokenForLogins",
    "Priority": 2,
    "Statement": {
      "AndStatement": {
        "Statements": [
          {
            "Statement": {
              "LabelMatchStatement": {
                "Scope": "LABEL",
                "Key": "awswaf:managed:token:absent"
              }
            }
          },
          {
            "ByteMatchStatement": {
              "SearchString": "/web/login",
              "FieldToMatch": {
                "UriPath": {}
              },
              "TextTransformations": [
                {
                  "Priority": 0,
                  "Type": "NONE"
                }
```

```
                ],
                "PositionalConstraint": "STARTS_WITH"
              }
            },
            {
              "ByteMatchStatement": {
                "SearchString": "POST",
                "FieldToMatch": {
                  "Method": {}
                },
                "TextTransformations": [
                  {
                    "Priority": 0,
                    "Type": "NONE"
                  }
                ],
                "PositionalConstraint": "EXACTLY"
              }
            }
          ]
        }
      },
      "Action": {
        "Block": {}
      },
      "VisibilityConfig": {
        "SampledRequestsEnabled": true,
        "CloudWatchMetricsEnabled": true,
        "MetricName": "RequireTokenForLogins"
      }
    }
  ],
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "exampleWebACL"
  },
  "Capacity": 51,
  "ManagedByFirewallManager": false,
  "LabelNamespace": "awswaf:111111111111:webacl:exampleWebACL:"
}
```

CloudFront オリジンである Application Load Balancer に必要な設定

ウェブ ACL を Application Load Balancer に関連付け、Application Load Balancer を CloudFront ディストリビューションのオリジンとしてデプロイする場合は、このセクションをお読みください。

このアーキテクチャでは、トークン情報が正しく処理されるには、次の追加設定を行う必要があります。

- Cookie `aws-waf-token` を Application Load Balancer CloudFront に転送するように を設定します。デフォルトでは、 はオリジンに転送する前にウェブリクエストから Cookie CloudFront を削除します。ウェブリクエストでトークン Cookie を保持するには、トークン Cookie のみまたはすべての Cookie を含めるように CloudFront キャッシュ動作を設定します。これを行う方法については、「Amazon CloudFront デベロッパーガイド」の「Cookie に基づくコンテンツのキャッシュ」を参照してください。
- ディストリビューションのドメインを CloudFront有効なトークンドメインとして認識 AWS WAF するように を設定します。デフォルトでは、 は `Host`ヘッダーを Application Load Balancer オリジン CloudFront に設定し、 AWS WAF それを保護されたリソースのドメインとして使用します。ただし、クライアントブラウザはディストリビューションを CloudFrontホストドメインと見なし、クライアント用に生成されたトークンは CloudFront ドメインをトークンドメインとして使用します。追加の設定がない場合、 が保護されたリソースドメイン AWS WAF をトークンドメインと照合すると、不一致が発生します。これを修正するには、ウェブ ACL 設定のトークンドメインリストに CloudFront ディストリビューションドメイン名を追加します。これを行う方法については、「AWS WAF ウェブ ACL トークンドメインリストの設定」を参照してください。

AWS WAF 不正防止アカウント作成詐欺防止 (ACFP)

アカウント作成の不正行為は、攻撃者が 1 つ以上の偽のアカウントの作成を試みるオンライン上の違法行為です。攻撃者は、プロモーションやサインアップボーナスの濫用、なりすまし、フィッシングなどのサイバー攻撃などの不正行為のために偽のアカウントを使用します。偽のアカウントの存在は、顧客からの評判に傷をつけたり、金銭的な被害を伴う不正行為のリスクを生じさせたりするものであり、ビジネスに悪影響を及ぼす可能性があります。

Fraud Control アカウント作成詐欺防止 (ACFP) 機能を実装することで、 AWS WAF アカウント作成の不正行為を監視および管理できます。 AWS WAF は、同梱のアプリケーション統合 SDK `AWSManagedRulesACFPRuleSet` とともに AWS Managed Rules ルールグループでこの機能を提供しています。

ACFP マネージドルールグループは、悪意のあるアカウント作成の試みの一部である可能性があるリクエストにラベルを付けて管理します。ルールグループは、クライアントでアプリケーションのアカウントサインアップエンドポイントに送信するアカウント作成の試みを検査することでこれを行います。

ACFP は、アカウントのサインアップリクエストをモニタリングして異常なアクティビティがないかを確認し、疑わしいリクエストを自動的にブロックすることで、アカウントサインアップページを保護します。ルールグループは、リクエスト ID、行動分析、機械学習を使用して不正なリクエストを検出します。

- 検査のリクエスト– ACFP を使用すると、アカウントの異常な作成の試みや、盗まれた認証情報を使用する試みを可視化して制御でき、不正なアカウントの作成を防止できます。ACFP は、盗まれた認証情報のデータベースに照らして E メールとパスワードの組み合わせをチェックします。このデータベースは、漏えいされた認証情報がダークウェブ上で新しく見つかると定期的に更新されます。ACFP は、メールアドレスで使用されているドメインを評価し、電話番号や住所のフィールドの使用をモニタリングして、エントリを検証するとともに、不正行為を検出します。ACFP は、IP アドレスやクライアントセッションごとにデータを集約し、不審なリクエストを大量に送信するクライアントを検出してブロックします。
- レスポンス検査 — CloudFront ディストリビューションの場合、ACFP ルールグループは、受信したアカウント作成リクエストを検査するだけでなく、アカウント作成の試行に対するアプリケーションの応答を検査し、成功率と失敗率を追跡します。この情報を使用して、ACFP は失敗した試行回数が過度に多いクライアントセッションまたは IP アドレスを一時的にブロックできます。AWS WAF は、レスポンス検査を非同期で実行するため、ウェブトラフィックのレイテンシーが大きくなることはありません。

> **Note**
>
> このマネージドルールグループを使用する場合、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

> **Note**
>
> ACFP 機能は、Amazon Cognito ユーザープールでは使用できません。

トピック

AWS WAF ACFP コンポーネント

AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) の主なコンポーネントは次のとおりです。

- **AWSManagedRulesACFPRuleSet** – この AWS マネージドルールルールグループのルールは、さまざまなタイプの不正なアカウント作成アクティビティを検出、ラベル付け、処理します。ルールグループは、指定されたアカウント登録エンドポイントにクライアントが送信する HTTP GET テキスト/HTML リクエストと、指定されたアカウントサインアップエンドポイントにクライアントが送信する POST ウェブリクエストを検査します。保護された CloudFront ディストリビューションの場合、ルールグループはディストリビューションがアカウント作成リクエストに送り返すレスポンスも検査します。このルールグループのルールのリストについては、「AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) ルールグループ」を参照してください。マネージドルールグループ参照ステートメントを使用して、このルールグループをウェブ ACL に含めます。このルールグループの使用については、「ACFP マネージドルールグループをウェブ ACL に追加」を参照してください。

  > **Note**
  >
  > このマネージドルールグループを使用する場合、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

- アプリケーションのアカウント登録ページと作成ページに関する詳細 –ウェブ ACL に AWSManagedRulesACFPRuleSet ルールグループを追加する際には、アカウント登録ページと作成ページに関する情報を提供する必要があります。これにより、ルールグループは検査するリクエストの範囲を絞り込み、アカウント作成ウェブリクエストを適切に検証できます。登録ページは GET テキスト/HTML リクエストを受け入れる必要があります。アカウント作成パスは POST リクエストを受け入れる必要があります。ACFP ルールグループは、電子メール形式のユーザー名に対応します。詳細については、「ACFP マネージドルールグループをウェブ ACL に追加」を参照してください。

- 保護された CloudFront ディストリビューションの場合、アプリケーションがアカウント作成の試行にどのように応答するかに関する詳細 – アカウント作成の試行に対するアプリケーションの応答に関する詳細を指定すると、ACFP ルールグループは 1 つの IP アドレスまたは 1 つのクライアントセッションから一括アカウント作成の試行を追跡および管理します。このオプションの設定については、「ACFP マネージドルールグループをウェブ ACL に追加」を参照してください。
- JavaScript およびモバイルアプリケーション統合 SDKs – ACFP 実装で AWS WAF JavaScript および モバイル SDKs を実装して、ルールグループが提供する機能の完全なセットを有効にします。ACFP ルールの多くは、セッションレベルのクライアント検証および動作集約に SDK から提供された情報を使用し、正規のクライアントトラフィックをボットトラフィックから分離するために必要です。SDK の詳細については、「AWS WAF クライアントアプリケーション統合」を参照してください。

ACFP 実装を次と組み合わせて、保護のモニタリング、チューニング、およびカスタマイズに役立てることができます。

- ログ記録とメトリクス – ログ、Amazon Security Lake データ収集、ウェブ ACL の Amazon CloudWatch メトリクスを設定および有効にすることで、トラフィックをモニタリングし、ACFP マネージドルールグループがトラフィックに与える影響を理解できます。がウェブリクエストAWSManagedRulesACFPRuleSetに追加するラベルは、データに含まれます。オプションの詳細については、、AWS WAF ウェブ ACL トラフィックのログ記録、Amazon によるモニタリング CloudWatchおよび Amazon Security Lake とはを参照してください。

  ニーズと確認できるトラフィックに応じて、AWSManagedRulesACFPRuleSet の実装をカスタマイズできます。例えば、一部のトラフィックを ACFP 評価から除外したり、スコープダウンステートメントやラベルマッチングルールなどの AWS WAF 機能を使用して、識別したアカウント作成の不正試行の処理方法を変更したりできます。
- ラベルとラベル一致ルール – AWSManagedRulesACFPRuleSet のどのルールでも、ブロック動作をカウントに切り替えて、ルールによって追加されたラベルと照合することができます。このアプローチを使用し、ACFP マネージドルールグループによって識別されるウェブリクエストの処理方法をカスタマイズします。ラベル付けおよびラベル一致ステートメントの使用の詳細については、「ラベル一致ルールステートメント」および「AWS WAF ウェブリクエストの ラベル」を参照してください。
- カスタムリクエストとレスポンス - 許可するリクエストにはカスタムヘッダーを追加し、ブロックするリクエストにはカスタムレスポンスを送信できます。これを行うには、ラベル一致を AWS WAF カスタムリクエストおよび応答機能とペアリングします。リクエストとレスポンスをカスタ

マイズする方法については、「AWS WAFのカスタマイズされたウェブリクエストとレスポンス」を参照してください。

ACFP でアプリケーション統合 SDK を使用する理由

ACFP ルールグループを最も効率的に使用するためにも、アプリケーション統合 SDK を実装することを強くお勧めします。

- 完全なルールグループ機能 – ACFP ルール `SignalClientHumanInteractivityAbsentLow` は、アプリケーション統合によって情報が提供されたトークンでのみ機能します。このルールは、アプリケーションページに対する人間の異常なインタラクションを検出および管理します。アプリケーション統合 SDK は、マウスの動き、キーの押下、その他の測定を通じて、人間の通常のインタラクティビティを検出できます。ルールアクション CAPTCHA および Challenge によって送信されるインタースティシャルは、このタイプのデータを提供できません。
- レイテンシーの短縮 – ルールグループのルール `AllRequests` は、チャレンジトークンをまだ持っていないリクエストに Challenge ルールアクションを適用します。これが発生すると、リクエストはルールグループによって 2 回評価されます。1 回目の評価はトークンなしで実行され、2 回目の評価は Challenge アクションインタースティシャルによってトークンが取得された後に実行されます。`AllRequests` ルールを使用するだけであれば追加料金は発生しませんが、このアプローチではウェブトラフィックに対するオーバーヘッドが大きくなり、エンドユーザーエクスペリエンスのレイテンシーが長くなります。アプリケーション統合を使用してクライアント側でトークンを取得する場合、アカウント作成リクエストを送信する前に、ACFP ルールグループがリクエストを 1 回評価します。

ルールグループ機能の情報については、「AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) ルールグループ」を参照してください。

SDK の詳細については、「AWS WAF クライアントアプリケーション統合」を参照してください。AWS WAF トークンの詳細については、「」を参照してくださいAWS WAF ウェブリクエストトークン。ルールアクションの情報については、「CAPTCHAChallengeの および AWS WAF」を参照してください。

ACFP マネージドルールグループをウェブ ACL に追加

ウェブトラフィックのアカウントの不正な作成アクティビティを認識するように ACFP マネージドルールグループを設定するには、クライアントが登録ページにアクセスする方法、およびアプリケー

ションにアカウント作成リクエストを送信する方法に関する情報を入力します。保護対象の Amazon CloudFront ディストリビューションでは、アプリケーションがアカウント作成リクエストにどのように応答するかについての情報も提供します。この設定は、マネージドルールグループの通常の設定に追加されます。

ルールグループの説明とルールリストについては、「AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) ルールグループ」を参照してください。

> **Note**
>
> 盗まれた認証情報の ACFP データベースには、E メール形式のユーザー名のみが含まれています。

このガイダンスは、 AWS WAF ウェブ ACL、ルール、およびルールグループを作成および管理する方法を一般的に認識しているユーザーを対象としています。これらのトピックは、このガイドの前のセクションでカバーされています。マネージドルールグループをウェブ ACL に追加する方法の基本については、「コンソールを通じたウェブ ACL へのマネージドルールグループの追加」を参照してください。

ベストプラクティスに従う

ACFP ルールグループは、「インテリジェントな脅威の軽減のためのベストプラクティス」に記載されているベストプラクティスに従って使用してください。

ウェブ ACL で **`AWSManagedRulesACFPRuleSet`** ルールグループを使用するには

1. AWS `AWSManagedRulesACFPRuleSet`マネージドルールグループをウェブ ACL に追加し、保存する前にルールグループの設定を編集します。

   > **Note**
   >
   > このマネージドルールグループを使用する場合、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

2. [ルールグループを設定] ペインで、ACFP ルールグループがアカウント作成リクエストの検査に使用する情報を入力します。

   a. [パスに正規表現を使用] で、登録ページとアカウント作成ページのパス指定に合わせて正規表現による照合を行う場合は AWS WAF 、これをオンに切り替えます。

AWS WAF PCRE `libpcre` ライブラリで使用されているパターン構文をサポートしますが、一部例外があります。ライブラリは、「PCRE - Perl Compatible Regular Expressions」で文書化されています。 AWS WAF サポートについて詳しくは、を参照してくださいでの正規表現パターンマッチング AWS WAF。

b. [登録ページのパス] で、アプリケーションの登録ページのエンドポイントのパスを指定します。このページは GET テキスト/HTML リクエストを受け入れる必要があります。ルールグループは、指定された登録ページのエンドポイントに対する HTTP GET テキスト/HTML リクエストのみを検査します。

> **Note**
>
> エンドポイントの照会では大文字と小文字が区別されません。正規表現の仕様には、大文字と小文字を区別しない照合を無効にするフラグ (`?-i`) を含めてはいけません。文字列の指定はフォワードスラッシュ「/」で始まる必要があります。

例えば、URL `https://example.com/web/registration` では、文字列パスの指定「/web/registration」を指定できます。指定したパスで始まる登録ページのパスは一致とみなされます。例えば、`/web/registration` は登録パス `/web/registration`、`/web/registration/`、`/web/registrationPage`、および `/web/registration/thisPage` に一致しますが、パス `/home/web/registration` または `/website/registration` には一致しません。

> **Note**
>
> エンドユーザーがアカウント作成リクエストを送信する前に、登録ページをロードするようにします。これは、クライアントからのアカウント作成リクエストに、有効なトークンが確実に含まれているようにするのに役立ちます。

c. [アカウント作成パス]には、入力済みの新規ユーザー情報を受け入れるウェブサイトの URI を指定します。この URI は POST リクエストを受け入れる必要があります。

> **Note**
>
> エンドポイントの照会では大文字と小文字が区別されません。正規表現の仕様には、大文字と小文字を区別しない照合を無効にするフラグ (`?-i`) を含めてはいけません。文字列の指定はフォワードスラッシュ「/」で始まる必要があります。

例えば、URL `https://example.com/web/newaccount` では、文字列パスの指定「`/web/newaccount`」を指定できます。指定したパスで始まるアカウント作成パスは一致とみなされます。例えば、`/web/newaccount` はアカウント作成パス `/web/newaccount`、`/web/newaccount/`、`/web/newaccountPage`、および `/web/newaccount/thisPage` に一致しますが、パス `/home/web/newaccount` または `/website/newaccount` には一致しません。

d. [リクエスト検査] で、リクエストのペイロードタイプと、ユーザー名、パスワード、他のアカウント作成の詳細が指定されているリクエスト本文内のフィールドの名前を指定して、アプリケーションがアカウント作成の試みを受け入れる方法を指定します。

> **Note**
>
> 主な住所フィールドと電話番号フィールドについては、リクエストペイロードに表示される順にフィールドを指定します。

これらのフィールド名の指定は、ペイロードタイプによって異なります。

- JSON ペイロードタイプ – JSON Pointer 構文でフィールド名を指定します。JSON ポインター構文については、インターネット技術標準化委員会 (IETF) のドキュメント「JavaScriptオブジェクト表記 (JSON) ポインター」を参照してください。

  例えば、次の JSON ペイロードの例では、ユーザー名フィールドの仕様は `/signupform/username` で、主な住所フィールドの仕様は `/signupform/addrp1`、`/signupform/addrp2`、および `/signupform/addrp3` です。

```
{
    "signupform": {
        "username": "THE_USERNAME",
        "password": "THE_PASSWORD",
```

```
            "addrp1": "PRIMARY_ADDRESS_LINE_1",
            "addrp2": "PRIMARY_ADDRESS_LINE_2",
            "addrp3": "PRIMARY_ADDRESS_LINE_3",
            "phonepcode": "PRIMARY_PHONE_CODE",
            "phonepnumber": "PRIMARY_PHONE_NUMBER"
        }
}
```

- FORM_ENCODED ペイロードタイプ – HTML 形式の名前を使用します。

  例えば、`username1` と `password1` という名前のユーザーおよびパスワードの入力要素を持つ HTML フォームの場合、ユーザー名フィールドの指定は `username1` で、パスワードフィールドの指定は `password1` です。

e. Amazon CloudFront ディストリビューションを保護する場合は、「レスポンス検査」で、アカウント作成の試行に対する応答でアプリケーションがどのように成功または失敗を示すかを指定します。

> **Note**
>
> ACFP レスポンスインスペクションは、ディストリビューションを保護するウェブ ACL でのみ使用できます。 CloudFront

ACFP で検査するアカウント作成レスポンスのコンポーネントを 1 つ指定します。Body コンポーネントタイプと JSON コンポーネントタイプでは、コンポーネントの最初の 65,536 バイト (64 KB) AWS WAF を検査できます。

インターフェイスに示されているように、コンポーネントタイプの検査基準を指定します。コンポーネント内で検査する成功基準と失敗基準の両方を指定する必要があります。

例えば、アプリケーションがアカウント作成の試みのステータスをレスポンスのステータスコードで示し、成功の場合は「`200 OK`」、失敗の場合は「`401 Unauthorized`」または「`403 Forbidden`」を使用するとします。レスポンス検査の [コンポーネントタイプ] を [ステータスコード] に設定し、[成功] テキストボックスに「`200`」と入力し、[失敗] テキストボックスの 1 行目に「`401`」、2 行目に「`403`」と入力します。

ACFP ルールグループは、成功または失敗の検査基準に一致するレスポンスのみをカウントします。ルールグループのルールは、アカウントの一括作成の試みを軽減するために、カウントされるレスポンスの成功率が高すぎるクライアントに対してアクションを実行します。

ルールグループのルールが正確に動作するように、アカウント作成の試みの成功と失敗の両方に関する詳細な情報を必ず入力してください。

アカウント作成のレスポンスを検査するルールを確認するには、「AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) ルールグループ」のルールリストで `VolumetricIPSuccessfulResponse` と `VolumetricSessionSuccessfulResponse` を探します。

3. ルールグループに必要な追加設定を指定します。

   マネージドルールグループステートメントにスコープダウンステートメントを追加することで、ルールグループが検査するリクエストの範囲をさらに限定できます。例えば、特定のクエリ引数または cookie を持つリクエストのみを検査できます。ルールグループは、スコープダウンステートメントの基準に一致し、ルールグループ設定で指定したアカウント登録パスとアカウント作成パスに送信されたリクエストのみを検査します。スコープダウンステートメントの詳細については、「スコープダウンステートメント」を参照してください。

4. ウェブ ACL に対する変更を保存します。

本番稼働トラフィックに ACFP 実装をデプロイする前に、トラフィックへの潜在的な影響に慣れるまで、ステージング環境またはテスト環境でテストおよびチューニングします。その後、ルールを有効にする前に、本番稼働用トラフィックでカウントモードでルールをテストしてチューニングします。ガイダンスについては、次のセクションを参照してください。

ACFP のテストとデプロイ

このセクションでは、サイトの AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) 実装を設定およびテストするための一般的なガイダンスを提供します。実行する具体的なステップは、ニーズ、リソース、および受け取るウェブリクエストによって異なります。

この情報は、AWS WAF 保護機能のテストと調整 で提供されているテストおよび調整に関する一般情報とは別です。

> **Note**
>
> AWS マネージドルールは、一般的なウェブ脅威から保護するように設計されています。ドキュメントに従って使用すると、 AWS マネージドルールのルールグループはアプリケーションに別のセキュリティレイヤーを追加します。ただし、 AWS マネージドルールのルールグループは、選択した AWS リソースによって決定されるセキュリティ責任に代わるもの

ではありません。の責任共有モデルを参照して、 のリソースが適切に保護 AWS されていることを確認してください。

⚠ 本番稼働トラフィックのリスク

本番稼働トラフィックに ACFP 実装をデプロイする前に、トラフィックへの潜在的な影響に慣れるまで、ステージング環境またはテスト環境でテストおよびチューニングします。その後、ルールを有効にする前に、本番稼働用トラフィックでカウントモードでルールをテストしてチューニングします。

AWS WAF は、ACFP 設定の検証に使用できるテスト認証情報を提供します。次の手順では、ACFP マネージドルールグループを使用するようにテストウェブ ACL を設定し、ルールグループによって追加されたラベルをキャプチャするルールを設定してから、これらのテスト認証情報を使用してアカウント作成の試みを実行します。Amazon CloudWatch メトリクスをチェックしてアカウント作成の試行をウェブ ACL が適切に管理していることを確認します。

このガイダンスは、 AWS WAF ウェブ ACL、ルール、およびルールグループを作成および管理する方法を一般的に認識しているユーザーを対象としています。これらのトピックは、このガイドの前のセクションでカバーされています。

Fraud Control Account Creation AWS WAF Fraud Prevention (ACFP) の実装を設定してテストするには

これらのステップを最初にテスト環境で実行し、次に本番環境で実行します。

1. AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) マネージドルールグループをカウントモードに追加する

   Note

   このマネージドルールグループを使用する場合、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

   AWS マネージドルールのルールグループ`AWSManagedRulesACFPRuleSet`を新規または既存のウェブ ACL に追加し、現在のウェブ ACL の動作を変更しないように設定します。このルー

ルグループのルールとラベルの詳細については、「AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) ルールグループ」を参照してください。

- マネージドルールグループを追加する際には、それを編集し、次の手順を実行します。
  - [ルールグループを設定] ペインで、アプリケーションのアカウント登録ページと作成ページの詳細を入力します。ACFP ルールグループは、この情報を使用してサインインアクティビティをモニタリングします。詳細については、「ACFP マネージドルールグループをウェブ ACL に追加」を参照してください。
  - [Rules] (ルール) ペインで、[Override all rule actions] (すべてのルールアクションをオーバーライド) ドロップダウンを開いて、[Count] を選択します。この設定では、 AWS WAF は、ルールグループ内のすべてのルールに対してリクエストを評価し、その結果の一致のみをカウントしつつ、引き続きリクエストにラベルを追加します。詳細については、「ルールグループ内のルールアクションのオーバーライド」を参照してください。

    このオーバーライドにより、ACFP マネージドルールの影響をモニタリングして、例外 (内部のユースケースの例外など) を追加するかどうか判断できます。
- ウェブ ACL の既存のルールの後に評価されるように、ルールグループを配置します。優先順位の設定の数値は、既に使用しているルールまたはルールグループよりも高くなります。詳細については、「ウェブ ACL でのルールおよびルールグループの処理順序」を参照してください。

  これにより、現在のトラフィックの処理が中断されることはありません。例えば、SQL インジェクションやクロスサイトスクリプティングなどの悪意のあるトラフィックを検出するルールがある場合、そのルールは引き続き検出し、それをログに記録します。または、既知の悪意のないトラフィックを許可するルールがある場合、ACFP マネージドルールグループによってブロックされるようにすることなく、そのトラフィックを許可し続けることができます。テストおよびチューニングのアクティビティ中に、処理順序を調整することもできます。

2. アプリケーション統合 SDK を実装する

AWS WAF JavaScript SDK をブラウザのアカウント登録パスとアカウント作成パスに統合します。 は、iOS デバイスと Android デバイスを統合するモバイル SDKs AWS WAF も提供します。統合 SDK の詳細については、「AWS WAF クライアントアプリケーション統合」を参照してください。このレコメンデーションについては、「ACFP でアプリケーション統合 SDK を使用する理由」を参照してください。

> **Note**
>
> アプリケーション統合 SDK を使用できない場合は、ウェブ ACL で ACFP ルールグループを編集し、`AllRequests` ルールに設定したオーバーライドを削除することで、その ACFP ルールグループをテストできます。これにより、ルールの Challenge アクションの設定が有効になり、有効なチャレンジトークンが確実にリクエストに含まれるようになります。
> これは最初にテスト環境で実行し、その後に本番環境で細心の注意を払って実行してください。このアプローチは、ユーザーをブロックする可能性があります。例えば、登録ページのパスが `GET` テキスト/HTML リクエストを受け入れない場合、このルール設定は、登録ページですべてのリクエストを効果的にブロックできます。

3. ウェブ ACL のログ記録とメトリクスを有効にする

   必要に応じて、ウェブ ACL のログ記録、Amazon Security Lake データ収集、リクエストサンプリング、および Amazon CloudWatch メトリクスを設定します。これらの可視化ツールを使用して ACFP マネージドルールグループとトラフィックとのインタラクションをモニタリングできます。

   - ログ作成の詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。
   - Amazon Security Lake の詳細については、「Amazon Security Lake ユーザーガイド」の「Amazon Security Lake とは」および「 AWS のサービスからのデータ収集」を参照してください。
   - Amazon CloudWatch メトリクスの詳細については、「」を参照してくださいAmazon によるモニタリング CloudWatch。
   - ウェブリクエストサンプリングの詳細については、「ウェブリクエストのサンプルの表示」を参照してください。

4. ウェブ ACL をリソースに関連付ける

   ウェブ ACL がテストリソースに関連付けられていない場合は、関連付けます。詳細については、「ウェブ ACL とリソースの関連付けまたは関連付け解除 AWS」を参照してください。

5. トラフィックと ACFP ルールの一致をモニタリングする

   通常のトラフィックがフローしていることと、ACFP マネージドルールグループのルールが一致するウェブリクエストにラベルを追加していることを確認します。ログにラベルが表示さ

れ、Amazon メトリクスに ACFP とラベルの CloudWatch メトリクスが表示されます。ログでは、ルールグループでカウントするようにオーバーライドしたルールが、カウントに設定された `action` と、オーバーライドした設定済のルールアクションを示す `overriddenAction` とともに、`ruleGroupList` に表示されます。

6. ルールグループの認証情報チェック機能をテストする

   テスト用の侵害された認証情報を使用してアカウント作成を試行し、ルールグループが想定どおりに照合することを確認します。

   a. 保護されたリソースのアカウント登録ページにアクセスし、新しいアカウントの追加を試みます。次の AWS WAF テスト認証情報ペアを使用して、任意のテストを入力します。

      - ユーザー: `WAF_TEST_CREDENTIAL@wafexample.com`
      - パスワード: `WAF_TEST_CREDENTIAL_PASSWORD`

      これらのテスト認証情報は侵害された認証情報として分類され、ACFP マネージドルールグループはアカウント作成リクエストに `awswaf:managed:aws:acfp:signal:credential_compromised` ラベル (ログでの確認が可能) を追加します。

   b. ウェブ ACL ログで、テストアカウント作成リクエストのログエントリの `labels` フィールドで `awswaf:managed:aws:acfp:signal:credential_compromised` ラベルを探します。ログ作成の詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。

   侵害された認証情報をルールグループが想定どおりにキャプチャすることを検証したら、保護されたリソースに必要な実装を設定するステップを実行できます。

7. CloudFront ディストリビューションの場合、ルールグループの一括アカウント作成試行の管理をテストします。

   ACFP ルールグループに設定したレスポンスの成功基準それぞれに対してこのテストを実行します。テストとテストの間は 30 分以上あけてください。

   a. 成功基準ごとに、レスポンス内のその成功基準で成功するアカウント作成の試みを特定します。その後、単一のクライアントセッションから、30 分未満で少なくとも 5 回のアカウント作成の正常な試みを実行します。通常、ユーザーはサイトでアカウントを 1 つだけ作成します。

最初のアカウント作成が成功した後、VolumetricSessionSuccessfulResponse ルールは他のアカウント作成レスポンスとの照合を開始し、ルールアクションのオーバーライドに基づいてそれらにラベル付けをしてカウントします。レイテンシーにより、ルールで最初の 1 回または 2 回が見逃される可能性があります。

b. ウェブ ACL ログで、テストアカウント作成ウェブリクエストのログエントリの labels フィールドで awswaf:managed:aws:acfp:aggregate:volumetric:session:successful_creation_i ラベルを探します。ログ作成の詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。

これらのテストは、ルールによって集計された成功数がルールのしきい値を超えていることを確認することで、成功基準がレスポンスと一致していることを検証します。しきい値に達した後も同じセッションからアカウント作成リクエストを送信し続けると、成功率がしきい値を下回るまでルールによる一致が継続されます。しきい値を超えている間、ルールはセッションアドレスからの正常なアカウント作成の試みと失敗したアカウント作成の試みの両方に一致させます。

8. ACFP ウェブリクエストの処理をカスタマイズする

必要に応じて、リクエストを明示的に許可またはブロックする独自のルールを追加して、ACFP ルールがそのリクエストを処理する方法を変更します。

例えば、ACFP ラベルを使用して、リクエストを許可またはブロックしたり、リクエスト処理をカスタマイズしたりできます。ACFP マネージドルールグループの後にラベル一致ルールを追加して、適用する処理のためにラベル付きリクエストをフィルタリングできます。テスト後、関連する ACFP ルールをカウントモードで維持し、カスタムルールでリクエストの処理に関する決定を維持します。例については、ACFP の例: 侵害された認証情報についてのカスタムレスポンスを参照してください。

9. テストルールを削除し、ACFP マネージドルールグループ設定を有効にする

状況によっては、一部の ACFP ルールをカウントモードのままにすると判断していた可能性もあります。ルールグループ内で設定したとおりに実行するルールについては、ウェブ ACL ルールグループ設定でカウントモードを無効にします。テストが終了したら、テストラベル一致ルールを削除することもできます。

10. モニタリングおよびチューニング

    ウェブリクエストが希望どおりに処理されていることを確認するには、使用することを希望する ACFP 機能を有効にした後、トラフィックを注意深くモニタリングします。ルールグループに対するルールカウントの上書きと独自のルールを使用して、必要に応じて動作を調整します。

ACFP ルールグループの実装のテストが完了したら、ブラウザのアカウント登録ページとアカウント作成ページに AWS WAF JavaScript SDK をまだ統合していない場合は、統合することを強くお勧めします。 には、iOS デバイスと Android デバイスを統合するモバイル SDKs AWS WAF も用意されています。統合 SDK の詳細については、「AWS WAF クライアントアプリケーション統合」を参照してください。このレコメンデーションについては、「ACFP でアプリケーション統合 SDK を使用する理由」を参照してください。

AWS WAF 不正防止アカウント作成詐欺防止 (ACFP) の例

このセクションでは、 AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) の実装の一般的なユースケースに対応できる設定例を示します。

各例は、ユースケースの説明を提供し、カスタム設定ルールの JSON リストにそのソリューションを示します。

> **Note**
>
> これらの例に示されているような JSON リストは、コンソールウェブ ACL JSON ダウンロードやルール JSON エディタを介して、または API やコマンドラインインターフェイスでの `getWebACL` オペレーションを介して取得できます。

トピック

- ACFP の例: シンプルな設定
- ACFP の例: 侵害された認証情報についてのカスタムレスポンス
- ACFP の例:レスポンスインスペクションの設定

ACFP の例: シンプルな設定

次の JSON リストは、 AWS WAF 不正防止アカウント作成詐欺防止 (ACFP) マネージドルールグループを含むウェブ ACL の例を示しています。検証するために、`CreationPath` およ

び RegistrationPagePath の追加設定と、ペイロードタイプ、およびペイロード内の新しいアカウント情報を見つけるために必要な情報に注意してください。ルールグループはこの情報を使用して、アカウント作成リクエストをモニタリングおよび管理します。この JSON には、ラベル名前空間やウェブ ACL のアプリケーション統合 URL など、ウェブ ACL の自動生成された設定が含まれます。

```
{
  "Name": "simpleACFP",
  "Id": "... ",
  "ARN": "arn:aws:wafv2:us-east-1:111122223333:regional/webacl/simpleACFP/... ",
  "DefaultAction": {
    "Allow": {}
  },
  "Description": "",
  "Rules": [
    {
      "Name": "AWS-AWSManagedRulesACFPRuleSet",
      "Priority": 0,
      "Statement": {
        "ManagedRuleGroupStatement": {
          "VendorName": "AWS",
          "Name": "AWSManagedRulesACFPRuleSet",
          "ManagedRuleGroupConfigs": [
            {
              "AWSManagedRulesACFPRuleSet": {
                "CreationPath": "/web/signup/submit-registration",
                "RegistrationPagePath": "/web/signup/registration",
                "RequestInspection": {
                  "PayloadType": "JSON",
                  "UsernameField": {
                    "Identifier": "/form/username"
                  },
                  "PasswordField": {
                    "Identifier": "/form/password"
                  },
                  "EmailField": {
                    "Identifier": "/form/email"
                  },
                  "PhoneNumberFields": [
                    {
                      "Identifier": "/form/country-code"
                    },
                    {
```

```
                    "Identifier": "/form/region-code"
                  },
                  {
                    "Identifier": "/form/phonenumber"
                  }
                ],
                "AddressFields": [
                  {
                    "Identifier": "/form/name"
                  },
                  {
                    "Identifier": "/form/street-address"
                  },
                  {
                    "Identifier": "/form/city"
                  },
                  {
                    "Identifier": "/form/state"
                  },
                  {
                    "Identifier": "/form/zipcode"
                  }
                ]
              },
              "EnableRegexInPath": false
            }
          }
        ]
      }
    },
    "OverrideAction": {
      "None": {}
    },
    "VisibilityConfig": {
      "SampledRequestsEnabled": true,
      "CloudWatchMetricsEnabled": true,
      "MetricName": "AWS-AWSManagedRulesACFPRuleSet"
    }
  }
],
"VisibilityConfig": {
  "SampledRequestsEnabled": true,
  "CloudWatchMetricsEnabled": true,
  "MetricName": "simpleACFP"
```

```
  },
  "Capacity": 50,
  "ManagedByFirewallManager": false,
  "LabelNamespace": "awswaf:111122223333:webacl:simpleACFP:"
}
```

ACFP の例: 侵害された認証情報についてのカスタムレスポンス

デフォルトでは、ルールグループ AWSManagedRulesACFPRuleSet によって実行される認証情報チェックは、リクエストにラベル付けしてブロックすることで、侵害された認証情報を処理します。ルールグループとルールの動作の詳細については、「AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) ルールグループ」を参照してください。

ユーザーが提供したアカウント認証情報が侵害されたことをユーザーに通知するために、次を実行できます。

- **SignalCredentialCompromised** ルールを Count にオーバーライド — これにより、ルールは一致するリクエストをカウントしてラベル付けのみします。
- カスタム処理でラベル一致ルールの追加 – ACFP ラベルと照合し、カスタム処理を実行するようにルールを設定します。

次のウェブ ACL リスティングは、SignalCredentialCompromised ルールアクションがカウントするようにオーバーライドされた状態で、前の例の ACFP マネージドルールグループを示しています。この設定では、このルールグループは、侵害された認証情報を使用するウェブリクエストを評価すると、リクエストにラベルを付けますが、ブロックすることはありません。

さらに、ウェブ ACL には aws-waf-credential-compromised という名前のカスタムレスポンスと AccountSignupCompromisedCredentialsHandling という名前の新しいルールが追加されました。ルールの優先順位にはルールグループよりも大きい数値が設定されているため、ウェブ ACL 評価では、ルールはルールグループの後に実行されます。新しいルールは、ルールグループの侵害された認証情報ラベルを持つすべてのリクエストと照合します。ルールは一致を見つけると、カスタムレスポンス本文を含むリクエストに Block アクションを適用します。カスタムレスポンス本文は、認証情報が侵害されたという情報をエンドユーザーに提供し、実行するアクションを提案します。

```
{
  "Name": "compromisedCreds",
  "Id": "... ",
  "ARN": "arn:aws:wafv2:us-east-1:111122223333:regional/webacl/compromisedCreds/...",
```

```
  "DefaultAction": {
    "Allow": {}
  },
  "Description": "",
  "Rules": [
    {
      "Name": "AWS-AWSManagedRulesACFPRuleSet",
      "Priority": 0,
      "Statement": {
        "ManagedRuleGroupStatement": {
          "VendorName": "AWS",
          "Name": "AWSManagedRulesACFPRuleSet",
          "ManagedRuleGroupConfigs": [
            {
              "AWSManagedRulesACFPRuleSet": {
                "CreationPath": "/web/signup/submit-registration",
                "RegistrationPagePath": "/web/signup/registration",
                "RequestInspection": {
                  "PayloadType": "JSON",
                  "UsernameField": {
                    "Identifier": "/form/username"
                  },
                  "PasswordField": {
                    "Identifier": "/form/password"
                  },
                  "EmailField": {
                    "Identifier": "/form/email"
                  },
                  "PhoneNumberFields": [
                    {
                      "Identifier": "/form/country-code"
                    },
                    {
                      "Identifier": "/form/region-code"
                    },
                    {
                      "Identifier": "/form/phonenumber"
                    }
                  ],
                  "AddressFields": [
                    {
                      "Identifier": "/form/name"
                    },
                    {
```

```
                    "Identifier": "/form/street-address"
                  },
                  {
                    "Identifier": "/form/city"
                  },
                  {
                    "Identifier": "/form/state"
                  },
                  {
                    "Identifier": "/form/zipcode"
                  }
                ]
              },
              "EnableRegexInPath": false
            }
          }
        ],
        "RuleActionOverrides": [
          {
            "Name": "SignalCredentialCompromised",
            "ActionToUse": {
              "Count": {}
            }
          }
        ]
      }
    },
    "OverrideAction": {
      "None": {}
    },
    "VisibilityConfig": {
      "SampledRequestsEnabled": true,
      "CloudWatchMetricsEnabled": true,
      "MetricName": "AWS-AWSManagedRulesACFPRuleSet"
    }
  },
  {
    "Name": "AccountSignupCompromisedCredentialsHandling",
    "Priority": 1,
    "Statement": {
      "LabelMatchStatement": {
        "Scope": "LABEL",
        "Key": "awswaf:managed:aws:acfp:signal:credential_compromised"
      }
```

```
      },
      "Action": {
        "Block": {
          "CustomResponse": {
            "ResponseCode": 406,
            "CustomResponseBodyKey": "aws-waf-credential-compromised",
            "ResponseHeaders": [
              {
                "Name": "aws-waf-credential-compromised",
                "Value": "true"
              }
            ]
          }
        }
      },
      "VisibilityConfig": {
        "SampledRequestsEnabled": true,
        "CloudWatchMetricsEnabled": true,
        "MetricName": "AccountSignupCompromisedCredentialsHandling"
      }
    }
  ],
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "compromisedCreds"
  },
  "Capacity": 51,
  "ManagedByFirewallManager": false,
  "LabelNamespace": "awswaf:111122223333:webacl:compromisedCreds:",
  "CustomResponseBodies": {
    "aws-waf-credential-compromised": {
      "ContentType": "APPLICATION_JSON",
      "Content": "{\n  \"credentials-compromised\": \"The credentials you provided have
 been found in a compromised credentials database.\\n\\nTry again with a different
 username, password pair.\"\n}"
    }
  }
}
```

ACFP の例:レスポンスインスペクションの設定

次の JSON リストは、送信元からの応答を検査するように設定された AWS WAF Fraud Control アカウント作成詐欺防止 (ACFP) マネージドルールグループを含むウェブ ACL の例を示しています。成功と応答のステータスコードを指定する応答検査設定に注意してください。ヘッダー、本文、本文の JSON の一致に基づいて成功と応答の設定を構成することもできます。この JSON には、ラベル名前空間やウェブ ACL のアプリケーション統合 URL など、ウェブ ACL の自動生成された設定が含まれます。

> **Note**
>
> ATP 応答検査は、 CloudFront ディストリビューションを保護する Web ACL でのみ使用できます。

```
{
  "Name": "simpleACFP",
  "Id": "... ",
  "ARN": "arn:aws:wafv2:us-east-1:111122223333:regional/webacl/simpleACFP/... ",
  "DefaultAction": {
    "Allow": {}
  },
  "Description": "",
  "Rules": [
    {
      "Name": "AWS-AWSManagedRulesACFPRuleSet",
      "Priority": 0,
      "Statement": {
        "ManagedRuleGroupStatement": {
          "VendorName": "AWS",
          "Name": "AWSManagedRulesACFPRuleSet",
          "ManagedRuleGroupConfigs": [
            {
              "AWSManagedRulesACFPRuleSet": {
                "CreationPath": "/web/signup/submit-registration",
                "RegistrationPagePath": "/web/signup/registration",
                "RequestInspection": {
                  "PayloadType": "JSON",
                  "UsernameField": {
                    "Identifier": "/form/username"
                  },
                  "PasswordField": {
```

```
        "Identifier": "/form/password"
      },
      "EmailField": {
        "Identifier": "/form/email"
      },
      "PhoneNumberFields": [
        {
          "Identifier": "/form/country-code"
        },
        {
          "Identifier": "/form/region-code"
        },
        {
          "Identifier": "/form/phonenumber"
        }
      ],
      "AddressFields": [
        {
          "Identifier": "/form/name"
        },
        {
          "Identifier": "/form/street-address"
        },
        {
          "Identifier": "/form/city"
        },
        {
          "Identifier": "/form/state"
        },
        {
          "Identifier": "/form/zipcode"
        }
      ]
    },
    "ResponseInspection": {
      "StatusCode": {
        "SuccessCodes": [
          200
        ],
        "FailureCodes": [
          401
        ]
      }
    },
```

```
                "EnableRegexInPath": false
              }
            }
          ]
        }
      },
      "OverrideAction": {
        "None": {}
      },
      "VisibilityConfig": {
        "SampledRequestsEnabled": true,
        "CloudWatchMetricsEnabled": true,
        "MetricName": "AWS-AWSManagedRulesACFPRuleSet"
      }
    }
  ],
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "simpleACFP"
  },
  "Capacity": 50,
  "ManagedByFirewallManager": false,
  "LabelNamespace": "awswaf:111122223333:webacl:simpleACFP:"
  }
```

AWS WAF 不正防止アカウント乗っ取り防止 (ATP)

アカウント乗っ取りは、攻撃者が個人のアカウントへの不正アクセスを得るオンラインの違法行為です。攻撃者は、盗まれた認証情報を使用したり、一連の試行を通じて被害者のパスワードを推測するなど、さまざまな方法でこれを行う可能性があります。攻撃者がアクセスできるようになると、被害者から金銭や情報を盗んだり、サービスを不正に利用したりする可能性があります。攻撃者は、被害者が所有する他のアカウントにアクセスしたり、他の人や組織のアカウントにアクセスしたりするために、被害者としてふるまう可能性があります。さらに、被害者であるユーザーが自分のアカウントからブロックされるようにするために、そのユーザーのパスワードを変更しようとする場合があります。

AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) 機能を実装することで、アカウント乗っ取りの試みを監視して制御できます。 AWS WAF この機能は AWS Managed Rules AWSManagedRulesATPRuleSet ルールグループとコンパニオンアプリケーション統合 SDK で提供されています。

ATP マネージドルールグループは、悪意のあるアカウント乗っ取りの試みの一部である可能性があるリクエストにラベルを付けて管理します。ルールグループは、クライアントでアプリケーションのログインエンドポイントに送信するログイン試行を検査することでこれを行います。

- リクエスト検査 – ATP を使用すると、異常なログイン試行や盗まれた認証情報を使用するログイン試行を可視化して制御できるため、不正行為につながる可能性のあるアカウントの乗っ取りを防ぐことができます。ATP は、盗まれた認証情報のデータベースに照らして E メールとパスワードの組み合わせをチェックします。このデータベースは、漏洩された認証情報がダークウェブ上で新しく見つかると定期的に更新されます。ATP は、IP アドレスやクライアントセッションごとにデータを集約し、不審なリクエストを大量に送信するクライアントを検出してブロックします。
- レスポンス検査 — CloudFront ディストリビューションでは、受信したログインリクエストを検査するだけでなく、ATP ルールグループはログイン試行に対するアプリケーションの応答を検査し、成功率と失敗率を追跡します。この情報を使用して、ATP はログイン失敗の回数が過度に多いクライアントセッションまたは IP アドレスを一時的にブロックできます。 AWS WAF は、レスポンス検査を非同期で実行するため、ウェブトラフィックのレイテンシーが大きくなることはありません。

> **Note**
>
> このマネージドルールグループを使用する場合、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

> **Note**
>
> ATP 機能は、Amazon Cognito ユーザープールでは使用できません。

トピック

- AWS WAF ATP コンポーネント
- ATP でアプリケーション統合 SDK を使用する理由
- ATP マネージドルールグループをウェブ ACL に追加
- ATP のテストとデプロイ
- AWS WAF 不正防止アカウント乗っ取り防止 (ATP) の例

AWS WAF ATP コンポーネント

AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) の主なコンポーネントは次のとおりです。

- **`AWSManagedRulesATPRuleSet`** – この AWS マネージドルールルールグループのルールは、さまざまなタイプのアカウント乗っ取りアクティビティを検出、ラベル付け、処理します。ルールグループは、クライアントから指定したログインエンドポイントに送信される HTTP POST ウェブリクエストを検査します。保護された CloudFront ディストリビューションの場合、ルールグループはディストリビューションがこれらのリクエストに送信するレスポンスも検査します。ルールグループのルールのリストについては、「AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループ」を参照してください。マネージドルールグループ参照ステートメントを使用して、このルールグループをウェブ ACL に含めます。このルールグループの使用については、「ATP マネージドルールグループをウェブ ACL に追加」を参照してください。

  > **Note**
  >
  > このマネージドルールグループを使用する場合、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

- アプリケーションのログインページに関する詳細 – `AWSManagedRulesATPRuleSet` ルールグループをウェブ ACL に追加する際、ログインページに関する情報を提供する必要があります。これにより、ルールグループは検査するリクエストの範囲を絞り込み、ウェブリクエストで認証情報の使用状況を適切に検証できます。ATP ルールグループは、電子メール形式のユーザー名に対応します。詳細については、「ATP マネージドルールグループをウェブ ACL に追加」を参照してください。
- 保護された CloudFront ディストリビューションの場合、アプリケーションがログイン試行にどのように応答するかに関する詳細 – ログイン試行に対するアプリケーションの応答に関する詳細を指定すると、ルールグループは失敗したログイン試行の送信回数が多すぎるクライアントを追跡および管理します。このオプションの設定については、「ATP マネージドルールグループをウェブ ACL に追加」を参照してください。
- JavaScript およびモバイルアプリケーション統合 SDKs – ATP 実装で AWS WAF JavaScript および モバイル SDKs を実装し、ルールグループが提供する機能の完全なセットを有効にします。ATP ルールの多くは、セッションレベルのクライアント検証および動作集約に SDK から提供された情報を使用し、正規のクライアントトラフィックをボットトラフィックから分離するために必要です。SDK の詳細については、「AWS WAF クライアントアプリケーション統合」を参照してください。

ATP 実装を次と組み合わせて、保護のモニタリング、チューニング、およびカスタマイズに役立てることができます。

- ログ記録とメトリクス – ログ、Amazon Security Lake データ収集、ウェブ ACL の Amazon CloudWatch メトリクスを設定および有効にすることで、トラフィックをモニタリングし、ACFP マネージドルールグループがトラフィックに与える影響を理解できます。がウェブリクエストAWSManagedRulesATPRuleSetに追加するラベルは、データに含まれます。オプションの詳細については、、AWS WAF ウェブ ACL トラフィックのログ記録、Amazon によるモニタリング CloudWatchおよび Amazon Security Lake とはを参照してください。

  ニーズと確認できるトラフィックに応じて、AWSManagedRulesATPRuleSet の実装をカスタマイズできます。例えば、ATP 評価から一部のトラフィックを除外したり、スコープダウンステートメントやラベルマッチングルールなどの AWS WAF 機能を使用して、識別したアカウント乗っ取り試行の処理方法を変更したりできます。
- ラベルとラベル一致ルール – AWSManagedRulesATPRuleSet のどのルールでも、ブロック動作をカウントに切り替えて、ルールによって追加されたラベルと照合することができます。このアプローチを使用し、ATP マネージドルールグループによって識別されるウェブリクエストの処理方法をカスタマイズします。ラベル付けおよびラベル一致ステートメントの使用の詳細については、「ラベル一致ルールステートメント」および「AWS WAF ウェブリクエストの ラベル」を参照してください。
- カスタムリクエストとレスポンス - 許可するリクエストにはカスタムヘッダーを追加し、ブロックするリクエストにはカスタムレスポンスを送信できます。これを行うには、ラベル一致を AWS WAF カスタムリクエストおよび応答機能とペアリングします。リクエストとレスポンスをカスタマイズする方法については、「AWS WAFのカスタマイズされたウェブリクエストとレスポンス」を参照してください。

ATP でアプリケーション統合 SDK を使用する理由

ATP マネージドルールグループには、アプリケーション統合 SDK が生成するチャレンジトークンが必要です。トークンは、ルールグループが提供するすべての保護を有効にします。

ATP ルールグループを最も効果的に使用するためにも、アプリケーション統合 SDK を実装することを強くお勧めします。チャレンジスクリプトが取得するトークンからのメリットを ATP ルールグループが得るには、ATP ルールグループの前にチャレンジスクリプトを実行する必要があります。アプリケーション統合 SDK を使用すると、これが自動的に行われます。SDK を使用できない場合は、代替手段として、ATP ルールグループが検査するすべてのリクエストに対して CAPTCHA または Challenge ルールアクションを実行するようにウェブ ACL を設定することができ

ます。Challenge または CAPTCHA ルールアクションを使用すると、追加料金が発生する場合があります。料金の詳細については、「AWS WAF の料金」を参照してください。

トークンを必要としない ATP ルールグループの機能

ウェブリクエストにトークンが含まれていないときは、ATP マネージドルールグループ以下のタイプのトラフィックをブロックできます。

- 多数のログインリクエストを行う単一の IP アドレス。
- 短時間で多数の失敗したログインリクエストが行われた単一の IP アドレス。
- 同じユーザー名を使用してもパスワードを変更してパスワードトラバーサルでログイン試行。

トークンを必要とする ATP ルールグループの機能

チャレンジトークンで提供される情報により、ルールグループとクライアントアプリケーションセキュリティ全体の機能が拡張されます。

このトークンは、ウェブリクエストごとにクライアント情報を提供します。これにより、ATP ルールグループは、正規のクライアントセッションと動作の悪いクライアントセッションが両方とも単一の IP アドレスから発信された場合でも、前者を後者から分離できます。ルールグループは、トークン内の情報を使用してクライアントセッションリクエストの動作を集約し、微調整した検出および軽減を実現します。

トークンがウェブリクエストで使用可能な場合、ATP ルールグループは次の追加カテゴリのクライアントをセッションレベルで検出してブロックできます。

- SDK が管理するサイレントチャレンジに失敗するクライアントセッション。
- ユーザー名またはパスワードを経由するクライアントセッション。これはクレデンシャルスタッフィングとも呼ばれます。
- 盗まれた認証情報を繰り返し使用してログインするクライアントセッション。
- ログインに長時間かかるクライアントセッション。
- 多数のログインリクエストを行うクライアントセッション。ATP ルールグループは、IP AWS WAF アドレスでクライアントをブロックできるレートベースのルールよりもクライアントを分離しやすくなります。ATP ルールグループでは、より低いしきい値も使用されています。
- 短時間で多数の失敗したログインリクエストが行われたクライアントセッション。この機能は、保護対象の Amazon CloudFront ディストリビューションで使用できます。

ルールグループ機能の情報については、「AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループ」を参照してください。

SDK の詳細については、「AWS WAF クライアントアプリケーション統合」を参照してください。AWS WAF トークンの詳細については、を参照してくださいAWS WAF ウェブリクエストトークン。ルールアクションの情報については、「CAPTCHAChallengeの および AWS WAF」を参照してください。

ATP マネージドルールグループをウェブ ACL に追加

ウェブトラフィックのアカウント乗っ取りアクティビティを認識するように ATP マネージドルールグループを設定するには、アプリケーションにログインリクエストを送信する方法に関する情報をクライアントで指定します。保護されている Amazon CloudFront ディストリビューションでは、アプリケーションがログインリクエストにどのように応答するかについての情報も提供します。この設定は、マネージドルールグループの通常の設定に追加されます。

ルールグループの説明とルールリストについては、「AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループ」を参照してください。

> **Note**
>
> 盗まれた認証情報の ATP データベースには、E メール形式のユーザー名のみが含まれています。

このガイダンスは、 AWS WAF ウェブ ACL、ルール、およびルールグループを作成および管理する方法を一般的に認識しているユーザーを対象としています。これらのトピックは、このガイドの前のセクションでカバーされています。マネージドルールグループをウェブ ACL に追加する方法の基本については、「コンソールを通じたウェブ ACL へのマネージドルールグループの追加」を参照してください。

ベストプラクティスに従う

ATP ルールグループは、「インテリジェントな脅威の軽減のためのベストプラクティス」に記載されているベストプラクティスに従って使用してください。

ウェブ ACL で **AWSManagedRulesATPRuleSet** ルールグループを使用するには

1. AWS `AWSManagedRulesATPRuleSet`マネージドルールグループをウェブ ACL に追加し、保存する前にルールグループの設定を編集します。

> **Note**
>
> このマネージドルールグループを使用する場合、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

2. [ルールグループを設定] ペインで、ATP ルールグループがログインリクエストの検査に使用する情報を入力します。

   a. ログインページのパス仕様に合わせて正規表現による照合を行う場合は、[AWS WAF パスに正規表現を使用] をオンに切り替えます。

      AWS WAF PCRE `libpcre` ライブラリで使用されているパターン構文をサポートしていますが、一部例外があります。ライブラリは、「PCRE - Perl Compatible Regular Expressions」で文書化されています。 AWS WAF サポートについて詳しくは、を参照してくださいでの正規表現パターンマッチング AWS WAF。

   b. [Login path] (ログインパス) で、アプリケーションのログインエンドポイントのパスを指定します。ルールグループは、指定されたログインエンドポイントに対する HTTP `POST` リクエストのみを検査します。

      > **Note**
      >
      > エンドポイントの照会では大文字と小文字が区別されません。正規表現の仕様には、大文字と小文字を区別しない照合を無効にするフラグ (`?-i`) を含めてはいけません。文字列の指定はフォワードスラッシュ「`/`」で始まる必要があります。

      例えば、URL `https://example.com/web/login` では、文字列パスの指定「`/web/login`」を指定できます。指定したパスで始まるログインパスは一致と見なされます。例えば、`/web/login` はログインパス `/web/login`、`/web/login/`、`/web/loginPage`、および `/web/login/thisPage` に一致しますが、ログインパス `/home/web/login` または `/website/login` には一致しません。

   c. [リクエスト検査] で、リクエストのペイロードタイプと、ユーザー名とパスワードが指定されているリクエスト本文内のフィールドの名前を指定して、アプリケーションがログイン試行を受け入れる方法を指定します。これらのフィールド名の指定は、ペイロードタイプによって異なります。

- JSON ペイロードタイプ – JSON Pointer 構文でフィールド名を指定します。JSON ポインター構文については、インターネット技術標準化委員会 (IETF) のドキュメント「JavaScriptオブジェクト表記 (JSON) ポインター」を参照してください。

  例えば、次の JSON ペイロードの例では、ユーザー名フィールドの指定は `/login/username` で、パスワードフィールドの指定は `/login/password` です。

  ```
  {
      "login": {
          "username": "THE_USERNAME",
          "password": "THE_PASSWORD"
      }
  }
  ```

- FORM_ENCODED ペイロードタイプ – HTML 形式の名前を使用します。

  例えば、`username1` と `password1` という名前の入力要素を持つ HTML フォームの場合、ユーザー名フィールドの指定は `username1` で、パスワードフィールドの指定は `password1` です。

d. Amazon CloudFront ディストリビューションを保護する場合は、「レスポンスインスペクション」で、アプリケーションがログイン試行に対する応答で成功または失敗をどのように示すかを指定します。

> **Note**
>
> ATP レスポンスインスペクションは、ディストリビューションを保護するウェブ ACL でのみ使用できます。 CloudFront

ATP で検査するログインレスポンスのコンポーネントを 1 つ指定します。本文および JSON コンポーネントタイプの場合、 AWS WAF はコンポーネントの最初の 65,536 バイト (64 KB) を検査できます。

インターフェイスに示されているように、コンポーネントタイプの検査基準を指定します。コンポーネント内で検査する成功基準と失敗基準の両方を指定する必要があります。

例えば、アプリケーションがログイン試行のステータスを応答のステータスコードで示し、成功の場合は「`200 OK`」、失敗の場合は「`401 Unauthorized`」または「`403 Forbidden`」を使用するとします。レスポンス検査の [コンポーネントタイプ] を [ステー

タスコード] に設定し、[成功] テキストボックスに「200」と入力し、[失敗] テキストボックスの 1 行目に「401」、2 行目に「403」と入力します。

ATP ルールグループは、成功または失敗の検査基準に一致する応答のみをカウントします。ルールグループのルールは、カウントされた応答の失敗率が過度に高いクライアントに適用されます。ルールグループのルールが正確に動作するように、ログイン試行の成功と失敗の両方に関する詳細な情報を必ず入力してください。

ログインレスポンスを検査するルールを確認するには、「AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループ」のルールリストで VolumetricIpFailedLoginResponseHigh と VolumetricSessionFailedLoginResponseHigh を探します。

3. ルールグループに必要な追加設定を指定します。

   マネージドルールグループステートメントにスコープダウンステートメントを追加することで、ルールグループが検査するリクエストの範囲をさらに限定できます。例えば、特定のクエリ引数または cookie を持つリクエストのみを検査できます。ルールグループは、スコープダウンステートメントの基準に一致する、指定したログインエンドポイントへの HTTP POST リクエストのみを検査します。スコープダウンステートメントの詳細については、「スコープダウンステートメント」を参照してください。

4. ウェブ ACL に対する変更を保存します。

本番稼働トラフィックに ATP 実装をデプロイする前に、トラフィックへの潜在的な影響に慣れるまで、ステージング環境またはテスト環境でテストおよびチューニングします。その後、ルールを有効にする前に、本番稼働用トラフィックでカウントモードでルールをテストしてチューニングします。ガイダンスについては、次のセクションを参照してください。

ATP のテストとデプロイ

このセクションでは、サイトの AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) 実装を設定およびテストするための一般的なガイダンスを提供します。実行する具体的なステップは、ニーズ、リソース、および受け取るウェブリクエストによって異なります。

この情報は、AWS WAF 保護機能のテストと調整 で提供されているテストおよび調整に関する一般情報とは別です。

> **Note**
>
> AWS マネージドルールは、一般的なウェブ脅威から保護するように設計されています。ドキュメントに従って使用すると、 AWS マネージドルールのルールグループはアプリケーションに別のセキュリティレイヤーを追加します。ただし、 AWS マネージドルールのルールグループは、選択した AWS リソースによって決定されるセキュリティ責任に代わるものではありません。の責任共有モデルを参照して、 のリソースが適切に保護 AWS されていることを確認してください。

> **本番稼働トラフィックのリスク**
>
> 本番稼働トラフィックに ATP 実装をデプロイする前に、トラフィックへの潜在的な影響に慣れるまで、ステージング環境またはテスト環境でテストおよびチューニングします。その後、ルールを有効にする前に、本番稼働用トラフィックでカウントモードでルールをテストしてチューニングします。

AWS WAF は、ATP 設定の検証に使用できるテスト認証情報を提供します。次の手順では、ATP マネージドルールグループを使用するようにテストウェブ ACL を設定し、ルールグループによって追加されたラベルをキャプチャするルールを設定してから、これらのテスト認証情報を使用してログイン試行を実行します。ログイン試行の Amazon CloudWatch メトリクスをチェックして、ウェブ ACL が試行を適切に管理していることを確認します。

このガイダンスは、 AWS WAF ウェブ ACL、ルール、およびルールグループを作成および管理する方法を一般的に認識しているユーザーを対象としています。これらのトピックは、このガイドの前のセクションでカバーされています。

AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) の実装を設定してテストするには

これらのステップを最初にテスト環境で実行し、次に本番環境で実行します。

1. AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) マネージドルールグループをカウントモードに追加する

> **Note**
>
> このマネージドルールグループを使用する場合、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

AWS マネージドルールのルールグループ`AWSManagedRulesATPRuleSet`を新規または既存のウェブ ACL に追加し、現在のウェブ ACL の動作を変更しないように設定します。このルールグループのルールとラベルの詳細については、「AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループ」を参照してください。

- マネージドルールグループを追加する際には、それを編集し、次の手順を実行します。
  - [Rule group configuration] (ルールグループを設定) ペインで、アプリケーションのログインページの詳細を入力します。ATP ルールグループは、この情報を使用してサインインアクティビティをモニタリングします。詳細については、「ATP マネージドルールグループをウェブ ACL に追加」を参照してください。
  - [Rules] (ルール) ペインで、[Override all rule actions] (すべてのルールアクションをオーバーライド) ドロップダウンを開いて、[Count] を選択します。この設定では、 AWS WAF は、ルールグループ内のすべてのルールに対してリクエストを評価し、その結果の一致のみをカウントしつつ、引き続きリクエストにラベルを追加します。詳細については、「ルールグループ内のルールアクションのオーバーライド」を参照してください。

    このオーバーライドにより、ATP マネージドルールの影響をモニタリングして、例外 (内部のユースケースの例外など) を追加するかどうか判断できます。
- ウェブ ACL の既存のルールの後に評価されるように、ルールグループを配置します。優先順位の設定の数値は、既に使用しているルールまたはルールグループよりも高くなります。詳細については、「ウェブ ACL でのルールおよびルールグループの処理順序」を参照してください。

  これにより、現在のトラフィックの処理が中断されることはありません。例えば、SQL インジェクションやクロスサイトスクリプティングなどの悪意のあるトラフィックを検出するルールがある場合、そのルールは引き続き検出し、それをログに記録します。または、既知の悪意のないトラフィックを許可するルールがある場合、ATP マネージドルールグループによって

ブロックされるようにすることなく、そのトラフィックを許可し続けることができます。テストおよびチューニングのアクティビティ中に、処理順序を調整することもできます。

2. ウェブ ACL のログ記録とメトリクスを有効にする

   必要に応じて、ウェブ ACL のログ記録、Amazon Security Lake データ収集、リクエストサンプリング、および Amazon CloudWatch メトリクスを設定します。これらの可視化ツールを使用して ATP マネージドルールグループとトラフィックとのインタラクションをモニタリングできます。

   - ログ記録の設定と使用については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。
   - Amazon Security Lake の詳細については、「Amazon Security Lake ユーザーガイド」の「Amazon Security Lake とは」および「 AWS のサービスからのデータ収集」を参照してください。
   - Amazon CloudWatch メトリクスの詳細については、「」を参照してくださいAmazon によるモニタリング CloudWatch。
   - ウェブリクエストサンプリングの詳細については、「ウェブリクエストのサンプルの表示」を参照してください。

3. ウェブ ACL をリソースに関連付ける

   ウェブ ACL がテストリソースに関連付けられていない場合は、関連付けます。詳細については、「ウェブ ACL とリソースの関連付けまたは関連付け解除 AWS」を参照してください。

4. トラフィックと ATP ルールの一致をモニタリングする

   通常のトラフィックがフローしていることと、ATP マネージドルールグループのルールが一致するウェブリクエストにラベルを追加していることを確認します。ログにラベルが表示され、Amazon メトリクスに ATP とラベルの CloudWatch メトリクスが表示されます。ログでは、ルールグループでカウントするようにオーバーライドしたルールが、カウントに設定された `action` と、オーバーライドした設定済のルールアクションを示す `overriddenAction` とともに、`ruleGroupList` に表示されます。

5. ルールグループの認証情報チェック機能をテストする

   テスト用の侵害された認証情報を使用してログインを試行し、ルールグループが想定どおりに照合することを確認します。

   a. 次の AWS WAF テスト認証情報ペアを使用して、保護されたリソースのログインページにログインします。

- ユーザー: `WAF_TEST_CREDENTIAL@wafexample.com`
- パスワード: `WAF_TEST_CREDENTIAL_PASSWORD`

これらのテスト認証情報は侵害された認証情報として分類され、ATP マネージドルールグループはログインリクエストに `awswaf:managed:aws:atp:signal:credential_compromised` ラベル (ログでの確認が可能) を追加します。

b. ウェブ ACL ログで、テストログインウェブリクエストのログエントリの `labels` フィールドで `awswaf:managed:aws:atp:signal:credential_compromised` ラベルを探します。ログ作成の詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。

侵害された認証情報をルールグループが想定どおりにキャプチャすることを検証したら、保護されたリソースに必要な実装を設定するステップを実行できます。

6. CloudFront ディストリビューションの場合は、ルールグループのログイン失敗管理をテストします。

a. ATP ルールグループに設定した応答の失敗基準それぞれに対してテストを実行します。テストとテストの間は 10 分以上あけてください。

単一の失敗基準をテストするには、その条件で失敗するログイン試行を応答内で特定します。次に、単一のクライアント IP アドレスからの失敗したログイン試行を、10 分以内に少なくとも 10 回実行します。

最初の 6 回の試行が失敗した後、ボリュームメトリックが失敗したログインルールが、残りのログイン試行に対する一致、ラベル付け、およびカウントを開始します。レイテンシーにより、ルールで最初の 1 回または 2 回が見逃される可能性があります。

b. ウェブ ACL ログで、テストログインウェブリクエストのログエントリの `labels` フィールドで `awswaf:managed:aws:atp:aggregate:volumetric:ip:failed_login_response:high` ラベルを探します。ログ作成の詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。

これらのテストでは、失敗したログイン回数がルール `VolumetricIpFailedLoginResponseHigh` のしきい値を超えているかどうかをチェックして、失敗基準が応答に一致しているかどうかを検証します。しきい値に達した後も同じ IP アドレスからログインリクエストを送信し続けると、失敗率がしきい値を下回るまでルールによる一致が継続されます。しきい値を超えている間、ルールは IP アドレスからの成功したログインと失敗したログインの両方に一致させます。

7. ATP ウェブリクエストの処理をカスタマイズする

   必要に応じて、リクエストを明示的に許可またはブロックする独自のルールを追加して、ATP ルールがそのリクエストを処理する方法を変更します。

   例えば、ATP ラベルを使用して、リクエストを許可またはブロックしたり、リクエスト処理をカスタマイズしたりできます。ATP マネージドルールグループの後にラベル一致ルールを追加して、適用する処理のためにラベル付きリクエストをフィルタリングできます。テスト後、関連する ATP ルールをカウントモードで維持し、カスタムルールでリクエストの処理に関する決定を維持します。例については、「ATP の例: 認証情報の不足および侵害された認証情報のカスタム処理」を参照してください。

8. テストルールを削除し、ATP マネージドルールグループ設定を有効にする

   状況によっては、一部の ATP ルールをカウントモードのままにすると判断していた可能性もあります。ルールグループ内で設定したとおりに実行するルールについては、ウェブ ACL ルールグループ設定でカウントモードを無効にします。テストが終了したら、テストラベル一致ルールを削除することもできます。

9. モニタリングおよびチューニング

   ウェブリクエストが希望どおりに処理されていることを確認するには、使用することを希望する ATP 機能を有効にした後、トラフィックを注意深くモニタリングします。ルールグループに対するルールカウントの上書きと独自のルールを使用して、必要に応じて動作を調整します。

ATP ルールグループの実装のテストが終了した後、まだ完了していない場合は、検出機能を強化するために、 AWS WAF JavaScript SDK をブラウザのログインページに統合することを強くお勧めします。 は、iOS デバイスと Android デバイスを統合するモバイル SDKs AWS WAF も提供します。統合 SDK の詳細については、「AWS WAF クライアントアプリケーション統合」を参照してください。このレコメンデーションについては、「ATP でアプリケーション統合 SDK を使用する理由」を参照してください。

AWS WAF 不正防止アカウント乗っ取り防止 (ATP) の例

このセクションでは、 AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) の実装の一般的なユースケースに対応できる設定例を示します。

各例は、ユースケースの説明を提供し、カスタム設定ルールの JSON リストにそのソリューションを示します。

> **Note**
>
> これらの例に示されているような JSON リストは、コンソールウェブ ACL JSON ダウンロードやルール JSON エディタを介して、または API やコマンドラインインターフェイスでの `getWebACL` オペレーションを介して取得できます。

トピック

ATP の例: シンプルな設定

次の JSON リストは、 AWS WAF 不正防止アカウント乗っ取り防止 (ATP) 管理ルールグループを含むウェブ ACL の例を示しています。追加のサインインページ設定は、ルールグループがログインリクエストをモニタリングおよび管理するために必要な情報を提供することにご注意ください。この JSON には、ラベル名前空間やウェブ ACL のアプリケーション統合 URL など、ウェブ ACL の自動生成された設定が含まれます。

```
{
    "WebACL": {
        "LabelNamespace": "awswaf:111122223333:webacl:ATPModuleACL:",
        "Capacity": 50,
        "Description": "This is a test web ACL for ATP.",
        "Rules": [
            {
                "Priority": 1,
                "OverrideAction": {
                    "None": {}
```

```
                },
                "VisibilityConfig": {
                    "SampledRequestsEnabled": true,
                    "CloudWatchMetricsEnabled": true,
                    "MetricName": "AccountTakeOverValidationRule"
                },
                "Name": "DetectCompromisedUserCredentials",
                "Statement": {
                    "ManagedRuleGroupStatement": {
                        "VendorName": "AWS",
                        "Name": "AWSManagedRulesATPRuleSet",
                        "ManagedRuleGroupConfigs": [
                          {
                            "AWSManagedRulesATPRuleSet": {
                              "LoginPath": "/web/login",
                              "RequestInspection": {
                                "PayloadType": "JSON",
                                "UsernameField": {
                                  "Identifier": "/form/username"
                                },
                                "PasswordField": {
                                  "Identifier": "/form/password"
                                }
                              },
                              "EnableRegexInPath": false
                            }
                          }
                        ]
                    }
                }
            }
        ],
        "VisibilityConfig": {
            "SampledRequestsEnabled": true,
            "CloudWatchMetricsEnabled": true,
            "MetricName": "ATPValidationAcl"
        },
        "DefaultAction": {
            "Allow": {}
        },
        "ManagedByFirewallManager": false,
        "Id": "32q10987-65rs-4tuv-3210-98765wxyz432",
        "ARN": "arn:aws:wafv2:us-east-1:111122223333:regional/webacl/
ATPModuleACL/32q10987-65rs-4tuv-3210-98765wxyz432",
```

```
        "Name": "ATPModuleACL"
    },
    "ApplicationIntegrationURL": "https://9z87abce34ea.us-
east-1.sdk.awswaf.com/9z87abce34ea/1234567a1b10/",
    "LockToken": "6d0e6966-95c9-48b6-b51d-8e82e523b847"
}
```

ATP の例: 認証情報の不足および侵害された認証情報のカスタム処理

デフォルトでは、ルールグループ AWSManagedRulesATPRuleSet によって実行される認証情報チェックは次のようにウェブリクエストを処理します。

- 認証情報の不足 – リクエストにラベルを付けてブロックします。
- [Compromised credentials] (侵害された認証情報) - リクエストにラベルを付けますが、ブロックしたりカウントしたりしません。

ルールグループとルールの動作の詳細については、「AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループ」を参照してください。

次の手順を実行して、認証情報が不足している、または侵害されたウェブリクエストのカスタム処理を追加できます。

- **MissingCredential** ルールを Count にオーバーライド — このルールアクションのオーバーライドにより、ルールは一致するリクエストをカウントしてラベル付けのみします。
- カスタム処理でラベル一致ルールの追加 – 両方の ATP ラベルと照合し、カスタム処理を実行するようにルールを設定します。例えば、顧客をサインアップページにリダイレクトできます。

次のルールは、MissingCredential ルールアクションがカウントするようにオーバーライドされた状態で、前の例の ATP マネージドルールグループを示しています。これにより、ルールはリクエストをブロックするのではなく、一致するリクエストにラベルを適用し、リクエストのみをカウントします。

```
"Rules": [
    {
        "Priority": 1,
        "OverrideAction": {
            "None": {}
        },
        "VisibilityConfig": {
```

```
            "SampledRequestsEnabled": true,
            "CloudWatchMetricsEnabled": true,
            "MetricName": "AccountTakeOverValidationRule"
        },
        "Name": "DetectCompromisedUserCredentials",
        "Statement": {
            "ManagedRuleGroupStatement": {
                "ManagedRuleGroupConfigs": [
                  {
                    "AWSManagedRulesATPRuleSet": {
                      "LoginPath": "/web/login",
                      "RequestInspection": {
                        "PayloadType": "JSON",
                        "UsernameField": {
                          "Identifier": "/form/username"
                        },
                        "PasswordField": {
                          "Identifier": "/form/password"
                        }
                      },
                      "EnableRegexInPath": false
                    }
                  }
                ]
                "VendorName": "AWS",
                "Name": "AWSManagedRulesATPRuleSet",
                "RuleActionOverrides": [
                  {
                    "ActionToUse": {
                      "Count": {}
                    },
                    "Name": "MissingCredential"
                  }
                ],
                "ExcludedRules": []
            }
        }
    }
],
```

この設定では、このルールグループは、認証情報が不足し、または侵害されたウェブリクエストを評価すると、リクエストにラベルを付けますが、ブロックすることはありません。

次のルールは、前のルールグループよりも優先順位が高く設定されています。 AWS WAF は、優先順位の低いルールから順番に評価するため、このルールはルールグループの後に評価されます。認証情報のラベルのどちらかと照合し、一致するリクエストにカスタムレスポンスを送信するようにルールが設定されています。

```
"Name": "redirectToSignup",
      "Priority": 10,
      "Statement": {
        "OrStatement": {
          "Statements": [
            {
              "LabelMatchStatement": {
                "Scope": "LABEL",
                "Key": "awswaf:managed:aws:atp:signal:missing_credential"
              }
            },
            {
              "LabelMatchStatement": {
                "Scope": "LABEL",
                "Key": "awswaf:managed:aws:atp:signal:credential_compromised"
              }
            }
          ]
        }
      },
      "Action": {
        "Block": {
          "CustomResponse": {
             your custom response settings
          }
        }
      },
      "VisibilityConfig": {
        "SampledRequestsEnabled": true,
        "CloudWatchMetricsEnabled": true,
        "MetricName": "redirectToSignup"
      }
```

ATP の例:応答検査設定

次の JSON リストは、 AWS WAF 送信元からの応答を検査するように設定された不正防止アカウント乗っ取り防止 (ATP) 管理ルールグループを含むウェブ ACL の例を示しています。成功と応答の

ステータスコードを指定するレスポンスインスペクションの設定に注意してください。ヘッダー、本文、本文の JSON の一致に基づいて成功と応答の設定を構成することもできます。この JSON には、ラベル名前空間やウェブ ACL のアプリケーション統合 URL など、ウェブ ACL の自動生成された設定が含まれます。

> **Note**
>
> ATP レスポンスインスペクションは、 CloudFront ディストリビューションを保護する Web ACL でのみ使用できます。

```
{
    "WebACL": {
        "LabelNamespace": "awswaf:111122223333:webacl:ATPModuleACL:",
        "Capacity": 50,
        "Description": "This is a test web ACL for ATP.",
        "Rules": [
            {
                "Priority": 1,
                "OverrideAction": {
                    "None": {}
                },
                "VisibilityConfig": {
                    "SampledRequestsEnabled": true,
                    "CloudWatchMetricsEnabled": true,
                    "MetricName": "AccountTakeOverValidationRule"
                },
                "Name": "DetectCompromisedUserCredentials",
                "Statement": {
                    "ManagedRuleGroupStatement": {
                        "VendorName": "AWS",
                        "Name": "AWSManagedRulesATPRuleSet",
                        "ManagedRuleGroupConfigs": [
                          {
                            "AWSManagedRulesATPRuleSet": {
                              "LoginPath": "/web/login",
                              "RequestInspection": {
                                "PayloadType": "JSON",
                                "UsernameField": {
                                  "Identifier": "/form/username"
                                },
```

```
                                "PasswordField": {
                                  "Identifier": "/form/password"
                                }
                              },
                              "ResponseInspection": {
                                "StatusCode": {
                                  "SuccessCodes": [
                                    200
                                  ],
                                  "FailureCodes": [
                                    401
                                  ]
                                }
                              },
                              "EnableRegexInPath": false
                            }
                          }
                        ]
                    }
                }
            }
        ],
        "VisibilityConfig": {
            "SampledRequestsEnabled": true,
            "CloudWatchMetricsEnabled": true,
            "MetricName": "ATPValidationAcl"
        },
        "DefaultAction": {
            "Allow": {}
        },
        "ManagedByFirewallManager": false,
        "Id": "32q10987-65rs-4tuv-3210-98765wxyz432",
        "ARN": "arn:aws:wafv2:us-east-1:111122223333:regional/webacl/
ATPModuleACL/32q10987-65rs-4tuv-3210-98765wxyz432",
        "Name": "ATPModuleACL"
    },
    "ApplicationIntegrationURL": "https://9z87abce34ea.us-
east-1.sdk.awswaf.com/9z87abce34ea/1234567a1b10/",
    "LockToken": "6d0e6966-95c9-48b6-b51d-8e82e523b847"
}
```

AWS WAF ボットコントロール

Bot Control を使用すると、スクレーパー、スキャナ、クローラ、ステータスモニター、検索エンジンなどのボットを簡単にモニタリング、ブロック、レート制限の適用ができます。ルールグループの対象検査レベルを使用する場合、自己識別しないボットにチャレンジを仕掛けることができるため、悪意のあるボットがウェブサイトを狙うことが難しくなり、ボットの運用コストも高くなります。アプリケーションを保護するには、Bot Control マネージドルールグループを単独で使用することも、AWS AWS WAF 他のマネージドルールルールグループや独自のカスタムルールと組み合わせて使用することもできます。

Bot Control には、リクエストサンプリングに基づいて、ボットからの現在のトラフィックの量を示すコンソールダッシュボードが含まれています。Bot Control マネージドルールグループをウェブ ACL に追加すると、ボットトラフィックに対してアクションを実行したり、アプリケーションへの一般的なボットトラフィックに関する詳細なリアルタイム情報を受け取ったりすることができます。

> **Note**
>
> このマネージドルールグループを使用する場合、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

Bot Control マネージドルールグループには、自己識別ボットにラベルを追加、一般的に望ましいボットを検証、信頼度の高いボットシグネチャを検出する基本かつ共通の保護レベルが用意されています。これにより、ボットトラフィックの共通カテゴリをモニタリングおよび制御できます。

Bot Control ルールグループには、自己識別を行わない高度なボットに対する検出機能が追加された、ターゲットを絞った保護レベルも用意されています。ターゲットを絞った保護では、ブラウザ調査、フィンガープリント、行動ヒューリスティックなどの検出技術を使用して不正なボットトラフィックを識別します。さらに、ターゲットを絞った保護では、ウェブサイトのトラフィック統計を機械学習で自動的に分析して、ボット関連のアクティビティを検出することもできます。機械学習を有効にすると、 AWS WAF はタイムスタンプ、ブラウザの特性、以前にアクセスした URL など、ウェブサイトのトラフィックに関する統計を使用して Bot Control の機械学習モデルを改善します。

Bot Control マネージドルールグループの詳細については、「AWS WAF Bot Control ルールグループ」を参照してください。

Web リクエストを Bot Control AWS WAF マネージドルグループと照合して評価すると、ルールグループはボットに関連していると検出されたリクエストに、ボットのカテゴリやボット名などのラベ

ルを追加します。 AWS WAF 独自のルールでこれらのラベルと照合して、処理をカスタマイズできます。Bot Control マネージドルールグループによって生成されるラベルは、Amazon CloudWatch メトリックスとウェブ ACL ログに含まれます。

また、 AWS Firewall Manager AWS WAF ポリシーを使用して、組織に属する複数のアカウントのアプリケーションに Bot Control マネージドルールグループをデプロイすることもできます AWS Organizations。

AWS WAF Bot Control コンポーネント

Bot Control の実装の主なコンポーネントは次のとおりです。

- **`AWSManagedRulesBotControlRuleSet`** – さまざまなカテゴリのボットを検出して処理するルールを持つ Bot Control マネージドルールグループ。このルールグループは、ボットトラフィックとして検出されたウェブリクエストにラベルを追加します。

  > **Note**
  >
  > このマネージドルールグループを使用する場合、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

  Bot Control マネージドルールグループには、次の 2 レベルの保護から選択できます。

  - 共通 – ウェブスクレイピングフレームワーク、検索エンジン、自動ブラウザなど、さまざまな自己識別ボットを検出します。このレベルの Bot Control 保護は、静的リクエストデータ分析など、従来のボット検出技術を使用して一般的なボットを識別します。ルールはこれらのボットからのトラフィックにラベルを付け、検証できないものはブロックします。
  - ターゲットを絞った – 一般的な保護機能に加え、自己識別を行わない高度なボットに対するターゲットを絞った検出機能も追加されています。ターゲットを絞った保護は、レート制限と CAPTCHA およびバックグラウンドブラウザのチャレンジの組み合わせを使用してボットアクティビティを軽減します。
    - **`TGT_`** – ターゲットを絞った保護を提供するルールには、`TGT_` で始まる名前が付いています。すべてのターゲットを絞った保護では、ブラウザ調査、フィンガープリント、行動ヒューリスティックなどの検出技術を使用して不正なボットトラフィックを識別します。
    - **`TGT_ML_`** – 機械学習を使用するターゲットを絞った保護のルールには、`TGT_ML_` で始まる名前が付いています。これらのルールでは、ウェブサイトトラフィック統計の自動機械学習分析を使用して、分散された調整されたボットアクティビティを示す異常な動作を検出します。 は、タイムスタンプ、ブラウザの特性、以前にアクセスした URL などのウェブサイトト

ラフィックに関する統計 AWS WAF を分析し、Bot Control 機械学習モデルを改善します。機械学習機能はデフォルトで有効になっていますが、ルールグループ設定で無効にすることができます。機械学習が無効になっている場合、 AWS WAF はこれらのルールを評価しません。

ルールグループルールに関する情報を含む詳細については、「AWS WAF Bot Control ルールグループ」を参照してください。

マネージドルールグループ参照ステートメントを使用して、このルールグループをウェブ ACL に含め、使用する検査レベルを指定します。ターゲットレベルでは、機械学習を有効にするかどうかも指定します。このマネージドルールグループをウェブ ACL に追加する方法については、「AWS WAF ボットコントロールマネージドルールグループをウェブ ACL に追加する」を参照してください。

- [Bot Control dashboard] (Bot Control ダッシュボード) – ウェブ ACL のボットモニタリングダッシュボード。ウェブ ACL Bot Control のタブから利用できます。トラフィックをモニタリングし、さまざまなタイプのボットからのトラフィックの量を理解するために、このダッシュボードを使用します。これは、このトピックで説明するように、ボット管理をカスタマイズするための開始点とすることができます。また、これを使用して、変更を検証し、さまざまなボットやボットカテゴリのアクティビティをモニタリングすることもできます。
- JavaScript およびモバイルアプリケーション統合 SDKs – Bot Control ルールグループのターゲットを絞った保護レベルを使用する場合は、 AWS WAF JavaScript および モバイル SDKs を実装する必要があります。ターゲットルールは、クライアントトークン内で SDK から提供された情報を使用し、悪意のあるボットに対する検出を強化します。SDK の詳細については、「AWS WAF クライアントアプリケーション統合」を参照してください。
- ログ記録とメトリクス – AWS WAF ログ、Amazon Security Lake、Amazon でウェブ ACL 用に収集されたデータを調査することで、ボットトラフィックをモニタリングし、Bot Control マネージドルールグループがトラフィックをどのように評価および処理するかを理解できます CloudWatch。Bot Control がウェブリクエストに追加するラベルは、データに含まれます。これらのオプションの詳細については、AWS WAF ウェブ ACL トラフィックの口グ記録「」、Amazon によるモニタリング CloudWatch「」、「Amazon Security Lake とは」を参照してください。

ニーズと確認できるトラフィックに応じて、Bot Control の実装をカスタマイズできます。最も一般的に使用されるオプションの一部は次のとおりです。

- スコープダウンステートメント – Bot Control マネージドルールグループの参照ステートメント内にスコープダウンステートメントを追加することにより、Bot Control マネージドルールグループが評価するウェブリクエストからの一部トラフィックを除外できます。スコープダウンステートメントは、ネスト可能なルールステートメントとすることができます。リクエストがスコープダウンステートメントと一致しない場合、 は、ルールグループに対して AWS WAF 評価せずに、ルール

グループ参照ステートメントと一致していないと評価します。スコープダウンステートメントの詳細については、「スコープダウンステートメント」を参照してください。

Bot Control マネージドルールグループの料金は、 AWS WAF がルールグループを使用して評価するウェブリクエスト数に応じて上がります。スコープダウンステートメントを使用してルールグループが評価するリクエストを制限することで、これらのコストを削減できます。たとえば、ボットを含むすべてのユーザーにホームページのロードを許可し、その後にアプリケーション API に送信されるリクエスト、あるいは特定のタイプのコンテンツを含むリクエストにルールグループのルールを適用できます。

- ラベルとラベルマッチングルール – Bot Control ルールグループが AWS WAF ラベル一致ルールステートメントを使用して識別したボットトラフィックの一部を処理する方法をカスタマイズできます。Bot Control ルールグループは、ウェブリクエストにラベルを追加します。Bot Control ラベルと一致する Bot Control ルールグループの後にラベル一致ルールを追加し、必要な処理を適用できます。ラベル付けおよびラベル一致ステートメントの使用の詳細については、「ラベル一致ルールステートメント」および「AWS WAF ウェブリクエストの ラベル」を参照してください。
- カスタムリクエストとレスポンス – 許可したリクエストにカスタムヘッダーを追加し、ラベルマッチングをカスタムリクエストとレスポンスの機能と組み合わせることで、ブロックしたリクエストに対して AWS WAF カスタムレスポンスを送信できます。リクエストとレスポンスをカスタマイズする方法については、「AWS WAFのカスタマイズされたウェブリクエストとレスポンス」を参照してください。

Bot Control でアプリケーション統合 SDK を使用する理由

Bot Control マネージドルールグループのターゲット保護のほとんどには、アプリケーション統合 SDK が生成するチャレンジトークンが必要です。リクエストにチャレンジトークンを必要としないルールは、Bot Control の共通レベルの保護とターゲットレベルの機械学習ルールです。ルールグループの保護レベルとルールの説明については、「AWS WAF Bot Control ルールグループ」を参照してください。

Bot Control ルールグループを最も効果的に使用するためにも、アプリケーション統合 SDK を実装することを強くお勧めします。チャレンジスクリプトが取得するトークンからのメリットを Bot Control ルールグループが得るには、Bot Control ルールグループの前にチャレンジスクリプトを実行する必要があります。

- アプリケーション統合 SDK では、スクリプトは自動的に実行されます。
- SDK を使用できない場合は、Bot Control ルールグループが検査するすべてのリクエストに対して Challenge または CAPTCHA ルールアクションを実行するようにウェブ ACL を設定することがで

きます。Challenge または CAPTCHA ルールアクションを使用すると、追加料金が発生する場合があります。料金の詳細については、「AWS WAF の料金」を参照してください。

アプリケーション統合 SDK をクライアントに実装、またはチャレンジスクリプトを実行するルールアクションの 1 つを使用するときは、ルールグループと、クライアントアプリケーションセキュリティ全体の機能が拡張されます。

トークンは、各ウェブリクエストでクライアント情報を提供します。この追加情報により、Bot Control のルールグループは、正規のクライアントセッションと動作の悪いクライアントセッションが両方とも単一の IP アドレスから発信された場合でも、前者を後者から分離できます。ルールグループは、トークン内の情報を使用してクライアントセッションリクエストの動作を集約し、ターゲットを絞った保護レベルが提供する微調整した検出および軽減を実現します。

SDK の詳細については、「AWS WAF クライアントアプリケーション統合」を参照してください。AWS WAF トークンの詳細については、を参照してくださいAWS WAF ウェブリクエストトークン。ルールアクションの情報については、「CAPTCHAChallengeの および AWS WAF」を参照してください。

AWS WAF ボットコントロールマネージドルールグループをウェブ ACL に追加する

Bot Control マネージドルールグループ `AWSManagedRulesBotControlRuleSet` は、実装する保護レベルを特定するための追加設定が必要です。

ルールグループの説明とルールリストについては、「AWS WAF Bot Control ルールグループ」を参照してください。

このガイダンスは、 AWS WAF ウェブ ACL、ルール、およびルールグループを作成および管理する方法を一般的に認識しているユーザーを対象としています。これらのトピックは、このガイドの前のセクションでカバーされています。マネージドルールグループをウェブ ACL に追加する方法の基本については、「コンソールを通じたウェブ ACL へのマネージドルールグループの追加」を参照してください。

ベストプラクティスに従う

Bot Control ルールグループは、「インテリジェントな脅威の軽減のためのベストプラクティス」に記載されているベストプラクティスに従って使用してください。

ウェブ ACL で **AWSManagedRulesBotControlRuleSet** ルールグループを使用するには

1. AWS AWSManagedRulesBotControlRuleSetマネージドルールグループをウェブ ACL に追加します。ルールグループの詳細な説明については、「the section called “Bot Control ルールグループ”」を参照してください。

   > **Note**
   >
   > このマネージドルールグループを使用する場合、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

   ルールグループを追加する際は、ルールグループの設定ページを開くように編集します。

2. ルールグループの設定ページの [Inspection level] (検査レベル) ペインで、使用する検査レベルを選択します。

   - 共通 – ウェブスクレイピングフレームワーク、検索エンジン、自動ブラウザなど、さまざまな自己識別ボットを検出します。このレベルの Bot Control 保護は、静的リクエストデータ分析など、従来のボット検出技術を使用して一般的なボットを識別します。ルールはこれらのボットからのトラフィックにラベルを付け、検証できないものはブロックします。
   - ターゲットを絞った – 一般的な保護機能に加え、自己識別を行わない高度なボットに対するターゲットを絞った検出機能も追加されています。ターゲットを絞った保護は、レート制限と CAPTCHA およびバックグラウンドブラウザのチャレンジの組み合わせを使用してボットアクティビティを軽減します。
     - **TGT_** – ターゲットを絞った保護を提供するルールには、TGT_ で始まる名前が付いています。すべてのターゲットを絞った保護では、ブラウザ調査、フィンガープリント、行動ヒューリスティックなどの検出技術を使用して不正なボットトラフィックを識別します。
     - **TGT_ML_** – 機械学習を使用するターゲットを絞った保護のルールには、TGT_ML_ で始まる名前が付いています。これらのルールは、ウェブサイトのトラフィック統計を機械学習で自動分析し、分散的かつ協調的なボットのアクティビティを示す異常な動作を検出します。AWS WAF タイムスタンプ、ブラウザーの特性、以前にアクセスした URL など、Web サイトのトラフィックに関する統計を分析して、Bot Control の機械学習モデルを改善します。機械学習機能はデフォルトで有効になっていますが、ルールグループ設定で無効にすることができます。機械学習が無効になっている場合、 AWS WAF これらのルールは評価されません。

3. ターゲットを絞った保護レベルを使用していて、 AWS WAF ウェブトラフィックの分析に機械学習 (ML) を使用して分散型ボットアクティビティを検出したくない場合は、機械学習オプションを無効にしてください。名前が TGT_ML_ で始まる Bot Control ルールでは機械学習が必要になります。これらのルールの詳細については、「Bot Control のルールリスト」を参照してください。
4. ルールグループの使用コストを抑えるスコープダウンステートメントを追加します。スコープダウンステートメントは、ルールグループが検査する一連のリクエストを絞り込みます。ユースケースの例として、ボットコントロールの例:ログインページにのみボットコントロールを使用する および ボットコントロールの例:ボットコントロールは動的コンテンツにのみ使用してください で始めてください。
5. ルールグループに必要な追加設定を指定します。
6. ウェブ ACL に対する変更を保存します。

本番稼働トラフィックに Bot Control 実装をデプロイする前に、トラフィックへの潜在的な影響に慣れるまで、ステージング環境またはテスト環境でテストおよびチューニングします。その後、ルールを有効にする前に、本番稼働用トラフィックでカウントモードでルールをテストしてチューニングします。ガイダンスについては、次のセクションを参照してください。

AWS WAF ボットコントロールによる誤検知

AWS WAF Bot Control が管理するルールグループのルールは、誤検出を最小限に抑えるために慎重に選択されています。グローバルトラフィックに対してルールをテストし、テストウェブ ACL に対する影響をモニタリングします。ただし、トラフィックパターンの変化が原因で誤検出が引き続き検出されることがあります。さらに、一部のユースケースは誤検出を引き起こすことが知られており、ウェブトラフィックに特化したカスタマイズが必要になります。

誤検出が発生する可能性のある状況には、次のような例が含まれます。

- 通常、モバイルアプリにはブラウザ以外のユーザーエージェントがあり、`SignalNonBrowserUserAgent` ルールではデフォルトでブロックされます。モバイルアプリからのトラフィックや、ブラウザ以外のユーザーエージェントによるその他の正当なトラフィックが予想される場合は、例外を追加して許可する必要があります。
- アップタイムのモニタリング、統合テスト、マーケティングツールなど、特定のボットトラフィックに依拠する場合があります。許可するボットトラフィックを Bot Control が識別してブロックする場合は、独自のルールを追加して処理を変更する必要があります。これは必ずしもすべてのお客様の誤検出のシナリオではありませんが、誤検出のシナリオに該当する場合、誤検出の場合と同じ処理が必要になります。

- Bot Control マネージドルールグループは、からの IP アドレスを使用してボットを検証します。AWS WAF Bot Control を使用し、プロキシまたはロードバランサーを介してルーティングするボットを検証した場合は、カスタムルールを使用して明示的に許可する必要がある場合があります。このタイプのカスタムルールを作成する方法については、「転送された IP アドレス」を参照してください。
- グローバルに誤検出率が低い Bot Control ルールが特定のデバイスまたはアプリケーションに大きな影響を与える可能性があります。例えば、テストや検証では、トラフィック量の少ないアプリケーションや、あまり一般的でないブラウザまたはデバイスからのリクエストが観察されなかった可能性があります。
- 以前から誤検出率が低い Bot Control ルールで有効なトラフィックの誤検出が増加している可能性があります。これは、新しいトラフィックパターンまたは有効なトラフィックを伴って出現するリクエスト属性が原因となっている可能性があります。これにより、以前は存在していなかったルールと一致するようになる可能性があります。これらの変更は、次のような状況によって発生する可能性があります。
  - ロードバランサーやコンテンツ配信ネットワーク (CDN) など、ネットワークアプライアンスを通じてトラフィックがフローする際に変更されたトラフィックの詳細。
  - トラフィックデータの新たな変化 (新しいブラウザや既存のブラウザの新しいバージョンなど)。

AWS WAF Bot Control マネージドルールグループから発生する可能性のある誤検出を処理する方法については、後のセクション「AWS WAF Bot Control のテストとデプロイ」のガイダンスを参照してください。

AWS WAF Bot Control のテストとデプロイ

このセクションでは、サイトの AWS WAF Bot Control 実装を設定およびテストするための一般的なガイダンスを提供します。実行する具体的なステップは、ニーズ、リソース、受け取るウェブリクエストによって異なります。

この情報は、AWS WAF 保護機能のテストと調整 で提供されているテストおよび調整に関する一般情報とは別です。

> **Note**
>
> AWS マネージドルールは、一般的なウェブ脅威から保護するように設計されています。ドキュメントに従って使用すると、 AWS マネージドルールのルールグループはアプリケーションに別のセキュリティレイヤーを追加します。ただし、 AWS マネージドルールのルールグループは、選択した AWS リソースによって決定されるセキュリティ責任に代わるもの

ではありません。の責任共有モデルを参照して、 のリソースが適切に保護 AWS されていることを確認してください。

⚠ 本番稼働トラフィックのリスク

本番稼働トラフィックに Bot Control 実装をデプロイする前に、トラフィックへの潜在的な影響に慣れるまで、ステージング環境またはテスト環境でテストおよびチューニングします。その後、ルールを有効にする前に、本番稼働用トラフィックでカウントモードでルールをテストしてチューニングします。

このガイダンスは、 AWS WAF ウェブ ACL、ルール、およびルールグループを作成および管理する方法を一般的に認識しているユーザーを対象としています。これらのトピックは、このガイドの前のセクションでカバーされています。

Bot Control の実装を設定およびテストするには

これらのステップを最初にテスト環境で実行し、次に本番環境で実行します。

1. Bot Control マネージドルールグループを追加する

   Note

   このマネージドルールグループを使用する場合、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

   マネージド AWS ルールグループ`AWSManagedRulesBotControlRuleSet`を新規または既存のウェブ ACL に追加し、現在のウェブ ACL の動作を変更しないように設定します。

   - マネージドルールグループを追加する際には、それを編集し、次の手順を実行します。
     - [Inspection level] (検査レベル) ペインで、使用する検査レベルを選択します。
       - 共通 – ウェブスクレイピングフレームワーク、検索エンジン、自動ブラウザなど、さまざまな自己識別ボットを検出します。このレベルの Bot Control 保護は、静的リクエストデータ分析など、従来のボット検出技術を使用して一般的なボットを識別します。ルールはこれらのボットからのトラフィックにラベルを付け、検証できないものはブロックします。

- ターゲットを絞った – 一般的な保護機能に加え、自己識別を行わない高度なボットに対するターゲットを絞った検出機能も追加されています。ターゲットを絞った保護は、レート制限と CAPTCHA およびバックグラウンドブラウザのチャレンジの組み合わせを使用してボットアクティビティを軽減します。
  - **TGT_** – ターゲットを絞った保護を提供するルールには、TGT_ で始まる名前が付いています。すべてのターゲットを絞った保護では、ブラウザ調査、フィンガープリント、行動ヒューリスティックなどの検出技術を使用して不正なボットトラフィックを識別します。
  - **TGT_ML_** – 機械学習を使用するターゲットを絞った保護のルールには、TGT_ML_ で始まる名前が付いています。これらのルールは、ウェブサイトトラフィック統計の自動機械学習分析を使用して、分散された調整されたボットアクティビティを示す異常な動作を検出します。 は、タイムスタンプ、ブラウザの特性、以前にアクセスした URL などのウェブサイトトラフィックに関する統計 AWS WAF を分析し、Bot Control 機械学習モデルを改善します。機械学習機能はデフォルトで有効になっていますが、ルールグループ設定で無効にすることができます。機械学習が無効になっている場合、 AWS WAF はこれらのルールを評価しません。

  この選択の詳細については、「AWS WAF Bot Control ルールグループ」を参照してください。

- [Rules] (ルール) ペインで、[Override all rule actions] (すべてのルールアクションをオーバーライド) ドロップダウンを開いて、[Count] を選択します。この設定では、 はルールグループ内のすべてのルールに対するリクエスト AWS WAF を評価し、その結果の一致のみをカウントし、リクエストにラベルを追加します。詳細については、「ルールグループ内のルールアクションのオーバーライド」を参照してください。

  このオーバーライドにより、Bot Control ルールがトラフィックに与える潜在的な影響をモニタリングでき、内部のユースケースや目的のボットなどの例外を追加するかどうか判断します。

- ウェブ ACL で最後に評価されるように、ルールグループを配置します。優先順位の設定の数値は、既に使用している他のルールまたはルールグループよりも高くなります。詳細については、「ウェブ ACL でのルールおよびルールグループの処理順序」を参照してください。

これにより、現在のトラフィックの処理が中断されることはありません。たとえば、SQL インジェクションやクロスサイトスクリプティングなどの悪意のあるトラフィックを検出するルールがある場合、そのようなリクエストを継続的に検出してログ記録します。または、既知の悪意のないトラフィックを許可するルールがある場合、Bot Control マネージドルールグ

ループによってブロックされるようにすることなく、そのトラフィックを許可し続けることができます。テストおよび調整アクティビティ中に処理順序を調整することができますが、開始する方法としてお勧めします。

2. ウェブ ACL のログ記録とメトリクスを有効にする

   必要に応じて、ウェブ ACL のログ記録、Amazon Security Lake データ収集、リクエストサンプリング、および Amazon CloudWatch メトリクスを設定します。これらの可視性ツールを使用して、Bot Control マネージドルールグループとトラフィックの相互作用をモニタリングできます。

   - ログ作成の詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。
   - Amazon Security Lake の詳細については、「Amazon Security Lake ユーザーガイド」の「Amazon Security Lake とは」および「 AWS のサービスからのデータ収集」を参照してください。
   - Amazon CloudWatch メトリクスの詳細については、「」を参照してくださいAmazon によるモニタリング CloudWatch。
   - ウェブリクエストサンプリングの詳細については、「ウェブリクエストのサンプルの表示」を参照してください。

3. ウェブ ACL をリソースに関連付ける

   ウェブ ACL がリソースに関連付けられていない場合は、関連付けます。詳細については、「ウェブ ACL とリソースの関連付けまたは関連付け解除 AWS」を参照してください。

4. トラフィックと Bot Control ルールの一致をモニタリングする

   トラフィックがフローしていることと、Bot Control マネージドルールグループのルールが一致するウェブリクエストにラベルを追加していることを確認します。ログにラベルが表示され、Amazon メトリクスにボットとラベルの CloudWatch メトリクスが表示されます。ログでは、ルールグループでカウントするようにオーバーライドしたルールが、カウントに設定された `action` と、オーバーライドした設定済のルールアクションを示す `overriddenAction` とともに、`ruleGroupList` に表示されます。

   > **Note**
   >
   > Bot Control マネージドルールグループは、 AWS WAFからの IP アドレスを使用してボットを検証します。Bot Control を使用し、プロキシまたはロードバランサーを介してルーティングするボットを検証した場合は、カスタムルールを使用して明示的に許可す

る必要がある場合があります。カスタムルールの作成方法については、「転送された IP アドレス」を参照してください。ルールを使用して Bot Control ウェブリクエストの処理をカスタマイズする方法については、次のステップを参照してください。

ウェブリクエスト処理を詳細にレビューして、カスタム処理で軽減する必要のある誤検出があるかどうかを確認してください。誤検知の例については、「AWS WAF ボットコントロールによる誤検知」を参照してください。

5. Bot Control ウェブリクエストの処理をカスタマイズする

   必要に応じて、リクエストを明示的に許可またはブロックする独自のルールを追加して、Bot Control ルールがそのリクエストを処理する方法を変更します。

   これをどのように実行するかはユースケースによって異なりますが、一般的な解決策は次のとおりです。

   - Bot Control マネージドルールグループの前に追加したルールを含むリクエストを明示的に許可します。これにより、許可されたリクエストが評価のためにルールグループに到達することはありません。これは、Bot Control マネージドルールグループの使用コストを抑えるのに役立ちます。
   - Bot Control マネージドルールグループのステートメント内のスコープダウンステートメントを追加し、Bot Control 評価からのリクエストを除外します。これは、前述のオプションと同じように機能します。スコープダウンステートメントと一致しないリクエストがルールグループの評価に到達することはないため、Bot Control マネージドルールグループの使用コストを抑えるのに役立ちます。スコープダウンステートメントの詳細については、「スコープダウンステートメント」を参照してください。

     例については、以下を参照してください。

     - ボット管理から IP 範囲を除外する
     - 制御するボットからのトラフィックを許可する
   - リクエスト処理に Bot Control ラベルを使用して、リクエストを許可またはブロックします。Bot Control マネージドルールグループの後にラベル一致ルールを追加して、ブロックするリクエストから、許可するラベル付きリクエストをブロックするリクエストを除外します。

     テスト後、関連する Bot Control ルールをカウントモードで維持し、カスタムルールでリクエストの処理に関する決定を維持します。ラベル一致ステートメントの詳細については、「ラベル一致ルールステートメント」を参照してください。

この種類のカスタマイズの例については、以下を参照してください。

- ブロックされたユーザーエージェントの例外を作成する
- 特定のブロックされたボットを許可する
- 検証済みボットをブロックする

その他の例については、「AWS WAF ボットコントロールの例」を参照してください。

6. 必要に応じて、Bot Control マネージドルールグループ設定を有効にします

   状況によっては、一部の Bot Control ルールをカウントモードの状態で維持する、あるいは異なるアクションのオーバーライドに適用すると判断する場合があります。ルールグループ内で設定されているときに実行するルールについては、通常のルール設定を有効にします。これを行うには、ウェブ ACL のルールグループステートメントを編集し、[Rules] (ルール) ペインで変更を行います。

AWS WAF ボットコントロールの例

このセクションでは、 AWS WAF Bot Control 実装のさまざまな一般的なユースケースを満たす設定例を紹介します。

各例は、ユースケースの説明を提供し、カスタム設定ルールの JSON リストにそのソリューションを示します。

> **Note**
>
> これらの例に示されている JSON リストは、ルールを設定し、Rule JSON エディタを使用して編集することにより、コンソールで作成されました。

トピック

- ボットコントロールの例:簡単な設定
- ボットコントロールの例:検証済みのボットを明示的に許可する
- ボットコントロールの例:検証済みボットをブロック
- ボットコントロールの例:ブロックされた特定のボットを許可する
- ボットコントロールの例:ブロックされたユーザーエージェントの例外を作成する

- ボットコントロールの例:ログインページにのみボットコントロールを使用する
- ボットコントロールの例:ボットコントロールは動的コンテンツにのみ使用してください
- ボット制御の例:IP 範囲をボット管理から除外
- ボットコントロールの例:自分がコントロールするボットからのトラフィックを許可する
- ボットコントロールの例:目標とする検査レベル
- ボットコントロールの例:2 つのステートメントを使用して対象とする検査レベルの使用を制限する

ボットコントロールの例:簡単な設定

次の JSON リストは、 AWS WAF ボットコントロールマネージドルールグループを含むウェブ ACL の例を示しています。可視性の設定に注意してください。これにより AWS WAF 、監視目的でリクエストサンプルとメトリクスが保存されます。

```
{
  "Name": "Bot-WebACL",
  "Id": "...",
  "ARN": "...",
  "DefaultAction": {
    "Allow": {}
  },
  "Description": "Bot-WebACL",
  "Rules": [
      {
        ...
      },
      {
         "Name": "AWS-AWSBotControl-Example",
         "Priority": 5,
         "Statement": {
            "ManagedRuleGroupStatement": {
               "VendorName": "AWS",
               "Name": "AWSManagedRulesBotControlRuleSet",
               "ManagedRuleGroupConfigs": [
                 {
                   "AWSManagedRulesBotControlRuleSet": {
                     "InspectionLevel": "COMMON"
                   }
                 }
               ],
```

```
                "RuleActionOverrides": [],
                "ExcludedRules": []
             },
             "VisibilityConfig": {
                "SampledRequestsEnabled": true,
                "CloudWatchMetricsEnabled": true,
                "MetricName": "AWS-AWSBotControl-Example"
              }
          }
       }
    ],
    "VisibilityConfig": {
       ...
    },
    "Capacity": 1496,
    "ManagedByFirewallManager": false
}
```

ボットコントロールの例:検証済みのボットを明示的に許可する

AWS WAF ボットコントロールは、一般的で検証可能なボットであることがわかっているボットをブロックしません。 AWS Bot Control が検証済みボットからのウェブリクエストを識別すると、ボットに名前を付けるラベルと、検証済みボットであることを示すラベルが追加されます。Bot Control は、既知の正常なボットがブロックされないように、シグナルラベルなどの他のラベルを追加しません。

AWS WAF 認証済みボットをブロックするルールは他にもあるかもしれません。検証済みのボットが確実に許可されるようにするには、Bot Control ラベルに基づいてそれらのボットを許可するカスタムルールを追加します。ラベルを照合できるように、新しいルールは Bot Control マネージドルールグループの後に実行される必要があります。

次のルールは、検証済みボットを明示的に許可します。

```
{
    "Name": "match_rule",
    "Statement": {
       "LabelMatchStatement": {
          "Scope": "LABEL",
          "Key": "awswaf:managed:aws:bot-control:bot:verified"
       }
    },
    "RuleLabels": [],
```

```
    "Action": {
      "Allow": {}
    }
}
```

ボットコントロールの例:検証済みボットをブロック

検証済みのボットをブロックするには、 AWS WAF Bot Control マネージドルールグループの後に実行されるボットをブロックするルールを追加する必要があります。これを行うには、ブロックするボット名を特定し、ラベル一致ステートメントを使用して、それらを識別してブロックします。検証済みのすべてのボットをブロックするだけの場合は、bot:name: ラベルとの照合を省略できます。

次のルールは、bingbot 検証済みボットのみをブロックします。このルールは、Bot Control マネージドルールグループの後に実行する必要があります。

```
{
    "Name": "match_rule",
    "Statement": {
      "AndStatement": {
        "Statements": [
          {
            "LabelMatchStatement": {
              "Scope": "LABEL",
              "Key": "awswaf:managed:aws:bot-control:bot:name:bingbot"
            }
          },
          {
            "LabelMatchStatement": {
              "Scope": "LABEL",
              "Key": "awswaf:managed:aws:bot-control:bot:verified"
            }
          }
        ]
      }
    },
    "RuleLabels": [],
    "Action": {
      "Block": {}
    }
  }
```

次のルールは、すべての検証済みボットをブロックします。

```
{
    "Name": "match_rule",
    "Statement": {
      "LabelMatchStatement": {
        "Scope": "LABEL",
        "Key": "awswaf:managed:aws:bot-control:bot:verified"
      }
    },
    "RuleLabels": [],
    "Action": {
      "Block": {}
    }
}
```

ボットコントロールの例:ブロックされた特定のボットを許可する

複数の Bot Control ルールによってボットがブロックされる可能性があります。各ブロッキングルールについて、次の手順を実行します。

AWS WAF ブロックしたくないボットをボットコントロールルールがブロックしている場合は、次の操作を行います。

1. ログをチェックして、ボットをブロックしている Bot Control ルールを特定します。ブロックルールは、名前が `terminatingRule` で始まるフィールドのログで指定されます。ウェブ ACL ログの詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。ルールがリクエストに追加するラベルに注意してください。
2. ウェブ ACL で、ブロッキングルールのアクションをカウントするようにオーバーライドします。コンソールでこれを行うには、ウェブ ACL でルールグループのルールを編集し、ルールに Count のルールアクションオーバーライドを選択します。これにより、ボットがルールによってブロックされないようにしても、ルールは一致するリクエストにラベルを適用します。
3. Bot Control マネージドルールグループの後に、ウェブ ACL にラベル一致ルールを追加します。オーバーライドされたルールのラベルと照合し、ブロックしたくないボットを除くすべての一致するリクエストをブロックするようにルールを設定します。

   これで、ウェブ ACL が設定され、許可するボットが、ログを通じて特定したブロックルールによってブロックされなくなります。

トラフィックとログをもう一度チェックして、ボットの通過が許可されていることを確認します。許可されていない場合は、上記の手順を再度実行してください。

例えば、pingdom を除くすべてのモニタリングボットをブロックするとします。この場合、CategoryMonitoring ルールがカウントするようにオーバーライドし、その後にボット名ラベル pingdom が付いているものを除くすべてのモニタリングボットをブロックするルールを記述します。

次のルールは、Bot Control マネージドルールグループを使用しますが、CategoryMonitoring のルールアクションがカウントするようにオーバーライドします。カテゴリモニタリングルールは、一致するリクエストに通常どおりラベルを適用しますが、通常のブロックアクションを実行するのではなく、カウントするだけです。

```
{
  "Name": "AWS-AWSBotControl-Example",
  "Priority": 5,
  "Statement": {
    "ManagedRuleGroupStatement": {
      "VendorName": "AWS",
      "Name": "AWSManagedRulesBotControlRuleSet",
      "ManagedRuleGroupConfigs": [
        {
          "AWSManagedRulesBotControlRuleSet": {
            "InspectionLevel": "COMMON"
          }
        }
      ],
   "RuleActionOverrides": [
        {
          "ActionToUse": {
            "Count": {}
          },
          "Name": "CategoryMonitoring"
        }
      ],
      "ExcludedRules": []
    }
  },
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "AWS-AWSBotControl-Example"
  }
}
```

次のルールは、前の CategoryMonitoring ルールが一致するウェブリクエストに追加するカテゴリモニタリングラベルと照合します。カテゴリモニタリングリクエストの中で、このルールはボット名 pingdom のラベルを持つものを除くすべてをブロックします。

次のルールは、ウェブ ACL の処理順序で、前の Bot Control マネージドルールグループの後に実行する必要があります。

```
{
      "Name": "match_rule",
      "Priority": 10,
      "Statement": {
        "AndStatement": {
          "Statements": [
            {
              "LabelMatchStatement": {
                "Scope": "LABEL",
                "Key": "awswaf:managed:aws:bot-control:bot:category:monitoring"
              }
            },
            {
              "NotStatement": {
                "Statement": {
                  "LabelMatchStatement": {
                    "Scope": "LABEL",
                    "Key": "awswaf:managed:aws:bot-control:bot:name:pingdom"
                  }
                }
              }
            }
          ]
        }
      },
      "Action": {
        "Block": {}
      },
      "VisibilityConfig": {
        "SampledRequestsEnabled": true,
        "CloudWatchMetricsEnabled": true,
        "MetricName": "match_rule"
      }
}
```

ボットコントロールの例:ブロックされたユーザーエージェントの例外を作成する

ブラウザ以外のユーザーエージェントからのトラフィックが誤ってブロックされている場合、 AWS WAF 問題となっているボットコントロールルールを Count `SignalNonBrowserUserAgent` に設定し、ルールのラベルを例外条件と組み合わせることで例外を作成できます。

> **Note**
>
> 通常、モバイルアプリにはブラウザ以外のユーザーエージェントがあり、`SignalNonBrowserUserAgent` ルールではデフォルトでブロックされます。

次のルールは、Bot Control マネージドルールグループを使用しますが、`SignalNonBrowserUserAgent` のルールアクションがカウントするようにオーバーライドします。シグナルルールは、一致するリクエストに通常どおりラベルを適用しますが、通常のブロックアクションを実行するのではなく、カウントするだけです。

```
{
  "Name": "AWS-AWSBotControl-Example",
  "Priority": 5,
  "Statement": {
    "ManagedRuleGroupStatement": {
      "VendorName": "AWS",
      "Name": "AWSManagedRulesBotControlRuleSet",
      "ManagedRuleGroupConfigs": [
        {
          "AWSManagedRulesBotControlRuleSet": {
            "InspectionLevel": "COMMON"
          }
        }
      ],
   "RuleActionOverrides": [
        {
          "ActionToUse": {
            "Count": {}
          },
          "Name": "SignalNonBrowserUserAgent"
        }
      ],
      "ExcludedRules": []
    }
  },
```

```
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "AWS-AWSBotControl-Example"
  }
}
```

次のルールは、Bot Control `SignalNonBrowserUserAgent` ルールが一致するウェブリクエストに追加したシグナルラベルと照合します。シグナルリクエストの中では、このルールは当社が許可するユーザーエージェントを持つものを除くすべてをブロックします。

次のルールは、ウェブ ACL の処理順序で、前の Bot Control マネージドルールグループの後に実行する必要があります。

```
{
    "Name": "match_rule",
    "Statement": {
      "AndStatement": {
        "Statements": [
          {
            "LabelMatchStatement": {
              "Scope": "LABEL",
              "Key": "awswaf:managed:aws:bot-control:signal:non_browser_user_agent"
            }
          },
          {
            "NotStatement": {
              "Statement": {
                "ByteMatchStatement": {
                  "FieldToMatch": {
                    "SingleHeader": {
                      "Name": "user-agent"
                    }
                  },
                  "PositionalConstraint": "EXACTLY",
                  "SearchString": "PostmanRuntime/7.29.2",
                  "TextTransformations": [
                    {
                      "Priority": 0,
                      "Type": "NONE"
                    }
                  ]
                }
```

```
              }
            }
          }
        ]
      }
    },
    "RuleLabels": [],
    "Action": {
      "Block": {}
    },
    "VisibilityConfig": {
      "SampledRequestsEnabled": true,
      "CloudWatchMetricsEnabled": true,
      "MetricName": "match_rule"
    }
}
```

ボットコントロールの例:ログインページにのみボットコントロールを使用する

次の例では、スコープダウンステートメントを使用して、URI パスで識別される Web AWS WAF サイトのログインページに到達するトラフィックにのみボットコントロールを適用します。loginログインページへの URI パスは、アプリケーションや環境によっては、この例とは異なる場合があります。

```
{
  "Name": "AWS-AWSBotControl-Example",
  "Priority": 5,
  "Statement": {
    "ManagedRuleGroupStatement": {
      "VendorName": "AWS",
      "Name": "AWSManagedRulesBotControlRuleSet",
   "ManagedRuleGroupConfigs": [
        {
          "AWSManagedRulesBotControlRuleSet": {
            "InspectionLevel": "COMMON"
          }
        }
      ],
      "RuleActionOverrides": [],
      "ExcludedRules": []
    },
    "VisibilityConfig": {
      "SampledRequestsEnabled": true,
```

```
      "CloudWatchMetricsEnabled": true,
      "MetricName": "AWS-AWSBotControl-Example"
    },
    "ScopeDownStatement": {
      "ByteMatchStatement": {
        "SearchString": "login",
        "FieldToMatch": {
          "UriPath": {}
        },
        "TextTransformations": [
          {
            "Priority": 0,
            "Type": "NONE"
          }
        ],
        "PositionalConstraint": "CONTAINS"
      }
    }
  }
}
```

ボットコントロールの例:ボットコントロールは動的コンテンツにのみ使用してください

この例では、 AWS WAF スコープダウンステートメントを使用してボットコントロールを動的コンテンツにのみ適用しています。

スコープダウンステートメントは、正規表現パターンセットの一致結果を否定することにより、静的コンテンツを除外します。

- 正規表現パターンセットは、静的コンテンツの拡張子と一致するように設定されています。例えば、正規表現パターンセットの指定は `(?i)\.(jpe?g|gif|png|svg|ico|css|js|woff2?)$` である場合があります。正規表現のパターンセットとステートメントの詳細については、「正規表現パターンセット一致ルールステートメント」を参照してください。
- スコープダウンステートメントでは、NOT ステートメント内に正規表現パターンセットのステートメントをネストすることにより、一致する静的コンテンツを除外します。NOT ステートメントについては、「NOT ルールステートメント」を参照してください。

```
{
  "Name": "AWS-AWSBotControl-Example",
  "Priority": 5,
  "Statement": {
```

```
    "ManagedRuleGroupStatement": {
      "VendorName": "AWS",
      "Name": "AWSManagedRulesBotControlRuleSet",
   "ManagedRuleGroupConfigs": [
        {
          "AWSManagedRulesBotControlRuleSet": {
            "InspectionLevel": "COMMON"
          }
        }
      ],
      "RuleActionOverrides": [],
      "ExcludedRules": []
    },
    "VisibilityConfig": {
      "SampledRequestsEnabled": true,
      "CloudWatchMetricsEnabled": true,
      "MetricName": "AWS-AWSBotControl-Example"
    },
    "ScopeDownStatement": {
      "NotStatement": {
        "Statement": {
          "RegexPatternSetReferenceStatement": {
            "ARN": "arn:aws:wafv2:us-east-1:123456789:regional/regexpatternset/
excludeset/00000000-0000-0000-0000-000000000000",
            "FieldToMatch": {
              "UriPath": {}
            },
            "TextTransformations": [
              {
                "Priority": 0,
                "Type": "NONE"
              }
            ]
          }
        }
      }
    }
  }
}
```

ボット制御の例:IP 範囲をボット管理から除外

AWS WAF ウェブトラフィックのサブセットをボットコントロール管理から除外したい場合で、ルールステートメントを使用してそのサブセットを特定できる場合は、ボットコントロールが管理するルールグループステートメントにスコープダウンステートメントを追加して除外します。

次のルールは、特定の IP アドレス範囲からのウェブリクエストを除き、すべてのウェブトラフィックに対して通常の Bot Control ボット管理を実行します。

```
{
  "Name": "AWS-AWSBotControl-Example",
  "Priority": 5,
  "Statement": {
    "ManagedRuleGroupStatement": {
      "VendorName": "AWS",
      "Name": "AWSManagedRulesBotControlRuleSet",
      "ManagedRuleGroupConfigs": [
        {
          "AWSManagedRulesBotControlRuleSet": {
            "InspectionLevel": "COMMON"
          }
        }
      ],
      "RuleActionOverrides": [],
      "ExcludedRules": []
    },
    "VisibilityConfig": {
      "SampledRequestsEnabled": true,
      "CloudWatchMetricsEnabled": true,
      "MetricName": "AWS-AWSBotControl-Example"
    },
    "ScopeDownStatement": {
      "NotStatement": {
        "Statement": {
          "IPSetReferenceStatement": {
            "ARN": "arn:aws:wafv2:us-east-1:123456789:regional/ipset/
friendlyips/00000000-0000-0000-0000-000000000000"
          }
        }
      }
    }
  }
}
```

ボットコントロールの例:自分がコントロールするボットからのトラフィックを許可する

一部のサイトモニタリングボットとカスタムボットは、カスタムヘッダーを送信するように設定できます。このようなボットからのトラフィックを許可する場合、ヘッダーに共有シークレットを追加するように設定できます。その後、 AWS WAF Bot Control マネージドルールグループステートメントに scope-down ステートメントを追加することで、ヘッダーを含むメッセージを除外できます。

次のルール例は、シークレットヘッダーを持つトラフィックを Bot Control 検査から除外します。

```
{
  "Name": "AWS-AWSBotControl-Example",
  "Priority": 5,
  "Statement": {
    "ManagedRuleGroupStatement": {
      "VendorName": "AWS",
      "Name": "AWSManagedRulesBotControlRuleSet",
      "ManagedRuleGroupConfigs": [
        {
          "AWSManagedRulesBotControlRuleSet": {
            "InspectionLevel": "COMMON"
          }
        }
      ],
      "RuleActionOverrides": [],
      "ExcludedRules": []
    },
    "VisibilityConfig": {
      "SampledRequestsEnabled": true,
      "CloudWatchMetricsEnabled": true,
      "MetricName": "AWS-AWSBotControl-Example"
    },
    "ScopeDownStatement": {
      "NotStatement": {
        "Statement": {
          "ByteMatchStatement": {
            "SearchString": "YSBzZWNyZXQ=",
            "FieldToMatch": {
              "SingleHeader": {
                "Name": "x-bypass-secret"
              }
            },
            "TextTransformations": [
              {
```

```
                "Priority": 0,
                "Type": "NONE"
              }
            ],
            "PositionalConstraint": "EXACTLY"
          }
        }
      }
    }
  }
}
```

ボットコントロールの例:目標とする検査レベル

保護レベルを高めるために、 AWS WAF Bot Control マネージドルールグループでターゲットインスペクションレベルを有効にすることができます。

次の例では、機械学習機能が有効になっています。EnableMachineLearningに設定すると、この動作をオプトアウトできますfalse。

```
{
  "Name": "AWS-AWSBotControl-Example",
  "Priority": 5,
  "Statement": {
    "ManagedRuleGroupStatement": {
      "VendorName": "AWS",
      "Name": "AWSManagedRulesBotControlRuleSet",
      "ManagedRuleGroupConfigs": [
        {
          "AWSManagedRulesBotControlRuleSet": {
            "InspectionLevel": "TARGETED",
            "EnableMachineLearning": true
          }
        }
      ],
      "RuleActionOverrides": [],
      "ExcludedRules": []
    },
    "VisibilityConfig": {
      "SampledRequestsEnabled": true,
      "CloudWatchMetricsEnabled": true,
      "MetricName": "AWS-AWSBotControl-Example"
    }
```

```
  }
}
```

ボットコントロールの例:2 つのステートメントを使用して対象とする検査レベルの使用を制限する

コストを最適化するため、ウェブ ACL では 2 つの AWS WAF Bot Control マネージドルールグループステートメントを使用し、インスペクションレベルとスコープを分けることができます。たとえば、対象とするインスペクションレベルステートメントのスコープを、より機密性の高いアプリケーションエンドポイントのみに限定できます。

次の例の 2 つのステートメントには、相互に排他的なスコープがあります。この構成がないと、1 回のリクエストで 2 回の課金評価が発生する可能性があります。

> **Note**
>
> コンソールのビジュアルエディターでは、AWSManagedRulesBotControlRuleSet複数のステートメントの参照はサポートされていません。代わりに JSON エディターを使用してください。

```
{
  "Name": "Bot-WebACL",
  "Id": "...",
  "ARN": "...",
  "DefaultAction": {
    "Allow": {}
  },
  "Description": "Bot-WebACL",
  "Rules": [
      {
        ...
      },
      {
       "Name": "AWS-AWSBotControl-Common",
       "Priority": 5,
       "Statement": {
          "ManagedRuleGroupStatement": {
             "VendorName": "AWS",
             "Name": "AWSManagedRulesBotControlRuleSet",
             "ManagedRuleGroupConfigs": [
               {
                 "AWSManagedRulesBotControlRuleSet": {
```

```
                "InspectionLevel": "COMMON"
              }
            }
          ],
          "RuleActionOverrides": [],
          "ExcludedRules": []
        },
        "VisibilityConfig": {
          "SampledRequestsEnabled": true,
          "CloudWatchMetricsEnabled": true,
          "MetricName": "AWS-AWSBotControl-Common"
        },
        "ScopeDownStatement": {
          "NotStatement": {
            "Statement": {
              "ByteMatchStatement": {
                "FieldToMatch": {
                  "UriPath": {}
                },
                "PositionalConstraint": "STARTS_WITH",
                "SearchString": "/sensitive-endpoint",
                "TextTransformations": [
                  {
                    "Type": "NONE",
                    "Priority": 0
                  }
                ]
              }
            }
          }
        }
      }
    },
    {
      "Name": "AWS-AWSBotControl-Targeted",
      "Priority": 6,
      "Statement": {
        "ManagedRuleGroupStatement": {
          "VendorName": "AWS",
          "Name": "AWSManagedRulesBotControlRuleSet",
          "ManagedRuleGroupConfigs": [
            {
              "AWSManagedRulesBotControlRuleSet": {
                "InspectionLevel": "TARGETED",
```

```
                    "EnableMachineLearning": true
                  }
                }
              ],
              "RuleActionOverrides": [],
              "ExcludedRules": []
            },
            "VisibilityConfig": {
              "SampledRequestsEnabled": true,
              "CloudWatchMetricsEnabled": true,
              "MetricName": "AWS-AWSBotControl-Targeted"
            },
            "ScopeDownStatement": {
               "Statement": {
                 "ByteMatchStatement": {
                   "FieldToMatch": {
                     "UriPath": {}
                   },
                   "PositionalConstraint": "STARTS_WITH",
                   "SearchString": "/sensitive-endpoint",
                   "TextTransformations": [
                     {
                       "Type": "NONE",
                       "Priority": 0
                     }
                   ]
                 }
               }
             }
          }
        }
    ],
    "VisibilityConfig": {
      ...
    },
    "Capacity": 1496,
    "ManagedByFirewallManager": false
}
```

AWS WAF クライアントアプリケーション統合

AWS WAF クライアントアプリケーション統合 API を使用すると、 AWS クライアント側の保護とサーバー側のウェブ ACL 保護を組み合わせて、保護対象リソースにウェブリクエストを送信するク

ライアントアプリケーションが目的のクライアントであり、エンドユーザーが人間であることを確認できます。

クライアント統合を使用して、サイレントブラウザのチャレンジと CAPTCHA パズルを管理し、ブラウザとエンドユーザーの応答が成功したことを証明するトークンを取得し、保護されたエンドポイントへのリクエストにこれらのトークンを含めます。トークンの一般的な情報については、を参照してください。 AWS WAF AWS WAF ウェブリクエストトークン

クライアント統合を、リソースへのアクセスに有効なトークンを必要とするウェブ ACL 保護と組み合わせます。次のセクション インテリジェントな脅威に対応した統合と AWS マネージドルール に示されているような、チャレンジトークンをチェックおよびモニタリングするルールグループを使用できるため、「CAPTCHAChallengeの および AWS WAF」の説明に従い、CAPTCHA および Challenge ルールアクションを使用してチェックします。

AWS WAF アプリケーションには 2 つの統合レベルがあり、 JavaScript モバイルアプリケーションには 1 つのレベルの統合があります。

- インテリジェントな脅威統合 — クライアントアプリケーションを検証し、 AWS トークンの取得と管理を行います。 AWS WAF Challengeこれはルールアクションによって提供される機能に似ています。この機能により、クライアントアプリケーションは、`AWSManagedRulesACFPRuleSet` マネージドルールグループ、`AWSManagedRulesATPRuleSet` マネージドルールグループ、および `AWSManagedRulesBotControlRuleSet` マネージドルールグループのターゲットを絞った保護レベルと完全に統合されます。

  インテリジェント・スレット・インテグレーション API は、 AWS WAF サイレント・ブラウザ・チャレンジを使用して、保護対象リソースへのログイン試行やその他の呼び出しが、クライアントが有効なトークンを取得した後にのみ許可されるようにします。API は、クライアントアプリケーションセッションのトークン認証を管理し、クライアントに関する情報を収集して、ボットによる操作か、人間による操作かを判断します。

  > **Note**
  >
  > これは Android JavaScript と iOS のモバイルアプリケーションの両方で使用できます。

- CAPTCHA 統合 – アプリケーションで管理するカスタマイズされた CAPTCHA パズルでエンドユーザーを検証します。 AWS WAF CAPTCHAこれはルールアクションの機能と似ていますが、パズルの配置と動作をさらに制御できます。

この統合では、 JavaScript インテリジェントな脅威統合を活用してサイレントチャレンジを実行し、 AWS WAF 顧客のページにトークンを提供します。

> **Note**
>
> JavaScript これはアプリケーションでも利用できます。

トピック

- インテリジェントな脅威に対応した統合と AWS マネージドルール
- AWS WAF クライアントアプリケーション統合 APIsへのアクセス
- AWS WAF JavaScript 統合
- AWS WAF モバイルアプリケーション統合

インテリジェントな脅威に対応した統合と AWS マネージドルール

インテリジェントな脅威統合 API は、インテリジェントな脅威ルールグループを使用するウェブ ACL と連携して、これらの高度なマネージドルールグループの全機能を有効にします。

- AWS WAF 不正防止アカウント作成詐欺防止 (ACFP) マネージドルールグループ。`AWSManagedRulesACFPRuleSet`

  アカウント作成の不正行為は、サインアップボーナスの受け取りやなりすましなどの目的で、攻撃者がアプリケーションで無効なアカウントを作成するオンライン上の違法行為です。ACFP マネージドルールグループには、不正なアカウント作成の試みの一部の可能性があるリクエストをブロック、ラベル付け、管理するためのルールが含まれています。API は、ACFP ルールが使用する微調整されたクライアントブラウザの検証および人間のインタラクティビティに関する情報を有効にし、有効なクライアントトラフィックを悪意のあるトラフィックから分離します。

  詳細については、AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) ルールグループおよびAWS WAF 不正防止アカウント作成詐欺防止 (ACFP)を参照してください。

- AWS WAF 不正防止アカウント乗っ取り防止 (ATP) が管理するルールグループ。`AWSManagedRulesATPRuleSet`

  アカウント乗っ取りは、攻撃者が個人のアカウントへの不正アクセスを得るオンラインの違法行為です。ATP マネージドルールグループには、悪意のあるアカウント乗っ取りの試行の一部の可

能性があるリクエストをブロック、ラベル付け、管理するためのルールが含まれています。API は、ATP ルールが使用する微調整されたクライアント検証および動作集約を有効にし、有効なクライアントトラフィックを悪意のあるトラフィックから分離します。

詳細については、AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループおよびAWS WAF 不正防止アカウント乗っ取り防止 (ATP)を参照してください。

- AWS WAF Bot Control 管理ルールグループの目標とする保護レベル。`AWSManagedRulesBotControlRuleSet`

  ボットには、ほとんどの検索エンジンやクローラーなど、自己識別型や便利なボットから、ウェブサイトを狙って動作し、自己識別を行わない悪意のあるものまでの範囲に及びます。Bot Control マネージドルールグループは、ウェブトラフィックのボットアクティビティをモニタリング、ラベル付け、管理するルールを提供します。このルールグループのターゲットを絞った保護レベルを使用すると、ターゲットルールは API が提供するクライアントセッション情報を使用し、悪意のあるボットをより適切に検出します。

  詳細については、AWS WAF Bot Control ルールグループおよびAWS WAF ボットコントロールを参照してください。

これらのマネージドルールグループのいずれかをウェブ ACL に追加するには、手順 ACFP マネージドルールグループをウェブ ACL に追加、ATP マネージドルールグループをウェブ ACL に追加、AWS WAF ボットコントロールマネージドルールグループをウェブ ACL に追加する を参照してください。

> **Note**
>
> マネージドルールグループは現在、トークンが不足しているリクエストをブロックしていません。トークンが不足しているリクエストをブロックするには、アプリケーション統合 API を実装した後、有効な AWS WAF トークンがないリクエストのブロック のガイダンスに従ってください。

AWS WAF クライアントアプリケーション統合 APIsへのアクセス

JavaScript 統合 APIs は一般公開されており、 を実行するブラウザやその他のデバイスに使用できます JavaScript。

AWS WAF は、Android および iOS モバイルアプリ用のカスタムのインテリジェントな脅威に対応した統合 SDKs を提供します。

- Android モバイルアプリの場合、 AWS WAF SDKsは Android API バージョン 23 (Android バージョン 6) 以降で動作します。Android バージョンの詳細については、「SDK Platform リリースノート」を参照してください。
- iOS モバイルアプリの場合、 AWS WAF SDKs iOS バージョン 13 以降で動作します。iOS バージョンの詳細については、「iOS と iPadOS のリリースノート」を参照してください。

コンソールで統合 API にアクセスするには

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/wafv2/ で AWS WAF コンソールを開きます。
2. ナビゲーションペインの [アプリケーション統合] を選択してから、関心のあるタブを選択します。

   - インテリジェントな脅威に対応した統合は、 JavaScript およびモバイルアプリケーションで利用できます。

     タブには次のものが含まれています。

     - インテリジェントな脅威に対応したアプリケーション統合が有効になっているウェブ ACL のリスト。リストには、`AWSManagedRulesACFPRuleSet` マネージドルールグループ、`AWSManagedRulesATPRuleSet` マネージドルールグループ、または `AWSManagedRulesBotControlRuleSet` マネージドルールグループのターゲットを絞った保護レベルを使用する各ウェブ ACL が含まれます。インテリジェントな脅威に対応した API を実装するときは、統合するウェブ ACL の統合 URL を使用します。
     - ユーザーがアクセスできる API。 JavaScript APIs使用できます。モバイル SDK にアクセスするには、「AWSへのお問い合わせ」にてサポート担当者までお問い合わせください。
   - CAPTCHA 統合は JavaScript アプリケーションで使用できます。

     タブには次のものが含まれています。

     - 統合で使用する統合 URL。
     - クライアントアプリケーションドメイン用に作成した API キー。CAPTCHA API を使用するには、暗号化された API キーが必要です。これにより、クライアントは自分のドメインから AWS WAF CAPTCHA にアクセスできます。統合するクライアントごとに、クライア

ントのドメインを含む API キーを使用します。これらの要件とキーの管理の詳細については、「JS キャプチャ API の API キーの管理」を参照してください。

AWS WAF JavaScript 統合

JavaScript 統合 APIs、 を実行するブラウザやその他のデバイスに AWS WAF アプリケーション統合を実装できます JavaScript。

CAPTCHA パズルとサイレントチャレンジは、ブラウザが HTTPS エンドポイントにアクセスしている場合にのみ実行できます。トークンを取得するには、ブラウザクライアントが安全なコンテキストで実行されている必要があります。

- インテリジェントな脅威に対応した API を使用すると、クライアント側のサイレントブラウザのチャレンジを通じてトークン認証を管理し、保護されたリソースに送信するリクエストにトークンを含めることができます。
- CAPTCHA 統合 API にインテリジェントな脅威に対応した API が追加され、クライアントアプリケーションでの CAPTCHA パズルの配置と特性をカスタマイズすることができるようになりました。この API は、インテリジェントな脅威に対応した API を活用し、エンドユーザーが CAPTCHA パズルの完成に成功した後にページで使用する AWS WAF トークンを取得します。

これらの統合を使用すると、クライアントによるリモートプロシージャコールに有効なトークンが含まれていることを確認できます。これらの統合 API がアプリケーションのページで実行されている場合、有効なトークンを含まないリクエストをブロックするなどの緩和ルールをウェブ ACL で実装できます。ルール内の Challenge または CAPTCHA アクションを使用して、クライアントアプリケーションが取得したトークンの使用を強制するルールを実装することもできます。

次のリストは、ウェブアプリケーションページでのインテリジェントな脅威に対応した API の一般的な実装の基本コンポーネントを示しています。

```
<head>
<script type="text/javascript" src="Web ACL integration URL/challenge.js" defer></
script>
</head>
<script>
const login_response = await AwsWafIntegration.fetch(login_url, {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json'
    },
```

```
    body: login_body
  });
</script>
```

CAPTCHA 統合 API を使用すると、エンドユーザーの CAPTCHA パズルエクスペリエンスをカスタマイズできます。CAPTCHA 統合は、ブラウザの検証とトークン管理のために JavaScript インテリジェントな脅威に対応した統合を活用し、CAPTCHA パズルを設定およびレンダリングする機能を追加します。

次のリストは、ウェブアプリケーションページにおける CAPTCHA JavaScript API の一般的な実装の基本コンポーネントを示しています。

```
<head>
    <script type="text/javascript" src="<Integration URL>/jsapi.js" defer></script>
</head>

<script type="text/javascript">
    function showMyCaptcha() {
        var container = document.querySelector("#my-captcha-container");

        AwsWafCaptcha.renderCaptcha(container, {
            apiKey: "...API key goes here...",
            onSuccess: captchaExampleSuccessFunction,
            onError: captchaExampleErrorFunction,
            ...other configuration parameters as needed...
        });
    }

    function captchaExampleSuccessFunction(wafToken) {
        // Use WAF token to access protected resources
        AwsWafIntegration.fetch("...WAF-protected URL...", {
            method: "POST",
            ...
        });
    }

    function captchaExampleErrorFunction(error) {
        /* Do something with the error */
    }
</script>

<div id="my-captcha-container">
    <!-- The contents of this container will be replaced by the captcha widget -->
```

```
</div>
```

トピック

- トークンで使用するドメインの提供
- JavaScript API とコンテンツセキュリティポリシーの併用
- インテリジェントな脅威 JavaScript API の使用
- キャプチャ API JavaScript を使用する

トークンで使用するドメインの提供

デフォルトでは、 AWS WAF トークンを作成すると、ウェブ ACL に関連付けられているリソースのホストドメインが使用されます。 JavaScript API AWS WAF 用に作成するトークンには、追加のドメインを指定できます。これを行うには、グローバル変数 `window.awsWafCookieDomainList` に 1 つ以上のトークンドメインを設定します。

AWS WAF トークンを作成すると、ウェブ ACL `window.awsWafCookieDomainList` に関連付けられているリソースのドメインとホストドメインの組み合わせの中から、最も適切で短いドメインが使用されます。

設定の例

```
window.awsWafCookieDomainList = ['.aws.amazon.com']
```

```
window.awsWafCookieDomainList = ['.aws.amazon.com', 'abc.aws.amazon.com']
```

このリストではパブリックサフィックスを使用できません。例えば、`gov.au` または `co.uk` をリストでトークンドメインとして使用することはできません。

このリストで指定するドメインは、他のドメインやドメイン設定と互換性がある必要があります。

- ドメインは、保護対象のホストドメインとウェブ ACL 用に設定されたトークンドメインリストに基づいて、受け入れられるドメインである必要があります。 AWS WAF 詳細については、「AWS WAF ウェブ ACL トークンドメインリストの設定」を参照してください。
- JavaScript CAPTCHA API を使用する場合、CAPTCHA API キー内の少なくとも 1 つのドメインが、内のトークンドメインの 1 `window.awsWafCookieDomainList` つと完全に一致するか、それらのトークンドメインのいずれかの apex ドメインである必要があります。

例えば、トークンドメイン mySubdomain.myApex.com の場合、API キー mySubdomain.myApex.com は完全に一致し、API キー myApex.com は apex ドメインです。どちらかのキーがトークンドメインに一致します。

API キーの詳細については、「JS キャプチャ API の API キーの管理」を参照してください。

AWSManagedRulesACFPRuleSet マネージドルールグループを使用する場合、ルールグループ設定に指定したアカウント作成パスのドメインと一致するドメインを設定できます。この設定の詳細については、「ACFP マネージドルールグループをウェブ ACL に追加」を参照してください。

AWSManagedRulesATPRuleSet マネージドルールグループを使用する場合、ルールグループ設定に指定したログインパスのドメインと一致するドメインを設定できます。この設定の詳細については、「ATP マネージドルールグループをウェブ ACL に追加」を参照してください。

JavaScript API とコンテンツセキュリティポリシーの併用

リソースにコンテンツセキュリティポリシー (CSP) を適用する場合、 JavaScript実装が機能するためには、 AWS WAF apex ドメインを許可リストに登録する必要があります。awswaf.com JavaScript SDK AWS WAF は異なるエンドポイントを呼び出すため、このドメインをホワイトリストに登録すると、SDK の運用に必要な権限が付与されます。

Apex ドメインを許可リストに登録する設定例を以下に示します。 AWS WAF

```
connect-src 'self' https://*.awswaf.com;
script-src 'self' https://*.awswaf.com;
script-src-elem 'self' https://*.awswaf.com;
```

CSP を使用するリソースで JavaScript SDK を使用しようとしても、 AWS WAF ドメインを許可リストに登録していないと、次のようなエラーが表示されます。

```
Refused to load the script ...awswaf.com/<> because it violates the following Content
 Security Policy directive: “script-src ‘self’
```

インテリジェントな脅威 JavaScript API の使用

インテリジェントな脅威 API では、ユーザーのブラウザに対してサイレントチャレンジを実行したり、 AWS WAF チャレンジの成功を証明するトークンや CAPTCHA レスポンスを処理したりするための操作を行うことができます。

JavaScript インテグレーションは、まずテスト環境で実装し、次に本番環境に実装します。追加のコーディングガイダンスについては、次のセクションを参照してください。

インテリジェントな脅威に対応した API を使用するには

1. API のインストール

   CAPTCHA API を使用している場合は、このステップをスキップできます。CAPTCHA API をインストールすると、スクリプトはインテリジェントな脅威に対応した API を自動的にインストールします。

   a. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   b. ナビゲーションペインで、[Application integration] (アプリケーション統合) を選択します。[アプリケーション統合] ページに、タブ付きのオプションが表示されます。

   c. [インテリジェントな脅威に対応した統合] を選択

   d. タブで、統合するウェブ ACL を選択します。ウェブ ACL リストには、`AWSManagedRulesACFPRuleSet` マネージドルールグループ、`AWSManagedRulesATPRuleSet` マネージドルールグループ、または `AWSManagedRulesBotControlRuleSet` マネージドルールグループのターゲットを絞った保護レベルを使用するウェブ ACL のみが含まれます。

   e. JavaScript SDK ペインを開き、インテグレーションで使用する script タグをコピーします。

   f. <head> セクションのアプリケーションページコードに、ウェブ ACL 用にコピーしたスクリプトタグを挿入します。この包含により、クライアントアプリケーションは、ページをロードする際にバックグラウンドでトークンを自動的に取得します。

   ```
   <head>
       <script type="text/javascript" src="Web ACL integration URL/challenge.js”
    defer></script>
   <head>
   ```

   この <script> リストは `defer` 属性で設定されていますが、ページに別の動作が必要な場合は、設定を `async` に変更できます。

2. (オプション) クライアントのトークンのドメイン設定を追加 — デフォルトでは、 AWS WAF トークンを作成すると、ウェブ ACL に関連付けられているリソースのホストドメインが使用

されます。 JavaScript API に追加のドメインを指定するには、のガイダンスに従ってくださいトークンで使用するドメインの提供。

3. インテリジェントな脅威に対応した統合をコーディングする – クライアントで保護されたエンドポイントにリクエストを送信する前に、トークンの取得が完了するようにコードを記述します。既に `fetch` API を使用して呼び出しを行っている場合は、 AWS WAF 統合 `fetch` ラッパーに置き換えることができます。`fetch`API を使用しない場合は、 AWS WAF `getToken`代わりに統合操作を使用できます。コーディングガイダンスについては、次のセクションを参照してください。

4. ウェブ ACL にトークン検証を追加する – クライアントで送信するウェブリクエスト内に有効なチャレンジトークンがないかチェックするルールを、ウェブ ACL に少なくとも 1 つ追加します。Bot Control マネージドルールグループのターゲットレベルのような、チャレンジトークンをチェックおよびモニタリングするルールグループを使用できるため、「CAPTCHAChallenge の および AWS WAF」の説明に従い、Challenge ルールアクションを使用してチェックします。

   ウェブ ACL を追加すると、保護されたエンドポイントへのリクエストに、クライアント統合で取得したトークンが含まれていることを確認できます。有効で期限が切れていないトークンを含むリクエストは、Challenge 検査に合格し、クライアントに別のサイレントチャレンジを送信することはありません。

5. (オプション) トークンが不足しているリクエストをブロックする – ACFP マネージドルールグループ、ATP マネージドルールグループまたは Bot Control ルールグループのターゲットを絞ったルールで API を使用する場合、これらのルールはトークンが不足しているリクエストをブロックしません。トークンが不足しているリクエストをブロックするには、有効な AWS WAF トークンがないリクエストのブロック のガイダンスに従ってください。

トピック

- インテリジェントな脅威に対応した API 仕様
- 統合 fetch ラッパーの使用方法
- 統合 getToken を使用する方法

インテリジェントな脅威に対応した API 仕様

このセクションでは、インテリジェント脅威対策 JavaScript API のメソッドとプロパティの仕様を一覧表示します。インテリジェントな脅威に対応したこれらの API と CAPTCHA 統合を使用します。

AwsWafIntegration.fetch()

AWS WAF インテグレーション実装を使用して HTTP fetch リクエストをサーバーに送信します。

AwsWafIntegration.getToken()

AWS WAF 保存されているトークンを取得し、現在のページの Cookie に名前を付けて保存しaws-waf-token、値をトークン値に設定します。

AwsWafIntegration.hasToken()

有効期限が切れていないトークンが現在 aws-waf-token cookie で保持されているかどうかを示すブール値を返します。

CAPTCHA 統合も使用している場合は、「キャプチャ API 仕様 JavaScript 」でその仕様を確認してください。

統合 **fetch** ラッパーの使用方法

AwsWafIntegration 名前空間の下で fetch API に対する通常の fetch 呼び出しを変更することにより、 AWS WAF fetch ラッパーを使用できます。 AWS WAF ラッパーは標準 JavaScript fetch API 呼び出しと同じオプションをすべてサポートし、統合用のトークン処理を追加します。このアプローチは、一般的に、アプリケーションを統合する最も簡単な方法です。

ラッパーの実装前

次のリスト例は、AwsWafIntegration fetch ラッパーを実装する前の標準コードを示しています。

```
const login_response = await fetch(login_url, {
     method: 'POST',
     headers: {
       'Content-Type': 'application/json'
     },
     body: login_body
   });
```

ラッパー実装後

次のリストは、AwsWafIntegration fetch ラッパー実装と同じコードを示しています。

```
const login_response = await AwsWafIntegration.fetch(login_url, {
```

```
    method: 'POST',
    headers: {
      'Content-Type': 'application/json'
    },
    body: login_body
  });
```

統合 **getToken** を使用する方法

AWS WAF 保護されたエンドポイントへのリクエストには、aws-waf-token現在のトークンの値で名前を付けられた Cookie を含める必要があります。

getToken オペレーションとは、 AWS WAF トークンを取得し、名前を aws-waf-token にして、値をトークン値に設定し、現在のページの cookie に保存する非同期 API コールです。このトークン cookie は、必要に応じてページで使用できます。

getToken を呼び出すと、次が実行されます。

- 期限切れでないトークンが既に使用可能な場合、コールは直ちにそれを返します。
- それ以外の場合、コールは トークンプロバイダーから新しいトークンを取得します。トークン取得ワークフローが完了するまで最大で 2 秒間の猶予があり、この待機期間が経過するとタイムアウトします。オペレーションがタイムアウトすると、呼び出しコードが処理しなければならないエラーがスローされます。

getToken オペレーションには付随する hasToken オペレーションがあり、aws-waf-token cookie が現在有効期限が切れていないトークンを保持しているかどうかを示します。

AwsWafIntegration.getToken()有効なトークンを取得して Cookie として保存します。ほとんどのクライアントコールは自動的にこの Cookie をアタッチしますが、アタッチしないクライアントコールもあります。たとえば、ホストドメイン間で行われた呼び出しは Cookie をアタッチしません。以下の実装の詳細では、両方のタイプのクライアント呼び出しを処理する方法を示します。

aws-waf-token Cookie **getToken** をアタッチする呼び出し用の基本実装

次のリストの例は、ログインリクエストで getToken オペレーションを実装するための標準コードを示しています。

```
const login_response = await AwsWafIntegration.getToken()
    .catch(e => {
        // Implement error handling logic for your use case
```

```
    })
    // The getToken call returns the token, and doesn't typically require special
 handling
    .then(token => {
        return loginToMyPage()
    })

 async function loginToMyPage() {
    // Your existing login code
 }
```

トークンが **getToken** から利用可能になった後にのみフォームを送信する

次のリストは、有効なトークンが使用可能になるまで、フォーム送信をインターセプトするイベントリスナーを登録する方法を示しています。

```
<body>
   <h1>Login</h1>
   <p></p>
   <form id="login-form" action="/web/login" method="POST" enctype="application/x-www-
form-urlencoded">
     <label for="input_username">USERNAME</label>
     <input type="text" name="input_username" id="input_username"><br>
     <label for="input_password">PASSWORD</label>
     <input type="password" name="input_password" id="input_password"><br>
     <button type="submit">Submit<button>
   </form>

 <script>
   const form = document.querySelector("#login-form");

   // Register an event listener to intercept form submissions
   form.addEventListener("submit", (e) => {
       // Submit the form only after a token is available
       if (!AwsWafIntegration.hasToken()) {
           e.preventDefault();
           AwsWafIntegration.getToken().then(() => {
               e.target.submit();
           }, (reason) => { console.log("Error:"+reason) });
         }
     });
 </script>
 </body>
```

クライアントがデフォルトで **aws-waf-token** Cookie をアタッチしない場合にトークンをアタッチする

AwsWafIntegration.getToken()有効なトークンを取得して Cookie として保存しますが、すべてのクライアント呼び出しがデフォルトでこの Cookie を添付するわけではありません。たとえば、ホストドメイン間で行われる呼び出しには Cookie は添付されません。

fetchラッパーはこれらのケースを自動的に処理しますが、fetchラッパーを使用できない場合は、x-aws-waf-tokenカスタムヘッダーを使用して処理できます。 AWS WAF Cookie からトークンを読み取るだけでなく、このヘッダーからもトークンを読み取ります。aws-waf-token次のコードはヘッダーの設定例を示しています。

```
const token = await AwsWafIntegration.getToken();
const result = await fetch('/url', {
    headers: {
        'x-aws-waf-token': token,
    },
});
```

デフォルトでは、 AWS WAF リクエストされたホストドメインと同じドメインを含むトークンのみを受け入れます。クロスドメイントークンには、ウェブ ACL トークンのドメインリストに対応するエントリが必要です。詳細については、「AWS WAF ウェブ ACL トークンドメインリストの設定」を参照してください。

クロスドメイントークンの使用に関する追加情報については、「aws-waf-bot-controlaws-samples/-」を参照してください。api-protection-with-captcha

キャプチャ API JavaScript を使用する

CAPTCHA JavaScript API では、CAPTCHA パズルを設定し、クライアントアプリケーションの任意の場所に配置できます。この API は、インテリジェントな脅威 JavaScript API の機能を活用して、エンドユーザーが CAPTCHA AWS WAF パズルを無事に完成させた後にトークンを取得して使用します。

JavaScript インテグレーションは、まずテスト環境で実装し、次に本番環境に実装します。追加のコーディングガイダンスについては、次のセクションを参照してください。

CAPTCHA 統合 API を使用するには

1. API のインストール

a. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

b. ナビゲーションペインで、[Application integration] (アプリケーション統合) を選択します。[アプリケーション統合] ページに、タブ付きのオプションが表示されます。

c. [CAPTCHA 統合] を選択します。

d. JavaScript リストされているインテグレーションスクリプトタグをコピーして、インテグレーションで使用します。

e. <head> セクションのアプリケーションページコードに、コピーしたスクリプトタグを挿入します。これにより、CAPTCHA パズルの設定および使用が可能になります。

```
<head>
    <script type="text/javascript" src="integrationURL/jsapi.js" defer></
script>
</head>
```

この <script> リストは defer 属性で設定されていますが、ページに別の動作が必要な場合は、設定を async に変更できます。

また、CAPTCHA スクリプトは、インテリジェントな脅威に対応した統合のスクリプトがまだ存在しない場合は自動的にロードします。インテリジェントな脅威に対応した統合のスクリプトにより、クライアントアプリケーションは、ページをロードする際にバックグラウンドでトークンを自動的に取得し、CAPTCHA API の使用に必要なその他のトークン管理機能を提供します。

2. (オプション) クライアントのトークンのドメイン設定を追加 — デフォルトでは、 AWS WAF トークンを作成すると、ウェブ ACL に関連付けられているリソースのホストドメインが使用されます。 JavaScript API に追加のドメインを指定するには、のガイダンスに従ってくださいトークンで使用するドメインの提供。

3. クライアントの暗号化された API キーを取得 — CAPTCHA API には、有効なクライアントドメインのリストを含む暗号化された API キーが必要です。 AWS WAF このキーを使用して、インテグレーションで使用しているクライアントドメインが CAPTCHA AWS WAF の使用を承認されていることを確認します。API キーを生成するには、「JS キャプチャ API の API キーの管理」のガイダンスに従ってください。

4. CAPTCHA ウィジェットの実装をコーディングする – renderCaptcha() API コールを、ページ内の使用したい場所に実装します。この機能の設定および使用方法については、以下のセク

ション「キャプチャ API 仕様 JavaScript 」と「CAPTCHA パズルをレンダリングする方法」を参照してください。

CAPTCHA 実装はインテリジェント脅威統合 API と統合され、トークンの管理とトークンを使用するフェッチ呼び出しの実行を行います。 AWS WAF これらの API の使用に関するガイダンスについては、「インテリジェントな脅威 JavaScript API の使用」を参照してください。

5. ウェブ ACL にトークン検証を追加する – クライアントで送信するウェブリクエスト内に有効な CAPTCHA トークンがないかチェックするルールを、ウェブ ACL に少なくとも 1 つ追加します。「CAPTCHAChallengeの および AWS WAF」の説明に従い、CAPTCHA ルールアクションを使用してチェックします。

   ウェブ ACL の追加により、保護されたエンドポイントに送信されるリクエストに、クライアント統合で取得したトークンが含まれていることを確認できます。有効で期限が切れていない CAPTCHA トークンを含むリクエストは、CAPTCHA ルールアクション検査に合格し、エンドユーザーに別の CAPTCHA パズルを提示することはありません。

トピック

- キャプチャ API 仕様 JavaScript
- CAPTCHA パズルをレンダリングする方法
- からのキャプチャレスポンスを処理する AWS WAF
- JS キャプチャ API の API キーの管理

キャプチャ API 仕様 JavaScript

このセクションでは、 JavaScript CAPTCHA API のメソッドとプロパティの仕様を一覧表示します。CAPTCHA JavaScript API を使用して、クライアントアプリケーションでカスタム CAPTCHA パズルを実行します。

この API は、 AWS WAF トークンの取得と使用を設定および管理するために使用するインテリジェントな脅威 API をベースにしています。「インテリジェントな脅威に対応した API 仕様」を参照してください。

`AwsWafCaptcha.renderCaptcha(container, configuration)`

エンドユーザーに AWS WAF CAPTCHA パズルを提示し、成功すると CAPTCHA 検証でクライアントトークンを更新します。これは CAPTCHA 統合でのみ使用できます。この呼び出しをインテリジェントな脅威に対応した API と組み合わせて使用すると、トークンの取得を管理した

り、fetch コールでトークンを提供したりできます。インテリジェントな脅威に対応した API については、「インテリジェントな脅威に対応した API 仕様」を参照してください。

AWS WAF 送信する CAPTCHA インタースティシャルとは異なり、このメソッドでレンダリングされた CAPTCHA パズルでは、最初のタイトル画面なしでパズルがすぐに表示されます。

container

ページ上のターゲットとなるコンテナ要素の Element オブジェクト。これは通常、document.getElementById() または document.querySelector() を呼び出すことで取得できます。

必須: はい

タイプ: Element

設定

以下のような CAPTCHA 構成設定を含むオブジェクト。

apiKey

クライアントのドメインの許可を有効にする暗号化された API キー。 AWS WAF コンソールを使用して、クライアントドメインの API キーを生成します。1 つのキーを最大 5 つのドメインに使用できます。詳細については、JS キャプチャ API の API キーの管理 を参照してください。

必須: はい

タイプ: string

onSuccess: (wafToken: string) => void;

エンドユーザーが CAPTCHA AWS WAF パズルを正常に完了すると、有効なトークンで呼び出されます。ウェブ ACL で保護するエンドポイントに送信するリクエストでは、このトークンを使用してください。 AWS WAF トークンは、パズルの完成に最後に成功したときの証明とタイムスタンプを提供します。

必須: はい

onError?: (error: CaptchaError) => void;

CAPTCHA 操作中にエラーが発生したときに、エラーオブジェクトとともに呼び出されます。

必須: いいえ

CaptchaError クラス定義 – onError ハンドラーは、次のクラス定義を使用してエラータイプを提供します。

```
CaptchaError extends Error {
    kind: "internal_error" | "network_error" | "token_error" | "client_error";
    statusCode?: number;
}
```

- kind – 返されたエラーの種類。
- statusCode – HTTP ステータスコード (利用可能な場合)。これは、エラーが HTTP エラーに起因する場合に network_error で使用されます。

onLoad?: () => void;

新しい CAPTCHA パズルがロードされると呼び出されます。

必須: いいえ

onPuzzleTimeout?: () => void;

CAPTCHA パズルが期限切れになる前に完成しなかった場合に呼び出されます。

必須: いいえ

onPuzzleCorrect?: () => void;

CAPTCHA パズルに正しい回答が入力されると呼び出されます。

必須: いいえ

onPuzzleIncorrect?: () => void;

CAPTCHA パズルに間違った回答が入力されると呼び出されます。

必須: いいえ

defaultLocale

CAPTCHA パズルに使用するデフォルトのロケール。CAPTCHA パズルの説明書は、アラビア語 (ar-SA)、簡体字中国語 (zh-CN)、オランダ語 (nl-NL)、英語 (en-US)、フランス語 (fr-FR)、ドイツ語 (de-DE)、イタリア語 (it-IT)、日本語 (ja-JP)、ブラジルポルトガル語 (pt-BR)、スペイン語 (es-ES)、およびトルコ語 (tr-TR) で提供されています。音声による説明は、中国語と日本語 (デフォルトでは英語) を除くすべての記述言語で利用できます。デフォルト言語を変更するには、国際言語とロケールコード (例:) を指定します。ar-SA

デフォルト: エンドユーザーのブラウザで現在使用されている言語

必須: いいえ

タイプ: `string`

`disableLanguageSelector`

`true` に設定すると、CAPTCHA パズルの言語セレクタが非表示になります。

デフォルト: `false`

必須: いいえ

タイプ: `boolean`

`dynamicWidth`

`true` に設定すると、CAPTCHA パズルの幅はブラウザウィンドウの幅に合わせて変更されます。

デフォルト: `false`

必須: いいえ

タイプ: `boolean`

`skipTitle`

`true` に設定すると、CAPTCHA パズルのパズルタイトルに「パズルを解く」という見出しが表示されません。

デフォルト: `false`

必須: いいえ

タイプ: `boolean`

CAPTCHA パズルをレンダリングする方法

AWS WAF `renderCaptcha`クライアントインターフェースで好きな場所でコールを使用できます。呼び出しは AWS WAF、CAPTCHA パズルを取得してレンダリングし、結果を検証用に送信します。 AWS WAF 呼び出しを行うときは、パズルのレンダリング設定と、エンドユーザーがパズルを完成させたときに実行するコールバックを指定します。オプションの詳細については、前のセクション「キャプチャ API 仕様 JavaScript 」を参照してください。

この呼び出しは、インテリジェントな脅威に対応した統合 API のトークン管理機能と組み合わせて使用します。この呼び出しにより、CAPTCHA パズルの完成に成功したことを検証するトークンがクライアントに渡されます。インテリジェントな脅威統合 API を使用してトークンを管理し、ウェブ ACL で保護されているエンドポイントへのクライアントの呼び出しでトークンを提供します。AWS WAF インテリジェントな脅威に対応した API の詳細については、「インテリジェントな脅威 JavaScript API の使用」を参照してください。

実装例

以下のリストの例は、 AWS WAF セクション内の統合 URL の配置を含む標準 CAPTCHA 実装を示しています。<head>

このリストは、インテリジェントな脅威に対応した統合 API の AwsWafIntegration.fetch ラッパーを使用する成功コールバックを含む renderCaptcha 関数で構成されています。この関数については、「統合 fetch ラッパーの使用方法」を参照してください。

```
<head>
    <script type="text/javascript" src="<Integration URL>/jsapi.js" defer></script>
</head>

<script type="text/javascript">
    function showMyCaptcha() {
        var container = document.querySelector("#my-captcha-container");

        AwsWafCaptcha.renderCaptcha(container, {
            apiKey: "...API key goes here...",
            onSuccess: captchaExampleSuccessFunction,
            onError: captchaExampleErrorFunction,
            ...other configuration parameters as needed...
        });
    }

    function captchaExampleSuccessFunction(wafToken) {
        // Captcha completed. wafToken contains a valid WAF token. Store it for
        // use later or call AwsWafIntegration.fetch() to use it easily.
        // It will expire after a time, so calling AwsWafIntegration.getToken()
        // again is advised if the token is needed later on, outside of using the
        // fetch wrapper.

        // Use WAF token to access protected resources
        AwsWafIntegration.fetch("...WAF-protected URL...", {
            method: "POST",
```

```
            headers: {
                "Content-Type": "application/json",
            },
            body: "{ ... }" /* body content */
        });
    }

    function captchaExampleErrorFunction(error) {
        /* Do something with the error */
    }
</script>

<div id="my-captcha-container">
    <!-- The contents of this container will be replaced by the captcha widget -->
</div>
```

構成設定の例

次のリストの例は、幅とタイトルのオプションをデフォルト以外で設定した renderCaptcha を示しています。

```
        AwsWafCaptcha.renderCaptcha(container, {
            apiKey: "...API key goes here...",
            onSuccess: captchaExampleSuccessFunction,
            onError: captchaExampleErrorFunction,
            dynamicWidth: true,
            skipTitle: true
        });
```

設定オプションの詳細な情報については、「キャプチャ API 仕様 JavaScript 」を参照してください。

からのキャプチャレスポンスを処理する AWS WAF

AWS WAF CAPTCHAアクションを含むルールは、リクエストに有効な CAPTCHA タイムスタンプのトークンがない場合に、一致するウェブリクエストの評価を終了します。リクエストが GET テキスト/HTML 呼び出しの場合、CAPTCHA アクションは CAPTCHA パズルを含むインタースティシャルをクライアントに提供します。CAPTCHA JavaScript API を統合しないと、インタースティシャルがパズルを実行し、エンドユーザーが問題を解くことができれば、自動的にリクエストを再送信します。

CAPTCHA JavaScript API を統合して CAPTCHA 処理をカスタマイズしたら、終了する CAPTCHA レスポンスを検出してカスタム CAPTCHA を配信し、エンドユーザーがパズルを正常に解決したら、クライアントのウェブリクエストを再送信する必要があります。

次のコード例は、これを実行する方法を説明しています。

> **Note**
>
> AWS WAF CAPTCHAアクションレスポンスのステータスコードは HTTP 405 で、これを使用してこのコード内のレスポンスを認識します。CAPTCHA保護されたエンドポイントが HTTP 405 ステータスコードを使用して同じ呼び出しのために他の種類のレスポンスを通信する場合、このサンプルコードはそれらのレスポンスのためにも CAPTCHA パズルをレンダリングします。

```
<!DOCTYPE html>
<html>
<head>
    <script type="text/javascript" src="<Integration URL>/jsapi.js" defer></script>
</head>
<body>
    <div id="my-captcha-box"></div>
    <div id="my-output-box"></div>

    <script type="text/javascript">
    async function loadData() {
        // Attempt to fetch a resource that's configured to trigger a CAPTCHA
        // action if the rule matches. The CAPTCHA response has status=HTTP 405.
        const result = await AwsWafIntegration.fetch("/protected-resource");

        // If the action was CAPTCHA, render the CAPTCHA and return

        // NOTE: If the endpoint you're calling in the fetch call responds with HTTP
 405
        // as an expected response status code, then this check won't be able to tell
 the
        // difference between that and the CAPTCHA rule action response.

        if (result.status === 405) {
            const container = document.querySelector("#my-captcha-box");
            AwsWafCaptcha.renderCaptcha(container, {
                apiKey: "...API key goes here...",
```

```
                onSuccess() {
                    // Try loading again, now that there is a valid CAPTCHA token
                    loadData();
                },
            });
            return;
        }

        const container = document.querySelector("#my-output-box");
        const response = await result.text();
        container.innerHTML = response;
    }

    window.addEventListener("load", () => {
        loadData();
    });
    </script>
</body>
</html>
```

JS キャプチャ API の API キーの管理

JavaScript API を使用して AWS WAF CAPTCHA をクライアントアプリケーションに統合するには、CAPTCHA パズルを実行するクライアントドメインの JavaScript API 統合タグと暗号化された API キーが必要です。

CAPTCHA JavaScript アプリケーション統合用は、暗号化された API キーを使用して、クライアントアプリケーションドメインに CAPTCHA API を使用する権限があることを確認します。 AWS WAF クライアントから CAPTCHA API を呼び出すときは、 JavaScript 現在のクライアントのドメインを含むドメインリストを含む API キーを指定します。1 つの暗号化キーに最大 5 つのドメインを一覧表示できます。

API キー要件

CAPTCHA 統合で使用する API キーには、そのキーを使用するクライアントに適用されるドメインが含まれている必要があります。

- クライアントのインテリジェントな脅威に対応した統合で `window.awsWafCookieDomainList` を指定する場合、API キーの少なくとも 1 つのドメインが `window.awsWafCookieDomainList` のトークンドメインの 1 つと完全一致するか、いずれかのトークンドメインの apex ドメインである必要があります。

例えば、トークンドメイン mySubdomain.myApex.com の場合、API キー mySubdomain.myApex.com は完全に一致し、API キー myApex.com は apex ドメインです。どちらかのキーがトークンドメインに一致します。

トークンドメインリストの設定については、「トークンで使用するドメインの提供」を参照してください。

- それ以外の場合は、現在のドメインが API キーに含まれている必要があります。現在のドメインは、ブラウザのアドレスバーで確認できるドメインです。

使用するドメインは、保護対象ホストドメインとウェブ ACL 用に設定されたトークンドメインリストに基づいて、受け入れられるドメインでなければなりません。 AWS WAF 詳細については、「AWS WAF ウェブ ACL トークンドメインリストの設定」を参照してください。

API キーのリージョンの選択方法

AWS WAF AWS WAF 利用可能なすべてのリージョンで CAPTCHA API キーを生成できます。

原則として、CAPTCHA API キーにはウェブ ACL と同じリージョンを使用する必要があります。ただし、リージョンのウェブ ACL に全世界のユーザーがいることが予想される場合は、スコープが CAPTCHA JavaScript CloudFront 統合タグとスコープ指定された API キーを入手して CloudFront、リージョナルウェブ ACL で使用することができます。このアプローチにより、クライアントは自分に最も近いリージョンから CAPTCHA パズルを読み込むことができるため、レイテンシーが短縮されます。

CAPTCHA API キーのスコープが、それ以外のリージョンに設定されている場合 CloudFront 、複数のリージョンでの使用はサポートされていません。スコープが指定されたリージョンでのみ使用できます。

クライアントドメインの API キーを生成するには

統合 URL を取得し、API キーを生成して取得するには、コンソールを使用します。

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。
2. ナビゲーションペインで、[Application integration] (アプリケーション統合) を選択します。
3. [アプリケーション統合が有効になっている Web ACL] ペインで、API キーに使用するリージョンを選択します。CAPTCHA インテグレーションタブの API キーペインでリージョンを選択することもできます。

4. タブ [CAPTCHA 統合] を選択します。このタブには、 JavaScript インテグレーションで使用できる CAPTCHA インテグレーションタグと API キーのリストが表示されます。どちらも選択したリージョンを対象としています。
5. [API キー] ペインで、[キーを生成] を選択します。[キー生成] ダイアログが表示されます。
6. キーに含めるクライアントドメインを入力します。最大 5 つまで入力できます。完了したら、[キーを生成] を選択します。インターフェイスが CAPTCHA 統合タブに戻り、新しいキーが一覧表示されます。

   一度作成された API キーは、イミュータブルです。キーを変更する必要がある場合は、新しいキーを生成し、代わりにそれを使用します。
7. (オプション) 新しく生成されたキーをコピーして、統合で使用します。

この作業には、REST API または言語固有の AWS SDK を使用することもできます。REST API 呼び出しは CreateApiKey と ListApiKeys です。

API キーを削除するには

API キーを削除するには、REST API または言語固有の AWS SDK を使用する必要があります。REST API 呼び出しは DeleteApiKey です。コンソールを使用してキーを削除することはできません。

キーを削除した後、 AWS WAF すべての地域でそのキーの使用が禁止されるまでに最大 24 時間かかることがあります。

AWS WAF モバイルアプリケーション統合

AWS WAF モバイルSDKを使用して、 AWS WAF AndroidおよびiOSモバイルアプリケーション用のインテリジェントな脅威統合SDKを実装できます。

- Android モバイルアプリの場合、 AWS WAF SDK は Android API バージョン 23 (Android バージョン 6) 以降で動作します。Android バージョンの詳細については、「SDK Platform リリースノート」を参照してください。
- iOS モバイルアプリの場合、 AWS WAF SDK は iOS バージョン 13 以降で動作します。iOS バージョンの詳細については、「iOS と iPadOS のリリースノート」を参照してください。

モバイル SDK を使用すると、トークン認証を管理し、保護されたリソースに送信するリクエストにトークンを含めることができます。SDK を使用すると、クライアントによるこれらのリモートプロ

シージャコールに有効なトークンが含まれていることを確認できます。さらに、この統合がアプリケーションのページで実行されている場合、有効なトークンを含まないリクエストをブロックするなど、ウェブ ACL で緩和ルールを実装できます。

モバイル SDK にアクセスするには、「AWSへのお問い合わせ」にてサポート担当者までお問い合わせください。

> **Note**
>
> AWS WAF モバイル SDK は CAPTCHA のカスタマイズには使用できません。

SDK を使用する基本的な方法は、設定オブジェクトを使用してトークンプロバイダーを作成し、そのトークンプロバイダーを使用してトークンを取得することです。 AWS WAFデフォルトでは、トークンプロバイダーは、保護されたリソースに対するウェブリクエストに取得したトークンを含めます。

主要なコンポーネントを示す SDK 実装の一部を次に示します。詳細な例については、「AWS WAF モバイル SDK のコードの記述」を参照してください。

iOS

```
let url: URL = URL(string: "Web ACL integration URL")!
 let configuration = WAFConfiguration(applicationIntegrationUrl: url, domainName:
 "Domain name")
 let tokenProvider = WAFTokenProvider(configuration)
 let token = tokenProvider.getToken()
```

Android

```
URL applicationIntegrationURL = new URL("Web ACL integration URL");
 String domainName = "Domain name";
 WAFConfiguration configuration =
 WAFConfiguration.builder().applicationIntegrationURL(applicationIntegrationURL).domainName(
 WAFTokenProvider tokenProvider = new WAFTokenProvider(Application context,
 configuration);
 WAFToken token = tokenProvider.getToken();
```

AWS WAF モバイル SDK のインストール

モバイル SDK にアクセスするには、「AWSへのお問い合わせ」にてサポート担当者までお問い合わせください。

モバイル SDK を最初にテスト環境で実装し、次に本番環境で実装します。

AWS WAF モバイル SDK をインストールするには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。
2. ナビゲーションペインで、[Application integration] (アプリケーション統合) を選択します。
3. [インテリジェントな脅威に対応した統合] タブで、次の操作を行います。
   a. [Web ACLs that are enabled for application integration] (アプリケーション統合が有効になっているウェブ ACL) ペインで、統合するウェブ ACL を見つけます。実装で使用するウェブ ACL 統合 URL をコピーして保存します。この URL は、API コール `GetWebACL` を通じて取得することもできます。
   b. モバイルデバイスのタイプとバージョンを選択してから、[Download] (ダウンロード) を選択します。どのバージョンでも選択できますが、最新バージョンを使用することをおすすめします。 AWS WAF `zip`デバイス用のファイルを標準のダウンロード場所にダウンロードします。
4. アプリケーション開発環境で、ファイルを任意の作業場所に解凍します。zip ファイルの最上位ディレクトリで、`README` を見つけて開きます。`README`ファイルの指示に従って、 AWS WAF モバイルアプリコードで使用するモバイル SDK をインストールします。
5. 次のセクションのガイダンスに従って、アプリケーションをプログラムします。

AWS WAF モバイル SDK 仕様

このセクションでは、 AWS WAF モバイル SDK の利用可能な最新バージョンに対応する SDK オブジェクト、オペレーション、および構成設定を一覧表示します。構成設定のさまざまな組み合わせでトークンプロバイダーとオペレーションがどのように連携するかについては、「AWS WAF モバイル SDK の仕組み」を参照してください。

`WAFToken`

AWS WAF トークンを保持します。

`getValue()`

WAFToken の String 表現を取得します。

`WAFTokenProvider`

モバイルアプリケーションでトークンを管理します。WAFConfiguration オブジェクトを使用してこれを実装します。

`getToken()`

バックグラウンド更新が有効になっている場合、キャッシュされたトークンが返されます。バックグラウンド更新が無効になっている場合は、 AWS WAF 新しいトークンを取得するための同期的なブロッキング呼び出しが行われます。

`onTokenReady(WAFTokenResultCallback)`

アクティブなトークンの準備ができたら、トークンを更新し、指定されたコールバックを呼び出すようにトークンプロバイダーに指示します。トークンプロバイダーは、トークンがキャッシュされて準備ができたときに、バックグラウンドスレッドでコールバックを呼び出します。アプリケーションが最初にロードされたときや、アクティブ状態に戻ったときにこれを呼び出します。アクティブ状態に戻ることの詳細については、「the section called “アプリケーションが非アクティブ状態になった後のトークンの取得”」を参照してください。

Android または iOS アプリケーションの場合、WAFTokenResultCallback を、リクエストされたトークンの準備ができたときにトークンプロバイダーが呼び出すオペレーションに設定できます。WAFTokenResultCallback の実装では、パラメータ WAFToken、SdkError を取得する必要があります。iOS アプリケーションでは、代わりにインライン関数を作成できます。

`storeTokenInCookieStorage(WAFToken)`

AWS WAF 指定されたトークンを SDK の Cookie WAFTokenProvider マネージャーに保存するようにに指示します。デフォルトでは、トークンは最初に取得されたときと更新されたときのみ、Cookie ストアに追加されます。アプリケーションが何らかの理由で共有 Cookie ストアをクリアしても、SDK AWS WAF は次回の更新までトークンを自動的に追加しません。

`WAFConfiguration`

WAFTokenProvider の実装のための設定を保持します。これを実装すると、ウェブ ACL の統合 URL、トークンで使用するドメイン名、トークンプロバイダーが使用するデフォルト以外の設定を指定します。

次のリストは、WAFConfiguration オブジェクトで管理できる構成設定を示しています。

applicationIntegrationUrl

アプリケーション統合 URL。 AWS WAF これをコンソールまたは getWebACL API 呼び出しから取得します。

必須: はい

タイプ: アプリケーション固有の URL。iOS の場合は、「iOS URL」を参照してください。Android の場合は、「java.net URL」を参照してください。

backgroundRefreshEnabled

トークンプロバイダーがバックグラウンドでトークンを更新するかどうかを示します。これを設定すると、トークンプロバイダーは、自動トークン更新アクティビティを管理する構成設定に従って、バックグラウンドでトークンを更新します。

必須: いいえ

タイプ: Boolean

デフォルト値: TRUE

domainName

トークンで使用するドメインは、トークン取得とクッキーの保存に使用されます。例えば、example.com、aws.amazon.com などです。これは通常、ウェブ ACL に関連付けられているリソースのホストドメインであり、ウェブリクエストの送信先です。ACFP マネージドルールグループ AWSManagedRulesACFPRuleSet の場合、通常はルールグループ設定で指定したアカウント作成パスのドメインと一致する単一のドメインになります。ATP マネージドルールグループ AWSManagedRulesATPRuleSet の場合、通常はルールグループ設定で指定したログインパスのドメインと一致する単一のドメインになります。

パブリックサフィックスは許可されません。たとえば、gov.au または co.uk をトークンドメインとして使用することはできません。

ドメインは、保護されているホストドメインとウェブ ACL のトークンドメインリストに基づいて、受け入れられるドメインである必要があります。 AWS WAF 詳細については、「AWS WAF ウェブ ACL トークンドメインリストの設定」を参照してください。

必須: はい

タイプ: String

maxErrorTokenRefreshDelayMsec

失敗した試行後にトークンの更新を繰り返すまでの待機時間の最大値 (ミリ秒)。この値は、トークンの取得が失敗し、maxRetryCount 回再試行された後に使用されます。

必須: いいえ

タイプ: Integer

デフォルト値: 5000 (5 秒)

許容される最小値: 1 (1 ミリ秒)

許容される最大値: 30000 (30 秒)

maxRetryCount

トークンがリクエストされたときに、エクスポネンシャルバックオフで実行する最大再試行回数。

必須: いいえ

タイプ: Integer

デフォルト値: バックグラウンドリフレッシュが有効になっている場合、5。そうでない場合は、3 です。

許容される最小値: 0

許容される最大値: 10

setTokenCookie

SDK の cookie マネージャーがリクエストにトークン cookie を追加するかどうかを示します。デフォルトでは、これは、すべてのリクエストにトークン cookie を追加します。cookie マネージャーは、パスが tokenCookiePath で指定されたパスの下にあるすべてのリクエストにトークン cookie を追加します。

必須: いいえ

タイプ: Boolean

デフォルト値: TRUE

tokenCookiePath

setTokenCookie が TRUE の場合に使用されます。SDK の cookie マネージャーでトークン cookie を追加する最上位レベルのパスを示します。マネージャーは、このパスに送信するすべてのリクエストとすべての子パスにトークン cookie を追加します。

例えば、これを /web/login に設定すると、マネージャーには、/web/login に送信されるすべてのトークン cookie と、その子パスのいずれかが含まれます (/web/login/help など)。/、/web、/web/order などの他のパスに送信されたリクエストのトークンは含まれません。

必須: いいえ

タイプ: String

デフォルト値: /

tokenRefreshDelaySec

バックグラウンドの更新に使用されます。バックグラウンドトークンが更新されるまでの最大時間 (秒)。

必須: いいえ

タイプ: Integer

デフォルト値: 88

許容される最小値: 88

許容される最大値: 300 (5 分)

AWS WAF モバイル SDK の仕組み

モバイル SDK は、トークンの取得と利用のために使用できる設定可能なトークンプロバイダーを提供します。トークンプロバイダーは、許可するリクエストが正規の顧客からのものであることを検証します。 AWS 保護対象のリソースにリクエストを送信するときは AWS WAF、そのトークンを Cookie に含めてリクエストを検証します。トークン cookie は手動で処理することも、トークンプロバイダーに処理させることもできます。

このセクションでは、モバイル SDK に含まれるクラス、プロパティ、およびメソッド間のインタラクションについて説明します。SDK の仕様については、「AWS WAF モバイル SDK 仕様」を参照してください。

トークンの取得とキャッシュ

モバイルアプリケーションでトークンプロバイダーインスタンスを作成するときに、トークンとトークンの取得を管理する方法を設定します。主に、アプリケーションのウェブリクエストで使用するための、有効で期限切れになっていないトークンを維持する方法を選択できます。

- [Background refresh enabled] (バックグラウンド更新が有効) - これがデフォルトのトランスコードプリセットです。トークンプロバイダーは、バックグラウンドでトークンを自動的に更新し、キャッシュします。バックグラウンド更新が有効になっている場合、`getToken()` を呼び出すと、オペレーションはキャッシュされたトークンを取得します。

  トークンプロバイダーは、設定可能な間隔でトークンの更新を実行します。これにより、アプリケーションがアクティブな間、期限切れでないトークンは常にキャッシュ内で利用可能な状態となります。アプリケーションが非アクティブ状態の間、バックグラウンドの更新が一時停止されます。詳細については、「アプリケーションが非アクティブ状態になった後のトークンの取得」を参照してください。
- [Background refresh disabled] (バックグラウンド更新が無効) - バックグラウンドトークンの更新を無効にして、オンデマンドでのみトークンを取得できます。オンデマンドで取得されたトークンはキャッシュされません。また、必要に応じて複数のトークンを取得できます。各トークンは、取得する他のトークンとは独立しており、有効期限を計算するために使用される独自のタイムスタンプを備えています。

  バックグラウンド更新が無効になっている場合のトークンの取得には、次の選択肢があります。

  - **`getToken()`**— `getToken()` バックグラウンド更新を無効にして呼び出しを行うと、呼び出しはから新しいトークンを同期的に取得します。 AWS WAFこれは、メインスレッドで呼び出すと、アプリケーションの応答性に影響する可能性のあるブロッキング呼び出しである場合があります。
  - **`onTokenReady(WAFTokenResultCallback)`** - この呼び出しは、新しいトークンを非同期的に取得し、トークンの準備ができたときに提供された結果コールバックをバックグラウンドスレッドで呼び出します。

トークンプロバイダーが失敗したトークンの取得を再試行する方法

トークンプロバイダーは、取得に失敗したときにトークンの取得を自動的に再試行します。再試行は、開始の再試行の待ち時間が 100 ミリ秒のエクスポネンシャルバックオフを使用して最初に実行されます。エクスポネンシャル再試行の詳細については、「AWSでのエラー再試行とエクスポネンシャルバックオフ」を参照してください。

再試行回数が設定された maxRetryCount に達すると、トークンプロバイダーは、トークン取得のタイプに応じて、試行を停止するか、maxErrorTokenRefreshDelayMsec ミリ秒ごとの試行に切り替えます。

- **onTokenReady()** – トークンプロバイダーは、試行間の待機時間を maxErrorTokenRefreshDelayMsec ミリ秒に切り替えて、トークン取得の試行を続行します。
- バックグラウンド更新 – トークンプロバイダーは、試行間の待機時間を maxErrorTokenRefreshDelayMsec ミリ秒に切り替えて、トークンの取得の試行を続行します。
- バックグラウンド更新が無効になっている場合のオンデマンド **getToken()** 呼び出し – トークンプロバイダーはトークンの取得の試行を停止し、前のトークン値を返します。前のトークンがない場合は null 値を返します。

アプリケーションが非アクティブ状態になった後のトークンの取得

バックグラウンド更新は、アプリケーションがアプリケーションタイプについてアクティブであるとみなされる場合にのみ実行されます。

- iOS – バックグラウンド更新は、アプリケーションがフォアグラウンドにあるときに実行されます。
- Android - バックグラウンドの更新は、アプリケーションがフォアグラウンドまたはバックグラウンドのいずれにあるかにかかわらず、アプリケーションが閉じられていないときに実行されます。

アプリケーションが設定された tokenRefreshDelaySec 秒より長くバックグラウンド更新をサポートしない状態のままである場合、トークンプロバイダーはバックグラウンド更新を一時停止します。例えば、iOS アプリケーションの場合、tokenRefreshDelaySec が 300 で、300 秒以を超える時間にわたって、アプリケーションが閉じられていたり、バックグラウンド状態になっていたりすると、トークンプロバイダーはトークンの更新を停止します。アプリケーションがアクティブな状態に戻ると、トークンプロバイダーは自動的にバックグラウンド更新を再開します。

アプリケーションがアクティブ状態に戻ったら、トークンプロバイダーが新しいトークンを取得してキャッシュしたときに通知を受け取ることができるように onTokenReady() を呼び出します。キャッシュには最新の有効なトークンがまだ含まれていない可能性があるため、単に getToken() を呼び出さないでください。

AWS WAF モバイル SDK のコードの記述

このセクションでは、モバイル SDK を使用するためのコード例を提供します。

トークンプロバイダーの初期化とトークンの取得

設定オブジェクトを使用して、トークンプロバイダーインスタンスを開始します。その後、使用可能なオペレーションを使用してトークンを取得できます。必要なコードの基本コンポーネントを次に示します。

iOS

```
let url: URL = URL(string: "Web ACL integration URL")!
let configuration = WAFConfiguration(applicationIntegrationUrl: url, domainName:
 "Domain name")
let tokenProvider = WAFTokenProvider(configuration)

//onTokenReady can be add as an observer for
 UIApplication.willEnterForegroundNotification
self.tokenProvider.onTokenReady() { token, error in
 if let token = token {
 //token available
 }

 if let error = error {
 //error occurred after exhausting all retries
 }
}

//getToken()
let token = tokenProvider.getToken()
```

Android

Java の例：

```
String applicationIntegrationURL = "Web ACL integration URL";
//Or
URL applicationIntegrationURL = new URL("Web ACL integration URL");

String domainName = "Domain name";

WAFConfiguration configuration =
 WAFConfiguration.builder().applicationIntegrationURL(applicationIntegrationURL).domainName(
WAFTokenProvider tokenProvider = new WAFTokenProvider(Application context,
 configuration);
```

```
// implement a token result callback
WAFTokenResultCallback callback = (wafToken, error) -> {
 if (wafToken != null) {
 // token available
 } else {
 // error occurred in token refresh
 }
};

// Add this callback to application creation or activity creation where token will
 be used
tokenProvider.onTokenReady(callback);

// Once you have token in token result callback
// if background refresh is enabled you can call getToken() from same tokenprovider
 object
// if background refresh is disabled you can directly call getToken()(blocking call)
 for new token
WAFToken token = tokenProvider.getToken();
```

Kotlin の例:

```
import com.amazonaws.waf.mobilesdk.token.WAFConfiguration
import com.amazonaws.waf.mobilesdk.token.WAFTokenProvider

private lateinit var wafConfiguration: WAFConfiguration
private lateinit var wafTokenProvider: WAFTokenProvider

private val WAF_INTEGRATION_URL = "Web ACL integration URL"
private val WAF_DOMAIN_NAME = "Domain name"

fun initWaf() {
 // Initialize the tokenprovider instance
 val applicationIntegrationURL = URL(WAF_INTEGRATION_URL)
 wafConfiguration =
  WAFConfiguration.builder().applicationIntegrationURL(applicationIntegrationURL)
   .domainName(WAF_DOMAIN_NAME).backgroundRefreshEnabled(true).build()
 wafTokenProvider = WAFTokenProvider(getApplication(), wafConfiguration)

  // getToken from tokenprovider object
  println("WAF: "+ wafTokenProvider.token.value)

  // implement callback for where token will be used
```

```
  wafTokenProvider.onTokenReady {
   wafToken, sdkError ->
  run {
   println("WAF Token:" + wafToken.value)
  }
 }
}
```

SDK による HTTP リクエストでのトークン cookie の提供の許可

`setTokenCookie` が `TRUE` である場合、トークンプロバイダーは、`tokenCookiePath` で指定されたパスの下のすべての場所に対するウェブリクエストにトークン cookie を含めます。デフォルトでは、`setTokenCookie` は `TRUE`、`tokenCookiePath` は `/` です。

トークン cookie のパスを指定することで、トークン cookie を含むリクエストの範囲を絞り込むことができます (例: `/web/login`)。これを行う場合は、他のパスに送信するリクエストのトークンが AWS WAF ルールで検査されていないことを確認します。`AWSManagedRulesACFPRuleSet` ルールグループを使用する場合、アカウントの登録パスと作成パスを設定すると、ルールグループはそれらのパスに送信されるリクエスト内のトークンをチェックします。詳細については、「ACFP マネージドルールグループをウェブ ACL に追加」を参照してください。同様に、`AWSManagedRulesATPRuleSet` ルールグループを使用する場合は、ログインパスを設定し、ルールグループはそのパスに送信されるリクエストのトークンをチェックします。詳細については、「ATP マネージドルールグループをウェブ ACL に追加」を参照してください。

iOS

`setTokenCookie` が の場合`TRUE`、トークンプロバイダーは AWS WAF トークンを に保存`HTTPCookieStorage.shared`し、 で指定したドメインへのリクエストに Cookie を自動的に含めます`WAFConfiguration`。

```
let request = URLRequest(url: URL(string: domainEndpointUrl)!)
//The token cookie is set automatically as cookie header
let task = URLSession.shared.dataTask(with: request) { data, urlResponse, error  in
}.resume()
```

Android

`setTokenCookie` が の場合`TRUE`、トークンプロバイダーはアプリケーション全体で共有されている`CookieHandler`インスタンスに AWS WAF トークンを保存します。トークンプロバイダーは、`WAFConfiguration` で指定したドメインへのリクエストに cookie を自動的に含めます。

Java の例：

```
URL url = new URL("Domain name");
//The token cookie is set automatically as cookie header
HttpsURLConnection connection = (HttpsURLConnection) url.openConnection();
connection.getResponseCode();
```

Kotlin の例:

```
val url = URL("Domain name")
//The token cookie is set automatically as cookie header
val connection = (url.openConnection() as HttpsURLConnection)
connection.responseCode
```

CookieHandler デフォルトインスタンスが既に初期化されている場合、トークンプロバイダーは、それを使用して cookie を管理します。そうでない場合、トークンプロバイダーは AWS WAF トークンを使用して新しいCookieManagerインスタンスを初期化CookiePolicy.ACCEPT_ORIGINAL_SERVERし、この新しいインスタンスを のデフォルトインスタンスとして設定しますCookieHandler。

次のコードは、アプリケーションで使用できない場合に cookie マネージャーと cookie ハンドラーを SDK が初期化する方法を示しています。

Java の例：

```
CookieManager cookieManager = (CookieManager) CookieHandler.getDefault();
if (cookieManager == null) {
 // Cookie manager is initialized with CookiePolicy.ACCEPT_ORIGINAL_SERVER
 cookieManager = new CookieManager();
 CookieHandler.setDefault(cookieManager);
}
```

Kotlin の例:

```
var cookieManager = CookieHandler.getDefault() as? CookieManager
if (cookieManager == null) {
 // Cookie manager is initialized with CookiePolicy.ACCEPT_ORIGINAL_SERVER
 cookieManager = CookieManager()
 CookieHandler.setDefault(cookieManager)
}
```

HTTP リクエストにおけるトークン cookie の手動による提供

`setTokenCookie` を FALSE に設定した場合、保護されたエンドポイントに対するリクエストで、cookie HTTP リクエストヘッダーとしてトークン cookie を手動で提供する必要があります。次のコードは、これを実行する方法を説明しています。

iOS

```
var request = URLRequest(url: wafProtectedEndpoint)
request.setValue("aws-waf-token=token from token provider", forHTTPHeaderField:
 "Cookie")
request.httpShouldHandleCookies = true
URLSession.shared.dataTask(with: request) { data, response, error in }
```

Android

Java の例 :

```
URL url = new URL("Domain name");
HttpsURLConnection connection = (HttpsURLConnection) url.openConnection();
String wafTokenCookie = "aws-waf-token=token from token provider";
connection.setRequestProperty("Cookie", wafTokenCookie);
connection.getInputStream();
```

Kotlin の例:

```
val url = URL("Domain name")
val connection = (url.openConnection() as HttpsURLConnection)
val wafTokenCookie = "aws-waf-token=token from token provider"
connection.setRequestProperty("Cookie", wafTokenCookie)
connection.inputStream
```

CAPTCHAChallengeの および AWS WAF

AWS WAF ルールの検査基準に一致するウェブリクエストに対して CAPTCHAまたは Challenge アクションを実行するようにルールを設定できます。CAPTCHA パズルとブラウザチャレンジを JavaScript ローカルで実行するようにクライアントアプリケーションをプログラムすることもできます。

CAPTCHA パズルとサイレントチャレンジは、ブラウザが HTTPS エンドポイントにアクセスしている場合にのみ実行できます。トークンを取得するには、ブラウザクライアントが安全なコンテキストで実行されている必要があります。

- CAPTCHA – エンドユーザーは、人間がリクエストを送信していることを証明するために CAPTCHA パズルを解決する必要があります。CAPTCHA パズルは、人間にとっては非常に簡単かつ短時間で完了に成功できる一方、コンピュータにとっては完了に成功、あるいは有意義な成功率でランダムに完了させることを困難にすることを意図しています。

  ウェブ ACL ルールでは、Blockアクションが正当なリクエストをあまりにも多く停止し、すべてのトラフィックを経由させると、ボットからのリクエストなど、許容できないほど高レベルの不要なリクエストが発生する場合に CAPTCHA が一般的に使用されます。ルールアクションの動作の詳細については、「」を参照してくださいAWS WAFCAPTCHA および Challenge ルールアクションの仕組み。

  クライアントアプリケーション統合 APIs で CAPTCHA パズル実装をプログラムすることもできます。これを行うと、クライアントアプリケーションでパズルの動作と配置をカスタマイズできます。詳細については、「AWS WAF クライアントアプリケーション統合」を参照してください。

- Challenge – クライアントセッションがボットではなくブラウザであることを確認する必要があるサイレントチャレンジを実行します。検証は、エンドユーザーの関与なしでバックグラウンドで実行されます。これは、CAPTCHA パズルでエンドユーザーのエクスペリエンスに悪影響を与えることなく、無効だと思われるクライアントを検証する場合に適したオプションです。ルールアクションの動作の詳細については、「」を参照してくださいAWS WAFCAPTCHA および Challenge ルールアクションの仕組み。

  Challenge ルールアクションは、「AWS WAF クライアントアプリケーション統合」で説明されている、インテリジェントな脅威に対応したクライアント統合 API が実行するチャレンジと似ています。

> **Note**
>
> CAPTCHA または Challenge ルールアクションを 1 つのルールで使用、あるいはルールグループでルールアクションのオーバーライドとして使用すると、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

すべてのルールアクションオプションの説明については、「」を参照してくださいルールアクション。

トピック

- AWS WAF キャプチャパズル
- AWS WAFCAPTCHA および Challenge ルールアクションの仕組み
- CAPTCHA および Challenge アクションを使用するベストプラクティス

AWS WAF キャプチャパズル

AWS WAF ユーザーが人間であることを確認するよう求める標準的な CAPTCHA 機能を提供します。CAPTCHA は、「Completely Automated Public Turing test to tell Computers and Humans Apart」(コンピューターと人間を区別する完全自動公開チューリングテスト) の略です。CAPTCHA パズルは、人間がリクエストを送信していることを検証し、ウェブスクレイピング、クレデンシャルスタッフィング、スパムなどのアクティビティを防ぐように設計されます。CAPTCHA パズルでは不要なリクエストをすべて取り除くことはできません。多くのパズルが機械学習と人工知能を使って解決されています。CAPTCHA を回避するため、一部の組織は人間が介入した自動化技術で補完しています。それにもかかわらず、CAPTCHA は洗練度の低いボットトラフィックを防ぎ、大規模な運営に必要なリソースを増やすための便利なツールとして役割を果たし続けています。

AWS WAF CAPTCHAパズルをランダムに生成して順番に並べることで、ユーザーにユニークな課題が提示されるようにします。 AWS WAF オートメーション技術に対抗できるよう、定期的に新しい種類やスタイルのパズルが追加されています。パズルに加えて、 AWS WAF CAPTCHA スクリプトはクライアントに関するデータを収集して、タスクが人間によって完了されていることを確認し、リプレイ攻撃を防ぎます。

各 CAPTCHA パズルには、エンドユーザーが新しいパズルをリクエストしたり、音声パズルと視覚パズルを切り替えたり、追加の指示にアクセスしたり、パズルの解答を送信したりするための標準的なコントロールセットが含まれています。すべてのパズルには、スクリーンリーダー、キーボードコントロール、コントラスト色のサポートが含まれています。

AWS WAF CAPTCHA パズルは Web コンテンツ・アクセシビリティ・ガイドライン (WCAG) の要件を満たしています。詳細については、World Wide Web Consortium (W3C) ウェブサイトの「Web Content Accessibility Guidelines (WCAG) Overview」(Web Content Accessibility Guidelines (WCAG) の概要) を参照してください。

トピック

キャプチャパズル言語のサポート

CAPTCHA パズルは、クライアントのブラウザー言語、またはブラウザー言語がサポートされていない場合は英語で書かれた指示から始まります。パズルには、ドロップダウンメニューから代替言語オプションが表示されます。

ページ下部のヘッドフォンアイコンを選択すると、音声による指示に切り替えることができます。音声版のパズルでは、ユーザーがテキストボックスに入力すべきテキストについて、バックグラウンドノイズが重なった音声による指示が表示されます。

次の表は、CAPTCHA パズルの指示書で選択できる言語と、各選択での音声サポートの一覧です。

AWS WAF CAPTCHA パズルがサポートする言語

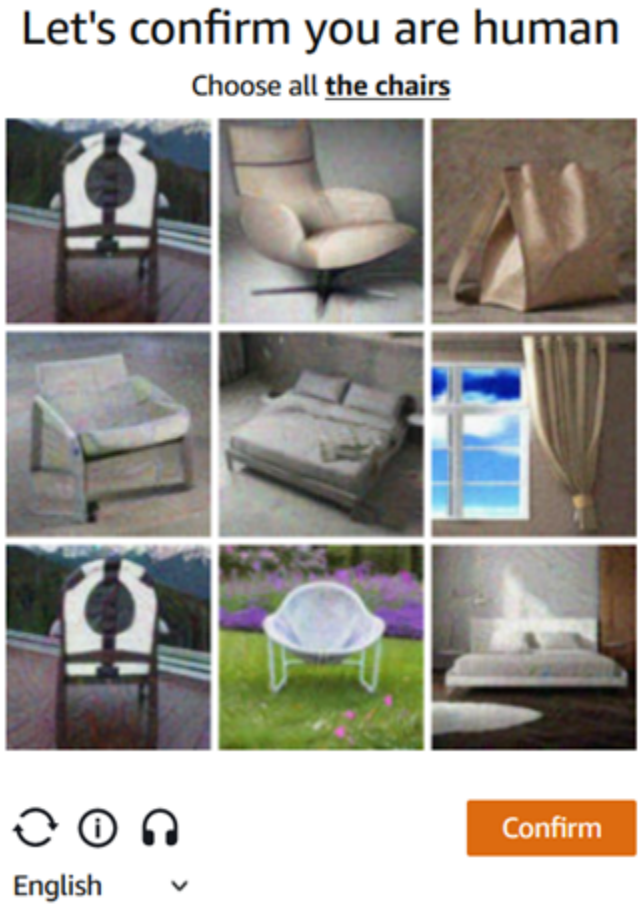
書面による指示サポート	ロケールコード	オーディオインストラクションサポート
アラビア語	ar-SA	アラビア語
簡体字中国語	zh-CN	英語での音声
オランダ語	nl-NL	オランダ語
英語	en-US	英語
フランス語	fr-FR	フランス語
ドイツ語	de-DE	ドイツ語
イタリア語	it-IT	イタリア語
日本語	ja-JP	英語での音声

書面による指示サポート	ロケールコード	オーディオインストラクションサポート
ブラジルポルトガル語	pt-BR	ブラジルポルトガル語
スペイン語	es-ES	スペイン語
トルコ語	tr-TR	トルコ語

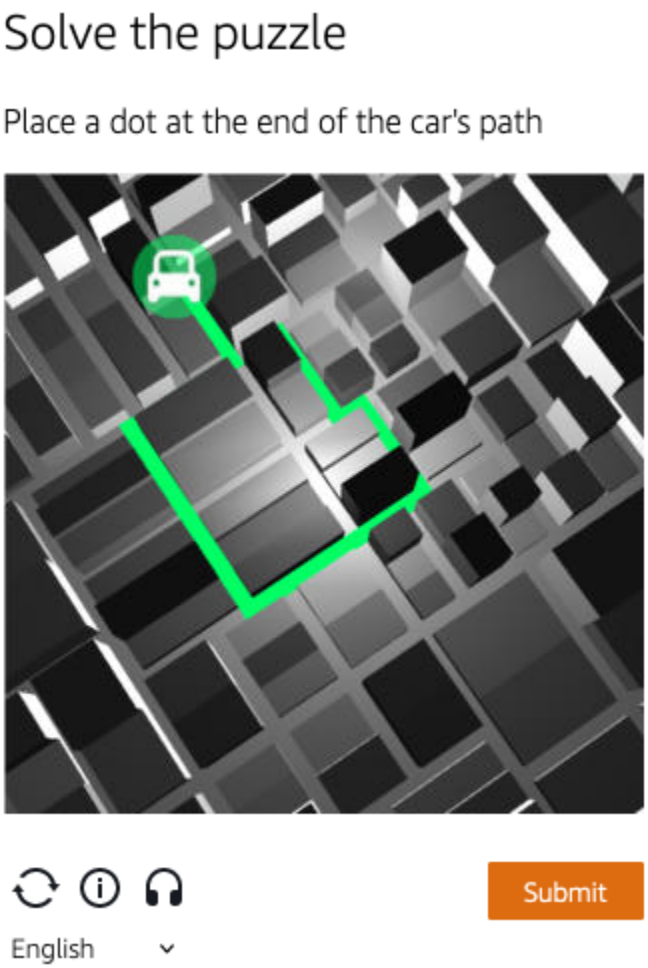
キャプチャパズルの例

一般的なビジュアル CAPTCHA パズルでは、ユーザーが 1 つまたは複数の画像を理解して操作できることを示すインタラクションが必要です。

以下のスクリーンショットは、ピクチャーグリッドパズルの例を示しています。このパズルでは、特定の種類のオブジェクトを含むグリッド内のすべての画像を選択する必要があります。

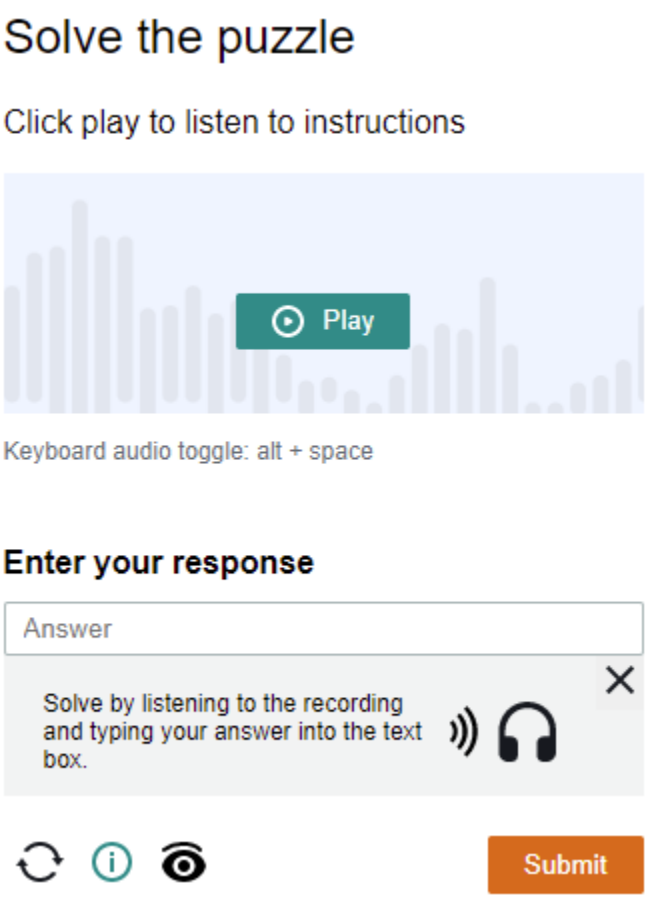


次のスクリーンショットは、図面上で車の進路の終点を特定する必要があるパズルの例を示しています。



オーディオパズルでは、ユーザーがテキストボックスに入力する必要があるテキストに関する音声による指示が背景に重なって表示されます。

次のスクリーンショットは、選択した音声パズルの表示内容を示しています。



AWS WAFCAPTCHA および Challenge ルールアクションの仕組み

AWS WAF CAPTCHA および Challengeは標準のルールアクションであるため、比較的簡単に実装できます。どちらかを使用するには、検査するリクエストを識別するルールの検査基準を作成し、2 つのルールアクションのうち 1 つを指定します。ルールアクションのオプションの一般的な情報については、「ルールアクション」を参照してください。

サーバー側からサイレントチャレンジと CAPTCHA パズルを実装することに加えて、サイレントチャレンジを JavaScript および iOS および Android クライアントアプリケーションに統合したり、JavaScript クライアントで CAPTCHA パズルをレンダリングしたりできます。これらの統合を使用すると、エンドユーザーがより良いパフォーマンスと CAPTCHA パズルエクスペリエンスを享受できるだけでなく、ルールアクションやインテリジェントな脅威の軽減を目的としたルールグループの使用に関連するコストを削減できます。これらのパラメータの詳細については、「AWS WAF クライアントアプリケーション統合」を参照してください。料金に関する情報については、[AWS WAF の料金]を参照してください。

トピック

CAPTCHA および Challenge アクション動作

CAPTCHAChallengeウェブリクエストがルールとアクションのインスペクション基準に一致する場合、 AWS WAF トークンの状態とイミュニティタイムの設定に従ってリクエストの処理方法を決定します。 AWS WAF また、リクエストが CAPTCHA パズルまたはチャレンジスクリプトのインタースティシャルを処理できるかどうかも考慮されます。スクリプトは HTML コンテンツとして処理されるように設計されており、HTML コンテンツを想定しているクライアントによってのみ適切に処理されることが可能です。

> **Note**
>
> CAPTCHA または Challenge ルールアクションを 1 つのルールで使用、あるいはルールグループでルールアクションのオーバーライドとして使用すると、追加料金が請求されます。詳細については、「AWS WAF の料金」を参照してください。

アクションがウェブリクエストを処理する方法

AWS WAF CAPTCHAor Challenge アクションをウェブリクエストに次のように適用します。

- 有効なトークン — AWS WAF Count これをアクションと同様に処理します。 AWS WAF ルールアクションに設定したラベルとリクエストのカスタマイズを適用し、ウェブ ACL の残りのルールを使用してリクエストの評価を続行します。
- トークンがない、無効、または期限切れのトークン — リクエストのウェブ ACL AWS WAF 評価を中止し、目的の宛先への送信をブロックします。

  AWS WAF ルールアクションタイプに従って、レスポンスを生成してクライアントに返送します。

  - Challenge – AWS WAF はレスポンスに次のものが含まれます。
    - 値が `challenge` のヘッダー `x-amzn-waf-action`。

> **Note**
>
> このヘッダーは、 JavaScript クライアントブラウザーで実行されるアプリケーションでは使用できません。詳細については、次のセクションを参照してください。

- HTTP ステータスコード `202 Request Accepted`。
- `Accept`リクエストに値がのヘッダーが含まれている場合`text/html`、JavaScript レスポンスにはチャレンジスクリプトを含むページインタースティシャルが含まれます。

- CAPTCHA— AWS WAF レスポンスには以下が含まれます。
  - 値が `captcha` のヘッダー `x-amzn-waf-action`。

> **Note**
>
> このヘッダーは、 JavaScript クライアントブラウザーで実行されるアプリケーションでは使用できません。詳細については、次のセクションを参照してください。

  - HTTP ステータスコード `405 Method Not Allowed`。
  - `Accept`リクエストに値がのヘッダーが含まれている場合`text/html`、レスポンスには CAPTCHA JavaScript スクリプトを含むページインタースティシャルが含まれます。

ウェブ ACL またはルールレベルでトークンの有効期限が切れるタイミングを設定するには、「タイムスタンプの有効期限： AWS WAF トークンのイミュニティ時間」を参照してください。

ヘッダーは、 JavaScript クライアントブラウザーで実行されるアプリケーションでは使用できません。

がクライアントのリクエストに CAPTCHA AWS WAF またはチャレンジレスポンスで応答する場合、クロスオリジンリソースシェアリング (CORS) ヘッダーは含まれません。CORS ヘッダーは、アプリケーションが使用できるドメイン、HTTP メソッド、および HTTP ヘッダーをクライアント Web ブラウザーに伝えるアクセス制御ヘッダーのセットです。 JavaScriptCORS ヘッダーがないと、 JavaScript クライアントブラウザーで実行されているアプリケーションには HTTP ヘッダーへのアクセスが許可されないため、`x-amzn-waf-action`および応答で提供されるヘッダーを読み取ることができません。CAPTCHA Challenge

チャレンジと CAPTCHA インタースティシャルの機能

チャレンジインタースティシャルが実行されると、クライアントが応答に成功した後、まだトークンがない場合、インタースティシャルがトークンを初期化します。その後、チャレンジ解決のタイムスタンプでトークンを更新します。

CAPTCHA インタースティシャルを実行するとき、クライアントがまだトークンを持っていない場合、CAPTCHA インタースティシャルはまずチャレンジスクリプトを呼び出し、ブラウザにチャレンジしてトークンを初期化します。その後、インタースティシャルは CAPTCHA パズルを実行します。エンドユーザーがパズルの完成に成功すると、インタースティシャルはトークンを CAPTCHA 解決のタイムスタンプで更新します。

いずれの場合も、クライアントが応答に成功してスクリプトがトークンを更新した後、スクリプトは更新されたトークンを使用して元のウェブリクエストを再送信します。

AWS WAF トークンの処理方法を設定できます。詳細については、「AWS WAF ウェブリクエストトークン」を参照してください。

ログとメトリクスの CAPTCHA および Challenge アクション

Challenge および CAPTCHA アクションは、Count のように終了しない場合もあれば、Block のように終了する場合もあります。結果は、リクエストがアクションタイプの有効期限が切れていない有効なトークンがあるかどうかによって異なります。

- 有効なトークン — アクションが有効なトークンを見つけてもリクエストをブロックしない場合、AWS WAF 次のようにメトリクスとログをキャプチャします。
  - `CaptchaRequests` および `RequestsWithValidCaptchaToken` または `ChallengeRequests` および `RequestsWithValidChallengeToken` のいずれかのメトリクスを増分します。
  - CAPTCHA または Challenge のアクションで `nonTerminatingMatchingRules` エントリとして一致を口グに記録します。次のリストは、CAPTCHA アクションを使ったこの一致タイプにおける口グのセクションを示しています。

```
"nonTerminatingMatchingRules": [
{
  "ruleId": "captcha-rule",
  "action": "CAPTCHA",
  "ruleMatchDetails": [],
  "captchaResponse": {
    "responseCode": 0,
    "solveTimestamp": 1632420429
  }
```

```
    }
  ]
```

- トークンが見つからない、無効である、または期限切れのトークン — トークンがないか無効であるためにアクションがリクエストをブロックすると、 AWS WAF 次のようにメトリクスとログがキャプチャされます。
  - CaptchaRequests または ChallengeRequests のメトリクスを増分させます。
  - 一致を HTTP 405 ステータスコードを含む CaptchaResponse エントリ、あるいは HTTP 202 ステータスコードを含む ChallengeResponse エントリとしてログ記録します。ログは、リクエストにトークンが不足しているか、トークンの有効期限が切れているか示します。ログには、CAPTCHA AWS WAF インタースティシャルページをクライアントに送信したのか、クライアントブラウザーにサイレントチャレンジを送信したのかも示されます。次のリストは、CAPTCHA アクションを含むこのタイプの一致におけるログのセクションを示しています。

```
    "terminatingRuleId": "captcha-rule",
    "terminatingRuleType": "REGULAR",
    "action": "CAPTCHA",
    "terminatingRuleMatchDetails": [],
    ...
    "responseCodeSent": 405,
    ...
    "captchaResponse": {
      "responseCode": 405,
      "solveTimestamp": 0,
      "failureReason": "TOKEN_MISSING"
    }
```

ログの詳細については、 AWS WAF を参照してください。AWS WAF ウェブ ACL トラフィックのログ記録

AWS WAF メトリクスについて詳しくは、を参照してくださいAWS WAF メトリクスとディメンション。

ルールアクションのオプションについては、「ルールアクション」を参照してください。

CAPTCHA および Challenge アクションを使用するベストプラクティス

このセクションのガイダンスに従って、 AWS WAF CAPTCHA またはチャレンジを計画および実装します。

CAPTCHA およびチャレンジの実装の計画

ウェブサイトの使用状況、保護するデータの機密性、リクエストのタイプに基づいて、CAPTCHA パズルまたはサイレントチャレンジを配置する場所を決定します。必要に応じてパズルを提示できるように、CAPTCHA に適用するリクエストを選択します。ただし、役に立たずにユーザーエクスペリエンスを低下させる可能性がある場合、パズルの提示を控えてください。Challenge アクションを使用して、エンドユーザーへの影響が少ないサイレントチャレンジを実行しますが、リクエストが JavaScript 有効なブラウザから送信されたことを確認するのに役立ちます。

CAPTCHA パズルとサイレントチャレンジは、ブラウザが HTTPS エンドポイントにアクセスしている場合にのみ実行できます。トークンを取得するには、ブラウザクライアントが安全なコンテキストで実行されている必要があります。

クライアントで CAPTCHA パズルやサイレントチャレンジを実行する場所を決定

CSS や画像のリクエストなど、CAPTCHA による影響を受けたくないリクエストを特定します。CAPTCHA は必要な場合にのみ使用してください。たとえば、ログイン時に CAPTCHA チェックを行う予定で、ユーザーが常にログインから別の画面に直接ダイレクトされる場合、2 つ目の画面で CAPTCHA チェックを要求することはおそらく不要であり、ユーザーエクスペリエンスを低下させる可能性があります。

`GET` `text/html` がリクエストに応答して CAPTCHA パズルとサイレントチャレンジ AWS WAF のみを送信するように Challengeと CAPTCHAを設定します。`POST`リクエスト、クロスオリジンリソース共有 (CORS) プリフライト`OPTIONS`要求、またはその他の非要求`GET`リクエストタイプに応答しても、パズルもチャレンジも実行することはできません。他のリクエストに対するブラウザの動作は異なる場合があり、インタースティシャルを適切に処理できない可能性があります。

クライアントが HTML を受け入れられても、CAPTCHA またはチャレンジインタースティシャルを処理できない場合があります。たとえば、小さな iFrame を持つウェブページ上のウィジェットは HTML を受け入れますが、CAPTCHA の表示またはその処理ができない場合があります。このようなタイプのリクエストのルールアクションは、HTML を受け入れないリクエストと同じように、設定しないでください。

以前に取得したトークンの確認に CAPTCHA または Challenge を使用

ルールアクションは、正規ユーザーが常にトークンを持っている必要がある場所で、有効なトークンの存在を確認する場合にのみ使用できます。このような状況では、リクエストがインタースティシャルを処理できるかどうかは関係ありません。

例えば、 JavaScript クライアントアプリケーションの CAPTCHA API を実装し、保護されたエンドポイントに最初のリクエストを送信する直前にクライアントで CAPTCHA パズルを実行する場合、

最初のリクエストにはチャレンジと CAPTCHA の両方に有効なトークンが常に含まれている必要があります。 JavaScript クライアントアプリケーション統合の詳細については、「」を参照してくださいAWS WAF JavaScript 統合。

この状況では、ウェブ ACL に、この最初の呼び出しと一致するルールを追加し、Challenge または CAPTCHA ルールアクションでルールを設定することができます。ルールが正規のエンドユーザーとブラウザに一致すると、アクションは有効なトークンを検索します。したがって、アクションがリクエストをブロックしたり、リクエストに応答してチャレンジや CAPTCHA パズルを送信したりすることはありません。ルールアクションの仕組みの詳細については、「CAPTCHA および Challenge アクション動作」を参照してください。

CAPTCHA および Challenge で機密性のある非 HTML データを保護する

次のアプローチで、API などの機密性の高い非 HTML データに CAPTCHA および Challenge 保護を使用できます。

1. HTML レスポンスを受け取り、機密性の高い HTML 以外のデータに対するリクエストの近くで実行されるリクエストを特定します。
2. HTML のリクエストと照合し、機密データのリクエストと照合する CAPTCHA または Challenge ルールを記述します。
3. CAPTCHA および Challenge イミュニティ時間の設定を調整し、通常のユーザーインタラクションで、クライアントが HTML リクエストから取得するトークンが、機密データのリクエストにおいて利用可能で有効期限が切れないようします。チューニングの情報については、「タイムスタンプの有効期限： AWS WAF トークンのイミュニティ時間」を参照してください。

機密データのリクエストが CAPTCHA または Challenge ルールに一致すると、クライアントが以前のパズルまたはチャレンジからの有効なトークンをまだ持っている場合、そのリクエストはブロックされません。トークンが利用できない、あるいはタイムスタンプが有効期限が切れている場合、機密データをアクセスするリクエストは失敗します。ルールアクションの仕組みの詳細については、「CAPTCHA および Challenge アクション動作」を参照してください。

CAPTCHA および Challenge を使用して既存ルールの調整

既存のルールを確認し、変更するか追加するかを確認します。一般的なシナリオをいくつか次に示します。

- トラフィックをブロックするレートベースルールがあるものの、正規ユーザーのブロックを避けるためにレート制限を比較的高く維持する場合、ブロックルールの後に 2 つ目のレートベースルー

ルを追加することを検討してください。2 つ目のルールにブロッキングルールよりも低い制限を設定し、ルールアクションを CAPTCHA または Challenge に設定します。ブロッキングルールは、高すぎるレートでリクエストを受け取らないように引き続きブロックします。新しいルールは、ほとんどの自動化トラフィックをさらに低いレートでブロックします。レートベースルールの詳細については、「レートベースのルールステートメント」を参照してください。

- リクエストをブロックするマネージドルールグループがある場合、一部またはすべてのルールの動作を Block から CAPTCHA または Challenge に切り替えることができます。これを行うには、マネージドルールグループの設定で、ルールアクション設定をオーバーライドします。ルールアクションのオーバーライドの情報については、「ルールグループのルールアクションの上書き」を参照してください。

デプロイする前に CAPTCHA およびチャレンジの実装をテストしてください

すべての新機能については、the section called “保護のテストとチューニング” のガイダンスに従ってください。

テストのとき、トークンタイムスタンプの有効期限要件を確認し、ウェブ ACL およびルールレベルのイミュニティ時間を設定して、ウェブサイトへのアクセスを制御および顧客に優れたエクスペリエンスを提供することのバランスが適切に維持できるようにします。詳細については、「タイムスタンプの有効期限： AWS WAF トークンのイミュニティ時間」を参照してください。

AWS WAF ウェブ ACL トラフィックのログ記録

ログ記録を有効にして、ウェブ ACL で分析されるトラフィックに関する詳細情報を取得できます。ログに記録された情報には、 がリソースから AWS ウェブリクエストを AWS WAF 受信した時間、リクエストに関する詳細情報、およびリクエストが一致したルールに関する詳細が含まれます。ウェブ ACL ログは、Amazon CloudWatch Logs ロググループ、Amazon Simple Storage Service (Amazon S3) バケット、または Amazon Data Firehose 配信ストリームに送信できます。

その他のデータ収集および分析オプション

ログ記録に加えて、データ収集と分析のために以下のオプションを有効にすることができます。

- Amazon Security Lake – ウェブ ACL データを収集するように Security Lake を設定できます。Security Lake は、正規化、分析、管理のためにさまざまなソースからログとイベントデータを収集します。このオプションの詳細については、「Amazon Security Lake ユーザーガイド」の「Amazon Security Lake とは」および「 AWS のサービスからデータを収集する」を参照してください。

AWS WAF は、このオプションの使用に対して課金しません。料金情報については、「Amazon Security Lake ユーザーガイド」の「Security Lake の料金」および「Security Lake の料金の決定方法」を参照してください。

- リクエストサンプリング – 評価されるウェブリクエストをサンプリングするようにウェブ ACL を設定して、アプリケーションが受信するトラフィックのタイプを把握できます。このオプションについては、「ウェブリクエストのサンプルの表示」を参照してください。

> **Note**
>
> ウェブ ACL ログ記録設定は AWS WAF ログにのみ影響します。特に、ログ記録用に編集されたフィールド設定は、リクエストサンプリングや Security Lake データ収集には影響しません。Security Lake データ収集は、Security Lake サービスを通じて完全に設定されます。サンプリングされたリクエストからフィールドを除外する唯一の方法は、ウェブ ACL のサンプリングを無効にすることです。

トピック

ウェブ ACL トラフィックのログ記録の料金に関する情報

ウェブ ACL トラフィックに関する情報のログ記録について、各ログの宛先タイプに関連するコストに応じて請求されます。これらの料金は、 AWS WAFの使用料に加算されます。コストは、選択した宛先タイプやログに記録するデータ量などの要因によって異なる場合があります。

各ログ記録の宛先タイプの料金に関する情報へのリンクを次に示します。

- CloudWatch ログ — 料金は、自動販売によるログ配信の料金です。Amazon CloudWatch ログの料金表を参照してください。「有料利用枠」で「ログ」タブを選択し、「ベンダーログ」で「CloudWatch ログへの配信」に関する情報を参照してください。

- Amazon S3 バケット — Amazon S3 の料金は、Amazon S3 CloudWatch バケットへのログ配信と Amazon S3 の使用料金を合わせたものです。
  - Amazon S3 については、「Amazon S3 の料金」を参照してください。
  - Amazon S3 CloudWatch へのログベンダーによるログ配信については、「Amazon CloudWatch ログ料金表」を参照してください。[Paid Tier] (有料の階層) で [Logs] (ログ) タブを選択し、[Vended Logs] (公開ログ) で [Delivery to S3] (S3 に配信) の情報を確認します。
- Firehose — Amazon データFirehose 料金表を参照してください。

AWS WAF 料金については、「料金表」を参照してくださいAWS WAF 。

AWS WAF ロギング先

このセクションでは、 AWS WAF ログ用に選択できるログ記録のオプションについて説明します。各セクションでは、ログを設定するためのガイダンスと、送信先の種類に固有の動作に関する情報を提供します。ログ記録の送信先を設定したら、ウェブ ACL ログ記録設定にその指定を入力して、送信先へのログ記録を開始することができます。

トピック

- Amazon CloudWatch Logs ロググループ
- Amazon Simple Storage Service バケット
- Amazon Data Firehose 配信ストリーム

Amazon CloudWatch Logs ロググループ

このトピックでは、ウェブ ACL トラフィックログを CloudWatch Logs ロググループに送信するための情報を提供します。

> **Note**
>
> AWS WAFの使用料金に加えて、ログ記録の料金が請求されます。詳細については、「ウェブ ACL トラフィックのログ記録の料金に関する情報」を参照してください。

Amazon CloudWatch Logs にログを送信するには、 CloudWatch ログロググループを作成します。でログ記録を有効にするときは AWS WAF、ロググループ ARN を指定します。ウェブ ACL のログ

記録を有効にすると、 はログストリームの CloudWatch Logs ロググループにログを AWS WAF 配信します。

CloudWatch Logs を使用すると、 コンソールで AWS WAF ウェブ ACL のログを調べることができます。ウェブ ACL ページで、[Logging insights] (ログ記録のインサイト) タブを選択します。このオプションは、 CloudWatch コンソールを介して CloudWatch ログに記録されるインサイトに追加されます。

AWS WAF ウェブ ACL と同じリージョンのウェブ ACL ログのロググループを設定し、ウェブ ACL の管理に使用するのと同じアカウントを使用します。 CloudWatch Logs ロググループの設定については、「ロググループとログストリームの使用」を参照してください。

Logs CloudWatch ロググループのクォータ

CloudWatch ログにはスループットのデフォルトの最大クォータがあり、リージョン内のすべてのロググループで共有されます。このクォータは引き上げをリクエストできます。ログ記録要件が現在のスループット設定に対して高すぎる場合、アカウントの PutLogEventsのスロットリングメトリクスが表示されます。Service Quotas コンソールで制限を表示して引き上げをリクエストするには、CloudWatch 「ログ PutLogEvents クォータ」を参照してください。

ロググループの命名

ロググループ名は aws-waf-logs- で始まる必要があり、末尾を任意のサフィックスにすることができます (例: aws-waf-logs-testLogGroup2)。

結果として生じる ARN 形式は次のとおりです。

```
arn:aws:logs:Region:account-id:log-group:aws-waf-logs-log-group-suffix
```

ログストリームの命名形式は次のとおりです。

```
Region_web-acl-name_log-stream-number
```

リージョン us-east-1 のウェブ ACL TestWebACL のログストリームの例を次に示します。

```
us-east-1_TestWebACL_0
```

ログを CloudWatch Logs に発行するために必要なアクセス許可

Logs ロググループのウェブ ACL CloudWatch トラフィックログ記録を設定するには、このセクションで説明するアクセス許可設定が必要です。アクセス許可は、 AWS WAF フルアクセ

ス管理ポリシーの 1 つ、AWSWAFConsoleFullAccessまたは を使用する場合に設定されますAWSWAFFullAccess。ログ記録と AWS WAF リソースへのよりきめ細かなアクセスを管理する場合は、アクセス許可を自分で設定できます。アクセス許可の管理の詳細については、「IAM ユーザーガイド」の「 AWS リソースのアクセス管理」を参照してください。 AWS WAF マネージドポリシーの詳細については、「AWS の マネージドポリシー AWS WAF」を参照してください。

これらのアクセス許可により、ウェブ ACL ログ記録設定を変更したり、 CloudWatch ログのログ配信を設定したり、ロググループに関する情報を取得したりできます。これらの許可は、 AWS WAF の管理に使用するユーザーにアタッチされる必要があります。

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Action":[

            "wafv2:PutLoggingConfiguration",
            "wafv2:DeleteLoggingConfiguration"
         ],
         "Resource":[
            "*"
         ],
         "Effect":"Allow",
         "Sid":"LoggingConfigurationAPI"
      }
      {
         "Sid":"WebACLLoggingCWL",
         "Action":[
            "logs:CreateLogDelivery",
            "logs:DeleteLogDelivery",
            "logs:PutResourcePolicy",
            "logs:DescribeResourcePolicies",
            "logs:DescribeLogGroups"
         ],
         "Resource":[
            "*"
         ],
         "Effect":"Allow"
      }
   ]
}
```

すべての AWS リソースでアクションが許可されている場合、ポリシーに "Resource"の設定で示されます"*"。つまり、各アクションが をサポートしているすべての AWS リソースでアクションが許可されます。 例えば、アクション wafv2:PutLoggingConfiguration は、wafv2 のログ記録設定リソースでのみサポートされます。

Amazon Simple Storage Service バケット

このトピックは、ウェブ ACL トラフィックログの Amazon S3 バケットへの送信に関する情報を提供します。

> **Note**
>
> AWS WAFの使用料金に加えて、ログ記録の料金が請求されます。詳細については、「ウェブ ACL トラフィックのログ記録の料金に関する情報」を参照してください。

ウェブ ACL トラフィックログを Amazon S3 に送信するには、ウェブ ACL を管理するのと同じアカウントから Amazon S3 バケットを設定し、バケットに aws-waf-logs- で始まる名前を付けます。でログ記録を有効にするときは AWS WAF、バケット名を指定します。ロギングバケットの作成については、「Amazon Simple Storage Service ユーザーガイド」の「バケットの作成」を参照してください。

Amazon Athena インタラクティブクエリサービスを使用して、Amazon S3 ログにアクセスし、分析することができます。Athena を使用すれば、標準 SQL を使用した Amazon S3 内のデータを直接分析しやすくなります。でいくつかのアクションを使用すると AWS Management Console、Amazon S3 に保存されているデータを Athena にポイントし、標準 SQL を使用してアドホッククエリを実行し、結果を取得できます。詳細については、「Amazon Athena ユーザーガイド」の AWS WAF 「ログのクエリ」を参照してください。 Amazon Athena その他のサンプル Amazon Athena クエリについては、 GitHub ウェブサイトの「aws-samples/waf-log-sample-athena-queries」を参照してください。

> **Note**
>
> AWS WAF は、キータイプ Amazon S3 キー (SSE-S3) および (SSE-KMS) の Amazon S3 バケットによる暗号化をサポートしています AWS KMS keys。 AWS WAF は、 によって管理される AWS Key Management Service キーの暗号化をサポートしていません AWS。 AWS Key Management Service

ウェブ ACL は、5 分間隔でログファイルを Amazon S3 バケットに発行します。各ログファイルには、前の 5 分間に記録されたトラフィックのログレコードが含まれています。

ログファイルの最大ファイルサイズは 75 MB です。ログファイルが 5 分以内にファイルサイズの上限に達した場合、ログはレコードの追加を停止し、Amazon S3 バケットに発行してから、新しいログファイルを作成します。

ログファイルは圧縮されます。Amazon S3 コンソールを使用してファイルを開くと、Amazon S3 はログレコードを解凍して表示します。ログファイルをダウンロードする場合、レコードを表示するには解凍する必要があります。

1 つのログファイルには、複数のレコードを含むインターリーブされたエントリが含まれます。ウェブ ACL のすべてのログファイルを表示するには、ウェブ ACL 名、リージョン、およびアカウント ID で集約されたエントリを探します。

命名要件と構文

AWS WAF ログ記録用のバケット名は で始まりaws-waf-logs-、任意のサフィックスで終わる可能性があります。例えば aws-waf-logs-DOC-EXAMPLE-BUCKET-SUFFIX です。

バケットの場所

バケットの場所は次の構文を使用します。

```
s3://aws-waf-logs-DOC-EXAMPLE-BUCKET-SUFFIX/
```

バケット ARN

バケットの Amazon リソースネーム (ARN) の形式は次のとおりです。

```
arn:aws:s3:::aws-waf-logs-DOC-EXAMPLE-BUCKET-SUFFIX
```

プレフィックスを使用したバケットの場所

オブジェクトキー名にプレフィックスを使用してバケットに保存するデータを整理する場合は、ロギングバケット名にプレフィックスを指定できます。

> **Note**
>
> このオプションはコンソールからは使用できません。 AWS WAF APIs 、CLI、または を使用します AWS CloudFormation。

Amazon S3 でのプレフィックスの使用については、「Amazon Simple Storage Service 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。

プレフィックスを使用したバケットの場所には、次の構文が使用されます。

```
s3://aws-waf-logs-DOC-EXAMPLE-BUCKET-SUFFIX/DOC-EXAMPLE-KEY-NAME-PREFIX/
```

バケットフォルダとファイル名

バケット内で、指定したプレフィックスに従って、 AWS WAF ログはアカウント ID、リージョン、ウェブ ACL 名、および日時によって決定されるフォルダ構造で書き込まれます。

```
AWSLogs/account-id/WAFLogs/Region/web-acl-name/YYYY/MM/dd/HH/mm
```

フォルダ内では、ログファイル名は同様の形式になります。

```
account-id_waflogs_Region_web-acl-name_timestamp_hash.log.gz
```

フォルダ構造およびログファイル名で使用される時間の指定は、タイムスタンプ形式の仕様 YYYYMMddTHHmmZ に準拠しています。

DOC-EXAMPLE-BUCKET という名前のバケット用の Amazon S3 バケットに存在するログファイルの例を次に示します。は AWS アカウント です11111111111。ウェブ ACL は TEST-WEBACL であり、リージョンは us-east-1 です。

```
s3://DOC-EXAMPLE-BUCKET/AWSLogs/11111111111/WAFLogs/us-east-1/
TEST-WEBACL/2021/10/28/19/50/11111111111_waflogs_us-east-1_TEST-
WEBACL_20211028T1950Z_e0ca43b5.log.gz
```

> **Note**
>
> AWS WAF ログ記録用のバケット名は で始まりaws-waf-logs-、任意のサフィックスで終わる可能性があります。

Amazon S3 にログを発行するために必須のアクセス許可

Amazon S3 バケットのウェブ ACL トラフィックログ記録を設定するには、次の許可設定が必要です。 AWS WAF フルアクセスマネージドポリシーのいずれか (AWSWAFConsoleFullAccess また

は AWSWAFFullAccess) を使用すると、これらの許可が設定されます。ログ記録と AWS WAF リソースへのよりきめ細かなアクセスを管理する場合は、これらのアクセス許可を自分で設定できます。アクセス許可の管理については、「IAM ユーザーガイド」の「AWS リソースのアクセス管理」を参照してください。 AWS WAF 管理ポリシーの詳細については、「」を参照してくださいAWS のマネージドポリシー AWS WAF。

次の許可を使用すると、ウェブ ACL ログ記録設定を変更し、Amazon S3 バケットへのログ配信を設定できます。これらの許可は、 AWS WAFの管理に使用するユーザーにアタッチされる必要があります。

> **Note**
>
> 以下に示すアクセス許可を設定すると、アクセスが拒否されたことを示すエラーが AWS CloudTrail ログに表示されることがありますが、そのアクセス許可は AWS WAF ログ記録に正確です。

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Action":[
            "wafv2:PutLoggingConfiguration",
            "wafv2:DeleteLoggingConfiguration"
         ],
         "Resource":[
            "*"
         ],
         "Effect":"Allow",
         "Sid":"LoggingConfigurationAPI"
      },
    {

       "Sid":"WebACLLogDelivery",

       "Action":[

          "logs:CreateLogDelivery",

          "logs:DeleteLogDelivery"
```

```
        ],

        "Resource": "*",

        "Effect":"Allow"

    },
      {
         "Sid":"WebACLLoggingS3",
         "Action":[
            "s3:PutBucketPolicy",
            "s3:GetBucketPolicy"
         ],
         "Resource": [
                "arn:aws:s3:::aws-waf-logs-DOC-EXAMPLE-BUCKET"
         ],
         "Effect":"Allow"
      }
   ]
}
```

すべての AWS リソースでアクションが許可されている場合、ポリシーに "Resource"の設定で示されます"*"。つまり、各アクションが をサポートしているすべての AWS リソースでアクションが許可されます。 例えば、アクション wafv2:PutLoggingConfiguration は、wafv2 のログ記録設定リソースでのみサポートされます。

デフォルトでは、Amazon S3 バケットとそれに含まれているオブジェクトはプライベートです。バケット所有者のみが、そのバケットとそれに含まれているオブジェクトにアクセスできます。ただし、バケット所有者は、アクセスポリシーを記述することで他のリソースおよびユーザーに許可を付与することができます。

フローログを作成しているユーザーがバケットを所有している場合、そのバケットにログを発行する許可をフローログに付与するため、サービスは次のポリシーを自動的にバケットにアタッチします。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AWSLogDeliveryWrite",
      "Effect": "Allow",
      "Principal": {
        "Service": "delivery.logs.amazonaws.com"
```

```
      },
      "Action": "s3:PutObject",
      "Resource": "arn:aws:s3:::aws-waf-logs-DOC-EXAMPLE-BUCKET/AWSLogs/account-id/*",
      "Condition": {
        "StringEquals": {
          "s3:x-amz-acl": "bucket-owner-full-control",
          "aws:SourceAccount": ["account-id"]
        },
        "ArnLike": {
          "aws:SourceArn": ["arn:aws:logs:region:account-id:*"]
        }
      }
    },
    {
      "Sid": "AWSLogDeliveryAclCheck",
      "Effect": "Allow",
      "Principal": {
        "Service": "delivery.logs.amazonaws.com"
      },
      "Action": "s3:GetBucketAcl",
      "Resource": "arn:aws:s3:::aws-waf-logs-DOC-EXAMPLE-BUCKET",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": ["account-id"]
        },
        "ArnLike": {
          "aws:SourceArn": ["arn:aws:logs:region:account-id:*"]
        }
      }
    }
  ]
}
```

> **Note**
>
> AWS WAF ログ記録用のバケット名は で始まりaws-waf-logs-、任意のサフィックスで終わる可能性があります。

ログを作成しているユーザーがバケットを所有していないか、バケットに対する GetBucketPolicy および PutBucketPolicy 許可がない場合、ログの作成は失敗します。この場合、バケット所有者はバケットに手動で前述のポリシーを追加して、ログ作成者の AWS アカウント

ID を指定する必要があります。詳細については、「Amazon Simple Storage Service ユーザーガイド」の「S3 バケットポリシーを追加する方法」を参照してください。バケットが複数のアカウントからログを受け取る場合は、各アカウントの AWSLogDeliveryWrite ポリシーステートメントに Resource エレメントエントリを追加します。

例えば、次のバケットポリシーでは AWS アカウント 、 111122223333が という名前のバケットにログを発行することを許可しますaws-waf-logs-*DOC-EXAMPLE-BUCKET*。

```
{
    "Version": "2012-10-17",
    "Id": "AWSLogDeliveryWrite20150319",
    "Statement": [
        {
            "Sid": "AWSLogDeliveryWrite",
            "Effect": "Allow",
            "Principal": {
                "Service": "delivery.logs.amazonaws.com"
            },
            "Action": "s3:PutObject",
                "Resource": "arn:aws:s3:::aws-waf-logs-DOC-EXAMPLE-BUCKET/
AWSLogs/111122223333/*",
            "Condition": {
                "StringEquals": {
                    "s3:x-amz-acl": "bucket-owner-full-control",
                    "aws:SourceAccount": ["111122223333"]
                },
                "ArnLike": {
                    "aws:SourceArn": ["arn:aws:logs:us-east-1:111122223333:*"]
                }
            }
        },
        {
            "Sid": "AWSLogDeliveryAclCheck",
            "Effect": "Allow",
            "Principal": {
                "Service": "delivery.logs.amazonaws.com"
            },
            "Action": "s3:GetBucketAcl",
            "Resource": "arn:aws:s3:::aws-waf-logs-DOC-EXAMPLE-BUCKET",
            "Condition": {
                "StringEquals": {
                "aws:SourceAccount": ["111122223333"]
                },
```

```
                "ArnLike": {
                "aws:SourceArn": ["arn:aws:logs:us-east-1:111122223333:*"]
                }
            }
        }
    ]
}
```

KMS キーで AWS Key Management Service を使用するための許可

ログ記録の送信先が AWS Key Management Service (SSE-KMS) に保存されているキーによるサーバー側の暗号化を使用していて、カスタマーマネージドキー (KMS キー) を使用する場合は、KMS キーを使用するアクセス AWS WAF 許可を付与する必要があります。そのためには、選択した送信先の KMS キーにキーポリシーを追加します。これにより、 AWS WAF ロギングがログファイルを送信先に書き込むことができます。

次のキーポリシーを KMS キーに追加して、 AWS WAF が Amazon S3 バケットにログインできるようにします。

```
{
    "Sid": "Allow AWS WAF to use the key",
    "Effect": "Allow",
    "Principal": {
        "Service": [
            "delivery.logs.amazonaws.com"
        ]
    },
    "Action": "kms:GenerateDataKey*",
    "Resource": "*"
}
```

Amazon S3 ログファイルへのアクセスに必要なアクセス許可

Amazon S3 は、アクセスコントロールリスト (ACL) を使用して、 AWS WAF ログによって作成されたログファイルへのアクセスを管理します。デフォルトでは、バケット所有者が各ログファイルで FULL_CONTROL 許可を持ちます。ログ配信の所有者 (バケット所有者とは異なる場合) は、許可を持ちません。ログ配信アカウントには、READ および WRITE 許可があります。詳細については、「Amazon Simple Storage Service 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。

Amazon Data Firehose 配信ストリーム

このセクションでは、ウェブ ACL トラフィックログを Amazon Data Firehose 配信ストリームに送信するための情報を提供します。

> **Note**
>
> AWS WAFの使用料金に加えて、ログ記録の料金が請求されます。詳細については、「ウェブ ACL トラフィックのログ記録の料金に関する情報」を参照してください。

Amazon Data Firehose にログを送信するには、ウェブ ACL から Firehose で設定した Amazon Data Firehose 配信ストリームにログを送信します。ログ記録を有効にすると、 は Firehose の HTTPS エンドポイントを介してストレージ宛先にログを AWS WAF 配信します。

1 つの AWS WAF ログは、1 つの Firehose レコードに相当します。通常、1 秒あたり 10,000 件のリクエストを受信し、フルログを有効にする場合は、Firehose で 1 秒あたり 10,000 件のレコードを設定する必要があります。Firehose を正しく設定しないと、すべてのログを記録 AWS WAF しません。詳細については、Amazon Kinesis Data Firehose クォータ」を参照してください。

Amazon Data Firehose 配信ストリームを作成し、保存されたログを確認する方法については、「Amazon Data Firehose とは」を参照してください。

配信ストリームの作成については、「Amazon Data Firehose 配信ストリームの作成」を参照してください。

ウェブ ACL の Amazon Data Firehose 配信ストリームの設定

ウェブ ACL の Amazon Data Firehose 配信ストリームを次のように設定します。

- ウェブ ACL の管理に使用するアカウントと同じアカウントを使用して作成します。
- ウェブ ACL と同じリージョンに作成します。Amazon のログをキャプチャする場合は CloudFront、米国東部 (バージニア北部) リージョン に Firehose を作成します。 `us-east-1`
- データ Firehose にプレフィックス `aws-waf-logs-` で始まる名前を付けます。例えば、`aws-waf-logs-us-east-2-analytics` です。
- Direct PUT 用に設定し、アプリケーションが配信ストリームに直接アクセスできるようにします。Amazon Data Firehose コンソールで、配信ストリームのソース設定で、直接 PUT またはその他のソース を選択します。API を通じて、配信ストリームのプロパティ `DeliveryStreamType` を `DirectPut` に設定します。

> **Note**
>
> `Kinesis stream` をソースとして使用しないでください。

Amazon Data Firehose 配信ストリームにログを発行するために必要なアクセス許可

Kinesis Data Firehose の設定に必要な許可を理解するには、「Controlling Access with Amazon Kinesis Data Firehose」(Amazon Kinesis Data Firehose によるアクセスの制御) を参照してください。

Amazon Data Firehose 配信ストリームでウェブ ACL ログ記録を正常に有効にするには、次のアクセス許可が必要です。

- `iam:CreateServiceLinkedRole`
- `firehose:ListDeliveryStreams`
- `wafv2:PutLoggingConfiguration`

サービスにリンクされたロールおよび `iam:CreateServiceLinkedRole` 許可の詳細については、「のサービスにリンクされたロールの使用 AWS WAF」を参照してください。

ウェブ ACL ログ記録設定

ウェブ ACL のログ記録はいつでも有効にしたり、無効にしたりできます。

> **Note**
>
> AWS WAFの使用料金に加えて、ログ記録の料金が請求されます。詳細については、「ウェブ ACL トラフィックのログ記録の料金に関する情報」を参照してください。

ログにログレコードが見つからない場合

まれに、 AWS WAF ログ配信が 100% 未満になり、ログがベストエフォートベースで配信されることがあります。この AWS WAF アーキテクチャは、他のすべての考慮事項よりもアプリケーションのセキュリティを優先します。ロギングフローでトラフィックスロットリングが発生する場合など、状況によってはレコードがドロップされることがあります。影響するレコードは数件に限られます。

ログエントリがいくつか欠落していることに気付いた場合は、AWS Support センターに連絡してください。

ウェブ ACL のログ記録設定では、 がログ AWS WAF に送信する内容をカスタマイズできます。

- フィールドのマスキング – 対応する一致設定を使用するルールのログレコードの次のフィールドをマスキングできます: [URI パス]、[クエリ文字列]、[単一ヘッダー]、および [HTTP メソッド]。マスキングされたフィールドは、ログに REDACTED と表示されます。例えば、ログ内の [クエリ文字列] フィールドをマスキングすると、[クエリ文字列] 一致コンポーネント設定を使用するすべてのルールで REDACTED としてリストされます。マスキングは、ルールで一致するように指定したリクエストコンポーネントにのみ適用されるため、[単一ヘッダー] コンポーネントのマスキングは、[ヘッダー] で照合するルールには適用されません。ログフィールドのリストについては、「ログフィールド」を参照してください。

  > **Note**
  >
  > この設定は、リクエストサンプリングには影響しません。リクエストサンプリングでは、フィールドを除外する唯一の方法は、ウェブ ACL のサンプリングを無効にすることです。

- ログのフィルタリング – フィルタリングを追加して、ログに保持されるウェブリクエストとドロップされるウェブリクエストを指定できます。ウェブリクエストの評価中 AWS WAF に適用される設定をフィルタリングします。次の設定でフィルタリングできます。
  - 完全修飾ラベル – 完全修飾ラベルには、プレフィックス、オプションの名前空間、およびラベル名があります。プレフィクスは、ラベルを追加したルールのルールグループまたはウェブ ACL コンテキストを識別します。ラベルの詳細については、「AWS WAF ウェブリクエストのラベル」を参照してください。
  - ルールアクション – 通常のルールアクション設定だけでなく、ルールグループのルールのレガシー EXCLUDED_AS_COUNT オーバーライドオプションをフィルタリングできます。ルールアクションの設定については、「ルールアクション」を参照してください。ルールグループのルールの現在のルールアクションオーバーライドとレガシールールアクションオーバーライドについては、「ルールグループのアクションオーバーライドオプション」を参照してください。
    - 通常のルールアクションフィルターは、ルールで設定されたアクションだけでなく、ルールグループのルールアクションをオーバーライドするための現在のオプションを使用して設定されたアクションにも適用されます。
    - EXCLUDED_AS_COUNT ログフィルターは、Count アクションログフィルターと重複しています。EXCLUDED_AS_COUNT は、ルールグループのルールアクションを Count にオーバーライドするための現在のオプションとレガシーオプションの両方をフィルタリングします。

ウェブ ACL のログ記録の有効化

ウェブ ACL のログ記録を有効にするには、ログ記録の送信先を既に設定しておく必要があります。ターゲットの選択肢とそれぞれの要件については、「AWS WAF ロギング先」を参照してください。

ウェブ ACL でログ記録を有効にするには

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/wafv2/ で AWS WAF コンソールを開きます。
2. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。
3. ログ記録を有効にするウェブ ACL の名前を選択します。コンソールでウェブ ACL の説明が表示され、そこで編集できます。
4. [Logging] (ログ記録) タブで [Enable logging] (ログの有効化) を選択します。
5. ログ記録の送信先タイプを選択し、設定したログ記録先を選択します。名前が `aws-waf-logs-` で始まるログ記録先を選択する必要があります。
6. (オプション) ログに一部のフィールドを含めたくない場合は、それらをマスキングします。マスキングするフィールドを選び、[Add] (追加) を選択します。必要に応じて手順を繰り返し、追加のフィールドをマスキングします。

   > **Note**
   >
   > この設定は、リクエストサンプリングには影響しません。リクエストサンプリングでは、フィールドを除外する唯一の方法は、ウェブ ACL のサンプリングを無効にすることです。

7. (オプション) すべてのリクエストをログに送信しない場合は、フィルタリング条件と動作を追加します。[Filter logs] (ログをフィルタリング) で、適用する各フィルターについて [Add filter] (フィルターを追加) を選択し、次にフィルター基準を選択して、基準に一致するリクエストを保持するかドロップするかを指定します。フィルターの追加が完了したら、必要に応じて、[Default logging behavior] (デフォルトのログ記録動作) を変更します。
8. [Enable logging] (ログの有効化) を選択します。

> **Note**
>
> ログ記録を正常に有効にすると、 AWS WAF はログ記録の送信先にログを書き込むために必要なアクセス許可を持つサービスにリンクされたロールを作成します。詳細については、「のサービスにリンクされたロールの使用 AWS WAF」を参照してください。

ログフィールド

次のリストは、可能なログフィールドについて説明しています。

アクション

リクエスト AWS WAF に適用された終了アクション。これは許可、ブロック、CAPTCHA、チャレンジのいずれかを示します。ウェブリクエストに有効なトークンが含まれていないとき、CAPTCHA および Challenge アクションは終了します。

args

クエリ文字列。

captchaResponse

アクションがリクエストに適用されたときに入力される、リクエストの CAPTCHA CAPTCHAアクションステータス。このフィールドは、終了中か非終了かにかかわらず、すべてのCAPTCHAアクションに対して入力されます。リクエストにCAPTCHAアクションが複数回適用されている場合、このフィールドはアクションが最後に適用された時点から入力されます。

リクエストにトークンが含まれていない、あるいはトークンが無効または有効期限切れているとき、CAPTCHA アクションはウェブリクエストの検査を終了します。CAPTCHA アクションが終了している場合、このフィールドにはレスポンスコードと失敗の理由が含まれます。アクションが終了していない場合、このフィールドには解決タイムスタンプが含まれます。終了アクションと非終了アクションを区別するには、このフィールドで空でない`failureReason`属性をフィルタリングします。

challengeResponse

アクションがリクエストに適用されたときに入力される、リクエストのチャレンジChallengeアクションステータス。このフィールドは、終了中か非終了かにかかわらず、すべてのChallengeアクションに対して入力されます。リクエストにChallengeアクションが複数回適用されている場合、このフィールドはアクションが最後に適用された時点から入力されます。

リクエストにトークンが含まれていない、あるいはトークンが無効または有効期限切れているとき、Challenge アクションはウェブリクエストの検査を終了します。Challenge アクションが終了している場合、このフィールドにはレスポンスコードと失敗の理由が含まれます。アクションが終了していない場合、このフィールドには解決タイムスタンプが含まれます。終了アクションと非終了アクションを区別するには、このフィールドで空でない`failureReason`属性をフィルタリングします。

clientIp

リクエストを送信するクライアントの IP アドレス。

country

リクエストの送信国。 AWS WAF が発信元の国を特定できない場合、このフィールドは に設定されます-。

excludedRules

ルールグループのルールにのみ使用されます。除外されているルールグループ内のルールのリスト。これらのルールのアクションは Count に設定されています。

オーバーライドルールアクションのオプションを使用してルールがカウントするようにオーバーライドする場合、一致するものはここには一覧表示されません。アクションペア `action` および `overriddenAction` として一覧表示されています。

exclusionType

除外されたルールにアクション Count があることを示すタイプ。

ruleId

除外されたルールグループ内のルールの ID。

formatVersion

ログの形式バージョン。

headers

ヘッダーの一覧。

httpMethod

リクエストの HTTP メソッド。

httpRequest

リクエストに関するメタデータです。

httpSourceId

関連付けられたリソースの ID。

- Amazon CloudFront ディストリビューションの場合、ID は ARN 構文*distribution-id*の です。

  arn:*partition*cloudfront::*account-id*:distribution/*distribution-id*

- Application Load Balancer の場合、ID は ARN 構文で *load-balancer-id* です。

  arn:*partition*:elasticloadbalancing:*region*:*account-id*:loadbalancer/app/*load-balancer-name/load-balancer-id*

- Amazon API Gateway REST API の場合、ID は ARN 構文で *api-id* です。

  arn:*partition*:apigateway:*region*::/restapis/*api-id*/stages/*stage-name*

- AWS AppSync GraphQL API の場合、ID は ARN 構文*GraphQLApiId*の です。

  arn:*partition*:appsync:*region*:*account-id*:apis/*GraphQLApiId*

- Amazon Cognito ユーザープールの場合、ID は ARN 構文で *user-pool-id* です。

  arn:*partition*:cognito-idp:*region*:*account-id*:userpool/*user-pool-id*

- AWS App Runner サービスの場合、ID は ARN 構文*apprunner-service-id*の です。

  arn:*partition*:apprunner:*region*:*account-id*:service/*apprunner-service-name/apprunner-service-id*

httpSourceName

リクエストの送信元。指定できる値: Amazon CFの場合は 、 CloudFrontAPIGWAmazon API Gateway ALBの場合は 、Application Load Balancer APPSYNCの場合は AWS AppSync、Amazon Cognito COGNITOIDPの場合は 、App Runner APPRUNNERの場合は 、Verified Access VERIFIED_ACCESSの場合は 。 Amazon API Gateway Amazon Cognito

httpVersion

HTTP のバージョン。

ja3Fingerprint

リクエストの JA3 フィンガープリント。

> **Note**
>
> JA3 フィンガープリント検査は、Amazon CloudFront ディストリビューションと Application Load Balancer でのみ使用できます。

JA3 フィンガープリントは、受信リクエストの TLS Client Hello から生成される 32 文字のハッシュです。このフィンガープリントは、クライアントの TLS 設定の一意の識別子として機能します。 AWS WAF は、計算に十分な TLS Client Hello 情報を持つ各リクエストについて、このフィンガープリントを計算してログに記録します。

この値は、ウェブ ACL ルールで JA3 フィンガープリントの一致を設定するときに指定します。JA3 フィンガープリントとの一致を作成する方法については、「リクエストコンポーネントオプション」の「JA3 フィンガープリント」に記載されているルールステートメントを参照してください。

ラベル

ウェブリクエストのラベル。これらのラベルは、最初の 100 個のラベルの request. AWS WAF logs の評価に使用されたルールによって適用されました。

nonTerminatingMatchingルール

リクエストに一致した非終了ルールのリスト。リスト内の各項目には、次の情報が含まれています。

アクション

リクエスト AWS WAF に適用されたアクション。カウント、CAPTCHA、チャレンジのいずれかを示します。ウェブリクエストに有効なトークンが含まれていると、CAPTCHA および Challenge は終了処理しません。

ruleId

リクエストに一致し、非終了ルールの ID。

ruleMatchDetails

リクエストに一致したルールに関する詳細情報。このフィールドは、SQL インジェクションおよびクロスサイトスクリプティング (XSS) 一致ルールステートメントに対してのみ設定されます。一致ルールでは、複数の検査基準の一致が必要になる場合があるため、これらの一致の詳細は、一致基準の配列として提供されます。

各ルールについて提供される追加情報は、ルール設定、ルール一致タイプ、一致の詳細などの要因によって異なります。例えば、 CAPTCHAまたは Challengeアクションを持つルールの場合、captchaResponseまたは が一覧表示challengeResponseされます。一致するルールがルールグループにあり、設定されたルールアクションを上書きした場合、設定されたアクションは で提供されますoverriddenAction。

oversizeFields

ウェブ ACL によって検査され、 AWS WAF 検査制限を超えているウェブリクエスト内のフィールドのリスト。フィールドがオーバーサイズであっても、ウェブ ACL が検査しない場合、ここには表示されません。

このリストには、REQUEST_BODY、REQUEST_JSON_BODY、REQUEST_HEADERS、および REQUEST_COOKIES の値が何個か含まれることも、含まれないこともあります。オーバーサイズフィールドの詳細については、「でのオーバーサイズリクエストコンポーネントの処理 AWS WAF」を参照してください。

rateBasedRuleリスト

このリクエストで動作したレートベースのルールのリスト。レートベースルールの詳細については、「レートベースのルールステートメント」を参照してください。

rateBasedRuleID

このリクエストで動作したレートベースのルールの ID。これがリクエストを終了した場合、rateBasedRuleId の ID は、terminatingRuleId の ID と同じです。

rateBasedRule名前

このリクエストで動作したレートベースのルールの名前。

limitKey

ルールが使用している集約のタイプ。指定できる値は、ウェブリクエストの発信元用の IP、リクエストのヘッダーで転送された IP 用の FORWARDED_IP、カスタム集約キー設定用の CUSTOMKEYS、および集約なしですべてのリクエストをまとめてカウントする用の CONSTANT です。

limitValue

単一の IP アドレスタイプでレート制限を行う場合にのみ使用される。リクエストに有効ではない IP アドレスが含まれている場合、limitvalue は INVALID です。

maxRateAllowed

特定の集約インスタンスに対して指定された時間枠で許可されるリクエストの最大数。集約インスタンスは、 limitKeyに加えて、レートベースのルール設定で指定した追加のキー仕様によって定義されます。

evaluationWindowSec

がリクエストに AWS WAF 含めた時間を秒単位でカウントします。

customValues

リクエスト内のレートベースのルールによって識別される一意の値。文字列値の場合、ログは文字列値の最初の 32 文字を出力します。キータイプに応じて、これらの値は HTTP メソッドやクエリ文字列といったキーだけの場合もあれば、ヘッダーやヘッダー名のようなキーと名前の場合もあります。

requestHeadersInserted

カスタムリクエストの処理用に挿入されるヘッダーのリスト。

requestId

基盤となるホストサービスによって生成されるリクエストの ID。Application Load Balancer では、これはトレース ID です。その他すべての場合、これはリクエスト ID です。

responseCodeSent

カスタムレスポンスで送信されるレスポンスコード。

ruleGroupId

ルールグループの ID ルールがリクエストをブロックした場合、`ruleGroupID` の ID は、`terminatingRuleId` の ID と同じです。

ruleGroupList

このリクエストに対して動作したルールグループのリスト (一致情報を含む)。

terminatingRule

リクエストを終了したルール。これが存在する場合は、次の情報が含まれます。

アクション

リクエスト AWS WAF に適用された終了アクション。これは許可、ブロック、CAPTCHA、チャレンジのいずれかを示します。ウェブリクエストに有効なトークンが含まれていないとき、CAPTCHA および Challenge アクションは終了します。

ruleId

リクエストに一致したルールの ID。

ruleMatchDetails

リクエストに一致したルールに関する詳細情報。このフィールドは、SQL インジェクションおよびクロスサイトスクリプティング (XSS) 一致ルールステートメントに対してのみ設定されます。一致ルールでは、複数の検査基準の一致が必要になる場合があるため、これらの一致の詳細は、一致基準の配列として提供されます。

各ルールについて提供される追加情報は、ルール設定、ルール一致タイプ、一致の詳細などの要因によって異なります。例えば、 CAPTCHAまたは Challengeアクションを持つルールの場合、`captchaResponse`または が一覧表示`challengeResponse`されます。一致するルールがルールグループにあり、設定されたルールアクションを上書きした場合、設定されたアクションは で提供されます`overriddenAction`。

terminatingRuleId

リクエストを終了したルールの ID。リクエストを終了したものがない場合、この値は `Default_Action` となります。

terminatingRuleMatch詳細

リクエストに一致した終了ルールに関する詳細情報。終了ルールには、ウェブリクエストに対する検査プロセスを終了するアクションがあります。終了ルールに可能なアクションには、Allow、Block、CAPTCHA、Challenge が含まれます。ウェブリクエストの検査中に、リクエストに一致し、終了アクションがある最初のルールで、 は検査 AWS WAF を停止し、アクションを適用します。ウェブリクエストには、一致する終了ルールのログで報告された脅威に加えて、他の脅威が含まれている可能性があります。

これは、SQL インジェクションおよびクロスサイトスクリプティング (XSS) 一致ルールステートメントに対してのみ設定されます。一致ルールでは、複数の検査基準の一致が必要になる場合があるため、これらの一致の詳細は、一致基準の配列として提供されます。

terminatingRuleType

リクエストを終了したルールのタイプ。可能な値: RATE_BASED、REGULAR、GROUP、MANAGED_RULE_GROUP。

timestamp

タイムスタンプ (ミリ秒単位)。

uri

リクエストの URI。

webaclId

ウェブ ACL の GUID。

ログの例

Example レートベースのルール 1: キーが 1 つで、**Header:dogname** に設定されたルール設定

```
{
  "Name": "RateBasedRule",
  "Priority": 1,
  "Statement": {
    "RateBasedStatement": {
      "Limit": 100,
      "AggregateKeyType": "CUSTOM_KEYS",
      "CustomKeys": [
        {
          "Header": {
            "Name": "dogname",
            "TextTransformations": [
              {
                "Priority": 0,
                "Type": "NONE"
              }
            ]
          }
        }
      ]
    }
  },
  "Action": {
    "Block": {}
  },
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "RateBasedRule"
  }
}
```

Example レートベースのルール 1: レートベースのルールによってブロックされたリクエストのログエントリ


```
{
   "timestamp":1683355579981,
   "formatVersion":1,
   "webaclId": ...,
   "terminatingRuleId":"RateBasedRule",
   "terminatingRuleType":"RATE_BASED",
   "action":"BLOCK",
   "terminatingRuleMatchDetails":[

   ],
   "httpSourceName":"APIGW",
   "httpSourceId":"EXAMPLE11:rjvegx5guh:CanaryTest",
   "ruleGroupList":[

   ],
   "rateBasedRuleList":[
      {
         "rateBasedRuleId": ...,
         "rateBasedRuleName":"RateBasedRule",
         "limitKey":"CUSTOMKEYS",
         "maxRateAllowed":100,
         "evaluationWindowSec":"120",
         "customValues":[
            {
               "key":"HEADER",
               "name":"dogname",
               "value":"ella"
            }
         ]
      }
   ],
   "nonTerminatingMatchingRules":[

   ],
   "requestHeadersInserted":null,
   "responseCodeSent":null,
   "httpRequest":{
      "clientIp":"52.46.82.45",
      "country":"FR",
      "headers":[
```

```
          {
             "name":"X-Forwarded-For",
             "value":"52.46.82.45"
          },
          {
             "name":"X-Forwarded-Proto",
             "value":"https"
          },
          {
             "name":"X-Forwarded-Port",
             "value":"443"
          },
          {
             "name":"Host",
             "value":"rjvegx5guh.execute-api.eu-west-3.amazonaws.com"
          },
          {
             "name":"X-Amzn-Trace-Id",
             "value":"Root=1-645566cf-7cb058b04d9bb3ee01dc4036"
          },
          {
             "name":"dogname",
             "value":"ella"
          },
          {
             "name":"User-Agent",
             "value":"RateBasedRuleTestKoipOneKeyModulePV2"
          },
          {
             "name":"Accept-Encoding",
             "value":"gzip,deflate"
          }
       ],
       "uri":"/CanaryTest",
       "args":"",
       "httpVersion":"HTTP/1.1",
       "httpMethod":"GET",
       "requestId":"Ed0AiHF_CGYF-DA="
    }
}
```

Example レートベースのルール 2: キーが 2 つで、**Header:dogname** および **Header:catname** に設定されたルール設定

```
{
  "Name": "RateBasedRule",
  "Priority": 1,
  "Statement": {
    "RateBasedStatement": {
      "Limit": 100,
      "AggregateKeyType": "CUSTOM_KEYS",
      "CustomKeys": [
        {
          "Header": {
            "Name": "dogname",
            "TextTransformations": [
              {
                "Priority": 0,
                "Type": "NONE"
              }
            ]
          }
        },
        {
          "Header": {
            "Name": "catname",
            "TextTransformations": [
              {
                "Priority": 0,
                "Type": "NONE"
              }
            ]
          }
        }
      ]
    }
  },
  "Action": {
    "Block": {}
  },
  "VisibilityConfig": {
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "RateBasedRule"
```

```
      }
    }
```

Example レートベースのルール 2: レートベースのルールによってブロックされたリクエストのログエントリ

```
{
   "timestamp":1633322211194,
   "formatVersion":1,
   "webaclId":...,
   "terminatingRuleId":"RateBasedRule",
   "terminatingRuleType":"RATE_BASED",
   "action":"BLOCK",
   "terminatingRuleMatchDetails":[

   ],
   "httpSourceName":"APIGW",
   "httpSourceId":"EXAMPLE11:rjvegx5guh:CanaryTest",
   "ruleGroupList":[

   ],
   "rateBasedRuleList":[
      {
         "rateBasedRuleId":...,
         "rateBasedRuleName":"RateBasedRule",
         "limitKey":"CUSTOMKEYS",
         "maxRateAllowed":100,
         "evaluationWindowSec":"120",
         "customValues":[
            {
               "key":"HEADER",
               "name":"dogname",
               "value":"ella"
            },
            {
               "key":"HEADER",
               "name":"catname",
               "value":"goofie"
            }
         ]
      }
   ],
   "nonTerminatingMatchingRules":[
```

```
],
"requestHeadersInserted":null,
"responseCodeSent":null,
"httpRequest":{
   "clientIp":"52.46.82.35",
   "country":"FR",
   "headers":[
      {
         "name":"X-Forwarded-For",
         "value":"52.46.82.35"
      },
      {
         "name":"X-Forwarded-Proto",
         "value":"https"
      },
      {
         "name":"X-Forwarded-Port",
         "value":"443"
      },
      {
         "name":"Host",
         "value":"23llbyn8v3.execute-api.eu-west-3.amazonaws.com"
      },
      {
         "name":"X-Amzn-Trace-Id",
         "value":"Root=1-64556629-17ac754c2ed9f0620e0f2a0c"
      },
      {
         "name":"catname",
         "value":"goofie"
      },
      {
         "name":"dogname",
         "value":"ella"
      },
      {
         "name":"User-Agent",
         "value":"Apache-HttpClient/UNAVAILABLE (Java/11.0.19)"
      },
      {
         "name":"Accept-Encoding",
         "value":"gzip,deflate"
      }
```

```
      ],
      "uri":"/CanaryTest",
      "args":"",
      "httpVersion":"HTTP/1.1",
      "httpMethod":"GET",
      "requestId":"EdzmlH5OCGYF1vQ="
   }
}
```

Example SQLi 検出 (終了) でトリガーされたルールのログ出力

```
{
    "timestamp": 1576280412771,
    "formatVersion": 1,
    "webaclId": "arn:aws:wafv2:ap-southeast-2:111122223333:regional/webacl/
STMTest/1EXAMPLE-2ARN-3ARN-4ARN-123456EXAMPLE",
    "terminatingRuleId": "STMTest_SQLi_XSS",
    "terminatingRuleType": "REGULAR",
    "action": "BLOCK",
    "terminatingRuleMatchDetails": [
        {
            "conditionType": "SQL_INJECTION",
            "sensitivityLevel": "HIGH",
            "location": "HEADER",
            "matchedData": [
                "10",
                "AND",
                "1"
            ]
        }
    ],
    "httpSourceName": "-",
    "httpSourceId": "-",
    "ruleGroupList": [],
    "rateBasedRuleList": [],
    "nonTerminatingMatchingRules": [],
    "httpRequest": {
        "clientIp": "1.1.1.1",
        "country": "AU",
        "headers": [
            {
                "name": "Host",
                "value": "localhost:1989"
```

```
            },
            {
                "name": "User-Agent",
                "value": "curl/7.61.1"
            },
            {
                "name": "Accept",
                "value": "*/*"
            },
            {
                "name": "x-stm-test",
                "value": "10 AND 1=1"
            }
        ],
        "uri": "/myUri",
        "args": "",
        "httpVersion": "HTTP/1.1",
        "httpMethod": "GET",
        "requestId": "rid"
    },
    "labels": [
        {
            "name": "value"
        }
    ]
}
```

Example SQLi 検出 (非終了) でトリガーされたルールのログ出力

```
{
    "timestamp":1592357192516
    ,"formatVersion":1
    ,"webaclId":"arn:aws:wafv2:us-east-1:123456789012:global/webacl/hello-
world/5933d6d9-9dde-js82-v8aw-9ck28nv9"
    ,"terminatingRuleId":"Default_Action"
    ,"terminatingRuleType":"REGULAR"
    ,"action":"ALLOW"
    ,"terminatingRuleMatchDetails":[]
    ,"httpSourceName":"-"
    ,"httpSourceId":"-"
    ,"ruleGroupList":[]
    ,"rateBasedRuleList":[]
    ,"nonTerminatingMatchingRules":
```

```
    [{
        "ruleId":"TestRule"
        ,"action":"COUNT"
        ,"ruleMatchDetails":
        [{
            "conditionType":"SQL_INJECTION"
            ,"sensitivityLevel": "HIGH"
            ,"location":"HEADER"
            ,"matchedData":[
                "10"
                ,"and"
                ,"1"]
            }]
    }]
    ,"httpRequest":{
        "clientIp":"3.3.3.3"
        ,"country":"US"
        ,"headers":[
            {"name":"Host","value":"localhost:1989"}
            ,{"name":"User-Agent","value":"curl/7.61.1"}
            ,{"name":"Accept","value":"*/*"}
            ,{"name":"myHeader","myValue":"10 AND 1=1"}
            ]
            ,"uri":"/myUri","args":""
            ,"httpVersion":"HTTP/1.1"
            ,"httpMethod":"GET"
            ,"requestId":"rid"
    },
    "labels": [
        {
            "name": "value"
        }
    ]
}
```

Example ルールグループ内でトリガーされた複数のルールのログ出力 (RuleA-XSS は終了、Rule-B は非終了)

```
{
    "timestamp":1592361810888,
    "formatVersion":1,
    "webaclId":"arn:aws:wafv2:us-east-1:123456789012:global/webacl/hello-
world/5933d6d9-9dde-js82-v8aw-9ck28nv9"
```

```
    ,"terminatingRuleId":"RG-Reference"
    ,"terminatingRuleType":"GROUP"
    ,"action":"BLOCK",
    "terminatingRuleMatchDetails":
    [{
        "conditionType":"XSS"
        ,"location":"HEADER"
        ,"matchedData":["<","frameset"]
    }]
    ,"httpSourceName":"-"
    ,"httpSourceId":"-"
    ,"ruleGroupList":
    [{
        "ruleGroupId":"arn:aws:wafv2:us-east-1:123456789012:global/rulegroup/hello-
world/c05lb698-1f11-4m41-aef4-99a506d53f4b"
        ,"terminatingRule":{
            "ruleId":"RuleA-XSS"
            ,"action":"BLOCK"
            ,"ruleMatchDetails":null
            }
        ,"nonTerminatingMatchingRules":
        [{
            "ruleId":"RuleB-SQLi"
            ,"action":"COUNT"
            ,"ruleMatchDetails":
            [{
                "conditionType":"SQL_INJECTION"
                ,"sensitivityLevel": "LOW"
                ,"location":"HEADER"
                ,"matchedData":[
                    "10"
                    ,"and"
                    ,"1"]
            }]
        }]
        ,"excludedRules":null
    }]
    ,"rateBasedRuleList":[]
    ,"nonTerminatingMatchingRules":[]
    ,"httpRequest":{
        "clientIp":"3.3.3.3"
        ,"country":"US"
        ,"headers":
        [
```

```
            {"name":"Host","value":"localhost:1989"}
            ,{"name":"User-Agent","value":"curl/7.61.1"}
            ,{"name":"Accept","value":"*/*"}
            ,{"name":"myHeader1","value":"<frameset onload=alert(1)>"}
            ,{"name":"myHeader2","value":"10 AND 1=1"}
            ]
        ,"uri":"/myUri"
        ,"args":""
        ,"httpVersion":"HTTP/1.1"
        ,"httpMethod":"GET"
        ,"requestId":"rid"
    },
    "labels": [
        {
            "name": "value"
        }
    ]
}
```

Example コンテンツタイプ JSON を使用したリクエストボディの検査のためにトリガーされたルールのログ出力

AWS WAF 現在、JSON UNKNOWN 本文検査の場所をとして報告しています。

```
{
    "timestamp": 1576280412771,
    "formatVersion": 1,
    "webaclId": "arn:aws:wafv2:ap-southeast-2:123456789012:regional/webacl/test/111",
    "terminatingRuleId": "STMTest_SQLi_XSS",
    "terminatingRuleType": "REGULAR",
    "action": "BLOCK",
    "terminatingRuleMatchDetails": [
        {
            "conditionType": "SQL_INJECTION",
            "sensitivityLevel": "LOW",
            "location": "UNKNOWN",
            "matchedData": [
                "10",
                "AND",
                "1"
            ]
        }
    ],
```

```
    "httpSourceName": "ALB",
    "httpSourceId": "alb",
    "ruleGroupList": [],
    "rateBasedRuleList": [],
    "nonTerminatingMatchingRules": [],
    "requestHeadersInserted":null,
    "responseCodeSent":null,
    "httpRequest": {
        "clientIp": "1.1.1.1",
        "country": "AU",
        "headers": [],
        "uri": "",
        "args": "",
        "httpVersion": "HTTP/1.1",
        "httpMethod": "POST",
        "requestId": "null"
    },
    "labels": [
        {
            "name": "value"
        }
    ]
}
```

Example 有効期限が切れていない有効な CAPTCHA トークンを使用したウェブリクエストに対する CAPTCHA ルールのログ出力

次のログリストは、CAPTCHA アクションを持つルールと一致したウェブリクエストについてのものです。ウェブリクエストには有効で有効期限が切れていない CAPTCHA トークンがあり、アクションの動作と同様に CAPTCHA マッチとしてのみ記録されます AWS WAF。CountこのCAPTCHA の一致については、nonTerminatingMatchingRules に記載されています。

```
{
  "timestamp": 1632420429309,
  "formatVersion": 1,
  "webaclId": "arn:aws:wafv2:us-east-1:123456789012:regional/webacl/captcha-web-
acl/585e38b5-afce-4d2a-b417-14fb08b66c67",
  "terminatingRuleId": "Default_Action",
  "terminatingRuleType": "REGULAR",
  "action": "ALLOW",
  "terminatingRuleMatchDetails": [],
  "httpSourceName": "APIGW",
  "httpSourceId": "123456789012:b34myvfw0b:pen-test",
```

```
  "ruleGroupList": [],
  "rateBasedRuleList": [],
  "nonTerminatingMatchingRules": [
    {
      "ruleId": "captcha-rule",
      "action": "CAPTCHA",
      "ruleMatchDetails": [],
      "captchaResponse": {
        "responseCode": 0,
        "solveTimestamp": 1632420429
      }
    }
  ],
  "requestHeadersInserted": [
    {
      "name": "x-amzn-waf-test-header-name",
      "value": "test-header-value"
    }
  ],
  "responseCodeSent": null,
  "httpRequest": {
    "clientIp": "72.21.198.65",
    "country": "US",
    "headers": [
      {
        "name": "X-Forwarded-For",
        "value": "72.21.198.65"
      },
      {
        "name": "X-Forwarded-Proto",
        "value": "https"
      },
      {
        "name": "X-Forwarded-Port",
        "value": "443"
      },
      {
        "name": "Host",
        "value": "b34myvfw0b.gamma.execute-api.us-east-1.amazonaws.com"
      },
      {
        "name": "X-Amzn-Trace-Id",
        "value": "Root=1-614cc24d-5ad89a09181910c43917a888"
      },
```

```
      {
        "name": "cache-control",
        "value": "max-age=0"
      },
      {
        "name": "sec-ch-ua",
        "value": "\"Chromium\";v=\"94\", \"Google Chrome\";v=\"94\", \";Not A Brand
\";v=\"99\""
      },
      {
        "name": "sec-ch-ua-mobile",
        "value": "?0"
      },
      {
        "name": "sec-ch-ua-platform",
        "value": "\"Windows\""
      },
      {
        "name": "upgrade-insecure-requests",
        "value": "1"
      },
      {
        "name": "user-agent",
        "value": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/94.0.4606.54 Safari/537.36"
      },
      {
        "name": "accept",
        "value": "text/html,application/xhtml+xml,application/xml;q=0.9,image/
avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9"
      },
      {
        "name": "sec-fetch-site",
        "value": "same-origin"
      },
      {
        "name": "sec-fetch-mode",
        "value": "navigate"
      },
      {
        "name": "sec-fetch-user",
        "value": "?1"
      },
      {
```

```
        "name": "sec-fetch-dest",
        "value": "document"
      },
      {
        "name": "referer",
        "value": "https://b34myvfw0b.gamma.execute-api.us-east-1.amazonaws.com/pen-
test/pets"
      },
      {
        "name": "accept-encoding",
        "value": "gzip, deflate, br"
      },
      {
        "name": "accept-language",
        "value": "en-US,en;q=0.9"
      },
      {
        "name": "cookie",
        "value": "aws-waf-token=51c71352-41f5-4f6d-b676-c24907bdf819:EQoAZ/J
+AAQAAAAA:t9wvxbw042wva7E2Y6lgud/
bS6YG0CJKVAJqaRqDZ140ythKW0Zj9wKB2O8lSkYDRqf1yONcVBFo5u0eYi0tvT4rtQCXsu
+KanAardW8go4QSLw4yoED59lgV7oAhGyCalAzE7ra29j+RvvZPsQyoQuDCrtoY/TvQyMTXIXzGPDC/rKBbg=="
      }
    ],
    "uri": "/pen-test/pets",
    "args": "",
    "httpVersion": "HTTP/1.1",
    "httpMethod": "GET",
    "requestId": "GINMHHUgoAMFxug="
  }
}
```

Example CAPTCHA トークンがないウェブリクエストに対する CAPTCHA ルールのログ出力

次のログリストは、CAPTCHA アクションを持つルールと一致したウェブリクエストについてのものです。ウェブリクエストには CAPTCHA トークンがなく、ブロックされました。 AWS WAF

```
{
  "timestamp": 1632420416512,
  "formatVersion": 1,
  "webaclId": "arn:aws:wafv2:us-east-1:123456789012:regional/webacl/captcha-web-
acl/585e38b5-afce-4d2a-b417-14fb08b66c67",
  "terminatingRuleId": "captcha-rule",
```

```
  "terminatingRuleType": "REGULAR",
  "action": "CAPTCHA",
  "terminatingRuleMatchDetails": [],
  "httpSourceName": "APIGW",
  "httpSourceId": "123456789012:b34myvfw0b:pen-test",
  "ruleGroupList": [],
  "rateBasedRuleList": [],
  "nonTerminatingMatchingRules": [],
  "requestHeadersInserted": null,
  "responseCodeSent": 405,
  "httpRequest": {
    "clientIp": "72.21.198.65",
    "country": "US",
    "headers": [
      {
        "name": "X-Forwarded-For",
        "value": "72.21.198.65"
      },
      {
        "name": "X-Forwarded-Proto",
        "value": "https"
      },
      {
        "name": "X-Forwarded-Port",
        "value": "443"
      },
      {
        "name": "Host",
        "value": "b34myvfw0b.gamma.execute-api.us-east-1.amazonaws.com"
      },
      {
        "name": "X-Amzn-Trace-Id",
        "value": "Root=1-614cc240-18b57ff33c10e5c016b508c5"
      },
      {
        "name": "sec-ch-ua",
        "value": "\"Chromium\";v=\"94\", \"Google Chrome\";v=\"94\", \";Not A Brand
\";v=\"99\""
      },
      {
        "name": "sec-ch-ua-mobile",
        "value": "?0"
      },
      {
```

```
        "name": "sec-ch-ua-platform",
        "value": "\"Windows\""
      },
      {
        "name": "upgrade-insecure-requests",
        "value": "1"
      },
      {
        "name": "user-agent",
        "value": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/94.0.4606.54 Safari/537.36"
      },
      {
        "name": "accept",
        "value": "text/html,application/xhtml+xml,application/xml;q=0.9,image/
avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9"
      },
      {
        "name": "sec-fetch-site",
        "value": "cross-site"
      },
      {
        "name": "sec-fetch-mode",
        "value": "navigate"
      },
      {
        "name": "sec-fetch-user",
        "value": "?1"
      },
      {
        "name": "sec-fetch-dest",
        "value": "document"
      },
      {
        "name": "accept-encoding",
        "value": "gzip, deflate, br"
      },
      {
        "name": "accept-language",
        "value": "en-US,en;q=0.9"
      }
    ],
    "uri": "/pen-test/pets",
    "args": "",
```

```
    "httpVersion": "HTTP/1.1",
    "httpMethod": "GET",
    "requestId": "GINKHEssoAMFsrg="
  },
  "captchaResponse": {
    "responseCode": 405,
    "solveTimestamp": 0,
    "failureReason": "TOKEN_MISSING"
  }
}
```

AWS WAF 保護機能のテストと調整

AWS WAF ウェブ ACL の変更は、Web サイトまたはウェブアプリケーションのトラフィックに適用する前にテストして調整することをお勧めします。

> **本番稼働トラフィックのリスク**
>
> 本番稼働トラフィックにウェブ ACL 実装をデプロイする前に、トラフィックへの潜在的な影響に慣れるまで、ステージング環境またはテスト環境でテストおよびチューニングします。その後、ルールを有効にする前に、本番稼働用トラフィックでカウントモードでルールをテストしてチューニングします。

このセクションでは、 AWS WAF ウェブ ACL、ルール、ルールグループ、IP セット、および正規表現パターンセットをテストおよび調整するためのガイダンスを提供します。

また、このセクションでは、他のユーザーによって管理されているルールグループの使用をテストするための一般的なガイダンスも提供します。これらには、 AWS マネージドルールルールグループ、AWS Marketplace マネージドルールグループ、別のアカウントで共有されているルールグループが含まれます。これらのルールグループについては、ルールグループプロバイダーから取得したガイダンスにも従ってください。

- Bot Control AWS マネージドルールルールグループについては、も参照してくださいAWS WAF Bot Control のテストとデプロイ。
- AWS アカウント乗っ取り防止マネージドルールルールグループについては、も参照してくださいATP のテストとデプロイ。
- AWS アカウント作成詐欺防止マネージドルールルールグループについては、も参照してくださいACFP のテストとデプロイ。

更新中の一時的な不一致

ウェブ ACL AWS WAF やその他のリソースを作成または変更した場合、その変更がリソースが保存されているすべての領域に反映されるまでに少し時間がかかります。伝播時間は、数秒から数分までかかります。

次の内容では、変更伝播中に直面する一時的な不整合性の例を紹介します。

- ウェブ ACL を作成した後、それをリソースに関連付けようとすると、ウェブ ACL が利用できないことを示す例外が表示される場合があります。
- ルールグループをウェブ ACL に追加した後、新しいルールグループのルールは、ウェブ ACL が使用されるエリアで有効になり、別のエリアでは有効にならない場合があります。
- ルールのアクション設定を変更した後、古いアクションを一部のエリアで確認され、新しいアクションを別のエリアで確認される場合があります。
- ブロックルールで使用されている IP セットに IP アドレスを追加した後、新しいアドレスはあるエリアではブロックされ、別のエリアでは許可される場合があります。

ハイレベルステップのテストとチューニング

このセクションでは、ウェブ ACL が使用するルールまたはルールグループなど、ウェブ ACL に対する変更をテストするための手順のチェックリストを示します。

> **Note**
>
> このセクションのガイダンスに従うには、ウェブ ACL、ルール、ルールグループなどの AWS WAF 保護の作成および管理方法を理解する必要があります。この情報は、このガイドの前のセクションで説明しています。

ウェブ ACL をテストしてチューニングするには

これらのステップを最初にテスト環境で実行し、次に本番環境で実行します。

1. テストの準備

   監視環境を準備し、 AWS WAF 新しい保護機能をテスト用のカウントモードに切り替え、必要なリソース関連付けを作成します。

   テストの準備 を参照してください。

2. テスト環境および本番環境での監視とチューニング

   AWS WAF 保護機能の監視と調整は、まずテスト環境またはステージング環境で行い、次に本番環境で、必要に応じてトラフィックを処理できるようになるまで行います。

   モニタリングとチューニング を参照してください。

3. 本番環境で保護を有効にする

   テスト保護に納得できたら、それらを本番モードに切り替え、不要なテストアーティファクトをクリーンアップして、監視を続行します。

   本番環境で保護の有効化 を参照してください。

変更の実装が完了したら、本番環境でウェブトラフィックと保護を監視し、必要に応じて動作していることを確認します。Web トラフィックパターンは時間の経過とともに変化することがあるため、時々保護を調整する必要がある場合があります。

テストの準備

このセクションでは、 AWS WAF 保護をテストおよび調整するためのセットアップ方法について説明します。

> **Note**
>
> このセクションのガイダンスに従うには、ウェブ ACLs、ルール、ルールグループなどの AWS WAF 保護を作成および管理する方法を一般的に理解する必要があります。この情報は、このガイドの前のセクションで説明しています。

テストを準備するには

1. ウェブ ACL のウェブ ACL ログ記録、Amazon CloudWatch メトリクス、およびウェブリクエストサンプリングを有効にする

   ログ記録、メトリクス、およびサンプリングを使用して、ウェブ ACL ルールとウェブトラフィックとの相互作用を監視します。

   - ログ記録 — ウェブ ACL が評価するウェブリクエストをログ AWS WAF に記録するようにを設定できます。ログは、 CloudWatch ログ、Amazon S3 バケット、または Amazon Data

Firehose 配信ストリームに送信できます。フィールドの修正やフィルタリングの適用も可能です。詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。

- Amazon Security Lake – ウェブ ACL データを収集するように Security Lake を設定できます。Security Lake は、正規化、分析、管理のためにさまざまなソースからログとイベントデータを収集します。このオプションの詳細については、「Amazon Security Lake ユーザーガイド」の「Amazon Security Lake とは」および「 AWS のサービスからデータを収集する」を参照してください。
- Amazon CloudWatch メトリクス – ウェブ ACL 設定で、モニタリングするすべてのメトリクス仕様を指定します。メトリクスは、 AWS WAF および CloudWatch コンソールから表示できます。詳細については、「Amazon によるモニタリング CloudWatch」を参照してください。
- ウェブリクエストサンプリング — ウェブ ACL が評価するすべてのウェブリクエストのサンプルを表示できます。ウェブリクエストサンプリングの詳細については、「ウェブリクエストのサンプルの表示」を参照してください。

2. 保護を Count モードに設定します。

ウェブ ACL 設定で、テストするものをカウントモードに切り替えます。これにより、テスト保護は、リクエストの処理方法を変更することなく、ウェブリクエストに対する一致を記録します。メトリクス、ログ、およびサンプリングされたリクエストで一致を確認し、一致条件を検証し、ウェブトラフィックにどのような影響があるかを理解することができます。一致するリクエストにラベルを追加するルールは、ルールのアクションに関係なくラベルを追加します。

- ウェブ ACL で定義されているルール — ウェブ ACL のルールを編集し、アクションをCount に設定します。
- ルールグループ — ウェブ ACL 設定で、ルールグループのルールステートメントを編集し、[Rules] (ルール) ペインで [Override all rule actions] (すべてのルールアクションをオーバーライド) ドロップダウンを開いて [Count] 選択します。JSON でウェブ ACL を管理する場合、ルールグループ参照ステートメントで `RuleActionOverrides` 設定にリールを追加し、`ActionToUse` を Count に設定します。次のリスト例は、`AWSManagedRulesAnonymousIpList` AWS マネージドルールのルールグループの 2 つのルールのオーバーライドを示しています。

```
"ManagedRuleGroupStatement": {
  "VendorName": "AWS",
  "Name": "AWSManagedRulesAnonymousIpList",
    "RuleActionOverrides": [
```

```
        {
          "ActionToUse": {
            "Count": {}
          },
          "Name": "AnonymousIPList"
        },
        {
          "ActionToUse": {
            "Count": {}
          },
          "Name": "HostingProviderIPList"
        }
      ],
      "ExcludedRules": []
    }
  },
```

ルールアクションのオーバーライドの詳細については、「ルールグループ内のルールアクションのオーバーライド」を参照してください。

独自のルールグループについては、ルールグループ自体のルールアクションを変更しないでください。ルールグループルールCountアクションは、テストに必要なメトリクスやその他のアーティファクトを生成しません。さらに、ルールグループを変更すると、それを使用するすべてのウェブ ACL に影響しますが、ウェブ ACL 設定内の変更は単一のウェブ ACL にのみ影響します。

- ウェブ ACL — 新しいウェブ ACL をテストする場合は、リクエストを許可するウェブ ACL のデフォルトのアクションを設定します。これにより、トラフィックに影響を与えずにウェブ ACL を試すことができます。

一般的に、カウントモードは本番稼働よりも多くの一致が生成されます。これは、リクエストをカウントするルールがウェブ ACL によるリクエストの評価を停止しないため、ウェブ ACL で後で実行されるルールもリクエストと一致する場合があるためです。ルールアクションを本番設定に変更すると、リクエストを許可またはブロックするルールは、一致するリクエストの評価を終了します。その結果、一致するリクエストは通常、ウェブ ACL 内のより少ないルールで検査されます。ウェブリクエストの全体的な評価に対するルールアクションの影響の詳細については、「ルールアクション」を参照してください。。

これらの設定を使用すると、新しい保護によってウェブトラフィックは変更されませんが、メトリクス、ウェブ ACL ログ、およびリクエストサンプルで一致情報が生成されます。

3. ウェブ ACL をリソースに関連付ける

   ウェブ ACL がリソースに関連付けられていない場合は、関連付けます。

   ウェブ ACL とリソースの関連付けまたは関連付け解除 AWS を参照してください。

これで、ウェブ ACL を監視してチューニングする準備ができました。

モニタリングとチューニング

このセクションでは、 AWS WAF 保護機能を監視および調整する方法について説明します。

> **Note**
>
> このセクションのガイダンスに従うには、ウェブ ACL、ルール、 AWS WAF ルールグループなどの保護を作成および管理する方法を一般的に理解している必要があります。この情報は、このガイドの前のセクションで説明しています。

ウェブトラフィックとルールの一致をモニタリングして、ウェブ ACL の動作を確認します。問題が見つかった場合は、ルールを調整して修正し、モニタリングして調整を確認します。

ウェブ ACL が必要に応じてウェブトラフィックを管理するまで、次の手順を繰り返します。

モニタリングおよびチューニング

1. トラフィックとルールの一致をモニタリングする

   トラフィックがフローしていることと、テストルールで一致するリクエストが検出されていることを確認します。

   テストしている保護については、次の情報を参照してください。

   - ログ — 以下はウェブリクエストに一致するルールに関するアクセス情報です。
     - ルール - Count アクションがあるウェブ ACL のルールは、`nonTerminatingMatchingRules` にリストされます。Allow または Block を持つルールは、`terminatingRule` として一覧表示されます。CAPTCHA または Challenge を持つルールは、終了する場合と終了しない場合があり、そのためにルール一致の結果に応じて、2 つのカテゴリのいずれかに一覧表示されます。

- ルールグループ - ルールグループは `ruleGroupId` フィールドで識別され、そのルールに一致するルールはスタンドアロンルールと同様に分類されます。
- ラベル - ルールがリクエストに適用したラベルが `Labels` フィールドに一覧表示されます。

詳細については、「ログフィールド」を参照してください。

- Amazon CloudWatch メトリックス — ウェブ ACL リクエストの評価では、以下のメトリックスにアクセスできます。
  - ルール — メトリックスはルールアクションごとにグループ化されます。たとえば、Count モードでルールをテストすると、一致したルールがウェブ ACL `Count` のメトリクスとして一覧表示されます。
  - ルールグループ — ルールグループのメトリックは、ルールグループメトリックの下に一覧表示されます。
  - 別のアカウントが所有するルールグループ — ルールグループメトリックは通常、ルールグループの所有者にのみ表示されます。ただし、ルールのルールアクションをオーバーライドすると、そのルールのメトリックがウェブ ACL メトリクスの下に一覧表示されます。さらに、ルールグループによって追加されたラベルはウェブ ACL メトリクスに一覧表示されます。

    このカテゴリのルールグループはAWS のマネージドルール AWS WAFAWS Marketplace マネージドルールグループ、他のサービスによって提供されるルールグループ、、別のアカウントと共有されているルールグループです。
  - ラベル-評価中にウェブリクエストに追加されたラベルは、ウェブ ACL ラベルメトリックスに一覧表示されます。自分のルールやルールグループによって追加されたのか、別のアカウントが所有するルールグループのルールによって追加されたのかに関係なく、すべてのラベルのメトリックスにアクセスできます。

  詳細については、「ウェブ ACL のメトリクスの表示」を参照してください。
- ウェブ ACL トラフィック概要ダッシュボード — AWS WAF コンソールのウェブ ACL のページに移動して [トラフィック概要] タブを開くと、ウェブ ACL が評価したウェブトラフィックの概要にアクセスできます。

  トラフィック概要ダッシュボードには、 AWS WAF アプリケーションのウェブトラフィックを評価する際に収集される Amazon CloudWatch メトリックスの概要がほぼリアルタイムで表示されます。

  詳細については、「ウェブ ACL トラフィック概要ダッシュボード」を参照してください。

- ウェブリクエストのサンプル — ウェブリクエストのサンプルと一致するルールのアクセス情報。サンプル情報では、ウェブ ACL 内のルールのメトリクス名で一致するルールを識別します。ルールグループの場合、メトリックはルールグループ参照ステートメントを識別します。ルールグループ内のルールの場合、サンプルは `RuleWithinRuleGroup` の一致ルール名をリストします。

  詳細については、「ウェブリクエストのサンプルの表示」を参照してください。

2. 誤検知に対処するための緩和を構成する

   ルールが、本来は一致しないはずのウェブリクエストに一致して、誤検出を発生させていると判断した場合、次のオプションを使用してウェブ ACL 保護を調整して緩和できます。

   ルールの検査基準の修正

   自分のルールについては、多くの場合、ウェブリクエストを検査するために使用する設定を調整する必要があります。例としては、正規表現パターンセット内の仕様の変更、検査前にリクエストコンポーネントに適用するテキスト変換の調整、転送 IP アドレスへの切り替えなどがあります。「ルールステートメントの基本」にある問題の原因となっているルールタイプのガイダンスを参照してください。

   より複雑な問題の修正

   制御しない検査基準や複雑なルールについては、要求を明示的に許可またはブロックするルールを追加したり、問題のあるルールによる評価から要求を排除したりするなど、その他の変更が必要になることがあります。管理ルールグループでは、通常、このタイプの緩和策が必要ですが、他のルールも可能です。例としては、レートベースのルールステートメントと SQL インジェクション攻撃ルールステートメントなどがあります。

   誤検出を軽減するために行う方法は、ユースケースによって異なります。一般的なアプローチは以下のとおりです。

   - 緩和ルールの追加：新しいルールの前に実行され、誤検出の原因となっているリクエストを明示的に許可するルールを追加します。ウェブ ACL でのルールの評価順序の詳細については、「ウェブ ACL でのルールおよびルールグループの処理順序」を参照してください。

     この方法では、許可されたリクエストは保護されたリソースに送信されるため、評価のために新しいルールに到達することはありません。新しいルールが有料管理ルールグループである場合、このアプローチはルールグループの使用コストを抑えるのにも役立ちます。

- 緩和ルールで論理ルールを追加する：論理ルールステートメントを使用して、新しいルールと誤検出を除外するルールを組み合わせます。詳細については、論理ルールステートメント を参照してください。

  たとえば、リクエストのカテゴリに対して誤検出を生成する SQL インジェクションアタック match ステートメントを追加するとします。これらの要求に一致するルールを作成し、論理ルールステートメントを使用してルールを組み合わせて、両方が誤検出条件に一致せず、SQL インジェクション攻撃条件に一致するリクエストに対してのみ一致するようにします。

- スコープダウンステートメントを追加する：レートベースのステートメントおよび管理ルールグループ参照ステートメントの場合、メインステートメント内にスコープダウンステートメントを追加して、誤検出の原因となるリクエストを評価から除外します。

  スコープダウンステートメントと一致しないリクエストは、ルールグループまたはレートベースの評価に到達することはありません。スコープダウンステートメントの詳細については、「スコープダウンステートメント」を参照してください。例については、ボット管理から IP 範囲を除外するを参照してください。

- ラベルマッチルールを追加する— ラベリングを使用するルールグループでは、問題のあるルールがリクエストに適用されているラベルを特定します。ルールグループのルールをまだカウントモードに設定していない場合は、最初にカウントモードに設定する必要がある場合があります。問題のあるルールによって追加されているラベルと一致するラベル一致ルールを、ルールグループの後に実行するように追加します。ラベル一致ルールでは、ブロックするリクエストから許可するリクエストをフィルタリングできます。

  この方法を使用する場合、テストが終了したら、問題のあるルールをルールグループ内でカウントモードで保持し、カスタムラベルマッチルールをそのまま維持します。ラベル一致ステートメントの詳細については、「ラベル一致ルールステートメント」を参照してください。例については、「特定のブロックされたボットを許可する」および「ATP の例: 認証情報の不足および侵害された認証情報のカスタム処理」を参照してください。

- マネージドルールグループのバージョンの変更— バージョン対応管理ルールグループの場合、使用しているバージョンを変更します。たとえば、正常に使用していた最後の静的バージョンに戻すことができます。

  これは通常、一時的な修正です。テスト環境またはステージング環境で最新バージョンのテストを継続している間、またはプロバイダーから互換性が高いバージョンを待っている間に、本番トラフィックのバージョンを変更する場合があるかもしれません。マネージドルールグループの詳細については、「マネージドルールグループ」を参照してください。

新しいルールが必要なリクエストと一致していることに納得できたら、テストの次の段階に進み、この手順を繰り返します。テストと調整の最終段階は、本番稼働環境で実行します。

ウェブ ACL のメトリクスの表示

ウェブ ACL を 1 AWS つ以上のリソースに関連付けると、その関連付けの結果のメトリックスを Amazon CloudWatch グラフで表示できます。

AWS WAF メトリックスについて詳しくは、を参照してくださいAWS WAF メトリクスとディメンション。 CloudWatch メトリックスについて詳しくは、Amazon CloudWatch ユーザーガイドを参照してください。

ウェブ ACL 内の各ルール、および関連リソースがウェブ ACL AWS WAF に転送するすべてのリクエストについて、 CloudWatch 次の操作を実行できます。

- 1 時間前または 3 時間前のデータを表示する
- データポイント間の間隔を変更する
- 最大値、最小値、平均値、合計など、 CloudWatch データに対して実行される計算を変更できます。

> **Note**
>
> AWS WAF with CloudFront はグローバルサービスであり、メトリクスはで米国東部 (バージニア北部) AWS Management Consoleリージョンを選択した場合にのみ使用できます。別のリージョンを選択した場合、 AWS WAF CloudWatchメトリックはコンソールに表示されません。

ウェブ ACL でルールのデータを表示するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/cloudwatch/ CloudWatch でコンソールを開きます。
2. 必要に応じて、 AWS リソースがあるリージョンを変更してください。には CloudFront、米国東部 (バージニア北部) リージョンを選択します。
3. ナビゲーションペインの [Metrics] (メトリクス) で、[All metrics] (すべてのメトリクス) を選択し、[Browse] (参照) タブで `AWS::WAFV2` メトリクスを検索します。
4. データを表示するウェブ ACL のチェックボックスをオンにします。

5. 該当する設定を変更します。

[Statistic] (統計)

CloudWatch データに対して実行する計算を選択します。

[Time range] (時間範囲)

前の 1 時間のデータを表示するか、前の 3 時間のデータを表示するかを選択します。

[Period] (期間)

グラフでのデータポイント間の間隔を変更します。

[Rules] (ルール)

データを表示するルールを選択します。

> **Note**
>
> ルールの名前を変更し、その変更をルールのメトリック名に反映させたい場合は、メトリック名も更新する必要があります。 AWS WAF ルール名を変更しても、ルールのメトリック名は自動的に更新されません。ルールの JSON エディターを使用して、コンソールでルールを編集するときに、メトリック名を変更できます。API や、ウェブ ACL またはルールグループの定義に使用する JSON リストを使用して、両方の名前を変更することもできます。

次の点に注意してください。

- 最近ウェブ ACL AWS をリソースに関連付けた場合は、データがグラフに表示され、ウェブ ACL のメトリクスが利用可能なメトリクスのリストに表示されるまで、数分かかる場合があります。
- ウェブ ACL に複数のリソースを関連付けると、 CloudWatchデータにはすべてのリソースのリクエストが含まれます。
- データポイントの上にマウスカーソルを置くと、詳細情報が表示されます。
- グラフは自動的には更新されません。表示を更新するには、更新 () アイコンを選択します。

CloudWatch メトリクスの詳細については、を参照してくださいAmazon によるモニタリング CloudWatch。

ウェブ ACL トラフィック概要ダッシュボード

このセクションでは、コンソールの Web ACL トラフィック概要ダッシュボードについて説明します。 AWS WAF ウェブ ACL を 1 AWS つ以上のリソースに関連付けてウェブ ACL のメトリクスを有効にすると、コンソールのウェブ ACL の [トラフィック概要] タブに移動して、ウェブ ACL が評価するウェブトラフィックの概要にアクセスできます。 AWS WAF ダッシュボードには、 AWS WAF アプリケーションのウェブトラフィックを評価する際に収集される Amazon CloudWatch メトリックスの概要がほぼリアルタイムで表示されます。

> **Note**
>
> ダッシュボードに何も表示されない場合は、ウェブ ACL のメトリクスが有効になっていることを確認してください。

ウェブ ACL の [トラフィック概要] タブには、次のカテゴリの情報を含むタブ付きダッシュボードがあります。

- [すべてのトラフィック] — ウェブ ACL で評価されるすべてのウェブリクエスト。

  ダッシュボードではアクションの終了に重点が置かれていますが、以下の場所でカウントルールに一致するものを確認できます。

  - このダッシュボードの [上位 10 件のルール] ペイン。[カウントアクションに切り替える] をオンにすると、一致するカウントルールが表示されます。
  - ウェブ ACL ページの [サンプルリクエスト] タブ。この新しいタブには、ルールに一致したすべてのグラフが含まれています。詳細については、ウェブリクエストのサンプルの表示 を参照してください。
- [Bot Control] — Bot Control マネージドルールグループを使用してウェブ ACL で評価されるウェブリクエスト。

  ウェブ ACL でこのルールグループを使用していない場合、このタブにはウェブトラフィックのサンプリングを Bot Control ルールと照合して評価した結果が表示されます。これにより、アプリケーションが受信するボットトラフィックがわかり、この機能は無料でご利用いただけます。

このルールグループは、提供されているインテリジェントな脅威軽減オプションの一部です。AWS WAF 詳細については、AWS WAF ボットコントロールおよびAWS WAF Bot Control ルールグループを参照してください。

- アカウント乗っ取り防止 — AWS WAF 詐欺防止アカウント乗っ取り防止 (ATP) 管理ルールグループを使用してウェブ ACL が評価するウェブリクエストです。このタブは、ウェブ ACL 内のこのルールグループを使用している場合にのみ使用できます。

  ATP ルールグループは、 AWS WAF インテリジェントな脅威の軽減保護の一部です。詳細については、AWS WAF 不正防止アカウント乗っ取り防止 (ATP)およびAWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループを参照してください。
- アカウント作成詐欺防止 — 詐欺防止アカウント作成詐欺防止 (AWS WAF ACFP) 管理ルールグループを使用してウェブ ACL が評価する Web リクエスト。このタブは、ウェブ ACL 内のこのルールグループを使用している場合にのみ使用できます。

  ACFP ルールグループは、 AWS WAF のインテリジェントな脅威の緩和保護機能の一部です。詳細については、AWS WAF 不正防止アカウント作成詐欺防止 (ACFP)およびAWS WAF Fraud Control Account Creation Fraud Prevention (ACFP) ルールグループを参照してください。

ダッシュボードはウェブ ACL CloudWatch のメトリックに基づいており、グラフから内の対応するメトリックにアクセスできます。 CloudWatchBot Control のようなインテリジェントな脅威軽減ダッシュボードでは、使用されるメトリクスは主にラベルメトリクスです。

- 表示されるメトリクスのリストについては AWS WAF 、を参照してくださいAWS WAF メトリクスとディメンション。
- CloudWatch メトリックスについて詳しくは、Amazon CloudWatch ユーザーガイドを参照してください。

ダッシュボードには、選択した終了アクションと日付範囲のトラフィックパターンの概要が表示されます。インテリジェント脅威軽減ダッシュボードには、マネージドルールグループ自体が終了アクションを適用したかどうかに関係なく、対応するマネージドルールグループが評価したリクエストが含まれます。たとえば、Block が選択されている場合、[Account Takeover Prevention] ダッシュボードには、ATP マネージドルールグループによって評価されたものと、ウェブ ACL 評価中のある時点でブロックされたものの両方のすべてのウェブリクエストの情報が表示されます。リクエストは ATP マネージドルールグループ、ウェブ ACL のルールグループの後に実行されたルール、またはウェブ ACL のデフォルトアクションによってブロックされる場合があります。

ウェブ ACL のダッシュボードを表示する

ウェブ ACL ダッシュボードにアクセスし、データフィルタリング条件を設定するには、このセクションの手順に従います。最近ウェブ ACL AWS をリソースに関連付けた場合は、ダッシュボードにデータが表示されるまで数分かかる場合があります。

ダッシュボードには、ウェブ ACL に関連付けたすべてのリソースのリクエストが含まれます。

ウェブ ACL の [トラフィックの概要] ダッシュボードを表示するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。
2. ナビゲーションペインで、[ウェブ ACL] を選択し、目的のウェブ ACL を検索します。
3. ウェブ ACL を選択します。コンソールでウェブ ACL のページが表示されます。[トラフィックの概要] タブがデフォルトで選択されています。
4. [データフィルター] の設定を必要に応じて変更します。

   - [終了ルールアクション] — ダッシュボードに含める終了アクションを選択します。ウェブ ACL 評価によって選択されたアクションが適用されたウェブリクエストのメトリクスがダッシュボードに要約されます。実行可能なアクションをすべて選択すると、ダッシュボードには評価されたウェブリクエストがすべて含まれます。アクションの詳細については、「がウェブ ACL でルールとルールグループのアクション AWS WAF を処理する方法」を参照してください。
   - [時間範囲] — ダッシュボードに表示する時間間隔を選択します。過去 3 時間や過去 1 週間など、現在を基準にした時間枠を表示したり、カレンダーから絶対的な時間範囲を選択したりできます。
   - [タイムゾーン] — この設定は、絶対時間で範囲を指定した場合に適用されます。ブラウザのローカルタイムゾーンまたは UTC (協定世界時) を使用できます。

関心があるタブの情報を確認します。データフィルターの選択は、すべてのダッシュボードに適用されます。グラフペインでは、データポイントまたは領域の上にカーソルを置くと、その他の詳細が表示されます。

Count アクションルール

カウントアクションの一致に関する情報は、2 つの場所のいずれかで表示できます。

- この [トラフィック概要] タブの [すべてのトラフィック] ダッシュボードで、[トップ 10 ルール] ペインを見つけ、[カウントアクションに切り替え] をオンにします。このトグルをオンにすると、ペインには一致したルールだけでなく、一致したルールも表示されます。
- ACL の [サンプルリクエスト] タブで、[トラフィック概要] で設定した期間のルールの一致とアクションの全グラフを参照します。[サンプルリクエスト] タブについて詳細は、「ウェブリクエストのサンプルの表示」を参照してください。

Amazon CloudWatch メトリクス

ダッシュボードのグラフペインでは、 CloudWatch グラフ化されたデータのメトリックスにアクセスできます。グラフペインの上部またはペイン右上の [⋮] (垂直省略記号) ドロップダウンメニューからオプションを選択します。

ダッシュボードの更新

ダッシュボードは自動的には更新されません。表示を更新するには、更新

アイコンを選択します。

ウェブ ACL のトラフィックの概要ダッシュボード例

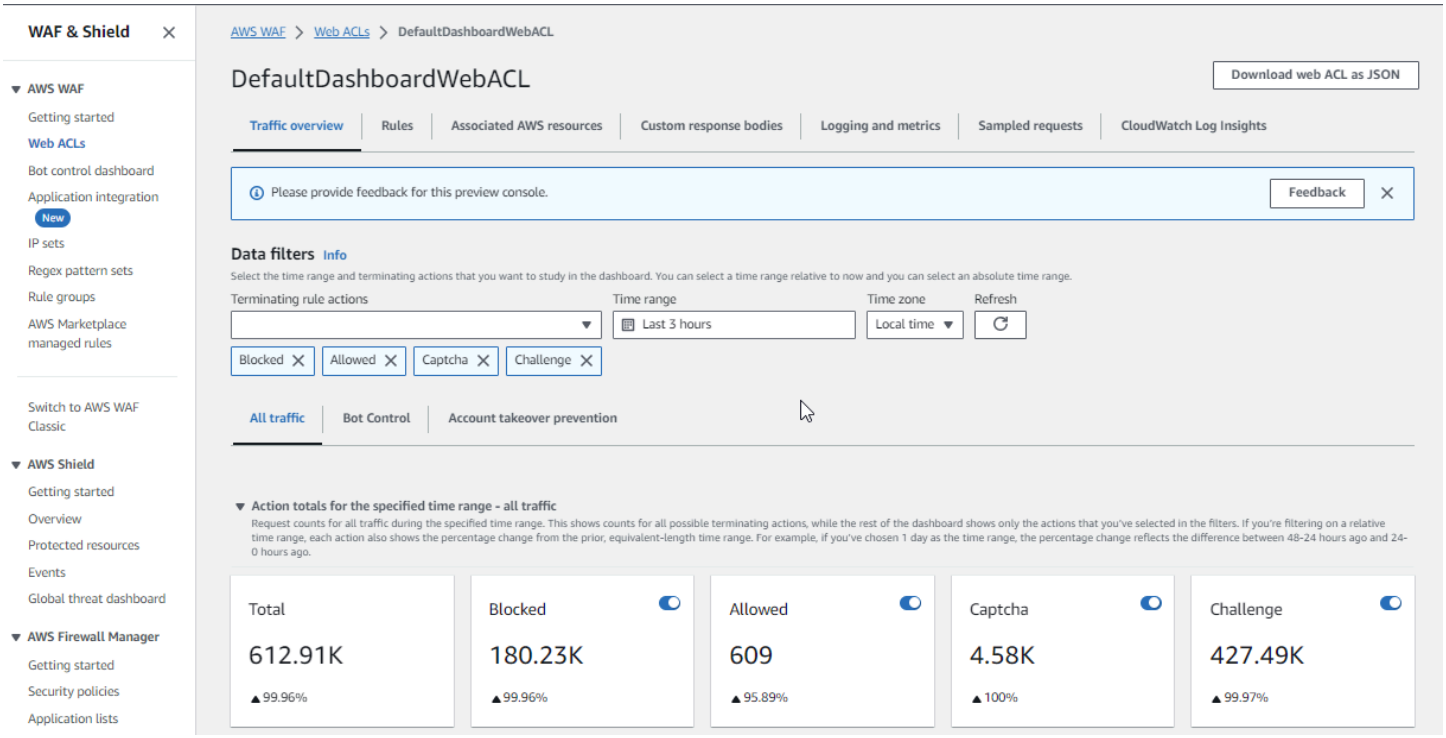
このセクションでは、ウェブ ACL のトラフィック概要ダッシュボードの画面例を示しています。

> **Note**
>
> AWS WAF アプリケーションリソースの保護にすでに使用している場合は、コンソールのページで任意のウェブ ACL のダッシュボードを確認できます。 AWS WAF 詳細については、ウェブ ACL のダッシュボードを表示する を参照してください。

画面例: データフィルターと [すべてのトラフィック] ダッシュボードのアクション数

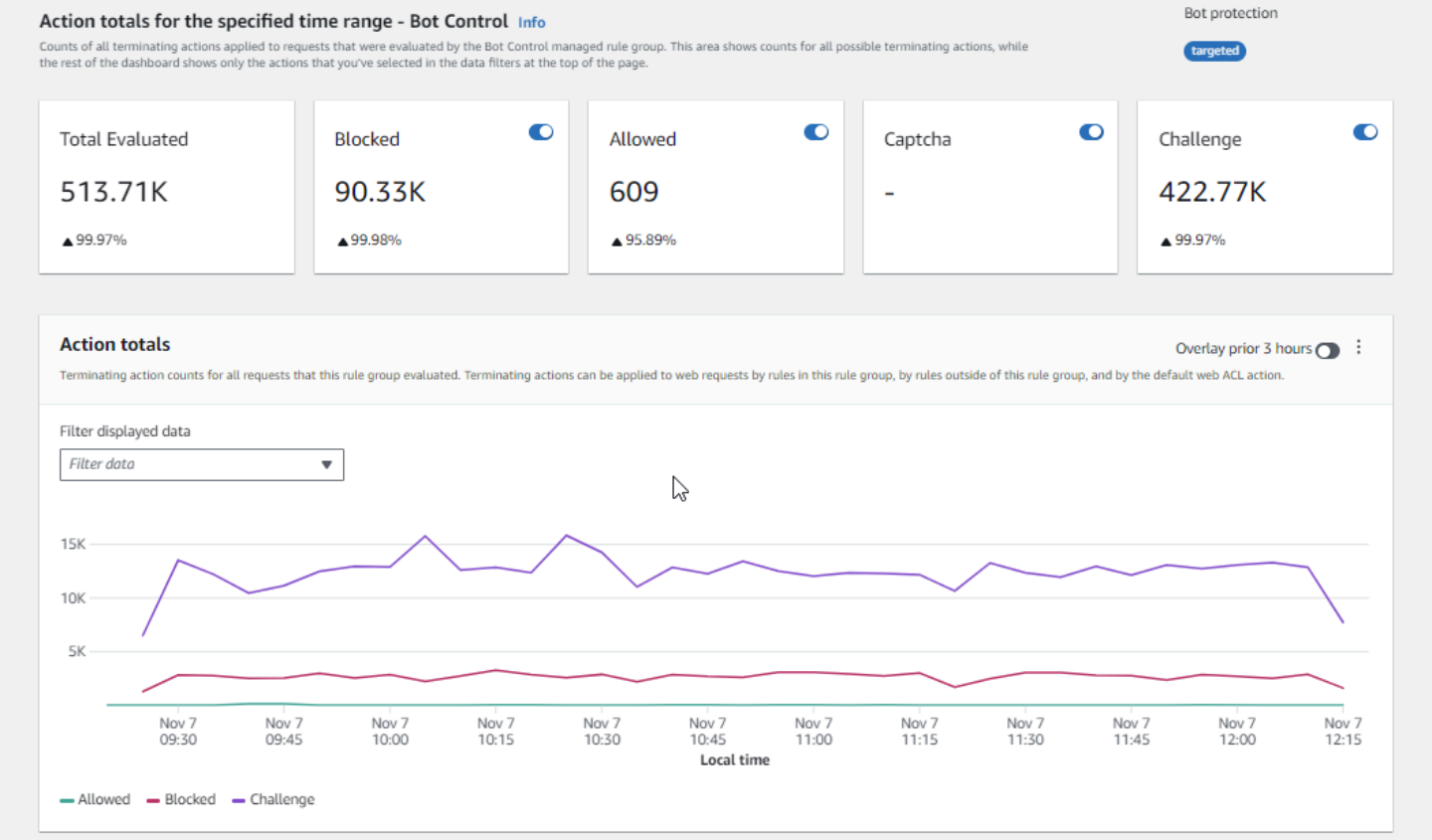
次のスクリーンショットは、[すべてのトラフィック] タブが選択された状態で、ウェブ ACL のトラフィック概要を示しています。データフィルターはデフォルトに設定されており、過去 3 時間のすべての終了アクションを示しています。

すべてのトラフィックダッシュボードには、さまざまな終了アクションのアクション合計が表示されます。各ペインにはリクエスト数が一覧表示され、過去 3 時間の時間範囲からの変化を示す上向き/下向きの矢印が表示されます。



画面の例: [Bot Control] ダッシュボードのアクション数

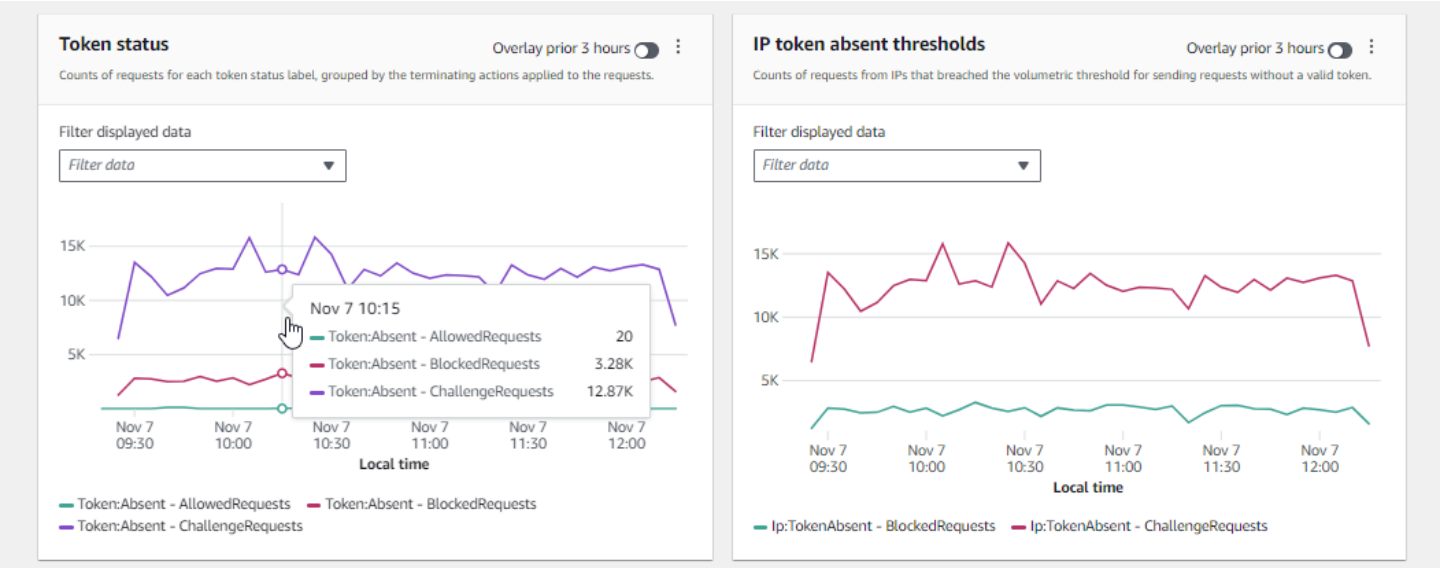
次のスクリーンショットは、Bot Control ダッシュボードのアクション数を示しています。これは時間範囲で同じ合計ペインを表示していますが、この数は Bot Control ルールグループが評価したリクエストのみを対象としています。さらに下の [アクション合計] ペインには、指定した 3 時間の時間範囲におけるアクション数が表示されます。この時間範囲では、ルールグループが評価したどのリクエストにも CAPTCHA アクションは適用されませんでした。



画面の例: [Bot Control] ダッシュボードのトークンステータスの概要グラフ

次のスクリーンショットは、Bot Control ダッシュボードに表示される 2 つの概要グラフィックを示しています。[トークンステータス] ペインには、さまざまなトークンステータスラベルの数と、リクエストに適用されたルールアクションが表示されます。[IP トークン不在しきい値] ペインには、トークンなしで大量のリクエストを送信していた IP からのリクエストのデータが表示されます。

グラフの任意の領域にカーソルを合わせると、利用可能な情報の詳細が表示されます。このスクリーンショットの [トークンステータス] ペインでは、マウスはグラフの線上にない特定の時点にカーソルを合わせているので、コンソールにはその時点のすべての線のデータが表示されます。



このセクションには、ウェブ ACL トラフィック概要ダッシュボードに表示されるトラフィックの概要の一部だけが表示されます。ウェブ ACL のダッシュボードを表示するには、コンソールでウェブ ACL のページを開きます。これを行う方法については、「ウェブ ACL のダッシュボードを表示する」でガイダンスを参照してください。

ウェブリクエストのサンプルの表示

このセクションでは、 AWS WAF コンソールのウェブ ACL サンプルリクエストタブについて説明します。このタブでは、 AWS WAF が検査したウェブリクエストのすべてのルール一致のグラフを表示できます。さらに、ウェブ ACL に対してリクエストサンプリングが有効になっている場合は、 AWS WAF が検査したウェブリクエストのサンプルのテーブルビューが表示されます。API コール を使用して、サンプリングされたリクエスト情報を取得することもできますGetSampledRequests。

リクエストのサンプルには、各ルールですべての基準に一致した最大 100 件のリクエストと、デフォルトアクションが適用された別の 100 件のリクエストが含まれます。デフォルトアクションは、すべての基準に一致するルールがなかったリクエストに適用されます。サンプルのリクエストは、過去 3 時間にコンテンツについてのリクエストを受け取ったすべての保護されたリソースからのものです。

ウェブリクエストがルール内の基準に一致し、そのルールのアクションがリクエスト評価を終了しない場合、 はウェブ ACL の後続のルールを使用してウェブリクエストの検査 AWS WAF を続行します。このため、ウェブリクエストが複数回表示される可能性があります。ルールアクションの動作については、「ルールアクション」を参照してください。

すべてのルールグラフとサンプルされたリクエストを表示するには

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/wafv2/ で AWS WAF コンソールを開きます。
2. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。
3. リクエストを表示するウェブ ACL の名前を選択します。コンソールでウェブ ACL の説明が表示され、そこで編集できます。
4. [サンプルリクエスト] タブには、次の内容が表示されます。

   - [すべてのルールグラフ] — このグラフには、指定した期間中に実行されたすべての Web リクエスト評価の一致ルールとルールアクションが表示されます。

     > **Note**
     >
     > このグラフの時間範囲は、ウェブ ACL の [トラフィック概要] タブの [データフィルター] セクションで設定されます。詳細については、「ウェブ ACL のダッシュボードを表示する」を参照してください。

   - サンプルリクエストテーブル — このテーブルには、過去 3 時間のサンプルリクエストデータが表示されます。この表には、エントリごとに以下のデータが表示されます。

     メトリクス名

     リクエストに一致したウェブ ACL 内のルールの CloudWatch メトリクス名。ウェブリクエストがウェブ ACL のルールと一致しない場合、この値はデフォルト。

     > **Note**
     >
     > ルールの名前を変更し、ルールのメトリクス名に変更を反映する場合は、メトリクス名も更新する必要があります。ルール名を変更しても、ルールのメトリクス名は自動的に更新 AWS WAF されません。ルールの JSON エディターを使用して、コンソールでルールを編集するときに、メトリック名を変更できます。API や、ウェブ ACL またはルールグループの定義に使用する JSON リストを使用して、両方の名前を変更することもできます。

[Source IP] (送信元 IP)

リクエストの発生元の IP アドレス (ビューワーが HTTP プロキシまたは Application Load Balancer を使用してリクエストを送信した場合は、そのプロキシまたは Application Load Balancer の IP アドレス)。

[URI]

URL 内でリソースを識別する部分 (`/images/daily-ad.jpg` など)。

ルールグループ内のルール

メトリック名がルールグループ参照ステートメントを識別する場合、これにより、リクエストに一致するルールグループ内のルールが識別されます。

アクション

対応するルールのアクションを示します。取りうるルールアクションの情報については、「ルールアクション」を参照してください。

時間

が保護されたリソースからリクエストを AWS WAF 受信した時刻。

ウェブリクエストのコンポーネントに関する追加情報を表示するには、リクエストの行にある URI の名前を選択します。

本番環境で保護の有効化

本番環境でのテストとチューニングの最終段階が終了したら、本番モードで保護を有効にします。

> **本番稼働トラフィックのリスク**
>
> 本番稼働トラフィックにウェブ ACL 実装をデプロイする前に、トラフィックへの潜在的な影響に慣れるまで、テスト環境でテストおよびチューニングします。また、本番稼働用トラフィックの保護を有効にする前に、本番稼働用トラフィックでカウントモードでテストおよびチューニングします。

> **Note**
>
> このセクションのガイダンスに従うには、ウェブ ACL、ルール、 AWS WAF ルールグループなどの保護を作成および管理する方法を一般的に理解している必要があります。この情報は、このガイドの前のセクションで説明しています。

これらのステップを最初にテスト環境で実行し、次に本番環境で実行します。

AWS WAF 本番環境で保護を有効にする

1. 本番環境保護に切り替える

   ウェブ ACL を更新し、本番環境の設定を切り替えます。

   a. 不要なテストルールをすべて削除する

      本番環境では必要ないテストルールを追加した場合は、それらを削除します。ラベル一致ルールを使用して管理ルールグループルールの結果をフィルタリングする場合は、必ずそれらのルールをそのままにしておいてください。

   b. 本番用アクションに切り替える

      新しいルールのアクション設定を、意図したプロダクション設定に変更します。

      - ウェブ ACL で定義されているルール— ウェブ ACL のルールを編集し、アクションをから変更します。Count彼らのプロダクションアクションに。
      - ルールグループ— ルールグループのウェブ ACL 設定で、独自のアクションを使用するようにルールを切り替えるか、Countテストおよびチューニングのアクティビティの結果に応じて、アクションオーバーライド。ラベル一致ルールを使用してルールグループルールの結果をフィルタリングする場合は、必ずそのルールのオーバーライドをそのまま残してください。

        ルールのアクションの使用に切り替えるには、ウェブ ACL 設定で、ルールグループのルールステートメントを編集し、ルールの Count オーバーライドを削除します。JSON でウェブ ACL を管理する場合、ルールグループ参照ステートメントで `RuleActionOverrides` リストからルールのエントリを削除します。
      - ウェブ ACL— ウェブ ACL のデフォルトアクションをテスト用に変更した場合は、本番用の設定に切り替えます。

これらの設定により、意図したとおりに新しい保護がウェブトラフィックを管理します。

ウェブ ACL を保存すると、Web ACL が関連付けられているリソースは本番設定を使用します。

2. モニタリングおよびチューニング

ウェブリクエストが希望どおりに処理されていることを確認するには、新しい機能を有効にした後、トラフィックを注意深く監視します。チューニング作業で監視していたカウントアクションではなく、本番ルールアクションのメトリクスとログを監視します。ウェブトラフィックの変化に適応するために、必要に応じて動作を監視し、調整してください。

Amazon AWS WAF CloudFront の機能との連携方法

ウェブ ACL を作成するときに、 CloudFront AWS WAF 検査するディストリビューションを 1 つ以上指定できます。 AWS WAF ウェブ ACL で指定した条件に基づいて、それらのディストリビューションのウェブリクエストの検査と管理を開始します。 CloudFront には、 AWS WAF 機能を強化する機能がいくつか用意されています。この章では、CloudFront CloudFront AWS WAF より効率的に連携して機能するように設定できるいくつかの方法について説明します。

トピック

- AWS WAF CloudFront カスタムエラーページとの併用
- CloudFront独自の HTTP サーバー上で動作するアプリケーションに AWS WAF with を使用する
- CloudFrontに応答する HTTP メソッドの選択

AWS WAF CloudFront カスタムエラーページとの併用

デフォルトでは、 AWS WAF 指定した条件に基づいてウェブリクエストをブロックすると、HTTP 403（Forbidden）ステータスコードがに返され CloudFront、 CloudFront そのステータスコードがビューアに返されます。ビューワーには、次のような簡潔で特に書式設定されていないデフォルトメッセージが表示されます。

```
Forbidden: You don't have permission to access /myfilename.html on this server.
```

AWS WAF ウェブ ACL ルール内のこの動作は、カスタムレスポンスを定義することで無効にできます。 AWS WAF ルールを使用してレスポンス動作をカスタマイズする方法の詳細については、を参照してくださいBlock アクションのカスタムレスポンス。

> **Note**
>
> AWS WAF ルールを使用してカスタマイズしたレスポンスは、 CloudFront カスタムエラーページで定義したレスポンス仕様よりも優先されます。

Web サイトの他の部分と同じ形式を使用してカスタムエラーメッセージを表示したい場合は、カスタムエラーメッセージを含むオブジェクト (HTML ファイルなど) CloudFront をビューアに返すように設定できます。 CloudFront

> **Note**
>
> CloudFront オリジンから返される HTTP ステータスコード 403 と、 AWS WAF リクエストがブロックされたときに返される HTTP ステータスコード 403 を区別できません。つまり、HTTP ステータスコード 403 のさまざまな原因に基づいて、異なるカスタムエラーページを返すことはできません。

CloudFront カスタムエラーページの詳細については、Amazon CloudFront 開発者ガイドの「カスタムエラーレスポンスの生成」を参照してください。

CloudFront独自の HTTP サーバー上で動作するアプリケーションに AWS WAF with を使用する

AWS WAF とを使用すると CloudFront、Amazon Elastic Compute Cloud (Amazon EC2) で実行されているウェブサーバーでも、プライベートに管理されているウェブサーバーでも、任意の HTTP ウェブサーバーで実行されているアプリケーションを保護できます。また、 CloudFrontと自分のウェブサーバー間、およびビューワーとの間で HTTPS CloudFront を要求するように設定することもできます。 CloudFront

CloudFront と自分のウェブサーバーとの間で HTTPS を要求する

CloudFront 独自のウェブサーバーとの間で HTTPS を要求するには、 CloudFront カスタムオリジン機能を使用して、特定のオリジンのオリジンプロトコルポリシーとオリジンドメイン名の設定を

構成できます。 CloudFront 構成では、 CloudFront オリジンからオブジェクトを取得するときに使用するポートとプロトコルとともに、サーバーの DNS 名を指定できます。また、カスタムオリジンサーバー上の SSL/TLS 証明書が、設定したオリジンドメイン名と一致することを確認する必要もあります。外部で独自の HTTP Web サーバーを使用する場合は AWS、Comodo、 DigiCertまたは Symantec などの信頼できるサードパーティの認証機関 (CA) によって署名された証明書を使用する必要があります。 CloudFrontと独自のウェブサーバー間の通信に HTTPS を要求する方法の詳細については、Amazon CloudFront 開発者ガイドの「 CloudFront とカスタムオリジン間の通信に HTTPS を要求する」を参照してください。

ビューアーと間の HTTPS の要求 CloudFront

ビューアとの間で HTTPS を要求するには CloudFront、 CloudFrontディストリビューション内の 1 つ以上のキャッシュ動作の Viewer プロトコルポリシーを変更できます。 CloudFront視聴者間でのHTTPSの使用の詳細については CloudFront、 CloudFront Amazon開発者ガイドのトピック「視聴者間の通信にHTTPSを要求する」を参照してください。視聴者が独自のドメイン名 (例:https://www.mysite.com) を使用して HTTPS CloudFront 経由でディストリビューションに接続できるように、独自の SSL 証明書を持ち込むこともできます。詳細については、Amazon CloudFront 開発者ガイドのトピック「代替ドメイン名と HTTPS の設定」を参照してください。

CloudFrontに応答する HTTP メソッドの選択

Amazon CloudFront ウェブディストリビューションを作成するときは、 CloudFront 処理してオリジンに転送する HTTP メソッドを選択します。次のオプションから選択できます。

- **GET**, **HEAD** — オリジンからのオブジェクトの取得、 CloudFront またはオブジェクトヘッダーの取得にのみ使用できます。
- **GET**,**HEAD**, **OPTIONS** — オリジンからのオブジェクトの取得、オブジェクトヘッダーの取得、CloudFront またはオリジンサーバーがサポートするオプションのリストの取得にのみ使用できます。
- **GET**、**HEADOPTIONS**、**PUT**、**POSTPATCH**、**DELETE** — オブジェクトの取得、追加、更新、削除、およびオブジェクトヘッダーの取得に使用できます CloudFront 。また、ウェブフォームからのデータの送信など、その他の POST オペレーションも実行できます。

で説明されているように、 AWS WAF バイトマッチルールステートメントを使用して HTTP メソッドに基づいてリクエストを許可または拒否することもできます文字列一致ルールステートメント。CloudFront GETやなどをサポートするメソッドを組み合わせて使用する場合はHEAD、 AWS WAF他のメソッドを使用するリクエストをブロックするように設定する必要はありません。、、など、

CloudFront サポートされていないメソッドの組み合わせを許可したい場合はGETHEAD、 CloudFront すべてのメソッドに応答するように設定し、 AWS WAF を使用して他のメソッドを使用するリクエストをブロックできます。POST

CloudFront 応答するメソッドの選択の詳細については、Amazon CloudFront 開発者ガイドの「ウェブディストリビューションを作成または更新するときに指定する値」の「許可される HTTP メソッド」を参照してください。

AWS WAF サービスの利用におけるセキュリティ

AWS クラウドセキュリティは最優先事項です。 AWS 顧客は、最もセキュリティに敏感な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の恩恵を受けることができます。

> **Note**
>
> このセクションでは、 AWS WAF Web ACL やルールグループなど、 AWS AWS WAF AWS サービスとそのリソースを使用する際の標準的なセキュリティガイダンスを提供します。 AWS を使用してリソースを保護する方法については AWS WAF、 AWS WAF ガイドの残りの部分を参照してください。

AWS セキュリティはユーザーとユーザー間で共有される責任です。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して説明しています。

- クラウドのセキュリティ — AWS AWS AWS クラウドクラウド内でサービスを実行するインフラストラクチャを保護する責任があります。 AWS また、安全に使用できるサービスも提供します。セキュリティの有効性は、AWS コンプライアンスプログラムの一環として、サードパーティーの審査機関によって定期的にテストおよび検証されています。に適用されるコンプライアンスプログラムについて詳しくは AWS WAF、「AWS コンプライアンスプログラム別の対象サービス」を参照してください。
- クラウドにおけるセキュリティ — お客様の責任は、 AWS 使用する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、組織の要件、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、利用時に責任分担モデルを適用する方法を理解するのに役立ちます AWS WAF。以下のトピックでは、 AWS WAF 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構

成する方法を示しています。また、 AWS AWS WAF リソースの監視と保護に役立つ他のサービスの使い方についても学びます。

トピック

- でのデータ保護 AWS WAF
- の Identity and Access Management AWS WAF
- ロギングとモニタリング AWS WAF
- のコンプライアンス検証 AWS WAF
- のレジリエンス AWS WAF
- AWS WAF内のインフラストラクチャセキュリティ

でのデータ保護 AWS WAF

AWS のデータ保護には、https://aws.amazon.com/compliance/shared-responsibility-model/、 (責任分担モデル) が適用されます AWS WAF。このモデルで説明したように、 AWS は、 AWS クラウドすべてを稼働させ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。お客様は、このインフラストラクチャでホストされてい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、「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。欧州でのデータ保護の詳細については、「AWS セキュリティブログ」に投稿された「AWS 責任共有モデルおよび GDPR」のブログ記事を参照してください。

データ保護のため、 AWS アカウント 認証情報を保護し、 AWS IAM Identity Center 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。こうすると、それぞれのジョブを遂行するために必要なアクセス許可のみを各ユーザーに付与できます。また、以下の方法でデータを保護することをお勧めします。

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用してリソースと通信します。 AWS TLS 1.2、できれば TLS 1.3 が必要です。
- を使用して API とユーザーアクティビティの口ギングを設定します。 AWS CloudTrail
- AWS 暗号化ソリューションと、 AWS のサービスその中に含まれるデフォルトのセキュリティコントロールをすべて使用してください。
- Amazon Macie などの高度なマネージド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 を介してアクセスするときに FIPS 140-2 で検証された暗号モジュールが必要な場合は、FIPS エンドポイントを使用してください。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) 140-2」を参照してください。

お客様の E メールアドレスなどの機密情報やセンシティブ情報は、タグや名前フィールドなどの自由形式のフィールドに配置しないことを強くお勧めします。これには、コンソール、API、 AWS WAF または SDK を操作する場合や、 AWS のサービス その他の方法でコンソール、API、SDK を使用する場合も含まれます。 AWS CLI AWS 名前に使用する自由記述のテキストフィールドやタグに入力したデータは、課金や診断ログに使用される場合があります。外部サーバーへの URL を提供する場合は、そのサーバーへのリクエストを検証するための認証情報を URL に含めないように強くお勧めします。

AWS WAF ウェブ ACL、ルールグループ、IP セットなどのエンティティは、中国 (北京) や中国 (寧夏) など、暗号化が利用できない特定の地域を除き、保存時に暗号化されます。リージョンごとに一意の暗号化キーが使用されます。

AWS WAF リソースの削除

AWS WAFで作成したリソースは削除できます。次のセクションの各リソースタイプのガイダンスを参照してください。

- ウェブ ACL の削除
- ルールグループの削除
- IP セットの削除
- 正規表現パターンセットの削除

の Identity and Access Management AWS WAF

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 (サインイン) し、誰に AWS WAF リソースの使用を承認する (アクセス許可を付与する) かを制御します。IAM は、追加料金なしで AWS のサービス 使用できる です。

トピック

- 対象者

- 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
- が IAM と AWS WAF 連携する方法
- AWS WAFのアイデンティティベースのポリシーの例
- AWS の マネージドポリシー AWS WAF
- AWS WAF ID とアクセスのトラブルシューティング
- のサービスにリンクされたロールの使用 AWS WAF

対象者

AWS Identity and Access Management (IAM) の使用方法は、 で行う作業によって異なります AWS WAF。

サービスユーザー – AWS WAF サービスを使用してジョブを実行する場合、管理者から必要な認証情報とアクセス許可が与えられます。さらに多くの AWS WAF 機能を使用して作業を行う場合は、追加のアクセス許可が必要になることがあります。アクセスの管理方法を理解すると、管理者から適切な権限をリクエストするのに役に立ちます。 AWS WAF機能にアクセスできない場合は、「AWS WAF ID 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。

サービス管理者 – 社内の AWS WAF リソースを担当している場合は、通常、 へのフルアクセスがあります AWS WAF。サービスユーザーがどの AWS WAF 機能やリソースにアクセスするかを決めるのは管理者の仕事です。その後、IAM 管理者にリクエストを送信して、サービスユーザーの権限を変更する必要があります。このページの情報を点検して、IAM の基本概念を理解してください。会社で IAM を で使用する方法の詳細については、 AWS WAF「」を参照してくださいが IAM と AWS WAF 連携する方法。

IAM 管理者 - 管理者は、 AWS WAFへのアクセスを管理するポリシーの書き込み方法の詳細について確認する場合があります。IAM で使用できる AWS WAF 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、「」を参照してくださいAWS WAFのアイデンティティベースのポリシーの例。

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。として、IAM ユーザーとして AWS アカウントのルートユーザー、または IAM ロールを引き受けて認証（ にサインイン AWS) される必要があります。

ID ソースを介して提供された認証情報を使用して、フェデレーティッド ID AWS として にサインインできます。 AWS IAM Identity Center（ IAM Identity Center) ユーザー、会社のシングルサイン

オン認証、Google または Facebook の認証情報は、フェデレーティッド ID の例です。フェデレーティッド ID としてサインインする場合、IAM ロールを使用して、前もって管理者により ID フェデレーションが設定されています。フェデレーション AWS を使用して にアクセスすると、間接的にロールを引き受けることになります。

ユーザーのタイプに応じて、 AWS Management Console または AWS アクセスポータルにサインインできます。へのサインインの詳細については AWS、「 ユーザーガイド」の「 にサインインする方法 AWS アカウントAWS サインイン 」を参照してください。

AWS プログラムで にアクセスする場合、 は Software Development Kit (SDK) とコマンドラインインターフェイス (CLI) AWS を提供し、認証情報を使用してリクエストに暗号で署名します。 AWS ツールを使用しない場合は、リクエストに自分で署名する必要があります。推奨される方法を使用してリクエストを自分で署名する方法の詳細については、IAM ユーザーガイドの API AWS リクエストの署名を参照してください。

使用する認証方法を問わず、追加セキュリティ情報の提供をリクエストされる場合もあります。例えば、 AWS では、アカウントのセキュリティを強化するために多要素認証 (MFA) を使用することをお勧めします。詳細については、「AWS IAM Identity Center ユーザーガイド」の「Multi-factor authentication」(多要素認証) および「IAM ユーザーガイド」の「AWSでの多要素認証 (MFA) の使用」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、アカウント内のすべての 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 1 つのサインインアイデンティティから始めます。この ID は AWS アカウント ルートユーザーと呼ばれ、アカウントの作成に使用した E メールアドレスとパスワードでサインインすることでアクセスできます。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。ルートユーザーの認証情報は保護し、ルートユーザーでしか実行できないタスクを実行するときに使用します。ルートユーザーとしてサインインする必要があるタスクの完全なリストについては、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。

フェデレーティッドアイデンティティ

ベストプラクティスとして、管理者アクセスを必要とするユーザーを含む人間のユーザーに、一時的な認証情報を使用して にアクセスするための ID プロバイダーとのフェデレーションの使用を要求 AWS のサービス します。

フェデレーティッド ID は、エンタープライズユーザーディレクトリ、ウェブ ID プロバイダー、AWS Directory Service、アイデンティティセンターディレクトリ、または ID ソースを通じて提供された認証情報 AWS のサービス を使用して にアクセスするユーザーです。フェデレーティッド ID が

にアクセスすると AWS アカウント、ロールが引き受けられ、ロールは一時的な認証情報を提供します。

アクセスを一元管理する場合は、 AWS IAM Identity Centerを使用することをお勧めします。IAM Identity Center でユーザーとグループを作成することも、独自の ID ソース内のユーザーとグループのセットに接続して同期して、すべての AWS アカウント とアプリケーションで使用することもできます。IAM Identity Center の詳細については、「AWS IAM Identity Center ユーザーガイド」の「What is IAM Identity Center?」(IAM Identity Center 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、単一のユーザーまたはアプリケーションに対して特定のアクセス許可 AWS アカウント を持つ 内のアイデンティティです。可能であれば、パスワードやアクセスキーなどの長期的な認証情報を保有する IAM ユーザーを作成する代わりに、一時的な認証情報を使用することをお勧めします。ただし、IAM ユーザーでの長期的な認証情報が必要な特定のユースケースがある場合は、アクセスキーをローテーションすることをお勧めします。詳細については、IAM ユーザーガイドの長期的な認証情報を必要とするユースケースのためにアクセスキーを定期的にローテーションするを参照してください。

IAM グループは、IAM ユーザーの集団を指定するアイデンティティです。グループとしてサインインすることはできません。グループを使用して、複数のユーザーに対して一度に権限を指定できます。多数のユーザーグループがある場合、グループを使用することで権限の管理が容易になります。例えば、IAMAdminsという名前のグループを設定して、そのグループにIAM リソースを管理する許可を与えることができます。

ユーザーは、ロールとは異なります。ユーザーは 1 人の人または 1 つのアプリケーションに一意に関連付けられますが、ロールはそれを必要とする任意の人が引き受けるようになっています。ユーザーには永続的な長期の認証情報がありますが、ロールでは一時的な認証情報が提供されます。詳細については、「IAM ユーザーガイド」の「IAM ユーザー (ロールではなく) の作成が適している場合」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 AWS アカウント を持つ 内のアイデンティティです。これは IAM ユーザーに似ていますが、特定のユーザーには関連付けられていません。ロール を切り替える AWS Management Console ことで、 で IAM ロールを一時的に引き受けることができます。ロールを引き受けるには、 または AWS API AWS CLI オペレーションを呼び出すか、カスタム URL を使用します。ロールを使用する方法の詳細については、「IAM ユーザーガイド」の「IAM ロールの使用」を参照してください。

IAM ロールと一時的な認証情報は、次の状況で役立ちます:

- フェデレーションユーザーアクセス – フェデレーティッド ID に許可を割り当てるには、ロールを作成してそのロールの許可を定義します。フェデレーティッド ID が認証されると、その ID はロールに関連付けられ、ロールで定義されている許可が付与されます。フェデレーションの詳細については、「IAM ユーザーガイド」の「Creating a role for a third-party Identity Provider」(サードパーティーアイデンティティプロバイダー向けロールの作成) を参照してください。IAM Identity Center を使用する場合は、許可セットを設定します。アイデンティティが認証後にアクセスできるものを制御するため、IAM Identity Center は、権限セットを IAM のロールに関連付けます。アクセス許可セットの詳細については、「AWS IAM Identity Center ユーザーガイド」の「アクセス許可セット」を参照してください。
- 一時的な IAM ユーザー権限 - IAM ユーザーまたはロールは、特定のタスクに対して複数の異なる権限を一時的に IAM ロールで引き受けることができます。
- クロスアカウントアクセス - IAM ロールを使用して、自分のアカウントのリソースにアクセスすることを、別のアカウントの人物 (信頼済みプリンシパル) に許可できます。クロスアカウントアクセス権を付与する主な方法は、ロールを使用することです。ただし、一部の では AWS のサービス、 (ロールをプロキシとして使用する代わりに) ポリシーをリソースに直接アタッチできます。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。
- クロスサービスアクセス — 一部の は、他の の機能 AWS のサービス を使用します AWS のサービス。例えば、あるサービスで呼び出しを行うと、通常そのサービスによって Amazon EC2 でアプリケーションが実行されたり、Amazon S3 にオブジェクトが保存されたりします。サービスでは、呼び出し元プリンシパルの許可、サービスロール、またはサービスリンクロールを使用してこれを行う場合があります。
  - 転送アクセスセッション (FAS) – IAM ユーザーまたはロールを使用して でアクションを実行する場合 AWS、ユーザーはプリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストのリクエストと組み合わせて使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。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。
  - サービスロール - 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 のサービスにアクセス許可を委任するロールの作成」を参照してください。

- サービスにリンクされたロール – 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、サービスにリンクされた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。

- Amazon EC2 で実行されているアプリケーション – IAM ロールを使用して、EC2 インスタンスで実行され、 AWS CLI または AWS API リクエストを行うアプリケーションの一時的な認証情報を管理できます。これは、EC2 インスタンス内でのアクセスキーの保存に推奨されます。 AWS ロールを EC2 インスタンスに割り当て、そのすべてのアプリケーションで使用できるようにするには、インスタンスにアタッチされたインスタンスプロファイルを作成します。インスタンスプロファイルにはロールが含まれ、EC2 インスタンスで実行されるプログラムは一時的な認証情報を取得できます。詳細については、IAM ユーザーガイドのAmazon EC2 インスタンスで実行されるアプリケーションに IAM ロールを使用して許可を付与するを参照してください。

IAM ロールと IAM ユーザーのどちらを使用するかについては、IAM ユーザーガイドの(IAM ユーザーではなく) IAM ロールをいつ作成したら良いのか?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、 AWS ID 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ときにアクセス許可を定義する のオブジェクトです。 は、プリンシパル (ユーザー、ルートユーザー、またはロールセッション) AWS 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。ポリシーでの権限により、リクエストが許可されるか拒否されるかが決まります。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構造と内容の詳細については、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 AWS JSON 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んなリソースにどんな条件でアクションを実行できるかということです。

デフォルトでは、ユーザーやロールに権限はありません。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。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き継ぐことができます。

IAM ポリシーは、オペレーションの実行方法を問わず、アクションの許可を定義します。例えば、`iam:GetRole`アクションを許可するポリシーがあるとします。そのポリシーを持つユーザーは、 AWS Management Console、、 AWS CLIまたは AWS API からロール情報を取得できます。

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ポリシーは、IAM ユーザーグループ、ユーザーのグループ、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 ポリシーの作成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、さらにインラインポリシーまたはマネージドポリシーに分類できます。インラインポリシーは、単一のユーザー、グループ、またはロールに直接埋め込まれています。管理ポリシーは、 内の複数のユーザー、グループ、ロールにアタッチできるスタンドアロンポリシーです AWS アカウント。管理ポリシーには、 AWS 管理ポリシーとカスタマー管理ポリシーが含まれます。マネージドポリシーまたは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 ユーザーガイドのマネージドポリシーとインラインポリシーの比較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。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制御できます。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は、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。

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。

アクセスコントロールリスト (ACL)

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。

Amazon S3、 AWS WAF、および Amazon VPC は、ACLs。ACL の詳細については、Amazon Simple Storage Service デベロッパ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。

その他のポリシータイプ

AWS は、一般的ではない追加のポリシータイプをサポートします。これらのポリシータイプでは、より一般的なポリシータイプで付与された最大の権限を設定できます。

- アクセス許可の境界 - アクセス許可の境界は、アイデンティティベースのポリシーによって IAM エンティティ (IAM ユーザーまたはロール) に付与できる権限の上限を設定する高度な機能です。エンティティにアクセス許可の境界を設定できます。結果として得られる権限は、エンティティのアイデンティティベースポリシーとそのアクセス許可の境界の共通部分になります。`Principal` フィールドでユーザーまたはロールを指定するリソースベースのポリシーでは、アクセス許可の境界は制限されません。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。アクセス許可の境界の詳細については、IAM ユーザーガイドのIAM エンティティのアクセス許可の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 (SCPs) – SCPs は、 の組織または組織単位 (OU) に対する最大アクセス許可を指定する JSON ポリシーです AWS Organizations。 AWS Organizations は、 AWS アカウント ビジネスが所有する複数の をグループ化して一元管理するサービスです。組織内のすべての機能を有効にすると、サービスコントロールポリシー (SCP) を一部またはすべてのアカウントに適用できます。SCP は、各 を含むメンバーアカウントのエンティティのアクセス許可を制限します AWS アカウントのルートユーザー。Organizations と SCP の詳細については、AWS Organizations ユーザーガイドの「SCP の仕組み」を参照してください。
- セッションポリシー - セッションポリシーは、ロールまたはフェデレーションユーザーの一時的なセッションをプログラムで作成する際にパラメータとして渡す高度なポリシーです。結果としてセッションの権限は、ユーザーまたはロールのアイデンティティベースポリシーとセッションポリシーの共通部分になります。また、リソースベースのポリシーから権限が派生する場合もあります。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。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権限を理解するのがさらに難しくなります。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。

が IAM と AWS WAF 連携する方法

IAM を使用して へのアクセスを管理する前に AWS WAF、 で使用できる IAM 機能について学びます AWS WAF。

で使用できる IAM の機能 AWS WAF

IAM 機能	AWS WAF 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	あり
ポリシーアクション	あり
ポリシーリソース	はい
ポリシー条件キー (サービス固有)	はい
ACL	なし
ABAC (ポリシー内のタグ)	部分的
一時的な認証情報	あり
転送アクセスセッション (FAS)	あり
サービスロール	あり
サービスリンクロール	あり

AWS WAF 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を把握するには、「IAM ユーザーガイド」の「IAM AWS と連携する のサービス」を参照してください。

のアイデンティティベースのポリシー AWS WAF

アイデンティティベースポリシーをサポートする	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 ポリシーの作成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。プリンシパルは、それが添付されているユーザーまたはロールに適用されるため、アイデンティティベースのポリシーでは指定できません。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。

AWS WAF 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、「」を参照してくださいAWS WAFのアイデンティティベースのポリシーの例。

内のリソースベースのポリシー AWS WAF

リソースベースのポリシーのサポート	あり

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制御できます。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は、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、アカウント全体、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。リソースベースのポリシーにクロスアカウントのプリンシパルを追加しても、信頼関係は半分しか確立されない点に注意してください。プリンシパルとリソースが異なる にある場合 AWS アカウント、信頼されたアカウントの IAM 管理者は、プリンシパルエンティティ (ユーザーまたはロール) にリソースへのアクセス許可も付与する必要があります。IAM 管理者は、アイデンティティベースのポリシーをエンティティにアタッチすることで権限を付与します。ただし、リソースベースのポリシーで、同じアカウントのプリンシパルへのアクセス権が付与されている場合は、アイデンティティベースのポリシーをさらに付与する必要はありません。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。

AWS WAF はリソースベースのポリシーを使用して、アカウント間でのルールグループの共有をサポートします。リソースベースのポリシー設定を AWS WAF API コールPutPermissionPolicy、または同等の CLI または SDK コールに提供することで、所有するルールグループを別の AWS アカウントと共有します。その他の利用可能な言語の例やドキュメントへのリンクなど、追加情報については、 AWS WAF API リファレンスPutPermissionPolicyの「」を参照してください。この機能は、コンソールや AWS CloudFormationなどの他の方法では使用できません。

のポリシーアクション AWS WAF

ポリシーアクションに対するサポート	あり

管理者は AWS JSON 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んなリソースにどんな条件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。ポリシーアクションの名前は通常、関連付けられた AWS API オペレーションと同じです。一致する API オペレーションのない許可のみのアクションなど、いくつかの例外があります。また、ポリシーに複数のアクションが必要なオペレーションもあります。これらの追加アクションは、依存アクションと呼ばれます。

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権限を付与するポリシーで使用されます。

それぞれの AWS WAF アクションとアクセス許可のリストを確認するには、「サービス認証リファレンス」のAWS WAF V2 で定義されるアクション」を参照してください。

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 AWS WAF を使用します。

```
wafv2
```

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": [
      "wafv2:action1",
      "wafv2:action2"
         ]
```

ワイルドカード (*) を使用すると、複数のアクションを指定することができます。例えば、 で始 AWS WAF まる のすべてのアクションを指定するにはList、次のアクションを含めます。

```
"Action": "wafv2:List*"
```

AWS WAF 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、「」を参照してくださいAWS WAFのアイデンティティベースのポリシーの例。

追加のアクセス許可設定が必要なアクション

一部のアクションには、「サービス認証リファレンス」のAWS WAF V2 で定義されるアクション」で完全に説明できないアクセス許可が必要です。このセクションは、追加のアクセス許可に関する情報を説明します。

トピック

- AssociateWebACL のアクセス権限
- DisassociateWebACL のアクセス権限
- GetWebACLForResource のアクセス権限
- ListResourcesForWebACL のアクセス権限

AssociateWebACL のアクセス権限

このセクションでは、 AWS WAF アクションAssociateWebACL を使用してウェブ ACL をリソースに関連付けるために必要なアクセス許可の一覧を示します。

Amazon CloudFront ディストリビューションでは、このアクションの代わりに、 CloudFront アクション を使用しますUpdateDistribution。詳細については、「Amazon CloudFront API リファレンスUpdateDistribution」の「」を参照してください。

Amazon API Gateway REST API

REST API リソースタイプSetWebACLで API Gateway を呼び出し、ウェブ ACL で を呼び AWS WAF AssociateWebACL出すアクセス許可が必要です。

```
{
    "Sid": "AssociateWebACL1",
    "Effect": "Allow",
    "Action": [
        "wafv2:AssociateWebACL"
```

```
    ],
    "Resource": [
        "arn:aws:wafv2:region:account-id:regional/webacl/*/*"
    ]
},
{
    "Sid": "AssociateWebACL2",
    "Effect": "Allow",
    "Action": [
        "apigateway:SetWebACL"
    ],
    "Resource": [
        "arn:aws:apigateway:*::/restapis/*/stages/*"
    ]
}
```

Application Load Balancer

Application Load Balancer リソースタイプで `elasticloadbalancing:SetWebACL`アクションを呼び出し、ウェブ ACL `AssociateWebACL`で を呼び AWS WAF 出すアクセス許可が必要です。

```
{
    "Sid": "AssociateWebACL1",
    "Effect": "Allow",
    "Action": [
        "wafv2:AssociateWebACL"
    ],
    "Resource": [
        "arn:aws:wafv2:region:account-id:regional/webacl/*/*"
    ]
},
{
    "Sid": "AssociateWebACL2",
    "Effect": "Allow",
    "Action": [
        "elasticloadbalancing:SetWebACL"
    ],
    "Resource": [
        "arn:aws:elasticloadbalancing:*:account-id:loadbalancer/app/*/*"
    ]
}
```

AWS AppSync GraphQL API

GraphQL API リソースタイプで を呼び出し AWS AppSync SetWebACL、ウェブ ACL で を呼び AWS WAF AssociateWebACL出すアクセス許可が必要です。

```
{
    "Sid": "AssociateWebACL1",
    "Effect": "Allow",
    "Action": [
        "wafv2:AssociateWebACL"
    ],
    "Resource": [
        "arn:aws:wafv2:region:account-id:regional/webacl/*/*"
    ]
},
{
    "Sid": "AssociateWebACL2",
    "Effect": "Allow",
    "Action": [
        "appsync:SetWebACL"
    ],
    "Resource": [
        "arn:aws:appsync:*:account-id:apis/*"
    ]
}
```

Amazon Cognito ユーザープール

ユーザープールリソースタイプで Amazon Cognito AssociateWebACLアクションを呼び出し、ウェブ ACL で を呼び AWS WAF AssociateWebACL出すアクセス許可が必要です。

```
{
    "Sid": "AssociateWebACL1",
    "Effect": "Allow",
    "Action": [
        "wafv2:AssociateWebACL"
    ],
    "Resource": [
        "arn:aws:wafv2:region:account-id:regional/webacl/*/*"
    ]
},
{
    "Sid": "AssociateWebACL2",
    "Effect": "Allow",
    "Action": [
```

```
        "cognito-idp:AssociateWebACL"
    ],
    "Resource": [
        "arn:aws:cognito-idp:*:account-id:userpool/*"
    ]
}
```

AWS App Runner サービス

App Runner サービスリソースタイプで App Runner `AssociateWebACL`アクションを呼び出し、ウェブ ACL で を呼び AWS WAF `AssociateWebACL`出すアクセス許可が必要です。

```
{
    "Sid": "AssociateWebACL1",
    "Effect": "Allow",
    "Action": [
        "wafv2:AssociateWebACL"
    ],
    "Resource": [
        "arn:aws:wafv2:region:account-id:regional/webacl/*/*"
    ]
},
{
    "Sid": "AssociateWebACL2",
    "Effect": "Allow",
    "Action": [
        "apprunner:AssociateWebAcl"
    ],
    "Resource": [
        "arn:aws:apprunner:*:account-id:service/*/*"
    ]
}
```

AWS Verified Access インスタンス

Verified Access インスタンスリソースタイプで `ec2:AssociateVerifiedAccessInstanceWebAcl`アクションを呼び出し、ウェブ ACL で を呼び AWS WAF `AssociateWebACL`出すアクセス許可が必要です。

```
{
    "Sid": "AssociateWebACL1",
    "Effect": "Allow",
    "Action": [
```

```
            "wafv2:AssociateWebACL"
        ],
        "Resource": [
            "arn:aws:wafv2:region:account-id:regional/webacl/*/*"
        ]
    },
    {
        "Sid": "AssociateWebACL2",
        "Effect": "Allow",
        "Action": [
            "ec2:AssociateVerifiedAccessInstanceWebAcl"
        ],
        "Resource": [
            "arn:aws:ec2:*:account-id:verified-access-instance/*"
        ]
    }
```

`DisassociateWebACL` のアクセス権限

このセクションでは、 AWS WAF アクション `DisassociateWebACL` を使用してウェブ ACL とリソースの関連付けを解除するために必要なアクセス許可を一覧表示します。

Amazon CloudFront ディストリビューションの場合、このアクションの代わりに、空のウェブ ACL ID `UpdateDistribution`で CloudFront アクションを使用します。詳細については、「Amazon CloudFront API リファレンスUpdateDistribution」の「」を参照してください。

Amazon API Gateway REST API

REST API リソースタイプで API ゲートウェイ `SetWebACL` を呼び出すアクセス許可が必要です。を呼び出すアクセス許可は必要ありません AWS WAF `DisassociateWebACL`。

```
{
    "Sid": "DisassociateWebACL",
    "Effect": "Allow",
    "Action": [
        "apigateway:SetWebACL"
    ],
    "Resource": [
        "arn:aws:apigateway:*::/restapis/*/stages/*"
    ]
}
```

Application Load Balancer

Application Load Balancer リソースタイプで elasticloadbalancing:SetWebACL アクションを呼び出すアクセス許可が必要です。を呼び出すアクセス許可は必要ありません AWS WAF DisassociateWebACL。

```
{
    "Sid": "DisassociateWebACL",
    "Effect": "Allow",
    "Action": [
        "elasticloadbalancing:SetWebACL"
    ],
    "Resource": [
        "arn:aws:elasticloadbalancing:*:account-id:loadbalancer/app/*/*"
    ]
}
```

AWS AppSync GraphQL API

GraphQL API リソースタイプで を呼び AWS AppSync SetWebACL出すアクセス許可が必要です。を呼び出すアクセス許可は必要ありません AWS WAF DisassociateWebACL。

```
{
    "Sid": "DisassociateWebACL",
    "Effect": "Allow",
    "Action": [
        "appsync:SetWebACL"
    ],
    "Resource": [
        "arn:aws:appsync:*:account-id:apis/*"
    ]
}
```

Amazon Cognito ユーザープール

ユーザープールリソースタイプで Amazon Cognito DisassociateWebACLアクションを呼び出し、を呼び出すアクセス許可が必要です AWS WAF DisassociateWebACL。

```
{
    "Sid": "DisassociateWebACL1",
    "Effect": "Allow",
    "Action": "wafv2:DisassociateWebACL",
    "Resource": "*"
},
```

```
{
    "Sid": "DisassociateWebACL2",
    "Effect": "Allow",
    "Action": [
        "cognito-idp:DisassociateWebACL"
    ],
    "Resource": [
        "arn:aws:cognito-idp:*:account-id:userpool/*"
    ]
}
```

AWS App Runner サービス

App Runner サービスリソースタイプで App Runner `DisassociateWebACL`アクションを呼び出し、 を呼び出すアクセス許可が必要です AWS WAF `DisassociateWebACL`。

```
{
    "Sid": "DisassociateWebACL1",
    "Effect": "Allow",
    "Action": "wafv2:DisassociateWebACL",
    "Resource": "*"
},
{
    "Sid": "DisassociateWebACL2",
    "Effect": "Allow",
    "Action": [
        "apprunner:DisassociateWebAcl"
    ],
    "Resource": [
        "arn:aws:apprunner:*:account-id:service/*/*"
    ]
}
```

AWS Verified Access インスタンス

Verified Access インスタンスリソースタイプで `ec2:DisassociateVerifiedAccessInstanceWebAcl`アクションを呼び出し、 を呼び出すアクセス許可が必要です AWS WAF `DisassociateWebACL`。

```
{
    "Sid": "DisassociateWebACL1",
    "Effect": "Allow",
```

```
    "Action": "wafv2:DisassociateWebACL",
    "Resource": "*"
},
{
    "Sid": "DisassociateWebACL2",
    "Effect": "Allow",
    "Action": [
        "ec2:DisassociateVerifiedAccessInstanceWebAcl"
    ],
    "Resource": [
        "arn:aws:ec2:*:account-id:verified-access-instance/*"
    ]
}
```

`GetWebACLForResource` のアクセス権限

このセクションでは、 AWS WAF アクション `GetWebACLForResource` を使用して保護対象リソースのウェブ ACL を取得するために必要なアクセス許可の一覧を示します。

Amazon CloudFront ディストリビューションの場合、このアクションの代わりに、 CloudFront アクション を使用します`GetDistributionConfig`。詳細については、「Amazon CloudFront API リファレンスGetDistributionConfig」の「」を参照してください。

> **Note**
>
> `GetWebACLForResource` によって `GetWebACL` を呼び出すにはアクセス許可が必要です。このコンテキストでは、 は、 が`GetWebACLForResource`返すウェブ ACL にアクセスするために必要なアクセス許可がアカウントにあることを確認するために`GetWebACL`のみ AWS WAF 使用します。を呼び出すと`GetWebACLForResource`、アカウントが resource `wafv2:GetWebACL`に対して実行する権限がないことを示すエラーが表示されることがあります。このタイプのエラー AWS WAF は AWS CloudTrail イベント履歴に追加されません。

Amazon API Gateway REST API、Application Load BalancerGraph AWS AppSync GraphQL API

ウェブ ACL `GetWebACL`の および を呼び出す AWS WAF `GetWebACLForResource`アクセス許可が必要です。

```
{
    "Sid": "GetWebACLForResource",
    "Effect": "Allow",
```

```
    "Action": [
        "wafv2:GetWebACLForResource",
        "wafv2:GetWebACL"
    ],
    "Resource": [
        "arn:aws:wafv2:region:account-id:regional/webacl/*/*"
    ]
}
```

Amazon Cognito ユーザープール

ユーザープールリソースタイプで Amazon Cognito `GetWebACLForResource`アクションを呼び出し、 および を呼び AWS WAF `GetWebACLForResource`出すアクセス許可が必要ですGetWebACL。

```
{
    "Sid": "GetWebACLForResource1",
    "Effect": "Allow",
    "Action": [
        "wafv2:GetWebACLForResource",
        "wafv2:GetWebACL"
    ],
    "Resource": [
        "arn:aws:wafv2:region:account-id:regional/webacl/*/*"
    ]
},
{
    "Sid": "GetWebACLForResource2",
    "Effect": "Allow",
    "Action": [
        "cognito-idp:GetWebACLForResource"
    ],
    "Resource": [
        "arn:aws:cognito-idp:*:account-id:userpool/*"
    ]
}
```

AWS App Runner サービス

App Runner サービスリソースタイプで App Runner `DescribeWebAclForService`アクションを呼び出し、 と `GetWebACLForResource` を呼び出す AWS WAF アクセス許可が必要ですGetWebACL。

```
{
    "Sid": "GetWebACLForResource1",
    "Effect": "Allow",
    "Action": [
        "wafv2:GetWebACLForResource",
        "wafv2:GetWebACL"
    ],
    "Resource": [
        "arn:aws:wafv2:region:account-id:regional/webacl/*/*"
    ]
},
{
    "Sid": "GetWebACLForResource2",
    "Effect": "Allow",
    "Action": [
        "apprunner:DescribeWebAclForService"
    ],
    "Resource": [
        "arn:aws:apprunner:*:account-id:service/*/*"
    ]
}
```

AWS Verified Access インスタンス

Verified Access インスタンスリソースタイプで ec2:GetVerifiedAccessInstanceWebAclアクションを呼び出し、 および を呼び AWS WAF GetWebACLForResource出すアクセス許可が必要ですGetWebACL。

```
{
    "Sid": "GetWebACLForResource1",
    "Effect": "Allow",
    "Action": [
        "wafv2:GetWebACLForResource",
        "wafv2:GetWebACL"
    ],
    "Resource": [
        "arn:aws:wafv2:region:account-id:regional/webacl/*/*"
    ]
},
{
    "Sid": "GetWebACLForResource2",
    "Effect": "Allow",
    "Action": [
```

```
        "ec2:GetVerifiedAccessInstanceWebAcl"
    ],
    "Resource": [
        "arn:aws:ec2:*:account-id:verified-access-instance/*"
    ]
}
```

`ListResourcesForWebACL` のアクセス権限

このセクションには、 AWS WAF アクション `ListResourcesForWebACL` を使用してウェブ ACL の保護対象リソースのリストを取得するために必要なアクセス許可の一覧が記載されています。

Amazon CloudFront ディストリビューションの場合、このアクションの代わりに、 CloudFront アクション を使用しますListDistributionsByWebACLId。詳細については、「Amazon CloudFront API ListDistributionsByWebACLId」を参照してください。

Amazon API Gateway REST API、Application Load BalancerGraph AWS AppSync GraphQL API

ウェブ ACL の を呼び AWS WAF `ListResourcesForWebACL`出すアクセス許可が必要です。

```
{
    "Sid": "ListResourcesForWebACL",
    "Effect": "Allow",
    "Action": [
        "wafv2:ListResourcesForWebACL"
    ],
    "Resource": [
        "arn:aws:wafv2:region:account-id:regional/webacl/*/*"
    ]
}
```

Amazon Cognito ユーザープール

ユーザープールリソースタイプで Amazon Cognito `ListResourcesForWebACL` アクションを呼び出し、 AWS WAF `ListResourcesForWebACL` を呼び出すためのアクセス許可が必要です。

```
{
    "Sid": "ListResourcesForWebACL1",
    "Effect": "Allow",
    "Action": [
        "wafv2:ListResourcesForWebACL"
```

```
    ],
    "Resource": [
        "arn:aws:wafv2:region:account-id:regional/webacl/*/*"
    ]
},
{
    "Sid": "ListResourcesForWebACL2",
    "Effect": "Allow",
    "Action": [
        "cognito-idp:ListResourcesForWebACL"
    ],
    "Resource": [
        "arn:aws:cognito-idp:*:account-id:userpool/*"
    ]
}
```

AWS App Runner サービス

App Runner サービスリソースタイプで App Runner `ListAssociatedServicesForWebAcl`アクションを呼び出し、 を呼び出すアクセス許可が必要です AWS WAF `ListResourcesForWebACL`。

```
{
    "Sid": "ListResourcesForWebACL1",
    "Effect": "Allow",
    "Action": [
        "wafv2:ListResourcesForWebACL"
    ],
    "Resource": [
        "arn:aws:wafv2:region:account-id:regional/webacl/*/*"
    ]
},
{
    "Sid": "ListResourcesForWebACL2",
    "Effect": "Allow",
    "Action": [
        "apprunner:ListAssociatedServicesForWebAcl"
    ],
    "Resource": [
        "arn:aws:apprunner:*:account-id:service/*/*"
    ]
}
```

AWS Verified Access インスタンス

検証済みアクセス インスタンスのリソース タイプで ec2:DescribeVerifiedAccessInstanceWebAclAssociations アクションを呼び出すには、AWS WAF ListResourcesForWebACL を呼び出すためのアクセス許可が必要です。

```
{
    "Sid": "ListResourcesForWebACL1",
    "Effect": "Allow",
    "Action": [
        "wafv2:ListResourcesForWebACL"
    ],
    "Resource": [
        "arn:aws:wafv2:region:account-id:regional/webacl/*/*"
    ]
},
{
    "Sid": "ListResourcesForWebACL2",
    "Effect": "Allow",
    "Action": [
        "ec2:DescribeVerifiedAccessInstanceWebAclAssociations"
    ],
    "Resource": [
        "arn:aws:ec2:*:account-id:verified-access-instance/*"
    ]
}
```

のポリシーリソース AWS WAF

ポリシーリソースに対するサポート	あり

管理者は AWS JSON 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どのような条件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。ステートメントには、Resource または NotResource 要素を含める必要があります。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。これは、リソースレベルの許可と呼ばれる特定のリソースタイプをサポートするアクションに対して実行できます。

オペレーションのリスト化など、リソースレベルの権限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": "*"
```

AWS WAF リソースタイプとその ARNs」のAWS WAF V2 で定義されるリソース」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は、AWS WAF V2 で定義されるアクション」を参照してください。 AWS WAF リソースのサブセットへのアクセスを許可または拒否するには、ポリシーの resource要素にリソースの ARN を含めます。

リソースARNs の形式は次のとおりです。 AWS WAF wafv2

```
arn:partition:wafv2:region:account-id:scope/resource-type/resource-name/resource-id
```

ARN の仕様に関する一般情報については、「 Amazon Web Services 全般のリファレンス」の「Amazon リソースネーム (ARN)」を参照してください。

wafv2 リソースの ARN に固有の要件は以下の通りです。

- *region* : Amazon CloudFront ディストリビューションの保護に使用する AWS WAF リソースの場合は、これを に設定しますus-east-1。それ以外の場合は、保護されたリージョンリソースで使用している領域を設定します。
- *####* : Amazon CloudFront ディストリビューションglobalで使用するか、 regionalが AWS WAF サポートするリージョンリソースで使用するスコープを に設定します。リージョンリソースは、Amazon API Gateway REST API、Application Load Balancer AWS AppSync GraphQL API、Amazon Cognito ユーザープール、 AWS App Runner サービス、および AWS Verified Access インスタンスです。
- *#######*: 次の値のいずれかを指定します。webacl、rulegroup、ipset、regexpatternset、managedruleset。
- *resource-name*: AWS WAF リソースに付けた名前を指定、あるいは ARN の他の仕様を満たすすべてのリソースを示すワイルドカード (*) を指定します。リソース名とリソース ID のどちらかを指定するか、両方にワイルドカードを指定する必要があります。
- *resource-id*: AWS WAF リソースの ID を指定、あるいは ARN の他の仕様を満たすすべてのリソースを示すワイルドカード (*) を指定します。リソース名とリソース ID のどちらかを指定するか、両方にワイルドカードを指定する必要があります。

例えば、次の ARN は、リージョン us-west-1 におけるアカウント 111122223333 のリージョンレベルの範囲のすべてのウェブ ACL を指定します。

```
arn:aws:wafv2:us-west-1:111122223333:regional/webacl/*/*
```

次の ARN は、リージョン us-east-1 のアカウント 111122223333 に対して、グローバルスコープを持つ MyIPManagementRuleGroup というルールグループを指定します。

```
arn:aws:wafv2:us-east-1:111122223333:global/rulegroup/MyIPManagementRuleGroup/1111aaaa-bbbb-cccc-dddd-example-id
```

AWS WAF 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、「」を参照してくださいAWS WAFのアイデンティティベースのポリシーの例。

のポリシー条件キー AWS WAF

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 AWS JSON 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んなリソースにどんな条件でアクションを実行できるかということです。

Condition 要素 (または Condition ブロック) を使用すると、ステートメントが有効な条件を指定できます。Condition 要素はオプションです。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することで、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。

1 つのステートメントに複数の Condition 要素を指定する場合、または 1 つの Condition 要素に複数のキーを指定する場合、 AWS では AND 論理演算子を使用してそれらを評価します。1 つの条件キーに複数の値を指定すると、 は論理ORオペレーションを使用して条件 AWS を評価します。ステートメントの権限が付与される前にすべての条件が満たされる必要があります。

条件を指定する際にプレースホルダー変数も使用できます。例えば IAM ユーザーに、IAM ユーザー名がタグ付けされている場合のみリソースにアクセスできる権限を付与することができます。詳細については、IAM ユーザーガイドのIAM ポリシーの要素: 変数およびタグを参照してください。

AWS は、グローバル条件キーとサービス固有の条件キーをサポートします。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。

さらに、 では、IAM ポリシーのきめ細かなフィルタリングを提供するために使用できる以下の条件キー AWS WAF がサポートされています。

- wafv2:LogDestinationResource

この条件キーは、ログ記録の送信先の Amazon リソースネーム (ARN) 仕様を使用します。これは、REST API コール を使用するときにログ記録先として指定する ARN ですPutLoggingConfiguration。

ARN を明示的に指定し、ARN のフィルタリングを指定できます。次の例では、特定の場所とプレフィックスを持つ Amazon S3 バケット ARNs のフィルタリングを指定します。

```
"Condition": { "ArnLike": { "wafv2:LogDestinationResource": "arn:aws:s3:::aws-waf-logs-suffix/custom-prefix/*" } }
```

- wafv2:LogScope

  この条件キーは、文字列内のログ記録設定のソースを定義します。現在、これは常にデフォルトのに設定されています。これはCustomer、ログ記録の送信先がユーザーによって所有および管理されていることを示します。

AWS WAF 条件キーのリストを確認するには、「サービス認証リファレンス」の AWS WAF V2 の条件キー」を参照してください。条件キーを使用できるアクションとリソースについては、AWS WAF V2 で定義されるアクション」を参照してください。

AWS WAF 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、「」を参照してくださいAWS WAFのアイデンティティベースのポリシーの例。

ACLs AWS WAF

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許可を持つかをコントロールします。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。

での ABAC AWS WAF

ABAC (ポリシー内のタグ) のサポート	部分的

属性ベースのアクセス制御 (ABAC) は、属性に基づいてアクセス許可を定義する認可戦略です。ではAWS、これらの属性はタグ と呼ばれます。タグは、IAM エンティティ (ユーザーまたはロール) およ

び多くの AWS リソースにアタッチできます。エンティティとリソースのタグ付けは、ABAC の最初の手順です。その後、プリンシパルのタグがアクセスしようとしているリソースのタグと一致した場合にオペレーションを許可するように ABAC ポリシーをします。

ABAC は、急成長する環境やポリシー管理が煩雑になる状況で役立ちます。

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 とは?を参照してください。ABAC をセットアップするステップ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に基づくアクセスコントロール (ABAC) を使用するを参照してください。

での一時的な認証情報の使用 AWS WAF

一時的な認証情報のサポート	あり

一部の AWS のサービス は、一時的な認証情報を使用してサインインすると機能しません。一時的な認証情報 AWS のサービス を使用する などの詳細については、IAM ユーザーガイドのAWS のサービス 「IAM と連携する 」を参照してください。

ユーザー名とパスワード以外の AWS Management Console 方法で にサインインする場合、一時的な認証情報を使用します。例えば、会社の Single Sign-On (SSO) リンク AWS を使用して にアクセスすると、そのプロセスによって一時的な認証情報が自動的に作成されます。また、ユーザーとしてコンソールにサインインしてからロールを切り替える場合も、一時的な認証情報が自動的に作成されます。ロールの切り替えに関する詳細については、IAM ユーザーガイドのロールへの切り替え (コンソール)を参照してください。

一時的な認証情報は、 AWS CLI または AWS API を使用して手動で作成できます。その後、これらの一時的な認証情報を使用して . AWS recommends にアクセスできます AWS。これは、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することを推奨しています。詳細については、IAM の一時的セキュリティ認証情報を参照してください。

サービスの転送アクセスセッション AWS WAF

転送アクセスセッション (FAS) をサポート	あり

IAM ユーザーまたはロールを使用して でアクションを実行すると AWS、プリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストのリクエストと組み合わせて使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。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。

AWS WAFのサービスロール

サービスロールに対するサポート	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 のサービスにアクセス許可を委任するロールの作成」を参照してください。

> **Warning**
>
> サービスロールのアクセス許可を変更すると、 AWS WAF 機能が破損する可能性があります。が指示する場合以外 AWS WAF は、サービスロールを編集しないでください。

のサービスにリンクされたロール AWS WAF

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、サービスにリンクされた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。

AWS WAF 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、「」を参照してくださいのサービスにリンクされたロールの使用 AWS WAF。

AWS WAF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、 AWS WAF リソースを作成または変更する権限はありません。また、 AWS Management Console、 AWS Command Line Interface (AWS CLI)、または AWS API を使用してタスクを実行することもできません。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。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き受けることができます。

これらサンプルの 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 ポリシーの作成」を参照してください。

各リソースタイプの ARN の形式など AWS WAF、で定義されるアクションとリソースタイプの詳細については、『サービス認証リファレンス』の「AWS WAF V2 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

- ポリシーのベストプラクティス
- AWS WAF コンソールを使用する
- 自分の許可の表示をユーザーに許可する
- 、、には読み取り専用アクセス権を付与します AWS WAF。 CloudFront CloudWatch
- AWS WAF、 CloudFront、へのフルアクセス権を付与します。 CloudWatch
- 1 人だけにアクセス権を付与します。 AWS アカウント
- 単一のウェブ ACL にアクセス権を付与
- ウェブ ACL およびルールグループに対して、CLI アクセス権を付与

ポリシーのベストプラクティス

ID ベースのポリシーは、 AWS WAF アカウント内のリソースを誰かが作成、アクセス、削除できるかどうかを決定します。これらのアクションを実行すると、 AWS アカウントに料金が発生する可能性があります。アイデンティティベースの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 管理ポリシーから始めて、最小権限の権限に移行する — ユーザーとワークロードへの権限の付与を開始するには、AWS 多くの一般的なユースケースで権限を付与する管理ポリシーを使用してください。これらのポリシーは、で利用できます。 AWS アカウント AWS ユースケースに固有のカスタマー管理ポリシーを定義して、権限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「AWS ジョブ機能の管理ポリシー」を参照してください。
- 最小特権を適用する – IAM ポリシーで許可を設定するときは、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。例えば、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。サービスアクションがなどの特定の用途で使用された場合は AWS のサービス、条件を使用して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 AWS CloudFormation。詳細については、「IAM ユーザーガイド」の [IAM JSON policy elements: Condition] (IAM JSON ポリシー要素：条件) を参照してください。
- IAM Access Analyzer 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 Access Analyzer 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 ポリシーの検証」を参照してください。
- 多要素認証 (MFA) が必要 — IAM ユーザーまたは root 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、セキュリティを強化するために MFA をオンにしてください。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 保護 API アクセスの設定」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。

AWS WAF コンソールを使用する

AWS WAF コンソールにアクセスするには、最低限の権限が必要です。これらの権限により、 AWS WAF 内のリソースの詳細を一覧表示したり表示したりできる必要があります 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、最低限のコンソール権限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。

AWS WAF ユーザーとロールがコンソールを使用できるようにするには、 AWS WAF AWSWAFConsoleReadOnlyAccess AWS 少なくとも管理ポリシーをエンティティにアタッチしてください。このマネージドポリシーの情報については、「AWS マネージドポリシー：AWSWAFConsoleReadOnlyAccess」を参照してください。マネージドポリシーをユーザーにアタッチする方法の詳細については、IAM ユーザーガイドの「Adding permissions to a user」(ユーザーに許可の追加) を参照してください。

自分の許可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添付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を作成する方法を示します。このポリシーには、コンソールで、またはまたは API を使用してこのアクションをプログラムで実行するための権限が含まれています。 AWS CLI AWS

```
{
    "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": "*"
        }
    ]
}
```

、、には読み取り専用アクセス権を付与します AWS WAF。 CloudFront CloudWatch

次のポリシーは、 AWS WAF リソース、Amazon CloudFront ウェブディストリビューション、および Amazon メトリックスへの読み取り専用アクセスをユーザーに付与します。 CloudWatch AWS WAF 条件、ルール、ウェブ ACL の設定を閲覧する権限が必要なユーザーにとって、どのディストリビューションがウェブ ACL に関連付けられているかを確認したり、そこに含まれるメトリックスやリクエストのサンプルを監視したりするのに便利です。 CloudWatchこれらのユーザーは、 AWS WAF リソースを作成、更新、または削除することはできません。

```
{
   "Version":"2012-10-17",
   "Statement": [
      {
         "Action": [
            "wafv2:Get*",
            "wafv2:List*",
            "cloudfront:GetDistribution",
            "cloudfront:GetDistributionConfig",
            "cloudfront:ListDistributions",
            "cloudfront:ListDistributionsByWebACLId",
            "cloudwatch:ListMetrics",
            "cloudwatch:GetMetricStatistics",
            "ec2:DescribeRegions"
         ],
         "Effect": "Allow",
         "Resource": "*"
      }
   ]
}
```

AWS WAF、 CloudFront、へのフルアクセス権を付与します。 CloudWatch

次のポリシーでは、 AWS WAF ユーザーはウェブディストリビューションであらゆる操作を実行したり、 CloudFront ウェブディストリビューションであらゆる操作を実行したり、メトリクスやリクエストのサンプルを監視したりできます。 CloudWatch AWS WAF これは管理者であるユーザーにとって便利です。

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Action": [
            "wafv2:*",
            "cloudfront:CreateDistribution",
            "cloudfront:GetDistribution",
            "cloudfront:GetDistributionConfig",
            "cloudfront:UpdateDistribution",
            "cloudfront:ListDistributions",
            "cloudfront:ListDistributionsByWebACLId",
            "cloudfront:DeleteDistribution",
            "cloudwatch:ListMetrics",
            "cloudwatch:GetMetricStatistics",
            "ec2:DescribeRegions"
         ],
         "Effect": "Allow",
         "Resource": "*"
      }
   ]
}
```

管理者許可を持つユーザーに対しては多要素認証 (MFA) を設定することを強くお勧めします。詳細については、「IAM ユーザーガイド」の「AWSでのMulti-Factor Authentication (MFA) の使用」を参照してください。

1 人だけにアクセス権を付与します。 AWS アカウント

このポリシーは、アカウント 444455556666 に次の許可を付与します。

- AWS WAF すべてのオペレーションとリソースへのフルアクセス。
- CloudFront すべてのディストリビューションへの読み取りおよび更新権限。これにより、ウェブ ACL CloudFront とディストリビューションを関連付けることができます。

- CloudWatch すべてのメトリクスとメトリクス統計への読み取り権限。これにより、 CloudWatch コンソールでデータやリクエストのサンプルを表示できます。 AWS WAF

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "wafv2:*"
         ],
         "Resource": [
            "arn:aws:wafv2:us-east-1:444455556666:*"
         ]
      },
      {
         "Effect": "Allow",
         "Action": [
            "cloudfront:GetDistribution",
            "cloudfront:GetDistributionConfig",
            "cloudfront:ListDistributions",
            "cloudfront:ListDistributionsByWebACLId",
            "cloudfront:UpdateDistribution",
            "cloudwatch:ListMetrics",
            "cloudwatch:GetMetricStatistics",
            "ec2:DescribeRegions"
         ],
         "Resource": [
            "*"
         ]
      }
   ]
}
```

単一のウェブ ACL にアクセス権を付与

以下のポリシーでは、ユーザーはアカウント内の特定のウェブ ACL AWS WAF に対してコンソールから任意の操作を実行できます444455556666。

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
        {
            "Effect": "Allow",
            "Action": [
                "wafv2:*"
            ],
            "Resource": [
                "arn:aws:wafv2:us-east-1:444455556666:regional/webacl/
test123/112233d7c-86b2-458b-af83-51c51example",
            ]
        },
        {
            "Sid": "consoleAccess",
            "Effect": "Allow",
            "Action": [
                "wafv2:ListWebACLs",
                "ec2:DescribeRegions"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

ウェブ ACL およびルールグループに対して、CLI アクセス権を付与

次のポリシーでは、ユーザーはアカウント内の特定のウェブ ACL と特定のルールグループに対して CLI AWS WAF を使用して任意の操作を実行できます444455556666。

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "wafv2:*"
         ],
         "Resource": [
        "arn:aws:wafv2:us-east-1:444455556666:regional/webacl/
test123/112233d7c-86b2-458b-af83-51c51example",
        "arn:aws:wafv2:us-east-1:444455556666:regional/rulegroup/
test123rulegroup/555555555-6666-1234-abcd-00d11example"
         ]
```

```
      }
   ]
}
```

次のポリシーでは、ユーザーはアカウント内の特定のウェブ ACL AWS WAF に対してコンソールから任意の操作を実行できます444455556666。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "wafv2:*"
            ],
            "Resource": [
                "arn:aws:wafv2:us-east-1:444455556666:regional/webacl/
test123/112233d7c-86b2-458b-af83-51c51example",
            ]
        },
        {
            "Sid": "consoleAccess",
            "Effect": "Allow",
            "Action": [
                "wafv2:ListWebACLs",
                "ec2:DescribeRegions"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

AWS の マネージドポリシー AWS WAF

AWS 管理ポリシーは、 によって作成および管理されるスタンドアロンポリシーです AWS。 AWS 管理ポリシーは、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ように設計されているため、ユーザー、グループ、ロールにアクセス許可の割り当てを開始できます。

AWS 管理ポリシーは、すべての AWS お客様が使用できるため、特定のユースケースに対して最小特権のアクセス許可を付与しない場合があることに注意してください。ユースケース別にカスタマーマネージドポリシーを定義して、マネージドポリシーを絞り込むことをお勧めします。

AWS 管理ポリシーで定義されているアクセス許可は変更できません。が AWS 管理ポリシーで定義されたアクセス許可 AWS を更新すると、ポリシーがアタッチされているすべてのプリンシパル ID (ユーザー、グループ、ロール) が更新されます。 は、新しい AWS のサービス が起動されたとき、または既存のサービスで新しい API AWS オペレーションが使用可能になったときに、 AWS 管理ポリシーを更新する可能性が最も高くなります。

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 管理ポリシー： AWSWAFReadOnlyAccess

このポリシーは、Amazon 、Amazon API Gateway 、Application Load Balancer CloudFront、、Amazon Cognito AWS AppSync、 AWS Verified Access などの統合サービスの AWS WAF リソース AWS App Runnerとリソースへのアクセスをユーザーに許可する読み取り専用アクセス許可を付与します。このポリシーを IAM ID にアタッチできます。 AWS WAF また、 は、ユーザーに代わって がアクションを実行できるようにするサービスロールにもこのポリシー AWS WAF をアタッチします。

このポリシーの詳細については、IAM コンソールAWSWAFReadOnlyAccessの「」を参照してください。

AWS マネージドポリシー： AWSWAFFullAccess

このポリシーは、Amazon 、Amazon API Gateway 、Application Load Balancer CloudFront、、Amazon Cognito AWS AppSync、 AWS Verified Access などの統合サービスのリソース AWS App Runnerと AWS WAF リソースへのフルアクセスを許可します。 Application Load Balancer このポリシーを IAM ID にアタッチできます。 AWS WAF また、 は、ユーザーに代わってがアクションを実行できるようにするサービスロールにもこのポリシー AWS WAF をアタッチします。

このポリシーの詳細については、IAM コンソールAWSWAFFullAccessの「」を参照してください。

AWS マネージドポリシー： AWSWAFConsoleReadOnlyAccess

このポリシーは、 AWS WAF コンソールに読み取り専用アクセス許可を付与します。これには、Amazon 、Amazon API Gateway CloudFront、Application Load Balancer 、Amazon Cognito AWS AppSync、 AWS Verified Access などの統合サービスの AWS App Runnerおよび AWS WAF リソースが含まれます。 Amazon API Gateway Amazon Cognito このポリシーを IAM アイデンティ

ティにアタッチできます。 は、ユーザーに代わって がアクションを実行できるようにする aiam/home#/policies/arn:aws:iam::aws:policy/AWSWAFConsoleFullAccess$serviceLevelSummary service ロール AWS WAF にもこのポリシー AWS WAF をアタッチします。

このポリシーの詳細については、IAM コンソールAWSWAFConsoleReadOnlyAccessの「」を参照してください。

AWS マネージドポリシー： AWSWAFConsoleFullAccess

このポリシーは、 AWS WAF Amazon 、Amazon API Gateway 、Application Load Balancer CloudFront、、Amazon AWS AppSync Amazon Cognito 、 AWS Verified Access などの統合サービスの AWS WAF および リソースを含む AWS App Runnerコンソールへのフルアクセスを許可します。このポリシーを IAM ID にアタッチできます。 は、ユーザーに代わって がアクションを実行できるようにするサービスロール AWS WAF にもこのポリシー AWS WAF をアタッチします。

このポリシーの詳細については、IAM コンソールAWSWAFConsoleFullAccessの「」を参照してください。

AWS マネージドポリシー: WAFV2LoggingServiceRolePolicy

このポリシーにより、 AWS WAF は Amazon Data Firehose にログを書き込むことができます。このポリシーは、 でロギングを有効にした場合にのみ使用されます AWS WAF。このポリシーは、AWSServiceRoleForWAFV2Logging サービスにリンクされたロールにアタッチされます。サービスにリンクされたロールの詳細については、「のサービスにリンクされたロールの使用 AWS WAF」を参照してください。

このポリシーの詳細については、IAM コンソールのWAFV2LoggingServiceRolePolicy」を参照してください。

AWS WAFAWS 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た AWS WAF 以降の 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を受け取るには、 の AWS WAF ドキュメント履歴ページの RSS フィードにサブスクライブしてくださいドキュメント履歴。

ポリシー	変更点の説明	日付
WAFV2LoggingServiceRolePolicy	このポリシーがアタッチされているサービスにリンクされ	2024-06-03

ポリシー	変更点の説明	日付
このポリシーにより、AWS WAF は Amazon Data Firehose にログを書き込むことができます。これは、ログ記録を有効にした場合にのみ使用されます。 IAM コンソールの詳細: WAFV2LoggingServiceRolePolicy。	たロールのアクセス許可設定にステートメント IDs (Sid) を追加しました。	
AWSServiceRoleForWAFV2Logging このサービスにリンクされたロールは、 が Amazon Data Firehose AWS WAF にログを書き込むことを許可するアクセス許可ポリシーを提供します。 IAM コンソールの詳細: AWSServiceRoleForWAFV2Logging。	アクセス許可設定にステートメント IDs (Sid) を追加しました。	2024-06-03
AWS WAF 変更追跡への の追加	AWS WAF は、 マネージドポリシー WAFV2LoggingServiceRolePolicy とサービスにリンクされたロール の変更の追跡を開始しましたAWSServiceRoleForWAFV2Logging 。	2024-06-03

ポリシー	変更点の説明	日付
`AWSWAFFullAccess` このポリシーにより、 AWS WAF は、 AWS WAF および統合サービスでユーザーに代わって AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFFullAccess。	で保護できるリソースタイプに AWS Verified Access インスタンスを追加するアクセス許可を拡張しました AWS WAF。	2023-06-17
`AWSWAFReadOnlyAccess` このポリシーにより、 AWS WAF は、 AWS WAF および統合サービスでユーザーに代わって AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFReadOnlyAccess。	で保護できるリソースタイプに AWS Verified Access インスタンスを追加するアクセス許可を拡張しました AWS WAF。	2023-06-17
`AWSWAFConsoleFullAccess` このポリシーにより、 AWS WAF は AWS WAF および統合サービスでユーザーに代わって AWS コンソールリソースやその他の AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFConsoleFullAccess。	で保護できるリソースタイプに AWS Verified Access インスタンスを追加するアクセス許可を拡張しました AWS WAF。	2023-06-17

ポリシー	変更点の説明	日付
`AWSWAFConsoleReadOnlyAccess` このポリシーにより、 AWS WAF は AWS WAF および統合サービスでユーザーに代わって AWS コンソールリソースやその他の AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFConsoleReadOnlyAccess。	で保護できるリソースタイプに AWS Verified Access インスタンスを追加するアクセス許可を拡張しました AWS WAF。	2023-06-17
`AWSWAFFullAccess` このポリシーにより、 AWS WAF は、 AWS WAF および統合サービスでユーザーに代わって AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFFullAccess。	AWS App Runner サービスのアクセス設定を修正するためのアクセス許可が拡張されました。	2023-06-06
`AWSWAFReadOnlyAccess` このポリシーにより、 AWS WAF は、 AWS WAF および統合サービスでユーザーに代わって AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFReadOnlyAccess。	AWS App Runner サービスのアクセス設定を修正するためのアクセス許可が拡張されました。	2023-06-06

ポリシー	変更点の説明	日付
`AWSWAFConsoleFullAccess` このポリシーにより、 AWS WAF は AWS WAF および統合サービスでユーザーに代わって AWS コンソールリソースやその他の AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFConsoleFullAccess。	AWS App Runner サービスのアクセス設定を修正するためのアクセス許可が拡張されました。	2023-06-06
`AWSWAFConsoleReadOnlyAccess` このポリシーにより、 AWS WAF は AWS WAF および統合サービスでユーザーに代わって AWS コンソールリソースやその他の AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFConsoleReadOnlyAccess。	AWS App Runner サービスのアクセス設定を修正するためのアクセス許可が拡張されました。	2023-06-06

ポリシー	変更点の説明	日付
`AWSWAFFullAccess` このポリシーにより、 AWS WAF は、 AWS WAF および統合サービスでユーザーに代わって AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFFullAccess。	で保護できるリソースタイプに AWS App Runner サービスを追加するためのアクセス許可を拡張しました AWS WAF。	2023-03-30
`AWSWAFReadOnlyAccess` このポリシーにより、 AWS WAF は、 AWS WAF および統合サービスでユーザーに代わって AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFReadOnlyAccess。	で保護できるリソースタイプに AWS App Runner サービスを追加するためのアクセス許可を拡張しました AWS WAF。	2023-03-30
`AWSWAFConsoleFullAccess` このポリシーにより、 AWS WAF は AWS WAF および統合サービスでユーザーに代わって AWS コンソールリソースやその他の AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFConsoleFullAccess。	で保護できるリソースタイプに AWS App Runner サービスを追加するためのアクセス許可を拡張しました AWS WAF。	2023-03-30

ポリシー	変更点の説明	日付
`AWSWAFConsoleReadOnlyAccess` このポリシーにより AWS WAF 、 は AWS WAF および統合サービスでユーザーに代わって AWS コンソールリソースやその他の AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFConsoleReadOnlyAccess。	で保護できるリソースタイプに AWS App Runner サービスを追加するためのアクセス許可を拡張しました AWS WAF。	2023-03-30
`AWSWAFFullAccess` このポリシーにより、 AWS WAF は、 AWS WAF および統合サービスでユーザーに代わって AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFFullAccess。	で保護できるリソースタイプに Amazon Cognito ユーザープールを追加するアクセス許可を拡張しました AWS WAF。	2022-08-25
`AWSWAFReadOnlyAccess` このポリシーにより、 AWS WAF は、 AWS WAF および統合サービスでユーザーに代わって AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFReadOnlyAccess。	で保護できるリソースタイプに Amazon Cognito ユーザープールを追加するアクセス許可を拡張しました AWS WAF。	2022-08-25

ポリシー	変更点の説明	日付
`AWSWAFConsoleFullAccess` このポリシーにより AWS WAF 、 は AWS WAF および統合サービスでユーザーに代わって AWS コンソールリソースやその他の AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFConsoleFullAccess。	で保護できるリソースタイプに Amazon Cognito ユーザープールを追加するアクセス許可を拡張しました AWS WAF。	2022-08-25
`AWSWAFConsoleReadOnlyAccess` このポリシーにより AWS WAF 、 は AWS WAF および統合サービスでユーザーに代わって AWS コンソールリソースやその他の AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFConsoleReadOnlyAccess。	で保護できるリソースタイプに Amazon Cognito ユーザープールを追加するアクセス許可を拡張しました AWS WAF。	2022-08-25

ポリシー	変更点の説明	日付
AWSWAFFullAccess このポリシーにより、 AWS WAF は、 AWS WAF および統合サービスでユーザーに代わって AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFFullAccess。	Amazon Simple Storage Service (Amazon S3) と Amazon CloudWatch Logs のログ配信のアクセス許可設定を修正しました。この変更により、ログ記録設定中に発生していたアクセス拒否エラーが解決されます。ウェブ ACL トラフィックのログ記録の詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。	2022-01-11
AWSWAFConsoleFullAccess このポリシーにより AWS WAF 、 は AWS WAF および統合サービスでユーザーに代わって AWS コンソールリソースやその他の AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFConsoleFullAccess。	Amazon Simple Storage Service (Amazon S3) と Amazon CloudWatch Logs のログ配信のアクセス許可設定を修正しました。この変更により、ログ記録設定中に発生していたアクセスエラーが解決されます。ウェブ ACL トラフィックのログ記録の詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。	2022-01-11

ポリシー	変更点の説明	日付
AWSWAFFullAccess このポリシーにより、 AWS WAF は、 AWS WAF および統合サービスでユーザーに代わって AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFFullAccess。	拡張ログ記録オプション用の新しい許可を追加しました。 この変更により、Amazon Simple Storage Service (Amazon S3) と Amazon CloudWatch Logs の追加のログ記録先 AWS WAF にアクセスできます。ウェブ ACL トラフィックのログ記録の詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。	2021-11-15
AWSWAFConsoleFullAccess このポリシーにより AWS WAF 、 は AWS WAF および統合サービスでユーザーに代わって AWS コンソールリソースやその他の AWS リソースを管理できます。 IAM コンソールの詳細: AWSWAFConsoleFullAccess。	拡張ログ記録オプション用の新しい許可を追加しました。 この変更により、Amazon Simple Storage Service (Amazon S3) と Amazon CloudWatch Logs の追加のログ記録先 AWS WAF にアクセスできます。ウェブ ACL トラフィックのログ記録の詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。	2021-11-15
AWS WAF が変更の追跡を開始しました	AWS WAF が AWS マネージドポリシーの変更の追跡を開始しました。	2021-3-01

AWS WAF ID とアクセスのトラブルシューティング

次の情報は、 および IAM の使用時に発生する可能性がある一般的な問題の診断 AWS WAF と修正に役立ちます。

トピック

でアクションを実行する権限がない AWS WAF

「I am not authorized to perform an action in Amazon Bedrock」というエラーが表示された場合、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wafv2:*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:
 wafv2:GetWidget on resource: my-example-widget
```

この場合、wafv2:*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。サインイン認証情報を提供した担当者が管理者です。

iam を実行する権限がありません。PassRole

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 AWS WAF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、そのサービスに既存のロールを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権限が必要です。

以下の例のエラーは、marymajor という IAM ユーザーがコンソールを使用して AWS WAFで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権限が必要です。メアリーには、ロールをサービスに渡す許可がありません。

```
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。サインイン認証情報を提供した担当者が管理者です。

自分の 以外のユーザーに自分の AWS WAF 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、以下を参照してください:

- がこれらの機能 AWS WAF をサポートしているかどうかを確認するには、「」を参照してくださいが IAM と AWS WAF 連携する方法。
- 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でのロールとリソースベースのポリシーの使用の違いについては、IAM ユーザーガイドの「IAM でのクロスアカウントリソースアクセス」を参照してください。

のサービスにリンクされたロールの使用 AWS WAF

AWS WAF 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は、 に直接リンクされた一意のタイプの IAM ロールで

す AWS WAF。サービスにリンクされたロールは、 によって事前定義 AWS WAF されており、ユーザーに代わってサービスから他の AWS の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、必要なアクセス許可を手動で追加する必要がなくなるため、 の設定 AWS WAF が簡単になります。 は、サービスにリンクされたロールのアクセス許可 AWS WAF を定義し、特に定義されている場合を除き、 のみがそのロールを引き受け AWS WAF ることができます。定義された許可には、信頼ポリシーと許可ポリシーが含まれます。こ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。

サービスにリンクされたロールを削除するには、まずそのロールの関連リソースを削除します。これにより、 AWS WAF リソースにアクセスするためのアクセス許可を誤って削除することがないため、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他のサービスについては、IAM と連携するAWS のサービスを参照して、Service-Linked Role] (サービスにリンクされたロール)列で Yes] (はい) のあるサービスを探してください。そのサービスに関する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、リンクが設定されている Yes] (はい) を選択します。

のサービスにリンクされたロールのアクセス許可 AWS WAF

AWS WAF は、サービスにリンクされたロールAWSServiceRoleForWAFV2Loggingを使用して Amazon Data Firehose にログを書き込みます。このロールは、 でログインを有効にした場合にのみ使用されます AWS WAF。ログ作成の詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。

このサービスにリンクされたロールは、 AWS マネージドポリシー にアタッチされますWAFV2LoggingServiceRolePolicy。管理ポリシーの詳細については、「AWS マネージドポリシー: WAFV2LoggingServiceRolePolicy」を参照してください。

AWSServiceRoleForWAFV2Logging サービスにリンクされたロールは、ロール wafv2.amazonaws.com を引き受けるためにサービスを信頼します。

ロールのアクセス許可ポリシーにより AWS WAF 、 は指定されたリソースに対して次のアクションを実行できます。

- Amazon Data Firehose アクション: Firehose データストリームリソースPutRecordBatchの PutRecordおよび は、 で始まる名前ですaws-waf-logs-。例えば aws-waf-logs-us-east-2-analytics です。
- AWS Organizations アクション: Organizations 組織のリソースDescribeOrganization。

IAM コンソールでサービスにリンクされたロール全体を参照してください: AWSServiceRoleForWAFV2Logging。

サービスリンクロールの作成、編集、削除を IAM エンティティ (ユーザー、グループ、ロールなど) に許可するには、アクセス許可を設定する必要があります。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許可を参照してください。

AWS WAFのサービスリンクロールの作成

サービスリンクロールを手動で作成する必要はありません。で AWS WAF ログ記録を有効にするか AWS Management Console、 AWS WAF CLI または AWS WAF API でPutLoggingConfigurationリクエストを行うと、 によってサービスにリンクされたロールが自動的に AWS WAF 作成されます。

ログ記録を有効化するためには、iam:CreateServiceLinkedRole 許可が必要です。

このサービスリンクロールを削除した後で再度作成する必要が生じた場合は、同じ方法でアカウントにロールを再作成できます。 AWS WAF ログ記録を有効にすると、 によってサービスにリンクされたロールが再度 AWS WAF 作成されます。

AWS WAFのサービスにリンクされたロールの編集

AWS WAF では、AWSServiceRoleForWAFV2Loggingサービスにリンクされたロールを編集することはできません。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「サービスにリンクされたロールの編集」を参照してください。

AWS WAFのサービスリンクロールの削除

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。そうすることで、積極的にモニタリングまたは保守されていない未使用のエンティティを排除できます。ただし、手動で削除する前に、サービスリンクロールのリソースをクリーンアップする必要があります。

> **Note**
>
> リソースを削除しようとしたときに AWS WAF サービスがロールを使用している場合、削除が失敗する可能性があります。失敗した場合は、数分待ってから操作を再試行してください。

が使用する AWS WAF リソースを削除するには **AWSServiceRoleForWAFV2Logging**

1. AWS WAF コンソールで、すべてのウェブ ACL からログ記録を削除します。詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。
2. API または CLI を使用して、ログ記録が有効化されている各ウェブ ACL に `DeleteLoggingConfiguration` リクエストを送信します。詳細については、「AWS WAF API リファレンス」を参照してください。

IAM を使用して、サービスにリンクされたロールを手動で削除するには

`AWSServiceRoleForWAFV2Logging` サービスにリンクされたロールを削除するには、IAM コンソール、IAM CLI、または IAM API を使用します。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。

AWS WAF のサービスにリンクされたロールをサポートするリージョン

AWS WAF は、サービスが利用可能な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、「AWS WAF エンドポイントとクォータ」を参照してください。

ログインとモニタリング AWS WAF

監視は、 AWS およびソリューションの信頼性、可用性、パフォーマンスを維持する上で重要な部分です。 AWS WAF AWS ソリューションのすべての部分から監視データを収集して、マルチポイント障害が発生した場合により簡単にデバッグできるようにする必要があります。 AWS には、 AWS WAF リソースを監視し、発生する可能性のあるイベントに対応するためのツールがいくつか用意されています。

Amazon CloudWatch アラーム

CloudWatch アラームを使用すると、指定した期間にわたって 1 つのメトリクスを監視できます。メトリックスが特定のしきい値を超えると、Amazon SNS CloudWatch AWS Auto Scaling トピックまたはポリシーに通知を送信します。詳細については、「Amazon によるモニタリング CloudWatch」を参照してください。

AWS CloudTrail ログ

CloudTrail 内のユーザ、ロール、 AWS またはサービスが実行したアクションの記録を提供します AWS WAF。によって収集された情報を使用して CloudTrail、要求の送信元 IP アドレス AWS WAF、要求の実行者、実行日時、その他の詳細情報を判断できます。詳細については、「での AWS CloudTrail API コールのログ記録」を参照してください。

AWS WAF ウェブ ACL トラフィックロギング

AWS WAF ウェブ ACL が分析するトラフィックのロギングを行います。ログには、 AWS WAF AWS 保護対象リソースからリクエストを受信した時刻、リクエストに関する詳細情報、リクエストが一致したルールのアクション設定などの情報が含まれます。詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。

のコンプライアンス検証 AWS WAF

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、コンプライアンスプログラムAWS のサービス による対象範囲内のコンプライアンスプログラムを参照し、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、「 でのレポートのダウンロード AWS Artifact」の」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。 では、コンプライアンスに役立つ以下のリソース AWS を提供しています。

- セキュリティとコンプライアンスのクイックスタートガイド – これらのデプロイガイドでは、アーキテクチャ上の考慮事項について説明し、セキュリティとコンプライアンスに重点を置いたベースライン環境 AWS を にデプロイする手順について説明します。
- アマゾン ウェブ サービスにおける HIPAA セキュリティとコンプライアンスのアーキテクチャ — このホワイトペーパーでは、企業が AWS を使用して HIPAA 対象アプリケーションを作成する方法について説明します。

  > **Note**
  >
  > すべて AWS のサービス HIPAA の対象となるわけではありません。詳細については、「HIPAA 対応サービスのリファレンス」を参照してください。

- AWS コンプライアンスリソース – このワークブックとガイドのコレクションは、お客様の業界や地域に適用される場合があります。
- AWS カスタマーコンプライアンスガイド — コンプライアンスの観点から責任共有モデルを理解します。このガイドでは、ガイダンスを保護し AWS のサービス 、複数のフレームワーク (米国国立標準技術研究所 (NIST)、Payment Card Industry Security Standards Council (PCI)、国際標準化機構 (ISO) を含む) のセキュリティコントロールにマッピングするためのベストプラクティスをまとめています。
- 「 デベロッパーガイド」の「 ルールによるリソースの評価」 – この AWS Config サービスは、リソース設定が社内プラクティス、業界ガイドライン、および規制にどの程度準拠しているかを評価します。 AWS Config
- AWS Security Hub – これにより AWS のサービス 、 内のセキュリティ状態を包括的に把握できます AWS。Security Hub では、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対するコンプライアンスをチェックします。サポートされているサービスとコントロールのリストについては、「Security Hub のコントロールリファレンス」を参照してください。

- Amazon GuardDuty – これにより AWS アカウント、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環境を監視することで、、、ワークロード、コンテナ、データに対する潜在的な脅威 AWS のサービス を検出します。 GuardDuty は、特定のコンプライアンスフレームワークで義務付けられている侵入検知要件を満たすことで、PCI DSS などのさまざまなコンプライアンス要件への対応に役立ちます。
- AWS Audit Manager – これにより AWS のサービス 、 AWS 使用状況を継続的に監査し、リスクの管理方法と規制や業界標準への準拠を簡素化できます。

のレジリエンス AWS WAF

AWS AWS リージョン グローバルインフラストラクチャはアベイラビリティーゾーンを中心に構築されています。 AWS リージョン 物理的に分離された複数のアベイラビリティーゾーンを提供し、低レイテンシー、高スループット、冗長性の高いネットワークで接続します。アベイラビリティーゾーンでは、アベイラビリティーゾーン間で中断せずに、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、耐障害性、およびスケーラビリティが優れています。

AWS リージョン およびアベイラビリティーゾーンの詳細については、「グローバルインフラストラクチャ」を参照してください。AWS

AWS WAF内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として、 AWS WAF AWS グローバルなネットワークセキュリティによって保護されています。 AWS AWS セキュリティサービスとインフラストラクチャの保護方法については、「AWS Cloud Security」を参照してください。 AWS 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設計するには、「Security Pillar AWS Well-Architected Framework におけるインフラストラクチャ保護」を参照してください。

AWS 公開されている API AWS WAF 呼び出しを使用してネットワーク経由でアクセスします。クライアントは以下をサポートする必要があります:

- Transport Layer Security (TLS)。TLS 1.2、できれば TLS 1.3 が必要です。
- DHE (Ephemeral Diffie-Hellman) や ECDHE (Elliptic Curve Ephemeral Diffie-Hellman) などの Perfect Forward Secrecy (PFS) を使用した暗号スイート。これらのモードは、Java 7 以降など、ほとんどの最新システムでサポートされています。

また、リクエストには、アクセスキー ID と、IAM プリンシパルに関連付けられているシークレットアクセスキーを使用して署名する必要があります。または、AWS Security Token ServiceAWS STS を使用して、一時的なセキュリティ認証情報を生成し、リクエストに署名することもできます。

AWS WAF クォータ

> **Note**
>
> これはの最新バージョンです AWS WAF。 AWS WAF クラシックについては、を参照してくださいAWS WAF クラシック。

AWS WAF には以下のクォータ (以前は制限と呼ばれていました) が適用されます。これらのクォータは、利用可能なすべてのリージョンで同じです。 AWS WAF 各リージョンでは、これらのクォータが個別に適用されます。クォータは、リージョンにまたがって累積されません。

AWS WAF アカウントごとに保持できるエンティティの最大数にはデフォルトのクォータが設定されています。このクォータの引き上げをリクエストできます。

リソース	1 リージョン、1 アカウントあたりのデフォルトのクォータ
ウェブ ACL の最大数	100
ルールグループの最大数	100
IP セットの最大数	100
ウェブ ACL ごとの 1 秒あたりの最大リクエスト数。	25,000
ウェブ ACL またはルールグループあたりのカスタムリクエストヘッダーの最大数	100
ウェブ ACL またはルールグループあたりのカスタムレスポンスヘッダーの最大数	100
ウェブ ACL またはルールグループあたりのカスタムレスポンス本文の最大数	50
ウェブ ACL トークンドメインリストのトークンドメインの最大数	10

1 秒あたりの最大リクエスト数 (RPS) CloudFront は、 AWS WAF CloudFront 開発者ガイドによって設定され、説明されています。CloudFront

AWS WAF は、地域ごとのアカウントごとの次のエンティティ設定の割り当てを固定しています。これらのクォータは変更できません。

リソース	1 アカウント、1 リージョンあたりのクォータ
ウェブ ACL あたりの最大ウェブ ACL キャパシティーユニット (WCU)*	5,000
ルールグループあたりの最大 WCU	5,000
ルールグループあたりの参照ステートメントの最大数。ルールグループでは、参照ステートメントは IP セットまたは正規表現パターンセットを参照できます。	50
ウェブ ACL あたりの参照ステートメントの最大数。ウェブ ACL では、参照ステートメントはルールグループ、IP セット、または正規表現パターンセットを参照できます。	50
IP セットあたりの CIDR 表記の IP アドレスの最大数	10,000
ウェブ ACL あたりのレートベースのルールの最大数	10
ルールグループあたりのレートベースのルールの最大数	4
レートベースのルールに対して定義できる最小リクエスト率	100
レートベースのルールごとにレート制限できる一意の IP アドレスの最大数	10,000
文字列一致ステートメントの最大文字数	200
各正規表現パターンの最大文字数	200
正規表現セットあたりの一意の正規表現パターンの最大数	10
正規表現セットの最大数	10

リソース	1 アカウント、1 リージョンあたりのクォータ
Application Load Balancer と保護について検査できるウェブリクエストボディの最大サイズ AWS AppSync	8 KB
API Gateway、Amazon Cognito、App Runner CloudFront、および検証済みアクセス保護について検査できるウェブリクエスト本文の最大サイズ**	64 KB
ルールステートメントあたりのテキスト変換の最大数	10
単一のカスタムレスポンス定義のカスタムレスポンス本文コンテンツの最大サイズ	4 KB
1 つのカスタムレスポンス定義のカスタムヘッダーの最大数	10
1 つのカスタムリクエスト定義のカスタムヘッダーの最大数	10
単一のルールグループまたは単一のウェブ ACL 用のすべてのレスポンス本文コンテンツの最大合計サイズ	50 KB

*ウェブ ACL で 1,500 WCU を超える容量を使用すると、ウェブ ACL の基本料金を超えるコストが発生します。詳細については、「AWS WAF ウェブ ACL キャパシティーユニット (WCUs)」と「AWS WAF 料金表」を参照してください。

**デフォルトでは、API Gateway、Amazon Cognito CloudFront、App Runner、および検証済みアクセスリソースの本体検査制限は 16 KB に設定されていますが、ウェブ ACL 設定でこれらのリソースのいずれについても、記載されている最大数まで増やすことができます。詳細については、「本文検査のサイズ制限の管理」を参照してください。

AWS WAF リージョンごとのアカウントあたりの呼び出しには、以下の固定クォータがあります。これらのクォータは、コンソール、CLI、 AWS CloudFormation、REST API、SDK など、利用可能な手段を通じてサービスへのコールの合計に適用されます。これらのクォータは変更できません。

コールタイプ	1 アカウント、1 リージョンあたりのクォータ
`AssociateWebACL` へのコールの最大数	2 秒あたり 1 回のリクエスト
`DisassociateWebACL` へのコールの最大数	2 秒あたり 1 回のリクエスト
`GetWebACLForResource` へのコールの最大数	1 秒あたり 1 回のリクエスト
`ListResourcesForWebACL` へのコールの最大数	1 秒あたり 1 回のリクエスト
個々の `Get` または `List` アクションに対するコールの最大数 (他にクォータが定義されていない場合)	1 秒あたり 5 回のリクエスト
個々の `Create`、`Put`、または `Update` アクションへのコールの最大数 (他にクォータが定義されていない場合)	1 秒あたり 1 回のリクエスト

AWS WAF クラシックリソースをに移行する AWS WAF

このセクションでは、ルールとウェブ ACL AWS WAF をクラシックからに移行するためのガイダンスを提供します。 AWS WAF AWS WAF 2019 年 11 月にリリースされました。 AWS WAF Classic を使用してルールや Web ACL などのリソースを作成した場合は、 AWS WAF Classic を使用して作業するか、この最新バージョンに移行する必要があります。

移行作業を開始する前に、 AWS WAF 一読して内容をよく理解しておいてください。AWS WAF

トピック

- AWS WAFに移行する理由
- 移行の仕組み
- 移行に関する注意事項と制限事項
- ウェブ ACL AWS WAF をクラシックからに移行する AWS WAF

AWS WAFに移行する理由

AWS WAF の最新バージョンでは、慣れ親しんだ概念や用語のほとんどを維持しながら、以前のバージョンに比べて多くの改良が加えられています。

最新の AWS WAFの主な変更点を次に示します。移行を続ける前に、この一覧を確認して、ガイドの残りの部分をよく理解してください。 AWS WAF

- AWS マネージドルール対象 AWS WAF — AWS マネージドルールで利用できるようになったルールグループは、一般的なウェブの脅威に対する保護を提供します。これらのルールグループのほとんどは、に無料で含まれています AWS WAF。詳細については、AWS マネージドルールルールグループリストおよびブログ記事「AWS のマネージドルールの発表」を参照してください。 AWS WAF
- 新しい AWS WAF API — 新しい API では、単一の API AWS WAF セットを使用してすべてのリソースを設定できます。リージョン別アプリケーションとグローバルアプリケーションを区別するために、新しい API には scope 設定が含まれています。API の詳細については、「AWS WAFV2 アクション」および「AWS WAFV2 データ型」を参照してください。

  API、SDK、CLI、 AWS CloudFormationおよびでは、 AWS WAF Classic はその命名規則を維持しており、 AWS WAF この最新バージョンのにはv2、V2コンテキストに応じてまたはが追加されています。
- サービスクォータ (制限) の簡略化 — ウェブ ACL あたりのルール数が増え、 AWS WAF より長い正規表現パターンを表現できるようになりました。詳細については、「AWS WAF クォータ」を参照してください。
- ウェブ ACL の制限はコンピューティングニーズに基づくものになり、ウェブ ACL の制限はウェブ ACL キャパシティユニット (WCU) に基づくようになりました。 AWS WAF ルールの実行に必要な操作容量に基づいてルールの WCU を計算します。ウェブ ACL の WCU は、ウェブ ACL 内のすべてのルールとルールグループの WCU の合計です。

  WCU の一般情報については、「AWS WAF 仕組み」を参照してください。各ルールの WCU の使用量については、「ルールステートメントの基本」を参照してください。
- ドキュメントベースのルールの記述 – ルール、ルールグループ、ウェブ ACL を JSON 形式で記述および表現できるようになりました。個々の API コールを使用してさまざまな条件を作成し、その条件をルールに関連付ける必要がなくなりました。これにより、コードの記述方法と保守方法が大幅に簡素化されます。ウェブ ACL を表示しているときに、[Download web ACL as JSON] (ウェブ ACL を JSON としてダウンロード) を選択することで、コンソールからウェブ ACL の JSON

形式にアクセスできます。独自のルールを作成する場合は、[Rule JSON editor] (ルール JSON エディタ) を選択することで、その JSON 表現にアクセスできます。

- ルールのネストと完全な論理オペレーションのサポート - 論理ルールステートメントとネストを使用して、複雑な結合ルールを記述できます。`[A AND NOT(B OR C)]` などのステートメントを作成できます。詳細については、「論理ルールステートメント」を参照してください。
- レートベースのルールの改良 — の最新バージョンでは AWS WAF、ルールが評価する時間枠と、ルールがリクエストを集約する方法をカスタマイズできます。ウェブリクエストのさまざまな特性を組み合わせて集計をカスタマイズできます。さらに、最新のレートベースのルールは、トラフィックの変化により迅速に対応します。詳細については、「レートベースのルールステートメント」を参照してください。
- IP セットに対する可変 CIDR 範囲のサポート - IP セット指定における IP 範囲の柔軟性が向上しました。IPv4 では、 AWS WAF をサポートします。/1 /32IPv6 では/1、 AWS WAF をサポートします。/128IP セットの詳細については、「IP セット一致ルールステートメント」を参照してください。
- チェーン可能なテキスト変換 — ウェブリクエストコンテンツを検査する前に、 AWS WAF 複数のテキスト変換を実行できます。詳細については、「テキスト変換オプション」を参照してください。
- コンソールエクスペリエンスの向上 — AWS WAF 新しいコンソールには、視覚的なルールビルダーと、より直感的なコンソールデザインが採用されています。
- Firewall Manager AWS WAF ポリシーの拡張オプション — Firewall Manager AWS WAF のウェブ ACL管理では、 AWS WAF AWS WAF 最初に処理するルールグループのセットと最後に処理するルールグループのセットを作成できるようになりました。 AWS WAF ポリシーを適用すると、ローカルアカウント所有者はこれら 2 AWS WAF つのセット間で処理する独自のルールグループを追加できます。Firewall Manager の AWS WAF ポリシーの詳細については、「AWS WAF ポリシー」を参照してください。
- AWS CloudFormation AWS CloudFormation すべてのルールステートメントタイプをサポート — AWS WAF AWS WAF コンソールと API がサポートするすべてのルールステートメントタイプをサポートします。さらに、JSON 形式で記述したルールを YAML 形式に簡単に変換できます。

移行の仕組み

AWS WAF 自動移行は従来のウェブ ACL 設定のほとんどを引き継ぐため、手動で処理する必要のあるものがいくつか残ります。

ウェブ ACL を移行するためのステップの概要を次に示します。

1. 自動移行では、 AWS WAF Classic 内の変更や削除を行わずに、既存のウェブ ACL に関連するすべてが読み取られます。これにより、と互換性のあるウェブ ACL と関連リソースの表現が作成されます AWS WAF。新しいウェブ ACL の AWS CloudFormation テンプレートを生成し、Amazon S3 バケットに保存します。
2. テンプレートをにデプロイして AWS CloudFormation、でウェブ ACL と関連リソースを再作成します。 AWS WAF
3. ウェブ ACL を確認して手動で移行を完了し、新しいウェブ ACL が最新の AWS WAFの機能を最大限に活用するようにします
4. 保護されたリソースを新しいウェブ ACL に手動で切り替えます。

移行に関する注意事項と制限事項

移行では、 AWS WAF Classic のすべての設定がそのまま引き継がれるわけではありません。マネージドルールと同様、2 つのバージョン間で正確に対応しないことがあります。ウェブ ACL の保護された AWS リソースとの関連付けなど、その他の設定は、新しいバージョンでは最初は無効になっているため、準備ができたら追加できます。

次のリストでは、移行の注意事項と、対処が必要となる可能性があるステップについて説明します。この概要を使用して、移行を計画してください。後述する移行の詳細なステップでは、推奨される緩和ステップについて順を追って説明します。

- 単一アカウント — AWS WAF 任意のアカウントのクラシックリソースは、 AWS WAF 同じアカウントのリソースにのみ移行できます。
- マネージドルール — AWS Marketplace 移行によって販売者からマネージドルールが引き継がれることはありません。 AWS Marketplace 一部の販売者には、同等のマネージドルールがあり AWS WAF 、再度購読することができます。その前に、 AWS AWS WAF最新バージョンのに付属しているマネージドルールを確認してください。 AWS WAF これらのほとんどはユーザーには無料です。マネージドルールの詳細については、「マネージドルールグループ」を参照してください。
- ウェブ ACL の関連付け – 移行では、ウェブ ACL と保護されたリソース間の関連付けは引き継がれません。これは仕様です。本番環境のワークロードに影響を与えないようにするためのものです。すべて正しく移行されたことを検証したら、新しいウェブ ACL をリソースに関連付けます。
- ログ記録 – 移行されたウェブ ACL のログ記録は、デフォルトで無効になっています。これは仕様です。 AWS WAF クラシックからに切り替える準備ができたら、ロギングを有効にします AWS WAF。
- AWS Firewall Manager ルールグループ — 移行では、Firewall Manager によって管理されるルールグループは処理されません。Firewall Manager によって管理されているウェブ ACL を移行する

ことはできますが、移行でルールグループは引き継がれません。このようなウェブ ACL には、移行ツールを使用するのではなく、Firewall Manager で新しい AWS WAF のポリシーを再作成します。

> **Note**
>
> Firewall Manager AWS WAF がクラシックで管理していたルールグループは、Firewall Manager ールールグループでした。 AWS WAFの新しいバージョンでは、ルールグループはルールグループです AWS WAF 。機能的には、これらは同じです。

- AWS WAF セキュリティオートメーション — AWS WAF セキュリティオートメーションを移行しようとしないでください。移行では、オートメーションにより使用される可能性がある Lambda 関数が変換されません。 AWS WAF 最新のセキュリティ自動化ソリューションと互換性のある新しいセキュリティ自動化ソリューションが利用可能になったら AWS WAF、そのソリューションを再デプロイしてください。

ウェブ ACL AWS WAF をクラシックからに移行する AWS WAF

ウェブ ACL を移行してそれに切り替えるには、自動移行を実行してから、一連の手動ステップを実行します。

トピック

- ウェブ ACL の移行: 自動移行
- ウェブ ACL の移行: 手動フォローアップ
- ウェブ ACL の移行: その他の考慮事項
- ウェブ ACL の移行: 切り替え

ウェブ ACL の移行: 自動移行

ウェブ ACL AWS WAF 設定をクラシックからに自動的に移行するには AWS WAF

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。
2. [AWS WAF クラシックに切り替え] を選択し、ウェブ ACL の設定を確認します。前のセクション「移行に関する注意事項と制限事項」で説明した注意事項と制限事項を考慮して、設定をメモしておきます。

3. 上部の情報ダイアログで、「Migrate web ACLs」(ウェブ ACL を移行する) で始まる文を見つけて、移行ウィザードへのリンクを選択します。移行ウィザードが起動します。

   情報ダイアログが表示されない場合は、 AWS WAF Classic コンソールを起動してから閉じている可能性があります。ナビゲーションバーで [Switch to new AWS WAF] を選択し、[Switch to AWS WAF Classic] を選択すると、情報ダイアログが再び表示されるはずです。

4. 移行するウェブ ACL を選択します。

5. [Migration configuration] (移行設定) では、テンプレートに使用する Amazon S3 バケットを指定します。 AWS CloudFormation 生成されたテンプレートを保存するには、移行 API 用に適切に設定された Amazon S3 バケットが必要です。

   - バケットが暗号化されている場合、暗号化は Amazon S3 (SSE-S3) キーを使用する必要があります。移行では AWS Key Management Service (SSE-KMS) キーによる暗号化はサポートされていません。
   - バケット名の先頭は `aws-waf-migration-` にする必要があります。例えば、`aws-waf-migration-my-web-acl` です。
   - バケットは、テンプレートをデプロイするリージョンに存在する必要があります。例えば、`us-west-2` のウェブ ACL の場合、`us-west-2` で Amazon S3 バケットを使用し、テンプレートスタックを `us-west-2` にデプロイする必要があります。

6. S3 バケットポリシーの場合、[Auto apply the bucket policy required for migration] (移行に必要なバケットポリシーを自動的に適用する) を選択することをお勧めします。または、自分でバケットを管理する場合は、次のバケットポリシーを手動で適用する必要があります。

   - グローバル Amazon CloudFront アプリケーション (`waf`) の場合:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "apiv2migration.waf.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::<BUCKET_NAME>/AWSWAF/<CUSTOMER_ACCOUNT_ID>/
*"
        }
    ]
```

```
}
```

- リージョンレベルの Amazon API Gateway または Application Load Balancer アプリケーションの場合 (`waf-regional`):

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "apiv2migration.waf-regional.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::<BUCKET_NAME>/AWSWAF/<CUSTOMER_ACCOUNT_ID>/
*"
        }
    ]
}
```

7. [Choose how to handle rules that cannot be migrated] (移行できないルールの処理方法を選択してください) で、移行できないルールを除外するか、移行を停止するかを選択します。移行できないルールの詳細については、「移行に関する注意事項と制限事項」を参照してください。
8. [次へ] を選択します。
9. [AWS CloudFormation テンプレートの作成] で設定を確認し、[AWS CloudFormation テンプレートの作成を開始] を選択して移行プロセスを開始します。ウェブ ACL の複雑さによっては、この処理に数分かかる場合があります。
10. [AWS CloudFormation スタックを作成して実行して移行を完了する] では、 AWS CloudFormation コンソールに移動してテンプレートからスタックを作成し、新しいウェブ ACL とそのリソースを作成できます。これを行うには、[AWS CloudFormation スタックの作成] を選択します。

自動移行プロセスが完了したら、手動でのフォローアップステップに進む準備が整います。「ウェブ ACL の移行: 手動フォローアップ」を参照してください。

ウェブ ACL の移行: 手動フォローアップ

自動移行が完了したら、新しく作成したウェブ ACL を確認し、移行によって引き継がれないコンポーネントを入力します。次の手順では、移行によって処理されないウェブ ACL 管理の側面について説明します。リストについては、「移行に関する注意事項と制限事項」を参照してください。

基本的な移行を完了するには - 手動ステップ

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF でコンソールを開きます。
2. コンソールには自動的に最新バージョンのが使用されるはずです AWS WAF。これを確認するには、ナビゲーションペインに [AWS WAF クラシックに切り替え] オプションが表示されていることを確認します。[Switch to new] が表示されたら AWS WAF、それを選択して最新バージョンに切り替えます。
3. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。
4. [Web ACLs] (ウェブ ACL) ページで、作成したリージョンのリストで新しいウェブ ACL を見つけます。ウェブ ACL の名前を選択して、ウェブ ACL の設定を表示します。
5. 新しいウェブ ACL のすべての設定を、 AWS WAF 以前のクラシックウェブ ACL と照合して確認します。デフォルトでは、ログ記録と保護されたリソースの関連付けは無効になっています。切り替えの準備ができたら有効にします。
6. AWS WAF クラシックウェブ ACL に条件付きのレートベースのルールがある場合、その条件は移行時に引き継がれませんでした。新しいウェブ ACL のルールに条件を追加できます。

   a. ウェブ ACL 設定ページで、[Rules] (ルール) タブを選択します。

   b. リストでレートベースのルールを見つけて選択し、[Edit] (編集) を選択します。

   c. [Criteria to count request towards rate limit] (リクエストをレート制限にカウントする条件) で、[Only consider requests that match the criteria in a rule statement] (ルールステートメントの条件に一致するリクエストのみを考慮する) を選択し、追加の条件を指定します。論理ステートメントなど、ネストできる任意のルールステートメントを使用して条件を追加できます。選択肢の詳細については、「レートベースのルールステートメント」を参照してください。
7. AWS WAF クラシックウェブ ACL にマネージドルールグループが含まれている場合、そのルールグループの追加は移行時に引き継がれませんでした。マネージドルールグループを新しいウェブ ACL に追加できます。新しいバージョンの、 AWS で利用できるマネージドルールのリストなど、マネージドルールグループに関する情報を確認してください。 AWS WAFマネージドルールグループマネージドルールグループを追加するには、次の手順を実行します。

a. ウェブ ACL 設定ページで、ウェブ ACL の [Rules] (ルール) タブを選択します。

b. [Add rules] (ルールの追加) を選択し、[Add managed rule groups] (マネージドルールグループの追加) を選択します。

c. 選択したベンダーのリストを展開し、追加するルールグループを選択します。 AWS Marketplace 出品者の場合は、ルールグループへの登録が必要な場合があります。ウェブ ACL でのマネージドルールグループの使用の詳細については、「マネージドルールグループ」および「ウェブ ACL ルールおよびルールグループの評価」を参照してください。

基本的な移行プロセスを完了したら、ニーズを見直して追加のオプションを検討することにより、新しい設定の効率が可能な限り高いことと、利用可能な最新のセキュリティオプションを使用していることを確認することをお勧めします。「ウェブ ACL の移行: その他の考慮事項」を参照してください。

ウェブ ACL の移行: その他の考慮事項

新しいウェブ ACL を確認し、新しいウェブ ACL AWS WAF で使用できるオプションを検討して、構成が可能な限り効率的であること、および利用可能な最新のセキュリティオプションを使用していることを確認してください。

AWS その他のマネージドルール

アプリケーションのセキュリティを強化するために、ウェブ ACL AWS に追加のマネージドルールを実装することを検討してください。 AWS WAF これらは追加費用なしで含まれています。 AWS マネージドルールには以下のタイプのルールグループがあります。

- ベースラインルールグループは、既知の不正な入力がアプリケーションに入らないようにしたり、管理ページへのアクセスを防ぐなど、さまざまな一般的な脅威に対して全般的な保護を提供します。
- ユースケース固有のルールグループは、多種多様なユースケースに対して段階的な保護を提供します。
- IP 評価レピュテーションリストは、クライアントの送信元 IP に基づく脅威インテリジェンスを提供します。

詳細については、「AWS のマネージドルール AWS WAF」を参照してください。

ルールの最適化とクリーンアップ

古いルールを再検討し、それらを書き換えたり、古いルールを削除して最適化することを検討してください。たとえば、過去に「OWASP Webアプリケーションの脆弱性トップ10」、「OWASPトップ10のWebアプリケーションの脆弱性への備え」、 AWS WAF および「当社の新しいホワイトペーパー」 AWS CloudFormation のテクニカルペーパーのテンプレートをデプロイしたことがある場合は、それをマネージドルールに置き換えることを検討してください。 AWS このドキュメントに記載されている概念はまだ適用可能で、独自のルールを作成する際に役立つかもしれませんが、テンプレートによって作成されたルールは、主にマネージドルールに取って代わられています。 AWS

Amazon CloudWatch メトリクスとアラーム

Amazon CloudWatch メトリックスを再確認し、必要に応じてアラームを設定します。CloudWatch 移行してもアラームは引き継がれず、メトリックス名が希望どおりにならない可能性があります。

アプリケーションチームとの確認

アプリケーションチームと協力して、セキュリティ体制を確認してください。アプリケーションによって頻繁に解析されるフィールドを調べ、それに応じて入力をサニタイズするルールを追加します。エッジケースをチェックし、アプリケーションのビジネスロジックがケースを処理できない場合にこれらのケースをキャッチするルールを追加します。

切り替えの計画

アプリケーションチームと切り替えのタイミングを計画します。古いウェブ ACL の関連付けから新しいものへの切り替えは、リソースが保存されているすべての領域に反映されるまで、少し時間がかかる場合があります。伝播時間は、数秒から数分までかかります。この際、一部のリクエストは古いウェブ ACL で処理される一方、他のものは新しいウェブ ACL で処理されます。リソースは切り替え作業を通して保護されますが、その過程中にリクエスト処理に一貫性がないことに気付く場合があります。

切り替えの準備ができたら、「ウェブ ACL の移行: 切り替え」の手順に従います。

ウェブ ACL の移行: 切り替え

新しいウェブ ACL 設定を検証したら、 AWS WAF Classic ウェブ ACL の代わりに使用を開始できます。

AWS WAF 新しいウェブ ACL の使用を開始するには

1. のガイダンスに従って、 AWS WAF ウェブ ACL ウェブ ACL とリソースの関連付けまたは関連付け解除 AWS を保護対象のリソースに関連付けます。これにより、古いウェブ ACL からリソースの関連付けが自動的に解除されます。

切り替えは、伝播するために数秒から数分までかかります。この際、一部のリクエストは古いウェブ ACL で処理される場合がある一方、他のものは新しいウェブ ACL で処理される場合があります。リソースは切り替え作業を通して保護されますが、終了するまでリクエスト処理に一貫性がないことに気付く場合があります。

2. 「AWS WAF ウェブ ACL トラフィックのログ記録」のガイダンスに従って、新しいウェブ ACL のログ記録を設定します。
3. (オプション) AWS WAF クラシックウェブ ACL がリソースに関連付けられなくなった場合は、AWS WAF クラシックから完全に削除することを検討してください。詳細については、「ウェブ ACL の削除」を参照してください。

AWS WAF クラシック

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS WAF クラシックは、Amazon API Gateway、 CloudFront またはApplication Load Balancer に転送される HTTP および HTTPS リクエストを監視できるウェブアプリケーションファイアウォールです。 AWS WAF Classic ではコンテンツへのアクセスを制御することもできます。リクエストの発信元の IP アドレスやクエリ文字列の値など、指定した条件に基づいて、API Gateway、または Application Load Balancer は、 CloudFront リクエストされたコンテンツまたは HTTP 403 ステータスコード (禁止) のいずれかでリクエストに応答します。 CloudFront リクエストがブロックされた場合にカスタムエラーページを返すように設定することもできます。

トピック

- AWS WAF Classic のセットアップ
- AWS WAF クラシックの仕組み
- AWS WAF クラシック価格設定
- AWS WAF クラシック入門
- ウェブアクセスコントロールリスト (ウェブ ACL) の作成と設定
- AWS WAF で使用するためのクラシックルールグループの使用 AWS Firewall Manager
- AWS Firewall ManagerAWS WAF クラシックルールを有効にするにはじめに
- チュートリアル: 階層ルールによる AWS Firewall Managerポリシーの作成
- ウェブ ACL トラフィック情報のログ記録
- レートベースのルールごとにブロックされている IP アドレスの一覧表示
- AWS WAF クラシックと Amazon CloudFront の機能との連携
- AWS WAF クラシックのセキュリティ
- AWS WAF クラシック· クォータ

AWS WAF Classic のセットアップ

> **Note**
>
> これは AWS WAF Classic ドキュメントです。このバージョンは、2019 年 11 月 AWS WAF より前にルールやウェブ ACLs などのリソースを作成し AWS WAF 、まだ最新バージョンに移行していない場合にのみ使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては、 AWS WAF「」を参照してくださいAWS WAF。

このトピックでは、 AWS WAF Classic を使用する準備をするためのユーザーアカウントの作成などの準備手順について説明します。これらに対しては請求されません。使用した AWS サービスに対してのみ課金されます。

> **Note**
>
> を初めて使用する場合は AWS WAF、 AWS WAF Classic のセットアップ手順を実行しないでください。代わりに、 の最新バージョンの手順に従ってください AWS WAFサービスを使用するためのアカウントのセットアップ。

これらのステップを完了したら、「」を参照して AWS WAF Classic の使用をAWS WAF クラシック入門続行します。

> **Note**
>
> AWS Shield Standard は AWS WAF Classic に含まれており、追加のセットアップは必要ありません。詳細については、「AWS Shield アンドシールドアドバンスドの仕組み」を参照してください。

AWS WAF Classic または AWS Shield Advanced を初めて使用する前に、このセクションのステップを完了してください。

トピック

- にサインアップする AWS アカウント
- 管理アクセスを持つユーザーを作成する

- ツールをダウンロード

にサインアップする AWS アカウント

がない場合は AWS アカウント、次のステップを実行して作成します。

にサインアップするには AWS アカウント

1. https://portal.aws.amazon.com/billing/signup を開きます。
2. オンラインの手順に従います。

   サインアップ手順の一環として、通話呼び出しを受け取り、電話キーパッドで検証コードを入力するように求められます。

   にサインアップすると AWS アカウント、 AWS アカウントのルートユーザーが作成されます。ルートユーザーには、アカウントのすべての AWS のサービス とリソースへのアクセス権があります。セキュリティのベストプラクティスとして、ユーザーに管理アクセスを割り当て、ルートユーザーのみを使用してルートユーザーアクセスが必要なタスクを実行してください。

AWS サインアッププロセスが完了すると、 から確認メールが送信されます。https://aws.amazon.com/ の [マイアカウント] を選んで、いつでもアカウントの現在のアクティビティを表示し、アカウントを管理できます。

管理アクセスを持つユーザーを作成する

にサインアップしたら AWS アカウント、 を保護し AWS アカウントのルートユーザー、 を有効にして AWS IAM Identity Center、日常的なタスクにルートユーザーを使用しないように管理ユーザーを作成します。

のセキュリティ保護 AWS アカウントのルートユーザー

1. ルートユーザーを選択し、 AWS アカウント E メールアドレスを入力して、アカウント所有者AWS Management Consoleとして にサインインします。次のページでパスワードを入力します。

   ルートユーザーを使用してサインインする方法については、AWS サインイン ユーザーガイドの「ルートユーザーとしてサインインする」を参照してください。
2. ルートユーザーの多要素認証 (MFA) を有効にします。

手順については、「IAM ユーザーガイド」の AWS アカウント 「ルートユーザーの仮想 MFA デバイスを有効にする (コンソール)」を参照してください。

管理アクセスを持つユーザーを作成する

1. IAM アイデンティティセンターを有効にします。

   手順については、「AWS IAM Identity Center ユーザーガイド」の「AWS IAM Identity Centerの有効化」を参照してください。

2. IAM アイデンティティセンターで、ユーザーに管理アクセスを付与します。

   を ID ソース IAM アイデンティティセンターディレクトリ として使用する方法のチュートリアルについては、「 ユーザーガイド」の「デフォルト でユーザーアクセスを設定する IAM アイデンティティセンターディレクトリAWS IAM Identity Center 」を参照してください。

管理アクセス権を持つユーザーとしてサインインする

- IAM アイデンティティセンターのユーザーとしてサインインするには、IAM アイデンティティセンターのユーザーの作成時に E メールアドレスに送信されたサインイン URL を使用します。

  IAM Identity Center ユーザーを使用してサインインする方法については、「 AWS サインイン ユーザーガイド」の AWS 「 アクセスポータルへのサインイン」を参照してください。

追加のユーザーにアクセス権を割り当てる

1. IAM アイデンティティセンターで、最小特権のアクセス許可を適用するというベストプラクティスに従ったアクセス許可セットを作成します。

   手順については、「AWS IAM Identity Center ユーザーガイド」の「権限設定を作成する」を参照してください。

2. グループにユーザーを割り当て、そのグループにシングルサインオンアクセス権を割り当てます。

   手順については、「AWS IAM Identity Center ユーザーガイド」の「グループの参加」を参照してください。

ツールをダウンロード

には AWS WAF Classic のコンソール AWS Management Console が含まれていますが、プログラムで AWS WAF Classic にアクセスする場合は、以下を参照してください。

- raw HTTP リクエストの組み立てなど、低レベルの詳細を処理せずに AWS WAF Classic API を呼び出す場合は、 AWS SDK を使用できます。 AWS SDKs AWS WAF Classic およびその他の AWS サービスの機能をカプセル化する関数とデータ型を提供します。 AWS SDK をダウンロードするには、該当するページを参照してください。このページには、前提条件とインストール手順も含まれています。
  - Java
  - JavaScript
  - .NET
  - Node.js
  - PHP
  - Python
  - Ruby

  AWS SDKs「Amazon Web Services のツール」を参照してください。
- が SDK を提供しないプログラミング言語を使用している場合、 AWS WAF API リファレンスは AWS WAF Classic AWS がサポートするオペレーションを文書化します。
- AWS Command Line Interface （ AWS CLI) は AWS WAF Classic をサポートしています。 AWS CLI を使用すると、コマンドラインから複数の AWS サービスを制御したり、スクリプトを使用して自動化したりできます。詳細については、「AWS Command Line Interface」を参照してください。
- AWS Tools for Windows PowerShell は AWS WAF Classic をサポートします。詳細については、「AWS Tools for PowerShell Cmdlet Reference」(Cmdlet リファレンス) を参照してください。

AWS WAF クラシックの仕組み

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールや Web ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合

にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS WAF Classic を使用して、API Gateway、Amazon、 CloudFront またはApplication Load Balancer がウェブリクエストに応答する方法を制御します。まず、条件、ルール、ウェブアクセスコントロールリスト (ウェブ ACL) を作成します。条件を定義し、ルールに追加したら、ルールをウェブ ACL に結合します。

> **Note**
>
> AWS WAF クラシックを使用して、Amazon Elastic Container Service (Amazon ECS) コンテナでホストされているアプリケーションを保護することもできます。Amazon ECS は、クラスターで Docker コンテナを簡単に実行、停止、管理できる非常にスケーラブルで高速なコンテナ管理サービスです。このオプションを使用するには、 AWS WAF クラシック対応のApplication Load Balancer を使用して、サービス内のタスク全体で HTTP/HTTPS (レイヤー 7) トラフィックをルーティングして保護するように Amazon ECS を設定します。詳細については、「Amazon Elastic Container Service デベロッパーガイド」の「Service load balancing」(サービスのロードバランシング) のトピックを参照してください。

条件

条件では、ウェブリクエストで AWS WAF Classic が監視する基本的な特徴を定義します。

- 悪意のある可能性が高いスクリプト。攻撃者は、ウェブアプリケーションの脆弱性を悪用できるスクリプトを埋め込みます。これはクロスサイトスクリプティングと呼ばれます。
- リクエストの発生元の IP アドレスまたはアドレス範囲。
- リクエスト送信元の国または地理的場所。
- クエリ文字列など、リクエストの指定された部分の長さ。
- 悪意のある可能性が高い SQL コード。攻撃者は、ウェブリクエストに悪意のある SQL コードを埋め込むことで、データベースからデータを抽出しようとします。これは SQL インジェクションと呼ばれます。
- リクエストに表示される文字列。例えば、`User-Agent` ヘッダーに表示される値、またはクエリ文字列に表示されるテキスト文字列です。正規表現を使用してこれらの文字列を指定することもできます。

条件によっては、複数の値を指定できる場合があります。例えば、IP 条件では最大 10,000 個の IP アドレスまたは IP アドレス範囲を指定できます。

ルール

条件を組み合わせてルールを作成し、許可、ブロック、カウントしたいリクエストを正確にターゲットにします。 AWS WAF Classic には次の 2 種類のルールがあります。

通常ルール

通常ルールでは、特定のリクエストを対象とするための条件のみが使用されます。例えば、攻撃者からの最近のリクエストに基づいて、次の条件を含むルールを作成できます。

- リクエストが 192.0.2.44 から発生した。
- リクエストの User-Agent ヘッダーに BadBot 値が含まれる。
- それらのクエリ文字列には、SQL などのコードが含まれる。

ルールに複数の条件をすべて含めると、この例のように AWS WAF Classic によってすべての条件に一致するリクエストが検索されます。つまり、これらの条件を AND で連結したことになります。

少なくとも 1 つの条件を通常ルールに追加します。条件のない通常のルールはどのリクエストにも一致しないため、ルールのアクション (許可、カウント、またはブロック) はトリガーされません。

レートベースのルール

レートベースのルールは、レート制限が追加された通常のルールに似ています。レートベースのルールは、ルールの条件を満たす IP アドレスから到着したリクエストをカウントします。IP アドレスからのリクエストが 5 分間でレート制限を超えた場合、ルールはアクションをトリガーできます。アクションがトリガーされるまで、1～2 分かかることがあります。

レートベースのルールの条件はオプションです。レートベースのルールに条件を追加しない場合、レート制限はすべての IP アドレスに適用されます。条件をレート制限と組み合わせると、レート制限は条件に一致する IP アドレスに適用されます。

例えば、攻撃者からの最近のリクエストに基づいて、次の条件を含むレートベースのルールを作成できます。

- リクエストが 192.0.2.44 から発生した。
- リクエストの User-Agent ヘッダーに BadBot 値が含まれる。

このレートベースのルールでは、レート制限を定義することもできます。この例では、1,000 のレート制限を作成します。前述の条件の両方を満たし、5 分間に 1,000 リクエストを超えるリクエストは、ウェブ ACL で定義されたルールのアクション (ブロックまたはカウント) をトリガーします。

両方の条件を満たさないリクエストは、レート制限に対してカウントされないため、このルールの影響を受けません。

2 つ目の例では、ウェブサイトの特定のページにリクエストを制限します。これを行うには、次の文字列一致条件をレートベースのルールに追加します。

- [Part of the request to filter on] (フィルタリングするリクエストの一部) は、`URI` です。
- [Match Type] (一致タイプ) は `Starts with` です。
- [Value to match] (一致する値) は、`login` です。

さらに、1,000 の `RateLimit` を指定します。

このレートベースのルールをウェブ ACL に追加することで、残りのサイトに影響を与えることなく、ログインページへのリクエストを制限することができます。

ウェブ ACL

条件を組み合わせてルールを作成した後、ルールを組み合わせてウェブ ACL を作成します。ここで、各ルールのアクション (許可、ブロック、カウント) とデフォルトアクションを定義します。

各ルールのアクション

ウェブリクエストがルールのすべての条件に一致すると、 AWS WAF Classic はリクエストをブロックするか、リクエストを API Gateway API、 CloudFront ディストリビューション、またはApplication Load Balancer に転送することを許可できます。 AWS WAF Classic に実行させたいアクションをルールごとに指定します。

AWS WAF Classic は、ルールをリストした順序で、リクエストをウェブ ACL 内のルールと比較します。 AWS WAF その後、Classic はリクエストが最初に一致したルールに関連するアクションを実行します。たとえば、ウェブリクエストが、リクエストを許可するルールとリクエストをブロックするルールに一致する場合、 AWS WAF Classic は、最初にリストされているルールに応じて、そのリクエストを許可または拒否します。

新しいルールを使用する前にテストしたい場合は、ルール内のすべての条件を満たすリクエストをカウントするように AWS WAF Classic を設定することもできます。リクエストを許可またはブロックするルールと同様に、リクエストをカウントするルールも、ウェブ ACL でリストされている順番によって影響を受けます。例えば、ウェブリクエストが、リクエストを許可

する 1 つのルールと一致し、リクエストをカウントする別のルールと一致する場合、リクエストを許可するルールが最初にリストされていると、リクエストはカウントされません。

デフォルトアクション

デフォルトアクションは、ウェブ ACL のどのルールにも当てはまらないリクエストを AWS WAF Classic が許可するかブロックするかを決定します。例えば、ウェブ ACL を作成し、前に定義したルールのみを追加するとします。

- リクエストが 192.0.2.44 から発生した。
- リクエストの `User-Agent` ヘッダーに `BadBot` 値が含まれる。
- それらのリクエストのクエリ文字列に悪意のある可能性がある SQL コードが含まれる。

リクエストがルールの 3 つの条件をすべて満たさず、デフォルトアクションがの場合`ALLOW`、 AWS WAF Classic はリクエストを API Gateway CloudFront または Application Load Balancer に転送し、サービスはリクエストされたオブジェクトで応答します。

ウェブ ACL に 2 つ以上のルールを追加した場合、 AWS WAF Classic はリクエストがどのルールの条件も満たさない場合にのみデフォルトアクションを実行します。例えば、以下の 1 つの条件を含む 2 つ目のルールを追加するとします。

- `User-Agent` ヘッダーに `BIGBadBot` 値が含まれるリクエスト。

AWS WAF Classic は、リクエストが最初のルールの 3 つの条件をすべて満たさず、2 番目のルールの 1 つの条件を満たさない場合にのみデフォルトアクションを実行します。

場合によっては、内部エラーが発生して、Amazon API Gateway、Amazon、 CloudFront または Application Load Balancer AWS WAF へのリクエストを許可するかブロックするかについての応答が遅れることがあります。 CloudFrontこのような場合は、通常、リクエストを許可するか、コンテンツを配信します。API Gateway および Application Load Balancer は、通常、リクエストを拒否し、コンテンツを提供しません。

AWS WAF クラシック価格設定

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールや Web ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。

の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS WAF Classic では、作成したウェブ ACL とルール、および AWS WAF Classic が検査する HTTP リクエストの数に対してのみ料金が発生します。詳細については、「AWS WAF Classic の料金」を参照してください。

AWS WAF クラシック入門

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールや Web ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

このチュートリアルでは、 AWS WAF Classic を使用して次のタスクを実行する方法を示します。

- AWS WAF Classic をセットアップします。
- AWS WAF Classic コンソールを使用して Web アクセスコントロールリスト (ウェブ ACL) を作成し、ウェブリクエストをフィルタリングするために使用する条件を指定します。例えば、リクエストの発生元の IP アドレスと、攻撃者によってのみ使用されるリクエスト内の値を指定できます。
- 条件をルールに追加します。ルールでは、ブロックまたは許可するウェブリクエストを対象にすることができます。 AWS WAF 指定した条件に基づいてClassic がリクエストをブロックまたは許可する前に、ウェブリクエストがルール内のすべての条件に一致する必要があります。
- ルールをウェブ ACL に追加します。ウェブ ACL では、各ルールに追加する条件に基づいて、ウェブリクエストをブロックするか許可するかを指定します。
- ブロックまたは許可のいずれかのデフォルトアクションを指定します。これは、ウェブリクエストがどのルールにも一致しない場合に AWS WAF Classic が実行するアクションです。
- AWS WAF Classic にウェブリクエストを検査させたい Amazon CloudFront ディストリビューションを選択します。このチュートリアルではの手順のみを説明しますが CloudFront、Application Load Balancer と Amazon API Gateway API のプロセスは基本的に同じです。 AWS WAF Classic for CloudFront AWS リージョンはすべてのユーザーが利用できます。 AWS WAF API Gateway ま

たはApplication Load Balancer で使用する Classic は、AWS サービスエンドポイントに記載されているリージョンでご利用いただけます。

> **Note**
>
> AWS 通常、このチュートリアルで作成したリソースについて、1 日あたり 0.25 USD 未満で請求されます。チュートリアルを終了したら、不要な料金が発生しないようにリソースを削除することをお勧めします。

トピック

ステップ 1: クラシックをセットアップする AWS WAF

AWS WAF Classic のセットアップ の一般的なセットアップ手順をまだ実行していない場合、今すぐ実行してください。

ステップ 2: ウェブ ACL を作成する

AWS WAF Classic コンソールでは、リクエストの発信元の IP アドレスやリクエストの値など、指定した条件に基づいてウェブリクエストをブロックまたは許可するように AWS WAF Classic を設定する手順を案内します。このステップでは、ウェブ ACL を作成します。

ウェブ ACL を作成するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF でコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。

2. クラシックを初めて使用する場合は、[AWS WAF AWS WAF クラシックに移動] を選択し、[ウェブ ACL の設定] を選択します。

   AWS WAF Classic を使用したことがある場合は、ナビゲーションペインで [ウェブ ACL] を選択し、[ウェブ ACL の作成] を選択します。

3. [Name web ACL] (ウェブ ACL に名前を付ける) ページで、[Web ACL name] (ウェブ ACL の名前) に名前を入力します。

   > **Note**
   >
   > ウェブ ACL の作成後は、名前を変更することはできません。

4. [CloudWatch metric name (メトリクス名)] に、名前を入力します。名前には、英数字のみ (A ~ Z、a ~ z、0 ~ 9) を使用できます。空白を含めることはできません。

   > **Note**
   >
   > ウェブ ACL の作成後は、名前を変更することはできません。

5. [Region] (リージョン) で、リージョンを選択します。このウェブ ACL CloudFront をディストリビューションに関連付ける場合は、[Global (CloudFront)] を選択します。

6. [AWS resource to associate (関連付けるリソース)] で、ウェブ ACL に関連付けるリソースを選択し、[Next (次へ)] を選択します。

ステップ 3: IP 一致条件を作成する

IP 一致条件では、リクエストの発生元の IP アドレスまたは IP アドレス範囲を指定します。このステップでは、IP 一致条件を作成します。後のステップで、指定した IP アドレスからのリクエストを許可するかブロックするかを指定します。

> **Note**
>
> IP 一致条件の詳細については、「IP 一致条件の使用」を参照してください。

IP 一致条件を作成するには

1. [Create conditions] (条件を作成) ページの [IP match conditions] (IP 一致条件) で、[Create condition] (条件を作成) を選択します。
2. [Create IP match condition] (IP 一致条件の作成) ダイアログボックスの [Name] (名前) に名前を入力します。名前に使用できるのは英数字 (A～Z、a～z、0～9) または特殊文字 _-!"#`+*},./ です。
3. [Address] (アドレス) で、192.0.2.0/24 と入力します。この IP アドレス範囲は、CIDR 表記で指定しており、192.0.2.0 から 192.0.2.255 までの IP アドレスが含まれます (192.0.2.0/24 の IP アドレス範囲は例のために予約されているため、ウェブリクエストはこれらの IP アドレスから発生しません)。

   AWS WAF Classic は IPv4 アドレス範囲 (/8) と /16 から /32 までの任意の範囲をサポートします。 AWS WAF クラシックは IPv6 アドレス範囲 (/24、/32、/48、/56、/64、/128) をサポートしています。(192.0.2.44 など 1 つの IP アドレスを指定するには、192.0.2.44/32 と入力します)。他の範囲はサポートされていません。

   CIDR 表記の詳細については、Wikipedia の「Classless Inter-Domain Routing」(クラスレスドメイン間ルーティング) 記事を参照してください。
4. [Create] (作成) を選択します。

ステップ 4: Geo 一致条件を作成する

Geo 一致条件では、リクエスト送信元の 1 つ以上の国を指定します。このステップでは、Geo 一致条件を作成します。後のステップで、指定した国から送信されたリクエストを許可するかブロックするかを指定します。

> **Note**
>
> Geo 一致条件の詳細については、「Geo (地理的) 一致条件の使用」を参照してください。

Geo 一致条件を作成するには

1. [Create conditions] (条件を作成) ページの [Geo match conditions] (Geo 一致条件) で、[Create condition] (条件を作成) を選択します。
2. [Create geo match condition] (geo 一致条件の作成) ダイアログボックスの [Name] (名前) に名前を入力します。名前に使用できるのは英数字 (A ~ Z、a ~ z、0 ~ 9) または特殊文字 _-!"#`+*},./ です。
3. [Location type] (場所のタイプ) と国を選択します。現時点では、[Location type] (場所のタイプ) は [Country] (国) にのみ設定できます。
4. [Add location] (場所を追加) を選択します。
5. [Create] (作成) を選択します。

ステップ 5: 文字列一致条件を作成する

文字列一致条件は、ヘッダーやクエリ文字列内の指定された値など、 AWS WAF Classic がリクエスト内で検索する文字列を識別します。通常、文字列は印刷可能な ASCII 文字で構成されますが、16 進数 0x00 ～ 0xFF (10 進数 0 ～ 255) の任意の文字を指定できます。このステップでは、文字列一致条件を作成します。後のステップで、指定した文字列を含むリクエストを許可するかブロックするかを指定します。

> **Note**
>
> 文字列一致条件の詳細については、「文字列一致条件の使用」を参照してください。

文字列一致条件を作成するには

1. [Create conditions] (条件を作成) ページの [String and regex match conditions] (文字列および正規表現の一致条件) で、[Create condition] (条件を作成) を選択します。
2. [Create string match condition] (文字列一致条件の作成) ダイアログボックスで、次の値を入力します。

   [Name] (名前)

   名前を入力します。名前に使用できるのは英数字 (A ~ Z、a ~ z、0 ~ 9) または特殊文字 _-!"#`+*},./ です。

[Type] (タイプ)

[String match] (文字列の一致) を選択します。

[Part of the request to filter on] (フィルタリングするリクエストの一部)

指定した文字列を AWS WAF Classic に検査させたいウェブリクエストの部分を選択します。

この例では、[Header] (ヘッダー) を選択します。

> **Note**
>
> フィルターするリクエストの Part の値に Body を選択した場合、 AWS WAF Classic は最初の 8192 バイト (8 KB) だけを検査します。これは、最初の 8192 CloudFront バイトだけを検査対象として転送するためです。本文が 8,192 バイトより長いリクエストを許可またはブロックするには、サイズ制約条件を作成します。(AWS WAF Classic はリクエストヘッダーから本文の長さを取得します)。詳細については、「サイズ制約条件の使用」を参照してください。

[Header] (ヘッダー) ([Part of the request to filter on] (フィルタリングするリクエストの一部) が [Header] (ヘッダー) の場合は必須)

フィルターするリクエストの一部として [ヘッダー] を選択したため、 AWS WAF Classic で検査するヘッダーを指定する必要があります。[User-Agent] を入力します。(この値は大文字と小文字が区別されません。)

[Match type] (一致タイプ)

指定した文字列が [User-Agent] ヘッダーに表示される場所 (文字列の先頭、末尾、または任意の場所など) を選択します。

この例では、「完全一致」を選択します。これは、指定した値と同じヘッダー値がないか AWS WAF Classic がウェブリクエストを検査することを示しています。

[Transformation] (変換)

AWS WAF Classic をバイパスするために、攻撃者は空白を追加したり、リクエストの一部または全部を URL エンコードしたりするなど、ウェブリクエストに通常とは異なる形式を使用します。[Transformation] (変換) を有効にすることで、ウェブリクエストはより標準的な

形式に変換されます。そのために、空白が削除されたり、リクエストが URL デコードされたり、攻撃者がよく使用する特殊な形式の多くが排除されたりします。

1 種類のテキスト変換しか指定できません。

この例では、[None] (なし) を選択します。

[Value is base64 encoded] (値が base64 エンコードされている)

[Value to match] (一致する値) に入力した値が既に base64 でエンコードされている場合は、このチェックボックスをオンにします。

この例では、このチェックボックスをオンにしないでください。

[Value to match] (照合する値)

[フィルタリングするリクエストの一部] で指定したウェブリクエストの一部で AWS WAF Classic に検索させたい値を指定します。

この例では、と入力しますBadBot。 AWS WAF Classic `User-Agent` BadBotはウェブリクエストのヘッダーの値を調べます。

[Value to match] (一致する値) は最大 50 文字です。base64 でエンコードされた値を指定する場合、エンコード前の長さで最大 50 文字指定できます。

3. `User-Agent`BadBotを含むヘッダーとを含むクエリ文字列など、複数の値を求めるウェブリクエストを AWS WAF Classic で検査する場合は`BadParameter`、次の 2 つの選択肢があります。

   - 両方の値が含まれているときにのみ (AND)、ウェブリクエストを許可またはブロックする場合は、値ごとに 1 つの文字列一致条件を作成します。
   - 一方か両方の値が含まれているときに (OR)、ウェブリクエストを許可またはブロックする場合は、両方の値を同じ文字列一致条件に追加します。

   この例では、[Create] (作成) を選択します。

ステップ 5A: (オプション) 正規表現条件を作成する

正規表現条件は文字列一致条件の一種で、ヘッダーやクエリ文字列の指定値など、 AWS WAF Classic がリクエスト内で検索する文字列を識別するという点で似ています。主な違いは、 AWS WAF Classic で検索する文字列パターンを正規表現 (regex) を使用して指定することです。このス

テップでは、正規表現一致条件を作成します。後のステップで、指定した文字列を含むリクエストを許可するかブロックするかを指定します。

> **Note**
>
> 正規表現一致条件の詳細については、「正規表現一致条件の使用」を参照してください。

正規表現一致条件を作成するには

1. [Create conditions] (条件を作成) ページの [String match and regex conditions] (文字列の一致および正規表現の条件)で、[Create condition] (条件を作成) を選択します。
2. [Create string match condition] (文字列一致条件の作成) ダイアログボックスで、次の値を入力します。

   [Name] (名前)

   名前を入力します。名前に使用できるのは英数字 (A ~ Z、a ~ z、0 ~ 9) または特殊文字 _-!"#`+*},./ です。

   [Type] (タイプ)

   [Regex match] (正規表現の一致) を選択します。

   [Part of the request to filter on] (フィルタリングするリクエストの一部)

   指定した文字列を AWS WAF Classic に検査させたいウェブリクエストの部分を選択します。

   この例では、[Body] (本文) を選択します。

   > **Note**
   >
   > フィルターするリクエストの Part の値に Body を選択した場合、 AWS WAF Classic は最初の 8192 バイト (8 KB) だけを検査します。これは、最初の 8192 CloudFront バイトだけを検査対象として転送するためです。本文が 8,192 バイトより長いリクエストを許可またはブロックするには、サイズ制約条件を作成します。(AWS WAF Classic はリクエストヘッダーから本文の長さを取得します)。詳細については、「サイズ制約条件の使用」を参照してください。

[Transformation] (変換)

AWS WAF Classic をバイパスするために、攻撃者はウェブリクエストに通常とは異なる形式を使用します。たとえば、空白を追加したり、リクエストの一部または全部を URL でエンコードしたりします。[Transformation] (変換) を有効にすることで、ウェブリクエストはより標準的な形式に変換されます。そのために、空白が削除されたり、リクエストが URL デコードされたり、攻撃者がよく使用する特殊な形式の多くが排除されたりします。

1 種類のテキスト変換しか指定できません。

この例では、[None] (なし) を選択します。

Regex patterns to match to request (リクエストに一致する正規表現パターン)

[Create regex pattern set] (正規表現パターンセットを作成) を選択します。

New pattern set name (新しいパターンセット名)

名前を入力し、Classic に検索させたい正規表現パターンを指定します。 AWS WAF

次に、正規表現 I [a@] mAb [a@] DRequest を入力します。 AWS WAF Classic `User-Agent` はウェブリクエストのヘッダーを調べて値を確認します。

- iAMA BadRequest
- IamAB@dRequest
- 私 @mA BadRequest
- I@mAB@dRequest

3. [Create pattern set and add filter] (パターンセットを作成してフィルターを追加) を選択します。
4. [Create] (作成) を選択します。

ステップ 6: SQL インジェクション一致条件を作成する

SQL インジェクション一致条件は、ヘッダーやクエリ文字列などのウェブリクエストのうち、悪意のある SQL AWS WAF コードがないかClassic に検査させたい部分を特定します。攻撃者は SQL クエリを使用してデータベースからデータを抽出します。このステップでは、SQL インジェクション一致条件を作成します。後のステップで、悪意のある可能性がある SQL コードを含むリクエストを許可するかブロックするかを指定します。

> **Note**
>
> 文字列一致条件の詳細については、「SQL インジェクション一致条件の使用」を参照してください。

ステップ 5: SQL インジェクション一致条件を作成するには

1. [Create conditions] (条件を作成) ページの [SQL injection match conditions] (SQL インジェクション一致条件) で、[Create condition] (条件を作成) を選択します。
2. [Create SQL injection match condition] (SQL インジェクション一致条件の作成) ダイアログボックスで、次の値を入力します。

   [Name] (名前)

   名前を入力します。

   [Part of the request to filter on] (フィルタリングするリクエストの一部)

   AWS WAF Classic に悪意のある SQL コードがないか検査させたいウェブリクエストの部分を選択します。

   この例では、[Query string] (クエリ文字列) を選択します。

   > **Note**
   >
   > フィルターするリクエストの Part の値に Body を選択した場合、 AWS WAF Classic は最初の 8192 バイト (8 KB) だけを検査します。これは、最初の 8192 CloudFront バイトだけを検査対象として転送するためです。本文が 8,192 バイトより長いリクエストを許可またはブロックするには、サイズ制約条件を作成します。(AWS WAF Classic はリクエストヘッダーから本文の長さを取得します)。詳細については、「サイズ制約条件の使用」を参照してください。

   [Transformation] (変換)

   この例では、[URL decode] (URL デコード) を選択します。

   攻撃者は Classic AWS WAF をバイパスしようとして、URL エンコーディングなどの通常とは異なる形式を使用します。[URL decode] (URL デコード) オプションを選択すると、 AWS

WAF Classic によってリクエストが検査される前に、ウェブリクエストでその形式の部分が削除されます。

1 種類のテキスト変換しか指定できません。

3. [Create] (作成) を選択します。
4. [Next] (次へ) を選択します。

ステップ 7: (オプション) 追加の条件を作成する

AWS WAF Classic には他にも次のような条件があります。

- サイズ制約条件 — ヘッダーやクエリ文字列など、 AWS WAF Classic に長さを確認させたいウェブリクエストの部分を指定します。詳細については、「サイズ制約条件の使用」を参照してください。
- クロスサイトスクリプティング一致条件 — ヘッダーやクエリ文字列など、ウェブリクエストの中で、 AWS WAF 悪意のあるスクリプトがないかどうかを調べたい部分を特定します。詳細については、「クロスサイトスクリプティング一致条件の使用」を参照してください。

この時点で、必要に応じてこれらの条件を作成するか、「ステップ 8: ルールを作成して条件を追加する」に進むことができます。

ステップ 8: ルールを作成して条件を追加する

AWS WAF Classic にウェブリクエストで検索させたい条件を指定するルールを作成します。ルールに複数の条件を追加した場合、 AWS WAF Classic がそのルールに基づいてリクエストを許可または拒否するには、ウェブリクエストがルールのすべての条件に一致する必要があります。

> **Note**
>
> ルールの詳細については、「ルールの使用」を参照してください。

ルールを作成して条件を追加するには

1. [Create rules] (ルールの作成) ページで、[Create rule] (ルールの作成) を選択します。
2. [Create rule] (ルールの作成) ダイアログボックスで、次の値を入力します。

[Name] (名前)

名前を入力します。

CloudWatch 指標名

AWS WAF Classic CloudWatch が作成してルールに関連付ける指標の名前を入力します。名前には、英数字のみ (A～Z、a～z、0～9) を使用できます。空白を含めることはできません。

[Rule type] (ルールタイプ)

[Regular rule] (通常ルール) または [Rate-based rule] (レートベースのルール) を選択します。レートベースのルールは通常ルールと同じですが、5 分間に識別された IP アドレスから着信したリクエストの数も考慮に入れます。ルールタイプの詳細については、「AWS WAF クラシックの仕組み」を参照してください。この例では、[Regular rule] を選択します。

[Rate limit] (レート制限)

レートベースのルールの場合、ルールの条件に一致する IP アドレスから 5 分間に許可するリクエストの最大数を入力します。

3. ルールに追加する最初の条件として、次の設定を指定します。

   - ウェブリクエストが条件の設定と一致するかしないかに基づいて、 AWS WAF Classic でリクエストを許可するか拒否するかを選択します。

     この例では、[does] (条件に該当) を選択します。
   - ルールに追加する条件のタイプ (IP 一致セット条件、文字列一致セット条件、または SQL インジェクション一致セット条件) を選択します。

     この例では、[originate from IP addresses in] (次の IP アドレスから発信) を選択します。
   - ルールに追加する条件を選択します。

     この例では、前のタスクで作成した IP 一致条件を選択します。
4. [Add condition] (条件を追加) を選択します。
5. 前に作成した Geo 一致条件を追加します。次の値を指定します。

   - [When a request does] (リクエストが条件に該当する場合)
   - [originate from a geographic location in] (発生元の地理的場所)
   - Geo 一致条件を選択します。

6. [Add another condition] (別の条件を追加) を選択します。
7. 前に作成した文字列一致条件を追加します。次の値を指定します。
   - [When a request does] (リクエストが条件に該当する場合)
   - [match at least one of the filters in the string match condition] (文字列一致条件のフィルターの少なくとも 1 つに一致)
   - 文字列一致条件を選択します。
8. [Add condition] (条件を追加) を選択します。
9. 前に作成した SQL インジェクション一致条件を追加します。次の値を指定します。
   - [When a request does] (リクエストが条件に該当する場合)
   - [match at least one of the filters in the SQL injection match condition] (SQL インジェクション一致条件のフィルターの少なくとも 1 つに一致)
   - SQL インジェクション一致条件を選択します。
10. [Add condition] (条件を追加) を選択します。
11. 前に作成したサイズ制約条件を追加します。次の値を指定します。
    - [When a request does] (リクエストが条件に該当する場合)
    - [match at least one of the filters in the size constraint condition] (サイズ制約条件のフィルターの少なくとも 1 つに一致)
    - サイズ制約条件を選択します。
12. 正規表現条件など、他の条件を作成した場合は、同様の方法でそれらの条件を追加します。
13. [Create] (作成) を選択します。
14. [Default action] (デフォルトのアクション) で、[Allow all requests that don't match any rules] (ルールに一致しないすべてのリクエストを許可) を選択します。
15. [Review and create] (確認および作成) を選択します。

ステップ 9: ルールをウェブ ACL に追加する

ルールをウェブ ACL に追加するときは、次の設定を指定します。

- ルールのすべての条件 (リクエストの許可、拒否、カウントなど) に一致するウェブリクエストに対して AWS WAF Classic で実行させたいアクション。

- ウェブ ACL のデフォルトアクション。ルール内のすべての条件に一致しないウェブリクエストに対して AWS WAF Classic に実行させたいアクション、つまり、リクエストを許可または拒否するアクションです。

AWS WAF Classic は、以下のすべての条件 (およびユーザーが追加した可能性のあるその他の条件) CloudFront に一致するウェブリクエストのブロックを開始します。

- `User-Agent` ヘッダーの値が `BadBot`
- (正規表現条件を作成して追加した場合) Body の値がパターン `I[a@]mAB[a@]dRequest` と一致する 4 つの文字列のいずれか
- リクエストの発生元が 192.0.2.0 ~ 192.0.2.255 の範囲の IP アドレス
- リクエストの送信元が Geo 一致条件で選択した国
- リクエスト内のクエリ文字列に悪意のある可能性がある SQL コードが含まれる

AWS WAF Classic では CloudFront 、これら 3 つの条件をすべて満たさないリクエストにも応答できます。

ステップ 10: リソースをクリーンアップする

これでチュートリアルは完了です。 AWS WAF アカウントに追加のクラシック料金が発生しないようにするには、 AWS WAF 作成したクラシックオブジェクトをクリーンアップする必要があります。または、許可、ブロック、カウントするウェブリクエストに合わせて設定を変更することもできます。

> **Note**
>
> AWS 通常、このチュートリアルで作成したリソースに対して請求されるのは、1 日あたり 0.25 USD 未満です。終了したら、不要な料金が発生しないようにリソースを削除することをお勧めします。

AWS WAF Classic が課金するオブジェクトを削除するには

1. ウェブ ACL CloudFront とディストリビューションの関連付けを解除します。

   a. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。

b. 削除するウェブ ACL の名前を選択します。これにより、右ペインで、ウェブ ACL の詳細を含むページが開きます。

c. 右ペインの [Rules] (ルール) タブで、[AWS resources using this web ACL] (このウェブ ACL を使用する リソース) セクションに移動します。ウェブ ACL CloudFront を関連付けたディストリビューションでは、Type 列の x を選択します。

2. ルールから条件を削除する:

a. ナビゲーションペインで [Rules] (ルール) を選択します。

b. チュートリアルで作成したルールを選択します。

c. [Edit rule] (ルールの編集) を選択します。

d. 各条件見出しの右にある [x] を選択します。

e. [Update] (更新) を選択します。

3. ウェブ ACL からルールを削除し、ウェブ ACL を削除する:

a. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。

b. チュートリアルで作成したウェブ ACL の名前を選択します。これにより、右ペインで、ウェブ ACL の詳細を含むページが開きます。

c. [Rules] (ルール) タブで、[Edit web ACL] (ウェブ ACL を編集) を選択します。

d. 各ルール見出しの右にある [x] を選択します。

e. [Actions] (アクション) を選択してから、[Delete web ACL] (ウェブ ACL を削除) を選択します。

4. ルールを削除する:

a. ナビゲーションペインで [Rules] (ルール) を選択します。

b. チュートリアルで作成したルールを選択します。

c. [Delete] (削除) を選択します。

d. [Delete] (削除) のダイアログボックスで、再度 [Delete] (削除) を選択して確認します。

AWS WAF Classic では条件は課金されませんが、クリーンアップを完了したい場合は、次の手順を実行して条件からフィルターを削除し、条件を削除します。

フィルターと条件を削除するには

1. IP 一致条件で IP アドレス範囲を削除し、IP 一致条件を削除します。
   a. AWS WAF Classic コンソールのナビゲーションペインで、[IP アドレス] を選択します。
   b. チュートリアルで作成した IP 一致条件を選択します。
   c. 追加した IP アドレス範囲のチェックボックスをオンにします。
   d. [Delete IP address or range] (IP アドレスまたは範囲を削除) を選択します。
   e. [IP match conditions] (IP 一致条件) ペインで、[Delete] (削除) を選択します。
   f. [Delete] (削除) のダイアログボックスで、再度 [Delete] (削除) を選択して確認します。
2. SQL インジェクションでフィルターを削除し、SQL インジェクション一致条件を削除する:
   a. ナビゲーションペインで、[SQL injection] (SQL インジェクション) を選択します。
   b. チュートリアルで作成した SQL インジェクション一致条件を選択します。
   c. 追加したフィルターのチェックボックスをオンにします。
   d. [Delete filter] (フィルターを削除) を選択します。
   e. [SQL injection match conditions] (SQL インジェクション一致条件) ペインで、[Delete] (削除) を選択します。
   f. [Delete] (削除) のダイアログボックスで、再度 [Delete] (削除) を選択して確認します。
3. 文字列一致条件でフィルターを削除し、文字列一致条件を削除します。
   a. ナビゲーションペインで、[String and regex matching] (文字列および正規表現の一致) を選択します。
   b. チュートリアルで作成した文字列一致条件を選択します。
   c. 追加したフィルターのチェックボックスをオンにします。
   d. [Delete filter] (フィルターを削除) を選択します。
   e. [String match conditions] (文字列一致条件) ペインで、[Delete] (削除) を選択します。
   f. [Delete] (削除) のダイアログボックスで、再度 [Delete] (削除) を選択して確認します。
4. フィルターを作成した場合は、正規表現一致条件でフィルタを削除してから、正規表現一致条件を削除します。
   a. ナビゲーションペインで、[String and regex matching] (文字列および正規表現の一致) を選択します。
   b. チュートリアルで作成した正規表現一致条件を選択します。

c. 追加したフィルターのチェックボックスをオンにします。
d. [Delete filter] (フィルターを削除) を選択します。
e. [Regex match conditions] (正規表現一致条件) ペインで、[Delete] (削除) を選択します。
f. [Delete] (削除) のダイアログボックスで、再度 [Delete] (削除) を選択して確認します。

5. サイズ制約条件でフィルターを削除し、サイズ制約条件を削除します。

a. ナビゲーションペインで、[Size constraints] (サイズ制約) を選択します。
b. チュートリアルで作成したサイズ制約条件を選択します。
c. 追加したフィルターのチェックボックスをオンにします。
d. [Delete filter] (フィルターを削除) を選択します。
e. [Size constraint conditions] (サイズ制約の条件) ペインで、[Delete] (削除) を選択します。
f. [Delete] (削除) のダイアログボックスで、再度 [Delete] (削除) を選択して確認します。

ウェブアクセスコントロールリスト (ウェブ ACL) の作成と設定

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

ウェブアクセスコントロールリスト (ウェブ ACL) を使用すると、Amazon API Gateway API、Amazon CloudFront ディストリビューション、またはApplication Load Balancer が応答するウェブリクエストをきめ細かく制御できます。次の種類のリクエストを許可またはブロックすることができます。

- 特定の IP アドレスまたは IP アドレス範囲が送信元である
- 特定の国が送信元である
- リクエストの特定の部分が、指定した文字列を含むか、正規表現パターンと一致する
- 指定した長さを超えている
- 悪意のある SQL コード (通称 SQL インジェクション) が含まれている可能性がある

- 悪意のあるスクリプト (通称クロスサイトスクリプティング) が含まれている可能性がある

また、これらの条件の組み合わせをテストしたり、指定された条件を満たすだけでなく、5 分間にわたって指定された数のリクエストを超えるウェブリクエストをブロックまたはカウントすることもできます。

コンテンツへのアクセスを許可またはブロックするリクエストを選り分けるには、次のタスクを実行します。

1. ウェブリクエストが指定条件のいずれとも一致しない場合のデフォルトアクション (許可またはブロック) を選択します。詳細については、「ウェブ ACL のデフォルトアクションの決定」を参照してください。
2. リクエストを許可またはブロックする条件を指定します。
   - 悪意のあるスクリプトが含まれている可能性があるかどうかに基づいてリクエストを許可またはブロックするには、クロスサイトスクリプティング一致条件を作成します。詳細については、「クロスサイトスクリプティング一致条件の使用」を参照してください。
   - 送信元の IP アドレスに基づいてリクエストを許可またはブロックするには、IP 一致条件を作成します。詳細については、「IP 一致条件の使用」を参照してください。
   - 送信元の国に基づいてリクエストを許可またはブロックするには、Geo 一致条件を作成します。詳細については、「Geo (地理的) 一致条件の使用」を参照してください。
   - 指定した長さを超えているかどうかに基づいてリクエストを許可またはブロックするには、サイズ制約条件を作成します。詳細については、「サイズ制約条件の使用」を参照してください。
   - 悪意のある SQL コードが含まれている可能性があるかどうかに基づいてリクエストを許可またはブロックするには、SQL インジェクション一致条件を作成します。詳細については、「SQL インジェクション一致条件の使用」を参照してください。
   - 含まれている文字列に基づいてリクエストを許可またはブロックするには、文字列一致条件を作成します。詳細については、「文字列一致条件の使用」を参照してください。
   - リクエストに含まれる正規表現パターンと一致する文字列に基づいてリクエストを許可またはブロックするには、正規表現一致条件を作成します。詳細については、「正規表現一致条件の使用」を参照してください。
3. 条件を 1 つまたは複数のルールに追加します。同じルールに複数の条件を追加する場合、 AWS WAF Classic がルールに基づいてリクエストを許可または拒否するには、ウェブリクエストがすべての条件に一致する必要があります。詳細については、「ルールの使用」を参照してくださ

い。必要に応じて、通常のルールの代わりにレートベースのルールを使用して、条件を満たす IP アドレスからのリクエスト数を制限できます。

4. ルールをウェブ ACL に追加します。ルールごとに、ルールに追加した条件に基づいて AWS WAF Classic でリクエストを許可するか拒否するかを指定します。ウェブ ACL に複数のルールを追加すると、 AWS WAF Classic はウェブ ACL にリストされている順序でルールを評価します。詳細については、「ウェブ ACL の使用」を参照してください。

   新しいルールを追加したり、既存のルールを更新したりすると、該当するすべてのウェブ ACL およびリソースでそれらの変更がアクティブになるまでに、最大 1 分かかることがあります。

トピック

条件の使用

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

リクエストを許可またはブロックする場合に条件を指定します。

- 悪意のあるスクリプトが含まれている可能性があるかどうかに基づいてリクエストを許可またはブロックするには、クロスサイトスクリプティング一致条件を作成します。詳細については、「クロスサイトスクリプティング一致条件の使用」を参照してください。
- 送信元の IP アドレスに基づいてリクエストを許可またはブロックするには、IP 一致条件を作成します。詳細については、「IP 一致条件の使用」を参照してください。
- 送信元の国に基づいてリクエストを許可またはブロックするには、Geo 一致条件を作成します。詳細については、「Geo (地理的) 一致条件の使用」を参照してください。

- 指定した長さを超えているかどうかに基づいてリクエストを許可またはブロックするには、サイズ制約条件を作成します。詳細については、「サイズ制約条件の使用」を参照してください。
- 悪意のある SQL コードが含まれている可能性があるかどうかに基づいてリクエストを許可またはブロックするには、SQL インジェクション一致条件を作成します。詳細については、「SQL インジェクション一致条件の使用」を参照してください。
- 含まれている文字列に基づいてリクエストを許可またはブロックするには、文字列一致条件を作成します。詳細については、「文字列一致条件の使用」を参照してください。
- リクエストに含まれる正規表現パターンと一致する文字列に基づいてリクエストを許可またはブロックするには、正規表現一致条件を作成します。詳細については、「正規表現一致条件の使用」を参照してください。

トピック

クロスサイトスクリプティング一致条件の使用

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

攻撃者は、ウェブアプリケーションの脆弱性を悪用するために、ウェブリクエスト内にスクリプトを挿入する場合があります。URI やクエリ文字列など、 AWS WAF Classic に悪意のあるスクリプトがないか検査させたいウェブリクエストの部分を特定するために、1 つ以上のクロスサイトスクリプ

ティング一致条件を作成できます。後でウェブ ACL を作成するときに、悪意のあるスクリプトが含まれている可能性があるリクエストを許可するかブロックするかを指定します。

トピック

クロスサイトスクリプティング一致条件の作成

クロスサイトスクリプティング一致条件を作成する場合は、フィルターを指定します。フィルターは、URI やクエリ文字列など、 AWS WAF Classic に悪意のあるスクリプトがないか検査させたいウェブリクエストの部分を示します。クロスサイトスクリプティング一致条件には複数のフィルターを追加できます。条件ごとに 1 つのフィルターを設定することもできます。各設定が AWS WAF Classic の動作にどのように影響するかを次に示します。

- クロスサイトスクリプティング一致条件ごとに複数のフィルター (推奨) — 複数のフィルターを含むクロスサイトスクリプティング一致条件をルールに追加し、そのルールをウェブ ACL に追加する場合、ウェブリクエストはクロスサイトスクリプティング一致条件の 1 つのフィルターにのみ一致する必要があります。 AWS WAF Classic がその条件に基づいてリクエストを許可または拒否するには、その条件に基づいてリクエストを許可または拒否します。

  例えば、1 つのクロスサイトスクリプティング一致条件を作成し、その条件に 2 つのフィルターを含めたとします。1 つのフィルターは AWS WAF Classic に URI に悪意のあるスクリプトがないか検査するように指示し、もう 1 つのフィルターは AWS WAF Classic にクエリ文字列を検査するように指示します。 AWS WAF Classic は、URI またはクエリ文字列に悪意のあるスクリプトが含まれていると思われるリクエストを許可またはブロックします。
- クロスサイトスクリプティング一致条件ごとに 1 つのフィルター — 個別のクロスサイトスクリプティング一致条件をルールに追加し、そのルールをウェブ ACL に追加する場合、ウェブリクエストはすべての条件に一致する必要があります。 AWS WAF Classic はその条件に基づいてリクエストを許可または拒否します。

  2 つの条件を作成し、各条件に前の例で示した 2 つのフィルターの 1 つを別個に含めたとします。両方の条件を同じルールに追加し、そのルールをウェブ ACL に追加すると、 AWS WAF Classic は URI とクエリ文字列の両方に悪意のあるスクリプトが含まれていると思われる場合にのみリクエストを許可または拒否します。

> **Note**
>
> クロスサイトスクリプティング一致条件をルールに追加すると、悪意のあるスクリプトを含んでいないと思われるウェブリクエストを許可またはブロックするように AWS WAF Classic を設定することもできます。

クロスサイトスクリプティング一致条件を作成するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで、[Cross-site scripting] (クロスサイトスクリプティング) を選択します。
3. [Create condition] (条件を作成) を選択します。
4. 適用するフィルター設定を指定します。詳細については、「クロスサイトスクリプティング一致条件の作成時または編集時に指定する値」を参照してください。
5. [Add another filter] (別のフィルターを追加) を選択します。
6. 別のフィルターを追加する場合は、ステップ 4～5 を繰り返します。
7. フィルターの追加が完了したら、[Create] (作成) を選択します。

クロスサイトスクリプティング一致条件の作成時または編集時に指定する値

クロスサイトスクリプティング一致条件を作成または更新するときに、次の値を指定します。

[Name] (名前)

クロスサイトスクリプティング一致条件の名前。

名前に使用できるのは A～Z、a～z、0～9 の英数字と特殊文字 _-!"#`+*},./ です。一度作成した条件の名前は変更できません。

[Part of the request to filter on] (フィルタリングするリクエストの一部)

各ウェブリクエストの中で、 AWS WAF Classic に悪意のあるスクリプトがないか検査させたい部分を選択します。

[Header] (ヘッダー)

指定したリクエストヘッダー (`User-Agent` や `Referer` など)。[Header] (ヘッダー) を選択した場合は、[Header] (ヘッダー) フィールドにヘッダー名を指定します。

[HTTP method] (HTTP メソッド)

リクエストがオリジンに実行を要求しているオペレーションのタイプを示す HTTP メソッド。CloudFront `DELETE`、、、、、`GET``HEAD``OPTIONS``PATCH``POST`、`PUT`およびの各メソッドをサポートします。

[Query string] (クエリ文字列)

URL 内で ? 文字の後に続く部分 (ある場合)。

> **Note**
>
> クロスサイトスクリプティングの一致条件については、[Part of the request to filter on] (フィルタリングするリクエストの一部) に [Query string] (クエリ文字列) ではなく、[All query parameters (values only)] (すべてのクエリパラメータ (値のみ)) を選択することをお勧めします。

[URI]

リクエストの URI パス。リソースを識別します (例: `/images/daily-ad.jpg`)。これには、URI のクエリ文字列またはフラグメントコンポーネントは含まれません。詳細については、「Uniform Resource Identifier (URI): 一般的な構文」を参照してください。

Transformation が指定されていない限り、URI は正規化されず、 AWS リクエストの一部としてクライアントから受け取った時点で検査されます。[Transformation] (変換) が指定されている場合、URI はその指定に従って形式が再設定されます。

[Body] (本文)

リクエスト内で、HTTP リクエストの本文としてウェブサーバーに送信する追加データ (フォームのデータなど) を含む部分。

> **Note**
>
> [Part of the request to filter on] (フィルタリングするリクエストの一部) の値として [Body] (本文) を選択した場合、 AWS WAF Classic によって最初の 8,192 バイト (8

KB) のみが検査されます。本文が 8,192 バイトより長いリクエストを許可またはブロックするには、サイズ制約条件を作成します。(AWS WAF Classic はリクエストヘッダーから本文の長さを取得します)。詳細については、「サイズ制約条件の使用」を参照してください。

[Single query parameter (value only)] (単一クエリパラメータ (値のみ))

クエリ文字列の一部として定義されているすべてのパラメータです。たとえば、URL が「www.xyz.com? UserName =abc& SalesRegion =seattle」の場合、またはパラメータにフィルタを追加できます。UserNameSalesRegion

[Single query parameter (value only)] (単一クエリパラメータ (値のみ)) を選択する場合は、[Query parameter name] (クエリパラメータ名) も指定します。これは、やなど、調べるクエリ文字列内のパラメータです。UserNameSalesRegion[Query parameter name] (クエリパラメータ名) の最大長は 30 文字です。[Query parameter name] (クエリパラメータ名) では、大文字と小文字は区別されません。たとえば、Query UserNameパラメータ名として指定すると、username や UserName などのすべてのバリエーションにマッチします。UserName

[All query parameters (values only)] (すべてのクエリパラメータ (値のみ))

単一クエリパラメーター (値のみ) と似ていますが、 AWS WAF Classic は 1 つのパラメーターの値を検査するのではなく、クエリ文字列内のすべてのパラメーター値を検査して、悪意のあるスクリプトがないかを調べます。たとえば、URL が「www.xyz.com? UserName =abc& SalesRegion =theatle」で、「すべてのクエリパラメーター (値のみ)」を選択した場合、 AWS WAF Classic は、その値に悪意のあるスクリプトが含まれているか、その値に悪意のあるスクリプトが含まれていると、一致をトリガーします。UserNameSalesRegion

[Header] (ヘッダー)

フィルターするリクエストの一部に [ヘッダー] を選択した場合は、一般的なヘッダーのリストからヘッダーを選択するか、Classic AWS WAF に悪意のあるスクリプトがないか検査させたいヘッダーの名前を入力します。

[Transformation] (変換)

AWS WAF Classic がリクエストを検査する前に、変換によってウェブリクエストが再フォーマットされます。これにより、攻撃者が Classic をバイパスしようとしてウェブリクエストで使用する、通常とは異なるフォーマットの一部が排除されます。 AWS WAF

1 種類のテキスト変換しか指定できません。

変換では次の操作を実行できます。

なし

AWS WAF Classic は、Value の文字列が一致するかどうかを調べるまで、ウェブリクエストのテキスト変換を一切行いません。

[Convert to lowercase] (小文字に変換)

AWS WAF Classic は大文字 (A ~ Z) を小文字 (a ~ z) に変換します。

[HTML decode] (HTML デコード)

AWS WAF Classic は HTML でエンコードされた文字をエンコードされていない文字に置き換えます。

- " を & に置き換えます。
-   を改行なしスペースに置き換えます。
- < を < に置き換えます。
- > を > に置き換えます。
- 16 進数形式の文字 (&#xhhhh;) を対応する文字に置き換えます。
- 10 進数形式の文字 (&#nnnn;) を対応する文字に置き換えます。

[Normalize white space] (空白の正規化)

AWS WAF Classic では、次の文字がスペース文字 (10 進数 32) に置き換えられます。

- \f、フォームフィード、10 進数 12
- \t、タブ、10 進数 9
- \n、改行、10 進数 10
- \r、キャリッジリターン、10 進数 13
- \v、垂直タブ、10 進数 11
- 改行なしスペース、10 進数 160

さらに、このオプションでは複数のスペースを 1 つのスペースに置き換えます。

[Simplify command line] (コマンドラインを簡素化)

オペレーティングシステムのコマンドラインのコマンドが含まれているリクエストの場合、このオプションを使用して次の変換を行います。

- 次の文字を削除します: \ " ' ^
- 次の文字の前にあるスペースを削除します: / (

- 次の文字をスペースに置き換えます: , ;
- 複数のスペースを 1 つのスペースに置き換えます。
- 大文字 (A〜Z) を小文字 (a〜z) に変換します。

[URL decode] (URL デコード)

URL エンコードされたリクエストをデコードします。

クロスサイトスクリプティング一致条件のフィルターの追加と削除

クロスサイトスクリプティング一致条件のフィルターを追加または削除できます。フィルターを変更するには、新しいフィルターを追加して古いフィルターを削除します。

クロスサイトスクリプティング一致条件のフィルターを追加または削除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで、[Cross-site scripting] (クロスサイトスクリプティング) を選択します。
3. フィルターを追加または削除する対象の条件を選択します。
4. フィルターを追加するには、次のステップを実行します。

   a. [Add filter] (フィルターを追加) を選択します。
   b. 適用するフィルター設定を指定します。詳細については、「クロスサイトスクリプティング一致条件の作成時または編集時に指定する値」を参照してください。
   c. [Add] (追加) を選択します。
5. フィルターを削除するには、次のステップを実行します。

   a. 削除するフィルターを選択します。
   b. [Delete filter] (フィルターを削除) を選択します。

クロスサイトスクリプティング一致条件の削除

クロスサイトスクリプティング一致条件を削除するには、最初にその条件からすべてのフィルターを削除し、その条件を使用しているすべてのルールから条件自体を削除します。次に手順を示します。

クロスサイトスクリプティング一致条件を削除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF にあるコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。

2. ナビゲーションペインで、[Cross-site scripting] (クロスサイトスクリプティング) を選択します。
3. [Cross-site scripting match conditions] (クロスサイトスクリプティング一致条件) ペインで、削除するクロスサイトスクリプティング一致条件を選択します。
4. 右ペインで、[Associated rules] (関連付けられたルール) タブを選択します。

   このクロスサイトスクリプティング一致条件を使用しているルールのリストが空の場合は、ステップ 6 に進みます。リストにルールが含まれている場合は、ルールを書き留めて、ステップ 5 に進みます。

5. クロスサイトスクリプティング一致条件を使用しているルールから、この条件を削除するには、次のステップを実行します。

   a. ナビゲーションペインで [Rules] (ルール) を選択します。
   b. 削除するクロスサイトスクリプティング一致条件を使用しているルールの名前を選択します。
   c. 右ペインで、ルールから削除するクロスサイトスクリプティング一致条件を選択し、[Remove selected condition] (選択した条件を削除) を選択します。
   d. 削除するクロスサイトスクリプティング一致条件を使用しているすべての残りのルールに対してステップ b とステップ c を繰り返します。
   e. ナビゲーションペインで、[Cross-site scripting] (クロスサイトスクリプティング) を選択します。
   f. [Cross-site scripting match conditions] (クロスサイトスクリプティング一致条件) ペインで、削除するクロスサイトスクリプティング一致条件を選択します。

6. [Delete] (削除) を選択して、選択した条件を削除します。

IP 一致条件の使用

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

リクエストの送信元の IP アドレスに基づいてウェブリクエストを許可またはブロックする場合は、IP 一致条件を作成します。IP 一致条件は、リクエストの送信元の IP アドレスまたは IP アドレス範囲を 10,000 件までリストアップします。後でウェブ ACL を作成するときに、これらの IP アドレスからのリクエストを許可するかブロックするかを指定します。

トピック

- IP 一致条件の作成
- IP 一致条件の編集
- IP 一致条件の削除

IP 一致条件の作成

リクエストの送信元の IP アドレスに基づいて許可またはブロックするウェブリクエストを選り分けるには、許可する IP アドレス用に 1 つの IP 一致条件を作成し、ブロックする IP アドレス用に別の IP 一致条件を作成します。

> **Note**
>
> IP 一致条件をルールに追加すると、条件で指定した IP アドレス以外からのウェブリクエストを許可または拒否するように AWS WAF Classic を設定することもできます。

IP 一致条件を作成するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。

2. ナビゲーションペインで、[IP addresses] (IP アドレス) を選択します。
3. [Create condition] (条件を作成) を選択します。
4. [Name] (名前) フィールドに名前を入力します。

   名前に使用できるのは英数字 (A ~ Z、a ~ z、0 ~ 9) または特殊文字 _-!"#`+*},./ です。一度作成した条件の名前は変更できません。
5. 正しい IP バージョンを選択し、IP アドレスまたは IP アドレス範囲を CIDR 表記で指定します。次に例を示します。

   - IPv4 アドレス 192.0.2.44 を指定するには、192.0.2.44/32 と入力します。
   - IPv6 アドレス 0:0:0:0:0:ffff:c000:22c を指定するには、0:0:0:0:0:ffff:c000:22c/128 と入力します。
   - IPv4 アドレス範囲 192.0.2.0 ~ 192.0.2.255 を指定するには、192.0.2.0/24 と入力します。
   - IPv6 アドレス範囲の 2620:0:2d0:200:0:0:0:0 ~ 2620:0:2d0:200:ffff:ffff:ffff:ffff を指定するには、2620:0:2d0:200::/64 と入力します。

   AWS WAF Classic は IPv4 アドレス範囲 (/8) と /16 ~ /32 の任意の範囲をサポートします。AWS WAF クラシックは IPv6 アドレス範囲 (/24、/32、/48、/56、/64、/128) をサポートしています。CIDR 表記の詳細については、Wikipedia の「Classless Inter-Domain Routing」(クラスレスドメイン間ルーティング) を参照してください。
6. [Add another IP address or range] (別の IP アドレスまたは範囲を追加) を選択します。
7. 別の IP アドレスまたは IP アドレス範囲を追加する場合は、ステップ 5〜6 を繰り返します。
8. 値の追加が終了したら、[Create IP match condition] (IP 一致条件を作成) を選択します。

IP 一致条件の編集

IP 一致条件の IP アドレス範囲を追加または削除できます。範囲を変更するには、新しい範囲を追加して古い範囲を削除します。

IP 一致条件を編集するには

1. にサインインし、https://console.aws.amazon.com/wafv2/ のコンソールを開きます。 AWS Management ConsoleAWS WAF

ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。

2. ナビゲーションペインで、[IP addresses] (IP アドレス) を選択します。
3. [IP match conditions] (IP 一致条件) ペインで、編集する IP 一致条件を選択します。
4. IP アドレス範囲を追加するには:

   a. ナビゲーションペインで、[Add IP address or range] (IP アドレスまたは範囲を追加) を選択します。

   b. 正しい IP バージョンを選択し、IP アドレス範囲を CIDR 表記で入力します。次に例を示します。

      - IPv4 アドレス 192.0.2.44 を指定するには、192.0.2.44/32 と入力します。
      - IPv6 アドレス 0:0:0:0:0:ffff:c000:22c を指定するには、0:0:0:0:0:ffff:c000:22c/128 と入力します。
      - IPv4 アドレス範囲 192.0.2.0 ~ 192.0.2.255 を指定するには、192.0.2.0/24 と入力します。
      - IPv6 アドレス範囲の 2620:0:2d0:200:0:0:0:0 ~ 2620:0:2d0:200:ffff:ffff:ffff:ffff を指定するには、2620:0:2d0:200::/64 と入力します。

      AWS WAF Classic は IPv4 アドレス範囲 (/8) と /16 ~ /32 の任意の範囲をサポートします。 AWS WAF クラシックは IPv6 アドレス範囲 (/24、/32、/48、/56、/64、/128) をサポートしています。CIDR 表記の詳細については、Wikipedia の「Classless Inter-Domain Routing」(クラスレスドメイン間ルーティング) を参照してください。

   c. IP アドレスをさらに追加するには、[Add another IP address] (別の IP アドレスの追加) を選択して、その値を入力します。

   d. [Add] (追加) を選択します。

5. IP アドレスまたは IP アドレス範囲を削除するには:

   a. 右ペインで、削除する値を選択します。

   b. [Delete IP address or range] (IP アドレスまたは範囲を削除) を選択します。

IP 一致条件の削除

IP 一致条件を削除するには、最初にすべての IP アドレスおよび IP アドレス範囲を条件から削除し、その条件を使用しているすべてのルールから条件自体を削除します。次に手順を示します。

IP 一致条件を削除するには

1. にサインインし、https://console.aws.amazon.com/wafv2/ のコンソールを開きます。 AWS Management ConsoleAWS WAF

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで、[IP addresses] (IP アドレス) を選択します。
3. [IP match conditions] (IP 一致条件) ペインで、削除する IP 一致条件を選択します。
4. 右ペインで、[Rules] (ルール) タブを選択します。

   この IP 一致条件を使用しているルールのリストが空の場合は、ステップ 6 に進みます。リストにルールが含まれている場合は、ルールを書き留めて、ステップ 5 に進みます。
5. IP 一致条件を使用しているルールから、この条件を削除するには、次のステップを実行します。

   a. ナビゲーションペインで [Rules] (ルール) を選択します。
   b. 削除する IP 一致条件を使用しているルールの名前を選択します。
   c. 右ペインで、ルールから削除する IP 一致条件を選択し、[Remove selected condition] (選択した条件を削除) を選択します。
   d. 削除する IP 一致条件を使用しているすべての残りのルールに対してステップ b とステップ c を繰り返します。
   e. ナビゲーションペインで、[IP match conditions] (IP 一致条件) を選択します。
   f. [IP match conditions] (IP 一致条件) ペインで、削除する IP 一致条件を選択します。
6. [Delete] (削除) を選択して、選択した条件を削除します。

Geo (地理的) 一致条件の使用

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行してい

ない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

リクエスト送信元の国に基づいてウェブリクエストを許可またはブロックする場合は、Geo 一致条件を作成します。Geo 一致条件は、リクエスト送信元の国のリストを返します。後でウェブ ACL を作成するときに、それらの国からのリクエストを許可するかブロックするかを指定します。

AWS WAF ジオマッチ条件を他のクラシック条件やルールと組み合わせて使用すると、高度なフィルタリングを構築できます。例えば、特定の国をブロックするがその国の特定の IP アドレスを許可する場合は、Geo 一致条件と IP 一致条件を含むルールを作成できます。その国から送信され、かつ承認済みの IP アドレスと一致しないリクエストをブロックするようにルールを設定します。別の例として、特定の国のユーザーのリソースを優先させる場合は、2 つの異なるレートベースのルールに Geo 一致条件を含めることができます。優先させる国のユーザーのレート制限をより高く設定し、他のすべてのユーザーのレート制限をより低く設定します。

> **Note**
>
> CloudFront 地域制限機能を使用して、ある国がコンテンツにアクセスすることをブロックしている場合、その国からのリクエストはすべてブロックされ、クラシックには転送されません。 AWS WAF そのため、地域やその他の AWS WAF Classic 条件に基づいてリクエストを許可または拒否したい場合は、地域制限機能を使用しないでください。 CloudFront 代わりに、 AWS WAF クラシックジオマッチ条件を使用してください。

トピック

- Geo 一致条件の作成
- Geo 一致条件の編集
- Geo 一致条件の削除

Geo 一致条件の作成

リクエスト送信元の国に基づいて許可またはブロックするウェブリクエストを選り分けるには、許可する国用に 1 つの Geo 一致条件を作成し、ブロックする国用に別の Geo 一致条件を作成します。

> **Note**
>
> 地域一致条件をルールに追加すると、条件で指定した国以外から発信されたウェブリクエストを許可または拒否するように AWS WAF Classic を構成することもできます。

Geo 一致条件を作成するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで、[Geo match] (Geo 一致) を選択します。
3. [Create condition] (条件を作成) を選択します。
4. [Name] (名前) フィールドに名前を入力します。

   名前に使用できるのは英数字 (A ~ Z、a ~ z、0 ~ 9) または特殊文字 _-!"#`+*},./ です。一度作成した条件の名前は変更できません。
5. [Region] (リージョン) を選択します。
6. [Location type] (場所のタイプ) と国を選択します。現時点では、[Location type] (場所のタイプ) は [Country] (国) にのみ設定できます。
7. [Add location] (場所を追加) を選択します。
8. [Create] (作成) を選択します。

Geo 一致条件の編集

Geo 一致条件に対して国を追加したり削除したりできます。

Geo 一致条件を編集するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF にあるコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで、[Geo match] (Geo 一致) を選択します。

3. [Geo match conditions] (Geo 一致条件) ペインで、編集する Geo 一致条件を選択します。
4. 国を追加するには:
   a. 右側のペインで、[Add filter] (フィルターを追加) を選択します。
   b. [Location type] (場所のタイプ) と国を選択します。現時点では、[Location type] (場所のタイプ) は [Country] (国) にのみ設定できます。
   c. [Add] (追加) を選択します。
5. 国を削除するには:
   a. 右ペインで、削除する値を選択します。
   b. [Delete filter] (フィルターを削除) を選択します。

Geo 一致条件の削除

Geo 一致条件を削除するには、まずその条件からすべての国を削除し、その条件を使用しているすべてのルールから条件自体を削除します。その手順を次に示します。

Geo 一致条件を削除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF にあるコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. Geo 一致条件を使用しているルールからその条件を削除します。
   a. ナビゲーションペインで [Rules] (ルール) を選択します。
   b. 削除する Geo 一致条件を使用しているルールの名前を選択します。
   c. 右側のペインで、[Edit rule] (ルールを編集) を選択します。
   d. 削除する条件の横にある [X] を選択します。
   e. [Update] (更新) を選択します。
   f. 削除する Geo 一致条件を使用している残りのすべてのルールに対してこの同じ手順を繰り返します。
3. 削除する条件からフィルターを削除します。
   a. ナビゲーションペインで、[Geo match] (Geo 一致) を選択します。

b. 削除する Geo 一致条件の名前を選択します。

c. 右側のペインで、[Filter] (フィルター) の横にあるチェックボックスをオンにして、すべてのフィルターを選択します。

d. [Delete filter] (フィルターを削除) を選択します。

4. ナビゲーションペインで、[Geo match] (Geo 一致) を選択します。

5. [Geo match conditions] (Geo 一致条件) ペインで、削除する Geo 一致条件を選択します。

6. [Delete] (削除) を選択して、選択した条件を削除します。

サイズ制約条件の使用

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

指定したウェブリクエスト部分の長さに基づいてリクエストを許可またはブロックする場合は、サイズ制約条件を作成します。サイズ制約条件は、Classic で検索させたいウェブリクエストの部分、AWS WAF Classic AWS WAF に検索させたいバイト数、および演算子 (「より大きい (>)」や「より小さい (<)」などの演算子を識別します。例えば、サイズ制約条件を使用して 100 バイトよりも長いクエリ文字列を探すことができます。後でウェブ ACL を作成するときに、これらの設定に基づいてリクエストを許可するかブロックするかを指定します。

AWS WAF Classic がリクエスト本文を検査するように設定した場合 (たとえば、本文から特定の文字列を検索するなど)、 AWS WAF Classic は最初の 8192 バイト (8 KB) だけを検査することに注意してください。ウェブリクエストの本文が 8192 バイトを超えない場合は、サイズ制約条件を作成して、8192 バイトを超える本文を持つリクエストをブロックできます。

トピック

- サイズ制約条件の作成
- サイズ制約条件の作成時または編集時に指定する値
- サイズ制約条件のフィルターの追加と削除

- サイズ制約条件の削除

サイズ制約条件の作成

サイズ制約条件を作成するときは、 AWS WAF Classic に長さを評価させたいウェブリクエストの部分を特定するフィルターを指定します。サイズ制約条件には複数のフィルターを追加できます。条件ごとに 1 つのフィルターを設定することもできます。各設定が AWS WAF Classic の動作にどのように影響するかを次に示します。

- サイズ制約条件ごとに 1 つのフィルター — 個別のサイズ制約条件をルールに追加し、そのルールをウェブ ACL に追加する場合、ウェブリクエストはすべての条件に一致する必要があります。AWS WAF クラシックでは、条件に基づいてリクエストを許可または拒否します。

  例えば、2 つの条件を作成するとします。1 つの条件は、クエリ文字列が 100 バイトを超えるウェブリクエストに一致します。もう 1 つの条件は、リクエストボディが 1024 バイトを超えるウェブリクエストに一致します。両方の条件を同じルールに追加し、そのルールをウェブ ACL に追加すると、 AWS WAF クラシックは両方の条件が満たされる場合にのみリクエストを許可または拒否します。
- サイズ制約条件ごとに複数のフィルター — 複数のフィルターを含むサイズ制約条件をルールに追加し、そのルールをウェブ ACL に追加する場合、ウェブリクエストはサイズ制約条件のフィルターの 1 つと一致するだけで、 AWS WAF Classic はその条件に基づいてリクエストを許可または拒否できます。

  2 つの条件ではなく 1 つの条件を作成し、その 1 つの条件に前の例と同じ 2 つのフィルターが含まれているとします。 AWS WAF Classic では、クエリ文字列が 100 バイトを超えるか、リクエスト本文が 1024 バイトを超える場合、リクエストを許可または拒否します。

> **Note**
>
> サイズ制約条件をルールに追加すると、条件の値と一致しないウェブリクエストを許可または拒否するように AWS WAF Classic を設定することもできます。

サイズ制約条件を作成するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。

2. ナビゲーションペインで、[Size constraints] (サイズ制約) を選択します。
3. [Create condition] (条件を作成) を選択します。
4. 適用するフィルター設定を指定します。詳細については、「サイズ制約条件の作成時または編集時に指定する値」を参照してください。
5. [Add another filter] (別のフィルターを追加) を選択します。
6. 別のフィルターを追加する場合は、ステップ 4〜5 を繰り返します。
7. フィルターの追加が完了したら、[Create size constraint condition] (サイズ制約の条件を作成) を選択します。

サイズ制約条件の作成時または編集時に指定する値

サイズ制約条件を作成または更新するときに、次の値を指定します。

[Name] (名前)

サイズ制約条件の名前を入力します。

名前に使用できるのは英数字 (A ~ Z、a ~ z、0 ~ 9) または特殊文字 _-!"#`+*},./ です。一度作成した条件の名前は変更できません。

[Part of the request to filter on] (フィルタリングするリクエストの一部)

各ウェブリクエストの中で、 AWS WAF Classic に長さを評価させたい部分を選択します。

[Header] (ヘッダー)

指定したリクエストヘッダー (`User-Agent` や `Referer` など)。[Header] (ヘッダー) を選択した場合は、[Header] (ヘッダー) フィールドにヘッダー名を指定します。

[HTTP method] (HTTP メソッド)

リクエストがオリジンに実行を要求しているオペレーションのタイプを示す HTTP メソッド。CloudFront `DELETE`、、、、、`GETHEADOPTIONSPATCHPOST`、およびの各メソッドをサポートしますPUT。

[Query string] (クエリ文字列)

URL 内で ? 文字の後に続く部分 (ある場合)。

[URI]

リクエストの URI パス。リソースを識別します (例: `/images/daily-ad.jpg`)。これには、URI のクエリ文字列またはフラグメントコンポーネントは含まれません。詳細については、「Uniform Resource Identifier (URI): 一般的な構文」を参照してください。

Transformation が指定されていない限り、URI は正規化されず、 AWS リクエストの一部としてクライアントから受け取った時点で検査されます。[Transformation] (変換) が指定されている場合、URI はその指定に従って形式が再設定されます。

[Body] (本文)

リクエスト内で、HTTP リクエストの本文としてウェブサーバーに送信する追加データ (フォームのデータなど) を含む部分。

[Single query parameter (value only)] (単一クエリパラメータ (値のみ))

クエリ文字列の一部として定義されているすべてのパラメータです。たとえば、URL が「www.xyz.com? UserName =abc& SalesRegion =seattle」の場合、またはパラメータにフィルタを追加できます。UserNameSalesRegion

[Single query parameter (value only)] (単一クエリパラメータ (値のみ)) を選択する場合は、[Query parameter name] (クエリパラメータ名) も指定します。これは、検査するクエリ文字列内のパラメータ (など) です。UserName[Query parameter name] (クエリパラメータ名) の最大長は 30 文字です。[Query parameter name] (クエリパラメータ名) では、大文字と小文字は区別されません。たとえば、Query UserNameパラメータ名として指定すると、username や UserName などのすべてのバリエーションにマッチします。UserName

[All query parameters (values only)] (すべてのクエリパラメータ (値のみ))

単一クエリパラメーター (値のみ) と似ていますが、 AWS WAF Classic は 1 つのパラメーターの値を検査するのではなく、クエリ文字列内のすべてのパラメーターの値にサイズ制約がないか調べます。たとえば、URL が「www.xyz.com? UserName =abc& SalesRegion =seattle」で、「すべてのクエリパラメーター (値のみ)」を選択した場合、 AWS WAF Classic は指定されたサイズのいずれかまたはそれ以上の値と一致する値をトリガーします。UserNameSalesRegion

[Header] (ヘッダー) ([Part of the request to filter on] (フィルタリングするリクエストの一部) が [Header] (ヘッダー) の場合のみ)

フィルターするリクエストの一部に Header を選択した場合は、一般的なヘッダーのリストからヘッダーを選択するか、Classic に長さを評価させたいヘッダーの名前を入力します。 AWS WAF

Comparison operator (比較演算子)

AWS WAF Classic が Size に指定した値を基準にしてウェブリクエスト内のクエリ文字列の長さを評価する方法を選択します。

たとえば、[比較演算子] に [より大きい] を選択し、[サイズ] に 100 と入力すると、 AWS WAF Classic は 100 バイトを超えるクエリ文字列のウェブリクエストを評価します。

サイズ

AWS WAF Classic がクエリ文字列で監視する長さをバイト単位で入力します。

> **Note**
>
> [Part of the request to filter on] (フィルタリングするリクエストの一部) の値に [URI] を選択した場合、URI の / は 1 文字としてカウントされます。例えば、URI パスの `/logo.jpg` は 9 文字の長さになります。

[Transformation] (変換)

AWS WAF Classic がリクエストの指定された部分の長さを評価する前に、変換によってウェブリクエストが再フォーマットされます。これにより、攻撃者が Classic をバイパスしようとしてウェブリクエストで使用する珍しいフォーマットの一部が排除されます。 AWS WAF

> **Note**
>
> フィルターの対象となるリクエストの一部に Body を選択した場合、検査のために最初の 8192 バイトだけが転送されるため、 AWS WAF Classic が変換を実行するように設定することはできません。ただし、HTTP リクエストボディのサイズに基づいてトラフィックをフィルタして、[None] (なし) の変換を指定することはできます。(AWS WAF Classic はリクエストヘッダーから本文の長さを取得します)。

1 種類のテキスト変換しか指定できません。

変換では次の操作を実行できます。

なし

AWS WAF Classic は、長さを確認するまではウェブリクエストのテキスト変換を一切行いません。

[Convert to lowercase] (小文字に変換)

AWS WAF Classic は大文字 (A ~ Z) を小文字 (a ~ z) に変換します。

[HTML decode] (HTML デコード)

AWS WAF Classic は HTML でエンコードされた文字をエンコードされていない文字に置き換えます。

- " を & に置き換えます。
-   を改行なしスペースに置き換えます。
- < を < に置き換えます。
- > を > に置き換えます。
- 16 進数形式の文字 (&#xhhhh;) を対応する文字に置き換えます。
- 10 進数形式の文字 (&#nnnn;) を対応する文字に置き換えます。

[Normalize white space] (空白の正規化)

AWS WAF Classic では、次の文字がスペース文字 (10 進数 32) に置き換えられます。

- \f、フォームフィード、10 進数 12
- \t、タブ、10 進数 9
- \n、改行、10 進数 10
- \r、キャリッジリターン、10 進数 13
- \v、垂直タブ、10 進数 11
- 改行なしスペース、10 進数 160

さらに、このオプションでは複数のスペースを 1 つのスペースに置き換えます。

[Simplify command line] (コマンドラインを簡素化)

オペレーティングシステムのコマンドラインのコマンドが含まれているリクエストの場合、このオプションを使用して次の変換を行います。

- 次の文字を削除します: \ " ' ^
- 次の文字の前にあるスペースを削除します: / (
- 次の文字をスペースに置き換えます: , ;
- 複数のスペースを 1 つのスペースに置き換えます。
- 大文字 (A〜Z) を小文字 (a〜z) に変換します。

[URL decode] (URL デコード)

URL エンコードされたリクエストをデコードします。

サイズ制約条件のフィルターの追加と削除

サイズ制約条件のフィルターを追加または削除できます。フィルターを変更するには、新しいフィルターを追加して古いフィルターを削除します。

サイズ制約条件のフィルターを追加または削除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで、[Size constraint] (サイズ制約) を選択します。
3. フィルターを追加または削除する対象の条件を選択します。
4. フィルターを追加するには、次のステップを実行します。

   a. [Add filter] (フィルターを追加) を選択します。

   b. 適用するフィルター設定を指定します。詳細については、「サイズ制約条件の作成時または編集時に指定する値」を参照してください。

   c. [Add] (追加) を選択します。
5. フィルターを削除するには、次のステップを実行します。

   a. 削除するフィルターを選択します。

   b. [Delete filter] (フィルターを削除) を選択します。

サイズ制約条件の削除

サイズ制約条件を削除するには、最初にその条件からすべてのフィルターを削除し、その条件を使用しているすべてのルールから条件自体を削除します。次に手順を示します。

サイズ制約条件を削除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF にあるコンソールを開きます。

ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。

2. ナビゲーションペインで、[Size constraints] (サイズ制約) を選択します。
3. [Size constraint conditions] (サイズ制約の条件) ペインで、削除するサイズ制約の条件を選択します。
4. 右ペインで、[Associated rules] (関連付けられたルール) タブを選択します。

   このサイズ制約条件を使用しているルールのリストが空の場合は、ステップ 6 に進みます。リストにルールが含まれている場合は、ルールを書き留めて、ステップ 5 に進みます。
5. サイズ制約条件を使用しているルールから、この条件を削除するには、次のステップを実行します。

   a. ナビゲーションペインで [Rules] (ルール) を選択します。
   b. 削除するサイズ制約条件を使用しているルールの名前を選択します。
   c. 右ペインで、ルールから削除するサイズ制約条件を選択し、[Remove selected condition] (選択した条件を削除) を選択します。
   d. 削除するサイズ制約条件を使用しているすべての残りのルールに対してステップ b とステップ c を繰り返します。
   e. ナビゲーションペインで、[Size constraint] (サイズ制約) を選択します。
   f. [Size constraint conditions] (サイズ制約の条件) ペインで、削除するサイズ制約の条件を選択します。
6. [Delete] (削除) を選択して、選択した条件を削除します。

SQL インジェクション一致条件の使用

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

攻撃者は、データベースからデータを取り出そうとしてウェブリクエスト内に悪意のある SQL コードを挿入する場合があります。悪意のある SQL コードが含まれている可能性があるウェブリクエストを許可またはブロックするには、SQL インジェクション一致条件を作成します。SQL インジェクション一致条件は、URI パスやクエリ文字列など、 AWS WAF Classic に検査させたいウェブリクエストの部分を特定します。後でウェブ ACL を作成するときに、悪意のある SQL コードが含まれている可能性があるリクエストを許可するかブロックするかを指定します。

トピック

SQL インジェクション一致条件の作成

SQL インジェクション一致条件を作成するときは、URI やクエリ文字列など、悪意のある SQL コードがないかどうかを AWS WAF Classic に検査させたいウェブリクエストの部分を示すフィルタを指定します。SQL インジェクション一致条件には複数のフィルターを追加できます。条件ごとに 1 つのフィルターを設定することもできます。各構成が AWS WAF Classic の動作にどのように影響するかを次に示します。

- SQL インジェクション一致条件ごとに複数のフィルター (推奨) — 複数のフィルターを含む SQL インジェクション一致条件をルールに追加し、そのルールをウェブ ACL に追加する場合、ウェブリクエストは SQL インジェクション一致条件のフィルターの 1 つに一致するだけで、 AWS WAF クラシックはその条件に基づいてリクエストを許可または拒否できます。

  例えば、SQL インジェクション一致条件を 1 つ作成し、この条件に 2 つのフィルターを含めたとします。1 つのフィルターは URI に悪意のある SQL コードを検査するよう AWS WAF Classic に指示し、もう 1 つのフィルターはクエリ文字列を検査するよう AWS WAF Classic に指示します。AWS WAF Classic は、URI またはクエリ文字列のいずれかに悪意のある SQL コードが含まれていると思われるリクエストを許可またはブロックします。
- SQL インジェクション一致条件ごとに 1 つのフィルタ — 個別の SQL インジェクション一致条件をルールに追加し、そのルールをウェブ ACL に追加する場合、ウェブリクエストはすべての条件に一致する必要があります。 AWS WAF Classic では、条件に基づいてリクエストを許可または拒否できます。

2 つの条件を作成し、各条件に前の例で示した 2 つのフィルターの 1 つを別個に含めたとします。両方の条件を同じルールに追加し、そのルールをウェブ ACL に追加すると、 AWS WAF Classic は URI とクエリ文字列の両方に悪意のある SQL コードが含まれていると思われる場合にのみリクエストを許可または拒否します。

> **Note**
>
> SQL インジェクション一致条件をルールに追加すると、悪意のある SQL コードが含まれていないと思われるウェブリクエストを許可またはブロックするように AWS WAF Classic を構成することもできます。

ステップ 5: SQL インジェクション一致条件を作成するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで、[SQL injection] (SQL インジェクション) を選択します。
3. [Create condition] (条件を作成) を選択します。
4. 適用するフィルター設定を指定します。詳細については、「SQL インジェクション一致条件の作成時または編集時に指定する値」を参照してください。
5. [Add another filter] (別のフィルターを追加) を選択します。
6. 別のフィルターを追加する場合は、ステップ 4～5 を繰り返します。
7. フィルターの追加が終了したら、[Create] (作成) を選択します。

SQL インジェクション一致条件の作成時または編集時に指定する値

SQL インジェクション一致条件を作成または更新するときに、次の値を指定します。

[Name] (名前)

SQL インジェクション一致条件の名前。

名前に使用できるのは英数字 (A～Z、a～z、0～9) または特殊文字 _-!"#`+*},./ です。一度作成した条件の名前は変更できません。

[Part of the request to filter on] (フィルタリングするリクエストの一部)

各ウェブリクエストの中で、 AWS WAF Classic で悪意のある SQL コードを検査させたい部分を選択します。

[Header] (ヘッダー)

指定したリクエストヘッダー (`User-Agent` や `Referer` など)。[Header] (ヘッダー) を選択した場合は、[Header] (ヘッダー) フィールドにヘッダー名を指定します。

[HTTP method] (HTTP メソッド)

リクエストがオリジンに実行を要求しているオペレーションのタイプを示す HTTP メソッド。CloudFront `DELETE`、、、、、`GETHEADOPTIONSPATCHPOST`、`PUT`およびの各メソッドをサポートします。

[Query string] (クエリ文字列)

URL 内で ? 文字の後に続く部分 (ある場合)。

> **Note**
>
> SQL インジェクションの一致条件については、[Part of the request to filter on] (フィルタリングするリクエストの一部) に [Query string] (クエリ文字列) ではなく、[All query parameters (values only)] (すべてのクエリパラメータ (値のみ)) を選択することをお勧めします。

[URI]

リクエストの URI パス。リソースを識別します (例: `/images/daily-ad.jpg`)。これには、URI のクエリ文字列またはフラグメントコンポーネントは含まれません。詳細については、「Uniform Resource Identifier (URI): 一般的な構文」を参照してください。

Transformation が指定されていない限り、URI は正規化されず、 AWS リクエストの一部としてクライアントから受け取った時点で検査されます。[Transformation] (変換) が指定されている場合、URI はその指定に従って形式が再設定されます。

[Body] (本文)

リクエスト内で、HTTP リクエストの本文としてウェブサーバーに送信する追加データ (フォームのデータなど) を含む部分。

> **Note**
>
> [Part of the request to filter on] (フィルタリングするリクエストの一部) の値として [Body] (本文) を選択した場合、 AWS WAF Classic によって最初の 8,192 バイト (8 KB) のみが検査されます。本文が 8,192 バイトより長いリクエストを許可またはブロックするには、サイズ制約条件を作成します。(AWS WAF Classic はリクエストヘッダーから本文の長さを取得します)。詳細については、「サイズ制約条件の使用」を参照してください。

[Single query parameter (value only)] (単一クエリパラメータ (値のみ))

クエリ文字列の一部として定義されているすべてのパラメータです。たとえば、URL が「www.xyz.com? UserName =abc& SalesRegion =seattle」の場合、またはパラメータにフィルタを追加できます。UserNameSalesRegion

[Single query parameter (value only)] (単一クエリパラメータ (値のみ)) を選択する場合は、[Query parameter name] (クエリパラメータ名) も指定します。これは、やなど、調べるクエリ文字列内のパラメータです。UserNameSalesRegion[Query parameter name] (クエリパラメータ名) の最大長は 30 文字です。[Query parameter name] (クエリパラメータ名) では、大文字と小文字は区別されません。たとえば、Query UserNameパラメータ名として指定すると、username や UserName などのすべてのバリエーションにマッチします。UserName

[All query parameters (values only)] (すべてのクエリパラメータ (値のみ))

単一クエリパラメーター (値のみ) と似ていますが、 AWS WAF Classic は 1 つのパラメーターの値を検査するのではなく、クエリ文字列内のすべてのパラメーターの値を検査して、悪意のある SQL コードがないかを調べます。たとえば、URL が「www.xyz.com? UserName =abc& SalesRegion =seattle」で、「すべてのクエリパラメーター (値のみ)」を選択した場合、 AWS WAF Classic は、いずれかの値が一致するか、悪意のある SQL コードが含まれている可能性がある場合に一致をトリガーします。UserNameSalesRegion

[Header] (ヘッダー)

フィルターするリクエストの一部に [ヘッダー] を選択した場合は、一般的なヘッダーのリストからヘッダーを選択するか、 AWS WAF Classic に悪意のある SQL コードがないか検査させたいヘッダーの名前を入力します。

[Transformation] (変換)

AWS WAF Classic がリクエストを検査する前に、変換によってウェブリクエストが再フォーマットされます。これにより、攻撃者が Classic をバイパスしようとしてウェブリクエストで使用する、通常とは異なるフォーマットの一部が排除されます。 AWS WAF

1 種類のテキスト変換しか指定できません。

変換では次の操作を実行できます。

なし

AWS WAF Classic は、Value の文字列が一致するかどうかを調べるまで、ウェブリクエストのテキスト変換を一切行いません。

[Convert to lowercase] (小文字に変換)

AWS WAF Classic は大文字 (A ~ Z) を小文字 (a ~ z) に変換します。

[HTML decode] (HTML デコード)

AWS WAF Classic は HTML でエンコードされた文字をエンコードされていない文字に置き換えます。

- `&quot;` を & に置き換えます。
- ` ` を改行なしスペースに置き換えます。
- `<` を < に置き換えます。
- `>` を > に置き換えます。
- 16 進数形式の文字 (`&#xhhhh;`) を対応する文字に置き換えます。
- 10 進数形式の文字 (`&#nnnn;`) を対応する文字に置き換えます。

[Normalize white space] (空白の正規化)

AWS WAF Classic では、次の文字がスペース文字 (10 進数 32) に置き換えられます。

- \f、フォームフィード、10 進数 12
- \t、タブ、10 進数 9
- \n、改行、10 進数 10

- \r、キャリッジリターン、10 進数 13
- \v、垂直タブ、10 進数 11
- 改行なしスペース、10 進数 160

さらに、このオプションでは複数のスペースを 1 つのスペースに置き換えます。

[Simplify command line] (コマンドラインを簡素化)

オペレーティングシステムのコマンドラインのコマンドが含まれているリクエストの場合、このオプションを使用して次の変換を行います。

- 次の文字を削除します: \ " ' ^
- 次の文字の前にあるスペースを削除します: / (
- 次の文字をスペースに置き換えます: , ;
- 複数のスペースを 1 つのスペースに置き換えます。
- 大文字 (A〜Z) を小文字 (a〜z) に変換します。

[URL decode] (URL デコード)

URL エンコードされたリクエストをデコードします。

SQL インジェクション一致条件のフィルターの追加と削除

SQL インジェクション一致条件のフィルターを追加または削除できます。フィルターを変更するには、新しいフィルターを追加して古いフィルターを削除します。

SQL インジェクション一致条件のフィルターを追加または削除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで、[SQL injection] (SQL インジェクション) を選択します。
3. フィルターを追加または削除する対象の条件を選択します。
4. フィルターを追加するには、次のステップを実行します。

   a. [Add filter] (フィルターを追加) を選択します。

   b. 適用するフィルター設定を指定します。詳細については、「SQL インジェクション一致条件の作成時または編集時に指定する値」を参照してください。

c. [Add] (追加) を選択します。

5. フィルターを削除するには、次のステップを実行します。

   a. 削除するフィルターを選択します。

   b. [Delete filter] (フィルターを削除) を選択します。

SQL インジェクション一致条件の削除

SQL インジェクション一致条件を削除するには、最初にその条件からすべてのフィルターを削除し、その条件を使用しているすべてのルールから条件自体を削除します。次に手順を示します。

SQL インジェクション一致条件を削除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF にあるコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。

2. ナビゲーションペインで、[SQL injection] (SQL インジェクション) を選択します。

3. [SQL injection match conditions] (SQL インジェクション一致条件) ペインで、削除する SQL インジェクション一致条件を選択します。

4. 右ペインで、[Associated rules] (関連付けられたルール) タブを選択します。

   この SQL インジェクション一致条件を使用しているルールのリストが空の場合は、ステップ 6 に進みます。リストにルールが含まれている場合は、ルールを書き留めて、ステップ 5 に進みます。

5. SQL インジェクション一致条件を使用しているルールから、この条件を削除するには、次のステップを実行します。

   a. ナビゲーションペインで [Rules] (ルール) を選択します。

   b. 削除する SQL インジェクション一致条件を使用しているルールの名前を選択します。

   c. 右ペインで、ルールから削除する SQL インジェクション一致条件を選択し、[Remove selected condition] (選択した条件を削除) を選択します。

   d. 削除する SQL インジェクション一致条件を使用しているすべての残りのルールに対してステップ b とステップ c を繰り返します。

   e. ナビゲーションペインで、[SQL injection] (SQL インジェクション) を選択します。

f. [SQL injection match conditions] (SQL インジェクション一致条件) ペインで、削除する SQL インジェクション一致条件を選択します。

6. [Delete] (削除) を選択して、選択した条件を削除します。

文字列一致条件の使用

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

ウェブリクエストに表示される文字列に基づいてリクエストを許可またはブロックする場合は、文字列一致条件を作成します。文字列一致条件は、検索する文字列と、 AWS WAF Classic にその文字列を検査させたいウェブリクエストの一部 (指定したヘッダーやクエリ文字列など) を識別します。後でウェブ ACL を作成するときに、文字列を含むリクエストを許可するかブロックするかを指定します。

トピック

- 文字列一致条件の作成
- 文字列一致条件の作成時または編集時に指定する値
- 文字列一致条件のフィルターの追加と削除
- 文字列一致条件の削除

文字列一致条件の作成

文字列一致条件を作成するときは、検索する文字列と、 AWS WAF Classic にその文字列を検査させたいウェブリクエストの部分 (URI やクエリ文字列など) を識別するフィルターを指定します。文字列一致条件には複数のフィルターを追加できます。文字列一致条件ごとに 1 つのフィルターを設定することもできます。各設定が AWS WAF Classic の動作にどのように影響するかを次に示します。

- 文字列一致条件ごとに 1 つのフィルター — 個別の文字列一致条件をルールに追加し、そのルールをウェブ ACL に追加する場合、ウェブリクエストはすべての条件に一致する必要があります。AWS WAF Classic では、条件に基づいてリクエストを許可または拒否します。

  例えば、2 つの条件を作成するとします。1 つの条件は、`User-Agent` ヘッダーに値 `BadBot` が含まれているウェブリクエストに一致します。もう 1 つの条件は、クエリ文字列に値 `BadParameter` が含まれているウェブリクエストに一致します。両方の条件を同じルールに追加し、そのルールをウェブ ACL に追加すると、 AWS WAF Classic は両方の値を含むリクエストのみを許可または拒否します。
- 文字列一致条件ごとに複数のフィルター — 複数のフィルターを含む文字列一致条件をルールに追加し、そのルールをウェブ ACL に追加すると、ウェブリクエストは文字列一致条件のフィルターの 1 つに一致するだけで、 AWS WAF Classic では 1 つの条件に基づいてリクエストを許可または拒否できます。

  2 つの条件ではなく 1 つの条件を作成し、その 1 つの条件に前の例と同じ 2 つのフィルターが含まれているとします。 AWS WAF Classic では、*`BadBot`*`User-Agent`BadParameterヘッダーまたはクエリ文字列のいずれかに含まれているリクエストを許可または拒否します。

> **Note**
>
> 文字列一致条件をルールに追加すると、条件の値と一致しないウェブリクエストを許可または拒否するように AWS WAF Classic を構成することもできます。

文字列一致条件を作成するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで、[String and regex matching] (文字列および正規表現の一致) を選択します。
3. [Create condition] (条件を作成) を選択します。
4. 適用するフィルター設定を指定します。詳細については、「文字列一致条件の作成時または編集時に指定する値」を参照してください。
5. [Add filter] (フィルターを追加) を選択します。

6. 別のフィルターを追加する場合は、ステップ 4〜5 を繰り返します。
7. フィルターの追加が終了したら、[Create] (作成) を選択します。

文字列一致条件の作成時または編集時に指定する値

文字列一致条件を作成または更新するときに、次の値を指定します。

[Name] (名前)

文字列一致条件の名前を入力します。名前に使用できるのは英数字 (A ~ Z、a ~ z、0 ~ 9) または特殊文字 _-!"#`+*},./ です。一度作成した条件の名前は変更できません。

[Type] (タイプ)

[String match] (文字列の一致) を選択します。

[Part of the request to filter on] (フィルタリングするリクエストの一部)

各ウェブリクエストの中で、[Value to match] で指定した文字列が一致するかを AWS WAF Classic に検査させたい部分を選択します。

[Header] (ヘッダー)

指定したリクエストヘッダー (`User-Agent` や `Referer` など)。[Header] (ヘッダー) を選択した場合は、[Header] (ヘッダー) フィールドにヘッダー名を指定します。

[HTTP method] (HTTP メソッド)

リクエストがオリジンに実行を要求しているオペレーションのタイプを示す HTTP メソッド。CloudFront はDELETE、、、、、GETHEADOPTIONSPATCHPOST、PUTおよびの各メソッドをサポートします。

[Query string] (クエリ文字列)

URL 内で ? 文字の後に続く部分 (ある場合)。

[URI]

リクエストの URI パス。リソースを識別します (例: `/images/daily-ad.jpg`)。これには、URI のクエリ文字列またはフラグメントコンポーネントは含まれません。詳細については、「Uniform Resource Identifier (URI): 一般的な構文」を参照してください。

Transformation が指定されていない限り、URI は正規化されず、 AWS リクエストの一部としてクライアントから受け取った時点で検査されます。[Transformation] (変換) が指定されている場合、URI はその指定に従って形式が再設定されます。

[Body] (本文)

リクエスト内で、HTTP リクエストの本文としてウェブサーバーに送信する追加データ (フォームのデータなど) を含む部分。

> **Note**
>
> [Part of the request to filter on] (フィルタリングするリクエストの一部) の値として [Body] (本文) を選択した場合、 AWS WAF Classic によって最初の 8,192 バイト (8 KB) のみが検査されます。本文が 8,192 バイトより長いリクエストを許可またはブロックするには、サイズ制約条件を作成します。(AWS WAF Classic はリクエストヘッダーから本文の長さを取得します)。詳細については、「サイズ制約条件の使用」を参照してください。

[Single query parameter (value only)] (単一クエリパラメータ (値のみ))

クエリ文字列の一部として定義されているすべてのパラメータです。たとえば、URL が「www.xyz.com? UserName =abc& SalesRegion =seattle」の場合、またはパラメータにフィルタを追加できます。UserNameSalesRegion

重複したパラメータがクエリ文字列に記述される場合、値は「OR」として評価されます。つまり、いずれかの値によって一致がトリガーされます。たとえば、URL「www.xyz.com? SalesRegion =boston& SalesRegion =シアトル」では、「一致する値」が「ボストン」または「シアトル」のどちらかによって一致がトリガーされます。

[Single query parameter (value only)] (単一クエリパラメータ (値のみ)) を選択する場合は、[Query parameter name] (クエリパラメータ名) も指定します。これは、やなど、調べるクエリ文字列内のパラメータです。UserNameSalesRegion[Query parameter name] (クエリパラメータ名) の最大長は 30 文字です。[Query parameter name] (クエリパラメータ名) では、大文字と小文字は区別されません。たとえば、Query UserNameパラメータ名として指定すると、username や UserName などのすべてのバリエーションにマッチします。UserName

[All query parameters (values only)] (すべてのクエリパラメータ (値のみ))

Single クエリパラメーター (値のみ) と似ていますが、 AWS WAF Classic は 1 つのパラメーターの値を調べるのではなく、クエリ文字列内のすべてのパラメーターの値を検査して、一致する Value がないかを調べます。たとえば、URL が「www.xyz.com? UserName =abc& SalesRegion =theatle」で、「すべてのクエリパラメーター (値のみ)」を選択した場合、

AWS WAF Classic は、「一致する値」UserNameとしてまたはのいずれかの値が指定されていれば、一致をトリガーします。SalesRegion

[Header] (ヘッダー) ([Part of the request to filter on] (フィルタリングするリクエストの一部) が [Header] (ヘッダー) の場合のみ)

リクエストの [部分] から [ヘッダー] を選択して一覧で絞り込んだ場合は、一般的なヘッダーのリストからヘッダーを選択するか、Classic に検査させたいヘッダーの名前を入力します。 AWS WAF

[Match type] (一致タイプ)

AWS WAF Classic に検査させたいリクエストの部分で、「Value to match」の文字列がこのフィルターと一致するように表示する必要がある場所を選択します。

[Contains] (次を含む)

文字列は、指定したリクエスト部分内の任意の場所に表示されます。

[Contains word] (単語を含む)

指定したウェブリクエスト部分には [Value to match] (一致する値) が含まれていること、[Value to match] (一致する値) には英数字または下線 (A〜Z、a〜z、0〜9、_) のみが含まれていることが必要です。さらに、[Value to match] (一致する値) は単語であること、つまり、次のいずれかであることが必要です。

- [Value to match] (一致する値) は、指定したウェブリクエスト部分 (ヘッダーの値など) の値と正確に一致する。
- [Value to match] (一致する値) は、指定したウェブリクエスト部分の先頭にあり、英数字または下線 (_) 以外の文字が続く (例: `BadBot;`)。
- [Value to match] (一致する値) は、指定したウェブリクエスト部分の末尾にあり、英数字または下線 (_) 以外の文字が続く (例: `;BadBot`)。
- [Value to match] (一致する値) は、指定したウェブリクエスト部分の中央にあり、英数字または下線 (_) 以外の文字が前後にある (例: `-BadBot;`)。

[Exactly matches] (完全に一致)

文字列と指定したリクエスト部分の値が正確に一致します。

[Starts with] (次で開始)

文字列は指定したリクエスト部分の先頭にあります。

[Ends with] (次で終了)

文字列は指定したリクエスト部分の末尾にあります。

[Transformation] (変換)

AWS WAF Classic がリクエストを検査する前に、変換によってウェブリクエストが再フォーマットされます。これにより、攻撃者が Classic をバイパスしようとしてウェブリクエストで使用する、通常とは異なるフォーマットの一部が排除されます。 AWS WAF

1 種類のテキスト変換しか指定できません。

変換では次の操作を実行できます。

なし

AWS WAF Classic は、Value の文字列が一致するかどうかを調べるまで、ウェブリクエストのテキスト変換を一切行いません。

[Convert to lowercase] (小文字に変換)

AWS WAF Classic は大文字 (A ~ Z) を小文字 (a ~ z) に変換します。

[HTML decode] (HTML デコード)

AWS WAF Classic は HTML でエンコードされた文字をエンコードされていない文字に置き換えます。

- " を & に置き換えます。
-   を改行なしスペースに置き換えます。
- < を < に置き換えます。
- > を > に置き換えます。
- 16 進数形式の文字 (&#xhhhh;) を対応する文字に置き換えます。
- 10 進数形式の文字 (&#nnnn;) を対応する文字に置き換えます。

[Normalize white space] (空白の正規化)

AWS WAF Classic では、次の文字がスペース文字 (10 進数 32) に置き換えられます。

- \f、フォームフィード、10 進数 12
- \t、タブ、10 進数 9
- \n、改行、10 進数 10
- \r、キャリッジリターン、10 進数 13
- \v、垂直タブ、10 進数 11
- 改行なしスペース、10 進数 160

さらに、このオプションでは複数のスペースを 1 つのスペースに置き換えます。

[Simplify command line] (コマンドラインを簡素化)

攻撃者がオペレーティングシステムのコマンドラインのコマンドを挿入し、通常と異なるフォーマットを使用してコマンドの一部または全部を偽装するおそれがある場合は、このオプションを使用して次の変換を行います。

- 次の文字を削除します: \ " ' ^
- 次の文字の前にあるスペースを削除します: / (
- 次の文字をスペースに置き換えます: , ;
- 複数のスペースを 1 つのスペースに置き換えます。
- 大文字 (A〜Z) を小文字 (a〜z) に変換します。

[URL decode] (URL デコード)

URL エンコードされたリクエストをデコードします。

[Value is base64 encoded] (値が base64 エンコードされている)

[Value to match] (一致する値) の値が base64 でエンコードされている場合は、このチェックボックスをオンにします。Base64 エンコーディングを使用して、攻撃者がリクエストに含める表示不可能な文字 (タブや改行など) を指定します。

[Value to match] (照合する値)

AWS WAF Classic にウェブリクエストで検索させたい値を指定します。最大長は 50 バイトです値を Base 64 でエンコードする場合は、エンコードする前の値に 50 バイトの最大長が適用されます。

文字列一致条件のフィルターの追加と削除

文字列一致条件のフィルターを追加または削除できます。フィルターを変更するには、新しいフィルターを追加して古いフィルターを削除します。

文字列一致条件のフィルターを追加または削除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。

2. ナビゲーションペインで、[String and regex matching] (文字列および正規表現の一致) を選択します。
3. フィルターを追加または削除する対象の条件を選択します。
4. フィルターを追加するには、次のステップを実行します。
   a. [Add filter] (フィルターを追加) を選択します。
   b. 適用するフィルター設定を指定します。詳細については、「文字列一致条件の作成時または編集時に指定する値」を参照してください。
   c. [Add] (追加) を選択します。
5. フィルターを削除するには、次のステップを実行します。
   a. 削除するフィルターを選択します。
   b. [Delete filter] (フィルターを削除) を選択します。

文字列一致条件の削除

文字列一致条件を削除するには、最初にその条件からすべてのフィルターを削除し、その条件を使用しているすべてのルールから条件自体を削除します。次に手順を示します。

文字列一致条件を削除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF にあるコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. 文字列一致条件を使用しているルールからその条件を削除します。
   a. ナビゲーションペインで [Rules] (ルール) を選択します。
   b. 削除する文字列一致条件を使用しているルールの名前を選択します。
   c. 右側のペインで、[Edit rule] (ルールを編集) を選択します。
   d. 削除する条件の横にある [X] を選択します。
   e. [Update] (更新) を選択します。
   f. 削除する文字列一致条件を使用している残りのすべてのルールに対してこの同じ手順を繰り返します。
3. 削除する条件からフィルターを削除します。

a. ナビゲーションペインで、[String and regex matching] (文字列および正規表現の一致) を選択します。

b. 削除する文字列一致条件の名前を選択します。

c. 右側のペインで、[Filter] (フィルター) の横にあるチェックボックスをオンにして、すべてのフィルターを選択します。

d. [Delete filter] (フィルターを削除) を選択します。

4. ナビゲーションペインで、[String and regex matching] (文字列および正規表現の一致) を選択します。

5. [String and regex match conditions] (文字列および正規表現の一致条件) ペインで、削除する文字列一致条件を選択します。

6. [Delete] (削除) を選択して、選択した条件を削除します。

正規表現一致条件の使用

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

リクエストに含まれる正規表現パターンと一致する文字列に基づいてウェブリクエストを許可またはブロックする場合は、1 つ以上の正規表現一致条件を作成します。正規表現一致条件は、検索するパターンと、 AWS WAF Classic にパターンを検査させたいウェブリクエストの部分 (指定したヘッダーやクエリ文字列など) を識別する文字列一致条件の一種です。後でウェブ ACL を作成するときに、そのパターンを含むリクエストを許可するかブロックするかを指定します。

トピック

- 正規表現一致条件の作成
- RegEx 一致条件を作成または編集するときに指定する値
- 正規表現一致条件の編集

正規表現一致条件の作成

正規表現一致条件を作成するときは、検索する文字列 (正規表現を使用) を識別するパターンセットを指定します。次に、それらのパターンセットを、URI やクエリ文字列など、 AWS WAF Classic にそのパターンセットを検査させたいウェブリクエストの部分を指定するフィルターに追加します。

1 つのパターンセットに複数の正規表現を追加できます。その場合、それらの表現は OR を使用して結合します。つまり、リクエストの該当する部分が、指定した式のいずれかと一致する場合、ウェブリクエストはパターンセットと一致することになります。

正規表現一致条件をルールに追加すると、条件の値と一致しないウェブリクエストを許可または拒否するように AWS WAF Classic を設定することもできます。

AWS WAF Classic は、ほとんどの標準の Perl 互換正規表現 (PCRE) をサポートしています。ただし、次はサポートしていません。

- 後方参照と部分式取得
- 任意のゼロ幅アサーション
- サブルーチン参照と再帰パターン
- 条件付きパターン
- バックトラック制御動詞
- \C シングルバイトディレクティブ
- \R 改行一致ディレクティブ
- \K 一致開始位置リセットディレクティブ
- コールアウトと埋め込みコード
- アトミックグループと所有格量指定子

正規表現一致条件を作成するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで、[String and regex matching] (文字列および正規表現の一致) を選択します。
3. [Create condition] (条件を作成) を選択します。

4. 適用するフィルター設定を指定します。詳細については、「RegEx 一致条件を作成または編集するときに指定する値」を参照してください。
5. [Create pattern set and add filter] (パターンセットを作成してフィルターを追加) を選択するか (新しいパターンセットを作成した場合)、[Add filter] (フィルターを追加) を選択します (既存のパターンセットを使用した場合)。
6. [Create] (作成) を選択します。

RegEx 一致条件を作成または編集するときに指定する値

正規表現一致条件を作成または更新するときに、次の値を指定します。

[Name] (名前)

正規表現一致条件の名前を入力します。名前に使用できるのは英数字 (A ~ Z、a ~ z、0 ~ 9) または特殊文字 _-!"#`+*},./ です。一度作成した条件の名前は変更できません。

[Type] (タイプ)

[Regex match] (正規表現の一致) を選択します。

[Part of the request to filter on] (フィルタリングするリクエストの一部)

[Value to match] で指定したパターンを AWS WAF Classic に検査させたい各ウェブリクエストの部分を選択します。

[Header] (ヘッダー)

指定したリクエストヘッダー (`User-Agent` や `Referer` など)。[Header] (ヘッダー) を選択した場合は、[Header] (ヘッダー) フィールドにヘッダー名を指定します。

[HTTP method] (HTTP メソッド)

リクエストがオリジンに実行を要求しているオペレーションのタイプを示す HTTP メソッド。CloudFront `DELETE`、、、、、`GET``HEAD``OPTIONS``PATCH``POST`、`PUT`およびの各メソッドをサポートします。

[Query string] (クエリ文字列)

URL 内で ? 文字の後に続く部分 (ある場合)。

[URI]

リクエストの URI パス。リソースを識別します (例: `/images/daily-ad.jpg`)。これには、URI のクエリ文字列またはフラグメントコンポーネントは含まれません。詳細については、「Uniform Resource Identifier (URI): 一般的な構文」を参照してください。

Transformation が指定されていない限り、URI は正規化されず、 AWS リクエストの一部としてクライアントから受け取った時点で検査されます。[Transformation] (変換) が指定されている場合、URI はその指定に従って形式が再設定されます。

[Body] (本文)

リクエスト内で、HTTP リクエストの本文としてウェブサーバーに送信する追加データ (フォームのデータなど) を含む部分。

> **Note**
>
> [Part of the request to filter on] (フィルタリングするリクエストの一部) の値として [Body] (本文) を選択した場合、 AWS WAF Classic によって最初の 8,192 バイト (8 KB) のみが検査されます。本文が 8,192 バイトより長いリクエストを許可またはブロックするには、サイズ制約条件を作成します。(AWS WAF Classic はリクエストヘッダーから本文の長さを取得します)。詳細については、「サイズ制約条件の使用」を参照してください。

[Single query parameter (value only)] (単一クエリパラメータ (値のみ))

クエリ文字列の一部として定義されているすべてのパラメータです。たとえば、URL が「www.xyz.com? UserName =abc& SalesRegion =seattle」の場合、またはパラメータにフィルタを追加できます。UserNameSalesRegion

重複したパラメータがクエリ文字列に記述される場合、値は「OR」として評価されます。つまり、いずれかの値によって一致がトリガーされます。たとえば、URL「www.xyz.com? SalesRegion =boston& SalesRegion =シアトル」では、「照合する値」が「ボストン」または「シアトル」のどちらかに一致するパターンが一致すると一致がトリガーされます。

[Single query parameter (value only)] (単一クエリパラメータ (値のみ)) を選択する場合は、[Query parameter name] (クエリパラメータ名) も指定します。これは、またはなど、調べるクエリ文字列内のパラメータです。UserNameSalesRegion[Query parameter name] (クエリパラメータ名) の最大長は 30 文字です。[Query parameter name] (クエリパラメータ名) では、大文字と小文字は区別されません。たとえば、Query UserNameパラメータ名として指定すると、username や UserName などのすべてのバリエーションにマッチします。UserName

[All query parameters (values only)] (すべてのクエリパラメータ (値のみ))

単一クエリパラメーター (値のみ) と似ていますが、 AWS WAF Classic は 1 つのパラメーターの値を調べるのではなく、クエリ文字列内のすべてのパラメーターの値を、Value to

match に指定されたパターンがないか調べます。たとえば、URL「www.xyz.com? UserName =abc& SalesRegion =seattle」の「Value to Match」のパターンは、内の値と一致するか、一致をトリガーします。UserNameSalesRegion

[Header] (ヘッダー) ([Part of the request to filter on] (フィルタリングするリクエストの一部) が [Header] (ヘッダー) の場合のみ)

リストでフィルタリングするリクエストの一部から Header を選択した場合は、一般的なヘッダーのリストからヘッダーを選択するか、Classic に検査させたいヘッダーの名前を入力します。 AWS WAF

[Transformation] (変換)

AWS WAF Classic がリクエストを検査する前に、変換によってウェブリクエストが再フォーマットされます。これにより、攻撃者が Classic をバイパスしようとしてウェブリクエストで使用する、通常とは異なるフォーマットの一部が排除されます。 AWS WAF

1 種類のテキスト変換しか指定できません。

変換では次の操作を実行できます。

なし

AWS WAF Classic は、Value の文字列が一致するかどうかを調べるまで、ウェブリクエストのテキスト変換を一切行いません。

[Convert to lowercase] (小文字に変換)

AWS WAF Classic は大文字 (A ~ Z) を小文字 (a ~ z) に変換します。

[HTML decode] (HTML デコード)

AWS WAF Classic は HTML でエンコードされた文字をエンコードされていない文字に置き換えます。

- `&quot;` を & に置き換えます。
- ` ` を改行なしスペースに置き換えます。
- `<` を < に置き換えます。
- `>` を > に置き換えます。
- 16 進数形式の文字 (`&#xhhhh;`) を対応する文字に置き換えます。
- 10 進数形式の文字 (`&#nnnn;`) を対応する文字に置き換えます。

[Normalize white space] (空白の正規化)

AWS WAF Classic では、次の文字がスペース文字 (10 進数 32) に置き換えられます。

- \f、フォームフィード、10 進数 12
- \t、タブ、10 進数 9
- \n、改行、10 進数 10
- \r、キャリッジリターン、10 進数 13
- \v、垂直タブ、10 進数 11
- 改行なしスペース、10 進数 160

さらに、このオプションでは複数のスペースを 1 つのスペースに置き換えます。

[Simplify command line] (コマンドラインを簡素化)

攻撃者がオペレーティングシステムのコマンドラインのコマンドを挿入し、通常と異なるフォーマットを使用してコマンドの一部または全部を偽装するおそれがある場合は、このオプションを使用して次の変換を行います。

- 次の文字を削除します: \ " ' ^
- 次の文字の前にあるスペースを削除します: / (
- 次の文字をスペースに置き換えます: , ;
- 複数のスペースを 1 つのスペースに置き換えます。
- 大文字 (A〜Z) を小文字 (a〜z) に変換します。

[URL decode] (URL デコード)

URL エンコードされたリクエストをデコードします。

[Regex pattern to match to request] (リクエストに一致する正規表現パターン)

既存のパターンセットを選択するか、新しいパターンセットを作成できます。新しいパターンセットを作成する場合は、次のように指定します。

New pattern set name (新しいパターンセット名)

名前を入力し、 AWS WAF Classic で検索させたい正規表現パターンを指定します。

パターンセットに複数の正規表現を追加すると、それらの表現は OR を使用して結合されます。つまり、リクエストの該当する部分が、指定した式のいずれかと一致する場合、ウェブリクエストはパターンセットと一致することになります。

[Value to match] (一致する値) は最大 70 文字です。

正規表現一致条件の編集

既存の正規表現一致条件に次の変更を加えることができます。

- 既存のパターンセットからパターンを削除する
- 既存のパターンセットにパターンを追加する
- 既存正規表現一致条件からフィルターを削除する
- 既存の正規表現一致条件にフィルターを追加する (正規表現一致条件には 1 つのフィルターのみを含めることができます。したがって、フィルターを追加するには、まず既存のフィルターを削除する必要があります。)
- 既存の正規表現一致条件を削除する

> **Note**
>
> 既存のフィルターに対してパターンセットを追加したり削除したりはできません。パターンセットを編集するか、または、フィルターを削除して新しいパターンセットで新しいフィルターを作成する必要があります。

既存のパターンセットからパターンを削除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで、[String and regex matching] (文字列および正規表現の一致) を選択します。
3. [View regex pattern sets] (正規表現パターンセットを表示) を選択します。
4. 編集するパターンセットの名前を選択します。
5. [Edit] (編集) を選択します。
6. 削除するパターンの横にある [X] を選択します。
7. [Save] (保存) を選択します。

既存のパターンセットにパターンを追加するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF にあるコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで、[String and regex matching] (文字列および正規表現の一致) を選択します。
3. [View regex pattern sets] (正規表現パターンセットを表示) を選択します。
4. 編集するパターンセットの名前を選択します。
5. [Edit] (編集) を選択します。
6. 新しい正規表現パターンを入力します。
7. 新しいパターンの横にある [+] を選択します。
8. [Save] (保存) を選択します。

既存の正規表現一致条件からフィルターを削除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF にあるコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで、[String and regex matching] (文字列および正規表現の一致) を選択します。
3. 削除するフィルターを含む条件の名前を選択します。
4. 削除するフィルターの横にあるボックスを選択します。
5. [Delete filter] (フィルターを削除) を選択します。

正規表現一致条件を削除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF にあるコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。

2. 正規表現条件からフィルターを削除します。その手順については、「既存の正規表現一致条件からフィルターを削除するには」を参照してください。
3. 正規表現一致条件を使用しているルールから条件自体を削除します。
   a. ナビゲーションペインで [Rules] (ルール) を選択します。
   b. 削除する正規表現一致条件を使用しているルールの名前を選択します。
   c. 右側のペインで、[Edit rule] (ルールを編集) を選択します。
   d. 削除する条件の横にある [X] を選択します。
   e. [Update] (更新) を選択します。
   f. 削除する正規表現一致条件を使用している残りのすべてのルールに対してこの同じ手順を繰り返します。
4. ナビゲーションペインで、[String and regex matching] (文字列および正規表現の一致) を選択します。
5. 削除する条件の横にあるボタンを選択します。
6. [Delete] (削除) を選択します。

既存の正規表現一致条件に対してフィルターを追加または変更するには

正規表現一致条件には 1 つのフィルターのみ含めることができます。フィルターを追加または変更するには、まず既存のフィルターを削除する必要があります。

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF にあるコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. 変更する正規表現条件からフィルターを削除します。その手順については、「既存の正規表現一致条件からフィルターを削除するには」を参照してください。
3. ナビゲーションペインで、[String and regex matching] (文字列および正規表現の一致) を選択します。
4. 変更する条件の名前を選択します。
5. [Add filter] (フィルターを追加) を選択します。
6. 新しいフィルターに適切な値を入力し、[Add] (追加) を選択します。

ルールの使用

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

ルールを使用すると、Classic に監視させたい正確な条件を指定することで、 AWS WAF Classic AWS WAF に許可または拒否させたいウェブリクエストを正確に絞り込むことができます。たとえば、 AWS WAF Classic では、リクエストの発信元の IP アドレス、リクエストに含まれる文字列、文字列が表示される場所、リクエストに悪意のある SQL コードが含まれている可能性があるかどうかを監視できます。

トピック

- ルールの作成と条件の追加
- ルールの条件の追加と削除
- ルールの削除
- AWS Marketplace ルールグループ

ルールの作成と条件の追加

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

ルールに複数の条件を追加する場合、 AWS WAF Classic がそのルールに基づくリクエストを許可または拒否するには、ウェブリクエストがすべての条件に一致する必要があります。

ルールを作成して条件を追加するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで Rules] (ルール) を選択します。
3. [Create rule] (ルールの作成) を選択します。
4. 次の値を入力します。

   [Name] (名前)

   名前を入力します。

   CloudWatch 指標名

   AWS WAF Classic CloudWatch が作成してルールに関連付ける指標の名前を入力します。名前には英数字 (A～Z、a～z、0～9) のみを使用することができ、最大 128 文字、最小 1 文字です。空白や「All」や「Default_Action」など、 AWS WAF クラシック専用の指標名は使用できません。

   [Rule type] (ルールタイプ)

   [Regular rule] または [Rate-based rule] のいずれかを選択します。レートベースのルールは通常ルールと同じですが、5 分間に IP アドレスから着信したリクエストの数も考慮に入れます。これらのルールタイプの詳細については、「AWS WAF クラシックの仕組み」を参照してください。

   [Rate limit] (レート制限)

   レートベースのルールの場合、ルールの条件に一致する IP アドレスから 5 分間に許可するリクエストの最大数を入力します。レート制限は 100 以上にする必要があります。

   レート制限を単独で指定することも、レート制限と条件を指定することもできます。レート制限のみを指定すると、すべての IP AWS WAF アドレスに制限が適用されます。レート制限と条件を指定すると、 AWS WAF 条件に一致する IP アドレスに制限が課されます。

   IP アドレスがレート制限のしきい値に達すると、割り当てられたアクション (ブロックまたはカウント) をできるだけ早く、通常は 30 AWS WAF 秒以内に適用します。アクションが実

行され、その IP アドレスからのリクエストが 5 分経過しなかった場合、 AWS WAF カウンタはゼロにリセットされます。

5. ルールに条件を追加するには、次の値を指定します。

[When a request does/does not] (リクエストが次の条件内/条件外)

AWS WAF Classic で条件内のフィルターに基づいてリクエストを許可または拒否するようにしたい場合は、[Does] を選択します。たとえば、IP 一致条件に 192.0.2.0/24 の IP アドレス範囲が含まれ、その IP アドレスからのリクエストを AWS WAF Classic で許可または拒否したい場合は、「する」を選択します。

AWS WAF 条件に含まれる逆のフィルタに基づいてリクエストを許可または拒否するようにClassic を設定する場合は、「しない」を選択します。たとえば、IP 一致条件に 192.0.2.0/24 の IP アドレス範囲が含まれ、その IP アドレスから送信されないリクエストを AWS WAF Classic で許可または拒否したい場合は、「しない」を選択します。

[match/originate from] (一致/次から生じている)

ルールに追加する条件のタイプを選択します。

- クロスサイトスクリプティング一致条件 - [match at least one of the filters in the cross-site scripting match condition] (クロスサイトスクリプティング一致条件の少なくとも 1 つのフィルターに一致する) を選択します。
- IP 一致条件 - [originate from an IP address in] (IP アドレスより送信) を選択します
- Geo 一致条件 - [originate from a geographic location in] (地理的場所より送信) を選択します
- サイズ制約条件 - [match at least one of the filters in the size constraint condition] (サイズ制約条件の少なくとも 1 つのフィルターに一致する) を選択します
- SQL インジェクション一致条件 - [match at least one of the filters in the SQL injection match condition] (SQL インジェクション一致条件の少なくとも 1 つのフィルターに一致する) を選択します
- 文字列一致条件 - [match at least one of the filters in the string match condition] (文字列一致条件の少なくとも 1 つのフィルターに一致する) を選択します
- 正規表現一致条件 - [match at least one of the filters in the regex match condition] (正規表現一致条件の少なくとも 1 つのフィルターに一致する) を選択します

[condition name] (条件名)

ルールに追加する条件を選択します。リストには、前のステップで選択したタイプの条件のみが表示されます。

6. ルールに別の条件を追加するには、[Add another condition] (別の条件を追加) を選択して、ステップ 4〜5 を繰り返します。次の点に注意してください。

   - 複数の条件を追加した場合、 AWS WAF Classic がそのルールに基づいてリクエストを許可または拒否するには、ウェブリクエストがすべての条件の少なくとも 1 つのフィルターに一致する必要があります。
   - 同じルールに 2 つの IP 一致条件を追加すると、 AWS WAF Classic は両方の IP 一致条件に含まれる IP アドレスから送信されたリクエストのみを許可または拒否します。

7. 条件の追加が終了したら、[Create] (作成) を選択します。

ルールの条件の追加と削除

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

条件を追加または削除することでルールを変更できます。

ルールの条件を追加または削除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。

2. ナビゲーションペインで [Rules] (ルール) を選択します。
3. 条件を追加または削除するルールの名前を選択します。
4. [Add Rule] (ルールの追加) を選択します。

5. 条件を追加するには、[Add condition] (条件を追加) を選択して次の値を指定します。

   [When a request does/does not] (リクエストが次の条件内/条件外)

   たとえば、IP アドレス 192.0.2.0/24 の範囲から発信されるウェブリクエストなど、 AWS WAF 条件内のフィルターに基づいてリクエストを許可または拒否するようにClassic を設定する場合は、[Does] を選択します。

   AWS WAF Classic で条件内のフィルターの逆に基づいてリクエストを許可または拒否したい場合は、「しない」を選択します。たとえば、IP 一致条件に 192.0.2.0/24 の IP アドレス範囲が含まれ、その IP アドレスから送信されないリクエストを AWS WAF Classic で許可または拒否したい場合は、「しない」を選択します。

   [match/originate from] (一致/次から生じている)

   ルールに追加する条件のタイプを選択します。

   - クロスサイトスクリプティング一致条件 - [match at least one of the filters in the cross-site scripting match condition] (クロスサイトスクリプティング一致条件の少なくとも 1 つのフィルターに一致する) を選択します。
   - IP 一致条件 - [originate from an IP address in] (IP アドレスより送信) を選択します
   - Geo 一致条件 - [originate from a geographic location in] (地理的場所より送信) を選択します
   - サイズ制約条件 - [match at least one of the filters in the size constraint condition] (サイズ制約条件の少なくとも 1 つのフィルターに一致する) を選択します
   - SQL インジェクション一致条件 - [match at least one of the filters in the SQL injection match condition] (SQL インジェクション一致条件の少なくとも 1 つのフィルターに一致する) を選択します
   - 文字列一致条件 - [match at least one of the filters in the string match condition] (文字列一致条件の少なくとも 1 つのフィルターに一致する) を選択します
   - 正規表現一致条件 - [match at least one of the filters in the regex match condition] (正規表現一致条件の少なくとも 1 つのフィルターに一致する) を選択します

   [condition name] (条件名)

   ルールに追加する条件を選択します。リストには、前のステップで選択したタイプの条件のみが表示されます。

6. 条件を削除するには、条件名の右側にある [X] を選択します。

7. [更新] を選択します。

ルールの削除

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

ルールを削除する場合は、最初にそのルールを使用しているウェブ ACL からルールを削除し、次にルール内に含まれている条件を削除します。

ルールを削除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ルールを使用しているウェブ ACL から、このルールを削除するには、ウェブ ACL のそれぞれについて、次のステップを実行します。

   a. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。

   b. 削除するルールを使用しているウェブ ACL の名前を選択します。

   c. [Rules] (ルール) タブを選択します。

   d. [Edit web ACL] (ウェブ ACL を編集) を選択します。

   e. 削除するルールの右側にある [X] を選択してから、[Update] (更新) を選択します。
3. ナビゲーションペインで [Rules] (ルール) を選択します。
4. 削除するルールの名前を選択します。
5. [削除] をクリックします。

AWS Marketplace ルールグループ

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS WAF Classic には、AWS Marketplace リソースを保護するのに役立つルールグループが用意されています。 AWS Marketplace ルールグループは、 AWS AWS パートナー企業によって作成および更新された、 ready-to-use 事前定義されたルールの集まりです。

AWS Marketplace ルールグループの中には WordPress、Joomla や PHP などの特定の種類のウェブアプリケーションの保護に役立つように設計されているものもあります。また、OWASP トップ 10 に挙げられているような既知の脅威や一般的な Web AWS Marketplace アプリケーションの脆弱性に対する幅広い保護を提供するルールグループもあります。

AWS 任意のパートナーが提供する 1 AWS Marketplace つのルールグループをインストールできます。また、 AWS WAF 保護を強化するためにカスタマイズした独自のクラシックルールを追加することもできます。PCI や HIPAA などの規制の遵守が必要な場合は、 AWS Marketplace ルールグループを使用してウェブアプリケーションファイアウォールの要件を満たすことができます。

AWS Marketplace ルールグループには長期契約や最低契約はありません。ルールグループをサブスクライブすると、月額料金 (時間数で按分) およびボリュームに基づく継続中のリクエスト料金が課金されます。詳細については、「AWS WAF クラシック価格設定」と、 AWS Marketplace に記載されている各ルールグループの説明を参照してください。 AWS Marketplace

自動更新

絶えず変化する脅威の状況を常に把握しておくのは、時間と費用がかかる場合があります。 AWS Marketplace ルールグループを使用すると、 AWS WAF クラシックを実装して使用する時間を節約できます。もう 1 つの利点は、 AWS 新しい脆弱性や脅威が出現したときに、 AWS AWS Marketplace パートナーがルールグループを自動的に更新できることです。

新しい脆弱性が公開前にパートナーの多くに通知されます。新しい脅威が広く知られる前でも、パートナーはルールグループを更新してお客様にデプロイできます。また、最新の脅威を調査して分析して最も関連性の高いルールを作成する脅威調査チームも数多くあります。

AWS Marketplace ルールグループ内のルールにアクセスできます。

AWS Marketplace 各ルールグループには、防御の対象となる攻撃や脆弱性の種類が包括的に説明されています。ルールグループプロバイダーの知的財産を保護するために、ルールグループ内の個々のルールを表示することはできません。この制限は、悪意のあるユーザーが公開されたルールを特に回避する脅威を設計するのを防ぐのにも役立ちます。

AWS Marketplace ルールグループ内の個々のルールは表示できないため、ルールグループ内のルールを編集することもできません。 AWS Marketplace ただし、ルールグループから特定のルールを除外できます。これは、「ルールグループ例外」と呼ばれます。ルールの除外は、これらのルールを削除しません。むしろ、ルールのアクションを COUNT に変更します。そのため、除外されたルールと一致するリクエストはカウントされますが、ブロックされません。除外されたルールごとに COUNT メトリクスを受信します。

予期せずにトラフィックをブロックしているルールグループのトラブルシューティングを行う場合、ルールを除外することが役立ちます (誤検出)。トラブルシューティングの手法の 1 つは、目的のトラフィックをブロックしているルールグループ内の特定のルールを識別し、その特定のルールを無効にする (除外する) ことです。

特定のルールを除外することに加えて、ルールグループ全体を有効または無効にすることで保護を絞り込むか、実行するルールグループアクションを選択できます。詳細については、「AWS Marketplace ルールグループの使用」を参照してください。

クォータ

AWS Marketplace 有効にできるルールグループは 1 つだけです。を使用して作成したカスタムルールグループを 1 つ有効にすることもできます AWS Firewall Manager。これらのルールグループは、ウェブ ACL ごとの 10 ルールの最大クォータにカウントされます。したがって、1 AWS Marketplace つのウェブ ACL には、1 つのルールグループ、1 つのカスタムルールグループ、および最大 8 つのカスタムルールを設定できます。

料金

AWS Marketplace ルールグループの料金については、AWS WAFAWS Marketplace クラシック価格と各ルールグループの説明を参照してください AWS Marketplace。

AWS Marketplace ルールグループの使用

AWS WAF Classic Console では、ルールグループを購読したり、 AWS Marketplace ルールグループから購読解除したりできます。ルールグループから特定のルールを除外することもできます。

AWS Marketplace ルールグループをサブスクライブして使用するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF でコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで [Marketplace] を選択します。
3. [Available marketplace products] (利用可能な Marketplace 製品) セクションで、ルールグループの名前を選択して、詳細と料金情報を表示します。
4. ルールグループにサブスクライブする場合は、[Continue] (続行) を選択します。

   > **Note**
   >
   > このルールグループをサブスクライブしたくない場合は、ブラウザでこのページを閉じるだけです。

5. [Set up your account] (アカウントをセットアップ) を選択します。
6. 個々のルールを追加するのと同様の方法で、ウェブ ACL にルールグループを追加します。詳細については、「ウェブ ACL の作成」または「ウェブ ACL の編集」を参照してください。

   > **Note**
   >
   > ウェブ ACL にルールグループを追加するときに、ルールグループ ([No override] (上書きしない) または [Override to count] (カウントに上書き) に設定したアクションは、ルールグループ上書きアクションと呼びます。詳細については、「ルールグループの上書き」を参照してください。

AWS Marketplace ルールグループのサブスクリプションを解除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF にあるコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. すべてのウェブ ACL からルールグループを削除します。詳細については、「ウェブ ACL の編集」を参照してください。

3. ナビゲーションペインで [Marketplace] を選択します。
4. [Manage your subscriptions] (サブスクリプションを管理) を選択します。
5. サブスクリプションを解除するルールグループの名前の横にある [Cancel subscription] (サブスクリプションをキャンセル) を選択します。
6. [Yes, cancel subscription] (はい、サブスクリプションをキャンセルします) を選択します。

ルールグループからルールを除外するには (ルールグループ例外)

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF にあるコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. まだ有効になっていない場合は、 AWS WAF クラシックロギングを有効にします。詳細については、「ウェブ ACL トラフィック情報のログ記録」を参照してください。 AWS WAF クラシックログを使用して、除外するルールの ID を特定します。これらは通常、正規のリクエストをブロックしているルールです。
3. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。
4. 編集するウェブ ACL の名前を選択します。これにより、右ペインで、ウェブ ACL の詳細を含むページが開きます。

   > **Note**
   >
   > 編集するルールグループは、そのルールグループからルールを除外する前にウェブ ACL に関連付ける必要があります。

5. 右ペインの [Rules] (ルール) タブで、[Edit web ACL] (ウェブ ACL を編集) を選択します。
6. [Rule group exceptions] (ルールグループ例外) セクションに、編集するルールグループを展開します。
7. 除外するルールの隣にある [X] を選択します。 AWS WAF クラシックログを使用すると、正しいルール ID を特定できます。
8. [Update] (更新) を選択します。

   ルールの除外は、ルールグループからこれらのルールを削除しません。むしろ、ルールのアクションを COUNT に変更します。そのため、除外されたルールと一致するリクエストはカウントされますが、ブロックされません。除外されたルールごとに COUNT メトリクスを受信します。

> **Note**
>
> この同じ手順を使用して、 AWS Firewall Managerで作成したカスタムルールグループからルールを除外できます。ただし、これらのステップを使用してカスタムルールグループからルールを除外するよりも、「AWS WAF クラシックルールグループからのルールの追加と削除」で説明されているステップを使用してカスタムルールグループを編集することもできます。

ルールグループの上書き

AWS Marketplace ルールグループには、[オーバーライドなし] と [カウント対象の上書き] の 2 つのアクションがあります。ルールグループをテストする場合は、アクションを [Override to count] (カウントに上書き) に設定します。このルールグループアクションは、グループに含まれる個々のルールで指定された BLOCK アクションを上書きします。つまり、ルールグループのアクションが [Override to count] (カウントに上書き) に設定されている場合は、グループ内の個々のルールのアクションに基づいて一致するリクエストをブロックするのではなく、それらのリクエストがカウントされます。逆に、ルールグループのアクションを [No override] (上書きしない) に設定すると、グループ内の個々のルールのアクションが使用されます。

AWS Marketplace ルールグループのトラブルシューティング

AWS Marketplace ルールグループが正当なトラフィックをブロックしていることがわかった場合は、次の手順を実行してください。

AWS Marketplace ルールグループをトラブルシューティングするには

1. 正当なトラフィックをブロックしている特定のルールを除外します。 AWS WAF クラシックログを使用して、どのルールがどのリクエストをブロックしているかを特定できます。ルールの除外の詳細については、ルールグループからルールを除外するには (ルールグループ例外) を参照してください。
2. 特定のルールを除外しても問題が解決しない場合は、 AWS Marketplace ルールグループのアクションを [オーバーライドなし] から [オーバーライドしてカウントする] に変更できます。これにより、ルールグループ内の個々のルールアクションに関係なく、ウェブリクエストが通過します。これにより、ルールグループの Amazon CloudWatch メトリックスも表示されます。
3. AWS Marketplace ルールグループのアクションを Override to count に設定したら、ルールグループプロバイダーのカスタマーサポートチームに連絡して、問題をさらにトラブルシューティ

ングしてください。連絡先については、 AWS Marketplaceの製品リストページのルールグループリストを参照してください。

カスタマーサポートへの問い合わせ

AWS WAF Classic またはによって管理されているルールグループに関する問題については AWS、お問い合わせください AWS Support。パートナーが管理するルールグループに問題がある場合は、AWS そのパートナーのカスタマーサポートチームに連絡してください。パートナーの連絡先情報については、のパートナーのリストを参照してください AWS Marketplace。

AWS Marketplace ルールグループの作成と販売

AWS Marketplace ルールグループを販売したい場合は AWS Marketplace、「ソフトウェアの販売方法」を参照してください AWS Marketplace。

ウェブ ACL の使用

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

ウェブ ACL にルールを追加するときは、 AWS WAF Classic でルールの条件に基づいてリクエストを許可するか拒否するかを指定します。ウェブ ACL に複数のルールを追加すると、 AWS WAF Classic はウェブ ACL にリストされている順序で各リクエストをルールと照合して評価します。ウェブリクエストがルールのすべての条件に一致すると、 AWS WAF Classic は直ちに対応するアクション (許可または拒否) を実行し、ウェブ ACL 内の残りのルール (存在する場合) と照合してリクエストを評価しません。

ウェブリクエストがウェブ ACL のどのルールにも一致しない場合、 AWS WAF Classic はウェブ ACL に指定したデフォルトアクションを実行します。詳細については、「ウェブ ACL のデフォルトアクションの決定」を参照してください。

リクエストを許可または拒否するためにルールを使用する前にルールをテストしたい場合は、ルールの条件に一致するウェブリクエストをカウントするように AWS WAF Classic を設定できます。詳細については、「ウェブ ACL のテスト」を参照してください。

トピック

- ウェブ ACL のデフォルトアクションの決定
- ウェブ ACL の作成
- ウェブ ACL と Amazon API Gateway API、 CloudFront ディストリビューション、または Application Load Balancer の関連付けまたは関連付けの解除
- ウェブ ACL の編集
- ウェブ ACL の削除
- ウェブ ACL のテスト

ウェブ ACL のデフォルトアクションの決定

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

ウェブ ACL を作成して設定する際、最初に行うべき最も重要な決定は、 AWS WAF Classic のデフォルトアクションをウェブリクエストを許可するのか、それともウェブリクエストをブロックするのかです。デフォルトアクションは、指定したすべての条件についてウェブリクエストを検査し、ウェブリクエストがこれらの条件のいずれにも一致しなかった場合に AWS WAF Classic に何をさせたいかを示します。

- [Allow] (許可) 大部分のユーザーに対してはウェブサイトへのアクセスを許可する一方、指定した IP アドレスからのリクエストまたは悪意のある SQL コードや指定した値が含まれている可能性があるリクエストを行う攻撃者に対してアクセスをブロックする場合は、デフォルトアクションとして [Allow] (許可) を選択します。

- [Block] (ブロック) 大部分の自称ユーザーに対してはウェブサイトへのアクセスを拒否する一方、指定した IP アドレスからのリクエストや指定した値が含まれているリクエストのユーザーに対してアクセスを許可する場合は、デフォルトアクションとして [Block] (ブロック) を選択します。

デフォルトアクションを決めた後は、通常、大部分のウェブリクエストを許可するかブロックするかで条件を決めます。例えば、大部分のウェブリクエストを許可する場合は、通常、次のようなウェブリクエストをブロックする一致条件を作成します。

- リクエスト数が不当に多い IP アドレスからのリクエスト
- お客様がビジネスを行っていない国、または頻繁に攻撃元になっている国からのリクエスト
- [User-Agent] ヘッダーに不正な値が含まれているリクエスト
- 悪意のある SQL コードが含まれている可能性があるリクエスト

ウェブ ACL の作成

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

ウェブ ACL を作成するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。

2. クラシックを初めて使用する場合は、[AWS WAF AWS WAF クラシックに移動] を選択し、[Web ACL の設定] を選択します。 AWS WAF Classic を以前に使用したことがある場合は、ナビゲーションペインで [Web ACL] を選択し、[ウェブ ACL の作成] を選択します。
3. [Web ACL name] (ウェブ ACL の名前) に名前を入力します。

> **Note**
>
> ウェブ ACL の作成後は、名前を変更することはできません。

4. [CloudWatch メトリック名] では、必要に応じてデフォルト名を変更します。名前には英数字 (A ~ Z、a ~ z、0 ~ 9) のみを使用することができ、最大 128 文字、最小 1 文字です。空白や「All」や「Default_Action」など、 AWS WAF クラシック専用の指標名は使用できません。

> **Note**
>
> ウェブ ACL の作成後は、名前を変更することはできません。

5. [Region] (リージョン) で、リージョンを選択します。
6. [AWS resource (リソース)] で、ウェブ ACL に関連付けるリソースを選択し、[Next (次へ)] を選択します。
7. AWS WAF Classic でウェブリクエストの検査に使用させたい条件をすでに作成している場合は、[次へ] を選択し、次のステップに進みます。

   条件をまだ作成していない場合は、ここで作成します。詳細については、次のトピックを参照してください。

   - クロスサイトスクリプティング一致条件の使用
   - IP 一致条件の使用
   - Geo (地理的) 一致条件の使用
   - サイズ制約条件の使用
   - SQL インジェクション一致条件の使用
   - 文字列一致条件の使用
   - 正規表現一致条件の使用
8. このウェブ ACL に追加するルールまたはルールグループをすでに作成している (AWS Marketplace またはルールグループに登録している) 場合は、ウェブ ACL にルールを追加します。

   a. [Rules] (ルール) リストで、ルールを選択します。

   b. [Add rule to web ACL] (ウェブ ACL にルールを追加) を選択します。

c. このウェブ ACL に関連付けるすべてのルールを追加するまでステップ a とステップ b を繰り返します。

d. ステップ 10 に進んでください。

9. ルールをまだ作成していない場合は、ここでルールを追加できます。

a. [Create rule] (ルールの作成) を選択します。

b. 次の値を入力します。

[Name] (名前)

名前を入力します。

CloudWatch メトリックス名

AWS WAF Classic CloudWatch が作成してルールに関連付ける指標の名前を入力します。名前には英数字 (A～Z、a～z、0～9) のみを使用することができ、最大 128 文字、最小 1 文字です。空白や、「All」および「Default_Action」など AWS WAF Classic 用に予約されたメトリクス名は使用できません。

> **Note**
> ルールの作成後はメトリクス名を変更できません。

c. ルールに条件を追加するには、次の値を指定します。

[When a request does/does not] (リクエストが次の条件内/条件外)

IP アドレス 192.0.2.0/24 の範囲から発信されるウェブリクエストなど、条件内のフィルターに基づいてリクエストを許可または拒否するようにするには、[Does] を選択します AWS WAF 。

AWS WAF Classic で条件内のフィルターの逆に基づいてリクエストを許可または拒否したい場合は、「しない」を選択します。たとえば、IP 一致条件に 192.0.2.0/24 の IP アドレス範囲が含まれ、その IP アドレスから送信されないリクエストを AWS WAF Classic で許可または拒否したい場合は、「しない」を選択します。

[match/originate from] (一致/次から生じている)

ルールに追加する条件のタイプを選択します。

- クロスサイトスクリプティング一致条件 - [match at least one of the filters in the cross-site scripting match condition] (クロスサイトスクリプティング一致条件の少なくとも 1 つのフィルターに一致する) を選択します。
- IP 一致条件 - [originate from an IP address in] (IP アドレスより送信) を選択します
- Geo 一致条件 - [originate from a geographic location in] (地理的場所より送信) を選択します
- サイズ制約条件 - [match at least one of the filters in the size constraint condition] (サイズ制約条件の少なくとも 1 つのフィルターに一致する) を選択します
- SQL インジェクション一致条件 - [match at least one of the filters in the SQL injection match condition] (SQL インジェクション一致条件の少なくとも 1 つのフィルターに一致する) を選択します
- 文字列一致条件 - [match at least one of the filters in the string match condition] (文字列一致条件の少なくとも 1 つのフィルターに一致する) を選択します
- 正規表現一致条件 - [match at least one of the filters in the regex match condition] (正規表現一致条件の少なくとも 1 つのフィルターに一致する) を選択します

[condition name] (条件名)

ルールに追加する条件を選択します。リストには、前のリストで選択したタイプの条件のみが表示されます。

d. ルールに別の条件を追加するには、[Add another condition] (別の条件を追加) を選択し、ステップ b とステップ c を繰り返します。次の点に注意してください。

- 複数の条件を追加した場合、 AWS WAF Classic がそのルールに基づいてリクエストを許可または拒否するには、ウェブリクエストがすべての条件の少なくとも 1 つのフィルターに一致する必要があります。
- 同じルールに 2 つの IP 一致条件を追加した場合、 AWS WAF Classic は両方の IP 一致条件に含まれる IP アドレスから発信されたリクエストのみを許可または拒否します。

e. ステップ 9 を繰り返して、このウェブ ACL に追加するすべてのルールを作成します。

f. [Create] (作成) を選択します。

g. ステップ 10 に進みます。

10. ウェブ ACL のルールまたはルールグループごとに、 AWS WAF Classic に提供する管理の種類を次のように選択します。

- ルールごとに、 AWS WAF Classic でルールの条件に基づいてウェブリクエストを許可、ブロック、カウントするかどうかを選択します。

- 許可 — API Gateway CloudFront またはApplication Load Balancer がリクエストされたオブジェクトで応答します。の場合 CloudFront、オブジェクトがエッジキャッシュにない場合は、 CloudFront リクエストをオリジンに転送します。
- ブロック — API Gateway CloudFront またはApplication Load Balancer が HTTP 403 (禁止) ステータスコードでリクエストに応答します。CloudFront カスタムエラーページで応答することもできます。詳細については、「AWS WAF Classic CloudFront とカスタムエラーページとの併用」を参照してください。
- カウント — AWS WAF Classic は、ルール内の条件に一致するリクエストのカウンタを増やし、ウェブ ACL の残りのルールに基づいてウェブリクエストを検査し続けます。

  ウェブ ACL を使用してウェブリクエストを許可またはブロックする前に、[Count] (カウント) でウェブ ACL をテストする方法については、「ウェブ ACL のルールに一致するウェブリクエストのカウント」を参照してください。

- ルールグループごとに、ルールグループの上書きアクションを設定します。
  - [No override] (上書きなし) - ルールグループ内の個々のルールのアクションが使用されます。
  - [Override to count] (カウントに上書き) - グループ内の個々のルールによって指定されたブロックアクションを上書きし、一致するリクエストのみがすべてカウントされるようにします。

  詳細については、「ルールグループの上書き」を参照してください。

11. ウェブ ACL 内のルールの順序を変更する場合は、「順序」列の矢印を使用してください。AWS WAF Classic は、ウェブ ACL にルールが表示される順序に基づいてウェブリクエストを検査します。
12. ウェブ ACL に追加したルールを削除する場合は、ルールの行にある [x] を選択します。
13. ウェブ ACL のデフォルトアクションを選択します。これは、ウェブリクエストがこのウェブ ACL のどのルールにも一致しない場合に AWS WAF Classic が実行するアクションです。詳細については、「ウェブ ACL のデフォルトアクションの決定」を参照してください。
14. [Review and create] (確認および作成) を選択します。
15. ウェブ ACL の設定を確認し、[Confirm and create] (確認して作成) を選択します。

ウェブ ACL と Amazon API Gateway API、 CloudFront ディストリビューション、またはApplication Load Balancer の関連付けまたは関連付けの解除

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

ウェブ ACL の関連付けまたは関連付けの解除を行うには、該当する手順を実行します。ディストリビューションを作成または更新するときに、ウェブ ACL CloudFront をディストリビューションに関連付けることもできることに注意してください。詳細については、Amazon CloudFront 開発者ガイドの「AWS WAF Classic を使用してコンテンツへのアクセスを制御する」を参照してください。

ウェブ ACL を関連付けるとき、次の制限が適用されます。

- 各 API Gateway API、Application Load Balancer、 CloudFront ディストリビューションは 1 つのウェブ ACL にのみ関連付けることができます。
- CloudFront ディストリビューションに関連付けられた Web ACL は、Application Load Balancer や API Gateway API に関連付けることはできません。ただし、ウェブ ACL CloudFront は他のディストリビューションと関連付けることができます。

ウェブ ACL を API Gateway API、 CloudFront ディストリビューション、またはApplication Load Balancer に関連付けるには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。
3. API Gateway API、 CloudFront ディストリビューション、またはApplication Load Balancer に関連付けるウェブ ACL の名前を選択します。これにより、右ペインで、ウェブ ACL の詳細を含むページが開きます。

4. 「ルール」タブの「このウェブ ACL AWS を使用するリソース」で、「関連付けを追加」を選択します。
5. プロンプトが表示されたら、リソースリストを使用して、このウェブ ACL に関連付ける API Gateway API、 CloudFrontディストリビューション、またはApplication Load Balancer を選択します。Application Load Balancer を選択した場合は、リージョンも指定する必要があります。
6. [Add] (追加) を選択します。
7. このウェブ ACL を追加の API Gateway API、 CloudFront ディストリビューション、または別の Application Load Balancer に関連付けるには、ステップ 4 ~ 6 を繰り返します。

ウェブ ACL と API Gateway API、 CloudFront ディストリビューション、またはApplication Load Balancer との関連付けを解除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。
3. API Gateway API、 CloudFront ディストリビューション、またはApplication Load Balancer との関連付けを解除するウェブ ACL の名前を選択します。これにより、右ペインで、ウェブ ACL の詳細を含むページが開きます。
4. 「ルール」タブの「このウェブ ACL AWS を使用するリソース」で、このウェブ ACL の関連付けを解除したい API Gateway API、 CloudFront ディストリビューション、またはApplication Load Balancer ごとに x を選択します。

ウェブ ACL の編集

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

ウェブ ACL のルールを追加または削除したり、デフォルトアクションを変更したりするには、次の手順を実行します。

ウェブ ACL を編集するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。
3. 編集するウェブ ACL の名前を選択します。これにより、右ペインで、ウェブ ACL の詳細を含むページが開きます。
4. 右ペインの [Rules] (ルール) タブで、[Edit web ACL] (ウェブ ACL を編集) を選択します。
5. ウェブ ACL にルールを追加するには、次のステップを実行します。

   a. [Rules] (ルール) リストで、追加するルールを選択します。

   b. [Add rule to web ACL] (ウェブ ACL にルールを追加) を選択します。

   c. ステップ a とステップ b を繰り返して、すべての必要なルールを追加します。
6. ウェブ ACL 内のルールの順序を変更する場合は、「順序」列の矢印を使用してください。AWS WAF Classic は、ウェブ ACL にルールが表示される順序に基づいてウェブリクエストを検査します。
7. ウェブ ACL からルールを削除するには、そのルールの行の右にある [x] を選択します。これにより AWS WAF Classic からルールが削除されるわけではなく、このウェブ ACL からルールが削除されるだけです。
8. ルールのアクションまたはウェブ ACL のデフォルトアクションを変更するには、希望するオプションを選択します。

   > **Note**
   >
   > ルールグループまたはルールグループ (1 AWS Marketplace つのルールではなく) にアクションを設定する場合、ルールグループに設定したアクション ([オーバーライドなし] または [カウントするオーバーライド] のいずれか) はオーバーライドアクションと呼ばれます。詳細については、「ルールグループの上書き」を参照してください。
9. [Save changes] (変更の保存) を選択します。

ウェブ ACL の削除

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

ウェブ ACL を削除するには、ウェブ ACL に含まれているルールを削除し、ウェブ ACL CloudFront からすべてのディストリビューションとアプリケーションロードバランサーの関連付けを解除する必要があります。次の手順を実行します。

ウェブ ACL を削除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ のコンソールを開きます。 AWS WAF

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。
3. 削除するウェブ ACL の名前を選択します。これにより、右ペインで、ウェブ ACL の詳細を含むページが開きます。
4. 右ペインの [Rules] (ルール) タブで、[Edit web ACL] (ウェブ ACL を編集) を選択します。
5. ウェブ ACL からすべてのルールを削除するには、各ルールの行の右にある [x] を選択します。AWS WAF クラシックからルールが削除されるわけではなく、このウェブ ACL からルールが削除されるだけです。
6. [更新] を選択します。
7. ウェブ ACL CloudFront をすべてのディストリビューションとアプリケーションロードバランサーから切り離します。「ルール」タブの「このウェブ ACL AWS を使用するリソース」で、各 API Gateway API、 CloudFront ディストリビューション、またはApplication Load Balancer の x を選択します。
8. [Web ACLs] (ウェブ ACL) ページで、削除するウェブ ACL が選択されていることを確認し、[Delete] (削除) を選択します。

ウェブ ACL のテスト

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

許可したいウェブリクエストやブロックしたいリクエストをブロックするように AWS WAF Classic を誤って設定してしまわないように、Web ACL をウェブサイトやウェブアプリケーションで使用する前に、ウェブ ACL を十分にテストすることをお勧めします。

トピック

- ウェブ ACL のルールに一致するウェブリクエストのカウント
- API Gateway CloudFront またはApplication Load Balancer が Classic に転送したウェブリクエストのサンプルを表示する AWS WAF

ウェブ ACL のルールに一致するウェブリクエストのカウント

ウェブ ACL にルールを追加するときは、 AWS WAF Classic でそのルールのすべての条件に一致するウェブリクエストを許可、ブロック、またはカウントするかどうかを指定します。次の設定から始めることをお勧めします。

- ウェブリクエストをカウントするようにウェブ ACL のすべてのルールを設定する
- リクエストを許可するウェブ ACL のデフォルトアクションを設定する

この設定では、 AWS WAF Classic は最初のルールの条件に基づいて各ウェブリクエストを検査します。ウェブリクエストがそのルールのすべての条件に一致すると、 AWS WAF Classic はそのルールのカウンタをインクリメントします。その後、 AWS WAF Classic は次のルールの条件に基づいてウェブリクエストを検査します。リクエストがそのルールのすべての条件に一致すると、 AWS WAF Classic はそのルールのカウンタをインクリメントします。この処理は、 AWS WAF Classic がすべてのルールの条件に基づいてリクエストを検査するまで続きます。

リクエストをカウントするようにウェブ ACL のすべてのルールを設定し、ウェブ ACL を Amazon API Gateway API、 CloudFront ディストリビューション、またはApplication Load Balancer に関連付けると、結果の数を Amazon CloudWatch グラフで表示できます。ウェブ ACL 内の各ルール、および API Gateway CloudFront またはApplication Load Balancer がウェブ ACL の AWS WAF Classic に転送するすべてのリクエストについて、 CloudWatch 次のことができます。

- 1 時間前または 3 時間前のデータを表示する
- データポイント間の間隔を変更する
- 最大値、最小値、平均値、合計など、 CloudWatch データに対して実行する計算を変更できます。

> **Note**
>
> AWS WAF Classic with CloudFront はグローバルサービスであり、メトリクスはで米国東部 (バージニア北部) AWS Management Consoleリージョンを選択した場合にのみ利用できます。別のリージョンを選択した場合、 AWS WAF Classic CloudWatch メトリクスはコンソールに表示されません。

ウェブ ACL でルールのデータを表示するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/cloudwatch/ CloudWatch のコンソールを開きます。
2. ナビゲーションペインの [Metrics] (メトリクス) で、[WAF] を選択します。
3. データを表示するウェブ ACL のチェックボックスをオンにします。
4. 該当する設定を変更します。

   [Statistic] (統計)

   CloudWatch データに対して実行する計算を選択します。

   [Time range] (時間範囲)

   前の 1 時間のデータを表示するか、前の 3 時間のデータを表示するかを選択します。

   [Period] (期間)

   グラフでのデータポイント間の間隔を変更します。

[Rules] (ルール)

データを表示するルールを選択します。

次の点に注意してください。

- ウェブ ACL を API Gateway API、 CloudFront ディストリビューション、またはApplication Load Balancer に関連付けたばかりの場合は、データがグラフに表示され、ウェブ ACL のメトリックスが利用可能なメトリクスのリストに表示されるまで、数分待つ必要がある場合があります。
- 複数の API Gateway API、 CloudFront ディストリビューション、またはApplication Load Balancer をウェブ ACL に関連付ける場合、 CloudWatch データにはウェブ ACL に関連付けられているすべてのディストリビューションのすべてのリクエストが含まれます。
- データポイントの上にマウスカーソルを置くと、詳細情報が表示されます。
- グラフは自動的には更新されません。表示を更新するには、更新 () アイコンを選択します。

5. (オプション) API Gateway CloudFront またはApplication Load Balancer AWS WAF がクラシックに転送した個々のリクエストに関する詳細情報を表示します。詳細については、「API Gateway CloudFront またはApplication Load Balancer が Classic に転送したウェブリクエストのサンプルを表示する AWS WAF」を参照してください。
6. ルールにより、傍受する必要のないリクエストが傍受されていると判断した場合は、該当する設定を変更します。詳細については、「ウェブアクセスコントロールリスト (ウェブ ACL) の作成と設定」を参照してください。

   すべてのルールにより、正しいリクエストのみが傍受されていることを確認したら、各ルールのアクションを [Allow] (許可) または [Block] (ブロック) に変更します。詳細については、「ウェブ ACL の編集」を参照してください。

API Gateway CloudFront またはApplication Load Balancer が Classic に転送したウェブリクエストのサンプルを表示する AWS WAF

AWS WAF クラシックコンソールでは、API Gateway CloudFront またはApplication Load Balancer AWS WAF が検査のためにクラシックに転送したリクエストのサンプルを表示できます。サンプリングされたリクエストごとに、発生元の IP アドレスやリクエストに含まれるヘッダーなど、リクエストに関する詳細なデータを表示できます。また、リクエストが一致したルールを表示したり、ルー

ルがリクエストを許可するように設定されているかブロックするように設定されているかを確認したりもできます。

リクエストのサンプルには、各ルールのすべての条件に一致した最大 100 件のリクエストと、デフォルトアクションが適用された別の 100 件のリクエストが含まれます。デフォルトアクションは、すべてのルールのすべての条件に一致しなかったリクエストに適用されます。サンプルのリクエストは、過去 15 分間にコンテンツのリクエストを受信したすべての API Gateway API、 CloudFront エッジロケーション、またはアプリケーションロードバランサーからのものです。

API Gateway、 CloudFront またはApplication Load Balancer が Classic に転送したウェブリクエストのサンプルを表示するには AWS WAF

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで、リクエストを表示するウェブ ACL を選択します。
3. 右ペインで、[Requests] (リクエスト) タブを選択します。

   [Sampled requests] (サンプリングされたリクエスト) テーブルには、リクエストごとに次の値が表示されます。

   [Source IP] (送信元 IP)

   リクエストの発生元の IP アドレス (ビューワーが HTTP プロキシまたは Application Load Balancer を使用してリクエストを送信した場合は、そのプロキシまたは Application Load Balancer の IP アドレス)。

   [URI]

   リクエストの URI パス。リソースを識別します (例: `/images/daily-ad.jpg`)。これには、URI のクエリ文字列またはフラグメントコンポーネントは含まれません。詳細については、「Uniform Resource Identifier (URI): 一般的な構文」を参照してください。

   [Matches rule] (一致ルール)

   ウェブリクエストがすべての条件に一致したウェブ ACL の最初のルールを識別します。ウェブリクエストがウェブ ACL のすべてのルールのすべての条件に一致しない場合、[Matches rule] (ルールに一致) の値は [Default] (デフォルト) です。

ウェブリクエストがルールのすべての条件に一致し、そのルールのアクションが Count の場合、 AWS WAF Classic はウェブ ACL の以降のルールに基づいてウェブリクエストを検査し続けることに注意してください。この場合、ウェブリクエストはサンプリングされたリクエストのリストに 2 回表示されることがあります。1 回は、アクションが [Count] (カウント) のルールに対応し、もう 1 回は、後続のルールまたはデフォルトアクションに対応します。

[Action] (アクション)

対応するルールのアクションが [Allow] (許可)、[Block] (ブロック)、[Count] (カウント) のいずれかであるかを示します。

[Time] (時間)

AWS WAF Classic が API Gateway CloudFront またはApplication Load Balancer からリクエストを受け取った時間。

4. リクエストに関する追加情報を表示するには、そのリクエストの IP アドレスの左側にある矢印を選択します。 AWS WAF Classic には以下の情報が表示されます。

[Source IP] (送信元 IP)

テーブルの [Source IP] (ソース IP) 列の値と同じ IP アドレス。

[Country] (国)

リクエスト送信元の国の 2 文字の国コード。ビューワーが HTTP プロキシまたは Application Load Balancer を使用してリクエストを送信する場合、これは HTTP プロキシまたは Application Load Balancer が存在する国の 2 文字の国コードです。

2 文字の国コードと対応する国名のリストについては、Wikipedia の「ISO 3166-1 alpha-2」記事を参照してください。

[Method] (メソッド)

リクエストの HTTP リクエストメソッド:
`GET`、`HEAD`、`OPTIONS`、`PUT`、`POST`、`PATCH`、`DELETE`。

[URI]

テーブルの [URI] 列の値と同じ URI。

[Request headers] (リクエストヘッダー)

リクエストのヘッダーとヘッダー値。

5. サンプルのリクエストのリストを更新するには、[Get new samples] (新しいサンプルを取得) を選択します。

AWS WAF で使用するためのクラシックルールグループの使用 AWS Firewall Manager

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールや Web ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS WAF クラシックルールグループは、 AWS WAF AWS Firewall Manager クラシックポリシーに追加するルールのセットです。独自のルールグループを作成することも、マネージドルールグループを購入することもできます AWS Marketplace。

> **Important**
>
> AWS Marketplace ルールグループを Firewall Manager ポリシーに追加する場合、組織内の各アカウントは最初にそのルールグループに登録する必要があります。すべてのアカウントをサブスクライブしたら、ルールグループをポリシーに追加できます。詳細については、「AWS Marketplace ルールグループ」を参照してください。

トピック

- AWS WAF クラシックルールグループの作成
- AWS WAF クラシックルールグループからのルールの追加と削除

AWS WAF クラシックルールグループの作成

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールや Web ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS WAF で使用するクラシックルールグループを作成するときは AWS Firewall Manager、そのグループに追加するルールを指定します。

ルールグループを作成するには (コンソール)

1. AWS Management Console AWS Firewall Manager 前提条件で設定した管理者アカウントを使用してにサインインし、で Firewall Manager コンソールを開きます。https://console.aws.amazon.com/wafv2/fms

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「ステップ 2: AWS Firewall Manager 既定の管理者アカウントを作成する」を参照してください。

2. ナビゲーションペインで [クラシックに切り替え] を選択します。 AWS WAF
3. AWS WAF クラシックナビゲーションペインで、[ルールグループ] を選択します。
4. [Create rule group] (ルールグループの作成) を選択します。

   > **Note**
   >
   > ルールグループにレートベースのルールを追加することはできません。

5. ルールグループに追加するルールを既に作成している場合は、[Use existing rules for this rule group] (このルールグループに既存のルールを使用する) を選択します。ルールグループに追加する新しいルールを作成する場合は、[Create rules and conditions for this rule group] (このルールグループにルールと条件を作成する) を選択します。
6. [Next] (次へ) を選択します。

7. ルールの作成を選択した場合は、「ルールの作成と条件の追加」のステップに従ってルールを作成します。

   > **Note**
   >
   > AWS WAF クラシックコンソールを使用してルールを作成します。

   必要なルールをすべて作成したら、次のステップに進みます。

8. ルールグループ名を入力します。
9. ルールグループにルールを追加するには、ルールを選択し、[Add rule] (ルールの追加) を選択します。ルールの条件に一致するリクエストを許可するか、ブロックするか、カウントするかを選択します。選択の詳細については、「AWS WAF クラシックの仕組み」を参照してください。
10. ルールの追加が完了したら、[Create] (作成) を選択します。

ルールグループを WebACL に追加し、 AWS WAF WebACL アクションを [Override to Count] に設定することでテストできます。このアクションは、グループに含まれるルールに対して選択したアクションを上書きし、一致するリクエストのみカウントします。詳細については、「ウェブ ACL の作成」を参照してください。

AWS WAF クラシックルールグループからのルールの追加と削除

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS WAF クラシックルールグループでは、ルールを追加または削除できます。

ルールグループからルールを削除しても、ルール自体は削除されません。ルールグループからルールが削除されるだけです。

ルールグループにルールを追加または削除するには (コンソール)

1. AWS Management Console AWS Firewall Manager 前提条件で設定した管理者アカウントを使用してにサインインし、で Firewall Manager コンソールを開きます。https://console.aws.amazon.com/wafv2/fms

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「ステップ 2: AWS Firewall Manager 既定の管理者アカウントを作成する」を参照してください。

2. ナビゲーションペインで [クラシックに切り替え] を選択します。 AWS WAF
3. AWS WAF クラシックナビゲーションペインで、[ルールグループ] を選択します。
4. 編集するルールグループを選択します。
5. [Edit rule group] (ルールグループの編集) を選択します。
6. ルールを追加するには、次のステップを実行します。

   a. ルールを選択し、[Add rule to rule group] (ルールグループへのルールの追加) を選択します。ルールの条件に一致するリクエストを許可するか、ブロックするか、カウントするかを選択します。選択の詳細については、「AWS WAF クラシックの仕組み」を参照してください。ルールグループにさらにルールを追加するには、この操作を繰り返します。

   > **Note**
   >
   > レートベースのルールをルールグループに追加することはできません。

   b. [Update] (更新) を選択します。
7. ルールを削除するには、次のステップを実行します。

   a. 削除するルールの横にある [X] を選択します。ルールグループからさらにルールを削除するには、この操作を繰り返します。

   b. [Update] (更新) を選択します。

AWS Firewall ManagerAWS WAF クラシックルールを有効にするにはじめに

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールや Web ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS Firewall Manager を使用して、 AWS WAF ルール、 AWS WAF クラシックルール、 AWS Shield Advanced 保護、Amazon VPC セキュリティグループを有効にできます。セットアップのステップはそれぞれ少し異なります。

- Firewall Manager AWS WAFを使用して最新バージョンのを使用するルールを有効にする場合は、このトピックを使用しないでください。代わりに、「AWS Firewall ManagerAWS WAF ポリシー入門」のステップに従います。
- Firewall Manager AWS Shield Advanced を使用して保護を有効にするには、AWS Firewall ManagerAWS Shield Advanced ポリシー入門の手順に従ってください。
- Firewall Manager を使用して Amazon VPC セキュリティグループを有効にするには、「AWS Firewall Manager Amazon VPC セキュリティグループポリシーの使用を開始する」のステップに従います。

Firewall Manager AWS WAF を使用してクラシックルールを有効にするには、次の手順を順番に実行します。

トピック

ステップ 1: 前提条件を満たす

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS Firewall Managerのアカウントを準備するには、いくつかの必須のステップがあります。それらのステップは、AWS Firewall Manager 前提条件 で説明されています。「ステップ 2: ルールを作成する」に進む前に、すべての前提条件を満たしてください。

ステップ 2: ルールを作成する

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

このステップでは、 AWS WAF Classic を使用してルールを作成します。 AWS WAF 一緒に使用したいクラシックルールが既にある場合は AWS Firewall Manager、このステップをスキップしてに進んでくださいステップ 3: ルールグループを作成する。

> **Note**
>
> AWS WAF クラシックコンソールを使用してルールを作成します。

AWS WAF クラシックルールを作成するには (コンソール)

- ルールを作成し、ルールに条件を追加します。詳細については、「ルールの作成と条件の追加」を参照してください。

これで「ステップ 3: ルールグループを作成する」に進む準備ができました。

ステップ 3: ルールグループを作成する

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

ルールグループは、特定の一連の条件が満たされたときに実行するアクションを定義する一連のルールです。からマネージドルールグループを使用することも AWS Marketplace、独自のルールグループを作成することもできます。マネージドルールグループの詳細については、「AWS Marketplace ルールグループ」を参照してください。

独自のルールグループを作成するには、次の手順を実行します。

ルールグループを作成するには (コンソール)

1. AWS Management Console AWS Firewall Manager 前提条件で設定した管理者アカウントを使用してにサインインし、で Firewall Manager コンソールを開きます。https://console.aws.amazon.com/wafv2/fms
2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. 前提条件を満たしていない場合、問題を修正する方法についての指示がコンソールに表示されます。指示に従った後、このステップ (ルールグループを作成) を再試行します。前提条件を満たしている場合は、[Close] (閉じる) を選択します。
4. [Create policy] (ポリシーの作成) を選択します。

   [Policy type (ポリシータイプ)] で、[AWS WAF Classic] を選択します。
5. [AWS Firewall Manager ポリシーを作成] を選択し、新しいルールグループを追加します。

6. を選択し AWS リージョン、[次へ] を選択します。
7. 既にルールを作成しているため、条件を作成する必要はありません。[Next] (次へ) を選択します。
8. 既にルールを作成しているため、ルールを作成する必要はありません。[Next] (次へ) を選択します。
9. [Create rule group] (ルールグループの作成) を選択します。
10. [Name] (名前) で、わかりやすい名前を入力します。
11. AWS WAF Classic CloudWatch が作成してルールグループに関連付けるメトリックの名前を入力します。名前に使用できるのは英数字 (A～Z、a～z、0～9) または特殊文字 _-!"#`+*},./ です。空白を含めることはできません。
12. ルールを選択してから、[Add rule] (ルールの追加) を選択します。ルールには、ルールの条件に一致するリクエストを許可するか、ブロックするか、カウントするかを選択できるアクション設定があります。このチュートリアルでは、[Count] (カウント) を選択します。ルールグループに必要なすべてのルールを追加するまで、ルールの追加を繰り返します。
13. [Create] (作成) を選択します。

これで「ステップ 4: AWS Firewall ManagerAWS WAF クラシックポリシーを作成して適用する」に進む準備ができました。

ステップ 4: AWS Firewall ManagerAWS WAF クラシックポリシーを作成して適用する

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

ルールグループを作成したら、 AWS Firewall Manager AWS WAF ポリシーを作成します。Firewall Manager AWS WAF ポリシーには、リソースに適用するルールグループが含まれています。

Firewall Manager AWS WAF ポリシーを作成するには (コンソール)

1. ルールグループを作成すると (前の手順の最後のステップ「ステップ 3: ルールグループを作成する」)、[Rule group summary] (ルールグループの概要) ページがコンソールに表示されます。[Next] (次へ) を選択します。
2. [Name] (名前) で、わかりやすい名前を入力します。
3. [Policy type] (ポリシータイプ) で、[WAF] を選択します。
4. [リージョン] では、を選択します AWS リージョン。Amazon CloudFront リソースを保護するには、「グローバル」を選択してください。

   複数のリージョン (リソース以外) CloudFront のリソースを保護するには、リージョンごとに個別の Firewall Manager ポリシーを作成する必要があります。
5. 追加するルールグループを選択して、[Add rule group] (ルールグループの追加) を選択します。
6. ポリシーには、[Action set by rule group] (ルールグループによって設定されたアクション) と [Count] (カウント) の 2 つのアクションがあります。ポリシーをテストする場合は、アクションを [Count] (カウント) に設定します。このアクションは、ポリシーに含まれるルールグループで指定されたブロックアクションを上書きします。つまり、ポリシーのアクションが [Count] (カウント) に設定されている場合、リクエストはカウントされ、ブロックされません。逆に、ポリシーのアクションを [Action set by rule group] (ルールグループによって設定されたアクション) に設定すると、ポリシー内のルールグループのアクションが使用されます。このチュートリアルでは、[Count] (カウント) を選択します。
7. [Next] (次へ) を選択します。
8. ポリシーに特定のアカウントのみを含める場合やポリシーから特定のアカウントを除外する場合には、[Select accounts to include/exclude from this policy (optional)] (このポリシーに含める/除外するアカウントを選択する (オプション)) を選択します。[Include only these accounts in this policy] (このアカウントのみをこのポリシーに含める) あるいは [Exclude these accounts from this policy](このアカウントをこのポリシーから除外する) のどちらかを選択します 1 つのオプションのみを選択できます。[Add] (追加) を選択します。含めるアカウント番号または除外するアカウント番号を選び、[OK] を選択します。

> **Note**
>
> このオプションを選択しない場合、Firewall Manager は AWS Organizationsの組織内のすべてのアカウントにポリシーを適用します。組織に新しいアカウントを追加すると、Firewall Manager はそのアカウントにポリシーを自動的に適用します。

9. 保護するリソースのタイプを選択します。
10. 特定のタグを持つリソースのみを保護する場合、または特定のタグを持つリソースを除外する場合は、[Use tags to include/exclude resources] (タグを使用してリソースを含める/除外する) を選択し、タグを入力してから、[Include] (含める) または [Exclude] (除外) を選択します。1 つのオプションのみを選択できます。

    複数のタグをコンマで区切って入力する場合、リソースにこれらのタグのいずれかがある場合は一致するとみなされます。

    タグの詳細については、「タグエディタの使用」を参照してください。
11. [Create and apply this policy to existing and new resources] (既存および新規のリソースにこのポリシーを作成して適用する) を選択します。

    このオプションでは、内の組織内の該当するアカウントごとにウェブ ACL が作成され AWS Organizations、そのウェブ ACL がアカウント内の指定されたリソースに関連付けられます。このオプションは、前述の基準 (リソースタイプとタグ) に一致するすべての新しいリソースにもポリシーを適用します。また、[Create but do not apply this policy to existing or new resources] (作成するが既存および新規のリソースにこのポリシーを適用しない) を選択する場合は、Firewall Manager により組織内の各関連アカウントにウェブ ACL が作成されますが、ウェブ ACL はいずれのリソースにも適用されません。ポリシーは後でリソースに適用する必要があります。
12. [Replace existing associated web ACLs] (既存の関連付けられたウェブ ACL を置換) の選択肢は、デフォルト設定のままにします。

    このオプションを選択すると、Firewall Manager は、新しいポリシーのウェブ ACL を関連付ける前に、範囲内のリソースから既存のウェブ ACL の関連付けをすべて削除します。
13. [Next] (次へ) を選択します。
14. 新しいポリシーを確認します。設定を変更するには、[Edit] (編集) を選択します。ポリシーが完成したら、[Create policy] (ポリシーの作成) を選択します。

チュートリアル: 階層ルールによる AWS Firewall Managerポリシーの作成

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS Firewall Managerでは、 AWS WAF 階層ルールを含むクラシック保護ポリシーを作成して適用できます。つまり、特定のルールを一元的に作成および適用し、アカウント固有ルールの作成やメンテナンスを個々のユーザーに委任できるということです。一元的に適用する (共通) ルールを常に一貫して適用するために、誤操作による削除やその他の処理が行われないようモニタリングできます。アカウント固有のルールでは、個々のチームのニーズに合わせてカスタマイズされた保護をさらに追加できます。

> **Note**
>
> の最新バージョンでは AWS WAF、この機能が組み込まれており、特別な処理は必要ありません。 AWS WAF Classic をまだ使用していない場合は、代わりに最新バージョンを使用してください。「の AWS Firewall Manager ポリシーの作成 AWS WAF」を参照してください。

次のチュートリアルでは、保護ルールの階層セットを作成する方法について説明します。

トピック

- ステップ 1: Firewall Manager 管理者アカウントを指定する
- ステップ 2: Firewall Manager 管理者アカウントを使用してルールグループを作成する
- ステップ 3: Firewall Manager ポリシーを作成して共通のルールグループをアタッチする
- ステップ 4: アカウント固有のルールを追加する
- 結論

ステップ 1: Firewall Manager 管理者アカウントを指定する

を使用するには AWS Firewall Manager、組織内のアカウントを Firewall Manager 管理者アカウントとして指定する必要があります。このアカウントは、組織内の管理アカウントまたはメンバーアカウントのいずれでもかまいません。

Firewall Manager 管理者アカウントを使用すると、組織内の他のアカウントに適用する一連の共通ルールを作成できます。組織内の他のアカウントでは、このように一元的に適用されたルールを変更することはできません。

アカウントを Firewall Manager 管理者アカウントに指定して、Firewall Manager を使用するための外の前提条件を満たすには、「AWS Firewall Manager 前提条件」の説明を参照してください。前提条件を既に満たしている場合は、このチュートリアルのステップ 2 に進むことができます。

このチュートリアルでは、この管理者アカウントを **Firewall-Administrator-Account** と呼びます。

ステップ 2: Firewall Manager 管理者アカウントを使用してルールグループを作成する

次に、**Firewall-Administrator-Account** を使用してルールグループを作成します。このルールグループには、次のステップで作成するポリシーで管理されるすべてのメンバーアカウントに適用する共通ルールを指定します。これらのルールとコンテナルールグループを変更できるのは、**Firewall-Administrator-Account** のみです。

このチュートリアルでは、このコンテナルールグループを **Common-Rule-Group** と呼びます。

ルールグループを作成するには、「AWS WAF クラシックルールグループの作成」の手順を参照してください。これらの手順に従う際には、Firewall Manager 管理者アカウント (**Firewall-Administrator-Account**) を使用してコンソールにサインインしておいてください。

ステップ 3: Firewall Manager ポリシーを作成して共通のルールグループをアタッチする

Firewall-Administrator-Account を使用して、Firewall Manager ポリシーを作成します。このポリシーを作成する場合は、次を実行する必要があります。

- 新しいポリシーに **Common-Rule-Group** を追加する。

- **Common-Rule-Group** を適用する組織内のすべてのアカウントを含める。
- **Common-Rule-Group** を適用するすべてのリソースを追加する。

ポリシーの作成手順については、「AWS Firewall Manager ポリシーの作成」を参照してください。

これにより、指定された各アカウントにウェブ ACL が作成され、各ウェブ ACL に **Common-Rule-Group** が追加されます。ポリシーの作成後、このウェブ ACL および共通ルールは、指定されたすべてのアカウントにデプロイされます。

このチュートリアルでは、このウェブ ACL を **Administrator-Created-ACL** と呼びます。これで、組織内の指定されたメンバーアカウントごとに、一意の **Administrator-Created-ACL** が作成されます。

ステップ 4: アカウント固有のルールを追加する

組織内の各メンバーアカウントは、アカウントに存在する **Administrator-Created-ACL** に、アカウント固有のルールを自分で追加できます。すでに導入されている共通ルールは、**Administrator-Created-ACL**新しいアカウント固有のルールとともに引き続き適用されます。 AWS WAF ウェブ ACL にルールが表示される順序に基づいてウェブリクエストを検査します。これは、**Administrator-Created-ACL** とアカウント固有のルールの両方に当てはまります。

にルールを追加するには**Administrator-Created-ACL**、を参照してくださいウェブ ACL の編集。

結論

これで、Firewall Manager 管理者が管理する共通ルールが含まれたウェブ ACL と、各メンバーアカウントで管理するアカウント固有のルールを作成できました。

各アカウントの **Administrator-Created-ACL** は、1 つの **Common-Rule-Group** を参照しています。このため、今後 Firewall Manager 管理者アカウントによって **Common-Rule-Group** が変更されると、各メンバーアカウントに直ちに反映されます。

メンバーアカウントでは、**Common-Rule-Group** に指定されている共通ルールを変更または削除することはできません。

アカウント固有のルールは、他のアカウントに影響しません。

ウェブ ACL トラフィック情報のログ記録

> **Note**
>
> これは AWS WAF Classic ドキュメントです。このバージョンは、2019 年 11 月 AWS WAF より前にルールやウェブ ACLs などのリソースを作成 AWS WAF しており、まだ最新バージョンに移行していない場合にのみ使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては、 AWS WAF「」を参照してくださいAWS WAF。

> **Note**
>
> Amazon Security Lake を使用して AWS WAF Classic データを収集することはできません。

ログ記録を有効にして、ウェブ ACL で分析されるトラフィックに関する詳細情報を取得できます。ログに含まれる情報には、 AWS WAF Classic が AWS リソースからリクエストを受信した時間、リクエストに関する詳細情報、および各リクエストが一致したルールのアクションが含まれます。

開始するには、Amazon Kinesis Data Firehose をセットアップします。このプロセスの一環として、ログの保存先を選択します。次に、ログ記録を有効にするウェブ ACL を選択します。ログ記録を有効にすると、 は Firehose を介してストレージ宛先にログを AWS WAF 配信します。

Amazon Kinesis Data Firehose を作成して保存されたログを確認する方法については、「Amazon Data Firehose とは」を参照してください。Kinesis Data Firehose の設定に必要な許可を理解するには、「Controlling Access with Amazon Kinesis Data Firehose」(Amazon Kinesis Data Firehose によるアクセスの制御) を参照してください。

ログ記録を正常に有効化するには、次の許可がある必要があります。

- `iam:CreateServiceLinkedRole`
- `firehose:ListDeliveryStreams`
- `waf:PutLoggingConfiguration`

サービスにリンクされたロールおよび `iam:CreateServiceLinkedRole` 許可の詳細については、「Classic でのサービスにリンクされたロールの使用 AWS WAF」を参照してください。

ウェブ ACL でログ記録を有効にするには

1. プレフィックスaws-waf-logs「-」で始まる名前を使用して Amazon Kinesis Data Firehose を作成します。例えば、 ですaws-waf-logs-us-east-2-analytics。PUT ソースを使用して、動作しているリージョンでデータ Firehose を作成します。Amazon のログをキャプチャする場合は CloudFront、米国東部 (バージニア北部) に Firehose を作成します。詳細については、「Creating an Amazon Data Firehose Delivery Stream」を参照してください。

   > **Important**
   >
   > ソースとして Kinesis stream を選択しないでください。
   > 1 つの AWS WAF Classic ログは、1 つの Firehose レコードに相当します。通常、1 秒あたり 10,000 件のリクエストを受信し、フルログを有効にする場合は、Firehose で 1 秒あたり 10,000 件のレコードを設定する必要があります。Firehose を正しく設定しないと、 AWS WAF Classic はすべてのログを記録しません。詳細については、「Amazon Kinesis Data Firehose Quotas」(Amazon Kinesis Data Firehose のクォータ) を参照してください。

2. にサインイン AWS Management Console し、https://console.aws.amazon.com/wafv2/ で AWS WAF コンソールを開きます。

   ナビゲーションペインに AWS WAF 「クラシックに切り替える」と表示されている場合は、それを選択します。

3. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。
4. ログ記録を有効にするウェブ ACL の名前を選択します。これにより、右ペインで、ウェブ ACL の詳細を含むページが開きます。
5. [Logging] (ログ記録) タブで [Enable logging] (ログの有効化) を選択します。
6. 最初のステップで作成した Kinesis Data Firehose を選択します。aws-waf-logs「-」で始まる Firehose を選択する必要があります。
7. (オプション) 特定のフィールドとその値がログに含まれることを希望しない場合には、このフィールドをマスキングします。マスキングするフィールドを選び、[Add] (追加) を選択します。必要に応じて手順を繰り返し、追加のフィールドをマスキングします。マスキングされたフィールドは、ログに REDACTED と表示されます。例えば、[cookie] フィールドをマスキングした場合、ログ内の [cookie] フィールドは REDACTED となります。
8. [Enable logging] (ログの有効化) を選択します。

> **Note**
>
> ログ記録を正常に有効にすると、 AWS WAF Classic は Amazon Kinesis Data Firehose にログを書き込むために必要なアクセス許可を持つサービスにリンクされたロールを作成します。詳細については、「Classic でのサービスにリンクされたロールの使用 AWS WAF」を参照してください。

ウェブ ACL でログ記録を無効にするには

1. ナビゲーションペインで [Web ACLs] (ウェブ ACL) を選択します。
2. ログ記録を無効にするウェブ ACL の名前を選択します。これにより、右ペインで、ウェブ ACL の詳細を含むページが開きます。
3. [Logging] (ログ記録) タブで [Disable logging] (ログの無効化) を選択します。
4. 確認ダイアログボックスで、[Disable logging] (ログの無効化) を選択します。

Example ログの例

```
{

 "timestamp":1533689070589,
 "formatVersion":1,
 "webaclId":"385cb038-3a6f-4f2f-ac64-09ab912af590",
 "terminatingRuleId":"Default_Action",
 "terminatingRuleType":"REGULAR",
 "action":"ALLOW",
 "httpSourceName":"CF",
 "httpSourceId":"i-123",
 "ruleGroupList":[
                        {
                         "ruleGroupId":"41f4eb08-4e1b-2985-92b5-e8abf434fad3",
                         "terminatingRule":null,
                         "nonTerminatingMatchingRules":[
                                                         {"action" : "COUNT",
                                                         "ruleId" :
 "4659b169-2083-4a91-bbd4-08851a9aaf74"}
                                                        ],
                         "excludedRules":               [
```

```
                                                          {"exclusionType" :
 "EXCLUDED_AS_COUNT",
                                                           "ruleId" :
 "5432a230-0113-5b83-bbb2-89375c5bfa98"}
                                                         ]
                          }
                         ],

 "rateBasedRuleList":[
                              {
                               "rateBasedRuleId":"7c968ef6-32ec-4fee-96cc-51198e412e7f",

                               "limitKey":"IP",
                               "maxRateAllowed":100

                              },
                              {
                               "rateBasedRuleId":"462b169-2083-4a93-bbd4-08851a9aaf30",
                               "limitKey":"IP",
                               "maxRateAllowed":100
                               }
                               ],

 "nonTerminatingMatchingRules":[
                                       {"action" : "COUNT",

                                       "ruleId" : "4659b181-2011-4a91-
bbd4-08851a9aaf52"}
                                      ],

 "httpRequest":{
                       "clientIp":"192.10.23.23",

                       "country":"US",

                       "headers":[

                                  {
                                   "name":"Host",
                                   "value":"127.0.0.1:1989"
                                  },
                                  {
                                   "name":"User-Agent",
                                   "value":"curl/7.51.2"
```

```
                                    },
                                    {
                                     "name":"Accept",
                                     "value":"*/*"
                                    }
                                  ],
                        "uri":"REDACTED",
                        "args":"usernam=abc",
                        "httpVersion":"HTTP/1.1",
                        "httpMethod":"GET",
                        "requestId":"cloud front Request id"
                        }
}
```

以下ではこれらのログに示されている各項目について説明しています。

timestamp

タイムスタンプ (ミリ秒単位)。

formatVersion

ログの形式バージョン。

webaclId

ウェブ ACL の GUID。

terminatingRuleId

リクエストを終了したルールの ID。リクエストを終了したものがない場合、この値は `Default_Action` となります。

terminatingRuleType

リクエストを終了したルールのタイプ。可能な値は、RATE_BASED、REGULAR、GROUP です。

action

アクション。終了ルールの可能な値は、ALLOW および BLOCK です。COUNT は、終了ルールの有効な値ではありません。

terminatingRuleMatch詳細

リクエストに一致した終了ルールに関する詳細情報。終了ルールには、ウェブリクエストに対する検査プロセスを終了するアクションがあります。終了ルールに対して実行できるアクション

は、ALLOW および BLOCK です。これは、SQL インジェクションおよびクロスサイトスクリプティング (XSS) 一致ルールステートメントに対してのみ設定されます。複数の対象を検査するすべてのルールステートメントと同様、 AWS WAF は最初の一致にアクションを適用し、ウェブリクエストの検査を停止します。終了アクションを伴うウェブリクエストには、ログで報告された脅威に加えて、他の脅威が含まれている可能性があります。

httpSourceName

リクエストの送信元。指定できる値: CF (ソースが Amazon の場合 CloudFront)、APIGW (ソースが Amazon API Gateway の場合)、ALB (ソースが Application Load Balancer の場合)。

httpSourceId

ソース ID。このフィールドには、関連付けられた Amazon CloudFront ディストリビューションの ID、API Gateway の REST API、または Application Load Balancer の名前が表示されます。

ruleGroupList

このリクエストで動作したルールグループのリスト。前述のコード例では、1 つのみです。

ruleGroupId

ルールグループの ID ルールがリクエストをブロックした場合、`ruleGroupID` の ID は、`terminatingRuleId` の ID と同じです。

terminatingRule

リクエストを終了したルールグループ内のルール。これが Null 以外の値の場合、[ruleid] および [action] (アクション) も含まれます。この場合、アクションは常に BLOCK です。

nonTerminatingMatchingルール

リクエストに一致するルールグループ内のルールのリスト。これは常に COUNT ルール (一致する非終了ルール) です。

アクション (nonTerminatingMatchingルールグループ)

これは常に COUNT (一致する非終了ルール) です。

ruleId (nonTerminatingMatchingルールグループ)

リクエストに一致する非終了ルールグループ内のルールの ID。これが COUNT ルールです。

excludedRules

除外されているルールグループ内のルールのリスト。これらのルールのアクションは [COUNT] に設定されます。

exclusionType (excludedRules group)

除外されたルールにアクション COUNT があることを示すタイプ。

ruleId (excludedRules group)

除外されたルールグループ内のルールの ID。

rateBasedRuleリスト

このリクエストで動作したレートベースのルールのリスト。

rateBasedRuleID

このリクエストで動作したレートベースのルールの ID。これがリクエストを終了した場合、`rateBasedRuleId` の ID は、`terminatingRuleId` の ID と同じです。

limitKey

が AWS WAF 1 つのソースからリクエストが到着する可能性が高いかどうかを判断するために使用するフィールド。したがって、レートモニタリングの対象となります。可能性のある値は IP です。

maxRateAllowed

5 分間に許可される `limitKey` で指定されたフィールドに同じ値を持つ、リクエストの最大数。リクエストの数が を超え`maxRateAllowed`、ルールで指定された他の述語も満たされた場合、はこのルールに指定されたアクションを AWS WAF トリガーします。

httpRequest

リクエストに関するメタデータです。

clientIp

リクエストを送信するクライアントの IP アドレス。

country

リクエストの送信国。 AWS WAF が発信元の国を特定できない場合、このフィールドは に設定されます-。

headers

ヘッダーの一覧。

uri

リクエストの URI。上記のコードの例では、このフィールドがマスキングされた場合の値を示しています。

args

クエリ文字列。

httpVersion

HTTP のバージョン。

httpMethod

リクエストの HTTP メソッド。

requestId

リクエストの ID。

レートベースのルールごとにブロックされている IP アドレスの一覧表示

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS WAF Classic には、レートベースのルールによってブロックされる IP アドレスのリストが表示されます。

レートベースのルールごとにブロックされているアドレスを表示するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ AWS WAF のコンソールを開きます。

   ナビゲーションペインに [AWS WAF クラシックに切り替え] が表示されている場合は、それを選択します。
2. ナビゲーションペインで [Rules] (ルール) を選択します。
3. [Name] (名前) 列で、レートベースのルールを選択します。

一覧には、ルールによって現在ブロックされている IP アドレスが表示されます。

AWS WAF クラシックと Amazon CloudFront の機能との連携

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

ウェブ ACL を作成するときに、 AWS WAF Classic CloudFront に検査させたいディストリビューションを 1 つ以上指定できます。 AWS WAF Classic は、ウェブ ACL で指定した条件に基づいて、それらのディストリビューションのウェブリクエストを許可、ブロック、またはカウントし始めます。 CloudFront には、 AWS WAF Classic の機能を強化する機能がいくつか用意されています。この章では、 AWS WAF Classic CloudFront CloudFront とをうまく連携させるために設定できるいくつかの方法について説明します。

トピック

- AWS WAF Classic CloudFront とカスタムエラーページとの併用
- CloudFront独自の HTTP サーバー上で動作するアプリケーションに AWS WAF Classic with を使用する
- CloudFrontに応答する HTTP メソッドの選択

AWS WAF Classic CloudFront とカスタムエラーページとの併用

AWS WAF Classic は、指定した条件に基づいてウェブリクエストをブロックすると、HTTP ステータスコード 403 (禁止) CloudFront をに返します。次に、 CloudFront そのステータスコードをビューアに返します。ビューワーには、以下のような簡潔で特に書式設定されていないデフォルトメッセージが表示されます。

```
Forbidden: You don't have permission to access /myfilename.html on this
server.
```

Web サイトの他の部分と同じ形式を使用してカスタムエラーメッセージを表示したい場合は、カスタムエラーメッセージを含むオブジェクト (HTML ファイルなど) CloudFront をビューアに返すように設定できます。

> **Note**
>
> CloudFront オリジンから返される HTTP ステータスコード 403 と、リクエストがブロックされたときに AWS WAF Classic から返される HTTP ステータスコード 403 を区別できない。つまり、HTTP ステータスコード 403 のさまざまな原因に基づいて、異なるカスタムエラーページを返すことはできません。

CloudFront カスタムエラーページの詳細については、Amazon CloudFront 開発者ガイドの「エラーレスポンスのカスタマイズ」を参照してください。

CloudFront独自の HTTP サーバー上で動作するアプリケーションに AWS WAF Classic with を使用する

AWS WAF クラシックをと共に使用すると CloudFront、Amazon Elastic Compute Cloud (Amazon EC2) で実行されているウェブサーバーでも、プライベートに管理されているウェブサーバーでも、任意の HTTP ウェブサーバーで実行されているアプリケーションを保護できます。また、CloudFrontと自分のウェブサーバー間、およびビューワーとの間で HTTPS CloudFront を要求するように設定することもできます。 CloudFront

CloudFront と自分のウェブサーバーとの間で HTTPS を要求する

CloudFront 独自のウェブサーバーとの間で HTTPS を要求するには、 CloudFront カスタムオリジン機能を使用して、特定のオリジンのオリジンプロトコルポリシーとオリジンドメイン名の設定を構成できます。 CloudFront 構成では、 CloudFront オリジンからオブジェクトを取得するときに使用するポートとプロトコルとともに、サーバーの DNS 名を指定できます。また、カスタムオリジンサーバー上の SSL/TLS 証明書が、設定したオリジンドメイン名と一致することを確認する必要もあります。外部で独自の HTTP Web サーバーを使用する場合は AWS、Comodo、 DigiCertまたは Symantec などの信頼できるサードパーティの認証機関 (CA) によって署名された証明書を使用する必要があります。 CloudFrontと独自のウェブサーバー間の通信に HTTPS を要求する方法の詳細については、Amazon CloudFront 開発者ガイドの「 CloudFront とカスタムオリジン間の通信に HTTPS を要求する」を参照してください。

ビューアーと間の HTTPS の要求 CloudFront

ビューアとの間で HTTPS を要求するには CloudFront、 CloudFrontディストリビューション内の 1 つ以上のキャッシュ動作の Viewer プロトコルポリシーを変更できます。 CloudFront視聴者間でのHTTPSの使用の詳細については CloudFront、 CloudFront Amazon開発者ガイドのトピック「視聴者間の通信にHTTPSを要求する」を参照してください。視聴者が独自のドメイン名 (例:https://www.mysite.com) を使用して HTTPS CloudFront 経由でディストリビューションに接続できるように、独自の SSL 証明書を持ち込むこともできます。詳細については、Amazon CloudFront 開発者ガイドのトピック「代替ドメイン名と HTTPS の設定」を参照してください。

CloudFrontに応答する HTTP メソッドの選択

Amazon CloudFront ウェブディストリビューションを作成するときは、 CloudFront 処理してオリジンに転送する HTTP メソッドを選択します。次のオプションから選択できます。

- GET, HEAD — オリジンからのオブジェクトの取得、 CloudFront またはオブジェクトヘッダーの取得にのみ使用できます。
- GET、HEAD、OPTIONS — オリジンからのオブジェクトの取得、オブジェクトヘッダーの取得、CloudFront またはオリジンサーバーがサポートするオプションのリストの取得にのみ使用できます。
- GET、HEAD、OPTIONS、PUT、POST、PATCH、DELETE — オブジェクトを取得、追加、更新、削除、およびオブジェクトヘッダーの取得に使用できます CloudFront 。また、ウェブフォームからのデータの送信など、その他の POST 操作も実行できます。

で説明されているように、 AWS WAF クラシック文字列一致条件を使用して HTTP メソッドに基づくリクエストを許可または拒否することもできます文字列一致条件の使用。 CloudFront やなどをサポートするメソッドを組み合わせて使用する場合は、他のメソッドを使用するリクエストをブロックするように AWS WAF Classic を設定する必要はありません。GET HEAD、、など、 CloudFront サポートされていないメソッドの組み合わせを許可したい場合はGETHEAD、 CloudFront すべてのメソッドに応答するように設定し、 AWS WAF クラシックを使用して他のメソッドを使用するリクエストをブロックできます。POST

CloudFront 応答するメソッドの選択の詳細については、Amazon CloudFront 開発者ガイドの「ウェブディストリビューションを作成または更新するときに指定する値」の「許可される HTTP メソッド」を参照してください。

AWS WAF クラシックのセキュリティ

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS クラウドセキュリティは最優先事項です。 AWS 顧客は、最もセキュリティに敏感な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の恩恵を受けることができます。

セキュリティは、 AWS お客様とお客様との間で共有される責任です。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して説明しています。

- クラウドのセキュリティ — AWS AWS AWS クラウドクラウド内でサービスを実行するインフラストラクチャを保護する責任があります。 AWS また、安全に使用できるサービスも提供します。セキュリティの有効性は、AWS コンプライアンスプログラムの一環として、サードパーティーの審査機関によって定期的にテストおよび検証されています。 AWS WAF Classic に適用されるコンプライアンスプログラムについては、「AWS コンプライアンスプログラム別の対象サービス」を参照してください。
- クラウドにおけるセキュリティ — お客様の責任は、 AWS 使用する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、組織の要件、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、 AWS WAF Classic を使用する際に責任分担モデルを適用する方法を理解するのに役立ちます。以下のトピックでは、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 AWS WAF Classic を構成する方法を示しています。また、 AWS WAF Classic AWS リソースの監視と保護に役立つ他のサービスの使い方についても学びます。

トピック

- AWS WAF Classic のコンプライアンス検証
- レジリエンス・イン・クラシック AWS WAF
- AWS WAF Classic のインフラストラクチャーセキュリティ

AWS WAF クラシックでのデータ保護

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールや Web ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS AWS WAF Classic のデータ保護には、。このモデルで説明したように、 AWS は、 AWS クラウドすべてを支え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。お客様は、このインフラストラクチャでホストされてい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、「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。欧州でのデータ保護の詳細については、「AWS セキュリティブログ」に投稿された「AWS 責任共有モデルおよび GDPR」のブログ記事を参照してください。

データ保護のため、 AWS アカウント 認証情報を保護し、 AWS IAM Identity Center 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。こうすると、それぞれのジョブを遂行するために必要なアクセス許可のみを各ユーザーに付与できます。また、以下の方法でデータを保護することをお勧めします。

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用してリソースと通信します。 AWS TLS 1.2、できれば TLS 1.3 が必要です。
- を使用して API とユーザーアクティビティの口ギングを設定します。 AWS CloudTrail
- AWS 暗号化ソリューションと、 AWS のサービスその中に含まれるデフォルトのセキュリティコントロールをすべて使用してください。
- Amazon Macie などの高度なマネージド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 を介してアクセスするときに FIPS 140-2 で検証された暗号モジュールが必要な場合は、FIPS エンドポイントを使用してください。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) 140-2」を参照してください。

お客様の E メールアドレスなどの機密情報やセンシティブ情報は、タグや名前フィールドなどの自由形式のフィールドに配置しないことを強くお勧めします。これには、コンソール、API、または SDK AWS のサービス を使用して AWS WAF Classic やその他のアプリケーションを操作する場合も含まれます。 AWS CLI AWS 名前に使用する自由記述のテキストフィールドやタグに入力したデータは、課金や診断ログに使用される場合があります。外部サーバーへの URL を提供する場合は、そのサーバーへのリクエストを検証するための認証情報を URL に含めないように強くお勧めします。

AWS WAF ウェブ ACL、ルール、条件などのクラシックエンティティは、中国 (北京) や中国 (寧夏) など、暗号化が利用できない特定の地域を除き、保存時に暗号化されます。リージョンごとに一意の暗号化キーが使用されます。

AWS WAF クラシックリソースの削除

AWS WAF Classic で作成したリソースは削除できます。次のセクションの各リソースタイプのガイダンスを参照してください。

- ウェブ ACL の削除
- AWS WAF クラシックルールグループからのルールの追加と削除
- ルールの削除

AWS WAF Classic の Identity and Access Management

> **Note**
>
> これは AWS WAF Classic ドキュメントです。このバージョンは、2019 年 11 月 AWS WAF より前にルールやウェブ ACLs などのリソースを作成し AWS WAF 、まだ最新バージョンに移行していない場合にのみ使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては、 AWS WAF「」を参照してくださいAWS WAF。

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 (サインイン) し、誰に AWS WAF Classic リソースの使用を承認する (アクセス許可を付与する) かを制御します。IAM は、追加料金なしで AWS のサービス 使用できる です。

トピック

対象者

AWS Identity and Access Management (IAM) の使用方法は、 AWS WAF Classic で行う作業によって異なります。

サービスユーザー – AWS WAF Classic サービスを使用してジョブを実行する場合、管理者から必要な認証情報とアクセス許可が与えられます。さらに多くの AWS WAF Classic 機能を使用して作業を行う場合は、追加のアクセス許可が必要になることがあります。アクセスの管理方法を理解しておくと、管理者に適切な許可をリクエストするうえで役立ちます。 AWS WAF Classic の特徴にアクセスできない場合は、「AWS WAF クラシック ID 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。

サービス管理者 – 社内の AWS WAF Classic リソースを担当している場合は、通常、 AWS WAF Classic へのフルアクセスがあります。サービスユーザーがどの AWS WAF Classic 機能やリソースにアクセスするかを決めるのは管理者の仕事です。その後、IAM 管理者にリクエストを送信して、サービスユーザーの権限を変更する必要があります。このページの情報を点検して、IAM の基本概念を理解してください。会社で AWS WAF Classic で IAM を使用する方法の詳細については、「」を参照してくださいAWS WAF Classic と IAM の連携方法。

IAM 管理者 – IAM 管理者は、 AWS WAF Classic へのアクセスを管理するポリシーの作成方法の詳細について確認する場合があります。IAM で使用できる AWS WAF Classic 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、「」を参照してくださいAWS WAF Classic のアイデンティティベースのポリシーの例。

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。として、IAM ユーザーとして AWS アカウントのルートユーザー、または IAM ロールを引き受けて認証 (にサインイン AWS) される必要があります。

ID ソースを介して提供された認証情報を使用して、フェデレーティッド ID AWS として にサインインできます。 AWS IAM Identity Center （ IAM Identity Center) ユーザー、会社のシングルサインオン認証、Google または Facebook の認証情報は、フェデレーティッド ID の例です。フェデレーティッド ID としてサインインする場合、IAM ロールを使用して、前もって管理者により ID フェデレーションが設定されています。フェデレーション AWS を使用して にアクセスすると、間接的にロールを引き受けることになります。

ユーザーのタイプに応じて、 AWS Management Console または AWS アクセスポータルにサインインできます。へのサインインの詳細については AWS、「 ユーザーガイド」の「 へのサインイン AWS アカウント方法AWS サインイン 」を参照してください。

AWS プログラムで にアクセスする場合、 は Software Development Kit (SDK) とコマンドラインインターフェイス (CLI) AWS を提供し、認証情報を使用してリクエストに暗号で署名します。 AWS ツールを使用しない場合は、リクエストに自分で署名する必要があります。推奨される方法を使用してリクエストを自分で署名する方法の詳細については、IAM ユーザーガイドの API AWS リクエストの署名を参照してください。

使用する認証方法を問わず、追加セキュリティ情報の提供をリクエストされる場合もあります。例えば、 AWS では、アカウントのセキュリティを強化するために多要素認証 (MFA) を使用することをお勧めします。詳細については、「AWS IAM Identity Center ユーザーガイド」の「Multi-factor authentication」(多要素認証) および「IAM ユーザーガイド」の「AWSでの多要素認証 (MFA) の使用」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、アカウント内のすべての 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 1 つのサインインアイデンティティから始めます。この ID は AWS アカウント ルートユーザーと呼ばれ、アカウントの作成に使用した E メールアドレスとパスワードでサインインすることでアクセスできます。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。ルートユーザーの認証情報は保護し、ルートユーザーでしか実行できないタスクを実行するときに使用します。ルートユーザーとしてサインインする必要があるタスクの完全なリストについては、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。

フェデレーティッドアイデンティティ

ベストプラクティスとして、管理者アクセスを必要とするユーザーを含む人間のユーザーに、一時的な認証情報を使用して にアクセスするための ID プロバイダーとのフェデレーションの使用を要求 AWS のサービス します。

フェデレーティッド ID は、エンタープライズユーザーディレクトリ、ウェブ ID プロバイダー、AWS Directory Service、アイデンティティセンターディレクトリのユーザー、または ID ソースを通じて提供された認証情報 AWS のサービス を使用して にアクセスするユーザーです。フェデレーティッド ID が にアクセスすると AWS アカウント、ロールが引き受けられ、ロールは一時的な認証情報を提供します。

アクセスを一元管理する場合は、 AWS IAM Identity Centerを使用することをお勧めします。IAM Identity Center でユーザーとグループを作成することも、独自の ID ソース内のユーザーとグループのセットに接続して同期して、すべての AWS アカウント とアプリケーションで使用することもできます。IAM Identity Center の詳細については、「AWS IAM Identity Center ユーザーガイド」の「What is IAM Identity Center?」(IAM Identity Center 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、単一のユーザーまたはアプリケーションに対して特定のアクセス許可 AWS アカウント を持つ 内のアイデンティティです。可能であれば、パスワードやアクセスキーなどの長期的な認証情報を保有する IAM ユーザーを作成する代わりに、一時的な認証情報を使用することをお勧めします。ただし、IAM ユーザーでの長期的な認証情報が必要な特定のユースケースがある場合は、アクセスキーをローテーションすることをお勧めします。詳細については、IAM ユーザーガイドの長期的な認証情報を必要とするユースケースのためにアクセスキーを定期的にローテーションするを参照してください。

IAM グループは、IAM ユーザーの集団を指定するアイデンティティです。グループとしてサインインすることはできません。グループを使用して、複数のユーザーに対して一度に権限を指定できます。多数のユーザーグループがある場合、グループを使用することで権限の管理が容易になります。例えば、IAMAdminsという名前のグループを設定して、そのグループにIAM リソースを管理する許可を与えることができます。

ユーザーは、ロールとは異なります。ユーザーは 1 人の人または 1 つのアプリケーションに一意に関連付けられますが、ロールはそれを必要とする任意の人が引き受けるようになっています。ユーザーには永続的な長期の認証情報がありますが、ロールでは一時的な認証情報が提供されます。詳細については、「IAM ユーザーガイド」の「IAM ユーザー (ロールではなく) の作成が適している場合」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 AWS アカウント を持つ 内のアイデンティティです。これは IAM ユーザーに似ていますが、特定のユーザーには関連付けられていません。ロール を切り替える AWS Management Console ことで、 で IAM ロールを一時的に引き受けることができます。ロールを引き受けるには、 または AWS API AWS CLI オペレーションを呼び出すか、カスタム URL を使用します。ロールを使用する方法の詳細については、「IAM ユーザーガイド」の「IAM ロールの使用」を参照してください。

IAM ロールと一時的な認証情報は、次の状況で役立ちます:

- フェデレーションユーザーアクセス – フェデレーティッド ID に許可を割り当てるには、ロールを作成してそのロールの許可を定義します。フェデレーティッド ID が認証されると、その ID はロールに関連付けられ、ロールで定義されている許可が付与されます。フェデレーションの詳細については、「IAM ユーザーガイド」の「Creating a role for a third-party Identity Provider」(サードパーティーアイデンティティプロバイダー向けロールの作成) を参照してください。IAM Identity Center を使用する場合は、許可セットを設定します。アイデンティティが認証後にアクセスできるものを制御するため、IAM Identity Center は、権限セットを IAM のロールに関連付けます。アクセス許可セットの詳細については、「AWS IAM Identity Center ユーザーガイド」の「アクセス許可セット」を参照してください。
- 一時的な IAM ユーザー権限 - IAM ユーザーまたはロールは、特定のタスクに対して複数の異なる権限を一時的に IAM ロールで引き受けることができます。
- クロスアカウントアクセス - IAM ロールを使用して、自分のアカウントのリソースにアクセスすることを、別のアカウントの人物 (信頼済みプリンシパル) に許可できます。クロスアカウントアクセス権を付与する主な方法は、ロールを使用することです。ただし、一部の では AWS のサービス、 (ロールをプロキシとして使用する代わりに) ポリシーをリソースに直接アタッチできます。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。
- クロスサービスアクセス — 一部の は、他の の機能 AWS のサービス を使用します AWS のサービス。例えば、あるサービスで呼び出しを行うと、通常そのサービスによって Amazon EC2 でアプリケーションが実行されたり、Amazon S3 にオブジェクトが保存されたりします。サービスでは、呼び出し元プリンシパルの許可、サービスロール、またはサービスリンクロールを使用してこれを行う場合があります。
  - 転送アクセスセッション (FAS) — IAM ユーザーまたはロールを使用して でアクションを実行する場合 AWS、ユーザーはプリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがありま

す。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストのリクエストと組み合わせて使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。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。

- サービスロール - 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 のサービスにアクセス許可を委任するロールの作成」を参照してください。
- サービスにリンクされたロール – 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、サービスにリンクされた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。

- Amazon EC2 で実行されているアプリケーション – IAM ロールを使用して、EC2 インスタンスで実行され、 AWS CLI または AWS API リクエストを行うアプリケーションの一時的な認証情報を管理できます。これは、EC2 インスタンス内でのアクセスキーの保存に推奨されます。 AWS ロールを EC2 インスタンスに割り当て、そのすべてのアプリケーションで使用できるようにするには、インスタンスにアタッチされたインスタンスプロファイルを作成します。インスタンスプロファイルにはロールが含まれ、EC2 インスタンスで実行されるプログラムは一時的な認証情報を取得できます。詳細については、IAM ユーザーガイドのAmazon EC2 インスタンスで実行されるアプリケーションに IAM ロールを使用して許可を付与するを参照してください。

IAM ロールと IAM ユーザーのどちらを使用するかについては、IAM ユーザーガイドの(IAM ユーザーではなく) IAM ロールをいつ作成したら良いのか?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、 AWS ID 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ときにアクセス許可を定義する のオブジェクトです。 は、プリンシパル (ユーザー、ルートユーザー、またはロールセッション) AWS 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。ポリシーでの権限により、リクエストが許可されるか拒否されるかが決まります。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構造と内容の詳細については、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 AWS JSON 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んなリソースにどんな条件でアクションを実行できるかということです。

デフォルトでは、ユーザーやロールに権限はありません。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。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き継ぐことができます。

IAM ポリシーは、オペレーションの実行方法を問わず、アクションの許可を定義します。例えば、`iam:GetRole`アクションを許可するポリシーがあるとします。そのポリシーを持つユーザーは、 AWS Management Console、、 AWS CLIまたは AWS API からロール情報を取得できます。

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ポリシーは、IAM ユーザーグループ、ユーザーのグループ、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 ポリシーの作成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、さらにインラインポリシーまたはマネージドポリシーに分類できます。インラインポリシーは、単一のユーザー、グループ、またはロールに直接埋め込まれています。管理ポリシーは、 内の複数のユーザー、グループ、ロールにアタッチできるスタンドアロンポリシーです AWS アカウント。管理ポリシーには、 AWS 管理ポリシーとカスタマー管理ポリシーが含まれます。マネージドポリシーまたは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 ユーザーガイドのマネージドポリシーとインラインポリシーの比較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。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制御できます。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は、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。

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。

アクセスコントロールリスト (ACL)

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。

Amazon S3、 AWS WAF、および Amazon VPC は、ACLs。ACL の詳細については、Amazon Simple Storage Service デベロッパ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。

その他のポリシータイプ

AWS は、一般的ではない追加のポリシータイプをサポートします。これらのポリシータイプでは、より一般的なポリシータイプで付与された最大の権限を設定できます。

- アクセス許可の境界 - アクセス許可の境界は、アイデンティティベースのポリシーによって IAM エンティティ (IAM ユーザーまたはロール) に付与できる権限の上限を設定する高度な機能です。エンティティにアクセス許可の境界を設定できます。結果として得られる権限は、エンティティのアイデンティティベースポリシーとそのアクセス許可の境界の共通部分になります。`Principal` フィールドでユーザーまたはロールを指定するリソースベースのポリシーでは、アクセス許可の境界は制限されません。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。アクセス許可の境界の詳細については、IAM ユーザーガイドのIAM エンティティのアクセス許可の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 (SCPs) – SCPs は、 の組織または組織単位 (OU) に対する最大アクセス許可を指定する JSON ポリシーです AWS Organizations。 AWS Organizations は、 AWS アカウント ビジネスが所有する複数の をグループ化して一元管理するサービスです。組織内のすべての機能を有効にすると、サービスコントロールポリシー (SCP) を一部またはすべてのアカウントに適用できます。SCP は、各 を含むメンバーアカウントのエンティティのアクセス許可を制限します AWS アカウントのルートユーザー。Organizations と SCP の詳細については、AWS Organizations ユーザーガイドの「SCP の仕組み」を参照してください。
- セッションポリシー - セッションポリシーは、ロールまたはフェデレーションユーザーの一時的なセッションをプログラムで作成する際にパラメータとして渡す高度なポリシーです。結果としてセッションの権限は、ユーザーまたはロールのアイデンティティベースポリシーとセッションポリシーの共通部分になります。また、リソースベースのポリシーから権限が派生する場合もあります。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。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権限を理解するのがさらに難しくなります。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。

AWS WAF Classic と IAM の連携方法

> **Note**
>
> これは AWS WAF Classic ドキュメントです。このバージョンは、2019 年 11 月 AWS WAF より前にルールやウェブ ACLs などのリソースを作成し AWS WAF 、まだ最新バージョンに移行していない場合にのみ使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては、 AWS WAF「」を参照してくださいAWS WAF。

IAM を使用して AWS WAF Classic へのアクセスを管理する前に、Classic で使用できる IAM AWS WAF 機能について学びます。

AWS WAF Classic で使用できる IAM の機能

IAM 機能	AWS WAF クラシックサポート
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	はい
ポリシー条件キー (サービス固有)	はい
ACL	なし
ABAC (ポリシー内のタグ)	部分的
一時的な認証情報	あり

IAM 機能	AWS WAF クラシックサポート
転送アクセスセッション (FAS)	あり
サービスロール	あり
サービスにリンクされたロール	あり

AWS WAF Classic 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を把握するには、「IAM ユーザーガイド」の「IAM AWS と連携する のサービス」を参照してください。

AWS WAF Classic 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ポリシーをサポートする	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 ポリシーの作成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。プリンシパルは、それが添付されているユーザーまたはロールに適用されるため、アイデンティティベースのポリシーでは指定できません。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。

AWS WAF Classic 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、「」を参照してくださいAWS WAF Classic のアイデンティティベースのポリシーの例。

AWS WAF Classic 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制御できます。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は、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、アカウント全体、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。リソースベースのポリシーにクロスアカウントのプリンシパルを追加しても、信頼関係は半分しか確立されない点に注意してください。プリンシパルとリソースが異なる にある場合 AWS アカウント、信頼されたアカウントの IAM 管理者は、プリンシパルエンティティ (ユーザーまたはロール) にリソースへのアクセス許可も付与する必要があります。IAM 管理者は、アイデンティティベースのポリシーをエンティティにアタッチすることで権限を付与します。ただし、リソースベースのポリシーで、同じアカウントのプリンシパルへのアクセス権が付与されている場合は、アイデンティティベースのポリシーをさらに付与する必要はありません。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。

AWS WAF Classic のポリシーアクション

ポリシーアクションに対するサポート	あり

管理者は AWS JSON 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んなリソースにどんな条件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。ポリシーアクションの名前は通常、関連付けられた AWS API オペレーションと同じです。一致する API オペレーションのない許可のみのアクションなど、いくつかの例外があります。また、ポリシーに複数のアクションが必要なオペレーションもあります。これらの追加アクションは、依存アクションと呼ばれます。

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権限を付与するポリシーで使用されます。

AWS WAF Classic アクションのリストを確認するには、「サービス認証リファレンス」の「 で定義されるアクション AWS WAF」およびAWS WAF 「リージョン で定義されるアクション」を参照してください。

AWS WAF Classic 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。

```
waf
```

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": [
      "waf:action1",
      "waf:action2"
         ]
```

ワイルドカード (*) を使用すると、複数のアクションを指定することができます。例えば、 で始まる AWS WAF Classic のすべてのアクションを指定するにはList、次のアクションを含めます。

```
"Action": "waf:List*"
```

AWS WAF Classic 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、「」を参照してくださいAWS WAF Classic のアイデンティティベースのポリシーの例。

AWS WAF Classic のポリシーリソース

ポリシーリソースに対するサポート	あり

管理者は AWS JSON 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どのような条件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。ステートメントには、Resource または NotResource 要素を含める必要があります。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。これは、リソースレベルの許可と呼ばれる特定のリソースタイプをサポートするアクションに対して実行できます。

オペレーションのリスト化など、リソースレベルの権限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": "*"
```

AWS WAF Classic リソースタイプとその ARNs」およびAWS WAF 「リージョン で定義されるリソース」を参照してください。 AWS WAF 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は、「 で定義されるアクション AWS WAF」およびAWS WAF 「リージョン で定義されるアクション」を参照してください。 AWS WAF Classic リソースのサブセットへのアクセスを許可または拒否するには、ポリシーの `resource`要素にリソースの ARN を含めます。

AWS WAF Classic では、リソースはウェブ ACLs とルール です。 AWS WAF Classic は、バイト一致 、IP 一致 、サイズ制約 などの条件もサポートしています。

これらのリソースと条件には、次の表に示すとおり、一意の Amazon リソースネーム (ARN) が関連付けられています。

AWS WAF コンソールの名前	AWS WAF SDK/CLI の名前	ARN 形式
ウェブ ACL	WebACL	arn:aws:waf:: *account*:*webacl*/*ID*
ルール	Rule	arn:aws:waf:: *account*:*rule*/*ID*
文字列一致条件	ByteMatchSet	arn:aws:waf:: *account*:*bytematchset* /*ID*
SQL のインジェクション一致の状態	SqlInjectionMatchSet	arn:aws:waf:: *account*:*sqlinjectionset* /*ID*
サイズ制約条件	SizeConstraintSet	arn:aws:waf:: *account*:*sizeconstraintset* /*ID*
IP 一致条件	IPSet	arn:aws:waf:: *account*:*ipset*/*ID*
クロスサイトスクリプティング一致条件	XssMatchSet	arn:aws:waf:: *account*:*xssmatchset* /*ID*

AWS WAF Classic リソースのサブセットへのアクセスを許可または拒否するには、ポリシーの resource要素にリソースの ARN を含めます。 AWS WAF Classic ARNs の形式は次のとおりです。

```
arn:aws:waf::account:resource/ID
```

[account] (アカウント)、*[resource]* (リソース)、および *[ID]* 変数を有効な値に置き換えます。有効な値は次のとおりです。

- ***#####*** : の ID AWS アカウント。値を指定する必要があります。
- ***####*** : AWS WAF Classic リソースのタイプ。
- *ID* : AWS WAF Classic リソースの ID、またはワイルドカード (*) 。指定した に関連付けられている指定したタイプのすべてのリソースを示します AWS アカウント。

例えば、次の ARN はアカウント 111122223333 のすべてのウェブ ACL を指定します。

```
arn:aws:waf::111122223333:webacl/*
```

AWS WAF Classic 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 AWS JSON 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んなリソースにどんな条件でアクションを実行できるかということです。

Condition 要素 (または Condition ブロック) を使用すると、ステートメントが有効な条件を指定できます。Condition 要素はオプションです。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することで、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。

1 つのステートメントに複数の Condition 要素を指定する場合、または 1 つの Condition 要素に複数のキーを指定する場合、 AWS では AND 論理演算子を使用してそれらを評価します。1 つの条件キーに複数の値を指定すると、 は論理ORオペレーションを使用して条件 AWS を評価します。ステートメントの権限が付与される前にすべての条件が満たされる必要があります。

条件を指定する際にプレースホルダー変数も使用できます。例えば IAM ユーザーに、IAM ユーザー名がタグ付けされている場合のみリソースにアクセスできる権限を付与することができます。詳細については、IAM ユーザーガイドのIAM ポリシーの要素: 変数およびタグを参照してください。

AWS は、グローバル条件キーとサービス固有の条件キーをサポートします。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。

AWS WAF Classic 条件キーのリストを確認するには、「サービス認証リファレンス」の「 およびリージョンで定義されるリソースの条件キー AWS WAF」を参照してください。 AWS WAF条件キーを使用できるアクションとリソースについては、「 で定義されるアクション AWS WAF」およびAWS WAF 「リージョン で定義されるアクション」を参照してください。

AWS WAF Classic 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、「」を参照してくださいAWS WAF Classic のアイデンティティベースのポリシーの例。

AWS WAF Classic ACLs

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許可を持つかをコントロールします。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。

AWS WAF Classic での ABAC

ABAC (ポリシー内のタグ) のサポート	部分的

属性ベースのアクセス制御 (ABAC) は、属性に基づいてアクセス許可を定義する認可戦略です。ではAWS、これらの属性はタグ と呼ばれます。タグは、IAM エンティティ (ユーザーまたはロール) および多くの AWS リソースにアタッチできます。エンティティとリソースのタグ付けは、ABAC の最初の手順です。その後、プリンシパルのタグがアクセスしようとしているリソースのタグと一致した場合にオペレーションを許可するように ABAC ポリシーをします。

ABAC は、急成長する環境やポリシー管理が煩雑になる状況で役立ちます。

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 とは?を参照してください。ABAC をセットアップするステップ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に基づくアクセスコントロール (ABAC) を使用するを参照してください。

AWS WAF Classic 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

一部の AWS のサービス は、一時的な認証情報を使用してサインインすると機能しません。一時的な認証情報 AWS のサービス を使用する などの詳細については、IAM ユーザーガイドのAWS のサービス 「IAM と連携する 」を参照してください。

ユーザー名とパスワード以外の AWS Management Console 方法で にサインインする場合、一時的な認証情報を使用します。例えば、会社の Single Sign-On (SSO) リンク AWS を使用して にアクセスすると、そのプロセスによって一時的な認証情報が自動的に作成されます。また、ユーザーとしてコンソールにサインインしてからロールを切り替える場合も、一時的な認証情報が自動的に作成されます。ロールの切り替えに関する詳細については、IAM ユーザーガイドのロールへの切り替え (コンソール)を参照してください。

一時的な認証情報は、 AWS CLI または AWS API を使用して手動で作成できます。その後、これらの一時的な認証情報を使用して . AWS recommends にアクセスできます AWS。これは、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することを推奨しています。詳細については、IAM の一時的セキュリティ認証情報を参照してください。

AWS WAF Classic の転送アクセスセッション

転送アクセスセッション (FAS) をサポート	あり

IAM ユーザーまたはロールを使用して でアクションを実行すると AWS、プリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストのリクエストと組み合わせ

て使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。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。

AWS WAF Classic のサービスロール

サービスロールに対するサポート	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 のサービスにアクセス許可を委任するロールの作成」を参照してください。

> **Warning**
>
> サービスロールのアクセス許可を変更すると、 AWS WAF Classic の機能が破損する可能性があります。 AWS WAF Classic 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。

AWS WAF Classic の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、サービスにリンクされた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。

AWS WAF Classic 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、「」を参照してくださいClassic でのサービスにリンクされたロールの使用 AWS WAF。

AWS WAF Classic のアイデンティティベースのポリシーの例

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

デフォルトでは、ユーザーとロールには AWS WAF Classic リソースを作成または変更する権限がありません。また、 AWS Management Console、 AWS Command Line Interface (AWS CLI)、 AWS API を使用してタスクを実行することもできません。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。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き受けることができます。

これらサンプルの 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 ポリシーの作成」を参照してください。

各リソースタイプの ARN の形式など、 AWS WAF Classic で定義されているアクションとリソースタイプの詳細については、『サービス認証リファレンス』の「AWS WAF リージョンのアクション、リソース、条件キー」を参照してください。 AWS WAF

トピック

- ポリシーのベストプラクティス
- クラシックコンソールの使用 AWS WAF
- 自分の許可の表示をユーザーに許可する

ポリシーのベストプラクティス

ID ベースのポリシーは、アカウント内の AWS WAF Classic リソースを誰かが作成、アクセス、削除できるかどうかを決定します。これらのアクションを実行すると、 AWS アカウントに料金が発生する可能性があります。アイデンティティベースの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 管理ポリシーから始めて、最小権限の権限に移行する — ユーザーとワークロードへのアクセス権限の付与を開始するには、AWS 多くの一般的なユースケースで権限を付与する管理ポリシーを使用してください。これらのポリシーは、で利用できます。 AWS アカウント AWS ユースケースに固有のカスタマー管理ポリシーを定義して、権限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「AWS ジョブ機能の管理ポリシー」を参照してください。
- 最小特権を適用する – IAM ポリシーで許可を設定するときは、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。例えば、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。サービスアクションがなどの特定の用途で使用された場合は AWS のサービス、条件を使用して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 AWS CloudFormation。詳細については、「IAM ユーザーガイド」の [IAM JSON policy elements: Condition] (IAM JSON ポリシー要素：条件) を参照してください。
- IAM Access Analyzer 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 Access Analyzer 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 ポリシーの検証」を参照してください。
- 多要素認証 (MFA) が必要 — IAM ユーザーまたは root 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、セキュリティを強化するために MFA をオンにしてください。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 保護 API アクセスの設定」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。

クラシックコンソールの使用 AWS WAF

AWS WAF Classic コンソールにアクセスするには、最低限の権限が必要です。これらの権限により、 AWS WAF 内のクラシックリソースの詳細を一覧表示して表示できる必要があります 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、最低限のコンソール権限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。

コンソールにアクセスして使用できるユーザーは、 AWS AWS WAF クラシックコンソールにもアクセスできます。追加のアクセス許可は必要ありません。

自分の許可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添付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を作成する方法を示します。このポリシーには、コンソールで、またはまたは API を使用してこのアクションをプログラム的に実行するための権限が含まれています。 AWS CLI AWS

```
{
    "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": "*"
        }
    ]
}
```

AWS WAF クラシック ID とアクセスのトラブルシューティング

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月より前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS WAF Classic と IAM を使用する際に発生する可能性のある一般的な問題の診断と修正に役立つ情報は次のとおりです。

トピック

- Classic でアクションを実行する権限がありません。 AWS WAF
- 私にはiam を実行する権限がありません:PassRole
- AWS アカウントAWS WAF 自分以外の人にもクラシックリソースへのアクセスを許可したい。

Classic でアクションを実行する権限がありません。 AWS WAF

「I am not authorized to perform an action in Amazon Bedrock」というエラーが表示された場合、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。

次の例は、mateojackson という IAM ユーザーがコンソールを使用して架空の *my-example-widget* リソースに関する詳細を表示しようとしたとき、架空の waf:*GetWidget* アクセス許可がない場合に発生するエラーを示しています。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 waf:GetWidget on resource: my-example-widget
```

この場合、waf:*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 管理者に問い合わせてください。サインイン資格情報を提供した担当者が管理者です。

私にはiam を実行する権限がありません:PassRole

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、 AWS WAF Classic にロールを渡せるようにポリシーを更新する必要があります。

新しいサービスロールやサービスにリンクされたロールを作成する代わりに、 AWS のサービス 既存のロールをそのサービスに渡すことができるものもあります。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。

以下の例のエラーは、 marymajor という IAM ユーザーがコンソールを使用して AWS WAF Classic で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権限が必要です。Mary には、ロールをサービスに渡す権限がありません。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

この場合、メアリー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、管理者に問い合わせてください。 AWS サインイン資格情報を提供した担当者が管理者です。

AWS アカウントAWS WAF 自分以外の人にもクラシックリソース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、以下を参照してください:

- AWS WAF Classic がこれらの機能をサポートしているかどうかについては、を参照してくださいAWS WAF Classic と IAM の連携方法。

- AWS アカウント 所有しているリソース全体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「AWS アカウント 所有する別の 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第三者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「AWS アカウント 第三者が所有するリソースへのアクセスの提供」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でのロールとリソースベースのポリシーの使用の違いの詳細については、「IAM ユーザーガイド」の「IAM ロールとリソースベースのポリシーとの相違点」を参照してください。

Classic でのサービスにリンクされたロールの使用 AWS WAF

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS WAF 従来は 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は、Classic に直接リンクされているユニークなタイプの IAM ロールです。 AWS WAF サービスにリンクされたロールは AWS WAF Classic によって事前定義されており、 AWS ユーザーに代わってサービスが他のサービスを呼び出すために必要なすべての権限が含まれています。

サービスにリンクされたロールを使用すると、必要な権限を手動で追加する必要がないため、 AWS WAF Classic の設定が容易になります。 AWS WAF Classic はサービスにリンクされたロールの権限を定義し、特に定義されていない限り、 AWS WAF Classic だけがロールを引き受けることができます。定義された許可には、信頼ポリシーと許可ポリシーが含まれます。こ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。

サービスにリンクされたロールを削除するには、まずそのロールの関連リソースを削除します。これにより、リソースへのアクセス権限を誤って削除してしまうことがなくなるため、 AWS WAF Classic リソースを保護できます。

サービスリンクロールをサポートする他のサービスについては、「IAM と連携するAWS のサービス」を参照して、[サービスリンクロール] 列が [はい] のサービスを探してください。そのサービスに関する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、リンクが設定されている [Yes] (はい) を選択します。

AWS WAF Classic 向けのサービスにリンクされたロール許可

AWS WAF Classic では以下のサービスにリンクされたロールを使用します。

- `AWSServiceRoleForWAFLogging`
- `AWSServiceRoleForWAFRegionalLogging`

AWS WAF クラシックはこれらのサービスにリンクされたロールを使用して Amazon Data Firehose にログを書き込みます。これらのロールはログインを有効にした場合にのみ使用されます。 AWS WAF詳細については、「ウェブ ACL トラフィック情報のログ記録」を参照してください。

`AWSServiceRoleForWAFLogging` および `AWSServiceRoleForWAFRegionalLogging` のサービスにリンクされたロールは、ロールを引き受ける上でそれぞれに対応する次のサービスを信頼します。

- `waf.amazonaws.com`

  `waf-regional.amazonaws.com`

ロールのアクセス権限ポリシーにより、 AWS WAF Classic は指定されたリソースに対して以下のアクションを実行できます。

- アクション:`firehose:PutRecord`および `firehose:PutRecordBatch` Amazon Data Firehose のデータストリームリソースで、名前が「aws-waf-logs-」で始まる。例えば、`aws-waf-logs-us-east-2-analytics` です。

サービスにリンクされたロールの作成、編集、削除をIAM エンティティ (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。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許可」を参照してください。

AWS WAF Classic 向けのサービスにリンクされたロールの作成

サービスリンクロールを手動で作成する必要はありません。 AWS WAF でクラシックロギングを有効にするか AWS Management Console、クラシック CLI または AWS WAF Classic API

PutLoggingConfiguration でリクエストを行うと、 AWS WAF AWS WAF Classic によってサービスにリンクされたロールが自動的に作成されます。

ログ記録を有効化するためには、iam:CreateServiceLinkedRole 許可が必要です。

このサービスリンクロールを削除した後で再度作成する必要が生じた場合は、同じ方法でアカウントにロールを再作成できます。 AWS WAF クラシックロギングを有効にすると、 AWS WAF Classic はサービスにリンクされたロールを再度作成します。

AWS WAF Classic 向けのサービスにリンクされたロールの編集

AWS WAF Classic では、AWSServiceRoleForWAFLoggingAWSServiceRoleForWAFRegionalLoggingおよびサービスにリンクされたロールを編集することはできません。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「サービスにリンクされたロールの編集」を参照してください。

AWS WAF Classic 向け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。そうすることで、積極的にモニタリングまたは保守されていない未使用のエンティティを排除できます。ただし、手動で削除する前に、サービスリンクロールのリソースをクリーンアップする必要があります。

> **Note**
>
> AWS WAF Classic サービスがそのロールを使用していたときにリソースを削除しようとすると、削除が失敗する可能性があります。失敗した場合は、数分待ってから操作を再試行してください。

AWSServiceRoleForWAFLoggingとが使用している AWS WAF Classic リソースを削除するには AWSServiceRoleForWAFRegionalLogging

1. AWS WAF Classic コンソールで、すべてのウェブ ACL からロギングを削除します。詳細については、「ウェブ ACL トラフィック情報のログ記録」を参照してください。
2. API または CLI を使用して、ログ記録が有効化されている各ウェブ ACL に DeleteLoggingConfiguration リクエストを送信します。詳細については、「AWS WAF Classic API リファレンス」を参照してください。

IAM を使用して、サービスにリンクされたロールを手動で削除するには

AWSServiceRoleForWAFLogging および AWSServiceRoleForWAFRegionalLogging サービスにリンクされたロールを削除するには、IAM コンソール、IAM CLI、または IAM API を使用します。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。

AWS WAF Classic サービスにリンクされたロールをサポートするリージョン

AWS WAF Classic では、以下の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。
AWS リージョン

リージョン名	リージョンアイデンティティ	AWS WAF クラシックでのSupport
米国東部 (バージニア北部)	us-east-1	あり
米国東部 (オハイオ)	us-east-2	あり
米国西部 (北カリフォルニア)	us-west-1	あり
米国西部 (オレゴン)	us-west-2	あり
アジアパシフィック (ムンバイ)	ap-south-1	あり
アジアパシフィック (大阪)	ap-northeast-3	あり
アジアパシフィック (ソウル)	ap-northeast-2	あり
アジアパシフィック (シンガポール)	ap-southeast-1	あり
アジアパシフィック (シドニー)	ap-southeast-2	あり
アジアパシフィック (東京)	ap-northeast-1	あり
カナダ (中部)	ca-central-1	あり
欧州 (フランクフルト)	eu-central-1	あり
欧州 (アイルランド)	eu-west-1	あり
欧州 (ロンドン)	eu-west-2	あり

リージョン名	リージョンアイデンティティ	AWS WAF クラシックでのSupport
欧州 (パリ)	eu-west-3	あり
南米 (サンパウロ)	sa-east-1	あり

AWS WAF クラシックでのロギングとモニタリング

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールや Web ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS WAF Classic AWS とそのソリューションの信頼性、可用性、パフォーマンスを維持するには、モニタリングが重要です。 AWS ソリューションのすべての部分から監視データを収集して、マルチポイント障害が発生した場合により簡単にデバッグできるようにする必要があります。 AWS には、 AWS WAF Classic リソースを監視し、発生する可能性のあるイベントに対応するためのツールがいくつか用意されています。

Amazon CloudWatch アラーム

CloudWatch アラームを使用すると、指定した期間にわたって 1 つのメトリクスを監視できます。メトリックスが特定のしきい値を超えると、Amazon SNS CloudWatch AWS Auto Scaling トピックまたはポリシーに通知を送信します。詳細については、「Amazon によるモニタリング CloudWatch」を参照してください。

AWS CloudTrail ログ

CloudTrail AWS WAF Classic でユーザー、ロール、 AWS またはサービスが実行したアクションの記録を提供します。によって収集された情報を使用して CloudTrail、 AWS WAF Classic に対して行われた要求、要求が行われた IP アドレス、要求の実行者、実行日時、その他の詳細情報を特定できます。詳細については、「での AWS CloudTrail API コールのログ記録」を参照してください。

AWS WAF Classic のコンプライアンス検証

> **Note**
>
> これは AWS WAF Classic ドキュメントです。このバージョンは、2019 年 11 月 AWS WAF より前にルールやウェブ ACLs などのリソースを作成し AWS WAF 、まだ最新バージョンに移行していない場合にのみ使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては、 AWS WAF「」を参照してくださいAWS WAF。

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、コンプライアンスプログラムAWS のサービス による対象範囲内のコンプライアンスプログラムを参照し、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、「 でのレポートのダウンロード AWS Artifact」の」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。 は、コンプライアンスに役立つ以下のリソース AWS を提供しています。

- セキュリティとコンプライアンスのクイックスタートガイド – これらのデプロイガイドでは、アーキテクチャ上の考慮事項について説明し、セキュリティとコンプライアンスに重点を置いたベースライン環境 AWS を にデプロイする手順について説明します。
- アマゾン ウェブ サービスにおける HIPAA セキュリティとコンプライアンスのアーキテクチャ — このホワイトペーパーでは、企業が AWS を使用して HIPAA 対象アプリケーションを作成する方法について説明します。

  > **Note**
  >
  > すべて AWS のサービス HIPAA の対象となるわけではありません。詳細については、「HIPAA 対応サービスのリファレンス」を参照してください。

- AWS コンプライアンスリソース – このワークブックとガイドのコレクションは、お客様の業界や地域に適用される場合があります。

- AWS カスタマーコンプライアンスガイド — コンプライアンスの観点から責任共有モデルを理解します。このガイドでは、ガイダンスを保護し AWS のサービス 、複数のフレームワーク (米国国立標準技術研究所 (NIST)、Payment Card Industry Security Standards Council (PCI)、国際標準化機構 (ISO) を含む) のセキュリティコントロールにマッピングするためのベストプラクティスをまとめています。
- 「 デベロッパーガイド」の「 ルールによるリソースの評価」 – この AWS Config サービスは、リソース設定が社内プラクティス、業界ガイドライン、および規制にどの程度準拠しているかを評価します。 AWS Config
- AWS Security Hub – これにより AWS のサービス 、 内のセキュリティ状態を包括的に確認できます AWS。Security Hub では、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対するコンプライアンスをチェックします。サポートされているサービスとコントロールのリストについては、「Security Hub のコントロールリファレンス」を参照してください。
- Amazon GuardDuty – これにより AWS アカウント、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環境を監視することで、、、ワークロード、コンテナ、データに対する潜在的な脅威 AWS のサービス を検出します。 GuardDuty は、特定のコンプライアンスフレームワークで義務付けられている侵入検知要件を満たすことで、PCI DSS などのさまざまなコンプライアンス要件への対応に役立ちます。
- AWS Audit Manager – これにより AWS のサービス 、 AWS 使用状況を継続的に監査し、リスクの管理方法と規制や業界標準への準拠を簡素化できます。

レジリエンス・イン・クラシック AWS WAF

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールや Web ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS グローバルインフラストラクチャは、 AWS リージョン アベイラビリティーゾーンを中心に構築されています。 AWS リージョン 物理的に分離された複数のアベイラビリティーゾーンを提供し、低レイテンシー、高スループット、冗長性の高いネットワークで接続します。アベイラビリティーゾーンでは、アベイラビリティーゾーン間で中断せずに、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、耐障害性、およびスケーラビリティが優れています。

AWS リージョン およびアベイラビリティーゾーンの詳細については、「グローバルインフラストラクチャ」を参照してください。AWS

AWS WAF Classic のインフラストラクチャーセキュリティ

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールやウェブ ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

マネージドサービスとして、 AWS WAF Classic AWS はグローバルネットワークセキュリティによって保護されています。 AWS AWS セキュリティサービスとインフラストラクチャの保護方法については、「AWS Cloud Security」を参照してください。 AWS 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設計するには、「Security Pillar AWS Well-Architected Framework におけるインフラストラクチャ保護」を参照してください。

AWS 公開されている API 呼び出しを使用して、ネットワーク経由で AWS WAF Classic にアクセスします。クライアントは以下をサポートする必要があります:

- Transport Layer Security (TLS)。TLS 1.2、できれば TLS 1.3 が必要です。
- DHE (Ephemeral Diffie-Hellman) や ECDHE (Elliptic Curve Ephemeral Diffie-Hellman) などの Perfect Forward Secrecy (PFS) を使用した暗号スイート。これらのモードは、Java 7 以降など、ほとんどの最新システムでサポートされています。

また、リクエストには、アクセスキー ID と、IAM プリンシパルに関連付けられているシークレットアクセスキーを使用して署名する必要があります。または、AWS Security Token ServiceAWS STS を使用して、一時的なセキュリティ認証情報を生成し、リクエストに署名することもできます。

AWS WAF クラシック・ クォータ

> **Note**
>
> これは AWS WAF Classic ドキュメントです。2019 年 11 AWS WAF 月以前にルールや Web ACL AWS WAF などのリソースを作成していて、まだ最新バージョンに移行していない場合にのみ、このバージョンを使用してください。リソースを移行するには、「AWS WAF クラシックリソースをに移行する AWS WAF」を参照してください。
> の最新バージョンについては AWS WAF、を参照してください。 AWS WAF

AWS WAF Classic には以下のクォータ (以前は制限と呼ばれていました) が適用されます。

AWS WAF Classic では、1 アカウントあたりのリージョンあたりのエンティティ数にデフォルトクォータが設定されています。このクォータの引き上げをリクエストすることができます。

リソース	1 リージョン、1 アカウントあたりのデフォルトのクォータ
ウェブ ACL	50
[Rules] (ルール)	100

リソース	1 リージョン、1 アカウントあたりのデフォルトのクォータ
R ate-based-rules	5
アカウントあたり、リージョンあたりの条件	正規表現一致と Geo 一致を除くすべての条件で、100 個の各条件タイプ。100 のサイズ制約条件と 100 の IP 一致条件はその一例です。正規表現一致条件については、次の表を参照してください。
1 秒あたりのリクエスト	ウェブ ACL あたり 25,000*

*このクォータは、Application Load Balancer AWS WAF のクラシックにのみ適用されます。 AWS WAF Classic 版の 1 秒あたりのリクエスト数 (RPS) CloudFront クォータは、開発者ガイドで説明されている RPS クォータのサポートと同じです。 CloudFront CloudFront

クラシックエンティティの以下のクォータは変更できません。 AWS WAF

リソース	1 アカウント、1 リージョンあたりのクォータ
ウェブ ACL あたりのルールグループ	2: 顧客が作成したルールグループ 1 つと、ルー

リソース	1 アカウント、1 リージョンあたりのクォータ
	ルグループ 1 つ AWS Marketplace
ウェブ ACL あたりのルールの数	10
ルールあたりの条件の数	10
IP 一致条件ごとの IP アドレス範囲 (CIDR 表記)	10,000 一度に更新できるアドレスは最大で 1,000 個です。API コール `UpdateIPSet` は、1 回のリクエストで最大 1,000 個のアドレスを受け入れます。
レートベースのルールごとにブロックされている IP アドレス	10,000
5 分間あたりの最小レートベースのルール制限	100
クロスサイトスクリプティング一致条件あたりのフィルターの数	10
サイズ制約条件あたりのフィルターの数	10
SQL インジェクション一致条件あたりのフィルターの数	10
文字列一致条件あたりのフィルターの数	10
文字列一致条件の HTTP ヘッダー名の文字数 (ウェブリクエストのヘッダーに特定の値があるかどうかを調べるように AWS WAF Classic を設定した場合)	40

リソース	1 アカウント、1 リージョンあたりのクォータ
文字列一致条件で、 AWS WAF Classic で検索させたい値の文字数	50
正規表現一致条件	10
正規表現一致条件では、 AWS WAF Classic で検索させたいパターン内の文字数	70
正規表現一致条件で、1 パターンセットあたりのパターンの数	10
正規表現一致条件で、1 正規表現条件あたりのパターンセットの数	1
パターンセット	5
Geo 一致条件	50
Geo 一致条件ごとの場所	50

AWS WAF Classic では、地域ごとのアカウントあたりのコール数について以下の固定クォータが設定されています。これらのクォータは、コンソール、CLI、REST API、SDK など、利用可能なあらゆる手段によるサービスへの呼び出し総数に適用されます。 AWS CloudFormationこれらのクォータは変更できません。

コールタイプ	1 アカウント、1 リージョンあたりのクォータ
`AssociateWebACL` へのコールの最大数	2 秒あたり 1 個のリクエスト
`DisassociateWebACL` へのコールの最大数	2 秒あたり 1 個のリクエスト
`GetWebACLForResource` へのコールの最大数	1 秒あたり 1 個のリクエスト

コールタイプ	1 アカウント、1 リージョンあたりのクォータ
`ListResourcesForWebACL` へのコールの最大数	1 秒あたり 1 個のリクエスト
`CreateWebACLMigrationStack` へのコールの最大数	1 秒あたり 1 個のリクエスト
`GetChangeToken` へのコールの最大数	1 秒あたり 10 個のリクエスト
`GetChangeTokenStatus` へのコールの最大数	1 秒あたり 1 個のリクエスト
個々の `List` アクションへのコールの最大数 (他にクォータが定義されていない場合)	1 秒あたり 5 個のリクエスト
個々の `Create`、`Put`、`Get`、または `Update` アクションへのコールの最大数 (他にクォータが定義されていない場合)	1 秒あたり 1 個のリクエスト

AWS Shield

Distributed Denial of Service (DDoS) 攻撃から保護することは、インターネットに直接接続するアプリケーションにとって極めて重要です。アプリケーションを構築すると AWS、 AWS 追加費用なしで提供される保護機能を利用できるようになります。さらに、 AWS Shield Advanced マネージド脅威対策サービスを利用すれば、DDoS 検知、軽減、対応機能を追加してセキュリティ体制を強化できます。

AWS は、インターネット上の不正行為者からの防御において、高い可用性、セキュリティ、および回復力を確保するためのツール、ベストプラクティス、サービスの提供に努めています。このガイドは、IT 関連事項の意思決定者やセキュリティエンジニアが、Shield と Shield Advanced を使用して DDoS 攻撃や他の外部の脅威からアプリケーションをより適切に保護する方法を理解できるように提供されています。

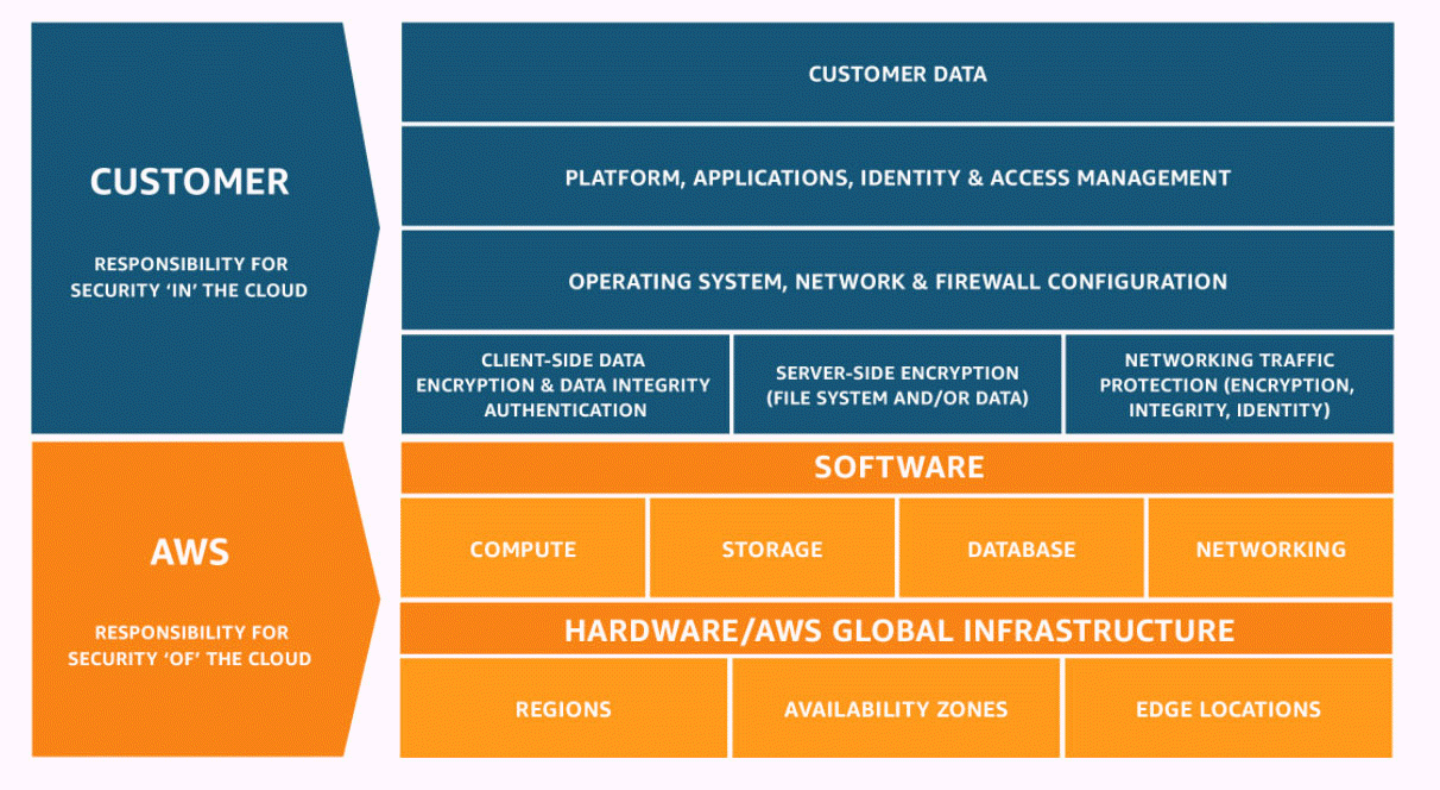
アプリケーションを構築すると AWS、UDP AWS リフレクション攻撃や TCP SYN フラッドなどの一般的なボリューム型 DDoS 攻撃ベクトルから自動的に保護されます。DDoS AWS レジリエンシーを考慮したアーキテクチャを設計および構成することで、これらの保護機能を活用して実行するアプリケーションの可用性を確保できます。

このガイドでは、DDoS レジリエンシーを実現するためのアプリケーションアーキテクチャの設計、作成、および設定に役立つ推奨事項について説明します。このガイドで提供されるベストプラクティスに準拠するアプリケーションは、大規模な DDoS 攻撃や広範囲の DDoS 攻撃ベクトルによってターゲットとされた場合に、改善された可用性の維持の恩恵を受けることができます。さらに、このガイドでは、Shield Advanced を使用して、重要なアプリケーション向けに最適化された DDoS 保護体制を実装する方法について説明します。これには、顧客に一定レベルの可用性が保証されているアプリケーションや、DDoS AWS イベント中からの運用サポートを必要とするアプリケーションが含まれます。

セキュリティは、 AWS お客様とお客様との間で共有される責任です。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して説明しています。

- クラウドのセキュリティ — AWS AWS AWS クラウドクラウド内でサービスを実行するインフラストラクチャを保護する責任があります。 AWS また、安全に使用できるサービスも提供します。セキュリティの有効性は、AWS コンプライアンスプログラムの一環として、サードパーティーの審査機関によって定期的にテストおよび検証されています。AWSServiceRoleForAWSShield に適用されるコンプライアンスプログラムについては、「コンプライアンスプログラムによる対象範囲内のAWS のサービス」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、 AWS 使用する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、組織の要件、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。



AWS Shield アンドシールドアドバンスドの仕組み

AWS Shield Standard また、ネットワーク層、トランスポート層 (レイヤー 3 と 4)、アプリケーションレイヤー (レイヤー 7) AWS のリソースに対する分散型サービス拒否 (DDoS) AWS Shield Advanced 攻撃からの保護も提供します。DDoS 攻撃は、侵害された複数のシステムが、ターゲットに対してトラフィックでフラッディングを試みる攻撃です。DDoS 攻撃は正当なエンドユーザーがターゲットのサービスにアクセスするのを妨げ、圧倒的なトラフィック量のためにターゲットがクラッシュする可能性があります。

AWS Shield さまざまな既知の DDoS 攻撃ベクトルやゼロデイ攻撃ベクトルからの保護を提供します。Shield の検出および緩和は、検出時において、サービスにとって明示的に既知でなくても、脅威に対するカバレッジを提供するように設計されています。Shield Standard は、 AWSの使用時に自動的に提供され、追加料金はかかりません。

Shield が検出する攻撃のクラスには、次のものが含まれます。

- [Network volumetric attacks (layer 3)] (ネットワークボリューム攻撃 (レイヤー 3)) – これは、インフラストラクチャレイヤー攻撃のベクトルのサブカテゴリです。これらのベクトルは、正当なユーザーへのサービスを拒否するために、ターゲットネットワークまたはリソースの容量を飽和させることを試みます。
- [Network protocol attacks (layer 4)] (ネットワークプロトコル攻撃 (レイヤー 4)) – これは、インフラストラクチャレイヤー攻撃のベクトルのサブカテゴリです。これらのベクトルは、プロトコルを悪用してターゲットリソースへのサービスを拒否します。ネットワークプロトコル攻撃の一般的な例は TCP SYN フラッドです。TCP SYN フラッドは、サーバー、ロードバランサー、ファイアウォールなどのリソースの接続状態を使い果たす可能性があります。ネットワークプロトコル攻撃は、帯域幅消費型である場合もあります。例えば、大規模な TCP SYN フラッドには、ターゲットリソースまたは中間リソースの状態を使い果たしながら、ネットワークの容量を飽和させる意図があるかもしれません。
- アプリケーションレイヤー攻撃 (レイヤー 7) - このカテゴリの攻撃ベクトルは、ウェブリクエストのフラッドなど、ターゲットに対して有効なクエリをアプリケーションにフラッディングすることによって、正当なユーザーへのサービス提供の拒否を試みます。

目次

- AWS Shield Advanced Web アプリケーションの緩和ロジック

AWS Shield Standard 概要

AWS Shield は、アプリケーションの境界を保護するマネージド型の脅威対策サービスです。境界は、ネットワーク外からのアプリケーショントラフィックの最初の侵入点です。 AWS

アプリケーションの境界がどこにあるかを判断するには、ユーザーがインターネットからアプリケーションにどのようにアクセスするかを検討します。 AWS 最初のエントリポイントがリージョン内にある場合、アプリケーションの境界は Amazon Virtual Private Cloud (VPC) です。ユーザーが Amazon Route 53 によってアプリケーションに誘導され、最初に Amazon CloudFront またはを使用してアプリケーションにアクセスする場合 AWS Global Accelerator、 AWS アプリケーションの境界はネットワークのエッジから始まります。

SShield dは、実行中のすべてのアプリケーションにDDoS検出と軽減の利点をもたらしますが AWS、アプリケーションアーキテクチャを設計する際に行う決定は、DDoS耐障害性のレベルに影響します。DDoS Resiliency は、攻撃を受けている最中に、想定されるパラメータ内でアプリケーションが動作し続ける能力です。

AWS すべてのお客様は、追加料金なしでShield Standardの自動保護の恩恵を受けることができます。Shield Standard は、お客様のウェブサイトやアプリケーションを標的として一般的かつ頻繁に発生するネットワークおよび転送レイヤーの DDoS 攻撃に対して防御します。Shield Standard AWS はすべての顧客を保護するのに役立ちますが、Amazon Route 53 ホストゾーン、Amazon CloudFront ディストリビューション、 AWS Global Accelerator および標準アクセラレータでは特にメリットがあります。これらのリソースは、既知のすべてのネットワークおよびトランスポートレイヤー攻撃に対して、可用性についての包括的な保護を受けます。

AWS Shield Advanced 概要

AWS Shield Advanced は DDoS 攻撃、ボリュームボット、脆弱性悪用の試みなど、外部の脅威からアプリケーションを保護するのに役立つマネージドサービスです。攻撃に対するより高いレベルの保護のために、 AWS Shield Advancedサブスクライブすることができます。

Shield Advanced をサブスクライブしてリソースに保護を追加すると、Shield Advanced は、それらのリソースのために拡張された DDoS 攻撃保護を提供します。Shield Advanced から受ける保護は、アーキテクチャと設定の選択内容によって異なります。このガイドの情報を参照して、Shield Advanced を使用して回復力のあるアプリケーションを構築して保護し、エキスパートのサポートが必要なときにエスカレーションできます。

Shield アドバンストのサブスクリプションと費用 AWS WAF

Shield アドバンスドサブスクリプションは、Shield AWS WAF アドバンスドで保護するリソースの標準機能を使用するコストをカバーします。Shield Advanced AWS WAF 保護の対象となる標準料金は、ウェブ ACL あたりのコスト、ルールあたりのコスト、およびウェブリクエストインスペクションの 100 万リクエストあたりの基本価格（最大 1,500 WCU、デフォルトのボディサイズまで）です。

Shield Advanced 自動アプリケーションレイヤー DDoS 軽減を有効にすると、150 ウェブ ACL キャパシティユニット (WCU) を使用するルールグループがウェブ ACL に追加されます。これらの WCU は、ウェブ ACL 内の WCU の使用量に対してカウントされます。詳細については、Shield Advanced アプリケーションレイヤー DDoS 自動緩和、Shield Advanced ルールグループ、およびAWS WAF ウェブ ACL キャパシティーユニット (WCUs)を参照してください。

Shield アドバンスドへのサブスクリプションは、Shield アドバンスドを使用して保護していないリソースの使用には適用されません。 AWS WAF また、 AWS WAF 保護対象リソースの標準外の追加費用もカバーされません。 AWS WAF 非標準費用の例としては、ボットコントロール、CAPTCHA ルールアクション、1,500 個以上の WCU を使用するウェブ ACL、デフォルトの本文サイズを超えるリクエスト本文の検査などがあります。全リストは料金ページに記載されています。 AWS WAF

詳細情報および料金の例については、「Shield の料金」および「AWS WAF の料金」を参照してください。

Shield Advanced サブスクリプションの請求

AWS チャネルリセラーの場合は、アカウントチームに相談して情報やガイダンスを受けてください。この請求情報は、 AWS チャネルリセラー以外の顧客向けです。

他のすべてについては、次のサブスクリプションと請求のガイドラインが適用されます。

- AWS Organizations 組織のメンバーであるアカウントの場合、支払いアカウント自体が購読されているかどうかに関係なく、組織の支払いアカウントに対してShield Advanced AWS サブスクリプションを請求します。
- 同じ AWS Organizations 一括請求 (コンソリデーティッドビリング) アカウントファミリー に属する複数のアカウントをサブスクライブする場合、1 つのサブスクリプション料金はファミリー内のすべてのサブスクライブアカウントに対するものです。組織は、すべての AWS アカウント とそのすべてのリソースを所有している必要があります。
- 複数の組織のために複数のアカウントをサブスクライブする場合でも、すべてを所有しているのであれば、すべての組織、アカウント、リソースのサブスクリプション料金の支払いを引き続き

1 回で行うことができます。 AWS アカウントマネージャーまたはサポートに連絡して、1 AWS Shield Advanced つを除くすべての組織のサブスクリプション料金の免除をリクエストしてください。

詳細な料金情報と例については、「AWS Shield の料金表」を参照してください。

トピック

- AWS Shield Advanced 保護リソース
- AWS Shield Advanced 機能とオプション
- AWS Shield Advanced をサブスクライブして追加の保護を適用するか否かの判断

AWS Shield Advanced 保護リソース

> **Note**
>
> Shield アドバンスドプロテクションは、Shield アドバンスドで明示的に指定したリソース、または Shield アドバンスドポリシーで保護したリソースに対してのみ有効になります。 AWS Firewall Manager Shield Advanced は、リソースを自動的に保護しません。

Shield Advanced を使用すると、次のリソースタイプで高度なモニタリングと保護を行うことができます。

- Amazon CloudFront ディストリビューション。Shield Advanced は、 CloudFront 継続的なデプロイのために、保護されたプライマリディストリビューションに関連するすべてのステージングディストリビューションを保護します。
- Amazon Route 53 ホストゾーン。
- AWS Global Accelerator 標準アクセラレータ。
- Amazon EC2 Elastic IP アドレス。Shield Advanced は、保護された Elastic IP アドレスに関連付けられているリソースを保護します。
- Amazon EC2 インスタンス (Amazon EC2 Elastic IP アドレスへの関連付け経由)
- 次の Elastic Load Balancing (ELB) ロードバランサー:
  - Application Load Balancer。
  - Classic Load Balancer。
  - Network Load Balancer (Amazon EC2 Elastic IP アドレスへの関連付け経由)。

これらのリソースタイプの保護に関する追加情報については、「AWS Shield Advanced リソースタイプ別の保護」を参照してください。

AWS Shield Advanced 機能とオプション

AWS Shield Advanced サブスクリプションには以下の機能とオプションが含まれます。これらは、すでに搭載されている DDoS 検出および軽減機能を補完するものです。 AWS

- AWS WAF 統合 — Shield Advanced は、 AWS WAF アプリケーション層保護の一環としてウェブ ACL、ルール、およびルールグループを使用します。の詳細については AWS WAF、を参照してください。AWS WAF 仕組み

> **Note**
>
> Shield アドバンスドサブスクリプションは、Shield AWS WAF アドバンスドで保護するリソースの標準機能を使用するコストをカバーします。Shield Advanced AWS WAF 保護の対象となる標準料金は、ウェブ ACL あたりのコスト、ルールあたりのコスト、およびウェブリクエストインスペクションの 100 万リクエストあたりの基本価格 (最大 1,500 WCU、デフォルトのボディサイズまで) です。
> Shield Advanced 自動アプリケーションレイヤー DDoS 軽減を有効にすると、150 ウェブ ACL キャパシティユニット (WCU) を使用するルールグループがウェブ ACL に追加されます。これらの WCU は、ウェブ ACL 内の WCU の使用量に対してカウントされます。詳細については、Shield Advanced アプリケーションレイヤー DDoS 自動緩和、Shield Advanced ルールグループ、およびAWS WAF ウェブ ACL キャパシティーユニット (WCUs)を参照してください。
> Shield アドバンスドへのサブスクリプションは、Shield アドバンスドを使用して保護していないリソースの使用には適用されません。 AWS WAF また、 AWS WAF 保護対象リソースの標準外の追加費用もカバーされません。 AWS WAF 非標準費用の例としては、ボットコントロール、CAPTCHAルールアクション、1,500 個以上の WCU を使用するウェブ ACL、デフォルトの本文サイズを超えるリクエスト本文の検査などがあります。全リストは料金ページに記載されています。 AWS WAF
> 詳細情報および料金の例については、「Shield の料金」および「AWS WAF の料金」を参照してください。

- アプリケーションレイヤーの DDoS の自動緩和 - Shield Advanced は、保護されたリソースに対するアプリケーションレイヤー (レイヤー 7) 攻撃を自動的に緩和して対応するように設定できます。Shield Advancedは、自動軽減機能により、 AWS WAF 既知のDDoSソースからのリクエストにレート制限を適用し、 AWS WAF 検出されたDDoS攻撃に対応してカスタムプロテクションを自

動的に追加および管理します。攻撃の一部であるウェブリクエストをカウントまたはブロックするように自動緩和を設定できます。

詳細については、「Shield Advanced アプリケーションレイヤー DDoS 自動緩和」を参照してください。

- ヘルスベースの検出 – Shield Advanced で Amazon Route 53 ヘルスチェックを使用して、イベントの検出と緩和の通知を受けることができます。ヘルスチェックは、仕様に従ってアプリケーションをモニタリングし、仕様が満たされた場合は正常、そうでない場合は異常を報告します。Shield Advanced でヘルスチェックを使用すると、誤検出を防止するのに役立つともに、保護されたリソースが異常な場合により迅速に検出および緩和できます。Route 53 ホストゾーンを除くリソースタイプのために、ヘルスベースの検出を使用できます。Shield Advanced のプロアクティブエンゲージメントは、ヘルスベースの検出が有効になっているリソースでのみ使用できます。

  詳細については、「ヘルスチェックを使用したHealth ベースの検出」を参照してください。
- 保護グループ – 保護グループを使用して、保護されたリソースの論理グループを作成し、グループ全体の検出と緩和を強化できます。新しく保護されたリソースを自動的に含めるように、保護グループのメンバーシップの条件を定義できます。保護されたリソースは、複数の保護グループに所属できます。

  詳細については、「AWS Shield Advanced 保護グループ」を参照してください。
- DDoS イベントと攻撃の可視性の向上 – Shield Advanced を使用すると、高度なリアルタイムのメトリクスとレポートにアクセスして、保護された AWS のリソースに対するイベントと攻撃の可視性を高めることができます。この情報には、Shield アドバンスド API とコンソール、および Amazon CloudWatch メトリクスからアクセスできます。

  詳細については、「DDoS イベントの可視性」を参照してください。
- AWS Firewall Managerによる Shield Advanced 保護の一元管理 – Firewall Manager を使用して、新しいアカウントとリソースに Shield Advanced 保護を自動的に適用し、ウェブ ACL に AWS WAF ルールをデプロイできます。Firewall Manager Shield Advanced 保護ポリシーは、Shield Advanced のお客様に追加料金なしでご利用いただけます。また、Amazon Simple Notification Service (SNS) トピックまたは AWS Security Hubで Firewall Manager を使用して、アカウントの Shield Advanced モニタリングアクティビティを一元化することもできます。

  Firewall Manager を使用して Shield Advanced 保護機能を管理する方法については、「AWS Firewall Manager」および「AWS Shield Advanced ポリシー」を参照してください。Firewall Manager の料金については、「AWS Firewall Manager の料金」を参照してください。

- AWS Shield レスポンスチーム (SRT) — SRTには AWS、Amazon.comとその子会社の保護に関する豊富な経験があります。 AWS Shield Advanced のお客様は、アプリケーションの可用性に影響を与える DDoS 攻撃を受けている最中に、いつでも SRT に連絡してサポートを求めることができます。SRT と連携して、リソースのカスタム緩和策を作成および管理することもできます。SRT のサービスを使用するには、ビジネスサポートプランまたはエンタープライズサポートプランをサブスクライブする必要もあります。

  詳細については、「Shield Response Team (SRT) のサポート」を参照してください。
- プロアクティブな関与 – Shield Advanced が検出したイベントの発生中に、保護されたリソースに関連付けられた Amazon Route 53 ヘルスチェックが異常になった場合、プロアクティブな関与によって、Shield Response Team (SRT) はお客様に直接ご連絡します。これにより、アプリケーションの可用性に影響を及ぼす可能性のある攻撃が疑われる場合に、エキスパートとより迅速に連携することができます。

  詳細については、「プロアクティブな関与の設定」を参照してください。
- コスト保護の機会 — Shield Advanced は、保護対象リソースに対する DDoS AWS 攻撃によって発生する可能性のある請求額の急増に対して、ある程度のコスト保護を提供します。これには、Shield Advanced データ転送アウト (DTO) 使用料の急増に対する補償が含まれる場合があります。Shield Advanced は、Shield Advanced サービスクレジットという形であらゆるコスト保護を提供します。

  詳細については、「でのクレジットのリクエスト AWS Shield Advanced」を参照してください。

AWS Shield Advanced をサブスクライブして追加の保護を適用するか否かの判断

AWS Shield Advanced をサブスクライブするアカウントと追加の保護を適用する場合を判断するのに役立てるため、このセクションのシナリオを確認してください。Shield Advanced では、一括請求 (コンソリデーティッドビリング) の支払いアカウントで作成されたすべてのアカウントについて、1 つの月額サブスクリプション料金をお支払いいただきます。さらに、転送されたデータの GB (OUT) に基づく使用料金もお支払いいただきます。Shield Advanced の料金については、「AWS Shield Advanced の料金」を参照してください。

Shield Advanced でアプリケーションとそのリソースを保護するには、アプリケーションを管理するアカウントを Shield Advanced にサブスクライブし、アプリケーションのリソースに保護を追加します。アカウントのサブスクライブとリソースの保護については、「の開始方法 AWS Shield Advanced」を参照してください。

Shield アドバンストのサブスクリプションと費用 AWS WAF

Shield アドバンスドサブスクリプションは、Shield AWS WAF アドバンスドで保護するリソースの標準機能を使用するコストをカバーします。Shield Advanced AWS WAF 保護の対象となる標準料金は、ウェブ ACL あたりのコスト、ルールあたりのコスト、およびウェブリクエストインスペクションの 100 万リクエストあたりの基本価格（最大 1,500 WCU、デフォルトのボディサイズまで）です。

Shield Advanced 自動アプリケーションレイヤー DDoS 軽減を有効にすると、150 ウェブ ACL キャパシティユニット (WCU) を使用するルールグループがウェブ ACL に追加されます。これらの WCU は、ウェブ ACL 内の WCU の使用量に対してカウントされます。詳細については、Shield Advanced アプリケーションレイヤー DDoS 自動緩和、Shield Advanced ルールグループ、およびAWS WAF ウェブ ACL キャパシティーユニット (WCUs)を参照してください。

Shield アドバンスドへのサブスクリプションは、Shield アドバンスドを使用して保護していないリソースの使用には適用されません。 AWS WAF また、 AWS WAF 保護対象リソースの標準外の追加費用もカバーされません。 AWS WAF 非標準費用の例としては、ボットコントロール、CAPTCHA ルールアクション、1,500 個以上の WCU を使用するウェブ ACL、デフォルトの本文サイズを超えるリクエスト本文の検査などがあります。全リストは料金ページに記載されています。 AWS WAF

詳細情報および料金の例については、「Shield の料金」および「AWS WAF の料金」を参照してください。

Shield Advanced サブスクリプションの請求

AWS チャネルリセラーの場合は、アカウントチームに相談して情報やガイダンスを受けてください。この請求情報は、 AWS チャネルリセラー以外の顧客向けです。

他のすべてについては、次のサブスクリプションと請求のガイドラインが適用されます。

- AWS Organizations 組織のメンバーであるアカウントの場合、支払いアカウント自体が購読されているかどうかに関係なく、組織の支払いアカウントに対してShield Advanced AWS サブスクリプションを請求します。
- 同じ AWS Organizations 一括請求 (コンソリデーティッドビリング) アカウントファミリー に属する複数のアカウントをサブスクライブする場合、1 つのサブスクリプション料金はファミリー内のすべてのサブスクライブアカウントに対するものです。組織は、すべての AWS アカウント とそのすべてのリソースを所有している必要があります。
- 複数の組織のために複数のアカウントをサブスクライブする場合でも、すべてを所有しているのであれば、すべての組織、アカウント、リソースのサブスクリプション料金の支払いを引き続き 1 回で行うことができます。 AWS アカウントマネージャーまたはサポートに連絡して、1 AWS

Shield Advanced つを除くすべての組織のサブスクリプション料金の免除をリクエストしてください。

詳細な料金情報と例については、「AWS Shield の料金表」を参照してください。

保護するアプリケーションの特定

次のいずれかが必要なアプリケーションには、Shield Advanced 保護を実装することを検討してください。

- アプリケーションのユーザーに保証された可用性。
- DDoS 攻撃によってアプリケーションが影響を受ける場合における、DDoS 緩和のエキスパートへの迅速なアクセス。
- アプリケーションが DDoS AWS 攻撃の影響を受ける可能性があることを認識し、セキュリティチームや運用チームからの攻撃の通知や、 AWS セキュリティチームや運用チームへのエスカレーションを通知します。
- クラウドコストの予測可能性 (DDoS 攻撃が AWS のサービスの利用に影響を与える場合を含む)。

アプリケーションまたはそのリソースが上記のいずれかを必要とする場合は、関連アカウントのサブスクリプションを作成することを検討してください。

保護するリソースの特定

サブスクライブした各アカウントについて、次のいずれかの特性を持つ各リソースに Shield Advanced 保護を追加することを検討してください。

- このリソースは、インターネット上の外部ユーザーにサービスを提供します。
- リソースはインターネットに公開されます。また、重要なアプリケーションの一部でもあります。インターネット上のユーザーがアクセスするかどうかにかかわらず、公開されているすべてのリソースを検討してください。
- AWS WAF リソースはウェブ ACL によって保護されています。

リソースの保護の作成と管理の詳細については、「でのリソース保護 AWS Shield Advanced」を参照してください。

さらに、このガイドの推奨事項は、DDoS レジリエンシーを考慮してアプリケーションを設計し、最適な保護を実現するために Shield Advanced の機能を適切に設定するのに役立ちます。

DDoS 攻撃の例

AWS Shield Advanced さまざまなタイプの攻撃に対する保護が強化されています。

次のリストでは、いくつかの一般的な攻撃の種類について説明します。

User Datagram Protocol (UDP) リフレクション攻撃

UDP リフレクション攻撃では、攻撃者はリクエストの発生元を偽装し、UDP を使用してサーバーから大量のレスポンスを引き出すことができます。攻撃対象の偽装 IP アドレスに向かう追加のネットワークトラフィックにより、対象のサーバーは遅くなり、正当なエンドユーザーが必要なリソースにアクセスできなくなります。

TCP SYN フラッド

TCP SYN フラッド攻撃の目的は、接続を半開状態にして、システムの利用可能なリソースを枯渇させることです。ユーザーがウェブサーバーのような TCP サービスに接続すると、クライアントは TCP SYN パケットを送信します。サーバーが肯定応答を返し、クライアントがそれ自体の肯定応答を返すことで、3 ウェイハンドシェイクが完了します。TCP SYN フラッドでは、3 回目の肯定応答が返されることはなく、サーバーはレスポンスを待ったままになります。このため、他のユーザーはサーバーに接続できなくなります。

DNS クエリフラッド

DNS クエリフラッドでは、攻撃者は複数の DNS クエリを使用して DNS サーバーのリソースを使い果たします。 AWS Shield Advanced Route 53 DNS サーバーに対する DNS クエリフラッド攻撃からの保護に役立ちます。

HTTP フラッド/キャッシュ無効化 (レイヤー 7) 攻撃

GET および POST フラッドを含む HTTP フラッドにより、攻撃者はウェブアプリケーションの実際のユーザーからのように見せかけて多数の HTTP リクエストを送信します。キャッシュ無効化攻撃は、HTTP フラッドの一種です。HTTP リクエストのクエリ文字列のバリエーションを使用して、エッジでキャッシュされているコンテンツが使用されないようにし、オリジンウェブサーバーからコンテンツが送信されるようにすることで、オリジンウェブサーバーに損害につながる可能性があるほどの負荷をかけます。

AWS Shield イベントの検出方法

AWS AWS AWS ネットワークと個々のサービスのサービスレベル検出システムを運用して、DDoS 攻撃を受けても常に利用可能な状態を維持できるようにする。さらに、 AWS リソースレベルの検出

システムが個々のリソースを監視して、リソースへのトラフィックが想定されるパラメータの範囲内であることを確認します。この組み合わせにより、既知の不正パケットをドロップしたり、潜在的に悪質なトラフィックをハイライトしたり、エンドユーザーからのトラフィックに優先順位を付けたりする緩和策を適用することで、 AWS AWS 対象となるリソースとサービスの両方を保護します。

検出されたイベントは、DDoS攻撃ベクトルの名前として、または評価がシグネチャではなくトラフィック量に基づいているかのようにVolumetric、Shield Advancedのイベントサマリー、攻撃の詳細、 CloudWatchおよびAmazonメトリクスに表示されます。DDoSDetected CloudWatchメトリックス内で利用できる攻撃ベクトルディメンションの詳細については、以下を参照してください。AWS Shield Advanced 指標

トピック

- インフラストラクチャレイヤーの脅威の検出ロジック
- アプリケーションレイヤーの脅威の検出ロジック
- アプリケーション内の複数のリソースの検出ロジック

インフラストラクチャレイヤーの脅威の検出ロジック

インフラストラクチャー層 (レイヤー 3 とレイヤー 4) で DDoS AWS 攻撃からターゲットリソースを保護するために使用される検出ロジックは、 AWS Shield Advancedリソースタイプとリソースが保護されているかどうかによって異なります。

CloudFront アマゾンとAmazon ルート 53 の検出

Route 53 CloudFront を使用してウェブアプリケーションを提供する場合、アプリケーションへのすべてのパケットは完全インラインの DDoS 軽減システムによって検査されるため、観測可能なレイテンシーは発生しません。 CloudFront ディストリビューションと Route 53 ホストゾーンに対する DDoS 攻撃はリアルタイムで軽減されます。これらの保護は、 AWS Shield Advancedを使用するかどうかにかかわらず適用されます。

DDoS CloudFront イベントの検出と軽減を最速で行うには、できる限りウェブアプリケーションのエントリポイントとして Route 53 を使用するというベストプラクティスに従ってください。

AWS Global Accelerator および地域サービスの検知

リソースレベルの検出は、クラシックロードバランサー、アプリケーションロードバランサー、Elastic IP アドレス (EIP) など、 AWS Global Accelerator AWS リージョンで起動される標準アクセラレーターとリソースを保護します。これらのリソースタイプは、緩和が必要な DDoS 攻撃の存在を示している可能性のあるトラフィックの増加をモニタリングします。毎分、各 AWS リソー

スへのトラフィックが評価されます。リソースへのトラフィックが上昇した場合は、リソースの容量を測定するために追加のチェックが実行されます。

Shield は次の標準チェックを実行します。

- Amazon Elastic Compute Cloud (Amazon EC2) インスタンス、Amazon EC2 インスタンスにアタッチされた EIP – Shield は保護されたリソースから容量を取得します。容量は、ターゲットのインスタンスタイプ、インスタンスサイズ、およびインスタンスが拡張ネットワークを使用しているかどうかなどの他の要因によって異なります。
- Classic Load Balancer と Application Load Balancer – Shield はターゲットのロードバランサーノードから容量を取得します。
- Network Load Balancer にアタッチされた EIP – Shield はターゲットのロードバランサーから容量を取得します。容量は、ターゲットのロードバランサーのグループ設定とは無関係です。
- AWS Global Accelerator 標準アクセラレータ — Shield は、エンドポイントの設定に基づいて容量を取得します。

これらの評価は、ポートやプロトコルなど、ネットワークトラフィックの複数のディメンションにわたって行われます。ターゲットリソースの容量を超えると、Shield は DDoS 緩和策を実行します。Shield によって実施された緩和策は DDoS トラフィックを削減しますが、完全に排除することはできない場合があります。Shield は、既知の DDoS 攻撃ベクトルと一致するトラフィックディメンションでリソースの容量がわずかに超過した場合も、緩和策を講じることがあります。Shield は、この緩和策を有効期限 (TTL) を設けて実施し、攻撃が続く限り延長されます。

> **Note**
>
> Shield によって実施された緩和策は DDoS トラフィックを削減しますが、完全に排除することはできない場合があります。Shield AWS Network Firewall iptables にオンホストファイアウォールなどのソリューションを追加して、アプリケーションにとって有効でないトラフィックや正当なエンドユーザーが生成したトラフィックをアプリケーションが処理しないようにすることができます。

Shield Advanced の保護では、既存の Shield の検出アクティビティに次が追加されます。

- [Lower detection thresholds] (検出しきい値を下げる) – Shield Advanced は、計算された容量の半分に緩和策を実施します。これにより、ゆっくり増加する攻撃をより迅速に緩和し、より曖昧なボリュームシグネチャを持つ攻撃を緩和できます。

- [Intermittent attack protection] (断続的な攻撃からの保護) – Shield Advanced は、攻撃の頻度と期間に基づいて、指数関数的に増加する有効期限 (TTL) で緩和策を実施します。これにより、リソースが頻繁にターゲットとなり、短いバーストで攻撃が発生する際に、緩和策が長く維持されます。
- [Health-based detection] (ヘルスベースの検出) – Route 53 ヘルスチェックを Shield Advanced で保護されたリソースに関連付けると、ヘルスチェックのステータスが検出ロジックで使用されます。検出されたイベント中、ヘルスチェックが正常である場合、Shield Advanced では、緩和策を行う前に、そのイベントが攻撃であるというより強力な確信が必要です。代わりにヘルスチェックが異常な場合は、信頼が確立される前でも Shield Advanced が緩和策を講じることがあります。この機能は、誤検出を回避するのに役立つとともに、アプリケーションに影響する攻撃への迅速な対応を提供します。Shield Advanced を使用したヘルスチェックの詳細については、「ヘルスチェックを使用したHealth ベースの検出」を参照してください。

アプリケーションレイヤーの脅威の検出ロジック

AWS Shield Advanced 保護されている Amazon CloudFront ディストリビューションとアプリケーションロードバランサーのウェブアプリケーションレイヤー検出機能を提供します。これらのリソースタイプを Shield Advanced で保護する場合、 AWS WAF ウェブ ACL を保護に関連付けて、ウェブアプリケーションレイヤーの検出を有効にすることができます。Shield Advanced は、関連付けられたウェブ ACL のリクエストデータを消費し、アプリケーションのトラフィックベースラインを構築します。ウェブアプリケーションレイヤーの検出は、Shield Advanced と AWS WAFのネイティブ統合に依存しています。Shield Advanced AWS WAF で保護されたリソースへのウェブ ACL の関連付けなど、アプリケーション層保護の詳細については、を参照してください。AWS Shield Advanced アプリケーション層 (レイヤー 7) 保護

ウェブアプリケーションレイヤーの検出では、Shield Advanced はアプリケーショントラフィックをモニタリングし、異常を検出する過去のベースラインと比較します。このモニタリングは、トラフィックの総量と構成をカバーします。DDoS 攻撃を受けている最中、トラフィックの量と構成の両方が変化することが想定され、Shield Advanced では、イベントを宣言するために両方における統計的に有意な偏差が必要です。

Shield Advanced は、過去のタイムウィンドウに照らして測定を実行します。このアプローチを使用すると、トラフィック量の正当な変化や、毎日同じ時間に提供される販売など、想定されるパターンに一致するトラフィックの変化による誤検出の通知が減少します。

> **Note**
>
> Shield Advanced が通常の正当なトラフィックパターンを表すベースラインを確立する時間を与えることで、Shield Advanced 保護の誤検出を回避できます。Shield Advanced は、

ウェブ ACL を保護対象リソースに関連付けると、ベースラインの情報の収集を開始します。ウェブトラフィックに異常なパターンを引き起こす可能性のある予定イベントの少なくとも 24 時間前に、ウェブ ACL を保護リソースに関連付けます。Shield Advanced のウェブアプリケーションレイヤー検出は、30 日間の通常のトラフィックが観測された場合に最も正確です。

Shield Advanced がイベントを検出するのにかかる時間は、トラフィック量で見られる変化の量の影響を受けます。量の変化が少ない場合、Shield Advanced は、イベントが発生していることについてのより強い確信を持つために、トラフィックをより長い期間にわたって観察します。量の変化が大きい場合、Shield Advanced はイベントをより迅速に検出してレポートします。

ウェブ ACL 内のレートベースのルールは、自分で追加したか Shield Advancedの自動アプリケーションレイヤー軽減機能によって追加したかにかかわらず、攻撃が検出可能なレベルに達する前に攻撃を軽減できます。アプリケーション層の自動 DDoS 軽減の詳細については、を参照してください。Shield Advanced アプリケーションレイヤー DDoS 自動緩和

> **Note**
>
> トラフィックの増加やロードに対応してスケールするようにアプリケーションを設計して、小規模なリクエストフラッドの影響を受けないようにすることができます。Shield Advanced を使用すると、保護されたリソースはコスト保護の対象となります。これは、DDoS 攻撃の結果として発生する可能性のあるクラウド料金の想定外の増加を防ぐのに役立ちます。Shield Advanced のコスト保護の詳細については、「でのクレジットのリクエスト AWS Shield Advanced」を参照してください。

アプリケーション内の複数のリソースの検出ロジック

AWS Shield Advanced 保護グループを使用して、同じアプリケーションの一部である保護対象リソースのコレクションを作成できます。グループに配置する保護されたリソースを選択するか、同じタイプのすべてのリソースを 1 つのグループとして扱うことを指定できます。例えば、すべての Application Load Balancer のグループを作成できます。保護グループを作成すると、Shield Advanced 検出は、グループ内の保護されたリソースに関するすべてのトラフィックを集約します。これは、トラフィック量は少ないが、集約されたボリュームが大きいリソースが多数ある場合に有益です。また、保護されたリソース間でトラフィックが転送されるブルーグリーンデプロイメント向けに、保護グループを使用してアプリケーションベースラインを保持することもできます。

次のいずれかの方法で保護グループ内のトラフィックを集約することを選択できます。

- [Sum] (合計) – この集計は、保護グループ内のリソース全体のすべてのトラフィックを合計します。この集約を使用して、新しく作成されたリソースに既存のベースラインが確実に存在しているようにしたり、検出の感度を下げて、誤検出を防いだりすることができます。
- [Mean] (平均) – この集計は、保護グループ全体のすべてのトラフィックの平均を使用します。この集約は、ロードバランサーのように、リソース間のトラフィックが均一であるアプリケーションに使用できます。
- [Max] (最大) – この集約は、保護グループ内のリソースの中で最も高いトラフィックを使用します。この集約は、保護グループにアプリケーションの階層が複数ある場合に使用できます。たとえば、 CloudFront ディストリビューション、そのApplication Load Balancer のオリジン、アプリケーションロードバランサーの Amazon EC2 インスタンスターゲットを含む保護グループがあるとします。

また、保護グループを使用して、複数のインターネットに接続する Elastic IP または AWS Global Accelerator 標準のアクセラレーターをターゲットとする攻撃に対して、Shield Advanced が緩和策を実施する速度を向上させることもできます。保護グループ内の 1 つのリソースがターゲットになると、Shield Advanced は、グループ内の他のリソースについての信頼を確立します。これにより、Shield Advanced 検出がアラート状態になり、追加の緩和策の作成に必要な時間を短縮できます。

保護グループの詳細については、「AWS Shield Advanced 保護グループ」を参照してください。

AWS Shield イベントの軽減方法

アプリケーションを保護する緩和ロジックは、アプリケーションのアーキテクチャによって異なります。Amazon と Amazon CloudFront Route 53 でウェブアプリケーションを保護すると、ウェブと DNS のユースケースに特有の、サービスのすべてのトラフィックを保護する緩和策の恩恵を受けることができます。 AWS アプリケーションのエントリポイントがリージョンで実行されるリソースである場合、緩和ロジックはサービス、リソースタイプ、および用途によって異なります。 AWS Shield Advanced

AWS DDoS軽減システムはSShield dのエンジニアによって開発され、サービスと緊密に統合されています。 AWS エンジニアは、ターゲットリソースの容量や健全性など、アーキテクチャの側面を考慮に入れます。Shield エンジニアは、DDoS 緩和システムの有効性とパフォーマンスを継続的に監視し、新しい脅威が発見または予測されたときに迅速に対応できます。

トラフィックの増加やロードに対応してスケールするようにアプリケーションを設計して、小規模なリクエストフラッドの影響を受けないようにする手助けをします。Shield Advanced を使用してリソースを保護すると、DDoS 攻撃の結果として発生する可能性のあるクラウド料金の予想外の増加を防ぐことができます。

インフラストラクチャの緩和

インフラストラクチャーレイヤーの攻撃では、 AWS Shield DDoS AWS 軽減システムがネットワークの境界とエッジロケーションに配置されています。 AWS AWS インフラストラクチャー全体に複数のレベルのセキュリティ制御を配置することで、 defense-in-depth クラウドアプリケーションに提供されます。

Shield により、インターネットからの進入のすべてのポイントで DDoS 軽減システムを維持されます。Shield は DDoS 攻撃を検出すると、進入ポイントごとに、同じ場所にある DDoS 緩和システムを介してトラフィックを再ルーティングします。これにより、観測可能な追加のレイテンシーが発生することはなく、 AWS すべてのリージョンとすべてのエッジロケーションで TeraBits 1 秒あたり 100 Tbps (Tbps) を超える緩和能力が得られます。Shield は、トラフィックを外部またはリモートのスクラビングセンターに再ルーティングすることなく、リソースの可用性を保護します。これにより、レイテンシーが増加する可能性があります。

- DDoS 緩和システムは、 AWS あらゆるサービスやリソースについて、 AWS ネットワークの境界で、インターネットからのインフラストラクチャ層攻撃を軽減します。Shield による検出または Shield Response Team (SRT) のエンジニアによる通知があると、システムは緩和を実行します。
- AWS エッジロケーションでは、DDoS 軽減システムが Amazon CloudFront ディストリビューションと Amazon Route 53 ホストゾーンに転送されるすべてのパケットを、その送信元に関係なく継続的に検査します。必要な場合、システムはウェブおよび DNS トラフィック向けに特別に設計された緩和策を適用します。Amazon CloudFront と Amazon Route 53 を使用してウェブアプリケーションを保護することのもう1つの利点は、Shield 検出からの信号を必要とせずに DDoS 攻撃を即座に軽減できることです。

アプリケーションレイヤーの緩和

Shield アドバンスドは、Shield アドバンスド保護を有効にした Amazon CloudFront ディストリビューションとアプリケーションロードバランサーにウェブアプリケーションレイヤーの軽減を提供します。保護を有効にすると、 AWS WAF ウェブ ACL をリソースに関連付けて、ウェブアプリケーションレイヤーの検出を有効にします。さらに、自動アプリケーションレイヤー軽減を有効にするオプションがあります。これは、DDoS 攻撃中に保護を管理するよう Shield Advanced に指示します。

Shieldは、Shield Advancedと自動アプリケーション層軽減を有効にしたリソースに対するアプリケーション層攻撃に対してのみカスタム緩和を提供します。Shield Advancedは、自動軽減機能により、 AWS WAF 既知のDDoSソースからのリクエストにレート制限を適用し、 AWS WAF 検出されたDDoS攻撃に対応してカスタムプロテクションを自動的に追加および管理します。このタイプの緩和の詳細については、「Shield Advanced が自動緩和を管理する方法」を参照してください。。

ウェブ ACL 内のレートベースのルールは、自分で追加したか、Shield Advancedの自動アプリケーションレイヤー軽減機能によって追加されたかにかかわらず、攻撃が検出可能なレベルに達する前に攻撃を軽減できます。検出の詳細については、を参照してください。アプリケーションレイヤーの脅威の検出ロジック

緩和機能

AWS Shield DDoS 対策の主な機能は次のとおりです。

- パケット検証 — これにより、検査されたすべてのパケットが期待される構造に適合し、そのプロトコルに対して有効であることが保証されます。サポートされているプロトコル検証には、IP、TCP (ヘッダーとオプションを含む)、UDP、ICMP、DNS、および NTP が含まれます。
- アクセスコントロールリスト (ACL) と Shaper — ACL は特定の属性に対してトラフィックを評価し、一致するトラフィックをドロップするか、シェーパーにマッピングします。シェーパーは一致するトラフィックのパケットレートを制限し、宛先に到達する量を抑えるために余分なパケットをドロップします。 AWS Shield Detection and Shield Response Team (SRT) のエンジニアは、予想されるトラフィックには専用のレート割り当てを行い、既知のDDoS攻撃ベクトルと一致する属性を持つトラフィックにはより制限の厳しいレート割り当てを行うことができます。ACL が照合できる属性には、パケットペイロード内のポート、プロトコル、TCP フラグ、宛先アドレス、送信元の国、および任意のパターンが含まれます。
- 疑惑のスコアリング — これにより、Shield が期待するトラフィックに関する知識を使用して、すべてのパケットにスコアを適用されます。既知の正常なトラフィックのパターンに近いパケットには、より低い疑惑スコアが割り当てられます。既知の不良トラフィック属性を観察すると、パケットの疑惑スコアが高まる可能性があります。レート制限パケットが必要な場合、Shield は疑惑スコアが高いパケットからドロップします。これは、Shield が誤検知を回避しながら、既知の DDoS 攻撃とゼロデイ攻撃の両方を軽減するのに役立ちます。
- TCP SYN プロキシ — これにより、TCP SYN Cookie が保護されたサービスに渡すのを許可する前に新しい接続に挑戦するために TCP SYN Cookie を送信することで、TCP SYN フラッドに対する保護が提供されます。Shield DDoS 緩和によって提供される TCP SYN プロキシはステートレスであり、ステートを使い果たすことなく、既知の最大の TCP SYN フラッド攻撃を軽減でき

ます。これは、 AWS クライアントと保護対象サービスの間で継続的なプロキシを維持するのではなく、サービスと統合して接続状態を引き継ぐことで実現されます。TCP SYN プロキシは現在、CloudFront アマゾンとAmazon ルート 53 で使用できます。

- レートの分布 — これにより、保護されたリソースへのトラフィックの入力パターンに基づいて、ロケーションシェーパーの値を継続的に調整します。これにより、 AWS ネットワークに均等に入らない可能性のある顧客トラフィックのレート制限が防止されます。

AWS Shield CloudFront と Route 53 の緩和ロジック

Shield の DDoS 対策は、および Route 53 CloudFront のトラフィックを継続的に検査します。これらのサービスは、 AWS 世界中に分散したエッジロケーションのネットワークから運用されます。これにより、Shield の DDoS 軽減機能に幅広くアクセスでき、エンドユーザーにより近いインフラストラクチャからアプリケーションを配信できます。

- CloudFront— Shield DDoS対策では、ウェブアプリケーションに有効なトラフィックのみがサービスに送信されます。これにより、UDP リフレクション攻撃など、多くの一般的な DDoS ベクトルに対して自動的に保護されます。

  CloudFront アプリケーションオリジンへの持続的な接続を維持し、Shield TCP SYN プロキシ機能との統合により TCP SYN フラッドが自動的に軽減され、トランスポート層セキュリティ (TLS) はエッジで終了します。これらの機能を組み合わせることで、アプリケーションオリジンは整形式のウェブリクエストのみを受信し、下位層の DDoS 攻撃、接続フラッド、および TLS 不正使用から保護されます。

  CloudFront DNS トラフィック方向とエニーキャストルーティングを組み合わせて使用します。これらの手法は、ソースに近い攻撃を緩和し、障害を分離し、容量へのアクセスを確保して既知の最大の攻撃を軽減することで、アプリケーションの復元力を向上させます。
- Route 53 — Shield 緩和では、有効な DNS リクエストのみがサービスに到達することを許可します。Shield は、既知の正常なクエリに優先順位を付け、疑わしいまたは既知の DDoS 攻撃属性を含むクエリの優先度を下げる、疑わしいスコアリングを使用して DNS クエリのフラッドを軽減します。

  Route 53 は、シャッフルシャーディングを使用して、IPv4 と IPv6 の両方のホストゾーンに 4 つのリゾルバー IP アドレスの一意のセットを提供します。各 IP アドレスは、Route 53 ロケーションの異なるサブセットに対応します。各ロケーションサブセットは、他のサブセットのインフラストラクチャと部分的にしか重複しない権限を持つ DNS サーバーで構成されます。これにより、何らかの理由でユーザークエリが失敗した場合、再試行時に正常に処理されるようになります。

Route 53 は、エニーキャストルーティングを使用して、ネットワークの近接度に基づいて DNS クエリを最も近いエッジロケーションに送信します。エニーキャストはまた、DDoS トラフィックを多くのエッジロケーションにファンアウトし、攻撃が単一のロケーションに集中するのを防ぎます。

迅速な緩和に加えて、 CloudFront Route 53 は世界中に分散された Shield のキャパシティへの幅広いアクセスを提供します。これらの機能を活用するには、これらのサービスを動的または静的ウェブアプリケーションのエントリーポイントとして使用します。

Route 53 を使用してウェブアプリケーションを保護する方法の詳細については、「Amazon CloudFront CloudFront と Amazon Route 53 を使用して DDoS 攻撃から動的ウェブアプリケーションを保護する方法」を参照してください。Route 53 での障害分離の詳細については、「グローバルフォールトアイソレーションのケーススタディ」を参照してください。

AWS ShieldAWS リージョンの緩和ロジック

AWS リージョンで起動されたリソースは、SShield AWS Shield dのリソースレベル検出によって配置されたDDoS軽減システムによって保護されます。リージョンリソースには、Elastic IP (EIP)、クラシックロードバランサー、アプリケーションロードバランサーなどがあります。

緩和を実施する前に、Shield はターゲットリソースとその容量を特定します。Shield は、キャパシティを使用して、緩和がリソースに転送できる最大合計トラフィックを決定します。アクセスコントロールリスト（ACL）および緩和策内のその他のシェーパによって、既知の DDoS 攻撃ベクトルに一致するトラフィックや、大量の受信が予想されないトラフィックなど、一部のトラフィックで許可されるボリュームが減少する可能性があります。これにより、UDP リフレクション攻撃や TCP SYN フラグまたは FIN フラグを持つ TCP トラフィックに対して、緩和によって許可されるトラフィック量がさらに制限されます。

Shield は、リソースタイプごとにキャパシティを決定し、緩和を別々に配置します。

- Amazon EC2 インスタンス、または Amazon EC2 インスタンスにアタッチされている EIP の場合、Shield はインスタンスタイプおよびその他のインスタンス属性 (インスタンスで拡張ネットワーキングが有効になっているかどうかなど) に基づいて容量を計算します。
- Application Load Balancer または Classic Load Balancer の場合、Shield はロードバランサーのターゲットノードごとに個別に容量を計算します。これらのリソースに対する DDoS 攻撃の緩和は、Shield DDoS 緩和とロードバランサーによる自動スケーリングの組み合わせによって提供されます。Shield Response Team (SRT) がApplication Load Balancer またはClassic Load Balancer の

リソースに対する攻撃に従事している場合、追加の保護対策としてスケーリングを加速する可能性があります。

- Shield は、 AWS AWS 基盤となるインフラストラクチャの利用可能な容量に基づいて一部のリソースの容量を計算します。これらのリソースタイプには、ネットワークロードバランサー (NLB) や、ゲートウェイロードバランサーまたはを経由してトラフィックをルーティングするリソースが含まれます。 AWS Network Firewall

> **Note**
>
> Shield Advanced で保護されている EIP をアタッチして、ネットワークロードバランサーを保護します。SRT と連携して、基盤となるアプリケーションの予想されるトラフィックと容量に基づくカスタム緩和を構築できます。

Shield が緩和策を設定すると、Shield が緩和ロジックで定義した初期レート制限が、すべての Shield DDoS 緩和システムに等しく適用されます。たとえば、Shield が 100,000 パケット/秒 (pps) の制限で緩和を設定した場合、最初はすべてのロケーションで 100,000 pps を許可します。次に、Shield は継続的に緩和メトリクスを集約してトラフィックの実際の比率を決定し、その比率を使用して各ロケーションのレート制限を調整します。これにより、誤検出を防ぎ、緩和が過度に許容されないようにします。

AWS ShieldAWS Global Accelerator 標準アクセラレータの緩和ロジック

Shield 緩和は、有効なトラフィックがグローバルアクセラレータの標準アクセラレータのリスナーエンドポイントに到達することのみを許可します。スタンダードアクセラレータは世界中に展開され、 AWS どのリージョンのリソースにもトラフィックをルーティングできる IP アドレスを提供します。 AWS Shield が Global Accelerator への緩和に対して適用するレート制限は、標準アクセラレータがトラフィックをルーティングするリソースの容量に基づいています。Shield は、総トラフィックが決められたレートを超えたとき、および既知の DDoS ベクトルに対してそのレートの割合を超えた場合に、緩和を実行します。

標準アクセラレータを設定するときは、アプリケーションのトラフィックをルーティングする AWS リージョンごとにエンドポイントグループを定義します。Shield が緩和策を設定すると、各エンドポイントグループの容量が計算され、それに応じて各 Shield DDoS 緩和システムのレート制限が更新されます。料金は、 AWS トラフィックがインターネットからリソースにどのようにルーティングされるかについてShield が行った仮定に基づいて、場所ごとに異なります。エンドポイントグループのキャパシティは、グループ内のリソースの数に、グループ内の任意のリソースの最小容量を乗じ

て計算されます。一定の間隔で、Shield はアプリケーションの容量を再計算し、必要に応じてレート制限を更新します。

> **Note**
>
> トラフィックダイヤルを使用してエンドポイントグループに誘導されるトラフィックの割合を変更しても、Shield DDoS 軽減システムに対してレート制限を計算または配布する方法は変わりません。トラフィックダイヤルを使用する場合は、リソースタイプと数量について相互にミラーリングするようにエンドポイントグループを構成します。これにより、Shield によって計算されたキャパシティが、アプリケーションのトラフィックを処理しているリソースを代表するようにできます。

Global Accelerator のエンドポイントグループとトラフィックダイヤルの詳細については、「AWS Global Accelerator 標準アクセラレーターのエンドポイントグループ」を参照してください。

AWS Shield Advanced エラスティック IP の緩和ロジック

Elastic IP (EIP) を保護すると AWS Shield Advanced、Shield アドバンスドは DDoS イベント時に Shield が講じる緩和策を強化します。Shield Advanced DDoS 緩和システムは、EIP が関連付けられているパブリックサブネットのネットワーク ACL (NACL) 設定を複製します。たとえば、NACL がすべての UDP トラフィックをブロックするように設定されている場合、Shield Advanced はそのルールを Shield が配置する緩和にマージします。

この追加機能は、アプリケーションに対して有効ではないトラフィックによる可用性リスクを回避するのに役立ちます。NACL を使用して、個々の送信元 IP アドレスまたは送信元 IP アドレス CIDR 範囲をブロックすることもできます。これは、分散されていない DDoS 攻撃の緩和ツールとして役立ちます。また、エンジニアの介入に頼ることなく、独自の許可リストを簡単に管理したり、アプリケーションと通信してはいけない IP アドレスをブロックしたりできます。 AWS

AWS Shield Advanced Web アプリケーションの緩和ロジック

AWS Shield Advanced AWS WAF ウェブアプリケーション層の攻撃を軽減するために使用します。AWS WAF はShield アドバンスドに追加費用なしで含まれています。

アプリケーションレイヤー標準保護

Amazon CloudFront ディストリビューションまたはApplication Load Balancer を Shield アドバンスドで保護する場合、 AWS WAF ウェブ ACL をまだ関連付けていない場合は、Shield アドバンスドを

使用して保護対象リソースにウェブ ACL を関連付けることができます。ウェブ ACL をまだ設定していない場合は、Shield Advanced コンソールウィザードを使用して作成して、レートベースのルールを追加できます。レートベースのルールは、各 IP アドレスの 5 分間の時間枠あたりの要求数を制限し、ウェブアプリケーション層のリクエストのフラッドに対する基本的な保護を提供します。最低値は 100 からレートを設定できます。詳細については、「Shield AWS WAF アドバンストアプリケーションレイヤーのウェブ ACL とレートベースのルール」を参照してください。

AWS WAF このサービスを使用してウェブ ACL を管理することもできます。これにより AWS WAF、ウェブ ACL の設定を拡張して、特定のウェブリクエストコンポーネントの文字列の一致やパターンを調べたり、リクエストやレスポンスのカスタム処理を追加したり、リクエスト元の位置情報との照合などを行うことができます。ルールの詳細については AWS WAF 、「」を参照してください。AWS WAF 規則

アプリケーションレイヤーの自動緩和

保護を強化するには、Shield Advanced アプリケーションレイヤーの自動緩和を有効にします。このオプションを使用すると、Shield Advanced は既知の DDoS AWS WAF ソースからのリクエストに対するレート制限ルールを維持し、検出された DDoS 攻撃に対するカスタム緩和策を提供します。

Shield Advanced は、保護されたリソースに対する攻撃を検出すると、アプリケーションへの通常のトラフィックから攻撃トラフィックを分離する攻撃シグネチャの特定を試みます。Shield Advanced は、識別された攻撃シグネチャを攻撃対象のリソースおよび同じウェブ ACL に関連付けられている他のリソースについての過去のトラフィックパターンに照らして評価します。

Shield Advanced は、攻撃シグネチャが DDoS 攻撃に関係するトラフィックのみを隔離していると判断した場合、関連するウェブ ACL AWS WAF 内のルールにシグネチャを実装します。Shield Advanced に、一致したトラフィックのみをカウントする、またはブロックする緩和を配置するように指示できます。この設定はいつでも変更できます。Shield Advanced は、緩和ルールが不要になったと判断すると、そのルールをウェブ ACL から削除します。アプリケーションレイヤーイベントの緩和の詳細については、「Shield Advanced アプリケーションレイヤー DDoS 自動緩和」を参照してください。

Shield Advanced アプリケーションレイヤー の緩和の詳細については、「AWS Shield Advanced アプリケーション層 (レイヤー 7) 保護」を参照してください。

基本的な DDoS に対する耐性が高いアーキテクチャの例

DDoS レジリエンシーとは、正規ユーザーに引き続きサービスを提供しながら、Distributed Denial of Service (DDoS) 攻撃に耐えるアプリケーションアーキテクチャの機能です。高い耐障害性を持つア

プリケーションは、エラーやレイテンシーなどのパフォーマンスメトリクスへの影響を最小限に抑えながら、攻撃を受けている最中も引き続き利用可能な状態を維持できます。このセクションでは、いくつかの一般的なアーキテクチャの例を示し、 AWS および Shield Advanced によって提供される DDoS 検出および緩和機能を使用して DDoS レジリエンシーを高める方法について説明します。

このセクションのアーキテクチャの例では、デプロイされたアプリケーションに DDoS レジリエンシーの最大のメリットを提供する AWS のサービスに焦点を当てています。主なサービスには、次のようなメリットがあります。

- グローバルに分散されたネットワーク容量へのアクセス — Amazon CloudFront AWS Global Accelerator、および Amazon Route 53 サービスにより、 AWS グローバルエッジネットワーク全体でインターネットへのアクセスと DDoS 軽減機能が提供されます。これは、大規模にテラビットに達する可能性がある、比較的大きなボリューム攻撃を緩和するのに有益です。 AWS アプリケーションはどのリージョンでも実行でき、これらのサービスを使用して正規ユーザーの可用性を保護し、パフォーマンスを最適化できます。
- ウェブアプリケーションレイヤーの DDoS 攻撃のベクトルに対する保護 – ウェブアプリケーションレイヤーの DDoS 攻撃は、アプリケーションのスケールとウェブアプリケーションファイアウォール (WAF) の組み合わせを使用することによって最も効果的に緩和されます。Shield Advancedは、 AWS WAF からのウェブリクエスト検査ログを使用して異常を検出します。異常は、自動的に、 AWS またはShield Response Team (SRT) との連携により軽減できます。自動緩和は、デプロイされた AWS WAF レートベースのルールと、Shield Advanced アプリケーションレイヤー DDoS 自動緩和を通じて利用できます。

これらの例を確認することに加えて、「AWS Best Practices for DDoS Resiliency」(DDoS レジリエンシーに関するのベストプラクティス) で該当するベストプラクティスを確認し、それに従います。

一般的なウェブアプリケーションの DDoS レジリエンシーの例

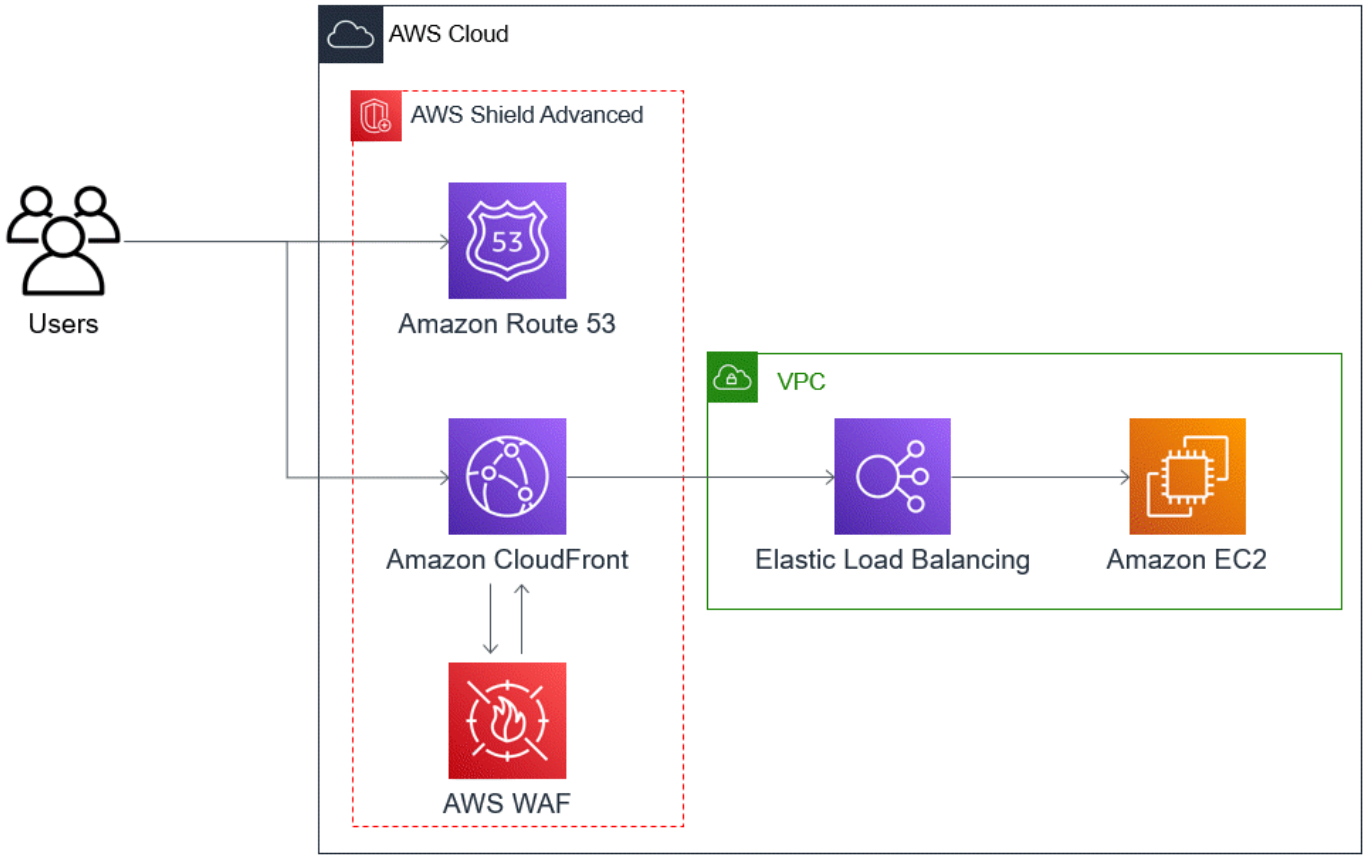
AWS どのリージョンでもウェブアプリケーションを構築でき、 AWS そのリージョンで提供される検出機能と軽減機能によって自動的に DDoS 防御を受けることができます。

この例は、Classic Load Balancer、Application Load Balancer、Network Load Balancer、 AWS Marketplace ソリューション、または独自のプロキシレイヤーなどのリソースを使用して、エンドユーザーをウェブアプリケーションにルーティングするアーキテクチャ向けです。 AWS WAF これらのウェブアプリケーションリソースとユーザーの間に Amazon Route 53 ホストゾーン、Amazon CloudFront ディストリビューション、ウェブ ACL を挿入することで DDoS 耐障害性を向上させることができます。これらの挿入により、アプリケーションのオリジンが難読化され、エンドユーザーにより近いリクエストを処理し、アプリケーションレイヤーのリクエストのフラッドを検出して緩和で

きます。 CloudFront と Route 53 を使用して静的または動的コンテンツをユーザーに提供するアプリケーションは、インフラストラクチャ層の攻撃をリアルタイムで軽減する統合された完全インラインの DDoS 軽減システムによって保護されます。

これらのアーキテクチャ上の改善を実施すれば、Route 53 CloudFront のホストゾーンとディストリビューションを Shield Advanced で保護できるようになります。 CloudFront ディストリビューションを保護する場合、Shield AWS WAF Advancedはウェブ ACLを関連付けてレートベースのルールを作成するように求めるプロンプトを表示し、アプリケーションレイヤーの自動DDoS軽減または積極的な関与を有効にするオプションも提供します。プロアクティブなエンゲージメントとアプリケーションレイヤー DDoS 自動緩和では、リソースに関連付ける Route 53 ヘルスチェックを使用します。これらのオプションの詳細については、「でのリソース保護 AWS Shield Advanced」を参照してください。

次の参照図は、ウェブアプリケーション用のこの DDoS 回復アーキテクチャを示しています。



このアプローチがウェブアプリケーションに提供するメリットには、次のものがあります。

- 頻繁に使用されるインフラストラクチャレイヤー (レイヤー 3 およびレイヤー 4) の DDoS 攻撃に対する保護。検出の遅延はありません。さらに、リソースが頻繁にターゲットになってい

る場合、Shield Advanced はより長期にわたって緩和策を実施します。また、Shield Advanced は、Network ACL (NACL) から推測されるアプリケーションコンテキストを使用して、望ましくないトラフィックがさらにアップストリームするのをブロックします。これにより、ソースにより近い障害を隔離でき、正当なエンドユーザーへの影響が最小限に抑えられます。

- TCP SYN フラッドに対する保護。Route 53 と統合された DDoS 軽減システムは CloudFront、TCP SYN プロキシ機能を備えており、新しい接続を試みても問題なく、 AWS Global Accelerator 正当なユーザーのみにサービスを提供します。
- DNS アプリケーションレイヤー攻撃に対する保護 (Route 53 は権威 DNS レスポンスを処理する責任があるため)。
- ウェブアプリケーションレイヤーリクエストのフラッドに対する保護。 AWS WAF ウェブ ACL で設定するレートベースのルールは、ルールで許可されているリクエストを超えるリクエストを送信すると、ソース IP をブロックします。
- このオプションを有効にした場合、 CloudFront ディストリビューションのアプリケーション層の DDoS 軽減が自動的に行われます。Shield Advanced は自動 DDoS 軽減機能により、 AWS WAF ディストリビューションの関連するウェブ ACL にレートベースのルールを維持し、既知の DDoS ソースからのリクエストの量を制限します。さらに、Shield Advanced は、アプリケーションのヘルスに影響を及ぼすイベントを検出すると、ウェブ ACL の緩和ルールを自動的に作成、テスト、および管理します。
- Shield Response Team (SRT) とのプロアクティブなエンゲージメント (このオプションを有効にした場合)。Shield Advanced がアプリケーションのヘルスに影響を与えるイベントを検出すると、SRT は、提供された連絡先情報を使用して、お客様のセキュリティチームまたはオペレーションチームとプロアクティブに対応します。SRT はトラフィックのパターンを分析し、ルールを更新して攻撃を阻止できます。 AWS WAF

TCP および UDP アプリケーションの DDoS レジリエンシーの例

この例は、Amazon Elastic Compute Cloud (Amazon EC2) インスタンスまたは Elastic IP (EIP) アドレスを使用する AWS リージョンの TCP および UDP アプリケーション向けの DDoS に対する耐性の高いアーキテクチャを示しています。

この一般的な例に従って、次のアプリケーションタイプの DDoS レジリエンシーを改善できます。

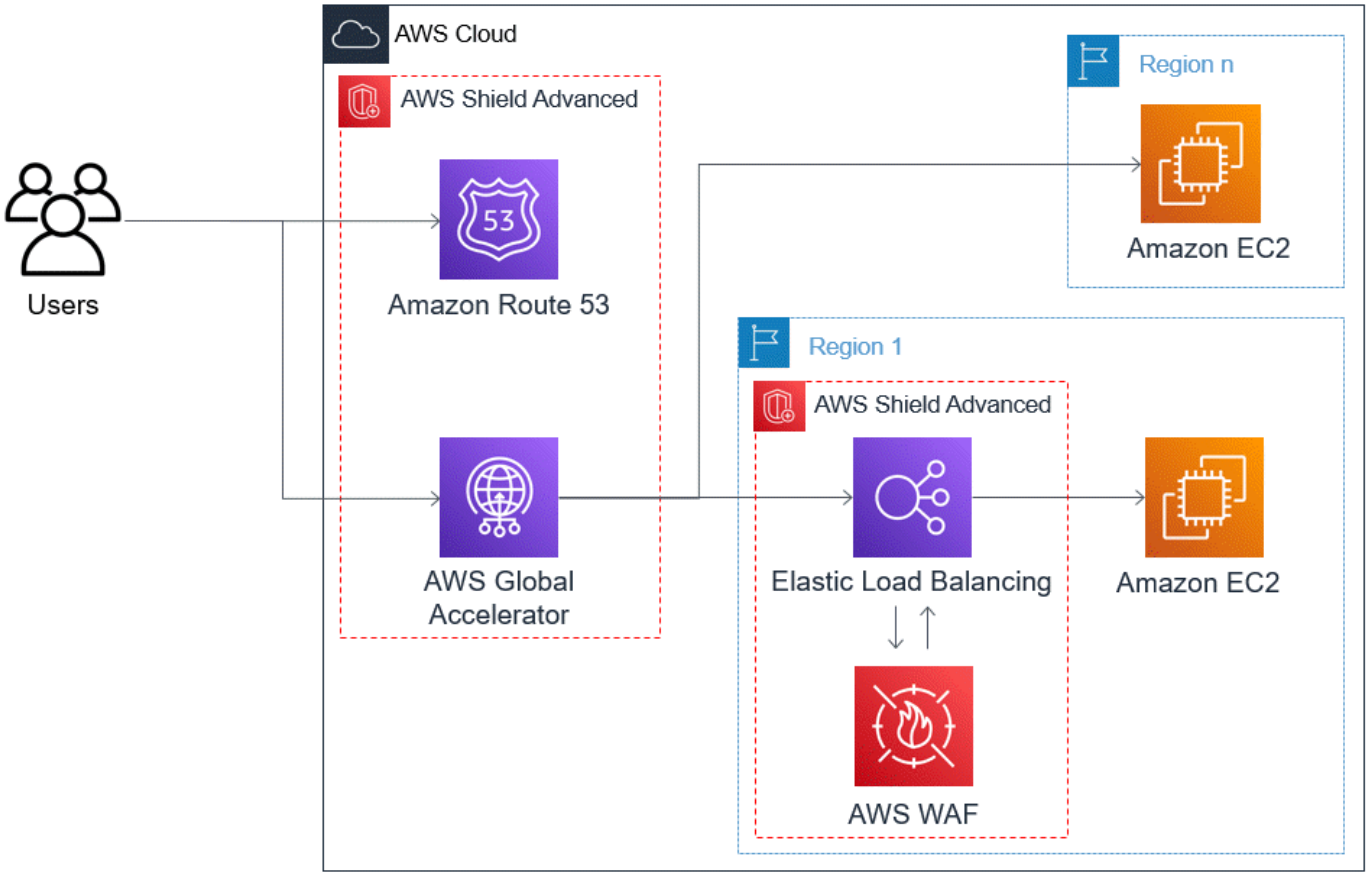
- TCP または UDP アプリケーション。ゲーム、IoT、およびボイスオーバー IP に使用されるアプリケーションはその一例です。
- 静的 IP アドレスを必要とするウェブアプリケーション、または Amazon CloudFront がサポートしていないプロトコルを使用するウェブアプリケーション。たとえば、アプリケーションに、ユー

ザーがファイアウォールの許可リストに追加でき、 AWS 他の顧客には使用されない IP アドレスが必要な場合があります。

Amazon Route 53 および AWS Global Acceleratorを導入することで、これらのアプリケーションタイプの DDoS レジリエンシーを改善できます。これらのサービスは、エンドユーザーをアプリケーションにルーティングでき、 AWS グローバルエッジネットワークを介してルーティングされるエニーキャストである静的 IP アドレスをアプリケーションに提供できます。Global Accelerator 標準アクセラレーターは、エンドユーザーのレイテンシーを最大 60% 改善できます。ウェブアプリケーションを使用している場合は、アプリケーションをApplication Load Balancer で実行し、ウェブ ACL でApplication Load Balancer を保護することで、ウェブアプリケーション層のリクエストフラッドを検出して軽減できます。 AWS WAF

アプリケーションの構築後、Shield Advanced を使用して、Route 53 ホストゾーン、Global Accelerator 標準アクセラレーター、および Application Load Balancer を保護します。アプリケーションロードバランサーを保護すると、 AWS WAF ウェブ ACL を関連付けてレートベースのルールを作成できます。新規または既存の Route 53 ヘルスチェックを関連付けることで、Global Accelerator 標準アクセラレーターと Application Load Balancerの両方のために、SRT とのプロアクティブエンゲージメントを設定できます。オプションの詳細については、「でのリソース保護 AWS Shield Advanced」を参照してください。

次の図は、TCP および UDP アプリケーションの DDoS に対する耐性が高いアーキテクチャの例を示しています。



このアプローチがアプリケーションに提供するメリットには、次のものがあります。

- 最大の既知のインフラストラクチャレイヤー (レイヤー 3 およびレイヤー 4) の DDoS 攻撃に対する保護。攻撃量がアップストリームからの輻輳を引き起こした場合 AWS、その障害は発生源の近くに分離され、正当なユーザーへの影響は最小限に抑えられます。
- DNS アプリケーションレイヤー攻撃に対する保護 (Route 53 は権威 DNS レスポンスを処理する責任があるため)。
- ウェブアプリケーションがある場合、このアプローチはウェブアプリケーションレイヤーのリクエストのフラッドに対する保護を提供します。 AWS WAF ウェブ ACL に設定するレートベースのルールは、ルールで許可されている量を超えるリクエストを送信している間、ソース IP をブロックします。
- Shield Response Team (SRT) とのプロアクティブなエンゲージメント (対象リソースのためにこのオプションを有効にした場合)。Shield Advanced がアプリケーションのヘルスに影響を与えるイベントを検出すると、SRT は、提供された連絡先情報を使用して、お客様のセキュリティチームまたはオペレーションチームとプロアクティブに対応します。

Shield Advanced のユースケースの例

Shield Advanced を使用すると、さまざまなシナリオでリソースを保護できます。ただし、場合によっては、他のサービスを使用するか、他のサービスと Shield Advanced を組み合わせて最高の保護を提供する必要があります。以下は、Shield Advanced AWS やその他のサービスを使用してリソースを保護する方法の例です。

目標	推奨するサービス	関連サービスドキュメント
DDoS 攻撃からウェブアプリケーションと RESTful API を保護する	Amazon CloudFront ディストリビューションとApplication Load Balancer を保護する Shield アドバンス	Elastic Load Balancing ドキュメント、Amazon CloudFront ドキュメント
DDoS 攻撃から TCP ベースのアプリケーションを保護する	AWS Global Accelerator 標準のアクセラレータを保護するシールドアドバンスド。Elastic IPアドレスに接続	AWS Global Accelerator ドキュメント、Elastic Load Balancing ドキュメント
DDoS 攻撃から UDP ベースのゲームサーバーを保護する	Elastic IP アドレスにアタッチされた Amazon EC2 インスタンスを保護する Shield Advanced	Amazon Elastic Compute Cloud のドキュメント

例えば、Shield Advanced を使用して Elastic IP アドレスを保護する場合、Shield Advanced はそれに関連付けられているすべてのリソースを保護します。攻撃中、Shield Advanced はネットワーク ACL をネットワークの境界に自動的に展開します AWS 。ネットワーク ACL がネットワークの境界にある場合、Shield Advanced はより大きな DDoS イベントに対する保護を提供できます。通常、ネットワーク ACL は Amazon VPC 内の Amazon EC2 インスタンスの近くで適用されます。ネットワーク ACL は、Amazon VPC とインスタンスが処理できるだけの大きさの攻撃を緩和できます。Amazon EC2 インスタンスにアタッチされたネットワークインターフェイスが最大 10 Gbps を処理できる場合、10 Gbps を超えるボリュームは遅くなり、そのインスタンスへのトラフィックがブロックされる可能性があります。攻撃を受けている最中に、Shield Advanced はネットワーク ACL を AWS 境界に昇格させ、数テラバイトのトラフィックを処理できます。ネットワーク ACL は、典型的なネットワークの容量を超えてリソースを十分に保護することができます。ネットワーク ACL の詳細については、「ネットワーク ACL」を参照してください。

の開始方法 AWS Shield Advanced

このチュートリアルでは、Shield Advanced コンソール AWS Shield Advanced の使用を開始する手順を説明します。

> **Note**
>
> Shield Advanced にはサブスクリプションが必要ですが、 AWS Shield Standard には必要ありません。Shield Standard で提供される保護は、すべての AWS のお客様に無料でご利用いただけます。

Shield Advanced では、ネットワークレイヤー (レイヤー 3)、トランスポートレイヤー (レイヤー 4)、およびアプリケーションレイヤー (レイヤー 7) 攻撃に対して、高度な DDoS 検出、緩和、保護が可能です。Shield Advanced の詳細については、「AWS Shield Advanced 概要」を参照してください。

AWS 技術コミュニティは、Infrastructure as Code (IaC) ツール AWS CloudFormation と Terraform を使用して Shield Advanced を設定するための自動プロセスの例を公開しました。アカウントがの組織の一部であり AWS Organizations 、Amazon Route 53 または 以外のリソースタイプを保護している場合は、このソリューション AWS Firewall Manager で を使用できます AWS Global Accelerator。このオプションを確認するには、aws-samples / aws-shield-advanced-one-click-deployment のコードリポジトリと、Shield Advanced のワンクリックデプロイのチュートリアルを参照してください。

> **Note**
>
> Distributed Denial of Service (DDoS) イベントが発生する前に Shield Advanced の設定を完了することが重要です。設定を完了すると、アプリケーションが保護され、アプリケーションが DDoS 攻撃の影響を受けた場合に対応する準備ができているようにします。

Shield Advanced の使用を開始するには、次のステップを順番に実行します。

目次

  - 保護のためのヘルスベースの検出を設定する
  - アラームと通知を設定する
  - 保護設定を確認して終了する
- AWS SRT サポートの設定
- DDoS CloudWatch ダッシュボードを作成してアラームを設定する CloudWatch

購読する AWS Shield Advanced

AWS アカウント 保護したいものごとにShield Advancedに登録する必要があります。Shield Standard に登録する必要はありません。

Shield Advanced サブスクリプションの請求

AWS チャネルリセラーの場合は、アカウントチームに相談して情報やガイダンスを受けてください。この請求情報は、 AWS チャネルリセラー以外の顧客向けです。

他のすべてについては、次のサブスクリプションと請求のガイドラインが適用されます。

- AWS Organizations 組織のメンバーであるアカウントの場合、支払いアカウント自体が購読されているかどうかに関係なく、組織の支払いアカウントに対してShield Advanced AWS サブスクリプションを請求します。
- 同じ AWS Organizations 一括請求 (コンソリデーティッドビリング) アカウントファミリー に属する複数のアカウントをサブスクライブする場合、1 つのサブスクリプション料金はファミリー内のすべてのサブスクライブアカウントに対するものです。組織は、すべての AWS アカウント とそのすべてのリソースを所有している必要があります。
- 複数の組織のために複数のアカウントをサブスクライブする場合でも、すべてを所有しているのであれば、すべての組織、アカウント、リソースのサブスクリプション料金の支払いを引き続き 1 回で行うことができます。 AWS アカウントマネージャーまたはサポートに連絡して、1 AWS Shield Advanced つを除くすべての組織のサブスクリプション料金の免除をリクエストしてください。

詳細な料金情報と例については、「AWS Shield の料金表」を参照してください。

以下の方法でサブスクリプションを簡素化します。 AWS Firewall Manager

アカウントが組織の一部である場合は、できればその組織のサブスクリプションと保護を自動化するために AWS Firewall Manager を使用することをおすすめします。Firewall Manager は、Amazon

Route 53 および AWS Global Acceleratorを除くすべての保護されたリソースタイプをサポートします。Firewall Manager を使用するには、「AWS Firewall Manager」と「AWS Firewall ManagerAWS Shield Advanced ポリシー入門」を参照してください。

Firewall Manager を使用しない場合は、保護するリソースを持つアカウントごとに、次の手順を使用してサブスクライブし、保護を追加します。

アカウントを購読するには AWS Shield Advanced

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ にある AWS WAF & Shield コンソールを開きます。
2. AWS Shield ナビゲーションバーで、[Getting started] (はじめに) を選択します。[Subscribe to Shield Advanced] (Shield Advanced をサブスクライブ) を選択します。
3. [Subscribe to Shield Advanced] (Shield Advanced をサブスクライブ) ページで、契約の各条件を読み、条件を承諾する意思を表示するために、すべてのチェックボックスを選択します。一括請求 (コンソリデーティッドビリング) ファミリーのアカウントについては、各アカウントの条件に同意する必要があります。

   > **Important**
   >
   > サブスクライブされている場合、サブスクライブを解除するには、AWS Support に連絡する必要があります。
   > サブスクリプションの自動更新を無効にするには、Shield API UpdateSubscriptionオペレーションまたは CLI コマンド update-subscription を使用する必要があります。

   [Subscribe to Shield Advanced] (Shield Advanced をサブスクライブ) を選択します。これにより、アカウントが Shield Advanced にサブスクライブされ、サービスが有効になります。

お客様のアカウントはサブスクライブされています。次のステップを続行して、Shield Advanced でアカウントのリソースを保護します。

> **Note**
>
> サブスクライブ後、Shield Advanced はリソースを自動的に保護しません。Shield Advanced で保護するリソースを指定して、保護を設定する必要があります。

リソースを追加して、保護したり、保護を設定したりする

Shield Advanced は、Shield Advanced を通じて、または Firewall Manager Shield Advanced ポリシーで指定したリソースのみを保護します。サブスクライブアカウントのリソースは自動的に保護されません。

保護に AWS Firewall Manager Shield アドバンスドポリシーを使用している場合は、この手順を実行する必要はありません。保護するリソースタイプを指定してポリシーを設定すると、Firewall Manager はポリシーの範囲内にあるリソースに自動的に保護を追加します。

Firewall Manager を使用しない場合は、保護するリソースを持つアカウントごとに次の手順を実行します。

Shield Advanced を使用して保護するリソースを選択するには

1. 前の手順のサブスクリプション確認ページから、または [Protected resources] (保護されたリソース) または [Overview] (概要) ページから、[Add resources to protect] (保護するリソースを追加) を選択します。
2. [Choose resources to protect with Shield Advanced] (Shield Advanced で保護するリソースの選択) ページの [Specify the Region and resource types] (リージョンとリソースタイプの指定) で、保護するリソースのリージョンとリソースタイプの仕様を指定します。[All Regions] (すべてのリージョン) を選択すると複数のリージョンのリソースを保護でき、[Global] (グローバル) を選択すると選択範囲をグローバルリソースに絞り込むことができます。保護しないリソースタイプは、すべて選択解除できます。リソースタイプの保護については、「AWS Shield Advanced リソースタイプ別の保護」を参照してください。
3. [Load resources] (リソースをロード) を選択します。Shield Advanced は、[Select Resources] (リソースの選択) セクションに条件に一致する AWS リソースを入力します。
4. [Select Resources] (リソースの選択) セクションでは、リソースリストで検索する文字列を入力して、リソースのリストをフィルタリングできます。

   保護するリソースを選択します。
5. 作成しようとしている Shield Advanced 保護にタグを追加する場合は、[Tags] (タグ) セクションでそれらを指定します。 AWS リソースのタグ付けの詳細については、「タグエディタの使用」を参照してください。
6. [Protect with Shield Advanced] (Shield Advanced で保護) を選択します。これにより、Shield Advanced 保護がリソースに追加されます。

コンソールウィザードの画面を続行して、リソース保護の設定を完了します。

トピック

- でアプリケーション層 (レイヤー 7) の DDoS 保護を設定します AWS WAF
- 保護のためのヘルスベースの検出を設定する
- アラームと通知を設定する
- 保護設定を確認して終了する

でアプリケーション層 (レイヤー 7) の DDoS 保護を設定します AWS WAF

アプリケーション層のリソースを保護するために、Shield Advanced AWS WAF はレートベースのルールを含むウェブ ACL を出発点として使用します。 AWS WAF は、アプリケーション層リソースに転送される HTTP および HTTPS リクエストを監視し、リクエストの特性に基づいてコンテンツへのアクセスを制御できるウェブアプリケーションファイアウォールです。レートベースのルールは、リクエスト集約条件に基づいてトラフィックの量を制限し、アプリケーションに基本的な DDoS 防御を提供します。詳細については、AWS WAF 仕組みおよびレートベースのルールステートメントを参照してください。

また、オプションで Shield Advanced の自動アプリケーションレイヤーの DDoS 談話を有効にして、既知の DDoS ソースから Shield Advanced レート制限リクエストを受け取り、インシデント固有の保護を自動的に提供することもできます。

> **Important**
>
> Shield AWS Firewall Manager アドバンスドポリシーを使用して Shield アドバンスド保護を管理している場合、ここでアプリケーション層保護を管理することはできません。Firewall Manager Shield Advanced ポリシーで管理する必要があります。

Shield アドバンストのサブスクリプションと費用 AWS WAF

Shield アドバンスドサブスクリプションは、Shield AWS WAF アドバンスドで保護するリソースの標準機能を使用するコストをカバーします。Shield Advanced AWS WAF 保護の対象となる標準料金は、ウェブ ACL あたりのコスト、ルールあたりのコスト、およびウェブリクエストインスペクションの 100 万リクエストあたりの基本価格 (最大 1,500 WCU、デフォルトのボディサイズまで) です。

Shield Advanced 自動アプリケーションレイヤー DDoS 軽減を有効にすると、150 ウェブ ACL キャパシティユニット (WCU) を使用するルールグループがウェブ ACL に追加されます。これらの WCU は、ウェブ ACL 内の WCU の使用量に対してカウントされます。詳細については、Shield Advanced アプリケーションレイヤー DDoS 自動緩和、Shield Advanced ルールグループ、およびAWS WAF ウェブ ACL キャパシティーユニット (WCUs)を参照してください。

Shield アドバンスドへのサブスクリプションは、Shield アドバンスドを使用して保護していないリソースの使用には適用されません。 AWS WAF また、 AWS WAF 保護対象リソースの標準外の追加費用も力バーされません。 AWS WAF 非標準費用の例としては、ボットコントロール、CAPTCHA ルールアクション、1,500 個以上の WCU を使用するウェブ ACL、デフォルトの本文サイズを超えるリクエスト本文の検査などがあります。全リストは料金ページに記載されています。 AWS WAF

詳細情報および料金の例については、「Shield の料金」および「AWS WAF の料金」を参照してください。

リージョン用にレイヤー 7 DDoS 保護を設定するには

Shield Advanced では、選択したリソースが配置されている各リージョンについて、レイヤー 7 DDoS 緩和策を設定するオプションを使用できます。複数のリージョンに保護を追加する場合、ウィザードはリージョンごとに次の手順を説明します。

1. [Configure layer 7 DDoS protections] (レイヤー 7 DDoS 保護の設定) ページには、ウェブ ACL にまだ関連付けられていない各リソースが一覧表示されます。これらのそれぞれについて、既存のウェブ ACL を選択するか、新しいウェブ ACL を作成します。すでにウェブ ACL が関連付けられているリソースについては、まず現在のウェブ ACL との関連付けを解除することでウェブ ACL を変更できます。 AWS WAF詳細については、「ウェブ ACL とリソースの関連付けまたは関連付け解除 AWS」を参照してください。

   レートベースのルールがまだないウェブ ACL の場合、設定ウィザードでルールを追加するよう求められます。レートベースのルールは、大量のリクエストを送信している IP アドレスからのトラフィックを制限します。レートベースのルールは、ウェブリクエストのフラッドからアプリケーションを保護するのに役立つとともに、DDoS 攻撃の可能性を示していることがあるトラフィックの急増についてアラートを出すことができます。[Add rate limit rule] (レート制限ルールを追加) を選択し、レート制限とルールアクションを指定して、レートベースのルールをウェブ ACL に追加します。を使用してウェブ ACL に追加の保護を設定できます。 AWS WAF

   レートベースのルールの追加設定オプションを含む、Shield Advanced 保護でのウェブ ACL およびレートベースのルールの使用方法については、「Shield AWS WAF アドバンストアプリケーションレイヤーのウェブ ACL とレートベースのルール」を参照してください。

2. アプリケーションレイヤーの自動DDoS軽減で、Shield Advancedがアプリケーション層リソースに対するDDoS攻撃を自動的に軽減するようにするには、[有効化] を選択し、Shield Advanced AWS WAF がカスタムルールで使用したいルールアクションを選択します。この設定は、このウィザードセッションで管理しているリソースのすべてのウェブ ACL に適用されます。

   Shield Advancedは、アプリケーションレイヤーの自動DDoS軽減機能により、 AWS WAF リソースのウェブ ACLにレートベースのルールを維持し、既知のDDoSソースからのリクエストの量を制限します。さらに、Shield Advanced は、現在のトラフィックパターンと過去のトラフィックベースラインを比較して、DDoS 攻撃を示している可能性のある逸脱を検出します。Shield Advanced が DDoS 攻撃を検出すると、 AWS WAF 対応するカスタムルールを作成、評価、展開することで対応します。カスタムルールが、ユーザーに代わって攻撃に対してカウントまたはブロックするかどうかを指定します。

   > **Note**
   >
   > アプリケーションレイヤーの自動DDoS対策は、最新バージョンの (v2) を使用して作成されたウェブ ACL でのみ機能します。 AWS WAF

   この機能を使用する際の注意点やベストプラクティスなど、Shield Advanced の自動アプリケーションレイヤー DDoS 軽減の詳細については、を参照してください。Shield Advanced アプリケーションレイヤー DDoS 自動緩和

3. [次へ] を選択します。コンソールウィザードは、ヘルスベースの検出のページに進みます。

保護のためのヘルスベースの検出を設定する

ヘルスベースの検出を使用するようにShield Advancedを設定して、攻撃の検出と軽減における応答性と精度を向上させます。イベントを正確に検出するには、適切に構成されたヘルスチェックが不可欠です。ヘルスベースの検出は、Route 53 ホストゾーンを除くすべてのリソースタイプに設定できます。

ヘルスベースの検出を使用するには、Route 53 でリソースのヘルスチェックを定義し、そのヘルスチェックを Shield Advanced 保護に関連付けます。設定するヘルスチェックがリソースの正常性を正確に反映していることが重要です。Shield Advanced で使用するヘルスチェックの設定に関する情報と例については、「ヘルスチェックを使用したHealth ベースの検出」を参照してください。

Shield Response Team (SRT) のプロアクティブなエンゲージメントサポートには、ヘルスチェックが必要です。プロアクティブなエンゲージメントの詳細については、「プロアクティブな関与の設定」を参照してください。

> **Note**
>
> ヘルスチェックを Shield Advanced 保護に関連付ける場合、正常であることが報告されている必要があります。

ヘルスベースの検出を設定するには

1. [Associated Health Check] (ヘルスチェックを関連付ける) で、保護に関連付けるヘルスチェックの ID を選択します。

   > **Note**
   >
   > 必要なヘルスチェックが表示されない場合は、Route 53 コンソールに移動して、ヘルスチェックとその ID を検証します。詳細については、「ヘルスチェックの作成と更新」を参照してください。

2. [Next] (次へ) を選択します。コンソールウィザードは、アラームと通知のページに進みます。

アラームと通知を設定する

オプションで、検出された Amazon CloudWatch アラームとレートベースのルールアクティビティに関する Amazon 簡易通知サービスの通知を設定できます。これらを使用すると、Shield が保護対象リソースでイベントを検出したとき、またはレートベースのルールで設定されたレート制限を超えたときに通知を受け取ることができます。

Shield CloudWatch アドバンスメトリックの詳細については、を参照してくださいAWS Shield Advanced 指標。Amazon SNS の詳細については、「Amazon Simple Notification Service デベロッパーガイド」を参照してください。

アラームと通知を設定するには

1. 通知の対象となる Amazon SNS トピックを選択します。すべての保護されたリソースとレートベースのルールに単一の Amazon SNSトピックを使用することも、組織に合わせてカスタマイ

ズされた異なるトピックを選択することもできます。例えば、特定のリソースセットのインシデント対応を担当するチームごとに SNS トピックを作成できます。

2. [次へ] を選択します。コンソールウィザードは、リソース保護の確認ページに進みます。

保護設定を確認して終了する

設定を確認および構成するには

1. [Review and configure DDoS mitigation and visibility] (DDoS の緩和と可視性を確認および設定) ページで、設定を確認します。変更するには、変更する部分で [Edit] (編集) を選択します。これにより、コンソールウィザードの関連するページに戻ります。変更を加えた後、[Review and configure DDoS mitigation and visibility] (DDoS の緩和と可視性を確認および設定) ページに戻るまで、後続のページで [Next] (次へ) を選択します。
2. [Finish configuration] (設定を終了) を選択します。[Protected resources] (保護されたリソース) ページには、新しく保護されたリソースが一覧表示されます。

AWS SRT サポートの設定

Shield レスポンスチーム (SRT、Shield Response Team) は DDoS イベントへの対応を専門とするセキュリティエンジニアの集団です。必要に応じて、DDoS イベント中に SRT がユーザーに代わってリソースを管理できるようにするアクセス許可を追加できます。さらに、保護対象リソースに関連する Route 53 ヘルスチェックが検出されたイベント中に異常が発生した場合に、事前に対処するように SRT を設定できます。この両方の保護機能を追加することで、DDoS イベントへの迅速な対応が可能になります。

> **Note**
>
> Shield Response Team (SRT) のサービスを使用するには、ビジネスサポートプランまたはエンタープライズサポートプランをサブスクライブする必要があります。

SRT は、 AWS WAF アプリケーションレイヤーのイベント中にリクエストデータとログを監視して、異常なトラフィックを特定できます。これらは、 AWS WAF 問題のあるトラフィックソースを軽減するためのカスタムルールを作成するのに役立ちます。SRT は、必要に応じてアーキテクチャ上の推奨事項を作成して、リソースを推奨事項とより適切に連携させることができるようにします。AWS

SRT 関数の詳細については、「Shield Response Team (SRT) のサポート」を参照してください。

SRT にアクセス許可を付与するには

1. AWS Shield コンソールの「概要」 ページの「 AWS SRT サポートの設定」で、「SRT アクセスを編集」を選択します。 AWS Shield 対応チーム (SRT) の編集アクセスページが開きます。
2. SRT アクセス設定には、次のいずれかのオプションを選択します。

   - アカウントへのアクセス権を SRT に付与しない – Shield は、アカウントとリソースにアクセスするために以前に SRT に付与したすべての許可を削除します。
   - [Create a new role for the SRT to access my account] (SRT が自分のアカウントにアクセスするための新しいロールを作成する) — Shield は、SRT を表すサービスプリンシパル `drt.shield.amazonaws.com` を信頼するロールを作成し、マネージドポリシー `AWSShieldDRTAccessPolicy` をそれにアタッチします。管理ポリシーにより、SRT AWS Shield Advanced がユーザーに代わって AWS WAF API 呼び出しを行ったり、 AWS WAF ログにアクセスしたりすることが許可されます。管理ポリシーの詳細については、「AWS 管理ポリシー: AWSShieldDRTAccessPolicy」を参照してください。
   - SRT がマイアカウントにアクセスするための既存のロールを選択 — このオプションでは、AWS Identity and Access Management (IAM) のロールの設定を次のように変更する必要があります。
     - マネージドポリシー `AWSShieldDRTAccessPolicy` をロールにアタッチします。この管理ポリシーにより、SRT はユーザーに代わって AWS WAF API AWS Shield Advanced 呼び出しを行い、ログにアクセスすることができます。 AWS WAF 管理ポリシーの詳細については、「AWS 管理ポリシー: AWSShieldDRTAccessPolicy」を参照してください。マネージドポリシーをロールにアタッチする方法については、「Attaching and Detaching IAM Policies」(IAM ポリシーのアタッチとデタッチ) を参照してください。
     - サービスプリンシパル `drt.shield.amazonaws.com` を信頼するようにロールを変更します。これは、SRT を表すサービスプリンシパルです。詳細については、「IAM JSON ポリシーエレメント: プリンシパル」を参照してください。
3. [保存] を選択して変更を保存します。

SRT に保護とデータへのアクセスを許可する方法の詳細については、「Shield Response Team (SRT) のためのアクセス権の設定」を参照してください。

SRT のプロアクティブな関与を有効にするには

1. AWS Shield コンソールの [概要] ページの [積極的なエンゲージメントと連絡先] の [連絡先] 領域で、[編集] を選択します。

   [Edit contacts] (連絡先を編集) ページで、SRT がプロアクティブな関与のために連絡する担当者の連絡先情報を入力します。

   複数の連絡先を提供する場合は、[Notes] (備考) に、各連絡先を使用する必要がある状況を記載してください。プライマリおよびセカンダリの連絡先指定を含めて、各連絡先の空き時間およびタイムゾーンを指定します。

   問い合わせメモの例:

   - これは、24 時間年中無休でスタッフが対応するホットラインです。応答するアナリストにご協力ください。この担当者は、適切な担当者を呼び出します。
   - 5 分以内にホットラインが応答しない場合は、私までお問い合わせください。
2. [Save] (保存) を選択します。

   [Overview] (概要) ページには、更新された連絡先情報が反映されます。
3. [Edit proactive engagement feature] (プロアクティブな関与機能を編集) を選択し、[Enable] (有効化) を選択してから、[Save] (保存) を選択してプロアクティブな関与を有効にします。

プロアクティブな関与の詳細については、「プロアクティブな関与の設定」を参照してください。

DDoS CloudWatch ダッシュボードを作成してアラームを設定する CloudWatch

Amazon を使用すると CloudWatch、潜在的な DDoS アクティビティを監視できます。Amazon は Shield Advanced から未加工データを収集し、それを読み取り可能でほぼリアルタイムのメトリクスに処理します。統計情報を使用して CloudWatch 、ウェブアプリケーションやサービスのパフォーマンスを把握できます。使用方法の詳細については CloudWatch、『Amazon CloudWatch ユーザーガイド』 CloudWatchの「内容」を参照してください。

- CloudWatch ダッシュボードの作成手順については、を参照してくださいAmazon によるモニタリング CloudWatch。
- ダッシュボードに追加できる Shield Advanced メトリクスの説明については、「AWS Shield Advanced 指標」を参照してください。

Shield Advancedは、 CloudWatch イベントが発生していないときよりも、DDoSイベント中の方がリソースメトリックスを頻繁に報告します。Shield Advanced は、イベント中は 1 分ごとに、およびイベント終了直後に 1 回、メトリクスをレポートします。イベントが発生していない間、Shield Advanced は 1 日に 1 回、リソースに割り当てられた時間にメトリクスを報告します。この定期レポートでは、メトリクスがアクティブな状態に保たれ、カスタムアラームで使用できるようになります。 CloudWatch

これで、Shield Advanced の開始方法のチュートリアルは完了です。選択した保護機能を最大限に活用するには、Shield Advanced の機能とオプションの検索を続けてください。まず、DDoS イベントの可視性 および DDoS イベントへの対応 でイベントを表示して対応するためのオプションをよく理解してください。

Shield Response Team (SRT) のサポート

Shield Response Team (SRT) は、Shield Advanced のお客様に追加のサポートを提供します。SRT は DDoS イベントへの対応を専門とするセキュリティエンジニアの集団です。 AWS Support プランに対するサポートの追加レイヤーとして、SRT と直接やり取りして、イベント対応ワークフローの一環として SRT の専門知識を活用できます。オプションの詳細および設定ガイダンスについては、次のトピックを参照してください。

> **Note**
>
> Shield Response Team (SRT) のサービスを使用するには、ビジネスサポートプランまたはエンタープライズサポートプランをサブスクライブする必要があります。

SRT のサポートアクティビティ

SRT との連携の主な目標は、アプリケーションの可用性とパフォーマンスを保護することです。DDoS イベントのタイプとアプリケーションのアーキテクチャに応じて、SRT は次のいずれかまたは複数のアクションを実行することがあります。

- AWS WAF ログ分析とルール — AWS WAF ウェブ ACL を使用するリソースの場合、SRT AWS WAF はログを分析して、アプリケーションのウェブリクエストに含まれる攻撃特性を特定できます。エンゲージメント中に承認を得た場合、SRT はウェブ ACL に変更を適用して、特定した攻撃をブロックできます。
- カスタムネットワーク緩和策の構築 – SRT は、インフラストラクチャレイヤーの攻撃に対して、お客様のためにカスタム緩和策を作成できます。SRT はユーザーと連携して、アプリケーション

について想定されるトラフィックを理解し、予期しないトラフィックをブロックし、パケット/秒のレート制限を最適化できます。詳細については、「Shield Response Team (SRT) によるカスタム緩和の設定」を参照してください。

- ネットワークトラフィックエンジニアリング — AWS SRTはネットワークチームと緊密に連携して、Shield Advancedのお客様を保護します。必要に応じて、 AWS インターネットトラフィックがネットワークに到達する方法を変更して、 AWS アプリケーションにより多くの緩和能力を割り当てることができます。
- アーキテクチャ上の推奨事項 — SRT は、攻撃に対する最善の緩和策には、 AWS ベストプラクティスに沿ったアーキテクチャの変更が必要であると判断する場合があります。これらの変更は、これらのプラクティスの実装を支援するのに役立ちます。詳細については、「DDoS 耐性向上のためのAWS のベストプラクティス」を参照してください。

トピック

Shield Response Team (SRT) のためのアクセス権の設定

Shield Response Team (SRT) に権限を付与して、 AWS WAF ユーザーに代わってログにアクセスし、 AWS Shield Advanced および AWS WAF API を呼び出して保護を管理することができます。アプリケーション層の DDoS イベント中、SRT AWS WAF はリクエストを監視して異常なトラフィックを特定し、 AWS WAF 問題のあるトラフィックソースを軽減するためのカスタムルールの作成を支援します。

さらに、Amazon S3 バケットに保存した他のデータ (Application Load Balancer、Amazon、またはサードパーティのソースからのパケットキャプチャやログなど) へのアクセスを SRT に許可できます。 CloudFront

> **Note**
>
> Shield Response Team (SRT) のサービスを使用するには、ビジネスサポートプランまたはエンタープライズサポートプランをサブスクライブする必要があります。

SRT の許可を管理するには

1. AWS Shield コンソールの [概要] ページの [AWS SRT サポートの設定] で、[SRT アクセスを編集] を選択します。 AWS Shield 対応チーム (SRT) の編集アクセスページが開きます。
2. SRT アクセス設定には、次のいずれかのオプションを選択します。

   - アカウントへのアクセス権を SRT に付与しない – Shield は、アカウントとリソースにアクセスするために以前に SRT に付与したすべての許可を削除します。
   - [Create a new role for the SRT to access my account] (SRT が自分のアカウントにアクセスするための新しいロールを作成する) — Shield は、SRT を表すサービスプリンシパル `drt.shield.amazonaws.com` を信頼するロールを作成し、マネージドポリシー `AWSShieldDRTAccessPolicy` をそれにアタッチします。管理ポリシーにより、SRT AWS Shield Advanced がユーザーに代わって AWS WAF API 呼び出しを行ったり、 AWS WAF ログにアクセスしたりすることが許可されます。管理ポリシーの詳細については、「AWS 管理ポリシー: AWSShieldDRTAccessPolicy」を参照してください。
   - SRT がマイアカウントにアクセスするための既存のロールを選択 — このオプションでは、AWS Identity and Access Management (IAM) のロールの設定を次のように変更する必要があります。
     - マネージドポリシー `AWSShieldDRTAccessPolicy` をロールにアタッチします。この管理ポリシーにより、SRT はユーザーに代わって AWS WAF API AWS Shield Advanced 呼び出しを行い、ログにアクセスすることができます。 AWS WAF 管理ポリシーの詳細については、「AWS 管理ポリシー: AWSShieldDRTAccessPolicy」を参照してください。マネージドポリシーをロールにアタッチする方法については、「Attaching and Detaching IAM Policies」(IAM ポリシーのアタッチとデタッチ) を参照してください。
     - サービスプリンシパル `drt.shield.amazonaws.com` を信頼するようにロールを変更します。これは、SRT を表すサービスプリンシパルです。詳細については、「IAM JSON ポリシーエレメント: プリンシパル」を参照してください。
3. (オプション): Amazon S3 バケットへの SRT アクセスを許可します。 AWS WAF ウェブ ACL ログにないデータを共有する必要がある場合は、これを設定します。たとえば、Application Load Balancer のアクセスログ、Amazon CloudFront ログ、またはサードパーティのソースからのログなどです。

> **Note**
>
> AWS WAF ウェブ ACL ログにはこれを行う必要はありません。SRT は、アカウントへのアクセス権が付与されると、それらにアクセスできるようになります。

a. 次のガイドラインに従って Amazon S3 バケットを設定します。

- バケットの場所は、前のステップの AWS Shield Response Team (SRT) アクセス権限で SRT AWS アカウント に一般アクセスを許可した場所と同じである必要があります。
- バケットは、プレーンテキストまたは SSE-S3 暗号化のいずれかです。Amazon S3 SSE-S3 暗号化の詳細については、「Amazon S3 ユーザーガイド」の「Amazon S3 が管理する暗号化キーによるサーバー側の暗号化 (SSE-S3) を使用したデータの保護」を参照してください。

  SRT は、() に格納されたキーで暗号化されたバケットに格納されたログを表示したり処理したりすることはできません。 AWS Key Management Service AWS KMS

b. Shield Advanced の [(Optional): Grant SRT access to an Amazon S3 bucket] ((オプション): Amazon S3 バケットへのアクセス権を SRT に付与) セクションで、データまたはログが保存されている各 Amazon S3 バケットについてバケットの名前を入力し、[Add Bucket] (バケットを追加) を選択します。バケットは最大 10 個まで追加できます。

これにより、SRT に各バケットに対する次の `s3:GetBucketLocation`、`s3:GetObject`、および `s3:ListBucket` 許可が付与されます。

10 個を超えるバケットにアクセスする許可を SRT に付与する場合は、追加のバケットポリシーを編集し、SRT についてここにリストされている許可を手動で付与することで、これを実行できます。

ポリシーリストの例を以下に示します。

```
{
    "Sid": "AWSDDoSResponseTeamAccessS3Bucket",
    "Effect": "Allow",
    "Principal": {
        "Service": "drt.shield.amazonaws.com"
    },
    "Action": [
        "s3:GetBucketLocation",
        "s3:GetObject",
        "s3:ListBucket"
    ],
    "Resource": [
        "arn:aws:s3:::bucket-name",
```

```
            "arn:aws:s3:::bucket-name/*"
        ]
}
```

4. [保存] を選択して変更を保存します。

IAM AWSShieldDRTAccessPolicy ロールを作成してポリシーをアタッチし、そのロールを AssociatedRole オペレーションに渡すことで、API を通じて SRT を承認することもできます。

プロアクティブな関与の設定

プロアクティブなエンゲージメントにより、攻撃の可能性があるためにアプリケーションの可用性またはパフォーマンスが影響を受ける場合は、Shield Response Team (SRT) から直接連絡します。このエンゲージメントモデルでは、SRT による最も迅速な対応が提供され、SRT がお客様と連絡を取り合う前であってもトラブルシューティングを開始できるため、推奨されています。

プロアクティブエンゲージメントは、Elastic IP AWS Global Accelerator アドレスと標準アクセラレータでのネットワークレイヤーイベントとトランスポートレイヤーイベント、Amazon ディストリビューションと Application Load Balancer でのウェブリクエストフラッドに対して利用できます。CloudFrontプロアクティブエンゲージメントは、Amazon Route 53 ヘルスチェックが関連付けられている Shield Advanced リソース保護でのみ利用できます。ヘルスチェックの管理と使用の詳細については、「ヘルスチェックを使用したHealth ベースの検出」を参照してください。

Shield Advanced によって検出されたイベント中、SRT はヘルスチェックの状態を使用して、イベントがプロアクティブなエンゲージメントの対象となるかどうかを判断します。その場合は、プロアクティブな関与の設定で提供された連絡ガイダンスに従って、SRT から連絡が送られます。

プロアクティブな関与のために最大 10 件の連絡先を設定でき、SRT がお客様に連絡する際に参考となるメモを入力できます。イベント中、お客様側のプロアクティブな関与の連絡先は、SRT と対話が可能な状態になっている必要があります。24 時間年中無休のオペレーションセンターがない場合は、ポケットベルの連絡先を提供し、連絡先メモにこの連絡先設定を記載できます。

プロアクティブなエンゲージメントのために、次を実行する必要があります。

- ビジネスサポートプランまたはエンタープライズサポートプランに加入している必要があります。
- Amazon Route 53 ヘルスチェックは、プロアクティブなエンゲージメントで保護するリソースに関連付ける必要があります。SRT は、イベントにプロアクティブなエンゲージメントが必要かどうかを判断するのをサポートするために、ヘルスチェックのステータスを使用します。したがって、ヘルスチェックが保護されたリソースの状態を正確に反映していることが重要です。詳細とガイダンスについては、「ヘルスチェックを使用したHealth ベースの検出」を参照してください。

- AWS WAF ウェブ ACL が関連付けられているリソースの場合は、の最新バージョンである AWS WAF (v2) を使用してウェブ ACL を作成する必要があります。 AWS WAF
- イベント中、プロアクティブなエンゲージメントのために、SRT による使用を目的として、少なくとも 1 名の連絡先を提供する必要があります。連絡先情報を完全かつ最新の状態に保ちます。

SRT のプロアクティブな関与を有効にするには

1. AWS Shield コンソールの [概要] ページの [積極的なエンゲージメントと連絡先] の [連絡先] 領域で、[編集] を選択します。

   [Edit contacts] (連絡先を編集) ページで、SRT がプロアクティブな関与のために連絡する担当者の連絡先情報を入力します。

   複数の連絡先を提供する場合は、[Notes] (備考) に、各連絡先を使用する必要がある状況を記載してください。プライマリおよびセカンダリの連絡先指定を含めて、各連絡先の空き時間およびタイムゾーンを指定します。

   問い合わせメモの例:

   - これは、24 時間年中無休でスタッフが対応するホットラインです。応答するアナリストにご協力ください。この担当者は、適切な担当者を呼び出します。
   - 5 分以内にホットラインが応答しない場合は、私までお問い合わせください。
2. [Save] (保存) を選択します。

   [Overview] (概要) ページには、更新された連絡先情報が反映されます。
3. [Edit proactive engagement feature] (プロアクティブな関与機能を編集) を選択し、[Enable] (有効化) を選択してから、[Save] (保存) を選択してプロアクティブな関与を有効にします。

Shield Response Team (SRT) への問い合わせ

次のいずれかの方法で Shield Response Team (SRT) に連絡できます。

サポートケース

[AWS サポートセンター] コンソールの [AWS Shield] でケースを開くことができます。

サポートケースの作成に関するガイダンスについては、「AWS Support センター」を参照してください。

状況に適した重要度を選択し、連絡先の詳細を入力します。説明では、可能な限り詳細に入力します。影響を受ける可能性があると思われる保護されたリソースと、エンドユーザーエクスペリエンスの現在の状態に関する情報を入力してください。例えば、ユーザーエクスペリエンスが低下したり、アプリケーションの一部が現在利用できない場合は、その情報を提供してください。

- DDoS 攻撃が疑われる状況 –アプリケーションの可用性またはパフォーマンスが DDoS 攻撃の可能性がある現象によって現在影響を受けている場合は、次の重要度と連絡先のオプションを選択します。
  - 重要度については、サポートプランで使用可能な最も高い重要度を選択します。
    - ビジネスサポートの場合、これは [Production system down: < 1 hour] (本番稼働用システムのダウン: 1 時間未満) です。
    - エンタープライズサポートの場合、これは [Business-critical system down: < 15 minutes] (ビジネスクリティカルなシステムのダウン: 15 分未満) です。
  - 連絡先オプションについては、[Phone] (電話) または [Chat] (チャット) のいずれかを選択し、詳細を入力してください。ライブのお問い合わせ方法を使用すると、最速で応答が得られます。

プロアクティブな関与

AWS Shield Advanced 事前対応により、イベントが検出された際に、保護対象リソースに関連付けられた Amazon Route 53 ヘルスチェックに異常が生じた場合、SRT はお客様に直接連絡します。このオプションの詳細については、「プロアクティブな関与の設定」を参照してください。

Shield Response Team (SRT) によるカスタム緩和の設定

Elastic IP (EIP) AWS Global Accelerator と標準アクセラレータについては、Shield レスポンスチーム (SRT) と協力してカスタム緩和策を設定できます。これは、緩和の導入時に適用すべき特定のロジックがわかっている場合に便利です。例えば、特定の国からのトラフィックのみを許可する、特定のレート制限を適用する、オプションの検証を設定する、フラグメントを許可しない、パケットペイロードの特定のパターンに一致するトラフィックのみを許可する、などの設定が可能です。

一般的なカスタム緩和の例には、次のようなものがあります。

- パターンが一致 - クライアントサイドアプリケーションとやり取りするサービスを運用する場合、それらのアプリケーションに固有の既知のパターンを照合するように選択できます。例えば、お客様が配布する特定のソフトウェアをエンドユーザーがインストールする必要があるゲームまたは通信サービスを運用する場合があります。アプリケーションがサービスに送信するすべてのパケットにマジックナンバーを含めることができます。フラグメント化されていない TCP または UDP パ

ケットペイロードおよびヘッダーを最大 128 バイト (個別または連続) まで照合できます。一致は、パケットペイロードの先頭からの特定のオフセット、または既知の値に続くダイナミックオフセットとして 16 進表記で表すことができます。たとえば、緩和はバイト `0x01` を探すことができ、次の 4 バイトとして `0x12345678` が予測されます。

- DNS 固有 — グローバルアクセラレータや Amazon Elastic Compute Cloud (Amazon EC2) などのサービスを使用して独自の権威ある DNS サービスを運用する場合、パケットが有効な DNS クエリであることを確認し、DNS トラフィックに固有の特定の属性を評価する疑惑スコアリングを適用するパケットを検証するカスタム緩和をリクエストできます。

SRT でのカスタム緩和策の構築に関する問い合わせは、 AWS Shieldでサポートケースを作成します。 AWS Support ケースの作成について詳しくは、「はじめに」を参照してください。 AWS Support

でのリソース保護 AWS Shield Advanced

AWS Shield Advanced リソースの保護を追加して設定できます。1 つのリソースのための保護を管理でき、保護されたリソースを論理コレクションにグループ化して、イベント管理を改善できます。AWS Configを使用してShield アドバンスドプロテクションの変更を追跡することもできます。

トピック

- AWS Shield Advanced リソースタイプ別の保護
- AWS Shield Advanced アプリケーション層 (レイヤー 7) 保護
- ヘルスチェックを使用したHealth ベースの検出
- でのリソース保護の管理 AWS Shield Advanced
- AWS Shield Advanced 保護グループ
- でのリソース保護の変更の追跡 AWS Config

AWS Shield Advanced リソースタイプ別の保護

Shield Advanced は、ネットワークレイヤー、トランスポートレイヤー (レイヤー 3 と 4)、およびアプリケーションレイヤー (レイヤー 7) AWS のリソースを保護します。一部のリソースを直接保護し、他のリソースを保護されたリソースとの関連付けを通じて保護することができます。Shield Advanced は IPv4 をサポートしていますが、IPv6 はサポートしていません。

このセクションは、各リソースタイプの Shield Advanced 保護に関する情報を提供します。

> **Note**
>
> Shield Advanced は、Shield Advanced で、または AWS Firewall Manager Shield Advanced ポリシーを通じて指定したリソースのみを保護します。リソースは自動的に保護されません。

Shield Advanced を使用すると、次のリソースタイプで高度なモニタリングと保護を行うことができます。

- Amazon CloudFront ディストリビューション Shield Advanced は、 CloudFront 継続的なデプロイのために、保護されたプライマリディストリビューションに関連するすべてのステージングディストリビューションを保護します。
- Amazon Route 53 ホストゾーン。
- AWS Global Accelerator 標準アクセラレータ。
- Amazon EC2 Elastic IP アドレス。Shield Advanced は、保護された Elastic IP アドレスに関連付けられているリソースを保護します。
- Amazon EC2 インスタンス (Amazon EC2 Elastic IP アドレスへの関連付け経由)
- 次の Elastic Load Balancing (ELB) ロードバランサー:
  - Application Load Balancer。
  - Classic Load Balancer。
  - Network Load Balancer (Amazon EC2 Elastic IP アドレスへの関連付け経由)。

Shield Advanced を使用して他のリソースタイプを保護することはできません。例えば、 AWS Global Accelerator カスタムルーティングアクセラレータや Gateway Load Balancer を保護することはできません。

AWS アカウントあたり各リソースタイプについて最大 1,000 のリソースをモニタリングおよび保護できます。たとえば、1 つのアカウントで、1,000 個の Amazon EC2 Elastic IP アドレス、1,000 CloudFront 個のディストリビューション、1,000 個のアプリケーションロードバランサーを保護できます。https://console.aws.amazon.com/servicequotas/ の Service Quotas コンソールで、Shield Advanced で保護できるリソースの数の増加をリクエストできます。

Shield Advanced を使用した Amazon EC2 インスタンスおよび Network Load Balancer の保護

Amazon EC2 インスタンスおよび Network Load Balancer を保護するには、まずこれらのリソースを Elastic IP アドレスにアタッチし、次に Shield Advanced で Elastic IP アドレスを保護します。

Elastic IP アドレスを保護すると、Shield Advanced は、それらがアタッチされているリソースを識別して保護します。Shield Advanced は、Elastic IP アドレスにアタッチされているリソースのタイプを自動的に識別し、そのリソースのために適切な検出および緩和策を適用します。これには、その Elastic IP アドレスに固有のネットワーク ACL を設定することが含まれます。 AWS リソースでの Elastic IP アドレスの使用の詳細については、Amazon Elastic Compute Cloud のドキュメントまたは Elastic Load Balancing のドキュメントのガイドを参照してください。

攻撃中、Shield Advanced はネットワーク ACL をネットワークの境界に自動的に展開します AWS 。ネットワーク ACL がネットワークの境界にある場合、Shield Advanced はより大きな DDoS イベントに対する保護を提供できます。通常、ネットワーク ACL は Amazon VPC 内の Amazon EC2 インスタンスの近くで適用されます。ネットワーク ACL は、Amazon VPC とインスタンスが処理できるだけの大きさの攻撃を緩和できます。例えば、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が最大 10 Gbps を処理できる場合、10 Gbps を超えるボリュームは遅くなり、そのインスタンスへのトラフィックをブロックする可能性があります。攻撃を受けている最中に、Shield Advanced はネットワーク ACL を AWS 境界に昇格させ、数テラバイトのトラフィックを処理できます。ネットワーク ACL は、典型的なネットワークの容量を超えてリソースを十分に保護することができます。ネットワーク ACL の詳細については、「ネットワーク ACL」を参照してください。

AWS Elastic Beanstalkなどの一部のスケーリングツールでは、Elastic IP アドレスをNetwork Load Balancer に自動的にアタッチできません。このような場合、Elastic IP アドレスを手動でアタッチする必要があります。

AWS Shield Advanced アプリケーション層 (レイヤー 7) 保護

Shield Advanced を使用してアプリケーションレイヤーのリソースを保護するには、まず AWS WAF ウェブ ACL をリソースに関連付けて、それに 1 つ以上のレートベースのルールを追加します。さらに、アプリケーションレイヤー DDoS 自動緩和を有効にすることもできます。これにより、DDoS 攻撃への対応として、Shield Advanced がユーザーのために自動的にウェブ ACL ルールを作成および管理するようになります。

Shield Advanced を使用してアプリケーションレイヤーリソースを保護する場合、Shield Advanced は、トラフィックを時間の経過に合わせて分析し、ベースラインを確立して維持します。Shield Advanced は、DDoS 攻撃を示している可能性のあるトラフィックパターンの異常を検出するために、これらのベースラインを使用します。Shield Advanced が攻撃を検出するポイントは、Shield Advanced が攻撃前にモニタリングできる状態になっていたトラフィックと、ウェブアプリケーショ

ンで使用するアーキテクチャによって異なります。Shield Advanced の動作に影響を与える可能性があるアーキテクチャのバリエーションには、使用するインスタンスのタイプ、インスタンスのサイズ、該当するインスタンスのタイプが拡張ネットワーキングをサポートするかどうかなどがあります。Shield Advanced を設定して、アプリケーションレイヤー攻撃に対して自動的に緩和策を実施することもできます。

Shield アドバンストのサブスクリプションと費用 AWS WAF

Shield アドバンスドサブスクリプションは、Shield AWS WAF アドバンスドで保護するリソースの標準機能を使用するコストをカバーします。Shield Advanced AWS WAF 保護の対象となる標準料金は、ウェブ ACL あたりのコスト、ルールあたりのコスト、およびウェブリクエストインスペクションの 100 万リクエストあたりの基本価格（最大 1,500 WCU、デフォルトのボディサイズまで）です。

Shield Advanced 自動アプリケーションレイヤー DDoS 軽減を有効にすると、150 ウェブ ACL キャパシティユニット (WCU) を使用するルールグループがウェブ ACL に追加されます。これらの WCU は、ウェブ ACL 内の WCU の使用量に対してカウントされます。詳細については、Shield Advanced アプリケーションレイヤー DDoS 自動緩和、Shield Advanced ルールグループ、およびAWS WAF ウェブ ACL キャパシティーユニット (WCUs)を参照してください。

Shield アドバンスドへのサブスクリプションは、Shield アドバンスドを使用して保護していないリソースの使用には適用されません。 AWS WAF また、 AWS WAF 保護対象リソースの標準外の追加費用もカバーされません。 AWS WAF 非標準費用の例としては、ボットコントロール、CAPTCHA ルールアクション、1,500 個以上の WCU を使用するウェブ ACL、デフォルトの本文サイズを超えるリクエスト本文の検査などがあります。全リストは料金ページに記載されています。 AWS WAF

詳細情報および料金の例については、「Shield の料金」および「AWS WAF の料金」を参照してください。

トピック

- 検出と緩和
- Shield AWS WAF アドバンストアプリケーションレイヤーのウェブ ACL とレートベースのルール
- Shield Advanced アプリケーションレイヤー DDoS 自動緩和

検出と緩和

このセクションでは、Shield Advanced によるアプリケーション層イベントの検出と軽減に影響する要因について説明します。

ヘルスチェック

アプリケーション全体の状態を正確に報告するHealth チェックは、アプリケーションで発生しているトラフィック状況に関する情報を Shield Advanced に提供します。Shield Advancedは、アプリケーションが異常を報告しているときに潜在的な攻撃を示す情報を必要とせず、アプリケーションが正常であると報告している場合は攻撃の証拠をより多く必要とします。

アプリケーションの状態を正確に報告するようにヘルスチェックを設定することが重要です。詳細とガイダンスについては、「ヘルスチェックを使用したHealth ベースの検出」を参照してください。

トラフィックベースライン

トラフィックベースラインは、アプリケーションの通常のトラフィックの特性に関するShield Advancedの情報を提供します。Shield Advancedはこれらのベースラインを使用して、アプリケーションが通常のトラフィックを受信していないことを認識します。これにより、ユーザーに通知し、設定に従って、潜在的な攻撃に対抗するための緩和オプションの考案とテストを開始できます。Shield Advanced がトラフィックベースラインを使用して潜在的なイベントを検出する方法の詳細については、アプリケーションレイヤーの脅威の検出ロジック概要セクションを参照してください。

Shield Advanced は、保護されたリソースに関連付けられているウェブ ACL によって提供される情報からベースラインを作成します。Shield Advanced がアプリケーションのベースラインを確実に決定できるようになるには、ウェブ ACL を少なくとも 24 時間、最大 30 日間リソースに関連付ける必要があります。必要な時間は、Shield アドバンスドまたはを介してウェブ ACL を関連付けた時点から始まります AWS WAF。

Shield Advanced アプリケーション層保護でウェブ ACL を使用する方法の詳細については、を参照してくださいShield AWS WAF アドバンストアプリケーションレイヤーのウェブ ACL とレートベースのルール。

レートベースのルール

レートベースのルールは攻撃の軽減に役立ちます。また、通常のトラフィックベースラインやヘルスチェックのステータスレポートに現れるほど大きな問題になる前に攻撃を軽減することで、攻撃をわかりにくくすることもできます。

Shield Advanced でアプリケーションリソースを保護する場合は、ウェブ ACL でレートベースのルールを使用することをお勧めします。緩和策は潜在的な攻撃を目立たなくすることができますが、防御の第一線としては価値があり、正当な顧客がアプリケーションを利用できるようにするのに役立

ちます。レートベースのルールで検出されたトラフィックとレート制限は、メトリクスに表示されます。 AWS WAF

独自のレートベースのルールに加えて、アプリケーションレイヤーの自動DDoS軽減を有効にすると、Shield Advanced は攻撃を軽減するために使用するルールグループをウェブ ACL に追加します。このルールグループでは、Shield Advancedには、DDoS攻撃のソースであることが知られているIPアドレスからのリクエストの量を制限するレートベースのルールが常に設定されています。Shield アドバンスドルールによって軽減されるトラフィックのメトリクスは表示できません。

レートベースのルールの詳細については、「レートベースのルールステートメント」を参照してください。Shield Advancedがアプリケーションレイヤーの自動DDoS軽減に使用するレートベースのルールについては、を参照してください。Shield Advanced ルールグループ

Shield AWS WAF アドバンスドとメトリクスの詳細については、を参照してくださいAmazon によるモニタリング CloudWatch。

Shield AWS WAF アドバンストアプリケーションレイヤーのウェブ ACL とレートベースのルール

Shield Advanced でアプリケーション層リソースを保護するには、 AWS WAF まずウェブ ACL をリソースに関連付けます。 AWS WAF は、アプリケーション層リソースに転送される HTTP および HTTPS リクエストを監視し、リクエストの特性に基づいてコンテンツへのアクセスを制御できるウェブアプリケーションファイアウォールです。リクエストの発信元、クエリ文字列とクッキーのコンテンツ、単一の IP アドレスからのリクエストのレートなどの要因に基づいて、リクエストをモニタリングおよび管理するウェブ ACL を設定できます。少なくとも、Shield Advanced 保護では、ウェブ ACL をレートベースのルールに関連付けて、各 IP アドレスのリクエストのレートを制限する必要があります。

関連付けられたウェブ ACL にレートベースのルールが定義されていない場合、Shield Advanced から少なくとも 1 つ定義するように求められます。レートベースのルールは、定義したしきい値を超えると、ソース IP からのトラフィックを自動的にブロックします。これらは、ウェブリクエストのフラッドからアプリケーションを保護するのに役立つとともに、DDoS 攻撃の可能性を示していることがあるトラフィックの急増についてアラートを出すことができます。

> **Note**
>
> レートベースのルールは、そのルールが監視しているトラフィックの急増に非常に迅速に対応します。このため、レートベースのルールでは、攻撃だけでなく、Shield Advanced検出による潜在的な攻撃の検出も防止できます。このトレードオフは、攻撃パターンを完全に可

視化するよりも防御に有利に働きます。攻撃に対する防御の最前線として、レートベースのルールを使用することをおすすめします。

ウェブ ACL を設定すると、DDoS 攻撃が発生した場合、ウェブ ACL にルールを追加して管理することで緩和策を適用します。これは、直接行うことができるほか、Shield レスポンスチーム (SRT、Shield Response Team) のサポートを受けて、またはアプリケーションレイヤー DDoS 自動緩和を通じて自動的に行うこともできます。

Important

アプリケーションレイヤーの自動DDoS対策も使用している場合は、「ウェブ ACL を管理するためのベストプラクティス」を参照してください。自動緩和の使用に関するベストプラクティス

デフォルトのレートベースのルール動作

レートベースのルールをデフォルト設定で使用すると、過去 5 AWS WAF 分の時間枠のトラフィックが定期的に評価されます。 AWS WAF リクエストレートが許容レベルまで下がるまで、ルールのしきい値を超える IP アドレスからのリクエストをブロックします。Shield Advancedを使用してレートベースのルールを設定する場合、そのレートしきい値を、5分間の任意のソースIPから予想される通常のトラフィックレートよりも高い値に設定します。

ウェブ ACL で複数のレートベースのルールを使用したい場合があります。例えば、高いしきい値を持つすべてのトラフィックについて 1 つのレートベースのルールを指定するとともに、ウェブアプリケーションの特定の部分と一致するように設定され、しきい値が低い追加のルールを 1 つ以上指定できます。例えば、ログインページに対する不正を緩和するために、しきい値を低くして URI `/login.html` に対する照合を行うことができます。

レートベースのルールを設定して、異なる評価時間枠を使用し、ヘッダー値、ラベル、クエリ引数などの多数のリクエストコンポーネントごとにリクエストを集約できます。詳細については、「レートベースのルールステートメント」を参照してください。

追加情報やガイダンスについては、セキュリティブログ記事「最も重要な 3 AWS WAF つのレートベースのルール」を参照してください。

構成オプションを次のように拡張しました。 AWS WAF

Shield Advanced コンソールでは、レートベースのルールを追加し、基本的なデフォルト設定で構成できます。レートベースのルールを管理することで、追加の設定オプションを定義できます。 AWS WAFたとえば、転送された IP アドレス、クエリ文字列、ラベルなどのキーに基づいてリクエストを集約するようにルールを設定できます。また、ルールにスコープダウンステートメントを追加して、一部のリクエストを評価およびレート制限から除外することも可能です。詳細については、「レートベースのルールステートメント」を参照してください。 AWS WAF を使用してウェブリクエストのモニタリングルールと管理ルールを管理する方法については、を参照してくださいウェブ ACL の作成。

Shield Advanced アプリケーションレイヤー DDoS 自動緩和

攻撃の一部であるウェブリクエストをカウントまたはブロックすることで、保護されたアプリケーションレイヤーリソースに対するアプリケーションレイヤー (レイヤー 7) 攻撃を自動的に緩和して対応するよう Shield Advanced を設定できます。このオプションは、Shield Advanced AWS WAF を通じてウェブ ACL と独自のレートベースのルールを使用して追加するアプリケーションレイヤー保護に追加されます。

リソースの自動緩和が有効になっている場合、Shield Advanced はリソースの関連するウェブ ACL にルールグループを管理し、リソースに代わって緩和ルールを管理します。ルールグループには、DDoS 攻撃のソースであることが判明している IP アドレスからのリクエストの量を追跡するレートベースのルールが含まれています。

さらに、Shield Advanced は、現在のトラフィックパターンと過去のトラフィックベースラインを比較して、DDoS 攻撃を示している可能性のある逸脱を検出します。Shield Advancedは、 AWS WAF ルールグループに追加のカスタムルールを作成、評価、展開することで、検出されたDDoS攻撃に対応します。

目次

- Shield Advanced ルールグループ
- アプリケーションレイヤー DDoS 自動緩和の管理
  - リソースのアプリケーションレイヤー DDoS 自動緩和設定の表示
  - アプリケーションレイヤー DDoS 自動緩和の有効化と無効化
  - アプリケーションレイヤー DDoS 自動緩和に使用されるアクションの変更
  - AWS CloudFormation アプリケーションレイヤーの自動DDoS対策との併用

自動緩和機能を使用する際の注意事項

次のリストでは、Shield Advanced アプリケーションレイヤー DDoS 自動緩和の注意事項と、対応が必要となる可能性があるステップについて説明します。

- アプリケーションレイヤーの自動DDoS対策は、最新バージョンの AWS WAF (v2) を使用して作成されたウェブ ACL でのみ機能します。
- Shield Advancedは、アプリケーションの通常の過去のトラフィックのベースラインを確立するのに時間がかかります。これを活用して攻撃トラフィックを通常のトラフィックから検出して分離し、攻撃トラフィックを軽減します。ベースラインを確立するまでの時間は、ウェブ ACL を保護対象アプリケーションリソースに関連付けた時点から 24 時間から 30 日の間です。トラフィックベースラインの詳細については、を参照してください。検出と緩和
- 自動アプリケーション層の DDoS 軽減を有効にすると、150 のウェブ ACL キャパシティユニット (WCU) を使用するルールグループがウェブ ACL に追加されます。これらの WCU は、ウェブ ACL 内の WCU の使用量に対してカウントされます。詳細については、「Shield Advanced ルールグループ」および「AWS WAF ウェブ ACL キャパシティーユニット (WCUs)」を参照してください。
- Shield AWS WAF アドバンスドルールグループはメトリクスを生成しますが、表示することはできません。これは、 AWS マネージドルールルールグループなど、ウェブ ACL で使用しているが所有していない他のルールグループと同じです。 AWS WAF メトリクスの詳細については、「」を参照してくださいAWS WAF メトリクスとディメンション。この Shield アドバンスドプロテクションオプションの詳細については、を参照してくださいShield Advanced アプリケーションレイヤー DDoS 自動緩和。
- 複数のリソースを保護するウェブ ACL の場合、自動緩和機能では、保護対象リソースのいずれにも悪影響を及ぼさないカスタム緩和策のみが展開されます。
- DDoS 攻撃の開始から Shield Advanced がカスタム自動緩和ルールを実行するまでの時間は、各イベントによって異なります。一部の DDoS 攻撃は、カスタムルールがデプロイされる前に終了することがあります。他の攻撃は、緩和策が既に実施されている場合に発生する可能性があるた

め、これらのルールによってイベントの開始時から緩和される可能性があります。さらに、ウェブ ACL と Shield Advanced ルールグループのレートベースのルールは、攻撃トラフィックが起こり得るイベントとして検出される前に軽減する可能性があります。

- Amazon CloudFront などのコンテンツ配信ネットワーク (CDN) を介してトラフィックを受信する Application Load Balancer の場合、それらのアプリケーションロードバランサーリソースに対する Shield Advanced のアプリケーション層自動軽減機能は低下します。Shield Advanced は、クライアントトラフィック属性を使用して、アプリケーションへの通常のトラフィックから攻撃トラフィックを識別および分離します。CDN は、元のクライアントトラフィック属性を保持または転送しない場合があります。を使用する場合は CloudFront、ディストリビューションの自動緩和機能を有効にすることをお勧めします。 CloudFront
- アプリケーションレイヤー DDoS 自動緩和は、保護グループとインタラクションしません。保護グループに含まれるリソースのために自動緩和を有効にできますが、Shield Advanced は、保護グループの検出結果に基づいて攻撃の緩和策を自動的に適用しません。Shield Advanced は、個々のリソースのために攻撃の自動緩和を適用します。

自動緩和の使用に関するベストプラクティス

自動緩和機能を使用する場合は、このセクションに記載されているガイダンスを遵守してください。

一般的な保護管理

自動緩和保護を計画および実装する場合は、以下のガイドラインに従ってください。

- Shield アドバンスドを使用するか、Shield アドバンスド自動緩和設定の管理に使用している場合はFirewall Manager AWS Firewall Manager を使用して、すべての自動緩和保護を管理します。これらの保護を管理するために Shield Advanced と Firewall Manager を合わせて使用しないでください。
- 同一のウェブ ACL と保護設定を使用して類似のリソースを管理し、別々のウェブ ACL を使用して類似していないリソースを管理してください。Shield Advanced は、保護されたリソースに対する DDoS 攻撃を緩和する場合、リソースに関連付けられているウェブ ACL のルールを定義し、ウェブ ACL に関連付けられているすべてのリソースのトラフィックに対してルールをテストします。Shield Advanced は、関連付けられたリソースに悪影響を及ぼさない場合にのみルールを適用します。詳細については、「Shield Advanced が自動緩和を管理する方法」を参照してください。
- すべてのインターネットトラフィックが Amazon CloudFront ディストリビューションを介してプロキシされるアプリケーションロードバランサーの場合は、ディストリビューションでのみ自動軽減を有効にします。 CloudFront CloudFront ディストリビューションには常に最大数のオリジナルトラフィック属性が存在し、Shield Advanced はそれを活用して攻撃を軽減します。

検出と軽減の最適化

以下のガイドラインに従って、自動緩和機能が保護対象リソースに提供する保護を最適化してください。アプリケーション層の検出と軽減の概要については、を参照してください。検出と緩和

- 保護対象リソースのヘルスチェックを設定し、それらを使用してShield Advanced 保護でヘルスベースの検出を有効にします。ガイダンスについては、「ヘルスチェックを使用したHealth ベースの検出」を参照してください。
- Shield Advancedが通常の過去のトラフィックのベースラインを確立するまで、Countモードで自動軽減を有効にします。Shield アドバンスドでは、ベースラインを確立するのに24時間から30日かかります。

  通常のトラフィックパターンのベースラインを確立するには、以下が必要です。

  - ウェブ ACL を保護対象リソースに関連付ける。を使用してウェブ ACL AWS WAF を直接関連付けることも、Shield アドバンストアプリケーションレイヤー保護を有効にして使用するウェブ ACL を指定するときに Shield アドバンスドに関連付けさせることもできます。
  - 保護対象アプリケーションへの通常のトラフィックフロー。アプリケーションが起動される前など、アプリケーションのトラフィックが正常でない場合や、本番環境のトラフィックが長期間不足している場合、履歴データを収集することはできません。

ウェブ ACL 管理

自動緩和策で使用するウェブ ACL を管理するには、以下のガイドラインに従ってください。

- 保護対象リソースに関連付けられているウェブ ACL を置き換える必要がある場合は、次の変更を順番に行ってください。

  1. Shield アドバンスドで、自動緩和機能を無効にします。
  2. で AWS WAF、古いウェブ ACL の関連付けを解除し、新しいウェブ ACL を関連付けます。
  3. Shield アドバンスドで、自動緩和機能を有効にします。

  Shield Advanced は、自動緩和を古いウェブ ACL から新しいウェブ ACL に自動的に移行しません。
- 名前が `ShieldMitigationRuleGroup` で始まるウェブ ACL からルールグループルールを削除しないでください。このルールグループを削除すると、ウェブ ACL に関連付けられているすべてのリソースの Shield Advanced 自動緩和によって提供される保護が無効になります。さらに、Shield Advanced が変更の通知を受け取り、設定を更新するのに時間がかかることがあります。この間、Shield Advanced コンソールページには誤った情報が表示されます。

ルールグループの詳細については、「Shield Advanced ルールグループ」を参照してください。

- 名前が ShieldMitigationRuleGroup で始まるルールグループルールの名前を変更しないでください。変更すると、ウェブ ACL を介して Shield Advanced 自動緩和機能によって提供される保護が妨げられる可能性があります。
- ルールとルールグループを作成するときには、ShieldMitigationRuleGroup で始まる名前を使用しないでください。この文字列は、Shield Advanced が自動緩和を管理するために使用します。
- ウェブ ACL ルールの管理では、優先順位の設定として 10,000,000 を割り当てないでください。Shield Advanced は、自動緩和ルールグループルールを追加するときに、この優先順位設定をそのルールグループルールに割り当てます。
- ShieldMitigationRuleGroup ルールの優先順位を維持し、ウェブ ACL 内の他のルールと関連して必要なときに実行できるようにします。Shield Advanced は、優先順位を 10,000,000 に設定したルールグループルールをウェブ ACL に追加し、他のルールよりも後に実行します。 AWS WAF コンソールウィザードを使用してウェブ ACL を管理する場合は、ウェブ ACL にルールを追加した後に、必要に応じて優先順位設定を調整してください。
- AWS CloudFormation を使用してウェブ ACL を管理する場合は、ShieldMitigationRuleGroupルールグループのルールを管理する必要はありません。「AWS CloudFormation アプリケーションレイヤーの自動DDoS対策との併用」のガイダンスに従います。

自動緩和を有効にするために必要な設定

Shield Advanced 自動緩和機能は、リソースのためのアプリケーションレイヤーの DDoS 保護の一部として有効にします。コンソールからこれを実行する方法については、「アプリケーションレイヤー DDoS 保護を設定する」を参照してください。

自動緩和機能を使用するには、次の操作を行う必要があります。

- [Associate a web ACL with the resource] (ウェブ ACL をリソースに関連付ける) – これは、Shield Advanced アプリケーションレイヤー保護のために必須です。複数のリソースに同じウェブ ACL を使用できます。同様のトラフィックを持つリソースについてのみ、これを行うことをお勧めします。ウェブ ACL を複数のリソースで使用するための要件など、ウェブ ACL の詳細については、「AWS WAF 仕組み」を参照してください。
- [Enable and configure Shield Advanced automatic application layer DDoS mitigation] (Shield Advanced アプリケーションレイヤー DDoS 自動緩和を有効にして設定する) – これを有効にする

際に、Shield Advanced が DDoS 攻撃の一部であると判断したウェブリクエストを自動的にブロックまたはカウントするかどうかを指定します。Shield Advanced は、関連付けられたウェブ ACL にルールグループを追加し、それを使用して、リソースに対する DDoS 攻撃に対する対応を動的に管理します。ルールアクションのオプションについては、「ルールアクション」を参照してください。

- (オプションですが、推奨されます) ウェブ ACL にレートベースのルールを追加する – デフォルトでは、レートベースのルールは、個々の IP アドレスによる短時間の大量リクエスト送信を防ぐことで、DDoS 攻撃に対する基本的な保護をリソースに提供します。カスタムリクエストの集約オプションや例など、レートベースのルールの詳細については、「レートベースのルールステートメント」を参照してください。

Shield Advanced が自動緩和を管理する方法

このセクションのトピックでは、Shield Advanced がアプリケーションレイヤー DDoS 自動緩和の設定変更をどのように処理するか、および自動緩和が有効になっている場合の DDoS 攻撃の処理方法について説明します。

トピック

- 自動緩和を有効にした場合の実行内容
- Shield Advanced が自動緩和機能で DDoS 攻撃に対応する方法
- Shield Advanced がルールアクション設定を管理する方法
- 攻撃が沈静化した場合に Shield Advanced が緩和を管理する方法
- 自動緩和を無効にした場合の実行内容

自動緩和を有効にした場合の実行内容

Shield Advanced は、自動緩和を有効にすると、次の処理を実行します。

- 必要に応じて、Shield アドバンスドで使用するルールグループを追加します。 AWS WAF リソースに関連付けたウェブ ACL に、自動アプリケーションレイヤーの DDoS AWS WAF 軽減専用のルールグループルールがまだない場合は、Shield アドバンスドが 1 つ追加します。

  グループルールのルール名は `ShieldMitigationRuleGroup` で始まります。ルールグループには常に `ShieldKnownOffenderIPRateBasedRule` という名前のレートベースのルールが含まれており、DDoS 攻撃のソースであることが判明している IP アドレスからのリクエストの量を制限します。Shield Advanced ルールグループと参照するウェブ ACL ルールの詳細については、「Shield Advanced ルールグループ」を参照してください。

- [Starts responding to DDoS attacks against the resource] (リソースに対する DDoS 攻撃への対応を開始) – Shield Advanced は、保護されたリソースに対する DDoS 攻撃に自動的に対応します。常に存在するレートベースのルールに加えて、Shield Advanced はルールグループを使用して DDoS AWS WAF 攻撃を軽減するためのカスタムルールを展開します。Shield Advanced は、これらのルールをアプリケーションとアプリケーションが経験する攻撃に合わせて調整し、デプロイする前にリソースの過去のトラフィックに照らし合わせてテストします。

Shield Advanced は、自動緩和に使用するウェブ ACL で単一のルールグループルールを使用します。Shield Advanced が、別の保護されたリソースのルールグループを追加している場合、ウェブ ACL には別のルールグループを追加しません。

アプリケーションレイヤー DDoS 自動緩和は、攻撃を緩和するためのルールグループの存在によって異なります。 AWS WAF ルールグループが何らかの理由でウェブ ACL から削除された場合、削除によってウェブ ACL に関連付けられているすべてのリソースの自動緩和が無効になります。

Shield Advanced が自動緩和機能で DDoS 攻撃に対応する方法

保護対象リソースで自動軽減機能を有効にすると、Shield Advanced ルールグループのレートベースのルール `ShieldKnownOffenderIPRateBasedRule` は、既知の DDoS ソースからのトラフィックの量の増加に自動的に応答します。このレート制限は迅速に適用され、攻撃に対する最前線の防御として機能します。

Shield Advanced が攻撃を検出すると、次の処理が実行されます。

1. アプリケーションへの通常のトラフィックから攻撃トラフィックを分離する攻撃シグネチャの特定を試みます。目標は、攻撃トラフィックにのみ影響し、アプリケーションへの通常のトラフィックに影響を与えない、質の高い DDoS 緩和ルールを作成することです。
2. 識別された攻撃シグネチャを、攻撃対象のリソースおよび同じウェブ ACL に関連付けられている他のリソースについての過去のトラフィックパターンに照らして評価します。Shield Advanced は、イベントに対応してルールをデプロイする前にこれを実行します。

   Shield Advanced は、評価結果に応じて、次のいずれかを実行します。

   - Shield Advanced は、攻撃シグネチャが DDoS 攻撃に関与するトラフィックのみを隔離すると判断した場合、ウェブ ACL の Shield Advanced 緩和ルールグループの AWS WAF ルールにシグネチャを実装します。Shield Advanced は、これらのルールに、リソースの自動緩和のために設定したアクション設定 (Count または Block) を提供します。
   - その他の場合、Shield Advanced は緩和策を講じません。

攻撃全体を通じて、Shield Advanced は、基本的な Shield Advanced アプリケーションレイヤー保護と同じ通知を送信し、同じイベント情報を提供します。Shield Advanced イベントコンソールで、イベントと DDoS 攻撃に関する情報、および攻撃に対する Shield Advanced の緩和策に関する情報を確認できます。詳細については、「DDoS イベントの可視性」を参照してください。

Block ルールアクションを使用するように自動緩和を設定し、Shield Advanced がデプロイした緩和ルールからの誤検出が発生した場合は、ルールアクションを Count に変更できます。これを行う方法については、「アプリケーションレイヤー DDoS 自動緩和に使用されるアクションの変更」を参照してください。

Shield Advanced がルールアクション設定を管理する方法

自動緩和策のルールアクションは Block または Count に設定できます。

保護されたリソースの自動緩和ルールアクション設定を変更すると、Shield Advanced は、リソースのすべてのルール設定を更新します。Sheild Advanced ルールグループのリソースに現在適用されているルールが更新され、新しいルールの作成時に新しいアクション設定が使用されます。

同じウェブ ACL を使用するリソースに対して異なるアクションを指定すると、Shield Advanced はルールグループのレートベースのルール `ShieldKnownOffenderIPRateBasedRule` の Block アクション設定を使用します。Shield Advanced は、特定の保護対象リソースに代わってルールグループ内の他のルールを作成および管理し、リソースに指定したアクション設定を使用します。ウェブ ACL 内の Shield Advanced ルールグループのすべてのルールは、関連するすべてのリソースのウェブトラフィックに適用されます。

アクション設定を変更すると、反映されるまでに数秒かかる場合があります。この間、ルールグループが使用されている場所によっては古い設定が表示され、他の場所では新しい設定が表示される場合があります。

自動緩和設定のルールアクション設定は、コンソールのイベントページ、およびアプリケーションレイヤー設定ページで変更できます。イベントページの詳細については、「DDoS イベントへの対応」を参照してください。設定ページについては、「アプリケーションレイヤー DDoS 保護を設定する」を参照してください。

攻撃が沈静化した場合に Shield Advanced が緩和を管理する方法

Shield Advanced は、特定の攻撃に対してデプロイされた緩和ルールが不要になったと判断すると、そのルールを Shield Advanced 緩和ルールグループから削除します。

緩和ルールの削除は、必ずしも攻撃の終了と一致しません。Shield Advanced は、保護されたリソースで検出された攻撃のパターンをモニタリングします。攻撃の最初の発生に対してデプロイしたルー

ルを所定の位置に保持することで、特定のシグネチャを使用した攻撃の再発を先回りして防御できる場合があります。必要に応じて、Shield Advanced はルールを保持する時間枠を拡大します。このようにして、Shield Advanced は、保護されたリソースに影響が及ぶ前に、特定のシグネチャで繰り返される攻撃を緩和する場合があります。

Shield Advanced が、DDoS 攻撃のソースであることが判明している IP アドレスからのリクエストの量を制限するレートベースのルール `ShieldKnownOffenderIPRateBasedRule` を削除することはありません。

自動緩和を無効にした場合の実行内容

Shield Advanced は、リソースのための自動緩和を無効にすると、次の処理を実行します。

- [Stops automatically responding to DDoS attacks] (DDoS 攻撃への自動対応を停止) – Shield Advanced は、リソースのための自動応答アクティビティを中止します。
- [Removes unneeded rules from the Shield Advanced rule group] (Shield Advanced ルールグループから不要なルールを削除) – Shield Advanced が保護されたリソースのためにマネージドルールグループ内のルールを維持している場合、それらを削除します。
- [Removes the Shield Advanced rule group, if it's no longer in use] (Shield Advanced ルールグループが使用されなくなった場合は削除) – リソースに関連付けたウェブ ACL が、自動緩和が有効になっている他のリソースに関連付けられていない場合、Shield Advanced は、そのルールグループルールをウェブ ACL から削除します。

Shield Advanced ルールグループ

Shield Advanced は、所有および管理するルールグループ内のルールを使用して、自動緩和アクティビティを管理します。Shield Advanced は、保護されたリソースに関連付けたウェブ ACL 内のルールを使用してルールグループを参照します。

ウェブ ACL におけるルールグループルール

ウェブ ACL の Shield Advanced ルールグループルールには、次のプロパティがあります。

- [Name] (名前) – `ShieldMitigationRuleGroup_`*`account-id_web-acl-id_unique-identifier`*
- [Web ACL capacity units (WCU)] (ウェブ ACL キャパシティーユニット (WCU)) – 150。これらの WCU は、ウェブ ACL 内の WCU の使用量に対してカウントされます。

Shield Advancedは、このルールを10,000,000の優先度設定でウェブ ACL に作成し、ウェブ ACL 内の他のルールやルールグループの後に実行されるようにします。 AWS WAF ウェブ ACL 内のルールを、最も低い数値優先度設定から順に実行します。ウェブ ACL の管理中に、この優先順位の設定が変更される場合があります。

自動緩和機能は、ウェブ ACL のルール グループによって使用される WCU を除き、アカウント内の追加の AWS WAF リソースを消費しません。例えば、Shield Advanced ルールグループは、アカウントのルールグループの 1 つとしてカウントされません。のアカウント制限については AWS WAF、を参照してくださいAWS WAF クォータ。

ルールグループ内のルール

参照先の Shield Advanced ルールグループ内では、Shield Advanced が DDoS 攻撃のソースであることが判明している IP アドレスからのリクエストの量を制限するレートベースのルール `ShieldKnownOffenderIPRateBasedRule` を維持します。このルールは常にルールグループに存在し、トラフィックパターンの分析に頼らずに攻撃を封じ込めるため、あらゆる攻撃に対する防御の最前線として機能します。このルールのアクションは、ルールグループの他のルールと同様に、自動緩和策で選択したアクションに設定されます。レートベースルールの詳細については、「レートベースのルールステートメント」を参照してください。

> **Note**
>
> レートベースのルールは、Shield `ShieldKnownOffenderIPRateBasedRule` アドバンスドイベント検出とは独立して動作します。自動軽減機能が有効になっている間は、このルールは DDoS 攻撃のソースであることが知られている IP アドレスをレート制限します。これらの IP アドレスについては、ルールのレート制限によって攻撃を防ぎ、Shield Advancedの検出情報に攻撃が表示されないようにすることもできます。このトレードオフは、攻撃パターンを完全に可視化するよりも防御に有利に働きます。

上記の恒久的なレートベースのルールに加えて、ルールグループには、Shield Advanced が現在 DDoS 攻撃を軽減するために使用しているすべてのルールが含まれます。Shield Advanced は、必要に応じてこれらのルールを追加、変更、削除します。詳細については、Shield Advanced が自動緩和を管理する方法 を参照してください。

メトリクス

AWS WAF ルールグループはメトリクスを生成しますが、このルールグループは Shield Advanced が所有しているため、これらのメトリクスは表示できません。詳細については、「AWS WAF メトリクスとディメンション」を参照してください。

アプリケーションレイヤー DDoS 自動緩和の管理

このセクションのガイダンスを使用して、アプリケーションレイヤー DDoS 自動緩和設定を管理します。自動緩和の仕組みについては、前のトピックを参照してください。

> **Note**
>
> で説明されているベストプラクティスに従ってください自動緩和の使用に関するベストプラクティス。

トピック

- リソースのアプリケーションレイヤー DDoS 自動緩和設定の表示
- アプリケーションレイヤー DDoS 自動緩和の有効化と無効化
- アプリケーションレイヤー DDoS 自動緩和に使用されるアクションの変更
- AWS CloudFormation アプリケーションレイヤーの自動DDoS対策との併用

リソースのアプリケーションレイヤー DDoS 自動緩和設定の表示

リソースのアプリケーションレイヤー DDoS 自動緩和設定は、[保護リソース] ページと個々の保護ページで表示できます。

リソースのアプリケーションレイヤー DDoS 自動緩和設定を表示するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ にある AWS WAF & Shield コンソールを開きます。
2. AWS Shield ナビゲーションペインで [保護されたリソース] を選択します。保護対象リソースのリストの [アプリケーションレイヤー DDoS 自動緩和] 列は、自動緩和が有効になっているかどうか、また有効になっている場合は緩和で Shield Advanced が使用するアクションを示します。

   任意のアプリケーションレイヤーリソースを選択することによっても、リソースの保護ページにリストされているのと同じ情報を表示することもできます。

アプリケーションレイヤー DDoS 自動緩和の有効化と無効化

次の手順では、保護されたリソースのための自動対応を有効または無効にする方法を示しています。

単一のリソースのアプリケーションレイヤー DDoS 自動緩和を有効または無効にするには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ にある AWS WAF & Shield コンソールを開きます。
2. AWS Shield ナビゲーションペインで [保護されたリソース] を選択します。
3. [Protections] (保護) タブで、自動緩和を有効にするアプリケーションレイヤーリソースを選択します。リソースの保護ページが開きます。
4. リソースの保護ページで、[Edit] (編集) を選択します。
5. [Configure layer 7 DDoS mitigation for global resources - optional] (グローバルリソース用のレイヤー 7 DDoS 緩和を設定 - オプション) ページの [Automatic application layer DDoS mitigation] (アプリケーションレイヤー DDoS 自動緩和) で、自動緩和に使用するオプションを選択します。コンソールのオプションを以下に示します。

   - [Keep current settings] (現在の設定を保持) — 保護されたリソースの自動緩和設定は変更されません。
   - [Enable] (有効化) — 保護されたリソースの自動緩和を有効にします。このオプションを選択する場合、ウェブ ACL ルールで使用するルールアクションも選択します。ルールアクションの設定については、「ルールアクション」を参照してください。

     保護対象リソースに通常のアプリケーショントラフィックの履歴がまだない場合は、Shield Advanced がベースラインを確立できるまで、Countモードで自動軽減を有効にします。Shield Advanced は、ウェブ ACL を保護対象リソースに関連付けるとベースラインとなる情報の収集を開始します。通常のトラフィックの適切なベースラインを確立するには 24 時間から 30 日かかる場合があります。
   - [Disable] (無効化) — 保護されたリソースの自動緩和を無効にします。
6. 残りのページを最後まで順を追って確認し、設定を保存します。

[Protections] (保護) ページで、リソースの自動軽減設定が更新されます。

アプリケーションレイヤー DDoS 自動緩和に使用されるアクションの変更

コンソール内の複数の場所で、Shield Advanced がアプリケーションレイヤーの自動対応に使用するアクションを変更できます。

- [Automatic mitigation configuration] (自動緩和設定) - リソースのための自動緩和を設定するときに、アクションを変更します。手順については、前のセクションの「アプリケーションレイヤー DDoS 自動緩和の有効化と無効化」を参照してください。
- [Event details page] (イベントの詳細ページ) – コンソールでイベント情報を表示しているときに、イベントの詳細ページでアクションを変更します。詳細については、「AWS Shield Advanced イベントの詳細」を参照してください。

ウェブ ACL を共有する保護対象リソースが 2 つあり、一方のアクションを Count に、他方のアクションを Block に設定した場合、Shield Advanced はルールグループのレートベースのルール `ShieldKnownOffenderIPRateBasedRule` のアクションを Block に設定します。

AWS CloudFormation アプリケーションレイヤーの自動DDoS対策との併用

を使用して保護とウェブ AWS CloudFormation ACL を管理する方法を理解してください。 AWS WAF

アプリケーションレイヤー DDoS 自動緩和の有効化または無効化

リソースを使用して、アプリケーションレイヤーの自動DDoS対策を有効または無効にできます AWS CloudFormation。`AWS::Shield::Protection`コンソールやその他のインターフェイスからでも、同じようにこの機能を有効または無効にできます。 AWS CloudFormation リソースについては、AWS::Shield::Protection AWS CloudFormationユーザーガイドのを参照してください。

自動緩和で使用されるウェブ ACL の管理

Shield Advanced は、 AWS WAF 保護対象リソースのウェブ ACL 内のルールグループルールを使用して、保護対象リソースの自動緩和を管理します。 AWS WAF コンソールと API を使用すると、ウェブ ACL `ShieldMitigationRuleGroup` ルールに名前がで始まるルールが表示されます。このルールはアプリケーションレイヤー DDoS 自動緩和専用で、Shield Advanced と AWS WAFがユーザーに代わって管理します。詳細については、Shield Advanced ルールグループおよびShield Advanced が自動緩和を管理する方法を参照してください。

ウェブ ACL AWS CloudFormation の管理に使用する場合は、ウェブ ACL テンプレートに Shield アドバンスドルールグループルールを追加しないでください。自動緩和保護で使用されているウェブ ACL を更新すると、ウェブ ACL AWS WAF 内のルールグループルールが自動的に管理されます。

管理する他のウェブ ACL と比べると、次のような違いがあります。 AWS CloudFormation

- AWS CloudFormation Shield Advanced ルールグループルールを使用したウェブ ACL の実際の設定と、ルールなしのウェブ ACL テンプレートの間のスタックドリフトステータスのドリフトは報

告されません。Shield Advanced ルールは、ドリフト詳細でリソースの実際のリストには表示されません。

Shield アドバンスドルールグループルールは AWS WAF、 AWS WAF コンソールや AWS WAF API などから取得したウェブ ACL リストに表示されます。

- スタック内のウェブ ACL テンプレートを変更した場合、Shield アドバンスドは、 AWS WAF 更新されたウェブ ACL の Shield アドバンスド自動緩和ルールを自動的に維持します。Shield Advanced が提供する自動緩和保護は、ウェブ ACL を更新しても中断されることはありません。

AWS CloudFormation ウェブ ACL テンプレートでは Shield アドバンスドルールを管理しないでください。ウェブ ACL テンプレートで Shield Advanced ルールを表示しないでください。「自動緩和の使用に関するベストプラクティス」のウェブ ACL 管理のベストプラクティスに従ってください。

ヘルスチェックを使用したHealth ベースの検出

ヘルスベースの検出を使用するように Shield Advanced を設定することで、攻撃の検出と緩和策の応答性と精度を改善できます。このオプションは、Route 53 ホストゾーンを除くすべてのリソースタイプで使用できます。

ヘルスベースの検出を設定するには、Route 53 でリソースのヘルスチェックを定義し、正常であると報告されていることを検証してから、Shield Advanced 保護と関連付けます。Route 53 ヘルスチェックの詳細については、「Amazon Route 53 デベロッパーガイド」の「Amazon Route 53 がリソースのヘルスをチェックする方法」および「ヘルスチェックの作成、更新、削除」を参照してください。

> **Note**
>
> Shield Response Team (SRT) のプロアクティブなエンゲージメントサポートには、ヘルスチェックが必要です。プロアクティブなエンゲージメントの詳細については、「プロアクティブな関与の設定」を参照してください。

ヘルスチェックでは、定義した要件に基づいて、リソースのヘルスを測定します。ヘルスチェックステータスは、Shield Advancedの検出メカニズムに重要な情報を提供し、特定のアプリケーションの現在の状態に対する感度を高めます。

Route 53 ホストゾーンを除くリソースタイプのために、ヘルスベースの検出を有効にできます。

- ネットワークおよびトランスポートレイヤー (レイヤー 3 /レイヤー 4) のリソース – ヘルスベースの検出により、ネットワークレイヤーおよびトランスポートレイヤーのイベント検出の精度と、Network Load Balancer、Elastic IP アドレス、および Global Accelerator 標準アクセラレーターのための緩和が改善されます。Shield Advanced を使用してこれらのリソースタイプを保護する場合、Shield Advanced は、トラフィックがアプリケーションの容量内である場合でも、小規模な攻撃の緩和と、攻撃のより迅速な緩和を行うことができます。

  ヘルスベースの検出を追加する場合、関連付けられたヘルスチェックが異常な期間中に、Shield Advanced を使用してより迅速に、低いしきい値で緩和策を行えます。
- アプリケーション層 (レイヤー 7) リソース — Health ベースの検出により、 CloudFrontディストリビューションとアプリケーションロードバランサーの Web リクエストフラッド検出の精度が向上します。Shield Advanced を使用してこれらのリソースタイプを保護する場合、リクエストの特性に基づいて、トラフィックパターンの大幅な変化を伴うトラフィック量の統計的に有意な偏差がある場合に、ウェブリクエストのフラッド検出アラートが送信されます。

  ヘルスベースの検出では、関連付けられた Route 53 ヘルスチェックが異常な期間中に、Shield Advanced は、アラートするためにより小さな偏差を必要とし、より迅速にイベントを報告します。逆に、関連付けられた Route 53 ヘルスチェックが正常である場合、Shield Advanced では、アラートするためにより大きな偏差が必要です。

目次

Shield Advanced でヘルスチェックを使用するためのベストプラクティス

Shield Advanced でヘルスチェックを作成して使用する場合は、このセクションのベストプラクティスに従います。

- モニタリングするインフラストラクチャのコンポーネントを特定して、ヘルスチェックを計画します。ヘルスチェックでは、次のリソースタイプを検討してください。
  - 重要なリソース。
  - Shield Advanced の検出と緩和で高感度が必要なリソース。
  - Shield Advanced からプロアクティブに通知を受けたいリソース。プロアクティブなエンゲージメントは、ヘルスチェックのステータスによって通知されます。

  モニタリングが必要なリソースの例としては、Amazon CloudFront ディストリビューション、インターネット向けロードバランサー、Amazon EC2 インスタンスなどがあります。
- 可能な限り少ない通知で、アプリケーションオリジンのヘルスを正確に反映するヘルスチェックを定義します。
  - アプリケーションが利用できない場合や許容可能なパラメータ内で動作しない場合のみ異常になるようにヘルスチェックを記述します。お客様は、アプリケーションの特定の要件に基づいてヘルスチェックを定義し、維持する責任があります。
  - アプリケーションのヘルスを引き続き正確に報告しながら、できる限り少ないヘルスチェックを使用します。例えば、すべてが同じ問題を報告するアプリケーションの複数の領域からの複数のアラームは、情報価値をもたらすことなく、レスポンスアクティビティにオーバーヘッドを追加する可能性があります。
  - 計算式ヘルスチェックを使用して、Amazon CloudWatch メトリックスを組み合わせてアプリケーションの状態を監視します。例えば、アプリケーションサーバーのレイテンシーと 5xx エラー率に基づいて、複合ヘルスを計算できます。これは、オリジンサーバーがリクエストを満たしていないことを示唆します。
  - CloudWatch 必要に応じて独自のアプリケーションヘルスインジケーターを作成してカスタムメトリックスに公開し、計算式ヘルスチェックで使用します。
- ヘルスチェックを実装および管理して、検出を改善し、不要なメンテナンス作業を減らします。
  - ヘルスチェックを Shield Advanced 保護に関連付ける前に、ヘルスチェックが正常な状態であることを確認してください。異常を報告しているヘルスチェックを関連付けると、保護されたリソースに関する Shield Advanced 検出メカニズムが歪む可能性があります。
  - ヘルスチェックは Shield Advanced で使用できるようにしておきます。Shield Advanced 保護に使用している Route 53 のヘルスチェックを削除しないでください。

- ステージング環境とテスト環境は、ヘルスチェックのテストにのみ使用します。本番稼働レベルのパフォーマンスと可用性を必要とする環境のヘルスチェックの関連付けのみを維持します。ステージングおよびテスト環境のために、Shield Advanced でヘルスチェックの関連付けを維持しないでください。

ヘルスチェックで一般的に使用されるメトリクス

このセクションでは、分散型サービス拒否 (DDoS) イベント中のアプリケーションの状態を測定するためにヘルスチェックで一般的に使用される Amazon CloudWatch メトリクスを一覧表示します。各リソースタイプの CloudWatch メトリクスの詳細については、表の後に続くリストを参照してください。

トピック

アプリケーションのヘルスをモニタリングするために使用されるメトリクス

リソース	メトリクス	説明
Route 53	`HealthCheckStatus`	ヘルスチェックエンドポイントのステータス。
CloudFront	`5xxErrorRate`	HTTP ステータスコードが 5xx であるすべてのリクエストの割合。これは、アプリケーションに影響を与えている攻撃を示しています。
Application Load Balancer	`HTTPCode_ELB_5XX_Count`	ロードバランサーによって生成された HTTP 5xx クライアントエラーコードの数。
Application Load Balancer	`RejectedConnectionCount`	ロードバランサーが接続の最大数に達したため、拒否された接続の数。

リソース	メトリクス	説明
Application Load Balancer	`TargetConnectionErrorCount`	ロードバランサーとターゲット間で正常に確立されなかった接続数。
Application Load Balancer	`TargetResponseTime`	リクエストがロードバランサーを離れてから、ターゲットからの応答を受信するまでの経過時間 (秒)。
Application Load Balancer	`UnHealthyHostCount`	異常とみなされるターゲットの数。
Amazon EC2	`CPUUtilization`	割り当てられた EC2 コンピューティングユニットのうち、現在使用されているものの割合。

各リソースタイプの Amazon CloudWatch メトリクス

保護されたリソースのために使用できるメトリクスの詳細については、リソースガイドの次のセクションを参照してください。

- Amazon Route 53 – Amazon Route 53 デベロッパーガイドの「Amazon Route 53 ヘルスチェックと Amazon によるリソースのモニタリング CloudWatch」。
- Amazon CloudFront – Amazon デベロッパーガイドの「Amazon CloudFront によるモニタリング CloudWatch」。 CloudFront
- Application Load Balancer – CloudWatch Application Load Balancer ユーザーガイドの「Application Load Balancer のメトリクス」。
- Network Load Balancer – CloudWatch Network Load Balancer ユーザーガイドの「Network Load Balancer のメトリクス」。
- AWS Global Accelerator – デベロッパーガイドの「 CloudWatch での Amazon AWS Global Accelerator の使用」。 AWS Global Accelerator
- Amazon Elastic Compute Cloud – 2/latest// にインスタンスで使用できる CloudWatch メトリクスを一覧表示します。 https://docs.aws.amazon.com/AWSECUserGuide

- Amazon EC2 Auto Scaling – Amazon EC2 Auto Scaling ユーザーガイドの Auto Scaling グループとインスタンスの CloudWatch メトリクスのモニタリング。 Auto Scaling

ヘルスチェックの関連付けの管理

アプリケーションが許容可能なパラメータ内で実行されている場合にのみヘルスチェックが正常であることを報告し、そうでないときにのみ異常であることを報告する場合に、Shield Advanced におけるヘルスチェックの使用から最も大きな恩恵を受けることができます。このセクションのガイダンスを使用して、Shield Advanced でヘルスチェックの関連付けを管理します。

> **Note**
>
> Shield Advanced は、ヘルスチェックを自動的に管理しません。

Shield Advanced でヘルスチェックを使用するには、次が必要です。

- ヘルスチェックは、Shield Advanced 保護に関連付けるときに、正常であると報告するものである必要があります。
- ヘルスチェックは、保護されたリソースのヘルスに関連している必要があります。お客様は、アプリケーションの特定の要件に基づいて、アプリケーションのヘルスを正確に報告するヘルスチェックを定義し、維持する責任があります。
- ヘルスチェックは、Shield Advanced 保護で使用できるようにしておく必要があります。Shield Advanced 保護に使用している Route 53 のヘルスチェックを削除しないでください。

トピック

- ヘルスチェックのリソースへの関連付け
- リソースからのヘルスチェックの関連付けの解除
- ヘルスチェックの関連付けのステータス

ヘルスチェックのリソースへの関連付け

次の手順は、Amazon Route 53 ヘルスチェックを保護されたリソースに関連付ける方法を示しています。

> **Note**
>
> ヘルスチェックを Shield Advanced 保護に関連付ける前に、ヘルスチェックが正常な状態であることを確認してください。詳細については、「Amazon Route 53 デベロッパーガイド」の「ヘルスチェックのステータスモニタリングと通知の受信」を参照してください。

ヘルスチェックを関連付けるには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ にある AWS WAF & Shield コンソールを開きます。
2. AWS Shield ナビゲーションペインで [保護されたリソース] を選択します。
3. [Protections] (保護) タブで、ヘルスチェックに関連付けるリソースを選択します。
4. [Configure protections] (保護を設定) を選択します。
5. [Configure health check based DDoS detection - optional] (ヘルスチェックベースの DDoS 検出を設定 - オプション) ページが表示されるまで [Next] (次へ) を選択します。
6. [Associated Health Check] (ヘルスチェックを関連付ける) で、保護に関連付けるヘルスチェックの ID を選択します。

   > **Note**
   >
   > 必要なヘルスチェックが表示されない場合は、Route 53 コンソールに移動して、ヘルスチェックとその ID を検証します。詳細については、「ヘルスチェックの作成と更新」を参照してください。

7. 設定を完了するまで残りのページを順を追って確認します。[Protections] (保護) ページに、リソースについて、更新されたヘルスチェックの関連付けが一覧表示されます。
8. [Protections] (保護) ページで、新しく関連付けられたヘルスチェックが正常であるとレポートされていることを確認します。

   ヘルスチェックが異常を報告している間は、Shield Advanced でヘルスチェックの使用を正常に開始することはできません。そうすることで、Shield Advanced は非常に低いしきい値で誤検出を検出し、Shield Response Team (SRT) がリソースにプロアクティブなエンゲージメントを提供する能力にも悪影響を及ぼす可能性があります。

   新しく関連付けられたヘルスチェックで異常が報告されている場合は、次の手順を実行します。

a. Shield Advanced で、ヘルスチェックと保護機能の関連付けを解除します。

b. Amazon Route 53 のヘルスチェックの仕様を再確認し、アプリケーション全体のパフォーマンスと可用性を検証します。

c. アプリケーションが良好なヘルスのためのパラメータ内で実行され、ヘルスチェックが正常であると報告している場合は、Shield Advanced でのヘルスチェックの関連付けをもう一度お試しください。

ヘルスチェックの関連付け手順は、新しいヘルスチェックの関連付けを確立し、Shield Advanced で正常であると報告されると完了します。

リソースからのヘルスチェックの関連付けの解除

次の手順は、保護されたリソースから Amazon Route 53 ヘルスチェックの関連付けを解除する方法を示しています。

ヘルスチェックの関連付けを解除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ にある AWS WAF & Shield コンソールを開きます。
2. AWS Shield ナビゲーションペインで [保護されたリソース] を選択します。
3. [Protections] (保護) タブで、ヘルスチェックから関連付けを解除するリソースを選択します。
4. [Configure protections] (保護を設定) を選択します。
5. [Configure health check based DDoS detection - optional] (ヘルスチェックベースの DDoS 検出を設定 - オプション) ページが表示されるまで [Next] (次へ) を選択します。
6. [Associated Health Check] (関連付けられたヘルスチェック) で、- としてリストされている空のオプションを選択します。
7. 設定を完了するまで残りのページを順を追って確認します。

[Protections] (保護) ページでは、リソースのヘルスチェックフィールドが - に設定されます。これは、ヘルスチェックの関連付けがないことを示しています。

ヘルスチェックの関連付けのステータス

保護に関連付けられているヘルスチェックのステータスは、 AWS WAF & Shield コンソールの [Protected resources] (保護されたリソース) ページと各リソースの詳細ページで確認できます。

- [Healthy] (正常) - ヘルスチェックが使用可能で、正常であると報告されています。
- [Unhealthy] (異常) - ヘルスチェックが使用可能で、異常があると報告されています。
- [Unavailable] (使用不可) - Shield Advanced によるヘルスチェックは使用できません。

[Unavailable] (使用不可) のヘルスチェックを解決するには

新しいヘルスチェックを作成して使用します。Shield Advanced でステータスが使用不可になった後、ヘルスチェックを再度関連付けないでください。

これらのステップの実行に関する詳細なガイダンスについては、前のトピックを参照してください。

1. Shield Advanced で、リソースからヘルスチェックの関連付けを解除します。
2. Route 53 で、リソースの新しいヘルスチェックを作成し、その ID を記録します。詳細については、「Amazon Route 53 デベロッパーガイド」の「ヘルスチェックの作成と更新」を参照してください。
3. Shield Advanced で、新しいヘルスチェックをリソースに関連付けます。

ヘルスチェックの例

このセクションは、計算されたヘルスチェックで使用できるヘルスチェックの例を示します。計算されたヘルスチェックでは、多数の個別のヘルスチェックを使用して、組み合わせたステータスを決定します。個々のヘルスチェックのステータスは、エンドポイントのヘルスまたは Amazon CloudWatch メトリックスの状態に基づいています。ヘルスチェックを計算されたヘルスチェックに組み込んでから、個々のヘルスチェックの組み合わせられたヘルスステータスに基づいてヘルスをレポートするように、計算されたヘルスチェックを設定します。アプリケーションのパフォーマンスと可用性の要件に応じて、計算されたヘルスチェックの感度をチューニングします。

計算されたヘルスチェックの詳細については、「Amazon Route 53 デベロッパーガイド」の「他のヘルスチェック (算出したヘルスチェック) のモニタリング」を参照してください。詳細については、「Route 53 Improvements – Calculated Health Checks and Latency Checks」(Route 53 の改善 – 計算されたヘルスチェックとレイテンシーチェック) のブログ記事を参照してください。

トピック

- Amazon CloudFront ディストリビューション
- ロードバランサー
- Amazon EC2 Elastic IP アドレス (EIP)

Amazon CloudFront ディストリビューション

以下の例では、 CloudFront ディストリビューションの計算式ヘルスチェックにまとめられるヘルスチェックについて説明しています。

- 動的コンテンツを提供するディストリビューション上のパスへのドメイン名を指定して、エンドポイントをモニタリングします。正常な応答には、HTTP レスポンスコード 2xx と 3xx が含まれます。
- CloudWatch CloudFront オリジンの状態を測定するアラームの状態を監視します。たとえば、Application Load Balancer CloudWatch メトリックのアラームを管理しTargetResponseTime、アラームのステータスを反映するヘルスチェックを作成できます。ヘルスチェックは、応答時間 (リクエストがロードバランサーから発信されてから、ロードバランサーがターゲットから応答を受け取るまでの時間) がアラームで設定されたしきい値を超えると、異常になることがあります。
- レスポンスの HTTP ステータスコードが 5xx CloudWatch であるリクエストの割合を測定するアラームの状態を監視します。 CloudFront ディストリビューションの 5xx CloudWatch エラー率がアラームで定義されたしきい値より高い場合、このヘルスチェックのステータスは異常に切り替わります。

ロードバランサー

次の例では、Application Load Balancer、Network Load Balancer、または Global Accelerator 標準アクセラレーターの計算されたヘルスチェックで使用できるヘルスチェックについて説明します。

- CloudWatch クライアントがロードバランサーに確立した新しい接続の数を測定するアラームの状態を監視します。新しい接続の平均数に関するアラームのしきい値は、毎日の平均よりある程度高い値に設定できます。各リソースタイプのメトリクスは次のとおりです。
  - Application Load Balancer: NewConnectionCount
  - Network Load Balancer: ActiveFlowCount
  - Global Accelerator: NewFlowCount
- Application Load Balancer とNetwork Load Balancer では、 CloudWatch 正常と見なされるロードバランサーの数を測定するアラームの状態を監視します。アラームのしきい値は、アベイラビリティーゾーン、またはロードバランサーが必要とする最小数の正常なホストで設定できます。ロードバランサーのリソースで使用可能なメトリクスは次のとおりです。
  - Application Load Balancer: HealthyHostCount
  - Network Load Balancer: HealthyHostCount

- Application Load Balancer では、ロードバランサーのターゲットによって生成された HTTP 5xx CloudWatch 応答コードの数を測定するアラームの状態を監視します。Application Load Balancer の場合、メトリクス `HTTPCode_Target_5XX_Count` を使用して、ロードバランサーのすべての 5xx エラーの合計に基づいてアラームしきい値を設定できます。

Amazon EC2 Elastic IP アドレス (EIP)

次のヘルスチェックの例を、Amazon EC2 Elastic IP アドレスの計算されたヘルスチェックに組み合わせることができます。

- Elastic IP アドレスへの IP アドレスを指定して、エンドポイントをモニタリングします。IP アドレスの背後にあるリソースと TCP 接続を確立できる限り、ヘルスチェックは正常なままです。
- 割り当てられた Amazon EC2 コンピュートユニットのうち、 CloudWatch インスタンスで現在使用中の割合を測定するアラームの状態を監視します。Amazon EC2 メトリクス `CPUUtilization` を使用して、高いと考えるアプリケーションの CPU 使用率 (90% など) に基づいてアラームしきい値を設定できます。

でのリソース保護の管理 AWS Shield Advanced

このセクションのガイダンスを使用して、リソースの Shield Advanced 保護を管理します。

> **Note**
>
> Shield アドバンスドは、Shield アドバンスドまたは Shield アドバンスドポリシーで指定したリソースのみを保護します。 AWS Firewall Manager リソースは自動的に保護されません。

AWS Firewall Manager Shield Advancedポリシーを使用している場合は、ポリシーの対象となるリソースの保護を管理する必要はありません。Firewall Manager は、ポリシーの設定に従って、ポリシーの範囲内にあるアカウントおよびリソースの保護を自動的に管理します。詳細については、「AWS Shield Advanced ポリシー」を参照してください。

トピック

- AWS Shield AdvancedAWS リソースへの保護の追加
- AWS Shield Advanced 保護の設定
- AWS Shield AdvancedAWS リソースからの保護の解除

AWS Shield AdvancedAWS リソースへの保護の追加

Shield Advanced 保護を 1 つ以上のリソースに追加するには、このセクションのガイダンスに従います。

AWS リソースに保護を追加するには:

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ にある AWS WAF & Shield コンソールを開きます。
2. ナビゲーションペインの [保護されたリソース] AWS Shield を選択します。
3. [Add resources to protect] (保護するリソースを追加) を選択します。
4. [Choose resources to protect with Shield Advanced] (Shield Advanced で保護するリソースの選択) ページの [Specify the Region and resource types] (リージョンとリソースタイプの指定) で、保護するリソースのリージョンとリソースタイプの仕様を指定します。[All Regions] (すべてのリージョン) を選択すると複数のリージョンのリソースを保護でき、[Global] (グローバル) を選択すると選択範囲をグローバルリソースに絞り込むことができます。保護しないリソースタイプは、すべて選択解除できます。リソースタイプの保護については、「AWS Shield Advanced リソースタイプ別の保護」を参照してください。
5. [Load resources] (リソースをロード) を選択します。Shield Advanced は、[Select Resources] (リソースの選択) セクションに条件に一致する AWS リソースを入力します。
6. [Select Resources] (リソースの選択) セクションでは、リソースリストで検索する文字列を入力して、リソースのリストをフィルタリングできます。

   保護するリソースを選択します。
7. 作成しようとしている Shield Advanced 保護にタグを追加する場合は、[Tags] (タグ) セクションでそれらを指定します。 AWS リソースのタグ付けの詳細については、「タグエディタの使用」を参照してください。
8. [Protect with Shield Advanced] (Shield Advanced で保護) を選択します。これにより、Shield Advanced 保護がリソースに追加されます。

AWS Shield Advanced 保護の設定

AWS Shield Advanced 保護の設定はいつでも変更できます。これを行うには、選択した保護のオプションを確認し、変更する必要がある設定を変更します。

保護されたリソースを管理するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ にある AWS WAF & Shield コンソールを開きます。
2. AWS Shield ナビゲーションペインで [保護されたリソース] を選択します。
3. [Protections] (保護) タブで、保護するリソースを選択します。
4. 必要な [Configure protections] (保護を設定) とリソース指定オプションを選択します。
5. 各リソース保護オプションを順を追って確認し、必要に応じて変更を行います。

アプリケーションレイヤー DDoS 保護を設定する

Amazon CloudFront リソースとApplication Load Balancer リソースへの攻撃から保護するために、 AWS WAF ウェブ ACL を追加し、レートベースのルールを追加できます。詳細については、「Shield AWS WAF アドバンストアプリケーションレイヤーのウェブ ACL とレートベースのルール」を参照してください。

Shield Advanced アプリケーションレイヤー DDoS 自動緩和機能を有効にすることもできます。AWS WAF 仕組みについては、「」を参照してください。AWS WAF自動緩和機能の詳細については、「Shield Advanced アプリケーションレイヤー DDoS 自動緩和」を参照してください。

> **Important**
>
> Shield AWS Firewall Manager アドバンスドポリシーを使用して Shield アドバンスド保護を管理している場合、ここでアプリケーション層の保護を管理することはできません。他のすべてのリソースについては、ウェブ ACL にルールが含まれていない場合でも、少なくとも各リソースにウェブ ACL をアタッチすることをお勧めします。

> **Note**
>
> リソースのためにアプリケーションレイヤー DDoS 自動緩和を有効にすると、必要に応じて、オペレーションは、サービスにリンクされたロールをアカウントに自動的に追加し、ウェブ ACL 保護の管理に必要な許可を Shield Advanced に付与します。詳細については、「Shield Advanced 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。

アプリケーションレイヤー DDoS 保護を設定するには

1. [Configure layer 7 DDoS protections] (レイヤー 7 DDoS 保護を設定) ページで、リソースがまだウェブ ACL に関連付けられていない場合は、既存のウェブ ACL を選択するか、独自のウェブ ACL を作成できます。

   ウェブ ACL を作成するには、次のステップに従います。

   a. [Create web ACL] (ウェブ ACL の作成) を選択します。

   b. 名前を入力します。ウェブ ACL の作成後は、名前を変更することはできません。

   c. [Create] (作成) を選択します。

   > **Note**
   >
   > リソースが既にウェブ ACL に関連付けられている場合、別のウェブ ACL に変更することはできません。ACL を変更する場合は、最初にリソースから関連するウェブ ACL を削除します。詳細については、「ウェブ ACL とリソースの関連付けまたは関連付け解除 AWS」を参照してください。

2. ウェブ ACL にレートベースのルールが定義されていない場合は、[Add rate limit rule] (レート制限ルールを追加) を選択し、次のステップを実行してルールを追加できます。

   a. 名前を入力します。

   b. レート制限を入力します。これは、レートベースのルールアクションが IP アドレスに適用される前の任意の 5 分間に許可される、単一の IP アドレスからのリクエストの最大数です。IP アドレスからのリクエストが制限を下回ると、アクションは中止されます。

   c. リクエストの数が制限を超えている間に IP アドレスからのリクエストをカウントまたはブロックするルールアクションを設定します。ルールアクションの適用と削除は、IP アドレスのリクエストレートが変更されてから有効になるまでに 1 ~ 2 分かかる場合があります。

   d. [Add Rule] (ルールの追加) を選択します。

3. [Automatic application layer DDoS mitigation] (アプリケーションレイヤー DDoS 自動緩和) の場合、次のように、Shield Advanced がユーザーのために DDoS 攻撃を自動的に緩和するかどうかを選択します。

   - 自動緩和機能を有効にするには、[有効化] を選択し、Shield Advanced AWS WAF にカスタムルールで使用させたいルールアクションを選択します。選択内容は Count と Block で

す。 AWS WAF これらのルールアクションの詳細については、を参照してくださいルールアクション。Shield Advanced がこのアクション設定を管理する方法については、「Shield Advanced がルールアクション設定を管理する方法」を参照してください。

- 自動緩和を無効にするには、[Disable] (無効化) を選択します。
- 管理しているリソースの自動緩和設定を変更しない場合は、デフォルトの選択である [Keep current settings] (現在の設定を保持) のままにします。

Shield Advanced アプリケーションレイヤー DDoS 自動緩和の詳細については、「Shield Advanced アプリケーションレイヤー DDoS 自動緩和」を参照してください。

4. [Next] (次へ) を選択します。

アラームと通知を作成する

以下の手順は、 CloudWatch 保護対象リソースのアラームを管理する方法を示しています。

> **Note**
>
> CloudWatch 追加費用が発生する。 CloudWatch 価格については、Amazon CloudWatch の料金表をご覧ください。

アラームと通知を作成するには

1. 保護に関する [Create alarms and notifications - optional] (アラームと通知を作成 - オプション) ページで、受け取るアラームと通知の SNS トピックを設定します。通知が不要なリソースでは、[No topic] (トピックなし) を選択します。Amazon SNS トピックを追加するか、新しいトピックを作成できます。
2. Amazon SNS トピックを作成するには、次のステップに従います。
   a. ドロップダウンリストで、[Create an SNS topic] (SNS トピックを作成) を選択します。
   b. トピック名を入力します。
   c. オプションで、Amazon SNS メッセージの送信先メールアドレスを入力し、[Add email] (E メールの追加) を選択します。複数入力できます。
   d. [Create] (作成) を選択します。
3. [Next] (次へ) を選択します。

AWS Shield AdvancedAWS リソースからの保護の解除

AWS Shield Advanced AWS どのリソースからも保護をいつでも解除できます。

> **Important**
>
> AWS リソースを削除しても、そのリソースはから削除されません AWS Shield Advanced。また、この手順で説明しているように AWS Shield Advanced、リソースの保護も解除する必要があります。

AWS Shield AdvancedAWS リソースから保護を解除します。

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ にある AWS WAF & Shield コンソールを開きます。
2. AWS Shield ナビゲーションペインで [保護されたリソース] を選択します。
3. [Protections] (保護) タブで、保護を削除するリソースを選択します。
4. [Delete protections] (保護を削除) を選択します。
   - 保護用に Amazon CloudWatch アラームが設定されている場合は、保護とともにアラームを削除するオプションが表示されます。この時点でアラームを削除しない場合は、CloudWatch コンソールを使用して後で削除できます。

> **Note**
>
> Amazon Route 53 ヘルスチェックが設定されている保護は、後から再度追加した場合でも、保護にヘルスチェックが含まれます。

前述の手順では、 AWS Shield Advanced AWS 特定のリソースからの保護が解除されます。 AWS Shield Advanced サブスクリプションはキャンセルされません。このサービスに対しては引き続き料金が発生します。 AWS Shield Advanced サブスクリプションについては、AWS Support センターにお問い合わせください。

Shield CloudWatch アドバンスドプロテクションからのアラームの削除

Shield CloudWatch アドバンスドプロテクションからアラームを削除するには、次のいずれかを実行します。

- 「AWS Shield AdvancedAWS リソースからの保護の解除」の説明に従って保護を削除します。[Also delete related DDoSDetection alarm] (関連する DDoSDetection アラームも削除する) の横にあるチェックボックスをオンにしてください。
- CloudWatch コンソールを使用してアラームを削除します。削除するアラームの名前は DDoS DetectedAlarmForProtection で始まります。

AWS Shield Advanced 保護グループ

保護グループを使用して、保護されたリソースの論理コレクションを作成し、その保護をグループとして管理します。リソース保護の管理の詳細については、「AWS Shield Advanced 保護の設定」を参照してください。

> **Note**
>
> アプリケーションレイヤー DDoS 自動緩和は、保護グループとインタラクションしません。保護グループに含まれるリソースのために自動緩和を有効にできますが、Shield Advanced は、保護グループの検出結果に基づいて攻撃の緩和策を自動的に適用しません。Shield Advanced は、個々のリソースのために攻撃の自動緩和を適用します。

AWS Shield Advanced 保護グループを利用すると、複数の保護対象リソースを 1 つの単位として扱うことで、検出と緩和の範囲をセルフサービスでカスタマイズできます。リソースのグループ化は、多くのメリットをもたらす場合があります。

- 検出の精度を向上させます。
- 実行不可能なイベントの通知を減らします。
- イベント中に影響を受ける可能性のある保護されたリソースを含めるように、緩和アクションの対象範囲を拡大します。
- 複数の類似ターゲットに対する攻撃の緩和にかかる時間を短縮します。
- 新しく作成された保護対象リソースの自動保護を容易にします。

保護グループは、リソースがゼロ負荷に近い状態と完全に負荷がかかっている状態を交互に繰り返すブルー/グリーンスワップなどの状況で誤検出を減らすのに役立ちます。もう 1 つの例は、グループのメンバー間で共有されるロードレベルを維持しながら、リソースを頻繁に作成および削除する場合です。このような状況では、個々のリソースをモニタリングすると誤検出が発生する可能性がありますが、リソースグループのヘルスのモニタリングでは発生しません。

保護グループを設定して、すべての保護されたリソース、特定のリソースタイプのすべてのリソース、または個別に指定したリソースを含めることができます。保護グループの基準を満たす新しく保護されたリソースは、自動的に保護グループに含まれます。保護されたリソースは、複数の保護グループに所属できます。

AWS Shield Advanced 保護グループの管理

このセクションのガイダンスを使用して、保護グループの設定を管理します。

Shield Advanced 保護グループの作成

保護グループを作成するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ にある AWS WAF & Shield コンソールを開きます。
2. AWS Shield ナビゲーションペインで [保護されたリソース] を選択します。
3. [Protection groups] (保護グループ) タブを選択してから、[Create protection group] (保護グループを作成) を選択します。
4. [Create protection group] (保護グループを作成) ページで、グループの名前を入力します。この名前を使用して、保護されたリソースのリスト内のグループを識別します。保護グループの作成後にその名前を変更することはできません。
5. [Protection grouping criteria] (保護のグループ化の基準) で、Shield Advanced がグループに含める保護されたリソースを識別するために使用する基準を選択します。選択した基準に基づいて追加の選択を行います。
6. [Aggregation] (集約) で、イベントの検出、緩和、およびレポートのために、Shield Advanced がグループのリソースデータをどのように組み合わせるかを選択します。

   - [Sum] (合計) – グループ全体のトラフィックの合計を使用します。これは、ほとんどの場合に適切な選択です。例には、手動または自動でスケールする Amazon EC2 インスタンスの Elastic IP アドレスが含まれます。
   - [Mean] (平均) – グループ全体のトラフィックの平均を使用します。これは、トラフィックを均一に共有するリソースに適した選択です。例には、アクセラレーターとロードバランサーが含まれます。
   - [Max] (最大) – 各リソースからの最大のトラフィックを使用します。これは、トラフィックを共有しないリソースや、不均一な方法でトラフィックを共有するリソースに有益です。例としては、Amazon CloudFront CloudFront ディストリビューションやディストリビューションのオリジンリソースなどがあります。

7. [Save] (保存) を選択して保護グループを保存し、[Protected resources] (保護されたリソース) ページに戻ります。

[Shield Events] (Shield イベント) ページでは、保護グループのイベントを表示し、ドリルダウンして、グループ内の保護されたリソースに関する追加情報を確認できます。

Shield Advanced 保護グループの更新

保護グループを更新するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ にある AWS WAF & Shield コンソールを開きます。
2. AWS Shield ナビゲーションペインで [保護されたリソース] を選択します。
3. [Protection groups] (保護グループ) タブで、変更する保護グループの横にあるチェックボックスを選択します。
4. 保護グループのページで、[Edit] (編集) を選択します。保護グループの設定を変更します。
5. [Save] (保存) を選択して変更を保存します。

Shield Advanced 保護グループの削除

保護グループを削除するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ にある AWS WAF & Shield コンソールを開きます。
2. AWS Shield ナビゲーションペインで [保護されたリソース] を選択します。
3. [Protection groups] (保護グループ) タブで、削除する保護グループの横にあるチェックボックスを選択します。
4. 保護グループのページで、[Delete] (削除) を選択し、アクションを確認します。

でのリソース保護の変更の追跡 AWS Config

AWS Shield Advanced リソース保護の変更は、を使用して記録できます AWS Config。この情報を使用して、監査およびトラブルシューティングのために設定変更履歴を維持することができます。

保護の変更を記録するには、 AWS Config 追跡したいリソースごとに有効にします。詳細については、「AWS Config デベロッパーガイド」の「AWS Configの使用開始」を参照してください。

AWS Config AWS リージョン 追跡対象のリソースを含む各リソースに対して有効にする必要があります。 AWS Config 手動で有効にすることも、 AWS CloudFormation AWS CloudFormation StackSets AWS CloudFormation ユーザーガイドのサンプルテンプレートにある「有効にする AWS Config」テンプレートを使用することもできます。

有効にすると AWS Config、AWS Config 料金ページに記載されているとおりに料金が請求されます。

> **Note**
>
> AWS Config 必要なリージョンとリソースをすでに有効にしている場合は、何もする必要はありません。 AWS Config リソースの保護変更に関するログが自動的に入力され始めます。

有効化したら AWS Config、 AWS Config コンソールの米国東部 (バージニア北部) AWS Shield Advanced リージョンを使用してグローバルリソースの設定変更履歴を表示します。

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)、米国西部 (北カリフォルニア)、ヨーロッパ (アイルランド)、ヨーロッパ (フランクフルト)、アジアパシフィック (東京)、およびアジアパシフィック (シドニー) AWS Shield Advanced AWS Config リージョンのリージョンリソースの変更履歴をコンソールで確認できます。

DDoS イベントの可視性

AWS Shield 次のカテゴリのイベントとイベントアクティビティが表示されます。

- グローバル – すべてのお客様は、直近 2 週間のグローバル脅威アクティビティの集約ビューにアクセスできます。この情報は、 AWS Shield コンソールの「はじめに」ページと「グローバル脅威ダッシュボード」ページで確認できます。詳細については、「AWS Shield グローバルアクティビティとアカウントアクティビティ」を参照してください。
- アカウント - すべてのお客様は、前年度のアカウントのイベントの概要にアクセスできます。この情報は、 AWS Shield コンソールの「はじめに」ページで確認できます。詳細については、「AWS Shield グローバルアクティビティとアカウントアクティビティ」を参照してください。

Shield Advanced をサブスクライブしてリソースに保護を追加すると、保護されたリソースに対するイベントや DDoS 攻撃に関する追加情報にアクセスできます。

- 保護対象リソースのイベント — Shield Advanced は、 AWS Shield コンソールのイベントページを通じて各イベントの詳細情報を提供します。詳細については、「AWS Shield Advanced イベント」を参照してください。
- 保護対象リソースのイベントメトリクス — Shield Advanced は、保護対象のすべてのリソースについて、検出、緩和、およびトップコントリビューターの Amazon CloudWatch メトリクスを公開しています。これらのメトリックスを使用して、 CloudWatch ダッシュボードとアラームを設定できます。詳細については、「AWS Shield Advanced 指標」を参照してください。
- 保護対象リソースのクロスアカウントイベントの可視化 — Shield Advanced保護の管理に使用している場合は、Firewall Manager AWS Firewall Manager をと組み合わせて使用することで、複数のアカウントの保護を可視化できます。 AWS Security Hub詳細については、「アカウント全体にわたるイベントの可視性」を参照してください。

アプリケーション層を保護するために自動アプリケーション層の DDoS 軽減を有効にすると、

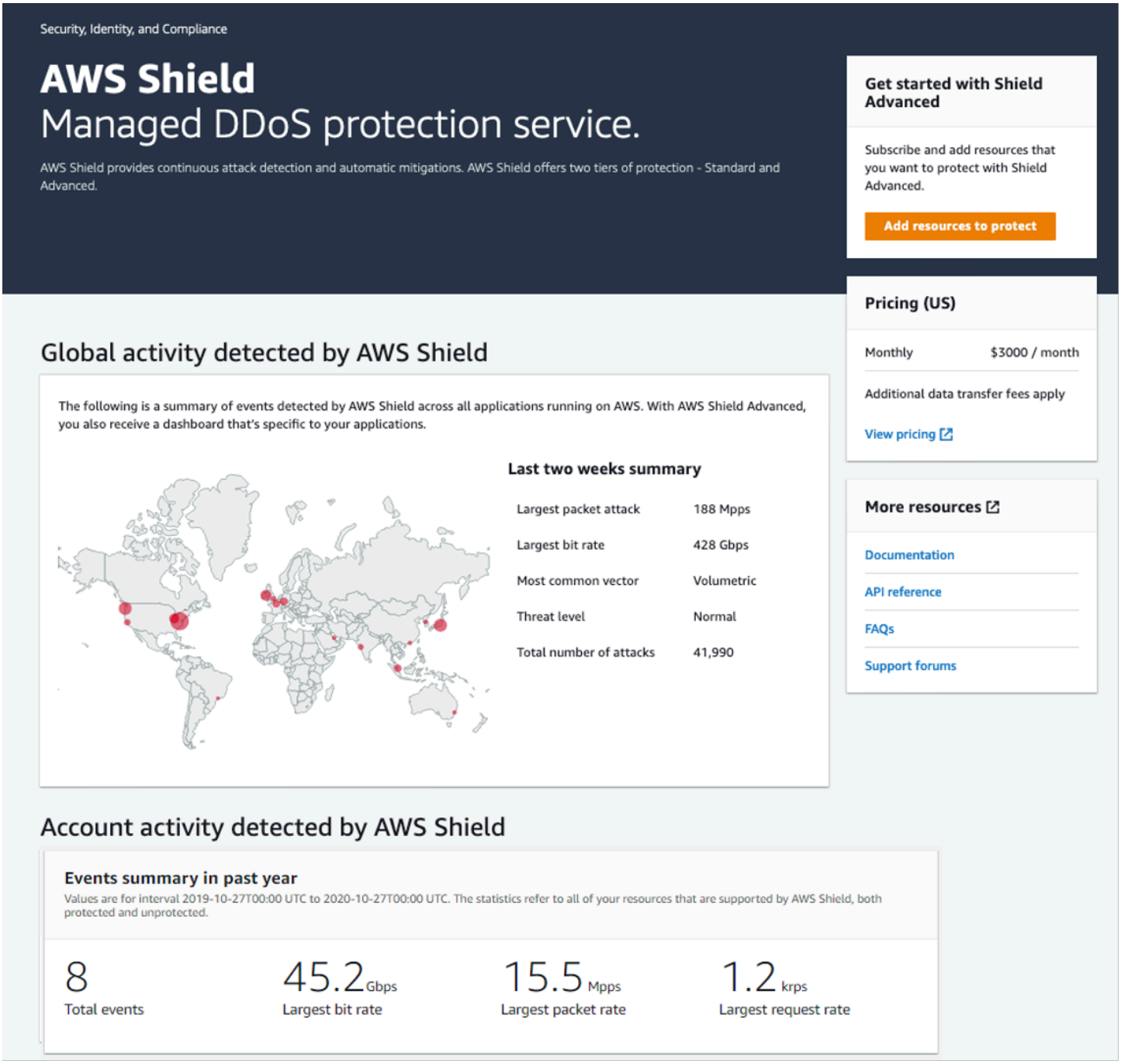
トピック

- AWS Shield グローバルアクティビティとアカウントアクティビティ
- AWS Shield Advanced イベント
- アカウント全体にわたるイベントの可視性

AWS Shield グローバルアクティビティとアカウントアクティビティ

AWS Shield コンソールの「はじめに」ページと「グローバル脅威ダッシュボード」ページで、グローバルな脅威アクティビティの集約ビューとアカウントごとのイベント概要にアクセスできます。

次のスクリーンショットは、[Getting Started] (開始方法) ページの例を示しています。



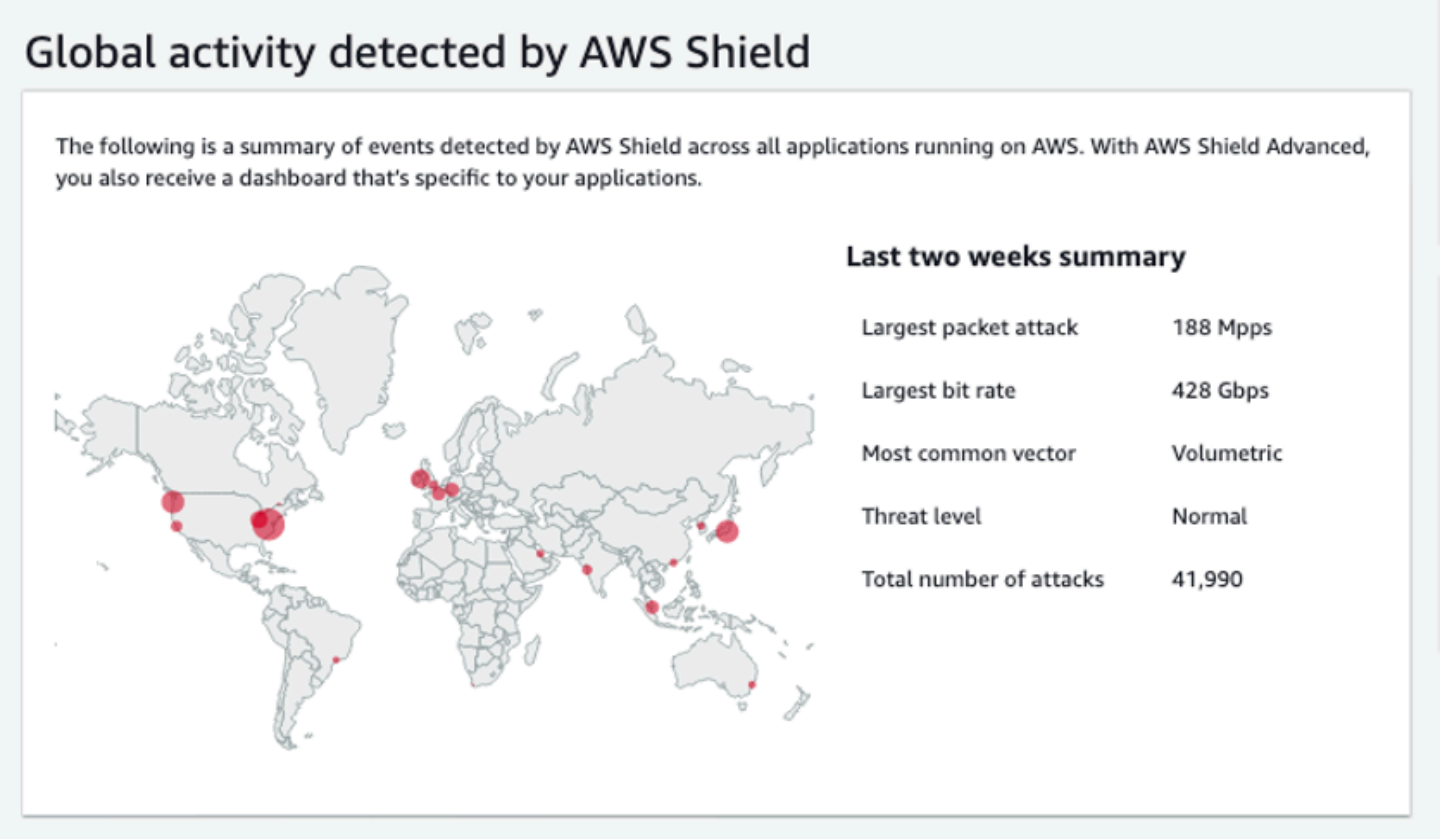
コンソールにアクセスするには AWS Shield

- AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ にある AWS WAF & Shield コンソールを開きます。

グローバルアクティビティとアカウントイベントの概要情報にアクセスするために、Shield Advanced のサブスクリプションは必要ありません。

グローバルアクティビティ

この情報は、 AWS Shield コンソールのグローバル脅威ダッシュボードと「はじめに」ページで確認できます。次のスクリーンショットは、グローバルアクティビティペインの例を示しています。



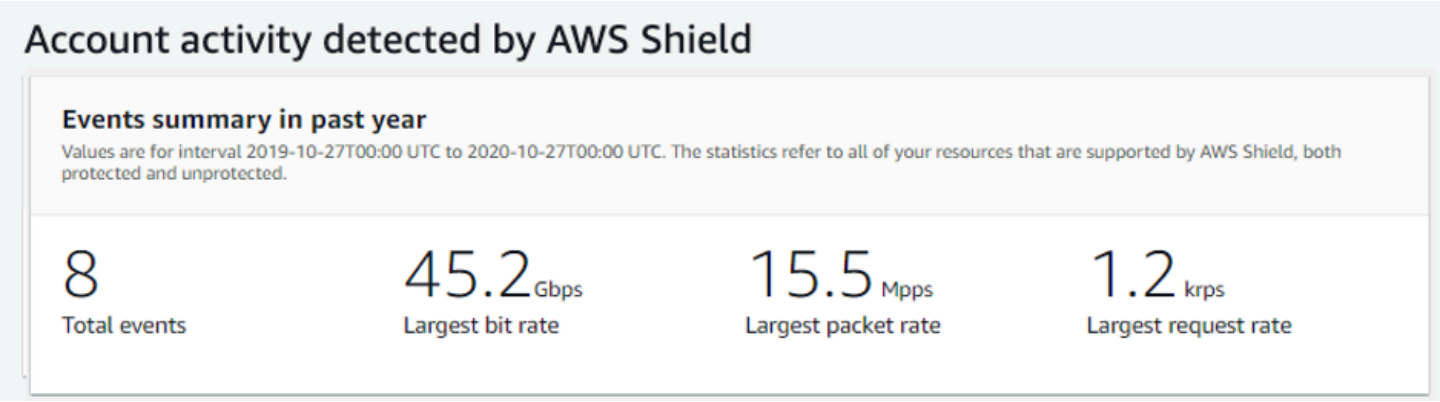
グローバルアクティビティは、すべての顧客で見られた DDoS イベントを示しています。 AWS 1 時間に 1 回、過去 2 AWS 週間の情報を更新します。コンソールペインでは、 AWS 結果を地域ごとに分割してワールドヒートマップに表示できます。マップの横には、最大パケット攻撃、最大ビットレート、最も一般的なベクトル、Shield の総数、脅威レベルなどの概要情報が表示されます。脅威レベルは、 AWS が通常観察するものと比較した現在のグローバルアクティビティの評価です。デフォルトの脅威レベル値は [Normal] (通常) です。 AWS は、昇格された DDoS アクティビティの値を自動的に [High] (高) に更新します。

[Global threat dashboard] (グローバル脅威ダッシュボード) は、時系列メトリクスも提供し、期間の変更を可能にします。重大な DDoS 攻撃の履歴を表示するには、ダッシュボードをカスタマイズして、最終日から過去 2 週間までを表示できます。時系列メトリックでは、 AWS Shield AWS 選択した期間中に実行中のアプリケーションによって検出されたすべてのイベントの最大ビットレート、パケットレート、またはリクエストレートが表示されます。

アカウントアクティビティ

AWS Shield この情報はコンソールの「はじめに」ページにあります。

次のスクリーンショットは、アカウントアクティビティペインの例を示しています。



アカウントアクティビティは、Shield Advanced による保護の対象となるリソースに関して Shield が検出した DDoS イベントを明らかにします。毎日、Shield は前日の 00:00 UTC で終わる 1 年の概要メトリクスを作成し、イベントの合計、最大ビットレート、最大パケットレート、および最大リクエストレートを表示します。

- Shield がアプリケーションを宛先とするトラフィックで疑わしい属性を観察するたびに、合計イベントメトリクスで反映されます。疑わしい属性には、通常のボリュームよりも大きいトラフィック、アプリケーションの過去のプロファイルに一致しないトラフィック、有効なアプリケーショントラフィック用に Shield で定義されているヒューリスティックに一致しないトラフィックが含まる場合があります。
- すべてのリソースで、最大ビットレートと最大のパケットレートの統計情報を使用できます。
- 最大リクエストレート統計は、ウェブ ACL が関連付けられている Amazon CloudFront ディストリビューションとアプリケーションロードバランサーでのみ利用できます。 AWS WAF

> **Note**
>
> API オペレーションを通じてアカウントレベルのイベント概要にアクセスすることもできます。 AWS Shield DescribeAttackStatistics

AWS Shield Advanced イベント

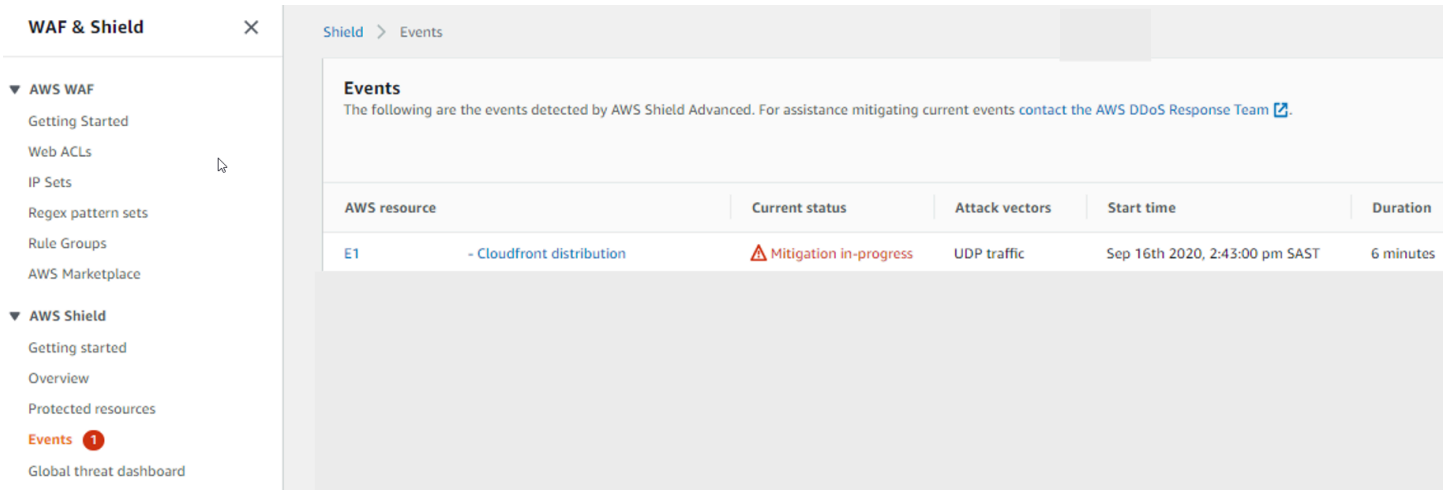
Shield Advanced をサブスクライブしてリソースを保護すると、リソースの追加の可視性機能にアクセスできます。これには、Shield Advanced によって検出されたイベントのほぼリアルタイムの通知や、検出されたイベントおよび緩和に関する追加情報が含まれます。

> **Note**
>
> Shield アドバンスコンソールのイベント情報は、Shield アドバンスドメトリクスに基づいています。Shield アドバンスドメトリクスの詳細については、以下を参照してください。AWS Shield Advanced 指標

AWS Shield 保護対象リソースへのトラフィックを複数の次元で評価します。異常が検出されると、Shield Advanced は影響を受けるリソースごとに個別のイベントを作成します。

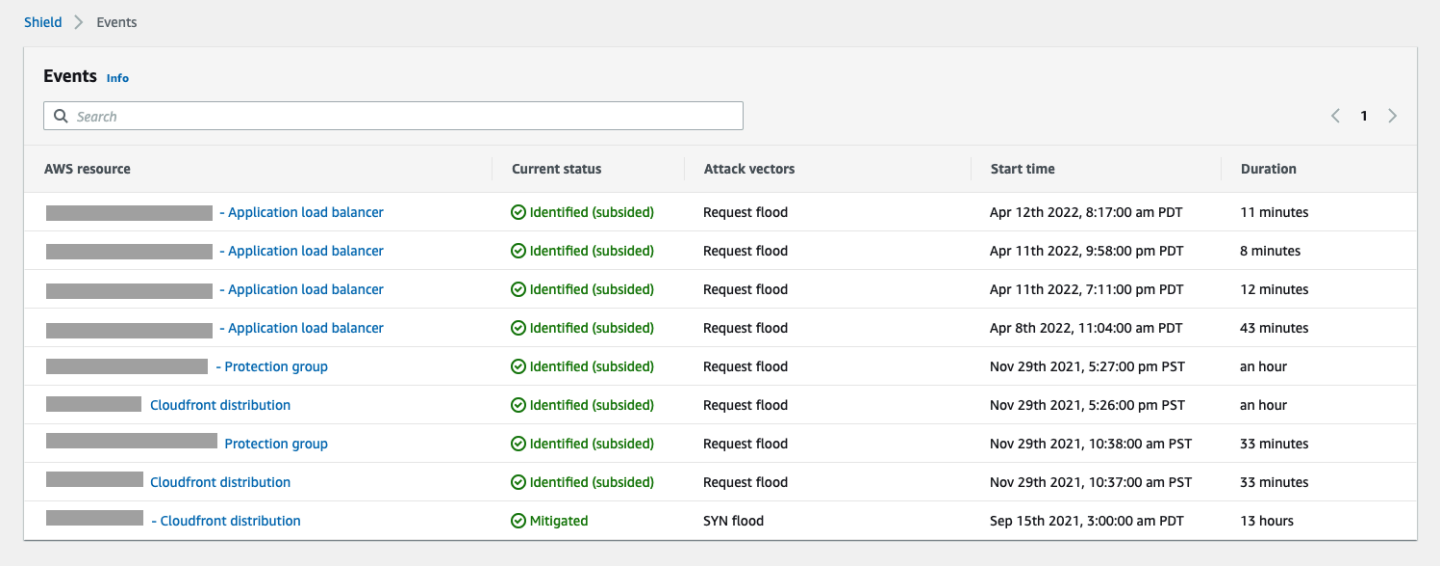
Shield コンソールの [Events] (イベント) ページからイベントの概要と詳細にアクセスできます。上位レベルの [Events] (イベント) ページには、現在および過去のイベントの概要が表示されます。

次のスクリーンショットは、単一の進行中のイベントを含む [Events] (イベント) ページの例を示しています。このアクティブなイベントには、左側のナビゲーションペインにもフラグが立てられます。



また、Shield Advanced は、トラフィックの種類と設定された保護に応じて、攻撃に対する緩和策を自動的に実施する場合があります。これらの緩和策により、過剰なトラフィックを受信したり、既知の DDoS 攻撃シグネチャに一致するトラフィックを受信したりしないようにリソースを保護できます。

次のスクリーンショットは、すべてのイベントが Shield Advanced によって緩和されたか、自然に沈静化した [Events] (イベント) のリストの例を示しています。



イベントが発生する前にリソースを保護する

DDoS 攻撃の対象となる前に、通常の予想トラフィックを受信している間に Shield Advanced でリソースを保護することで、イベント検出の精度を向上させます。

保護されたリソースのイベントを正確に報告するには、Shield Advanced はまずそのリソースの想定されるトラフィックパターンのベースラインを確立する必要があります。

- Shield Advanced は、少なくとも 15 分間保護されたリソースのインフラストラクチャレイヤーイベントをレポートします。
- Shield Advanced は、少なくとも 24 時間保護されたリソースのウェブアプリケーションレイヤーイベントをレポートします。アプリケーションレイヤーのイベントの検出精度は、Shield Advanced が 30 日間にわたって想定されるトラフィックをモニタリングした後に最適となります。

コンソールのイベント情報にアクセスするには AWS Shield

1. AWS Management Console にサインインし、https://console.aws.amazon.com/wafv2/ にある AWS WAF & Shield コンソールを開きます。
2. AWS Shield ナビゲーションペインで [イベント] を選択します。コンソールに [Events] (イベント) ページが表示されます。

3. [Events] (イベント) ページから、リスト内の任意のイベントを選択して、イベントの追加の概要情報と詳細を表示できます。

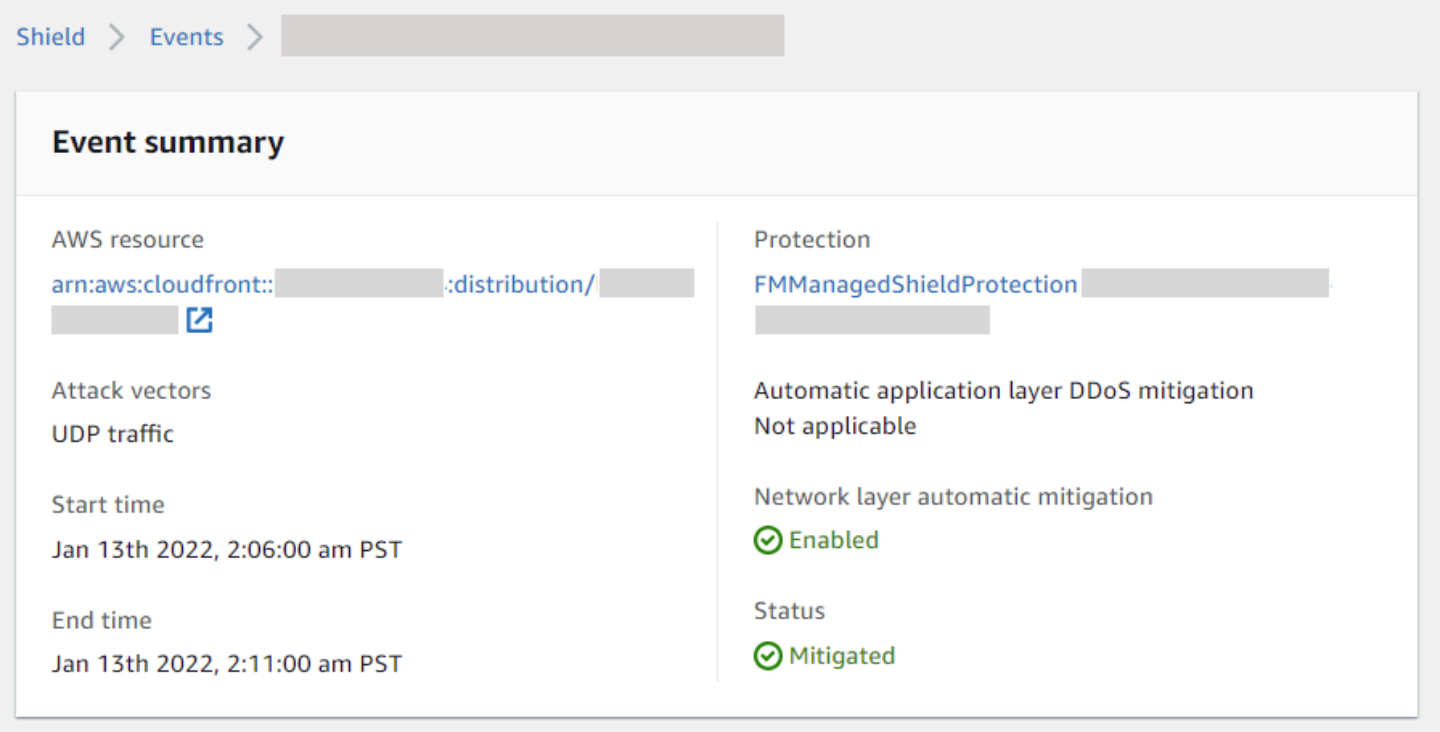
トピック

- AWS Shield Advanced イベントサマリー
- AWS Shield Advanced イベントの詳細

AWS Shield Advanced イベントサマリー

イベントの概要と詳細情報は、イベントのコンソールページに表示できます。イベントのページを開くには、 AWS イベントページリストからリソース名を選択します。

次のスクリーンショットは、ネットワークレイヤーイベントのイベント概要の例を示しています。



イベントページの概要に関する情報には次が含まれます。

- [Current status] (現在のステータス) - イベントの状態と Shield Advanced がイベントに対して実行したアクションを示す値。ステータス値は、インフラストラクチャレイヤー (レイヤー 3 または 4) およびアプリケーションレイヤー (レイヤー 7) のイベントに適用されます。
  - [Identified (ongoing)] (識別済み (進行中)) および [Identified (subsided)] (識別済み (沈静化済み)) – これらは、Shield Advanced がイベントを検出したが、これまでのところアクションを実行して

いないことを示します。[Identified (subsided)] (識別済み (沈静化済み)) は、Shield が検出した疑わしいトラフィックが介入なしに停止したことを示します。

  - [Mitigation in progress] (緩和中) および [Mitigated] (緩和済み) – これらは、Shield Advanced がイベントを検出し、それに対してアクションが実行されたことを示します。軽減は、対象リソースが Amazon CloudFront ディストリビューションまたは Amazon Route 53 ホストゾーンであり、これらには独自の自動インライン軽減機能がある場合にも使用されます。

- [Attack vectors] (攻撃ベクトル) - TCP SYN フラッドや Shield Advanced 検出ヒューリスティックなどの DDoS 攻撃ベクトル (リクエストフラッドなど)。これらは DDoS 攻撃を示している場合があります。
- [Start time] (開始時刻) – 最初の異常なトラフィックデータポイントが検出された日時。
- [Duration or end time] (期間または終了時刻) – イベントの開始時刻から Shield Advanced が最後に観察した異常なデータポイントまでの経過時間を示します。イベントが進行中であっても、これらの値は引き続き増加します。
- [Protection] (保護) – リソースに関連付けられている Shield Advanced 保護に名前を付け、その保護ページへのリンクを提供します。これは、個々のイベントのページで確認できます。
- [Automatic application layer DDoS mitigation] (アプリケーションレイヤーの DDoS の自動緩和) - Shield Advanced アプリケーションレイヤー DDoS 自動緩和がリソースのために有効になっているかどうかを示すために、アプリケーションレイヤーの保護に使用されます。有効になっている場合、これは、設定にアクセスして管理するためのリンクを提供します。これは、個々のイベントのページで確認できます。
- [Network layer automatic mitigation] (ネットワークレイヤーの自動緩和) – リソースがネットワークレイヤーで自動緩和されているかどうかを示します。リソースにネットワークレイヤーコンポーネントがある場合、これは有効になります。この情報は、個々のイベントのページで入手できます。

頻繁にターゲットが設定されるリソースの場合、Shield は、さらに繰り返し発生するイベントを防ぐために、過剰なトラフィックが沈静化した後も緩和策をそのままにすることができます。

> **Note**
>
> API オペレーションを通じて、保護対象リソースのイベント概要にアクセスすることもできます。 AWS Shield ListAttacks

AWS Shield Advanced イベントの詳細

イベントのコンソールページの下部のセクションで、イベントの検出、緩和策、および上位の寄稿者に関する詳細を確認できます。このセクションには、正当なトラフィックと望ましくないものである可能性があるトラフィックが混在している可能性があり、保護されたリソースに渡されたトラフィックと Shield 緩和によってブロックされたトラフィックの両方が表われている場合があります。

- [検出と緩和] – 観察されたイベントと、それに対して適用された緩和策に関する情報を提供します。イベントの緩和については、「DDoS イベントへの対応」を参照してください。
- [上位の寄稿者] – イベントに関係するトラフィックを分類し、Shield がカテゴリごとに特定したトラフィックの主要なソースを一覧表示します。アプリケーション層のイベントについては、上位の貢献者の情報を使用してイベントの性質を大まかに把握し、 AWS WAF ログはセキュリティの判断に使用します。詳細については、次のセクションを参照してください。

Shield アドバンスコンソールのイベント情報は、Shield アドバンスドメトリクスに基づいています。Shield アドバンスドメトリクスの詳細については、以下を参照してください。AWS Shield Advanced 指標

Amazon CloudFront や Amazon Route 53 のリソースには軽減メトリックスは含まれていません。なぜなら、これらのサービスは常に有効になっており、個々のリソースの軽減を必要としない軽減システムによって保護されているからです。

詳細セクションは、情報がインフラストラクチャ層またはアプリケーションレイヤーイベントのどちらに関するものであるかによって異なります。

アプリケーションレイヤーのイベントの詳細

イベントの検出、緩和策、および上位の寄稿者に関する詳細は、イベントのコンソールページの下部のセクションで確認できます。このセクションには、正当なトラフィックと望ましくない可能性のあるトラフィックが混在している可能性があり、保護対象リソースに渡されたトラフィックと、Shield Advancedの緩和策によってブロックされたトラフィックの両方が含まれる場合があります。

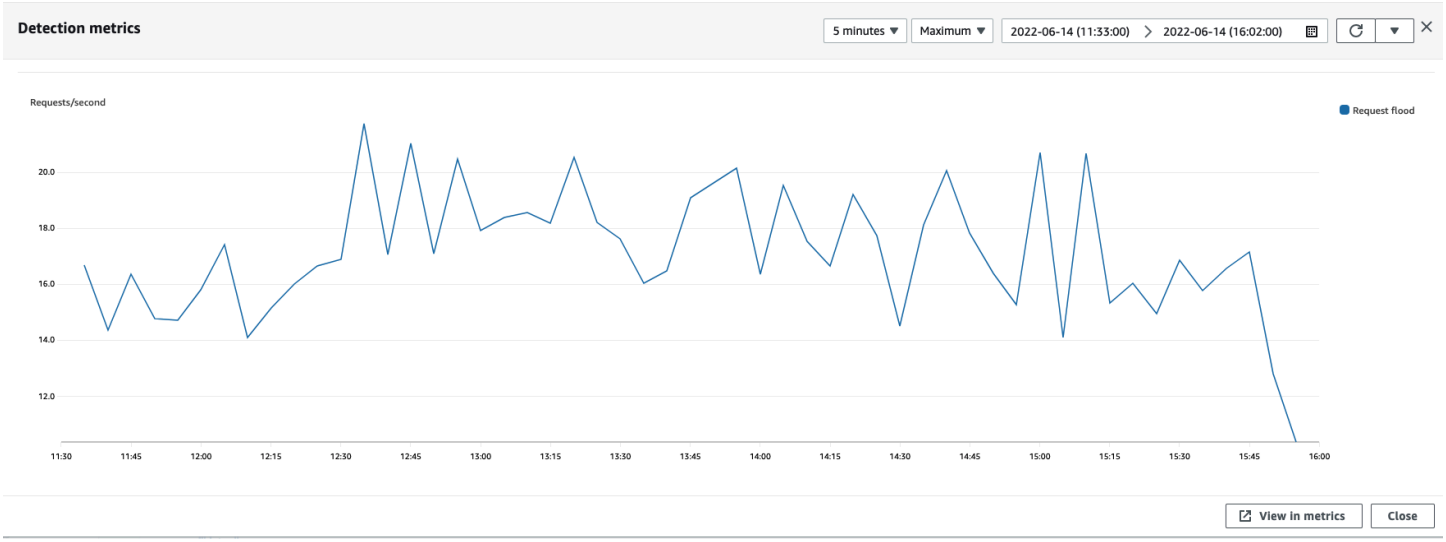
軽減策の詳細は、攻撃に対応して特別に展開されたルールや、ウェブ ACL で定義されているレートベースのルールなど、リソースに関連するウェブ ACL 内のすべてのルールに関するものです。アプリケーション層の DDoS の自動軽減機能をアプリケーションに対して有効にすると、緩和メトリクスにはそれらの追加ルールのメトリクスが含まれます。これらのアプリケーション層保護について詳しくは、を参照してください。AWS Shield Advanced アプリケーション層 (レイヤー 7) 保護

検出と緩和

アプリケーションレイヤー (レイヤー 7) のイベントの場合、[Detection and Mitigation] タブには、ログから取得した情報に基づく検出メトリクスが表示されます。 AWS WAF 緩和メトリクスは、望ましくないトラフィックをブロックするように設定された、関連付けられたウェブ ACL の AWS WAF ルールに基づいています。

Amazon CloudFront ディストリビューションでは、自動的に緩和策を適用するように Shield Advanced を設定できます。任意のアプリケーションレイヤーリソースを使用して、ウェブ ACL で独自の緩和ルールを定義することを選択したり、Shield レスポンスチーム (SRT、Shield Response Team) にサポートをリクエストしたりできます。これらのオプションについては、「DDoS イベントへの対応」を参照してください。

次のスクリーンショットは、数時間後に沈静化したアプリケーションレイヤーイベントの検出メトリクスの例を示しています。



緩和ルールが有効になる前に沈静化するイベントトラフィックは、緩和メトリクスに表れません。これにより、検出グラフに表示されるウェブリクエストのトラフィックと、緩和グラフに表示される許可およびブロックメトリクスが異なることがあります。

上位の寄稿者

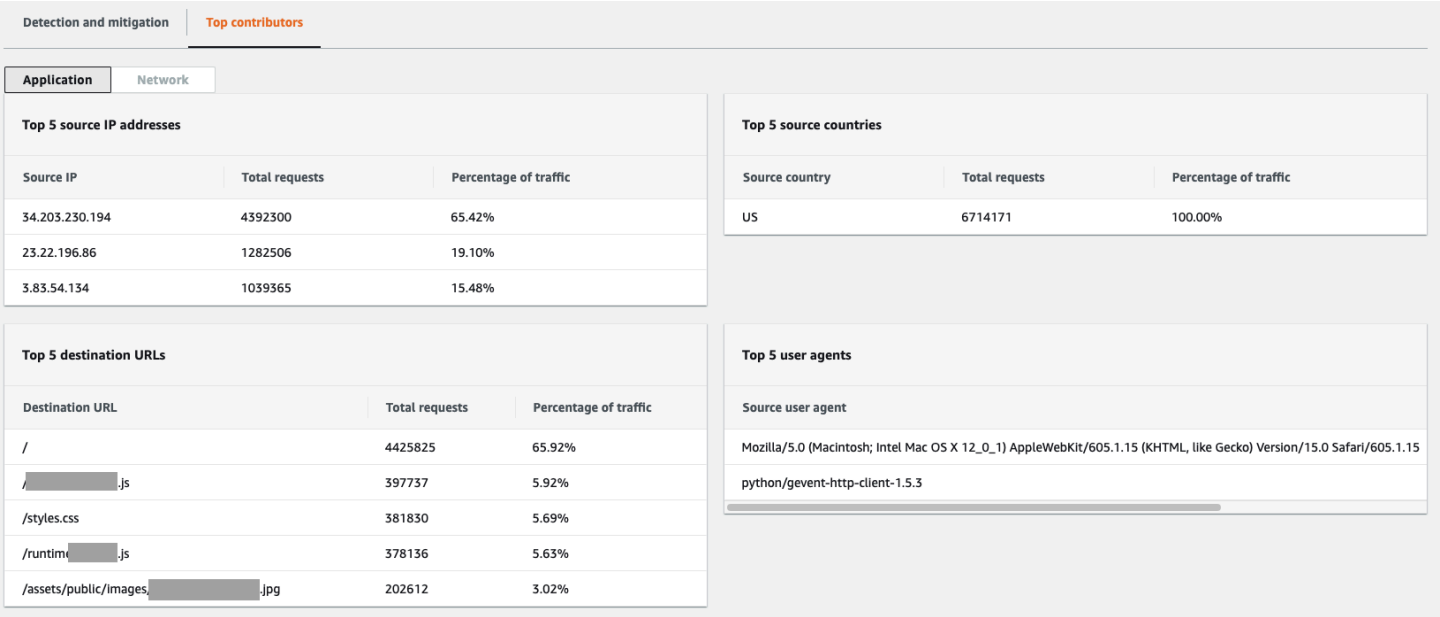
アプリケーションレイヤーイベントの [トップコントリビューター] タブには、 AWS WAF 取得したログに基づいて Shield がイベントについて特定した上位 5 人のコントリビューターが表示されます。Shield は、上位の寄稿者情報をソースIP、送信元国、送信先 URLなどのディメンションで分類します。

> **Note**
>
> アプリケーションレイヤーイベントに寄与しているトラフィックに関する最も正確な情報については、ログを使用してください。 AWS WAF

Shield アプリケーションレイヤーの上位の寄稿者情報は、攻撃の性質を大まかに把握するためにのみ使用し、それに基づいてセキュリティ上の決定を行うべきではありません。アプリケーション層のイベントの場合、攻撃の原因を把握し、軽減戦略を考案するには、 AWS WAF ログが最適な情報源です。

Shield の上位貢献者情報は、 AWS WAF 必ずしもログ内のデータを完全に反映しているとは限りません。Shield は、ログを取り込む際に、ログから完全なデータを取得することよりも、システムパフォーマンスへの影響を減らすことを優先します。その結果、Shield で分析に使用できるデータの粒度が失われる可能性があります。ほとんどの場合、情報の大部分は入手可能ですが、上位の寄稿者のデータは、攻撃によってはある程度偏る可能性があります。

次のスクリーンショットは、アプリケーションレイヤーイベントの [上位の寄稿者] タブの例を示しています。



コントリビューターに関する情報は、正当なトラフィックと望ましくない可能性があるトラフィックの両方に対するリクエストに基づいています。大量のイベントやリクエストソースが高度に分散されていないイベントは、識別可能な上位の寄稿者を持っている可能性が比較的高いです。大幅に分散した攻撃では、任意の数のソースが存在する可能性があり、攻撃の上位の寄稿者を特定することが困難

です。Shield Advanced が特定のカテゴリの重要な寄稿者を特定しない場合、データは利用不可として表示されます。

インフラストラクチャレイヤーのイベントの詳細

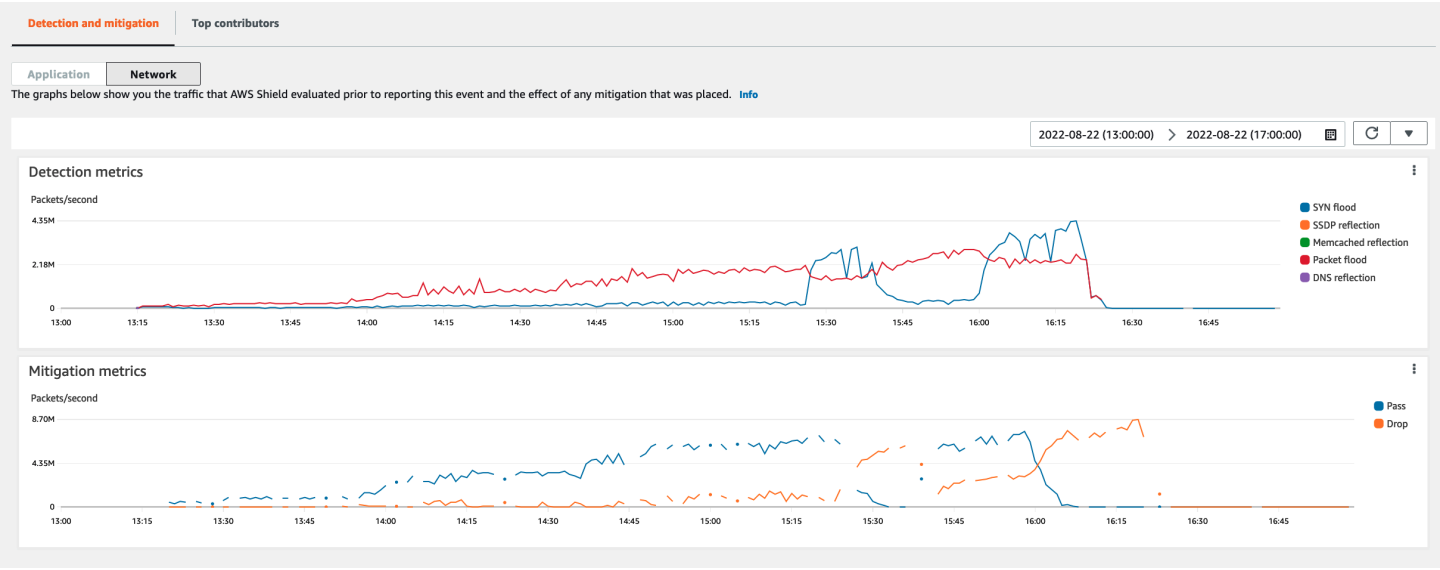
イベントのコンソールページの下部のセクションで、イベントの検出、緩和策、および上位の寄稿者に関する詳細を確認できます。このセクションには、正当なトラフィックと望ましくないものである可能性があるトラフィックが混在している可能性があり、保護されたリソースに渡されたトラフィックと Shield 緩和によってブロックされたトラフィックの両方が表われている場合があります。

検出と緩和

インフラストラクチャレイヤー (レイヤー 3 または 4) イベントの場合、[Detection and mitigation] (検出と緩和) タブには、サンプリングされたネットワークフローに基づく検出メトリクスと、緩和システムによって観察されたトラフィックに基づく緩和メトリクスが表示されます。緩和メトリクスは、リソースへのトラフィックをより正確に測定します。

Shield は、保護対象リソースタイプの Elastic IP (EIP)、Classic Load Balancer (CLB)、Application Load Balancer (ALB)、およびスタンダードアクセラレーターの軽減策を自動的に作成します。 AWS Global Accelerator EIP AWS Global Accelerator アドレスと標準アクセラレータの軽減メトリクスは、通過したパケットとドロップされたパケットの数を示します。

次のスクリーンショットは、インフラストラクチャレイヤーイベントの [Detection and mitigation] (検出と緩和) タブの例を示しています。

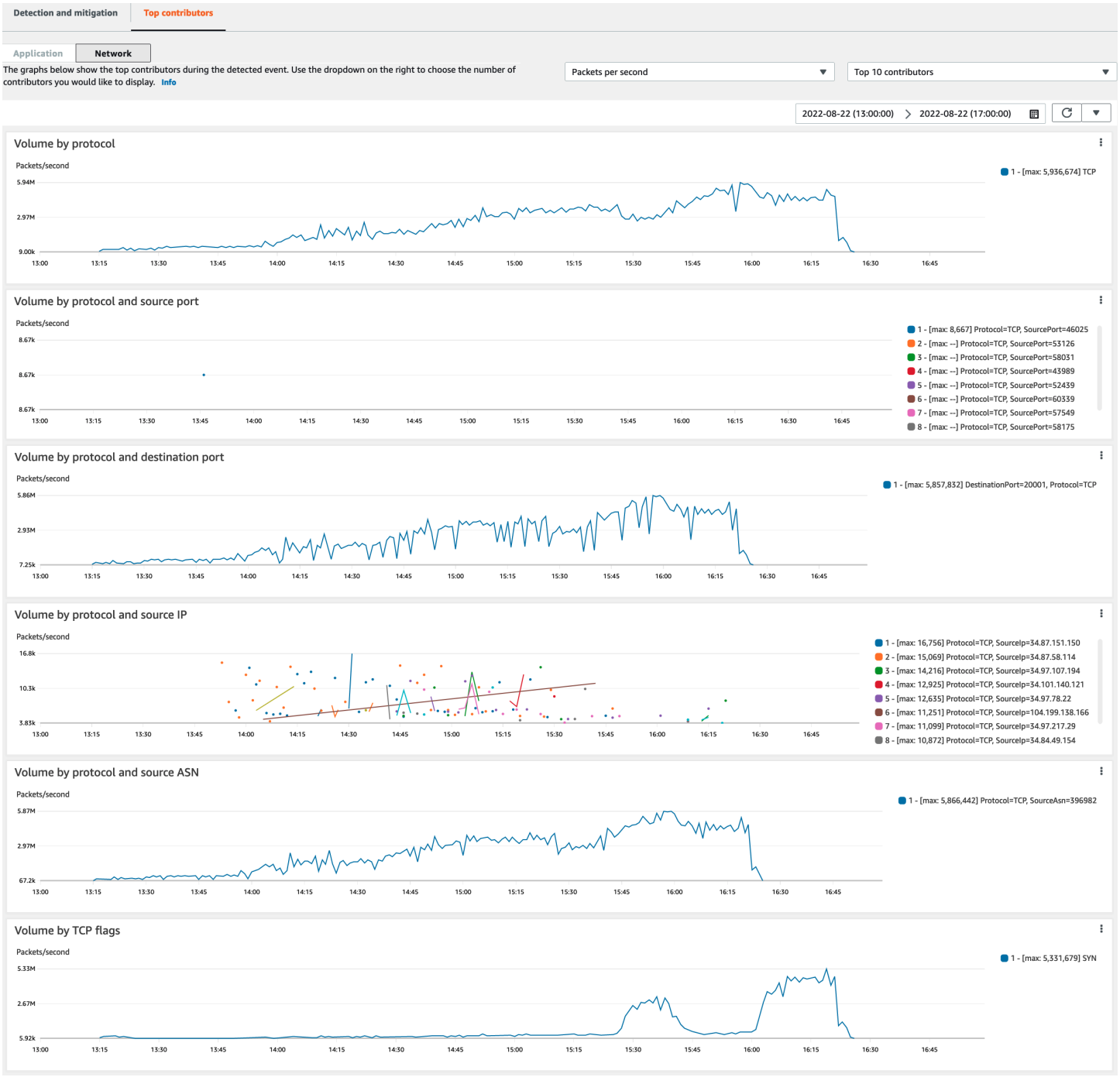


Shield が緩和策を行う前に沈静化するイベントトラフィックは、緩和メトリクスに表れません。これにより、検出グラフに表示されるトラフィックと、緩和グラフに表示されるパスおよびドロップメトリクスが異なることがあります。

上位の寄稿者

インフラストラクチャレイヤーイベントの [上位の寄稿者] タブには、いくつかのトラフィックディメンションで最大 100 の上位の寄稿者に関するメトリクスが一覧表示されます。詳細には、少なくとも 5 つの重要なトラフィックソースを特定できる任意のディメンションのネットワークレイヤープロパティが含まれます。トラフィックのソースの例としては、ソース IP とソース ASN があります。

次のスクリーンショットは、インフラストラクチャレイヤーイベントの [上位の寄稿者] タブの例を示しています。



コントリビューターメトリクスは、正当なトラフィックと望ましくない可能性があるトラフィックの両方についてサンプリングされたネットワークフローに基づいています。大量のイベントやトラフィックソースが高度に分散されていないイベントは、識別可能な上位の寄稿者を持っている可能性が比較的高いです。大幅に分散した攻撃では、任意の数のソースが存在する可能性があり、攻撃に対する上位の寄稿者を特定することが困難です。Shield が特定のメトリクスまたはカテゴリの重要な寄稿者を特定しない場合、データは利用不可として表示されます。

インフラストラクチャレイヤー DDoS 攻撃では、トラフィックソースがスプーフィングまたはリフレクトされる可能性があります。スプーフィングされたソースは、攻撃者によって意図的に偽造されます。反映されたソースは、検出されたトラフィックの実際のソースですが、攻撃に積極的に参加しているわけではありません。例えば、攻撃者は、通常は正当なインターネット上のサービスの攻撃を反射して、ターゲットに対する大規模で増幅されたトラフィックのフラッドを生成する可能性があります。この場合、ソース情報は、有効ではあるものの、実際の攻撃元ではない可能性があります。これらの要因により、パケットヘッダーに基づいてソースをブロックする緩和技術の実行可能性が制限される可能性があります。

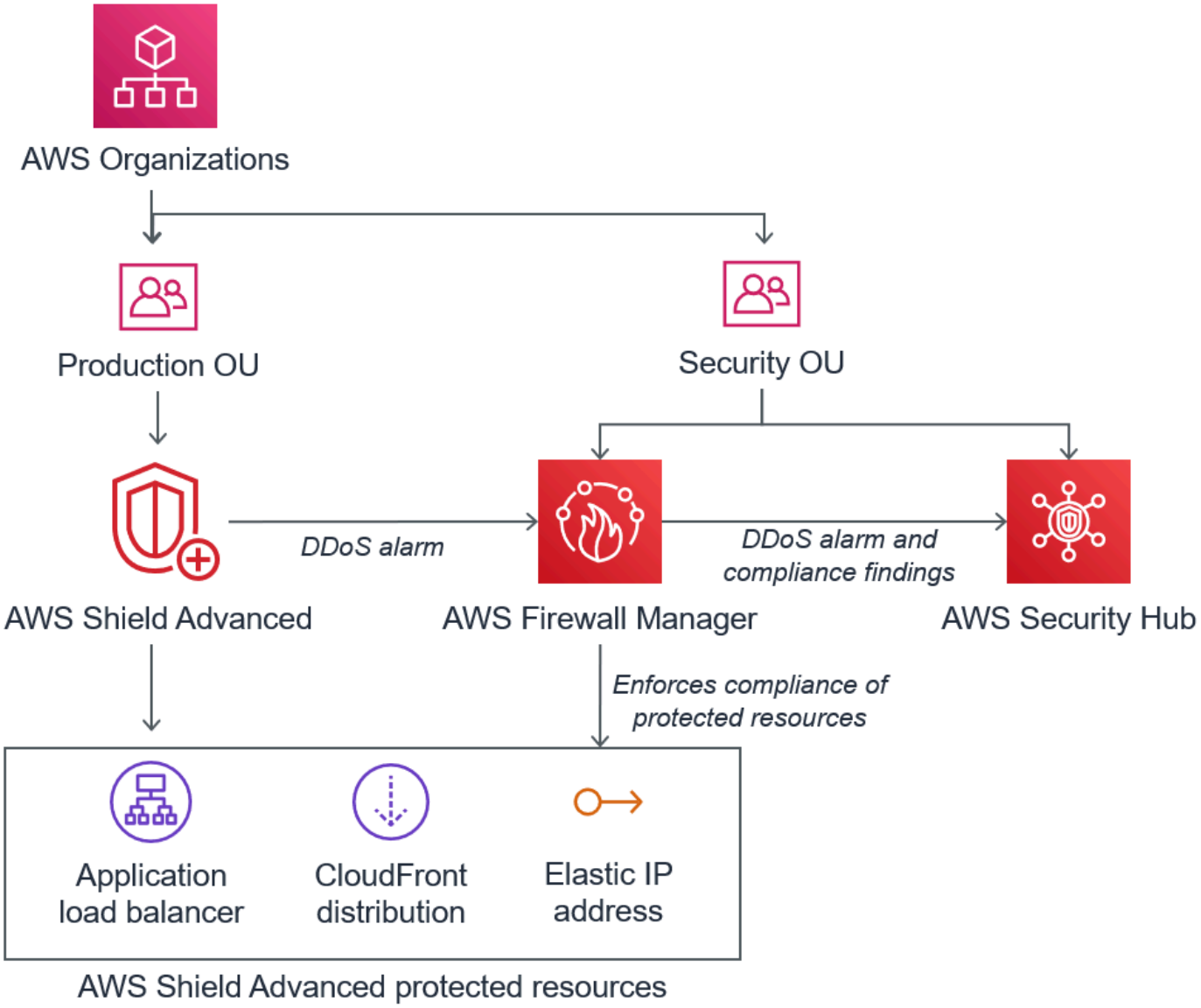
アカウント全体にわたるイベントの可視性

AWS Firewall Manager AWS Security Hub とを使用して、 AWS Shield Advanced 複数のアカウントの保護対象リソースを管理および監視できます。

Firewall Manager を使用すると、すべてのアカウントで DDoS 保護のコンプライアンスを報告および適用する Shield Advanced セキュリティポリシーを作成できます。Firewall Manager は、Shield Advanced ポリシーの対象となる新しいリソースへの保護の追加を含め、保護されたリソースをモニタリングします。

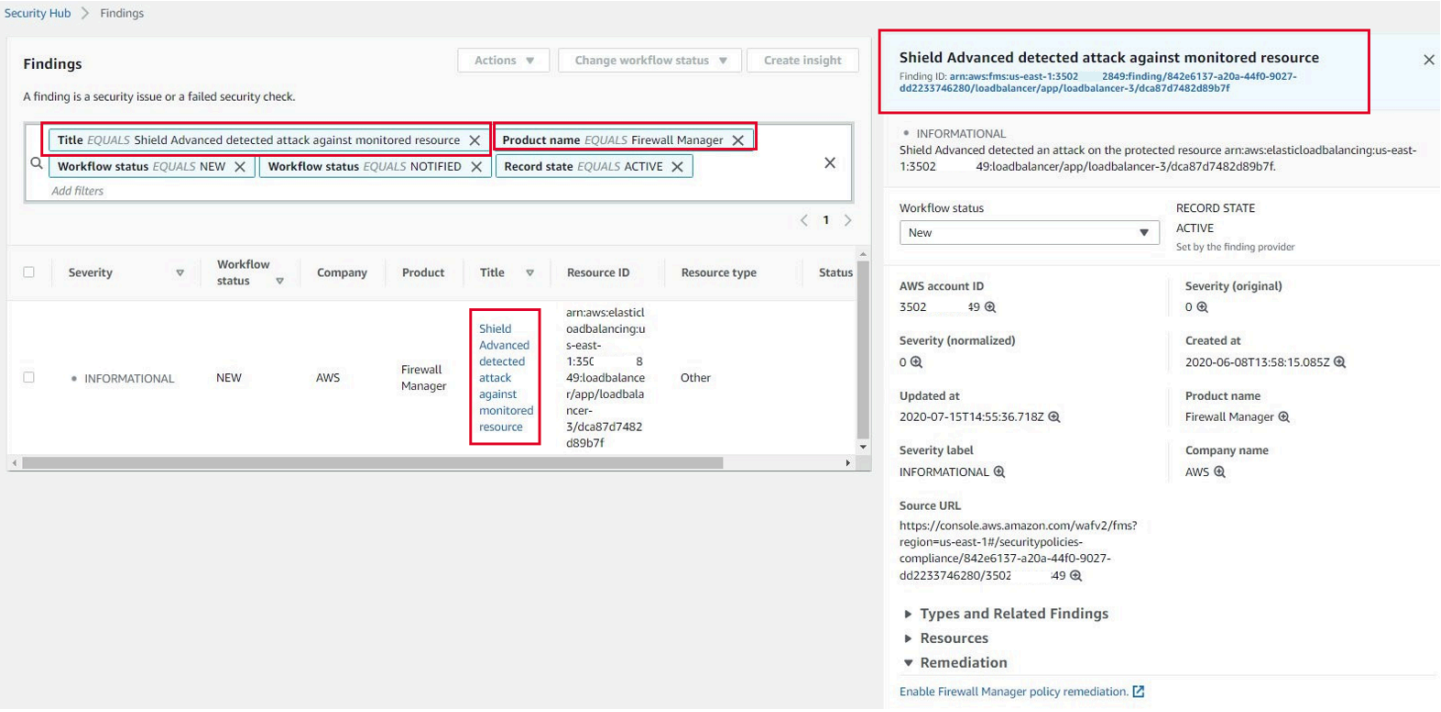
AWS Security Hub ファイアウォールマネージャーをと統合すると、Firewall Manager Shield アドバンスドセキュリティポリシーに準拠していないリソースを特定したときに、Shield AdvancedとFirewall Manager のコンプライアンス結果によって検出されたDDoSイベントを報告する単一のダッシュボードを作成できます。

次の図は、Firewall Manager とSecurity Hub を使用して Shield Advanced で保護されたリソースをモニタリングする一般的なアーキテクチャを示しています。



Firewall Manager と Security Hub を統合すると、 AWSで実行するアプリケーションの他のアラートやコンプライアンスステータス情報とともに、セキュリティの検出結果を 1 か所で表示できます。

次のスクリーンショットは、このタイプの統合時に Security Hub コンソール内で Shield Advanced イベントについて表示される情報を強調するためのものです。



Firewall Manager と Security Hub を Shield Advanced と統合して、保護対象アカウント全体のイベントとコンプライアンスの監視を一元化する方法については、 AWS セキュリティブログの「DDoS イベントの集中監視を設定し、準拠していないリソースを自動修復する」を参照してください。

DDoS イベントへの対応

AWS ネットワークおよびトランスポート層 (レイヤー 3 およびレイヤー 4) の分散型サービス拒否 (DDoS) 攻撃を自動的に軽減します。Shield Advanced を使用して Amazon EC2 インスタンスを保護する場合、攻撃を受けている最中に、Shield Advanced は、Amazon VPC ネットワーク ACL を AWS ネットワークの境界に自動的にデプロイします。これにより、Shield Advanced は、大規模な DDoS イベントに対する保護を提供できます。ネットワーク ACL の詳細については、「ネットワーク ACL」を参照してください。

アプリケーション層 (レイヤー 7) の DDoS 攻撃では、 AWS AWS Shield Advanced アラームを通じて顧客を検知して通知しようとします。 CloudWatch デフォルトでは、有効なユーザートラフィックが誤ってブロックされないように、緩和策は自動的に適用されません。

アプリケーションレイヤー (レイヤー 7) リソースでは、攻撃に対応するために次のオプションを使用できます。

- [Provide your own mitigations] (独自の緩和策を提供する) – 独自に攻撃を調査して緩和することができます。詳細については、「アプリケーションレイヤー DDoS 攻撃の手動による緩和」を参照してください。
- [Contact support] (サポートに問い合わせる) – Shield Advanced をご利用の場合は、AWS Support Center に問い合わせて緩和に関するサポートを受けることができます。重大かつ緊急のケースは DDoS エキスパートに直接、ルーティングされます。詳細については、「アプリケーションレイヤー DDoS 攻撃を受けている最中のサポートセンターへの問い合わせ」を参照してください。

さらに、攻撃が発生する前に、次の緩和オプションをプロアクティブに有効にできます。

- Amazon CloudFront ディストリビューションの自動緩和 — このオプションでは、Shield Advanced がウェブ ACL 内の緩和ルールをユーザーに代わって定義および管理します。アプリケーションレイヤーの自動緩和の詳細については、「Shield Advanced アプリケーションレイヤー DDoS 自動緩和」を参照してください。
- プロアクティブエンゲージメント — アプリケーションの 1 AWS Shield Advanced つに対する大規模なアプリケーションレイヤー攻撃を検出すると、SRT は事前に連絡します。SRT は DDoS イベントをトリアージし、 AWS WAF 緩和策を作成します。SRT はお客様に連絡し、お客様の同意を得て、 AWS WAF ルールを適用することができます。このオプションの詳細については、「プロアクティブな関与の設定」を参照してください。

アプリケーションレイヤー DDoS 攻撃を受けている最中のサポートセンターへの問い合わせ

AWS Shield Advanced お客様であれば、AWS Support センターに連絡して緩和策の支援を受けることができます。重大かつ緊急のケースは DDoS エキスパートに直接、ルーティングされます。これにより AWS Shield Advanced、複雑なケースは、Amazon.com、 AWS AWSおよびその子会社の保護に豊富な経験を持つShield 対応チーム (SRT) に報告できます。SRT 関数の詳細については、「Shield Response Team (SRT) のサポート」を参照してください。

Shield Response Team (SRT) のサポートを受けるには、AWS Support Center に問い合わせてください。ケースへの応答時間は、選択した重要度と応答回数によって異なります (「AWS Support プラン」ページを参照)。

次のオプションを選択してください。

- ケースタイプ: テクニカルサポート
- サービス: Distributed Denial of Service (DDoS)

- カテゴリー:インバウンド先 AWS
- 重要度: 適切なオプションを選択してください

担当者と話し合う際には、 AWS Shield Advanced お客様が DDoS 攻撃を受けている可能性があることを説明してください。担当者がお客様の問い合わせを適切な DDoS エキスパートに取り次ぎます。[Distributed Denial of Service (DDoS)] サービスタイプを使用して AWS Support Center でケースを開いた場合は、チャットや電話で DDoS エキスパートと直接お話しいただけます。DDoS サポートエンジニアは、攻撃を特定したり、 AWS アーキテクチャの改善を推奨したり、DDoS AWS 攻撃を軽減するためのサービスの利用に関するガイダンスを提供したりするお手伝いをします。

アプリケーションレイヤー攻撃の場合、SRT は疑わしいアクティビティを分析するのをサポートします。リソースについて自動緩和が有効になっている場合、SRT は Shield Advanced が攻撃に対して自動的に実施している緩和策を確認できます。いずれの場合でも、SRT は問題の確認と緩和をサポートできます。SRT が推奨する緩和策では、多くの場合 SRT AWS WAF がアカウントのウェブアクセスコントロールリスト (ウェブ ACL) を作成または更新する必要があります。SRT がこの作業を行うには、お客様の許可が必要となります。

> **Important**
>
> 有効にする一環として AWS Shield Advanced、の手順に従い、Shield Response Team (SRT) のためのアクセス権の設定攻撃時に支援するのに必要な権限を積極的にSRTに提供することをお勧めします。事前に許可を付与することで、実際に攻撃が発生した場合の遅延を防ぐことができます。

SRT は、DDoS 攻撃を分類して、攻撃シグネチャとパターンを識別できるようにお客様を支援します。ユーザーの同意を得て、SRT AWS WAF は攻撃を軽減するためのルールを作成して展開します。

また、緩和策を確認したり、カスタム緩和策を開発してデプロイしたりするために、攻撃の可能性が生じる前であっても、または生じている最中に SRT に問い合わせることもできます。例えば、ウェブアプリケーションを実行していて、ポート 80 と 443 だけを開く必要がある場合は、SRT と連携してポート 80 と 443 だけを [Allow] (許可) するようにウェブ ACL を事前設定できます。

SRT への連絡と承認はアカウントレベルで行います。つまり、Firewall Manager Shield Advanced ポリシー内で Shield Advanced を使用する場合、アカウント所有者 (Firewall Manager 管理者ではない) は SRT にサポートを依頼する必要があります。Firewall Manager 管理者は、所有しているアカウントに関してのみ SRT に問い合わせることができます。

アプリケーションレイヤー DDoS 攻撃の手動による緩和

リソースのイベントページのアクティビティが DDoS 攻撃であると判断した場合は、ウェブ ACL AWS WAF に独自のルールを作成して攻撃を軽減できます。Shield Advancedをご利用でない場合は、これが唯一のオプションです。 AWS WAF AWS Shield Advanced には追加料金なしで含まれています。ウェブ ACL でのルール作成の詳細については、「AWS WAF ウェブアクセスコントロールリスト (ウェブ ACLs)」を参照してください。

を使用する場合は AWS Firewall Manager、 AWS WAF ルールをFirewall Manager AWS WAF ポリシーに追加できます。

アプリケーションレイヤー DDoS 攻撃の可能性がある状況を手動で緩和するには

1. 異常な動作に一致する条件を使用して、ウェブ ACL にルールステートメントを作成します。まず、一致するリクエストをカウントするように設定します。ウェブ ACL およびルールステートメントの設定については、「ウェブ ACL ルールおよびルールグループの評価」および「AWS WAF 保護機能のテストと調整」を参照してください。

   > **Note**
   >
   > 最初に Block の代わりにルールアクション Count を使用して、常に最初にルールをテストしてください。新しいルールが正しいリクエストを識別していることを確認した後、リクエストをブロックするためにそれらを変更できます。

2. リクエスト数をモニタリングして、一致するリクエストをブロックするかどうかを決定します。リクエストの量が異常に多い状況が継続しており、そのような状況を引き起こしているリクエストをルールがキャプチャしていると確信できる場合は、ウェブ ACL のルールを変更してそのリクエストをブロックします。
3. イベントページのモニタリングを続行して、トラフィックが希望どおりに処理されるようにします。

AWS には設定済みのテンプレートが用意されているため、すぐに使い始めることができます。テンプレートには、一般的な Web AWS WAF ベースの攻撃をブロックするためにカスタマイズして使用できる一連のルールが含まれています。詳細については、「AWS WAF セキュリティオートメーション」を参照してください。

でのクレジットのリクエスト AWS Shield Advanced

にサブスクライブ AWS Shield Advanced していて、Shield Advanced で保護されたリソースの使用率を高める DDoS 攻撃が発生した場合は、Shield Advanced によって緩和されない範囲で、使用率の増加に関連する料金の Shield Advanced サービスクレジットをリクエストできます。

> **Note**
>
> このプロセスで受け取ったクレジットは、Shield Advanced の使用にのみ適用されます。Shield Advanced クレジットは、他の サービスでは使用できません。

クレジットは、次のタイプの請求でのみ使用できます。

- Shield Advanced データ転送アウト
- Amazon CloudFront HTTP/HTTPS リクエスト
- CloudFront データ転送アウト
- Amazon Route 53 クエリ
- AWS Global Accelerator 標準アクセラレーターデータ転送
- Application Load Balancer のロードバランサー容量ユニット
- 攻撃に対応して自動スケーリングポリシーによって作成、保護された Amazon Elastic Compute Cloud (Amazon EC2) インスタンスのインスタンスコスト

クレジットをリクエストするための前提条件

クレジットの付与を受けるための要件を満たすには、攻撃が始まる前に、次のことを完了している必要があります。

- クレジットをリクエストするリソースに Shield Advanced 保護を追加しておく必要があります。攻撃を受けている最中に追加された保護対象リソースは、コスト保護の対象外です。

> **Note**
>
> で Shield Advanced を有効に AWS アカウント しても、個々のリソースの Shield Advanced 保護は自動的に有効になりません。

Shield Advanced を使用して AWS リソースを保護する方法の詳細については、「」を参照してくださいAWS Shield AdvancedAWS リソースへの保護の追加。

- 適用可能なリソース CloudFront と Application Load Balancer で保護されたリソースについては、AWS WAF ウェブ ACL を関連付け、ウェブ ACL にレートベースのルールを Blockモードで実装する必要があります。 AWS WAF のレートベースルールの詳細については、「レートベースのルールステートメント」を参照してください。ウェブ ACLs「」を参照してくださいAWS WAF ウェブアクセスコントロールリスト (ウェブ ACLs)。 AWS
- DDoS 攻撃を受けている最中のコストを最小限に抑えるようにアプリケーションを設定するには、「DDoS レジリエンシーに関するAWS のベストプラクティス」の適切なベストプラクティスを実装しておく必要があります。

クレジットの申請方法

クレジットの付与を受けるために要件を満たすには、攻撃が発生した請求月の直後から 15 日以内にクレジットリクエストを送信する必要があります。

クレジットを申請するには、AWS Support Center を通じて請求ケースを提出してください。リクエストには次の内容を含めます。

- 件名の「DDoS Concession」という文字
- クレジットをリクエストしている各イベントまたは可用性の中断の日時
- 影響を受けた AWS サービスと特定のリソース

リクエストを送信すると、 AWS Shield Response Team (SRT) は DDoS 攻撃が発生したかどうか、発生した場合は、保護されたリソースが DDoS 攻撃を吸収するためにスケーリングされたかどうかを検証します。が、保護されたリソースが AWS DDoS 攻撃を吸収するためにスケールされたと判断した場合、 AWS は、 が DDoS 攻撃によって引き起こされた AWS と判断したトラフィックのその部分にクレジットを発行します。クレジットは 12 か月間有効です。

AWS Shield サービスの利用におけるセキュリティ

AWS クラウドセキュリティは最優先事項です。 AWS 顧客は、最もセキュリティに敏感な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の恩恵を受けることができます。

> **Note**
>
> このセクションでは、Shield Advanced 保護など、 AWS AWS Shield AWS サービスとそのリソースを使用する際の標準的なセキュリティガイダンスを提供します。
> Shield と Shield AWS Advancedを使用してリソースを保護する方法については、 AWS Shield ガイドの残りの部分を参照してください。

AWS セキュリティはユーザーとユーザー間で共有される責任です。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して説明しています。

- クラウドのセキュリティ — AWS AWS AWS クラウドクラウド内でサービスを実行するインフラストラクチャを保護する責任があります。 AWS また、安全に使用できるサービスも提供します。セキュリティの有効性は、AWS コンプライアンスプログラムの一環として、サードパーティーの審査機関によって定期的にテストおよび検証されています。Shield に適用するコンプライアンスプログラムの詳細については、「コンプライアンスプログラムによる対象範囲内のAWS のサービス」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、 AWS 使用する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、組織の要件、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、Shield 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。次のトピックでは、セキュリティとコンプライアンスの目的を達成するために Shield を設定する方法を示します。また、Shield AWS リソースの監視と保護に役立つ他のサービスの使用方法についても学習します。

トピック

Shield でのデータ保護

AWS のデータ保護には、https://aws.amazon.com/compliance/shared-responsibility-model/、（ 責任分担モデル ）が適用されます AWS Shield。このモデルで説明したように、 AWS は、 AWS クラウドすべてを稼働させ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。お客様は、このインフラストラクチャでホストされてい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、「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。欧州でのデータ保護の詳細については、「AWS セキュリティブログ」に投稿された「AWS 責任共有モデルおよび GDPR」のブログ記事を参照してください。

データ保護の観点から、 AWS アカウント 認証情報を保護し、 AWS IAM Identity Center 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。こうすると、それぞれのジョブを遂行するために必要なアクセス許可のみを各ユーザーに付与できます。また、以下の方法でデータを保護することをお勧めします。

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用してリソースと通信します。 AWS TLS 1.2、できれば TLS 1.3 が必要です。
- を使用して API とユーザーアクティビティの口ギングを設定します。 AWS CloudTrail
- AWS 暗号化ソリューションと、 AWS のサービスその中に含まれるデフォルトのセキュリティコントロールをすべて使用してください。
- Amazon Macie などの高度なマネージド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 を介してアクセスするときに FIPS 140-2 で検証された暗号モジュールが必要な場合は、FIPS エンドポイントを使用してください。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) 140-2」を参照してください。

お客様の E メールアドレスなどの機密情報やセンシティブ情報は、タグや名前フィールドなどの自由形式のフィールドに配置しないことを強くお勧めします。これには、コンソール、API AWS CLI、または AWS SDK AWS のサービス を使用して Shield やその他のユーザーと連携する場合も含まれます。名前に使用する自由記述のテキストフィールドやタグに入力したデータは、課金や診断ログに使用される場合があります。外部サーバーへの URL を提供する場合は、そのサーバーへのリクエストを検証するための認証情報を URL に含めないように強くお勧めします。

Shield のエンティティ (保護など) は、中国 (北京) や中国 (寧夏) など、暗号化が利用できない特定のリージョンを除き、保管時に暗号化されます。リージョンごとに一意の暗号化キーが使用されます。

の Identity and Access Management AWS Shield

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、Shield リソースを使用するための、認証 (サインイン) および認可 (アクセス許可を持つ) ができるユーザーを制御します。IAM は、追加料金なしで AWS のサービス 使用できる です。

トピック

対象者

AWS Identity and Access Management (IAM) の使用方法は、Shield で行う作業によって異なります。

サービスユーザー - Shield サービスを使用してジョブを実行する場合は、必要なアクセス許可と認証情報を管理者が用意します。作業を実行するためにさらに多くの Shield 機能を使用するとき、追加の許可が必要になる場合があります。アクセスの管理方法を理解しておくと、管理者に適切な許可をリクエストするうえで役立ちます。Shield の特徴にアクセスできない場合は、「AWS Shield ID 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。

サービス管理者 - 社内の Shield リソースを担当している場合は、通常、Shield へのフルアクセスがあります。サービスのユーザーがどの Shield 機能やリソースにアクセスするかを決めるのは管理者の仕事です。その後、IAM 管理者にリクエストを送信して、サービスユーザーの権限を変更する必要があります。このページの情報を点検して、IAM の基本概念を理解してください。お客様の会社にて Shield により IAM を利用する方法の詳細については、と IAM の AWS Shield 連携方法 をご参照ください。

IAM 管理者 - IAM 管理者は、Shield へのアクセスを管理するためのポリシーの作成方法について、詳細を知りたい場合があります。IAM で使用できる Shield 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、「AWS Shield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。

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。として、IAM ユーザーとして AWS アカウントのルートユーザー、または IAM ロールを引き受けて認証 (にサインイン AWS) される必要があります。

ID ソースを介して提供された認証情報を使用して、フェデレーティッド ID AWS として にサインインできます。 AWS IAM Identity Center （ IAM Identity Center) ユーザー、会社のシングルサインオン認証、Google または Facebook の認証情報は、フェデレーション ID の例です。フェデレーティッド ID としてサインインする場合、IAM ロールを使用して、前もって管理者により ID フェデレーションが設定されています。フェデレーション AWS を使用して にアクセスすると、間接的にロールを引き受けることになります。

ユーザーのタイプに応じて、 AWS Management Console または AWS アクセスポータルにサインインできます。へのサインインの詳細については AWS、「 ユーザーガイド」の「 へのサインイン AWS アカウント方法AWS サインイン 」を参照してください。

AWS プログラムで にアクセスする場合、 は Software Development Kit (SDK) とコマンドラインインターフェイス (CLI) AWS を提供し、認証情報を使用してリクエストに暗号で署名します。 AWS ツールを使用しない場合は、リクエストに自分で署名する必要があります。推奨される方法を使用してリクエストを自分で署名する方法の詳細については、IAM ユーザーガイドの API AWS リクエストの署名を参照してください。

使用する認証方法を問わず、追加セキュリティ情報の提供をリクエストされる場合もあります。例えば、 AWS では、多要素認証 (MFA) を使用してアカウントのセキュリティを向上させることをお勧めします。詳細については、「AWS IAM Identity Center ユーザーガイド」の「Multi-factor authentication」(多要素認証) および「IAM ユーザーガイド」の「AWSでの多要素認証 (MFA) の使用」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、アカウント内のすべての 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 1 つのサインインアイデンティティから始めます。この ID は AWS アカウント ルートユーザーと呼ばれ、アカウントの作成に使用した E メールアドレスとパスワードでサインインすることでアクセスできます。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。ルートユーザーの認証情報は保護し、ルートユーザーでしか実行できないタスクを実行するときに使用します。ルートユーザーとしてサインインする必要があるタスクの完全なリストについては、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。

フェデレーティッドアイデンティティ

ベストプラクティスとして、管理者アクセスを必要とするユーザーを含む人間のユーザーに、一時的な認証情報を使用して にアクセスするための ID プロバイダーとのフェデレーションの使用を要求 AWS のサービス します。

フェデレーティッド ID は、エンタープライズユーザーディレクトリ、ウェブ ID プロバイダー、AWS Directory Service、Identity Center ディレクトリのユーザー、または ID ソースを通じて提供された認証情報 AWS のサービス を使用して にアクセスするユーザーです。フェデレーティッド ID がにアクセスすると AWS アカウント、ロールを引き受け、ロールは一時的な認証情報を提供します。

アクセスを一元管理する場合は、 AWS IAM Identity Centerを使用することをお勧めします。IAM Identity Center でユーザーとグループを作成することも、独自の ID ソース内のユーザーとグループのセットに接続して同期して、すべての AWS アカウント とアプリケーションで使用することもできます。IAM Identity Center の詳細については、「AWS IAM Identity Center ユーザーガイド」の「What is IAM Identity Center?」(IAM Identity Center 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、単一のユーザーまたはアプリケーションに対して特定のアクセス許可 AWS アカウント を持つ 内のアイデンティティです。可能であれば、パスワードやアクセスキーなどの長期的な認証情報を保有する IAM ユーザーを作成する代わりに、一時的な認証情報を使用することをお勧めします。ただし、IAM ユーザーでの長期的な認証情報が必要な特定のユースケースがある場合は、アクセスキーをローテーションすることをお勧めします。詳細については、IAM ユーザーガイドの長期的な認証情報を必要とするユースケースのためにアクセスキーを定期的にローテーションするを参照してください。

IAM グループは、IAM ユーザーの集団を指定するアイデンティティです。グループとしてサインインすることはできません。グループを使用して、複数のユーザーに対して一度に権限を指定できます。多数のユーザーグループがある場合、グループを使用することで権限の管理が容易になります。例えば、IAMAdminsという名前のグループを設定して、そのグループにIAM リソースを管理する許可を与えることができます。

ユーザーは、ロールとは異なります。ユーザーは 1 人の人または 1 つのアプリケーションに一意に関連付けられますが、ロールはそれを必要とする任意の人が引き受けるようになっています。ユー

ザーには永続的な長期の認証情報がありますが、ロールでは一時的な認証情報が提供されます。詳細については、「IAM ユーザーガイド」の「IAM ユーザー (ロールではなく) の作成が適している場合」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 AWS アカウント を持つ 内のアイデンティティです。これは IAM ユーザーに似ていますが、特定のユーザーには関連付けられていません。ロール を切り替える AWS Management Console ことで、 で IAM ロールを一時的に引き受けることができます。ロールを引き受けるには、 または AWS API AWS CLI オペレーションを呼び出すか、カスタム URL を使用します。ロールを使用する方法の詳細については、「IAM ユーザーガイド」の「IAM ロールの使用」を参照してください。

IAM ロールと一時的な認証情報は、次の状況で役立ちます:

- フェデレーションユーザーアクセス – フェデレーティッド ID に許可を割り当てるには、ロールを作成してそのロールの許可を定義します。フェデレーティッド ID が認証されると、その ID はロールに関連付けられ、ロールで定義されている許可が付与されます。フェデレーションの詳細については、「IAM ユーザーガイド」の「Creating a role for a third-party Identity Provider」(サードパーティーアイデンティティプロバイダー向けロールの作成) を参照してください。IAM Identity Center を使用する場合は、許可セットを設定します。アイデンティティが認証後にアクセスできるものを制御するため、IAM Identity Center は、権限セットを IAM のロールに関連付けます。アクセス許可セットの詳細については、「AWS IAM Identity Center ユーザーガイド」の「アクセス許可セット」を参照してください。
- 一時的な IAM ユーザー権限 - IAM ユーザーまたはロールは、特定のタスクに対して複数の異なる権限を一時的に IAM ロールで引き受けることができます。
- クロスアカウントアクセス - IAM ロールを使用して、自分のアカウントのリソースにアクセスすることを、別のアカウントの人物 (信頼済みプリンシパル) に許可できます。クロスアカウントアクセス権を付与する主な方法は、ロールを使用することです。ただし、一部の では AWS のサービス、 (ロールをプロキシとして使用する代わりに) ポリシーをリソースに直接アタッチできます。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。
- クロスサービスアクセス — 一部の は、他の の機能 AWS のサービス を使用します AWS のサービス。例えば、あるサービスで呼び出しを行うと、通常そのサービスによって Amazon EC2 でアプリケーションが実行されたり、Amazon S3 にオブジェクトが保存されたりします。サービスでは、呼び出し元プリンシパルの許可、サービスロール、またはサービスリンクロールを使用してこれを行う場合があります。

- 転送アクセスセッション (FAS) – IAM ユーザーまたはロールを使用して でアクションを実行する場合 AWS、ユーザーはプリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストのリクエストと組み合わせて使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。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。
- サービスロール - 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 のサービスにアクセス許可を委任するロールの作成」を参照してください。
- サービスにリンクされたロール – 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、サービスにリンクされた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。

- Amazon EC2 で実行されているアプリケーション – IAM ロールを使用して、EC2 インスタンスで実行され、 AWS CLI または AWS API リクエストを行うアプリケーションの一時的な認証情報を管理できます。これは、EC2 インスタンス内でのアクセスキーの保存に推奨されます。 AWS ロールを EC2 インスタンスに割り当て、そのすべてのアプリケーションで使用できるようにするには、インスタンスにアタッチされたインスタンスプロファイルを作成します。インスタンスプロファイルにはロールが含まれ、EC2 インスタンスで実行されるプログラムは一時的な認証情報を取得できます。詳細については、IAM ユーザーガイドのAmazon EC2 インスタンスで実行されるアプリケーションに IAM ロールを使用して許可を付与するを参照してください。

IAM ロールと IAM ユーザーのどちらを使用するかについては、IAM ユーザーガイドの(IAM ユーザーではなく) IAM ロールをいつ作成したら良いのか?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、 AWS ID またはリソースにアタッチします。ポリシーは AWS 、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ときにアクセス許可を定義する のオブジェクトです。 は、プリンシパル (ユーザー、ルートユーザー、またはロールセッション) 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。ポリシーでの権限に

より、リクエストが許可されるか拒否されるかが決まります。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構造と内容の詳細については、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 AWS JSON 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んなリソースにどんな条件でアクションを実行できるかということです。

デフォルトでは、ユーザーやロールに権限はありません。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。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き継ぐことができます。

IAM ポリシーは、オペレーションの実行方法を問わず、アクションの許可を定義します。例えば、`iam:GetRole`アクションを許可するポリシーがあるとします。そのポリシーを持つユーザーは、 AWS Management Console、、 AWS CLIまたは AWS API からロール情報を取得できます。

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ポリシーは、IAM ユーザーグループ、ユーザーのグループ、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 ポリシーの作成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、さらにインラインポリシーまたはマネージドポリシーに分類できます。インラインポリシーは、単一のユーザー、グループ、またはロールに直接埋め込まれています。管理ポリシーは、 内の複数のユーザー、グループ、ロールにアタッチできるスタンドアロンポリシーです AWS アカウント。管理ポリシーには、 AWS 管理ポリシーとカスタマー管理ポリシーが含まれます。マネージドポリシーまたは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 ユーザーガイドのマネージドポリシーとインラインポリシーの比較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。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制御できます。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は、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。

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。

アクセスコントロールリスト (ACL)

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。

Amazon S3、 AWS WAF、および Amazon VPC は、ACLs。ACL の詳細については、Amazon Simple Storage Service デベロッパ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。

その他のポリシータイプ

AWS は、一般的ではない追加のポリシータイプをサポートします。これらのポリシータイプでは、より一般的なポリシータイプで付与された最大の権限を設定できます。

- アクセス許可の境界 - アクセス許可の境界は、アイデンティティベースのポリシーによって IAM エンティティ (IAM ユーザーまたはロール) に付与できる権限の上限を設定する高度な機能です。エンティティにアクセス許可の境界を設定できます。結果として得られる権限は、エンティティのアイデンティティベースポリシーとそのアクセス許可の境界の共通部分になります。`Principal` フィールドでユーザーまたはロールを指定するリソースベースのポリシーでは、アクセス許可の境界は制限されません。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。アクセス許可の境界の詳細については、IAM ユーザーガイドのIAM エンティティのアクセス許可の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 (SCPs) – SCPs は、 の組織または組織単位 (OU) に対する最大アクセス許可を指定する JSON ポリシーです AWS Organizations。 AWS Organizations は、 AWS アカウント ビジネスが所有する複数の をグループ化して一元管理するサービスです。組織内のすべての機能を有効にすると、サービスコントロールポリシー (SCP) を一部またはすべてのアカウントに適用できます。SCP は、各 を含むメンバーアカウントのエンティティのアクセス許可を制限します AWS アカウントのルートユーザー。Organizations と SCP の詳細については、AWS Organizations ユーザーガイドの「SCP の仕組み」を参照してください。
- セッションポリシー - セッションポリシーは、ロールまたはフェデレーションユーザーの一時的なセッションをプログラムで作成する際にパラメータとして渡す高度なポリシーです。結果としてセッションの権限は、ユーザーまたはロールのアイデンティティベースポリシーとセッションポリシーの共通部分になります。また、リソースベースのポリシーから権限が派生する場合もあります。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。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権限を理解するのがさらに難しくなります。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。

と IAM の AWS Shield 連携方法

IAM を使用して Shield へのアクセスを管理する前に、Shield で利用できる IAM の機能について学びます。

で使用できる IAM の機能 AWS Shield

IAM 機能	Shield 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	はい
ポリシー条件キー (サービス固有)	はい
ACL	なし
ABAC (ポリシー内のタグ)	部分的
一時的な認証情報	あり
転送アクセスセッション (FAS)	あり
サービスロール	あり
サービスにリンクされたロール	あり

Shield 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を把握するには、「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。

Shield 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ポリシーをサポートする	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 ポリシーの作成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。プリンシパルは、それが添付されているユーザーまたはロールに適用されるため、アイデンティティベースのポリシーでは指定できません。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。

Shield 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、「AWS Shield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。

Shield 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制御できます。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は、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、アカウント全体、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。リソースベースのポリシーにクロスアカウントのプリンシパルを追加しても、信頼関係は半分しか確立されない点に注意してく

ださい。プリンシパルとリソースが異なる にある場合 AWS アカウント、信頼されたアカウントの IAM 管理者は、プリンシパルエンティティ (ユーザーまたはロール) にリソースへのアクセス許可も付与する必要があります。IAM 管理者は、アイデンティティベースのポリシーをエンティティにアタッチすることで権限を付与します。ただし、リソースベースのポリシーで、同じアカウントのプリンシパルへのアクセス権が付与されている場合は、アイデンティティベースのポリシーをさらに付与する必要はありません。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。

Shield のポリシーアクション

ポリシーアクションに対するサポート	あり

管理者は AWS JSON 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んなリソースにどんな条件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。ポリシーアクションの名前は通常、関連付けられた AWS API オペレーションと同じです。一致する API オペレーションのない許可のみのアクションなど、いくつかの例外があります。また、ポリシーに複数のアクションが必要なオペレーションもあります。これらの追加アクションは、依存アクションと呼ばれます。

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権限を付与するポリシーで使用されます。

Shield アクションのリストを確認するには、「サービス認証リファレンス」の「AWS Shieldで定義されるアクション」を参照してください。

Shield のポリシーアクションは、アクションの前に以下のプレフィックス を使用します。

```
shield
```

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": [
      "shield:action1",
      "shield:action2"
         ]
```

ワイルドカード (*) を使用すると、複数のアクションを指定することができます。たとえば、Shield で List で始まるすべてのアクションを指定するには、次のアクションを含めます。

```
"Action": "shield:List*"
```

Shield 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、「AWS Shield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。

Shield のポリシーリソース

ポリシーリソースに対するサポート	あり

管理者は AWS JSON 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どのような条件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。ステートメントには、Resource または NotResource 要素を含める必要があります。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。これは、リソースレベルの許可と呼ばれる特定のリソースタイプをサポートするアクションに対して実行できます。

オペレーションのリスト化など、リソースレベルの権限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": "*"
```

Shield リソースタイプとそれらの ARN のリストを確認するには、「サービス認証リファレンス」の「AWS Shieldで定義されるリソース」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は、「AWS Shieldで定義されるアクション」を参照してください。Shield リソースのサブセットへのアクセスを許可または拒否するには、ポリシーの resource 要素にリソースの ARN を含めます。

では AWS Shield、リソースは保護と攻撃です。これらのリソースには、次の表に示すとおり、一意の Amazon リソースネーム (ARN) が関連付けられています。

AWS Shield コンソールの名前	AWS Shield SDK/CLI の名前	ARN 形式
イベントまたは攻撃	AttackDetail	arn:aws:shield:: *account*:attack/*ID*
保護	Protection	arn:aws:shield:: *account*:protection/ *ID*

Shield リソースのサブセットへのアクセスを許可または拒否するには、ポリシーの resource 要素にリソースの ARN を含めます。Shield の ARN の形式は次のとおりです。

```
arn:partition:shield::account:resource/ID
```

[account] (アカウント)、*[resource]* (リソース)、および *[ID]* 変数を有効な値に置き換えます。有効な値は次のとおりです。

- ***#####*** : の ID AWS アカウント。値を指定する必要があります。
- ***resource***: Shield リソースのタイプ (attack または protection のいずれか)。
- ***ID***: Shield リソースの ID、または ワイルドカード (*)。ワイルドカードは、指定した AWS アカウントに関連付けられている、指定したタイプのすべてのリソースを示します。

例えば、次の ARN はアカウント 111122223333 のすべての保護を指定します。

```
arn:aws:shield::111122223333:protection/*
```

Shield リソースの ARN の形式は次のとおりです。

```
arn:partition:shield:region:account-id:scope/resource-type/resource-name/resource-id
```

ARN の仕様に関する一般情報については、「 Amazon Web Services 全般のリファレンス」の「Amazon リソースネーム (ARN)」を参照してください。

wafv2 リソースの ARN に固有の要件は以下の通りです。

- *region* : Amazon CloudFront ディストリビューションの保護に使用する Shield リソースの場合は、これを に設定しますus-east-1。それ以外の場合は、保護されたリージョンリソースで使用している領域を設定します。
- *####* : Amazon CloudFront ディストリビューションglobalで使用するか、 が AWS WAF サポートするリージョンリソースregionalで使用するスコープを に設定します。リージョンリソースは、Amazon API Gateway REST API、Application Load Balancer AWS AppSync GraphQL API、Amazon Cognito ユーザープール、 AWS App Runner サービス、および AWS Verified Access インスタンスです。
- *resource-type*: イベント用または攻撃用の attack、保護用の protection のいずれかの値を指定します。
- *resource-name*: Shield リソースに付けた名前を指定するか、ARN の他の仕様を満たすすべてのリソースを示すワイルドカード (*) を指定します。リソース名とリソース ID のどちらかを指定するか、両方にワイルドカードを指定する必要があります。
- *resource-id*: Shield リソースの ID を指定するか、ワイルドカード (*) を指定して ARN の他の仕様を満たすすべてのリソースを指定します。リソース名とリソース ID のどちらかを指定するか、両方にワイルドカードを指定する必要があります。

例えば、次の ARN は、リージョン us-west-1 におけるアカウント 111122223333 のリージョンレベルの範囲のすべてのウェブ ACL を指定します。

```
arn:aws:wafv2:us-west-1:111122223333:regional/webacl/*/*
```

次の ARN は、リージョン us-east-1 のアカウント 111122223333 に対して、グローバルスコープを持つ MyIPManagementRuleGroup というルールグループを指定します。

```
arn:aws:wafv2:us-east-1:111122223333:global/rulegroup/MyIPManagementRuleGroup/1111aaaa-bbbb-cccc-dddd-example-id
```

Shield 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、「AWS Shield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。

Shield 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 AWS JSON 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んなリソースにどんな条件でアクションを実行できるかということです。

Condition 要素 (または Condition ブロック) を使用すると、ステートメントが有効な条件を指定できます。Condition 要素はオプションです。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することで、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。

1 つのステートメントに複数の Condition 要素を指定する場合、または 1 つの Condition 要素に複数のキーを指定する場合、 AWS では AND 論理演算子を使用してそれらを評価します。1 つの条件キーに複数の値を指定すると、 は論理ORオペレーションを使用して条件 AWS を評価します。ステートメントの権限が付与される前にすべての条件が満たされる必要があります。

条件を指定する際にプレースホルダー変数も使用できます。例えば IAM ユーザーに、IAM ユーザー名がタグ付けされている場合のみリソースにアクセスできる権限を付与することができます。詳細については、IAM ユーザーガイドのIAM ポリシーの要素: 変数およびタグを参照してください。

AWS は、グローバル条件キーとサービス固有の条件キーをサポートします。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。

Shield の条件キーのリストを確認するには、「サービス認証リファレンス」の「AWS Shieldの条件キー」を参照してください。条件キーを使用できるアクションとリソースについては、「 で定義されるアクション AWS Shield」を参照してください。

Shield 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、「AWS Shield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。

Shield の 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許可を持つかをコントロールします。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。

Shield 付き ABAC

ABAC (ポリシー内のタグ) のサポート	部分的

属性ベースのアクセス制御 (ABAC) は、属性に基づいてアクセス許可を定義する認可戦略です。では AWS、これらの属性はタグ と呼ばれます。タグは、IAM エンティティ (ユーザーまたはロール) および多くの AWS リソースにアタッチできます。エンティティとリソースのタグ付けは、ABAC の最初の手順です。その後、プリンシパルのタグがアクセスしようとしているリソースのタグと一致した場合にオペレーションを許可するように ABAC ポリシーをします。

ABAC は、急成長する環境やポリシー管理が煩雑になる状況で役立ちます。

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 とは?を参照してください。ABAC をセットアップするステップ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に基づくアクセスコントロール (ABAC) を使用するを参照してください。

Shield 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

一部の AWS のサービス は、一時的な認証情報を使用してサインインすると機能しません。一時的な認証情報 AWS のサービス を使用する機能などの詳細については、IAM ユーザーガイドのAWS のサービス「IAM と連携する 」を参照してください。

ユーザー名とパスワード以外の AWS Management Console 方法で にサインインする場合、一時的な認証情報を使用します。例えば、会社の Single Sign-On (SSO) リンク AWS を使用して にアクセスすると、そのプロセスによって一時的な認証情報が自動的に作成されます。また、ユーザーとしてコンソールにサインインしてからロールを切り替える場合も、一時的な認証情報が自動的に作成されます。ロールの切り替えに関する詳細については、IAM ユーザーガイドのロールへの切り替え (コンソール)を参照してください。

一時的な認証情報は、 AWS CLI または AWS API を使用して手動で作成できます。その後、これらの一時的な認証情報を使用して . AWS recommends にアクセスできます AWS。これは、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することを推奨しています。詳細については、IAM の一時的セキュリティ認証情報を参照してください。

Shield の転送アクセスセッション

転送アクセスセッション (FAS) をサポート	あり

IAM ユーザーまたはロールを使用して でアクションを実行すると AWS、プリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストのリクエストと組み合わせて使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。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。

Shield のサービスロール

サービスロールに対するサポート	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 のサービスにアクセス許可を委任するロールの作成」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、Shield の機能が破損する可能性があります。Shield 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。

Shield のサービスリンクロール

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、サービスにリンクされた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。

Shield 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、「Shield Advanced 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。

AWS Shield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、Shield リソースを作成または変更するアクセス許可はありません。また、 AWS Management Console、 AWS Command Line Interface (AWS CLI)、または AWS API を使用してタスクを実行することもできません。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。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き受けることができます。

これらサンプルの 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 ポリシーの作成」を参照してください。

Shield 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、「サービス認証リファレンス」の「AWS Shield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

- ポリシーのベストプラクティス
- Shield コンソールの使用
- 自分の許可の表示をユーザーに許可する
- Shield Advanced の保護機能に対する読み取りアクセスの許可
- Shield、 CloudFront、への読み取り専用アクセスを許可する CloudWatch
- Shield へのフルアクセスを許可し CloudFront、 CloudWatch

ポリシーのベストプラクティス

ID ベースのポリシーは、ユーザーのアカウントで誰かが Shield 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を実行すると、 AWS アカウントに料金が発生する可能性があります。アイデンティティベースの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 管理ポリシーから始めて、最小権限の権限に移行する — ユーザーとワークロードへのアクセス権限の付与を開始するには、AWS 多くの一般的なユースケースで権限を付与する管理ポリシーを使用してください。これらのポリシーは、で利用できます。 AWS アカウント AWS ユースケースに固有のカスタマー管理ポリシーを定義して、権限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「AWS ジョブ機能の管理ポリシー」を参照してください。
- 最小特権を適用する – IAM ポリシーで許可を設定するときは、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。例えば、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。サービスアクションがなどの特定の用途で使用された場合は AWS のサービス、条件を使用して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 AWS CloudFormation。詳細については、「IAM ユーザーガイド」の [IAM JSON policy elements: Condition] (IAM JSON ポリシー要素：条件) を参照してください。
- IAM Access Analyzer 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 Access Analyzer 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 ポリシーの検証」を参照してください。
- 多要素認証 (MFA) が必要 — IAM ユーザーまたは root 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、セキュリティを強化するために MFA をオンにしてください。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 保護 API アクセスの設定」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。

Shield コンソールの使用

AWS Shield コンソールにアクセスするには、最低限の権限が必要です。これらの権限により、内の Shield リソースに関する詳細を一覧表示および表示できる必要があります 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、最低限のコンソール権限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。

コンソールにアクセスして使用できるユーザーは、 AWS コンソールにもアクセスできます。 AWS Shield 追加のアクセス許可は必要ありません。

自分の許可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添付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を作成する方法を示します。このポリシーには、コンソールで、またはまたは API を使用してこのアクションをプログラム的に実行するための権限が含まれています。 AWS CLI AWS

```
{
    "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": "*"
        }
    ]
}
```

Shield Advanced の保護機能に対する読み取りアクセスの許可

AWS Shield クロスアカウントのリソースアクセスは許可しますが、クロスアカウントのリソース保護を作成することはできません。リソースの保護は、それらのリソースを所有するアカウント内からのみ作成できます。

すべてのリソースの `shield:ListProtections` アクションの許可を付与するポリシーの例を次に示します。Shield は、一部の API アクションについて、リソース ARN (リソースレベルの許可と呼ばれる) を使用した特定のリソースの識別をサポートしていません。そのため、ワイルドカード文字 (*) を指定する必要があります。これは、アクション `ListProtections` を通して取得できるリソースへのアクセスのみを許可するものです。

```
{
    "Version": "2016-06-02",
    "Statement": [
        {
            "Sid": "ListProtections",
            "Effect": "Allow",
            "Action": [
                "shield:ListProtections"
            ],
            "Resource": "*"
        }
    ]
}
```

Shield、 CloudFront、への読み取り専用アクセスを許可する CloudWatch

以下のポリシーは、Shield および関連リソース (Amazon リソース、Amazon CloudFront CloudWatch メトリックスなど) への読み取り専用アクセスをユーザーに付与します。Shield の保護と攻撃の設定を表示したり、メトリックを監視したりする権限が必要なユーザーにとって便利です。 CloudWatchこれらのユーザーは、Shield リソースを作成、更新、または削除することはできません。

```
{
        "Version": "2012-10-17",
        "Statement": [
            {
                "Sid": "ProtectedResourcesReadAccess",
                "Effect": "Allow",
                "Action": [
                    "cloudfront:List*",
                    "elasticloadbalancing:List*",
                    "route53:List*",
                    "cloudfront:Describe*",
                    "elasticloadbalancing:Describe*",
                    "route53:Describe*",
                    "cloudwatch:Describe*",
                    "cloudwatch:Get*",
                    "cloudwatch:List*",
                    "cloudfront:GetDistribution*",
                    "globalaccelerator:ListAccelerators",
                    "globalaccelerator:DescribeAccelerator"
                ],
                "Resource": [
                    "arn:aws:elasticloadbalancing:*:*:*",
                    "arn:aws:cloudfront::*:*",
                    "arn:aws:route53:::hostedzone/*",
                    "arn:aws:cloudwatch:*:*:*:*",
                    "arn:aws:globalaccelerator::*:*"
                ]
            },
            {
                "Sid": "ShieldReadOnly",
                "Effect": "Allow",
                "Action": [
                    "shield:List*",
                    "shield:Describe*",
                    "shield:Get*"
                ],
                "Resource": "*"
            }
    ]
}
```

Shield へのフルアクセスを許可し CloudFront、 CloudWatch

以下のポリシーにより、ユーザーは Shield のすべての操作、 CloudFront ウェブディストリビューションでの任意の操作の実行、およびメトリクスとリクエストのサンプルの監視を行うことができます。 CloudWatchこれは、Shield の管理者であるユーザーにとって便利です。

```
{
        "Version": "2012-10-17",
        "Statement": [
            {
                "Sid": "ProtectedResourcesReadAccess",
                "Effect": "Allow",
                "Action": [
                    "cloudfront:List*",
                    "elasticloadbalancing:List*",
                    "route53:List*",
                    "cloudfront:Describe*",
                    "elasticloadbalancing:Describe*",
                    "route53:Describe*",
                    "cloudwatch:Describe*",
                    "cloudwatch:Get*",
                    "cloudwatch:List*",
                    "cloudfront:GetDistribution*",
                    "globalaccelerator:ListAccelerators",
                    "globalaccelerator:DescribeAccelerator"
                ],
                "Resource": [
                    "arn:aws:elasticloadbalancing:*:*:*",
                    "arn:aws:cloudfront::*:*",
                    "arn:aws:route53:::hostedzone/*",
                    "arn:aws:cloudwatch:*:*:*:*",
                    "arn:aws:globalaccelerator::*:*"
                ]
            },
            {
                "Sid": "ShieldFullAccess",
                "Effect": "Allow",
                "Action": [
                    "shield:*"
                ],
                "Resource": "*"
            }
      ]
```

```
}
```

管理者許可を持つユーザーに対しては多要素認証 (MFA) を設定することを強くお勧めします。詳細については、「IAM ユーザーガイド」の「AWSでのMulti-Factor Authentication (MFA) の使用」を参照してください。

AWS の管理ポリシー AWS Shield

AWS 管理ポリシーは、によって作成および管理されるスタンドアロンのポリシーです。 AWS AWS 管理ポリシーは、ユーザー、グループ、ロールにアクセス権限を割り当てることができるように、多くの一般的な使用事例にアクセス許可を与えるように設計されています。

AWS 管理ポリシーでは、 AWS すべての顧客が使用できるため、特定のユースケースでは最小権限のアクセス権限が付与されない場合があることに注意してください。ユースケースに固有の カスタマーマネージドポリシーを定義して、許可をさらに減らすことをお勧めします。

AWS 管理ポリシーで定義されている権限は変更できません。 AWS 管理ポリシーで定義されている権限を更新すると AWS 、その更新はポリシーがアタッチされているすべてのプリンシパル ID (ユーザー、グループ、ロール) に影響します。 AWS AWS 管理ポリシーが更新される可能性が最も高いのは、新しい API 操作が既存のサービスで開始されたときや、新しい API AWS のサービス 操作が使用可能になったときです。

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」を参照してください。

AWS 管理ポリシー: AWSShieldDRTAccessPolicy

AWS Shield Shield Response Team (SRT) にあなたに代わって行動する権限を付与すると、この管理ポリシーが適用されます。このポリシーでは、重大度の高いイベント発生時の DDoS 攻撃の軽減を支援するため、SRT AWS にアカウントへのアクセスを制限します。このポリシーにより、SRT AWS WAF はルールと Shield Advanced 保護を管理し、ログにアクセスできるようになります AWS WAF 。

SRT に代行操作を許可する方法については、「Shield Response Team (SRT) のためのアクセス権の設定」を参照してください。

このポリシーの詳細については、IAM AWSShieldDRTAccessPolicyコンソールのを参照してください。

AWS 管理ポリシー: AWSShieldServiceRolePolicy

Shield Advanced は、アプリケーションレイヤーの自動 DDoS 軽減を有効にする際に、この管理ポリシーを使用して、アカウントのリソース管理に必要な権限を設定します。このポリシーにより、Shield Advanced は、保護対象リソースに関連付けたウェブ ACL AWS WAF にルールとルールグループを作成して適用し、DDoS 攻撃に自動的に対応できるようになります。

IAM AWSShieldServiceRolePolicy エンティティにはアタッチできません。Shield はこのポリシーをサービス連動ロール `AWSServiceRoleForAWSShield` に添付し、Shield が代わりにアクションを実行できるようにします。

アプリケーションレイヤーの DDoS 自動緩和機能を有効にすると、Shield Advanced でこのポリシーの使用が可能になります。このポリシーの使用の詳細については、Shield Advanced アプリケーションレイヤー DDoS 自動緩和 を参照してください。

AWSServiceRoleForAWSShield このポリシーを使用するサービスにリンクされたロールについては、を参照してください。Shield Advanced のサービスにリンクされたロールの使用

このポリシーの詳細については、IAM AWSShieldServiceRolePolicyコンソールのを参照してください。

AWS 管理ポリシーへの更新をShield

このサービスが変更の追跡を開始して以降の Shield AWS の管理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通知については、ドキュメント履歴 の Shield 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。

ポリシー	変更点の説明	日付
`AWSShieldServiceRolePolicy` このポリシーにより、Shield AWS dはリソースにアクセスして管理し、ユーザーに代わってアプリケーションレイヤーのDDoS攻撃に自動的に対応することができます。	アプリケーションレイヤー DDoS 自動緩和機能に必要な許可を Shield Advanced に提供するため、このポリシーを追加しました。この機能については、「Shield Advanced アプリケーションレイヤー DDoS 自動緩和」を参照してください。	2021 年 12 月 1 日

ポリシー	変更点の説明	日付
IAM コンソールの詳細: AWSShieldServiceRolePolicy サービスにリンクされたロール AWSServiceRoleForAWSShield はこのポリシーを使用します。詳細については、「Shield Advanced 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。		
Shield は変更の追跡を開始しました	Shield AWS は管理ポリシーの変更の追跡を開始しました。	2021 年 3 月 3 日

AWS Shield ID とアクセスのトラブルシューティング

次の情報は、Shield と IAM の使用に伴って発生する可能性がある一般的な問題の診断や修復に役立ちます。

トピック

- Shield でアクションを実行する権限がない
- 私にはiam を実行する権限がありません:PassRole
- 自分以外のユーザーにもShield AWS アカウント リソースへのアクセスを許可したい

Shield でアクションを実行する権限がない

ある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、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。

次の例は、mateojackson という IAM ユーザーがコンソールを使用して架空の *my-example-widget* リソースに関する詳細を表示しようとしたとき、架空の shield:*GetWidget* アクセス許可がない場合に発生するエラーを示しています。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 shield:GetWidget on resource: my-example-widget
```

この場合、shield:*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 管理者に問い合わせてください。サインイン資格情報を提供した担当者が管理者です。

私にはiam を実行する権限がありません:PassRole

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 Shield にロールを渡せるようにする必要があります。

新しいサービスロールやサービスにリンクされたロールを作成する代わりに、 AWS のサービス 既存のロールをそのサービスに渡すことができるものもあります。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。

以下の例のエラーは、marymajor という IAM ユーザーがコンソールを使用して Shield で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権限が必要です。Mary には、ロールをサービスに渡す権限がありません。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

この場合、メアリー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、管理者に問い合わせてください。 AWS サインイン資格情報を提供した担当者が管理者です。

自分以外のユーザーにもShield AWS アカウント リソース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、以下を参照してください:

- Shield でこれらの特徴がサポートされるかどうかを確認するには、「と IAM の AWS Shield 連携方法」を参照してください。
- AWS アカウント 所有しているリソース全体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「AWS アカウント 所有する別の 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第三者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「AWS アカウント 第三者が所有するリソースへのアクセスの提供」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でのロールとリソースベースのポリシーの使用の違いの詳細については、「IAM ユーザーガイド」の「IAM ロールとリソースベースのポリシーとの相違点」を参照してください。

Shield Advanced のサービスにリンクされたロールの使用

AWS Shield Advanced AWS Identity and Access Management (IAM) サービスにリンクされたロールを使用する。サービスにリンクされたロールは、Shield Advanced に直接リンクされた一意のタイプの IAM ロールです。サービスにリンクされたロールは Shield Advanced によって事前定義されており、 AWS ユーザーに代わってサービスが他のサービスを呼び出すために必要なすべての権限が含まれます。

サービスにリンクされたロールを使用することで、必要な許可を手動で追加する必要がなくなるため、Shield Advanced の設定が簡単になります。Shield Advanced は、サービスにリンクされたロールの許可を定義します。特に定義されている場合を除き、Shield Advanced 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、信頼ポリシーと許可ポリシーが含まれており、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。

サービスにリンクされたロールは、まずその関連リソースを削除しなければ削除できません。これにより、リソースへの意図しないアクセスによる許可の削除が防止され、Shield Advanced リソースは保護されます。

サービスにリンクされたロールをサポートする他のサービスについては、「IAM と連携するAWS のサービス」を参照して、[Service-Linked Role] (サービスにリンクされたロール)列が [Yes] (はい) になっているサービスを探してください。そのサービスに関する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表示するには、リンクが設定されている [Yes] (はい) を選択します。

Shield Advanced のサービスにリンクされたロールの許可

Shield アドバンスドは、という名前のサービスにリンクされたロールを使用します。AWSServiceRoleForAWSShieldこのロールにより、Shield AWS Advancedはリソースにアクセスして管理し、ユーザーに代わってアプリケーションレイヤーのDDoS攻撃に自動的に対応することができます。この関数の詳細については、「Shield Advanced アプリケーションレイヤー DDoS 自動緩和」を参照してください。

AWSServiceRoleForAWSShield サービスにリンクされたロールは、以下のサービスを信頼してロールを引き受けます。

- `shield.amazonaws.com`

AWSShieldServiceRolePolicy という名前のロール権限ポリシーにより、Shield Advanced AWS はすべてのリソースで以下のアクションを実行できます。

- `wafv2:GetWebACL`
- `wafv2:UpdateWebACL`
- `wafv2:GetWebACLForResource`
- `wafv2:ListResourcesForWebACL`
- `cloudfront:ListDistributions`
- `cloudfront:GetDistribution`

AWS すべてのリソースに対してアクションが許可されている場合、ポリシーではと表示されます`"Resource": "*"`。これは、サービスにリンクされたロールが、 AWS アクションがサポートするすべてのリソースに対して、指定されたアクションをそれぞれ実行できるということだけです。例えば、アクション `wafv2:GetWebACL` は `wafv2` ウェブ ACL リソースでのみサポートされます。

Shield Advanced は、アプリケーションレイヤー保護機能を有効にしている保護されたリソースと、それらの保護されたリソースに関連付けられているウェブ ACL についてのみ、リソースレベルの API コールを実行します。

サービスにリンクされたロールの作成、編集、削除をIAM エンティティ (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。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許可」を参照してください。

Shield Advanced のサービスにリンクされたロールの作成

サービスリンクロールを手動で作成する必要はありません。、、または AWS API 内のリソースに対して自動アプリケーション層の DDoS 軽減を有効にすると AWS Management Console、Shield Advanced がサービスにリンクされたロールを自動的に作成します。 AWS CLI

このサービスリンクロールを削除した後で再度作成する必要が生じた場合は、同じ方法でアカウントにロールを再作成できます。リソースのためにアプリケーションレイヤー DDoS 自動緩和を有効にすると、Shield Advanced は、サービスにリンクされたロールを再作成します。

Shield Advanced のサービスにリンクされたロールの編集

Shield アドバンスドでは、 AWSServiceRoleForAWSShield サービスにリンクされたロールを編集することはできません。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 を使用したロールの説明の編集はできます。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。

Shield Advanced 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。そうすることで、積極的にモニタリングまたは保守されていない未使用のエンティティを排除できます。ただし、手動で削除する前に、サービスにリンクされたロールのリソースをクリーンアップする必要があります。

> **Note**
>
> リソースを削除する際に、Shield Advanced でロールが使用されている場合、削除は失敗することがあります。失敗した場合は、数分待ってからオペレーションを再試行してください。

が使用している Shield アドバンスドリソースを削除するには AWSServiceRoleForAWSShield

アプリケーションレイヤー DDoS 保護が設定されているすべてのリソースについて、アプリケーションレイヤー DDoS 自動緩和を無効にします。コンソールの手順については、「アプリケーションレイヤー DDoS 保護を設定する」を参照してください。

IAM を使用して、サービスにリンクされたロールを手動で削除するには

IAM コンソール、 AWS CLI、または AWS API を使用して、 AWSServiceRoleForAWSShield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。

Shield Advanced のサービスにリンクされたロールをサポートするリージョン

Shield Advanced では、このサービスが利用可能なすべての 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ています。詳細については、「Shield Advanced エンドポイントとクォータ」を参照してください。

Shield でのログ記録とモニタリング

監視は、SShield AWS とソリューションの信頼性、可用性、パフォーマンスを維持する上で重要な部分です。マルチポイント障害が発生した場合により簡単にデバッグできるように、 AWS ソリューションのあらゆる部分からモニタリングデータを収集する必要があります。 AWS には、Shield リソースを監視し、発生する可能性のあるイベントに対応するためのツールがいくつか用意されています。

Amazon CloudWatch アラーム

CloudWatch アラームを使用すると、指定した期間にわたって 1 つのメトリクスを監視できます。メトリックスが特定のしきい値を超えると、Amazon SNS CloudWatch AWS Auto Scaling トピックまたはポリシーに通知を送信します。詳細については、「Amazon によるモニタリング CloudWatch」を参照してください。

AWS CloudTrail ログ

CloudTrail Shield 内のユーザー、ロール、 AWS またはサービスが実行したアクションの記録を提供します。によって収集された情報を使用して CloudTrail、Shield に対して行われたリクエスト、リクエストが行われたIPアドレス、リクエストの実行者、リクエストの実行日時、その他の詳細を判断できます。詳細については、「での AWS CloudTrail API コールのログ記録」を参照してください。

Shield のコンプライアンス検証

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、コンプライアンスプログラムAWS のサービス による対象範囲内のコンプライアンスプログラムを参照し、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、「 でのレポートのダウンロード AWS Artifact」の」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。 では、コンプライアンスに役立つ以下のリソース AWS を提供しています。

- セキュリティとコンプライアンスのクイックスタートガイド – これらのデプロイガイドでは、アーキテクチャ上の考慮事項について説明し、セキュリティとコンプライアンスに重点を置いたベースライン環境 AWS を にデプロイする手順について説明します。
- アマゾン ウェブ サービスにおける HIPAA セキュリティとコンプライアンスのアーキテクチャ — このホワイトペーパーでは、企業が AWS を使用して HIPAA 対象アプリケーションを作成する方法について説明します。

  > **Note**
  >
  > すべての AWS のサービス が HIPAA の対象となるわけではありません。詳細については、「HIPAA 対応サービスのリファレンス」を参照してください。

- AWS コンプライアンスリソース – このワークブックとガイドのコレクションは、お客様の業界や地域に適用される場合があります。
- AWS カスタマーコンプライアンスガイド — コンプライアンスの観点から責任共有モデルを理解します。このガイドでは、ガイダンスを保護し AWS のサービス 、複数のフレームワーク (米国国立標準技術研究所 (NIST)、Payment Card Industry Security Standards Council (PCI)、国際標準化機構 (ISO) を含む) のセキュリティコントロールにマッピングするためのベストプラクティスをまとめています。
- 「 デベロッパーガイド」の「 ルールによるリソースの評価」 – この AWS Config サービスは、リソース設定が社内プラクティス、業界ガイドライン、および規制にどの程度準拠しているかを評価します。 AWS Config
- AWS Security Hub – これにより AWS のサービス 、 内のセキュリティ状態を包括的に確認できます AWS。Security Hub では、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対するコンプライアンスをチェックします。サポートされているサービスとコントロールのリストについては、「Security Hub のコントロールリファレンス」を参照してください。

- Amazon GuardDuty — これにより AWS アカウント、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環境を監視することで、、、ワークロード、コンテナ、データに対する潜在的な脅威 AWS のサービス を検出します。 GuardDuty は、特定のコンプライアンスフレームワークで義務付けられている侵入検知要件を満たすことで、PCI DSS などのさまざまなコンプライアンス要件への対応に役立ちます。
- AWS Audit Manager – これにより AWS のサービス 、 AWS 使用状況を継続的に監査し、リスクの管理方法と規制や業界標準への準拠を簡素化できます。

Shield の回復力

AWS AWS リージョン グローバルインフラストラクチャはアベイラビリティーゾーンを中心に構築されています。 AWS リージョン 物理的に分離された複数のアベイラビリティーゾーンを提供し、低レイテンシー、高スループット、冗長性の高いネットワークで接続します。アベイラビリティーゾーンでは、アベイラビリティーゾーン間で中断せずに、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、耐障害性、およびスケーラビリティが優れています。

AWS リージョン およびアベイラビリティーゾーンの詳細については、「グローバルインフラストラクチャ」を参照してください。AWS

AWS Shield内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として、 AWS Shield AWS グローバルなネットワークセキュリティによって保護されています。 AWS AWS セキュリティサービスとインフラストラクチャの保護方法については、「AWS クラウドセキュリティ」を参照してください。 AWS 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設計するには、「Security Pillar AWS Well-Architected Framework におけるインフラストラクチャ保護」を参照してください。

AWS 公開されている API 呼び出しを使用して、ネットワーク経由で Shield にアクセスします。クライアントは以下をサポートする必要があります:

- Transport Layer Security (TLS)。TLS 1.2、できれば TLS 1.3 が必要です。

- DHE (Ephemeral Diffie-Hellman) や ECDHE (Elliptic Curve Ephemeral Diffie-Hellman) などの Perfect Forward Secrecy (PFS) を使用した暗号スイート。これらのモードは、Java 7 以降など、ほとんどの最新システムでサポートされています。

また、リクエストには、アクセスキー ID と、IAM プリンシパルに関連付けられているシークレットアクセスキーを使用して署名する必要があります。または、AWS Security Token ServiceAWS STS を使用して、一時的なセキュリティ認証情報を生成し、リクエストに署名することもできます。

AWS Shield Advanced クォータ

AWS Shield Advanced リージョンごとのエンティティ数にはデフォルトのクォータが設定されています。このクォータの引き上げをリクエストできます。

リソース	デフォルトのクォータ
保護の対象となる各リソースタイプの、 AWS Shield Advanced アカウントあたりの保護対象リソースの最大数。	1,000
アカウントごとの保護グループの最大数。	100
保護グループに具体的に含めることができる個別の保護されたリソースの最大数。API では、これは保護グループ `Pattern` を `ARBITRARY` に設定するときに指定する `Members` に適用されます。コンソールでは、これは保護グループのために選択するリソースに適用されます [Choose from protected resources] (保護されたリソースから選択)	1,000

AWS Firewall Manager

AWS Firewall Manager は、、、 AWS WAF AWS Shield Advanced Amazon VPC セキュリティグループとネットワーク ACLs、Amazon Route 53 Resolver DNS Firewall など AWS Network Firewall、さまざまな保護のために複数のアカウントとリソースにわたる管理およびメンテナンスタスクを簡素化します。Firewall Manager を使用すると、保護を 1 回設定するだけで、アカウントとリソースに (追加する新しいアカウントとリソースにも) その保護が自動的に適用されます。

Firewall Manager には、次のような利点があります。

- アカウント間でリソースを保護するのに役立ちます
- すべての Amazon CloudFront ディストリビューションなど、特定のタイプのすべてのリソースを保護するのに役立ちます。
- 特定のタグですべてのリソースを保護するのに役立ちます
- アカウントに追加されたリソースへの保護を自動的に追加します
- AWS Organizations 組織内のすべてのメンバーアカウントを にサブスクライブし AWS Shield Advanced、組織に参加する新しい範囲内のアカウントを自動的にサブスクライブできます。
- AWS Organizations 組織内のすべてのメンバーアカウントまたはアカウントの特定のサブセットにセキュリティグループルールを適用し、組織に参加する新しい範囲内アカウントにルールを自動的に適用できます。
- 独自のルールを使用するか、 から マネージドルールを購入できます。 AWS Marketplace

Firewall Manager は、少数の特定のアカウントやリソースではなく、組織全体を保護したい場合や、保護したい新しいリソースを頻繁に追加する場合に、特に有効です。Firewall Manager では、組織全体の DDoS 攻撃を一元的にモニタリングすることもできます。

トピック

- AWS Firewall Manager 検出結果
- AWS Firewall Manager サービスの使用におけるセキュリティ
- AWS Firewall Manager クォータ

AWS Firewall Manager 価格設定

AWS Firewall Manager で発生する料金は、 AWS WAF やなどの基盤となるサービスに対するものです。 AWS Config詳細については、「AWS Firewall Manager の料金」を参照してください。

AWS Firewall Manager 前提条件

このトピックでは、管理の準備をする方法を説明します。 AWS Firewall Manager AWS Organizationsで組織のすべてのための Firewall Manager セキュリティポリシーを管理するには、1 つの Firewall Manager 管理者アカウントを使用します。特に記載がない限り、Firewall Manager 管理者として使用するアカウントを使用して、前提条件となるステップを実行します。

Firewall Manager を初めて使用する前に、次のステップを順番に実行してください。

トピック

- ステップ 1: 参加と設定 AWS Organizations
- ステップ 2: AWS Firewall Manager 既定の管理者アカウントを作成する
- ステップ 3: を有効にする AWS Config
- ステップ 4: サードパーティのポリシーについては、 AWS Marketplace でサブスクライブし、サードパーティーの設定を行う
- ステップ 5: Network Firewall ポリシーと DNS Firewall ポリシー用にリソース共有を有効にする
- ステップ 6: AWS Firewall Manager デフォルトで無効になっているリージョンで使用するには

ステップ 1: 参加と設定 AWS Organizations

Firewall Manager を使用するには、アカウントが Firewall Manager ポリシーを使用する AWS Organizations のサービスの組織のメンバーである必要があります。

> **Note**
>
> Organizations の詳細については、「AWS Organizations ユーザーガイド」を参照してください。

AWS Organizations 必要なメンバーシップと構成を確立するには

1. Organizations で組織の Firewall Manager 管理者として使用するアカウントを選択します。
2. 選択したアカウントがまだ組織のメンバーでない場合は、そのアカウントに参加させてください。「AWS アカウント 組織へのメンバーの招待」のガイダンスに従ってください。
3. AWS Organizations には、一括請求機能と全機能の 2 つの機能セットがあります。Firewall Manager を使用するには、組織ですべての機能を有効にする必要があります。組織が一括請求 (コンソリデーティッドビリング) のためにのみ設定されている場合は、「組織内のすべての機能の有効化」のガイダンスに従ってください。

ステップ 2: AWS Firewall Manager 既定の管理者アカウントを作成する

この手順では、前のステップで選択および設定したアカウントと組織を使用します。

Firewall Manager のデフォルト管理者アカウントを作成できるのは、組織の管理アカウントのみです。デフォルトの管理者アカウントは、最初に作成する管理者アカウントです。デフォルトの管理者アカウントはサードパーティのファイアウォールを管理でき、完全な管理権限範囲を持ちます。デフォルトの管理者アカウントを設定すると、Firewall Manager は自動的にそれをFirewall Manager AWS Organizations の委任管理者として設定します。これにより、Firewall Manager は、対象の組織内の組織単位 (OU) に関する情報にアクセスできます。OU を使用して、Firewall Manager ポリシーの範囲を指定できます。ポリシーの範囲の設定の詳細については、「AWS Firewall Manager ポリシーの作成」の個々のポリシータイプに関するガイダンスを参照してください。Organizations と管理アカウントの詳細については、「AWS 組織内のアカウントの管理」を参照してください。

組織の管理アカウントに必須な設定

組織を Firewall Manager にオンボーディングし、さらにデフォルト管理者を作成するためには、組織の管理アカウントに以下の設定が必要です。

- Firewall Manager ポリシーを適用する組織のメンバーである必要があります。 AWS Organizations

デフォルトの管理者アカウントを設定するには

1. AWS Management Console AWS Organizations 既存の管理アカウントを使用してFirewall Manager にサインインします。
2. Firewall Manager コンソール (https://console.aws.amazon.com/wafv2/fmsv2) を開きます。
3. ナビゲーションペインで [設定] を選択します。
4. Firewall Manager 管理者として使用することを選択したアカウントのアカウント ID を入力します。 AWS

   > **Note**
   >
   > デフォルトの管理者は完全な管理権限範囲を保持します。完全な管理権限範囲とは、このアカウントが組織内のすべてのアカウントと組織単位 (OU) にポリシーを適用でき、すべてのリージョンでアクションの実行が可能で、また、Firewall Manager のすべてのポリシータイプを管理できることを意味します。

5. [管理者アカウントを作成] を選択してアカウントを作成します。

Firewall Manager 管理者アカウントの管理の詳細については、「AWS Firewall Manager 管理者との連携」を参照してください。

ステップ 3: を有効にする AWS Config

Firewall Manager を使用するには、 AWS Configを有効にする必要があります。

> **Note**
>
> AWS Config 料金に従って、 AWS Config 設定に対して料金が発生します。詳細については、「 の開始方法 AWS Config」を参照してください。

> **Note**
>
> Firewall Manager がポリシーコンプライアンスをモニタリングするには、保護されたリソースの設定変更を継続的に記録 AWS Config する必要があります。 AWS Config 設定では、記録頻度をデフォルト設定である連続 に設定する必要があります。

Firewall Manager AWS Config で を有効にするには

1. Firewall Manager 管理者アカウントを含む、 AWS Organizations メンバーアカウント AWS Config ごとに を有効にします。詳細については、「 の開始方法 AWS Config」を参照してください。
2. 保護するリソース AWS リージョン を含む各 AWS Config に対して を有効にします。 AWS Config を手動で有効にすることも、サンプル AWS CloudFormation テンプレート AWS ConfigAWS CloudFormation StackSets でテンプレート「有効化」を使用することもできます。

   すべてのリソース AWS Config に対して を有効にしない場合は、使用する Firewall Manager ポリシーのタイプに従って以下を有効にする必要があります。

   - WAF ポリシー – リソースタイプ CloudFront Distribution、Application Load Balancer (リストから ElasticLoadBalancingV2 を選択)、API Gateway、WAF WebACL、WAF Regional WebACL、および WAFv2 WebACL に対して Config を有効にします。 AWS Config で CloudFront ディストリビューションを保護するには、米国東部 (バージニア北部) リージョンにいる必要があります。他のリージョンでは、オプション CloudFront として はありません。
   - Shield ポリシー – Shield Protection、 ShieldRegional Protection、Application Load Balancer、EC2 EIP、WAF WebACL、WAF Regional WebACL、および WAFv2 WebACL のリソースタイプに対して Config を有効にします。
   - セキュリティグループポリシー – リソースタイプ EC2 SecurityGroup、EC2 インスタンス、および EC2 の Config EC2 を有効にしますNetworkInterface。
   - ネットワーク ACL ポリシー – リソースタイプ Amazon EC2 サブネットと Amazon EC2 ネットワーク ACL の Config を有効にします。
   - Network Firewall ポリシー – リソースタイプ NetworkFirewall FirewallPolicy、、NetworkFirewallEC2 VPCRuleGroup、EC2、EC2 InternetGatewayEC2 RouteTable、および EC2 サブネットの Config を有効にします。
   - DNS Firewall ポリシー - リソースタイプ EC2 VPC の Config を有効にします。
   - サードパーティーのファイアウォールポリシー – リソースタイプ Amazon EC2 VPC、Amazon EC2、Amazon EC2 InternetGateway、Amazon EC2 RouteTable サブネット、および Amazon EC2Amazon EC2 VPCEndpoint の Config を有効にします。

   > **Note**
   >
   > カスタム IAM ロールを使用するように AWS Config レコーダーを設定する場合は、IAM ポリシーに Firewall Manager ポリシーに必要なリソースタイプを記録するための適切な

アクセス許可があることを確認する必要があります。適切なアクセス許可がないと、必要なリソースが記録されず、Firewall Manager がリソースを適切に保護できなくなる可能性があります。Firewall Manager では、これらのアクセス許可の設定ミスを可視化することはできません。で IAM を使用する方法については AWS Config、「 の IAM AWS Config」を参照してください。

ステップ 4: サードパーティのポリシーについては、 AWS Marketplace でサブスクライブし、サードパーティーの設定を行う

Firewall Manager のサードパーティ製ファイアウォールポリシーを開始するには、次の前提条件を満たします。

Fortigate Cloud Native Firewall (CNF) as a Service ポリシーの前提条件

Fortigate CNF を Firewall Manager で使用するには

1. Marketplace でFortigateクラウドネイティブファイアウォール（CNF）をサービスとして購読してください。 AWS
2. まず、Fortigate CNF 製品ポータルにテナントを登録します。次に、Fortigate CNF 製品ポータルのテナントの下に Firewall Manager の管理者アカウントを追加します。詳細については、「Fortigate CNF documentation」(Fortigate CNF ドキュメント) を参照してください。

Fortigate CNF ポリシーの操作については、「Fortigate Cloud Native Firewall (CNF) as a Service ポリシー」を参照してください。

Palo Alto Networks Cloud Next Generation Firewall ポリシーの要件

Palo Alto Networks Cloud NGFW を Firewall Manager で使用するには

1. Marketplace でパロアルトネットワークスのクラウド次世代ファイアウォール従量課金制サービスを購読してください。 AWS
2. 導入ガイドのパロアルトネットワークスクラウド次世代ファイアウォールに記載されているパロアルトネットワークスクラウドNGFWの導入手順を、 AWS パロアルトネットワークスクラウド次世代ファイアウォール導入ガイドのトピックに従って実行してください。 AWS Firewall Manager AWS

Palo Alto Networks Cloud NGFW ポリシーの操作については、「Palo Alto Networks Cloud NGFW ポリシー」を参照してください。

ステップ 5: Network Firewall ポリシーと DNS Firewall ポリシー用にリソース共有を有効にする

Firewall Manager Network Firewall と DNS ファイアウォールポリシーを管理するには、 AWS Organizations in との共有を有効にする必要があります AWS Resource Access Manager。これにより、Firewall Manager は、これらのポリシータイプを作成するときに、アカウント全体で保護をデプロイできます。

AWS Organizations in との共有を有効にするには AWS Resource Access Manager

- 「AWS Resource Access Manager ユーザーガイド」の「AWS Organizations内でリソース共有を有効にする」のガイダンスに従ってください。

リソース共有で問題が発生した場合は、「Network Firewall ポリシーと DNS Firewall ポリシーのリソース共有」のガイダンスを参照してください。

ステップ 6: AWS Firewall Manager デフォルトで無効になっているリージョンで使用するには

デフォルトで無効になっているリージョンで Firewall Manager を使用するには、 AWS 組織の管理アカウントと Firewall Manager のデフォルト管理者アカウントの両方でリージョンを有効にする必要があります。デフォルトで無効になっているリージョンとそれを有効にする方法については、「AWS 全般のリファレンス」の「AWS リージョンの管理」を参照してください。

無効にしたリージョンを有効にするには

- Organizations の管理アカウントと Firewall Manager のデフォルトの管理者アカウントの両方について、「AWS 全般のリファレンス」の「リージョンを有効にする」のガイダンスに従います。

次のステップを実行した後、Firewall Manager を設定してリソースの保護を開始できます。詳細については、「AWS Firewall ManagerAWS WAF ポリシー入門」を参照してください。

AWS Firewall Manager 管理者との連携

AWS Firewall Manager を使用すると、組織のファイアウォールリソースを管理できる 1 人または複数の管理者を配置できます。組織内で複数の Firewall Manager 管理者を配置する場合は、各管理者の管理権限範囲の条件を適用することで、それぞれが管理できるリソースを定義できます。これにより、組織内のさまざまな管理者に役割を割り当てる上での柔軟性が提供され、最小権限の原則を維持しやすくなります。例えば、組織の一連の組織単位 (OU) を 1 人の管理者に管理させ、別の管理者には、特定の Firewall Manager ポリシータイプのみの管理を任せるといったことが可能です。Organizations と管理アカウントの詳細については、「AWS 組織内のアカウントの管理」を参照してください。

組織ごとに設定できる管理者の最大数については、「AWS Firewall Manager クォータ」を参照してください。

Firewall Manager 管理者の使用開始

Firewall Manager 管理者の使用を開始する前に、「AWS Firewall Manager 前提条件」に記載されている前提条件を完了する必要があります。前提条件として、 AWS Organizations 組織をファイアウォールマネージャーに登録し、ファイアウォールマネージャーのデフォルト管理者アカウントを作成します。デフォルトの管理者アカウントは完全な管理権限範囲を持ち、サードパーティのファイアウォールを管理することができます。

管理権限範囲

管理権限範囲により、Firewall Manager 管理者が管理できるリソースが定義されます。 AWS Organizations 管理アカウントが組織をFirewall Manager に登録すると、管理アカウントは管理範囲が異なるFirewall Manager 管理者を追加で作成できます。 AWS Organizations 管理アカウントは、管理者にすべての管理範囲を付与することも、制限付きで管理範囲を付与することもできます。完全な管理権限範囲を持つ管理者は、前述のすべてのリソースタイプに対し完全なアクセスが可能です。制限付きの管理権限範囲とは、前述のリソースに関して部分的な管理者権限が付与されることを指します。各管理者には、それぞれの役割の遂行に必要な権限のみを付与することをお勧めします。以下の管理権限範囲の条件は、任意に組み合わせて管理者に適用できます。

- 管理者がポリシーを適用できる組織内のアカウントまたは OU。
- 管理者がアクションを実行できるリージョン。
- 管理者が管理できる Firewall Manager のポリシータイプ。

管理者のロール

Firewall Manager には、デフォルト管理者 とFirewall Manager 管理者の 2 種類の管理者ロールがあります。

- デフォルト管理者 – 組織の管理アカウントは、組織を Firewall Manager にオンボーディングする際に、AWS Firewall Manager 前提条件 を完了する段階で、Firewall Manager のデフォルト管理者アカウントを作成します。デフォルトの管理者はサードパーティのファイアウォールを管理でき、完全な管理権限範囲を持ちます。ただし、他に複数の管理者を配置する場合には、他の管理者と同じレベルの権限にもできます。
- Firewall Manager 管理者 – Firewall Manager 管理者は、 AWS Organizations の管理アカウントが管理権限範囲の設定で指定したリソースを管理できます。組織ごとに配置できる管理者の最大数については、「AWS Firewall Manager クォータ」を参照してください。Firewall Manager 管理者アカウントを作成すると、サービスはアカウントが既に組織内の Firewall Manager の委任管理者であるかどうかを確認します。 AWS Organizations そうでない場合、Firewall Manager は Organizations を呼び出して、対象のアカウントに Firewall Manager の管理者を委任します。Organizations での委任された管理者の詳細については、「AWS Organizations ユーザーガイド」の「AWS Organizations の用語と概念」を参照してください。

既存の管理者

現在 Firewall Manager をご利用のお客様で、すでに管理者を設定している場合には、この既存の管理者が Firewall Manager のデフォルト管理者になります。既存のフローには、特に影響を与えません。さらに管理者を追加したい場合は、この章の手順に従って設定します。

Firewall Manager の管理者アカウントの作成、更新、および取り消し

以下のトピックでは、Firewall Manager の管理者アカウントを作成、更新、および取り消すための手順を説明します。Firewall Manager 管理者アカウントの作成および更新は、組織の管理アカウントのみが実行できます。個々の Firewall Manager の管理者のみが、その Firewall Manager の管理者アカウントを取り消すことができます。

Firewall Manager 管理者アカウントの作成。

次に、Firewall Manager コンソールを使用して、Firewall Manager 管理者を作成するための手順を説明します

Firewall Manager 管理者アカウントを作成するには

1. AWS Management Console AWS Organizations 既存の管理アカウントを使用してFirewall Manager にサインインします。

2. Firewall Manager コンソール (https://console.aws.amazon.com/wafv2/fmsv2) を開きます。
3. ナビゲーションペインで [設定] を選択します。
4. [管理者アカウントを作成] を選択します。
5. [詳細] ペインの [AWS アカウント ID] に、Firewall Manager 管理者として追加するメンバーアカウントの AWS ID を入力します。
6. [管理権限範囲] では、以下のいずれかのオプションを選択します。

   - [完全] – これにより、管理者は組織内のすべてのアカウントと組織単位 (OU) にポリシーを適用したり、すべてのリージョンでアクションを実行したりできます。また、サードパーティのファイアウォールを除くすべてのFirewall Manager ポリシータイプを適用できます。デフォルトの管理者のみが、サードパーティのファイアウォールを作成および管理できます。このレベルのアクセス許可を管理者に付与する場合は注意が必要です。最小特権の原則を考慮した場合、管理者にはその役割の遂行に必要なアクセス許可のみを付与することが推奨されます。
   - 制限付き – 制限付きの管理権限の範囲を適用する場合は、[管理権限範囲を設定] で、そのアカウントで管理が可能なアカウントと組織単位、リージョン、ポリシータイプを設定します。

     [アカウントと組織単位] では、以下のようにオプションを選択します。

     - 組織内のすべてのアカウントまたは組織単位にポリシーを適用する場合は、「 AWS 自分の組織のアカウントをすべて含める」を選択します。
     - AWS Organizations 特定のアカウントまたは特定の組織単位 (OU) に属するアカウントにのみポリシーを適用する場合は、[指定したアカウントと組織単位のみを含める] を選択し、含めるアカウントと OU を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。
     - AWS Organizations 特定のアカウントまたは組織単位 (OU) 以外のすべてにポリシーを適用する場合は、[指定したアカウントと組織単位を除外し、その他すべてを含める] を選択してから、除外するアカウントと OU を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。

     [リージョン] で、以下のようにオプションを選択します。

     - 使用可能なすべてのリージョンで管理者がアクションを実行できるようにするには、[すべてのリージョンを含める] を選択します。
     - 管理者に特定のリージョンでのみアクションを実行させたい場合は、[指定したリージョンのみを含める] を選択した後、含めるリージョンを指定します。

> **Note**
>
> デフォルトで無効になっているリージョンを含めるには、 AWS Organizations 組織管理アカウントとデフォルト管理アカウントの両方でリージョンを有効にする必要があります。アカウントのリージョンを有効にする方法については、「Amazon Web Services 全般のリファレンス」の「リージョンを有効にする」を参照してください。

[ポリシータイプ] で、以下のようにオプションを選択します。

- すべてのポリシータイプの管理を管理者に許可するには、[すべてのポリシータイプを含める] を選択します。
- 管理者に特定のポリシータイプのみを管理させたい場合は、[指定したポリシータイプのみを含める] を選択し、対象のポリシータイプを指定します。

7. [管理者アカウントを作成] を選択し、管理者アカウントを作成します。作成時に、Firewall Manager は、 AWS Organizations 管理者がすでに組織の委任管理者であるかどうかを確認するための呼び出しを行います。そうでない場合、Firewall Manager は対象のアカウントを委任された管理者として指定します。Organizations の委任された管理者の詳細については、「AWS Organizations ユーザーガイド」の「AWS Organizations の用語と概念」を参照してください。

[制限付き] の管理権限範囲を適用している場合、Firewall Manager は、設定に対応させながら新しいリソースを自動的に評価します。例えば、特定のアカウントのみを含めると、Firewall Manager は新しいアカウントにポリシーを適用しません。別の例として OU を使用しており、その OU またはその子である任意の OU にアカウントを追加した場合、Firewall Manager は、そのアカウントを対応した管理権限範囲に自動的に含めます。

Firewall Manager 管理者アカウントの更新

以下で、Firewall Manager コンソールを使用してFirewall Manager 管理者アカウントを更新するための手順について説明します。

> **Note**
>
> デフォルトで無効になっているリージョンを含むように管理者の範囲を更新するには、AWS Organizations 組織管理アカウントとデフォルト管理アカウントの両方でリージョンを有効にする必要があります。アカウントのリージョンを有効にする方法については、

「Amazon Web Services 全般のリファレンス」の「リージョンを有効にする」を参照してください。

管理者アカウントを更新するには (コンソール)

1. AWS Management Console AWS Organizations 既存の管理アカウントを使用してFirewall Manager にサインインします。
2. Firewall Manager コンソール (https://console.aws.amazon.com/wafv2/fmsv2) を開きます。
3. ナビゲーションペインで [設定] を選択します。
4. [Firewall Manager 管理者テーブル] で、更新する対象のアカウントを選択します。
5. 管理者アカウントの詳細を変更するには、[編集] を選択します。[アカウント ID] を変更することはできません。
6. [保存] を選択して変更を保存します。

管理者アカウントの取り消し

以下で、Firewall Manager 管理者アカウントを取り消すための手順について説明します。デフォルト管理者がアカウントを取り消せるようになるためには、組織内のすべての Firewall Manager 管理者アカウントが、まず自分のアカウントを取り消す必要があります。管理者アカウントを取り消すには、以下の手順に従います

管理者アカウントを取り消すには (コンソール)

1. Firewall Manager AWS Management Console 管理者アカウントを使用してにサインインし、でFirewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。
2. ナビゲーションペインで [設定] を選択します。
3. [管理者アカウント] ペインで、「管理者アカウントを取り消す] を選択して、アカウントを取り消します。

> **Important**
>
> 管理者アカウントから管理者権限を取り消すと、そのアカウントによって作成されたすべての Firewall Manager ポリシーが削除されます。

デフォルトの管理者アカウントの変更

組織内では、1 つのアカウントのみをデフォルトの Firewall Manager 管理者アカウントとして指定できます。デフォルトの管理者アカウントは、先入れ後出しの原則に従います。異なるデフォルト管理者アカウントを指定する場合には、まず個々の管理者アカウントが自分のアカウントを取り消す必要があります。その後、既存のデフォルト管理者は自分のアカウントを取り消すことが可能になります。これにより、組織は Firewall Manager からオフボーディングされます。管理者が自分のアカウントを取り消すと、そのアカウントによって作成されたすべての Firewall Manager ポリシーが削除されます。新しいデフォルト管理者アカウントを指定するには、管理アカウントで Firewall Manager にサインインし、新しい管理者アカウントを指定する必要があります。 AWS Organizations 組織のデフォルトの管理者アカウントを変更するには、次の手順を実行します。

デフォルトの管理者アカウントを変更するには

1. AWS Management Console AWS Organizations 既存の管理アカウントを使用してFirewall Manager にサインインします。
2. Firewall Manager コンソール (https://console.aws.amazon.com/wafv2/fmsv2) を開きます。
3. ナビゲーションペインで [設定] を選択します。
4. Firewall Manager 管理者として使用するように選択したアカウントの ID を入力します。

   > **Note**
   >
   > このアカウントには、組織内のすべてのアカウントで Firewall Manager ポリシーを作成および管理するための許可が付与されます。

5. [管理者アカウントを作成] を選択します。
6. Firewall Manager 管理者として使用することを選択したアカウントの AWS ID を入力します。

   > **Note**
   >
   > このアカウントには完全な管理権限範囲が与えられます。完全な管理権限範囲とは、このアカウントが組織内のすべてのアカウントと組織単位 (OU) にポリシーを適用でき、すべてのリージョンでアクションの実行が可能で、また、Firewall Manager のすべてのポリシータイプを管理できることを意味します。

7. [管理者アカウントを作成] を選択して、デフォルトの管理者アカウントを作成します。

管理者アカウントに対する変更の却下

管理者アカウントに加えられた変更の一部が却下され、管理者アカウントが維持できないことがあります。

このセクションでは、管理者アカウントを失格にする可能性のある変更と、Firewall Manager がこれらの変更を処理する方法について説明します AWS 。

の組織からアカウントが削除されました AWS Organizations

AWS Firewall Manager 管理者アカウントがで組織から削除されると AWS Organizations、その管理者アカウントは組織のポリシーを管理できなくなります。Firewall Manager は、次のいずれかのアクションを実行します。

- ポリシーのないアカウント - Firewall Manager 管理者アカウントに Firewall Manager ポリシーがない場合、Firewall Manager は管理者アカウントを取り消します。
- Firewall Manager ポリシーのあるアカウント — Firewall Manager 管理者アカウントにFirewall Manager agerポリシーが設定されている場合、Firewall Manager は、 AWS セールスアカウント担当者の協力を得て、状況を通知し、実行できるオプションを提供するメールを送信します。

アカウントが閉鎖された

AWS Firewall Manager 管理者用に使用しているアカウントを閉鎖する場合 AWS 、Firewall Manager は次のように閉鎖を処理します。

- AWS Firewall Manager からのアカウントの管理者アクセスを取り消し、Firewall Manager は管理者アカウントによって管理されていたすべてのポリシーを無効にします。それらのポリシーによって提供された保護は、組織全体で停止されます。
- AWS 管理者アカウントの閉鎖の発効日から 90 日間、アカウントの Firewall Manager ポリシーデータを保持します。この 90 日間の期間中、閉鎖したアカウントを再開できます。
  - 90 日以内に閉鎖されたアカウントを再度開くと、そのアカウントを Firewall Manager AWS 管理者として再割り当てし、そのアカウントの Firewall Manager ポリシーデータを回復します。
  - それ以外の場合は、90 日の期間の終了時に、アカウントのすべての Firewall Manager AWS ポリシーデータが完全に削除されます。

AWS Firewall Manager ポリシーの開始方法

AWS Firewall Manager を使用して、さまざまなタイプのセキュリティポリシーを有効にできます。セットアップのステップはそれぞれ少し異なります。

トピック

AWS Firewall ManagerAWS WAF ポリシー入門

AWS Firewall Manager AWS WAF を使用して組織全体でルールを有効にするには、次の手順を順番に実行してください。

トピック

ステップ 1: 前提条件を満たす

AWS Firewall Managerのアカウントを準備するには、いくつかの必須のステップがあります。それらのステップは、AWS Firewall Manager 前提条件 で説明されています。「ステップ 2: AWS WAF ポリシーを作成して適用する」に進む前に、すべての前提条件を満たしてください。

ステップ 2: AWS WAF ポリシーを作成して適用する

Firewall Manager AWS WAF ポリシーには、リソースに適用するルールグループが含まれます。Firewall Manager は、ポリシーを適用する各アカウントに Firewall Managerウェブ ACL を作成します。各アカウントマネージャーは、生成されたウェブ ACL に、お客様がここで定義するルールグループに加えて、ルールとルールグループを追加できます。Firewall Manager AWS WAF ポリシーの詳細については、「」を参照してくださいAWS WAF ポリシー。

Firewall Manager AWS WAF ポリシーを作成するには (コンソール)

Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

1. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
2. [Create policy] (ポリシーの作成) を選択します。
3. [Policy type] (ポリシータイプ) では [AWS WAF] を選択します。
4. リージョン で、 を選択します AWS リージョン。Amazon CloudFront ディストリビューションを保護するには、グローバル を選択します。

   複数のリージョン (CloudFront ディストリビューション以外) のリソースを保護するには、リージョンごとに個別の Firewall Manager ポリシーを作成する必要があります。
5. [Next] (次へ) を選択します。
6. [Policy name] (ポリシー名) で、わかりやすい名前を入力します。Firewall Manager は、管理するウェブ ACL の名前にポリシー名を含めます。ウェブ ACL 名の後に、`FMManagedWebACLV2-`、ここに入力するポリシー名、`-`、およびウェブ ACL 作成タイムスタンプ (UTC ミリ秒) が続きます。例えば、`FMManagedWebACLV2-MyWAFPolicyName-1621880374078` です。

   > **Important**
   >
   > ウェブ ACL 名は作成後に変更できません。ポリシー名を更新しても、Firewall Manager は関連するウェブ ACL 名を更新しません。Firewall Manager で別の名前のウェブ ACL を作成できるようにするには、新しいポリシーを作成する必要があります。
7. [Policy rules (ポリシールール)] の [First rule groups (最初のルールグループ)] で、[Add rule groups (ルールグループの追加)] を選択します。[AWS managed rule groups] (マネージドルー

ルグループ) を展開します。[Core rule set] (コアルールセット) で、[Add to web ACL] (ウェブ ACL に追加) に切り替えます。[AWS の既知の不正な入力] で、[ウェブ ACL に追加] に切り替えます。[Add rules] (ルールの追加) を選択します。

[Last rule groups] (最後のルールグループ) で、[Add rule groups] (ルールグループの追加) を選択します。[AWS マネージドルールグループ] を展開し、[Amazon IP 評価リスト] で [ウェブ ACL に追加] に切り替えます。[Add rules] (ルールの追加) を選択します。

「最初のルールグループ」で、「コアルールセット」を選択し、「下に移動」を選択します。は、ウェブリクエストをAWS 既知の不正な入力ルールグループに対して AWS WAF 評価してから、 Core ルールセット に対して評価します。

AWS WAF コンソールを使用して、必要に応じて独自の AWS WAF ルールグループを作成することもできます。作成したルールグループは、[Describe policy : Add rule groups page] (ポリシーを記述: ルールグループの追加ページ) の [Your rule groups] (ルールグループ) の下に表示されます。

Firewall Manager で管理する最初と最後の AWS WAF ルールグループの名前はPOSTFMManaged-、それぞれ PREFMManaged-または で始まり、その後に Firewall Manager ポリシー名とルールグループ作成タイムスタンプが UTC ミリ秒単位で続きます。例えば、PREFMManaged-MyWAFPolicyName-1621880555123 です。

8. ウェブ ACL のデフォルトアクションは [Allow] (許可) のままにします。
9. [Policy action] (ポリシーアクション) はデフォルトのままにして、準拠していないリソースが自動的に修復されないようにします。このオプションは後で変更できます。
10. [Next] (次へ) を選択します。
11. [Policy scope] (ポリシーの範囲) で、ポリシーを適用するリソースを識別するアカウント、リソースタイプ、タグを設定します。このチュートリアルでは、[AWS アカウント] と [Resources] (リソース) の設定はそのままにし、1 つ以上のリソースタイプを選択します。
12. リソース では、指定したタグを持つリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

    複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。

13. [次へ] をクリックします。
14. ポリシータグ には、Firewall Manager ポリシーリソースに追加する識別タグを追加します。タグの詳細については、「タグエディタの使用」を参照してください。
15. [Next] (次へ) を選択します。
16. 新しいポリシー設定を確認し、調整が必要なページに戻ります。

    [Policy actions] (ポリシーアクション) が [Identify resources that don't comply with the policy rules, but don't auto remediate] (ポリシールールに準拠していないリソースを特定するが、自動修復しない) に設定されていることを確認します。これにより、ポリシーを有効にする前に、ポリシーが行う変更を確認できます。
17. ポリシーが完成したら、[Create policy] (ポリシーの作成) を選択します。

    [AWS Firewall Manager ポリシー] ペインにポリシーが一覧表示されます。おそらく、アカウントの見出しの下に「保留中」と表示され、自動修復設定のステータスを示します。ポリシーの作成には数分かかることがあります。[Pending] (保留中) ステータスがアカウント数に置き換えられたら、ポリシー名を選択して、アカウントとリソースの準拠ステータスを調べることができます。詳細については、「AWS Firewall Manager ポリシーのコンプライアンス情報の表示」を参照してください。

ステップ 3: クリーンアップする

余分な料金が発生しないようにするには、不要なポリシーとリソースを削除します。

ポリシーを削除するには (コンソール)

1. [AWS Firewall Manager ポリシー] ページで、ポリシー名の横にあるラジオボタンを選択し、[削除] を選択します。
2. [Delete] (削除) 確認ボックスで [Delete all policy resources] (すべてのポリシーリソースの削除) を選択してから、もう一度 [Delete] (削除) を選択します。

    AWS WAF ポリシーと、アカウントで作成されたすべての関連リソース (ウェブ ACL など) を削除します。変更がすべてのアカウントに反映されるまでに数分かかる場合があります。

AWS Firewall ManagerAWS Shield Advanced ポリシー入門

AWS Firewall Manager を使用して、 AWS Shield Advanced 組織全体の保護を有効にすることができます。

> **Important**
>
> Firewall Manager は、Amazon Route 53 または AWS Global Acceleratorをサポートしていません。Shield Advanced を使用してこれらのリソースを保護する必要がある場合、Firewall Manager ポリシーは使用できません。代わりに、「AWS Shield AdvancedAWS リソースへの保護の追加」の手順に従ってください。

Firewall Manager を使用して Shield Advanced 保護を有効にするには、次のステップを実行します。

トピック

ステップ 1: 前提条件を満たす

AWS Firewall Managerのアカウントを準備するには、いくつかの必須のステップがあります。それらのステップは、AWS Firewall Manager 前提条件 で説明されています。「ステップ 2: Shield Advanced ポリシーを作成して適用する」に進む前に、すべての前提条件を満たしてください。

ステップ 2: Shield Advanced ポリシーを作成して適用する

前提条件を満たしたら、 AWS Firewall Manager Shield Advanced ポリシーを作成します。Firewall Manager Shield Advanced ポリシーには、Shield Advanced で保護するアカウントおよびリソースが含まれています。

> **Important**
>
> Firewall Manager は、Amazon Route 53 または AWS Global Acceleratorをサポートしていません。Shield Advanced を使用してこれらのリソースを保護する必要がある場合、Firewall

Manager ポリシーは使用できません。代わりに、「AWS Shield AdvancedAWS リソースへの保護の追加」の手順に従ってください。

Firewall Manager Shield Advanced ポリシーを作成するには (コンソール)

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. [Create policy] (ポリシーの作成) を選択します。
4. [Policy type] (ポリシータイプ) で、[Shield Advanced] を選択します。

   Shield Advanced ポリシーを作成するには、Firewall Manager 管理者アカウントが Shield Advanced にサブスクライブしている必要があります。登録されていない場合は、登録するよう求められます。サブスクリプションの費用については、「AWS Shield Advanced の料金」を参照してください。

   > **Note**
   >
   > Shield Advanced に各メンバーアカウントを手動でサブスクライブさせる必要はありません。Firewall Manager は、ポリシーの作成時にこれを行います。各アカウント内のリソースを引き続き保護するには、アカウントが Firewall Manager と Shield Advanced に登録されている必要があります。

5. リージョン で、 を選択します AWS リージョン。Amazon CloudFront リソースを保護するには、グローバル を選択します。

   複数のリージョン (リソース以外) の CloudFront リソースを保護するには、リージョンごとに個別の Firewall Manager ポリシーを作成する必要があります。
6. [Next] (次へ) を選択します。

7. [Name] (名前) で、わかりやすい名前を入力します。
8. (グローバルリージョンのみ) [Global] (グローバル) リージョンポリシーの場合、Shield Advanced アプリケーションレイヤー DDoS 自動緩和を管理するかどうかを選択できます。このチュートリアルでは、この選択をデフォルト設定の [Ignore] (無視) のままにします。
9. [Policy action] (ポリシーアクション) で、自動的に修復されないオプションを選択します。
10. [Next] (次へ) を選択します。
11. AWS アカウント このポリシーは に適用され、含めるアカウントまたは除外するアカウントを指定して、ポリシーの範囲を絞り込むことができます。このチュートリアルでは、[Include all accounts under my organization] (組織のすべてのアカウントを含める) を選択します。
12. 保護するリソースのタイプを選択します。

    Firewall Manager は、Amazon Route 53 または AWS Global Acceleratorをサポートしていません。Shield Advanced を使用してこれらのリソースを保護する必要がある場合、Firewall Manager ポリシーは使用できません。代わりに、「AWS Shield AdvancedAWS リソースへの保護の追加」の Shield Advanced のガイダンスに従ってください。
13. リソース では、指定したタグでリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

    複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

    リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。
14. [次へ] をクリックします。
15. ポリシータグ には、Firewall Manager ポリシーリソースに追加する識別タグを追加します。タグの詳細については、「タグエディタの使用」を参照してください。
16. [Next] (次へ) を選択します。
17. 新しいポリシー設定を確認し、調整が必要なページに戻ります。

    [Policy actions] (ポリシーアクション) が [Identify resources that don't comply with the policy rules, but don't auto remediate] (ポリシールールに準拠していないリソースを特定するが、自動修復しない) に設定されていることを確認します。これにより、ポリシーを有効にする前に、ポリシーが行う変更を確認できます。
18. ポリシーが完成したら、[Create policy] (ポリシーの作成) を選択します。

[AWS Firewall Manager ポリシー] ペインにポリシーが一覧表示されます。おそらく、アカウントの見出しの下に「保留中」と表示され、自動修復設定のステータスを示します。ポリシーの作成には数分かかることがあります。[Pending] (保留中) ステータスがアカウント数に置き換えられたら、ポリシー名を選択して、アカウントとリソースの準拠ステータスを調べることができます。詳細については、「AWS Firewall Manager ポリシーのコンプライアンス情報の表示」を参照してください。

「ステップ 3: (オプション) Shield Response Team (SRT) に権限を付与する」に進みます。

ステップ 3: (オプション) Shield Response Team (SRT) に権限を付与する

の利点の1つは、Shield レスポンスチーム（ SRT ） AWS Shield Advanced からのサポートです。DDoS 攻撃の兆候を見つけた場合は、AWS Support センターにお問い合わせください。必要に応じて、サポートセンターがお客様の問題を SRT にエスカレートします。SRT は疑わしいアクティビティを分析し、お客様に問題の緩和策を提供します。この緩和策には、多くの場合、 AWS WAF アカウント内のルールとウェブ ACL の作成または更新が含まれます。SRT AWS WAF AWS WAF は設定を検査してルールやウェブ ACL を作成または更新してくれますが、そのためにはチームによる許可が必要です。セットアップの一環として AWS Shield Advanced、必要な権限を事前に SRT に提供することをお勧めします。事前に権限を付与することで、実際に攻撃が発生した場合の遅延を防ぐことができます。

SRT への連絡と承認はアカウントレベルで行います。つまり、攻撃を緩和する権限を SRT に付与するには、Firewall Manager 管理者ではなくアカウント所有者が次のステップを実行する必要があります。Firewall Manager 管理者は、所有しているアカウントに関してのみ SRT に権限を付与することができます。同様に、SRT に連絡してサポートを依頼できるのは、アカウントの所有者のみです。

> **Note**
>
> SRT のサービスを使用するには、ビジネスサポートプランまたはエンタープライズサポートプランをサブスクライブする必要があります。

ユーザーに代わって攻撃を緩和する権限を SRT に付与するには、「Shield Response Team (SRT) のサポート」の手順に従ってください。SRT の許可と許可は、同じステップを使用していつでも変更できます。

「ステップ 4: Amazon SNS CloudWatch 通知とアマゾンアラームを設定する」に進みます。

ステップ 4: Amazon SNS CloudWatch 通知とアマゾンアラームを設定する

Amazon SNS CloudWatch の通知やアラームを設定しなくても、このステップから続行できます。ただし、これらのアラームと通知を設定すると、発生する可能性のある DDoS イベントの可視性が大幅に向上します。

Amazon SNS を使用すると、潜在的な DDoS アクティビティから保護されたリソースをモニタリングできます。潜在的な攻撃の通知を受け取るには、リージョンごとに Amazon SNS トピックを作成します。

Firewall Manager で Amazon SNS トピックを作成するには (コンソール)

1. Firewall Manager AWS Management Console 管理者アカウントを使用してにサインインし、で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで [AWS FMS] の [Settings] (設定) を選択します。
3. [Create new topic] (新しいトピックを作成) を選択します。
4. トピック名を入力します。
5. Amazon SNS メッセージの送信先となる E メールアドレスを入力し、[Add email address] (E メールアドレスの追加) を選択します。
6. [Update SNS configuration] (SNS 設定の更新) を選択します。

Amazon CloudWatch アラームの設定

Shield Advancedは、検出、軽減、 CloudWatch および上位貢献者の指標を監視可能な範囲に記録します。詳細については、「」を参照してください。AWS Shield Advanced 指標 CloudWatch 追加費用が発生する。 CloudWatch 価格については、Amazon CloudWatch の料金表をご覧ください。

CloudWatch アラームを作成するには、「Amazon CloudWatch アラームの使用」の指示に従ってください。デフォルトでは、Shield Advanced は潜在的な DDoS イベントが 1 CloudWatch 回発生した

らアラートを出すように設定しています。必要に応じて、CloudWatch コンソールを使用してこの設定を変更し、複数の指標が検出された後にのみ警告するようにできます。

> **Note**
>
> アラームに加えて、 CloudWatch ダッシュボードを使用して潜在的な DDoS アクティビティを監視することもできます。ダッシュボードは Shield Advanced から raw データを収集し、ほぼリアルタイムの読み取り可能なメトリクスに加工します。Amazon CloudWatch の統計情報を使用して、ウェブアプリケーションやサービスのパフォーマンスを把握できます。詳細については、「Amazon CloudWatch CloudWatch ユーザーガイドの内容」を参照してください。
>
> CloudWatch ダッシュボードの作成方法については、を参照してくださいAmazon によるモニタリング CloudWatch。ダッシュボードに追加できる特定の Shield Advanced メトリクスの詳細については、「AWS Shield Advanced 指標」を参照してください。

Shield Advanced の設定が完了したら、DDoS イベントの可視性 でイベントを表示するためのオプションをよく理解してください。

AWS Firewall Manager Amazon VPC セキュリティグループポリシーの使用を開始する

を使用して組織全体で Amazon VPC AWS Firewall Manager セキュリティグループを有効にするには、次の手順を順番に実行してください。

トピック

- ステップ 1: 前提条件を満たす
- ステップ 2: ポリシーで使用するセキュリティグループを作成する
- ステップ 3: 共通セキュリティグループポリシーを作成して適用する

ステップ 1: 前提条件を満たす

AWS Firewall Managerのアカウントを準備するには、いくつかの必須のステップがあります。それらのステップは、AWS Firewall Manager 前提条件 で説明されています。「ステップ 2: ポリシーで使用するセキュリティグループを作成する」に進む前に、すべての前提条件を満たしてください。

ステップ 2: ポリシーで使用するセキュリティグループを作成する

このステップでは、Firewall Manager を使用して組織全体に適用できるセキュリティグループを作成します。

> **Note**
>
> このチュートリアルでは、セキュリティグループポリシーを組織内のリソースに適用しません。ポリシーを作成し、ポリシーのセキュリティグループをリソースに適用した場合どうなるかを確認するだけです。これを行うには、ポリシーの自動修復を無効にします。

一般的なセキュリティグループが既に定義されている場合は、このステップを省略して「ステップ 3: 共通セキュリティグループポリシーを作成して適用する」に進みます。

Firewall Manager 共通のセキュリティグループポリシーで使用するセキュリティグループを作成するには

- 「Amazon VPC ユーザーガイド」の「Security Groups for Your VPC」(VPC のセキュリティグループ) のガイダンスに従って、組織内のすべてのアカウントとリソースに適用できるセキュリティグループを作成します。

  セキュリティグループルールオプションの詳細については、「セキュリティグループのルールのリファレンス」を参照してください。

これで「ステップ 3: 共通セキュリティグループポリシーを作成して適用する」に進む準備ができました。

ステップ 3: 共通セキュリティグループポリシーを作成して適用する

前提条件を完了したら、 AWS Firewall Manager 共通のセキュリティグループポリシーを作成します。共通セキュリティグループポリシーは、 AWS 組織全体に一元管理されたセキュリティグループを提供します。また、セキュリティグループが適用される AWS アカウント および リソースも定義します。Firewall Manager では、共通セキュリティグループポリシーに加えて、組織内で使用中のセキュリティグループルールを管理するためのコンテンツ監査セキュリティグループポリシーと、未使用および冗長セキュリティグループを管理するための使用状況監査セキュリティグループポリシーがサポートされています。詳細については、「セキュリティグループポリシー」を参照してください。

このチュートリアルでは、共通セキュリティグループポリシーを作成し、そのアクションを自動的に修復しないように設定します。これにより、 AWS 組織を変更せずにポリシーがどのような影響を与えるかを確認できます。

Firewall Manager の一般的なセキュリティグループポリシーを作成するには (コンソール)

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. 前提条件を満たしていない場合、問題を修正する方法についての指示がコンソールに表示されます。指示に従ってから、このステップに戻り、共通セキュリティグループポリシーを作成します。
4. [Create policy] (ポリシーの作成) を選択します。
5. [Policy type] (ポリシータイプ) で、[Security group] (セキュリティグループ) を選択します。
6. [Security group policy type] (セキュリティグループポリシータイプ) で、[Common security groups] (共通セキュリティグループ) を選択します。
7. リージョン で、 を選択します AWS リージョン。
8. [Next] (次へ) を選択します。
9. [Policy name] (ポリシー名) で、わかりやすい名前を入力します。
10. [Policy rules] (ポリシールール) を使用すると、このポリシーのセキュリティグループの適用方法と保守方法を選択できます。このチュートリアルでは、オプションのチェックをオフのままにします。
11. [Add primary security group] (プライマリセキュリティグループの追加) を選択し、このチュートリアル用に作成したセキュリティグループを選択して、[Add security group] (セキュリティグループの追加) を選択します。
12. [Policy action] (ポリシーアクション) で、[Identify resources that don't comply with the policy rules, but don't auto remediate] (ポリシールールに準拠していないリソースを特定するが、自動修復しない) を選択します。

13. [Next] (次へ) を選択します。
14. AWS アカウント このポリシーの影響を受けると、含めるアカウントまたは除外するアカウントを指定して、ポリシーの範囲を絞り込むことができます。このチュートリアルでは、[Include all accounts under my organization] (組織のすべてのアカウントを含める) を選択します。
15. リソースタイプ で、 AWS 組織用に定義したリソースに応じて、1 つ以上のタイプを選択します。
16. リソース では、指定したタグでリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

    複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

    リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。
17. [次へ] をクリックします。
18. ポリシータグ には、Firewall Manager ポリシーリソースに追加する識別タグを追加します。タグの詳細については、「タグエディタの使用」を参照してください。
19. [Next] (次へ) を選択します。
20. 新しいポリシー設定を確認し、調整が必要なページに戻ります。

    [Policy actions] (ポリシーアクション) が [Identify resources that don't comply with the policy rules, but don't auto remediate] (ポリシールールに準拠していないリソースを特定するが、自動修復しない) に設定されていることを確認します。これにより、ポリシーを有効にする前に、ポリシーが行う変更を確認できます。
21. ポリシーが完成したら、[Create policy] (ポリシーの作成) を選択します。

    [AWS Firewall Manager ポリシー] ペインにポリシーが一覧表示されます。おそらく、アカウントの見出しの下に「保留中」と表示され、自動修復設定のステータスを示します。ポリシーの作成には数分かかることがあります。[Pending] (保留中) ステータスがアカウント数に置き換えられたら、ポリシー名を選択して、アカウントとリソースの準拠ステータスを調べることができます。詳細については、「AWS Firewall Manager ポリシーのコンプライアンス情報の表示」を参照してください。
22. 調べ終わったら、このチュートリアルで作成したポリシーを保持しない場合は、ポリシー名を選択して [Delete] (削除) を選択し、[Clean up resources created by this policy.] (このポリシーに

よって作成されたリソースをクリーンアップします。) を選択して、最後に [Delete] (削除) を選択します。

Firewall Manager セキュリティグループポリシーの詳細については、「セキュリティグループポリシー」を参照してください。

AWS Firewall Manager Amazon VPC ネットワーク ACL ポリシーの開始方法

AWS Firewall Manager を使用して組織全体ACLs を有効にするには、このセクションの手順を順番に実行します。

ネットワーク ACLs「Amazon VPC ユーザーガイド」の「ネットワーク ACLs」を参照してください。

トピック

- ステップ 1: 前提条件を満たす
- ステップ 2: ネットワーク ACL ポリシーを作成する

ステップ 1: 前提条件を満たす

AWS Firewall Managerのアカウントを準備するには、いくつかの必須のステップがあります。それらのステップは、AWS Firewall Manager 前提条件 で説明されています。「ステップ 2: ネットワーク ACL ポリシーを作成する」に進む前に、すべての前提条件を満たしてください。

ステップ 2: ネットワーク ACL ポリシーを作成する

前提条件を満たしたら、Firewall Manager ネットワーク ACL ポリシーを作成します。ネットワーク ACL ポリシーは、 AWS 組織全体に一元管理されたネットワーク ACL 定義を提供します。また、ネットワーク ACL が適用される AWS アカウント および サブネットも定義します。

Firewall Manager のネットワーク ACL ポリシーの詳細については、「」を参照してくださいネットワーク ACL ポリシー。

Firewall Manager のネットワーク ACL ポリシーの一般的な情報については、「」を参照してくださいネットワーク ACL ポリシー。

> **Note**
>
> このチュートリアルでは、ネットワーク ACL ポリシーを組織内のサブネットに適用しません。ポリシーを作成し、ポリシーのネットワーク ACL をサブネットに適用した場合どうなるかを確認します。これを行うには、ポリシーの自動修復を無効にします。

Firewall Manager ネットワーク ACL ポリシーを作成するには (コンソール)

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. 前提条件を満たしていない場合、問題を修正する方法についての指示がコンソールに表示されます。手順に従ってから、このステップに戻り、ネットワーク ACL ポリシーを作成します。
4. [ポリシーの作成] を選択します。
5. リージョン で、 を選択します AWS リージョン。
6. ポリシータイプ で、ネットワーク ACL を選択します。
7. [Next] (次へ) を選択します。
8. [Policy name] (ポリシー名) で、わかりやすい名前を入力します。
9. ネットワーク ACL ポリシールール では、インバウンドトラフィックとアウトバウンドトラフィックの両方について、最初と最後のルールを定義します。

   Firewall Manager でネットワーク ACL ルールを定義する方法は、Amazon VPC でルールを定義する方法と同様です。唯一の違いは、ルール番号を自分で割り当てるのではなく、ルールの各セットを実行する順序を割り当て、ポリシーを保存するときに Firewall Manager が番号を割り当てることです。最大 5 つのインバウンドルールを定義し、最初と最後に任意の方法で分割できます。また、最大 5 つのアウトバウンドルールを定義できます。

ネットワーク ACL ルールを指定するガイダンスについては、「Amazon VPC ユーザーガイド」の「ネットワーク ACL ルールの追加と削除」を参照してください。

Firewall Manager ポリシーで定義するルールは、ネットワーク ACL がネットワーク ACL ポリシーに準拠する必要がある最小ルール設定を指定します。例えば、ネットワーク ACL のインバウンドルールは、ポリシーで指定されているのと同じ順序で、ポリシーのインバウンドファーストルールとして で始まらない限り、ポリシーに準拠できません。詳細については、「ネットワーク ACL ポリシー」を参照してください。

10. [Policy action] (ポリシーアクション) で、[Identify resources that don't comply with the policy rules, but don't auto remediate] (ポリシールールに準拠していないリソースを特定するが、自動修復しない) を選択します。
11. [Next] (次へ) を選択します。
12. AWS アカウント このポリシーの影響を受けると、含めるアカウントまたは除外するアカウントを指定して、ポリシーの範囲を絞り込むことができます。このチュートリアルでは、[Include all accounts under my organization] (組織のすべてのアカウントを含める) を選択します。

    ネットワーク ACL ポリシーのリソースタイプは常にサブネットです。
13. リソース では、指定したタグを持つリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

    複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

    リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。
14. [次へ] をクリックします。
15. ポリシータグ には、Firewall Manager ポリシーリソースに追加する識別タグを追加します。タグの詳細については、「タグエディタの使用」を参照してください。
16. [Next] (次へ) を選択します。
17. 新しいポリシー設定を確認し、調整が必要なページに戻ります。

    [Policy actions] (ポリシーアクション) が [Identify resources that don't comply with the policy rules, but don't auto remediate] (ポリシールールに準拠していないリソースを特定するが、自動

修復しない) に設定されていることを確認します。これにより、ポリシーを有効にする前に、ポリシーが行う変更を確認できます。

18. ポリシーが完成したら、[Create policy] (ポリシーの作成) を選択します。

    [AWS Firewall Manager ポリシー] ペインにポリシーが一覧表示されます。おそらく、アカウントの見出しの下に「保留中」と表示され、自動修復設定のステータスを示します。ポリシーの作成には数分かかることがあります。[Pending] (保留中) ステータスがアカウント数に置き換えられたら、ポリシー名を選択して、アカウントとリソースの準拠ステータスを調べることができます。詳細については、「AWS Firewall Manager ポリシーのコンプライアンス情報の表示」を参照してください。

19. 調べ終わったら、このチュートリアルで作成したポリシーを保持しない場合は、ポリシー名を選択して [Delete] (削除) を選択し、[Clean up resources created by this policy.] (このポリシーによって作成されたリソースをクリーンアップします。) を選択して、最後に [Delete] (削除) を選択します。

Firewall Manager のネットワーク ACL ポリシーの詳細については、「」を参照してくださいネットワーク ACL ポリシー。

AWS Firewall ManagerAWS Network Firewall ポリシー入門

AWS を使用して組織全体でNetwork Firewall AWS Firewall Manager を有効にするには、次の手順を順番に実行してください。Firewall Manager の Network Firewall ポリシーについては、「AWS Network Firewall ポリシー」を参照してください。

トピック

- ステップ 1: 一般的な前提条件を満たす
- ステップ 2: ポリシーで使用する Network Firewall ルールグループを作成する
- ステップ 3: Network Firewall ポリシーを作成して適用する

ステップ 1: 一般的な前提条件を満たす

AWS Firewall Managerのアカウントを準備するには、いくつかの必須のステップがあります。それらのステップは、AWS Firewall Manager 前提条件 で説明されています。次のステップに進む前に、すべての前提条件を満たしてください。

ステップ 2: ポリシーで使用する Network Firewall ルールグループを作成する

このチュートリアルに従うには、 AWS Network Firewall ルールグループとファイアウォールポリシーに精通し、その設定方法を知っている必要があります。

AWS Firewall Manager ポリシーで使用される Network Firewall には、少なくとも 1 つのルールグループが必要です。Network Firewall でルールグループをまだ作成していない場合は、ここで作成します。Network Firewall の使用については、「AWS Network Firewall デベロッパーガイド」を参照してください。

ステップ 3: Network Firewall ポリシーを作成して適用する

前提条件を満たしてから、 AWS Firewall Manager Network Firewall ポリシーを作成します。Network Firewall ポリシーは、組織全体 AWS に一元管理された AWS Network Firewall ファイアウォールを提供します。また、ファイアウォールが適用される AWS アカウント および リソースも定義します。

Firewall Manager が Network Firewall ポリシーを管理する方法の詳細については、「AWS Network Firewall ポリシー」を参照してください。

Firewall Manager の Network Firewall ポリシーを作成するには (コンソール)

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. 前提条件を満たしていない場合、問題を修正する方法についての指示がコンソールに表示されます。指示を実行してから、このステップに戻り、Network Firewall ポリシーを作成します。
4. [Create security policy] (セキュリティポリシーを作成) を選択します。
5. [Policy type] (ポリシータイプ) では [AWS Network Firewall] を選択します。
6. リージョン で、 を選択します AWS リージョン。

7. [Next] (次へ) を選択します。
8. [Policy name] (ポリシー名) で、わかりやすい名前を入力します。
9. ポリシー設定では、ファイアウォールポリシーを定義できます。これは、 AWS Network Firewall コンソールで使用するプロセスと同じプロセスです。ポリシーで使用するルールグループを追加し、デフォルトのステートレスアクションを指定します。このチュートリアルでは、Network Firewall におけるファイアウォールポリシーと同様に、このポリシーを設定します。

> **Note**
>
> 自動修復は AWS Firewall Manager Network Firewall ポリシーに対して自動的に行われるため、ここで自動修復を選択しないオプションは表示されません。

10. [Next] (次へ) を選択します。
11. [Firewall endpoints] (ファイアウォールエンドポイント) で、[Multiple firewall endpoints] (複数のファイアウォールエンドポイント) を選択します。このオプションは、ファイアウォールについて高い可用性を実現します。ポリシーを作成すると、Firewall Manager は、保護するパブリックサブネットがある各アベイラビリティーゾーンにファイアウォールサブネットを作成します。
12. [AWS Network Firewall ルート設定] で [モニタリング] をクリックします。これにより、Firewall Manager が VPC のルート設定違反をモニタリングし、ルートのコンプライアンス準拠に役立つ改善策のアラートを発信します。オプションで、ルート設定を Firewall Manager でモニタリングしてこれらのアラートを受信したくない場合は、[Off] (オフ) を選択します。

> **Note**
>
> モニタリングにより、ルート設定の誤りが原因で非準拠となっているリソースに関する詳細が提供され、Firewall Manager `GetViolationDetails` API からの修復アクションが提案されます。例えば、Network Firewall は、ポリシーによって作成されたファイアウォールエンドポイントを通じてトラフィックがルーティングされない場合に警告します。

> **Warning**
>
> [Monitor] (モニタリング) を選択した場合、同じポリシーについて将来 [Off] (オフ) に変更することはできません。新しいポリシーを作成する必要があります。

13. [Traffic type] (トラフィックの種類) で、[Add to firewall policy] (ファイアウォールポリシーに追加) を選択して、インターネットゲートウェイを介してトラフィックをルーティングします。
14. AWS アカウント このポリシーの影響を受ける では、包含または除外するアカウントを指定することで、ポリシーの範囲を絞り込むことができます。このチュートリアルでは、[Include all accounts under my organization] (組織のすべてのアカウントを含める) を選択します。

    Network Firewall ポリシーの [Resource type] (リソースタイプ) は常に [VPC] です。
15. リソース では、指定したタグでリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

    複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

    リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。
16. [次へ] をクリックします。
17. ポリシータグ には、Firewall Manager ポリシーリソースに追加する識別タグを追加します。タグの詳細については、「タグエディタの使用」を参照してください。
18. [Next] (次へ) を選択します。
19. 新しいポリシー設定を確認し、調整が必要なページに戻ります。

    [Policy actions] (ポリシーアクション) が [Identify resources that don't comply with the policy rules, but don't auto remediate] (ポリシールールに準拠していないリソースを特定するが、自動修復しない) に設定されていることを確認します。これにより、ポリシーを有効にする前に、ポリシーが行う変更を確認できます。
20. ポリシーが完成したら、[Create policy] (ポリシーの作成) を選択します。

    [AWS Firewall Manager ポリシー] ペインにポリシーが一覧表示されます。おそらく、アカウントの見出しの下に保留中と表示され、自動修復設定のステータスを示します。ポリシーの作成には数分かかることがあります。[Pending] (保留中) ステータスがアカウント数に置き換えられたら、ポリシー名を選択して、アカウントとリソースの準拠ステータスを調べることができます。詳細については、「AWS Firewall Manager ポリシーのコンプライアンス情報の表示」を参照してください。
21. 調べ終わったら、このチュートリアルで作成したポリシーを保持しない場合は、ポリシー名を選択して [Delete] (削除) を選択し、[Clean up resources created by this policy.] (このポリシーに

よって作成されたリソースをクリーンアップします。) を選択して、最後に [Delete] (削除) を選択します。

Firewall Manager の Network Firewall ポリシーの詳細については、「AWS Network Firewall ポリシー」を参照してください。

AWS Firewall Manager DNS ファイアウォールポリシー入門

を使用して Amazon Route 53 リゾルバー DNS AWS Firewall Manager ファイアウォールを組織全体で有効にするには、以下の手順を順番に実行してください。Firewall Manager の DNS Firewall ポリシーについては、「Amazon Route 53 Resolver DNS Firewall ポリシー」を参照してください。

トピック

- ステップ 1: 一般的な前提条件を満たす
- ステップ 2: ポリシーで使用する DNS Firewall ルールグループを作成する
- ステップ 3: DNS Firewall ポリシーを作成して適用する

ステップ 1: 一般的な前提条件を満たす

AWS Firewall Managerのアカウントを準備するには、いくつかの必須のステップがあります。それらのステップは、AWS Firewall Manager 前提条件 で説明されています。次のステップに進む前に、すべての前提条件を満たしてください。

ステップ 2: ポリシーで使用する DNS Firewall ルールグループを作成する

このチュートリアルの手順を実行するには、Amazon Route 53 Resolver DNS Firewall に精通しており、そのルールグループを設定する方法を理解している必要があります。

AWS Firewall Manager ポリシーで使用される DNS Firewall には少なくとも 1 つのルールグループが必要です。DNS Firewall でルールグループをまだ作成していない場合は、ここで作成します。DNS Firewall の使用方法については、「Amazon Route 53 デベロッパーガイド」の「Amazon Route 53 Resolver DNS Firewall」を参照してください。

ステップ 3: DNS Firewall ポリシーを作成して適用する

前提条件を満たしたら、 AWS Firewall Manager DNS Firewall ポリシーを作成します。DNS Firewall ポリシーは、 AWS 組織全体に対して一元管理された DNS Firewall ルールグループの関連付けの

セットを提供します。また、ファイアウォールが適用される AWS アカウント とリソースも定義します。

Firewall Manager が DNS Firewall ルールグループの関連付けを管理する方法の詳細については、「Amazon Route 53 Resolver DNS Firewall ポリシー」を参照してください。

Firewall Manager の DNS Firewall ポリシーを作成するには (コンソール)

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。
2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. 前提条件を満たしていない場合、問題を修正する方法についての指示がコンソールに表示されます。指示を実行してから、このステップに戻り、DNS Firewall ポリシーを作成します。
4. [Create security policy] (セキュリティポリシーを作成) を選択します。
5. [Policy type] (ポリシータイプ) で、[Amazon Route 53 Resolver DNS Firewall] を選択します。
6. リージョン で、 を選択します AWS リージョン。
7. [Next] (次へ) を選択します。
8. [Policy name] (ポリシー名) で、わかりやすい名前を入力します。
9. ポリシー設定を使用すると、Firewall Manager から管理する DNS Firewall ルールグループの関連付けを定義できます。ポリシーで使用するルールグループを追加します。VPC について、最初に評価する関連付けと、最後に評価する関連付けを定義できます。このチュートリアルでは、必要に応じて 1 つまたは 2 つのルールグループの関連付けを追加します。
10. [Next] (次へ) を選択します。
11. AWS アカウント このポリシーの影響を受ける では、包含または除外するアカウントを指定することで、ポリシーの範囲を絞り込むことができます。このチュートリアルでは、[Include all accounts under my organization] (組織のすべてのアカウントを含める) を選択します。

    DNS Firewall ポリシーの [Resource type] (リソースタイプ) は常に [VPC] です。
12. リソース では、指定したタグを持つリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

    複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。

13. [次へ] をクリックします。
14. ポリシータグ には、Firewall Manager ポリシーリソースに追加する識別タグを追加します。タグの詳細については、「タグエディタの使用」を参照してください。
15. [Next] (次へ) を選択します。
16. 新しいポリシー設定を確認し、調整が必要なページに戻ります。

    [Policy actions] (ポリシーアクション) が [Identify resources that don't comply with the policy rules, but don't auto remediate] (ポリシールールに準拠していないリソースを特定するが、自動修復しない) に設定されていることを確認します。これにより、ポリシーを有効にする前に、ポリシーが行う変更を確認できます。
17. ポリシーが完成したら、[Create policy] (ポリシーの作成) を選択します。

    [AWS Firewall Manager ポリシー] ペインにポリシーが一覧表示されます。アカウントの見出しの下には「保留中」と表示され、自動修復設定のステータスを示します。ポリシーの作成には数分かかることがあります。[Pending] (保留中) ステータスがアカウント数に置き換えられたら、ポリシー名を選択して、アカウントとリソースの準拠ステータスを調べることができます。詳細については、「AWS Firewall Manager ポリシーのコンプライアンス情報の表示」を参照してください。
18. 調べ終わったら、このチュートリアルで作成したポリシーを保持しない場合は、ポリシー名を選択して [Delete] (削除) を選択し、[Clean up resources created by this policy.] (このポリシーによって作成されたリソースをクリーンアップします。) を選択して、最後に [Delete] (削除) を選択します。

Firewall Manager の DNS Firewall ポリシーの詳細については、「Amazon Route 53 Resolver DNS Firewall ポリシー」を参照してください。

AWS Firewall Managerパロアルトネットワークスのクラウド次世代ファイアウォールポリシー入門

パロアルトネットワークスのクラウド次世代ファイアウォール (NGFW) AWS Firewall Manager ポリシーを有効にするには、以下の手順を順番に実行してください。Palo Alto Networks Cloud NGFW ポリシーについては、「Palo Alto Networks Cloud NGFW ポリシー」を参照してください。

トピック

ステップ 1: 一般的な前提条件を満たす

AWS Firewall Managerのアカウントを準備するには、いくつかの必須のステップがあります。それらのステップは、AWS Firewall Manager 前提条件 で説明されています。次のステップに進む前に、すべての前提条件を満たしてください。

ステップ 2: Complete the Palo Alto Networks Cloud NGFW ポリシーの前提条件を満たす

Palo Alto Networks Cloud NGFW ポリシーを使用するには、いくつかの追加の必須ステップを完了する必要があります。それらのステップは、Palo Alto Networks Cloud Next Generation Firewall ポリシーの要件 で説明されています。次のステップに進む前に、すべての前提条件を満たしてください。

ステップ 3: Palo Alto Networks Cloud NGFW ポリシーを作成して適用する

前提条件を満たしたら、 AWS Firewall Manager Palo Alto Networks Cloud NGFW ポリシーを作成します。

Firewall Manager の Palo Alto Networks Cloud NGFW ポリシーの詳細については、「Palo Alto Networks Cloud NGFW ポリシー」を参照してください。

Palo Alto Networks Cloud NGFW の Firewall Manager ポリシーを作成するには (コンソール)

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. [Create policy] (ポリシーの作成) を選択します。
4. [Policy type] (ポリシータイプ) で、[Palo Alto Networks Cloud NGFW] を選択します。 AWS Marketplace で Palo Alto Networks Cloud NGFW サービスをまだサブスクライブしていない場合は、まずサブスクライブする必要があります。 AWS Marketplace でサブスクライブするには、AWS Marketplace の詳細を表示 を選択します。
5. [Deployment model] (デプロイモデル) で、[Distributed model] (分散モデル) または [Centralized model] (集約型モデル) のいずれかを選択します。デプロイモデルによって、Firewall Manager がポリシーのエンドポイントを管理する方法が決まります。分散モデルでは、Firewall Manager は、ポリシーの範囲内の各 VPC にファイアウォールエンドポイントを維持します。集約型モデルでは、Firewall Manager は検査 VPC に単一のエンドポイントを維持します。
6. リージョン で、 を選択します AWS リージョン。複数のリージョンのリソースを保護するには、各リージョンに別々の ポリシーを作成する必要があります。
7. [Next] (次へ) を選択します。
8. [Policy name] (ポリシー名) で、わかりやすい名前を入力します。
9. ポリシー設定で、このポリシーに関連付ける Palo Alto Networks Cloud NGFW ファイアウォールポリシーを選択します。Palo Alto Networks Cloud NGFW ファイアウォールポリシーの一覧には、Palo Alto Networks Cloud NGFW テナントに関連付けられているすべての Cloud NGFW ファイアウォールポリシーが含まれています。Palo Alto Networks Cloud NGFW ファイアウォールポリシーの作成と管理については、「デプロイガイド」の「Palo Alto Networks Cloud NGFW for トピック AWSAWS Firewall Manager」で「 の Palo Alto Networks Cloud NGFW のデプロイ AWS 」を参照してください。
10. Palo Alto Networks Cloud NGFW ログ記録 - オプションで、ポリシーのログ記録する Palo Alto Networks Cloud NGFW ログタイプ (複数可) をオプションで選択します。Palo Alto Networks Cloud NGFW ログタイプの詳細については、「デプロイガイド」の「Palo Alto Networks Cloud NGFW のログ記録の設定 AWS」を参照してください。 AWS

    [log destination] (ログの宛先) で、Firewall Manager がログを書き込む場合を指定します。
11. [Next] (次へ) を選択します。
12. [Configure third-party firewall endpoint] (サードパーティーのファイアウォールエンドポイントを設定) で、ファイアウォールエンドポイントの作成に分散デプロイモデルと集約型デプロイモデルのいずれを使用しているかに応じて、次のいずれかを実行します。

    - このポリシーに分散デプロイモデルを使用している場合は、[Availability Zones] (アベイラビリティーゾーン) で、ファイアウォールエンドポイントを作成するアベイラビリティーゾーン

を選択します。アベイラビリティーゾーンは、[Availability Zone name] (アベイラビリティーゾーン名) または [Availability Zone ID] (アベイラビリティーゾーン ID) で選択できます。

- このポリシーに集約型デプロイモデルを使用している場合は、[Inspection VPC configuration] (検査 VPC 設定) の [AWS Firewall Manager endpoint configuration] (エンドポイント設定) で、検査 VPC の所有者の AWS アカウント ID と検査 VPC の VPC ID を入力します。
  - [Availability Zones] (アベイラビリティーゾーン) で、ファイアウォールエンドポイントを作成するアベイラビリティーゾーンを選択します。アベイラビリティーゾーンは、[Availability Zone name] (アベイラビリティーゾーン名) または [Availability Zone ID] (アベイラビリティーゾーン ID) で選択できます。

13. [Next] (次へ) を選択します。
14. [Policy scope] (ポリシーの範囲) の [AWS アカウント this policy applies to] (このポリシーが適用される) で、次のようにオプションを選択します。

    - 組織内のすべてのアカウントにポリシーを適用する場合は、デフォルトの選択のままにし、AWS 組織 のすべてのアカウントを含めます。
    - ポリシーを特定の AWS Organizations 組織単位 (OUs) 内の特定のアカウントにのみ適用する場合は、指定したアカウントと組織単位のみを含める を選択し、含めるアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。
    - 特定のアカウントまたは AWS Organizations 組織単位 (OUs) 以外のすべてのアカウントにポリシーを適用する場合は、指定したアカウントと組織単位を除外し、その他すべての を含めて、除外するアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。

    選択できるオプションは 1 つのみです。

    ポリシーを適用すると、Firewall Manager は新しいアカウントを設定と照合して自動的に評価します。例えば、特定のアカウントのみを含めると、Firewall Manager は新しいアカウントにポリシーを適用しません。別の例として、OU を含めた場合、OU またはその子である OU にアカウントを追加すると、Firewall Manager は新しいアカウントにポリシーを自動的に適用します。

    Network Firewall ポリシーの [Resource type] (リソースタイプ) は [VPC] です。
15. リソース では、指定したタグを持つリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。

16. [Grant cross-account access] (クロスアカウントアクセスを付与) で、[Download AWS CloudFormation template] (テンプレートをダウンロード) を選択します。これにより、 AWS CloudFormation スタックの作成に使用できる AWS CloudFormation テンプレートがダウンロードされます。このスタックは、Palo Alto Networks Cloud NGFW リソースを管理するためのクロスアカウントアクセス許可を Firewall Manager に付与する AWS Identity and Access Management ロールを作成します。スタックの詳細については、「AWS CloudFormation ユーザーガイド」の「StackSets の操作」を参照してください。
17. [Next] (次へ) を選択します。
18. ポリシータグ には、Firewall Manager ポリシーリソースに追加する識別タグを追加します。タグの詳細については、「タグエディタの使用」を参照してください。
19. [Next] (次へ) を選択します。
20. 新しいポリシー設定を確認し、調整が必要なページに戻ります。

    [Policy actions] (ポリシーアクション) が [Identify resources that don't comply with the policy rules, but don't auto remediate] (ポリシールールに準拠していないリソースを特定するが、自動修復しない) に設定されていることを確認します。これにより、ポリシーを有効にする前に、ポリシーが行う変更を確認できます。
21. ポリシーが完成したら、[Create policy] (ポリシーの作成) を選択します。

    [AWS Firewall Manager ポリシー] ペインにポリシーが一覧表示されます。おそらく、アカウントの見出しの下に「保留中」と表示され、自動修復設定のステータスを示します。ポリシーの作成には数分かかることがあります。[Pending] (保留中) ステータスがアカウント数に置き換えられたら、ポリシー名を選択して、アカウントとリソースの準拠ステータスを調べることができます。詳細については、「AWS Firewall Manager ポリシーのコンプライアンス情報の表示」を参照してください。

Firewall Manager Palo Alto Networks Cloud NGFW ポリシーの詳細については、「Palo Alto Networks Cloud NGFW ポリシー」を参照してください。

AWS Firewall ManagerフォーティゲートCNFポリシーの使用を開始する

Fortigate Cloud Native Firewall (CNF) as a Serviceは、ポリシーに使用できるサードパーティのファイアウォールサービスです。 AWS Firewall Manager Fortigate CNF for Firewall Manager を使用すると、Fortigate CNFのリソースとポリシーセットを作成し、すべてのアカウントに一元的に導入できます。 AWS を使用してFortigate AWS Firewall Manager CNFポリシーを有効にするには、以下の手順を順番に実行してください。Fortigate CNF ポリシーの詳細については、「Fortigate Cloud Native Firewall (CNF) as a Service ポリシー」を参照してください。

トピック

ステップ 1: 一般的な前提条件を満たす

AWS Firewall Managerのアカウントを準備するには、いくつかの必須のステップがあります。それらのステップは、AWS Firewall Manager 前提条件 で説明されています。次のステップに進む前に、すべての前提条件を満たしてください。

ステップ 2: Fortigate CNF ポリシーの前提条件を完了する

Fortigate CNF ポリシーを使用するには、追加の必須ステップを完了する必要があります。それらのステップは、Fortigate Cloud Native Firewall (CNF) as a Service ポリシーの前提条件 で説明されています。次のステップに進む前に、すべての前提条件を満たしてください。

ステップ 3: Fortigate CNF ポリシーを作成して適用する

前提条件を満たしたら、Fortigate AWS Firewall Manager CNF ポリシーを作成します。

Fortigate CNF の Firewall Manager ポリシーの詳細については、「Fortigate Cloud Native Firewall (CNF) as a Service ポリシー」を参照してください。

Fortigate CNF の Firewall Manager ポリシーを作成するには (コンソール)

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

> **Note**
>
> Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. [Create policy] (ポリシーの作成) を選択します。
4. [Policy type] (ポリシータイプ) には、Fortigate CNF を選択してください。 AWS Marketplace で Fortigate CNF サービスをまだサブスクライブしていない場合は、まずサブスクライブする必要があります。 AWS Marketplace でサブスクライブするには、 AWS Marketplace の詳細を表示を選択します。
5. [Deployment model] (デプロイモデル) で、[Distributed model] (分散モデル) または [Centralized model] (集約型モデル) のいずれかを選択します。デプロイモデルによって、Firewall Manager がポリシーのエンドポイントを管理する方法が決まります。分散モデルでは、Firewall Manager は、ポリシーの範囲内の各 VPC にファイアウォールエンドポイントを維持します。集約型モデルでは、Firewall Manager は検査 VPC に単一のエンドポイントを維持します。
6. リージョン で、 を選択します AWS リージョン。複数のリージョンのリソースを保護するには、各リージョンに別々の ポリシーを作成する必要があります。
7. [Next] (次へ) を選択します。
8.
9. ポリシー設定で、このポリシーに関連付ける Fortigate CNF ファイアウォールポリシーを選択します。Fortigate CNF ファイアウォールポリシーのリストには、Fortigate CNF テナントに関連付けられているすべての CNF ファイアウォールポリシーが含まれています。Fortigate CNF ファイアウォールポリシーの作成と管理については、「Fortigate CNF documentation」(Fortigate CNF のドキュメント) を参照してください。
10. [Next] (次へ) を選択します。
11. [Configure third-party firewall endpoint] (サードパーティーのファイアウォールエンドポイントを設定) で、ファイアウォールエンドポイントの作成に分散デプロイモデルと集約型デプロイモデルのいずれを使用しているかに応じて、次のいずれかを実行します。

    - このポリシーに分散デプロイモデルを使用している場合は、[Availability Zones] (アベイラビリティーゾーン) で、ファイアウォールエンドポイントを作成するアベイラビリティーゾーンを選択します。アベイラビリティーゾーンは、[Availability Zone name] (アベイラビリティーゾーン名) または [Availability Zone ID] (アベイラビリティーゾーン ID) で選択できます。

- このポリシーに集約型デプロイモデルを使用している場合は、[Inspection VPC configuration] (検査 VPC 設定) の [AWS Firewall Manager endpoint configuration] (エンドポイント設定) で、検査 VPC の所有者の AWS アカウント ID と検査 VPC の VPC ID を入力します。
  - [Availability Zones] (アベイラビリティーゾーン) で、ファイアウォールエンドポイントを作成するアベイラビリティーゾーンを選択します。アベイラビリティーゾーンは、[Availability Zone name] (アベイラビリティーゾーン名) または [Availability Zone ID] (アベイラビリティーゾーン ID) で選択できます。

12. [Next] (次へ) を選択します。
13. [Policy scope] (ポリシーの範囲) の [AWS アカウント this policy applies to] (このポリシーが適用される) で、次のようにオプションを選択します。

    - 組織内のすべてのアカウントにポリシーを適用する場合は、デフォルトの選択のままにし、AWS 組織 のすべてのアカウントを含めます。
    - ポリシーを特定のアカウントまたは特定の AWS Organizations 組織単位 (OUsにあるアカウントにのみ適用する場合は、指定されたアカウントと組織単位のみを含める を選択し、含めるアカウントと OUsを追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。
    - 特定のアカウントまたは AWS Organizations 組織単位 (OUs) を除くすべてのアカウントにポリシーを適用する場合は、指定されたアカウントと組織単位を除外し、その他すべての を含めて、除外するアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。

    選択できるオプションは 1 つのみです。

    ポリシーを適用すると、Firewall Manager は新しいアカウントを設定と照合して自動的に評価します。例えば、特定のアカウントのみを含めると、Firewall Manager は新しいアカウントにポリシーを適用しません。別の例として、OU を含めた場合、OU またはその子である OU にアカウントを追加すると、Firewall Manager は新しいアカウントにポリシーを自動的に適用します。

    Fortigate CNF ポリシーのリソースタイプは VPC です。
14. リソース では、指定したタグでリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。

15. [Grant cross-account access] (クロスアカウントアクセスを付与) で、[Download AWS CloudFormation template] (テンプレートをダウンロード) を選択します。これにより、 AWS CloudFormation スタックの作成に使用できる AWS CloudFormation テンプレートがダウンロードされます。このスタックは、Fortigate CNF リソースを管理するためのクロスアカウントアクセス許可を Firewall Manager に付与する AWS Identity and Access Management ロールを作成します。スタックの詳細については、「AWS CloudFormation ユーザーガイド」の「StackSets の操作」を参照してください。スタックを作成するには、Fortigate CNF ポータルのアカウント ID が必要です。
16. [次へ] をクリックします。
17. ポリシータグ には、Firewall Manager ポリシーリソースに追加する識別タグを追加します。タグの詳細については、「タグエディタの使用」を参照してください。
18. [Next] (次へ) を選択します。
19. 新しいポリシー設定を確認し、調整が必要なページに戻ります。

    [Policy actions] (ポリシーアクション) が [Identify resources that don't comply with the policy rules, but don't auto remediate] (ポリシールールに準拠していないリソースを特定するが、自動修復しない) に設定されていることを確認します。これにより、ポリシーを有効にする前に、ポリシーが行う変更を確認できます。
20. ポリシーが完成したら、[Create policy] (ポリシーの作成) を選択します。

    [AWS Firewall Manager ポリシー] ペインにポリシーが一覧表示されます。アカウントの見出しの下には「保留中」と表示され、自動修復設定のステータスを示します。ポリシーの作成には数分かかることがあります。[Pending] (保留中) ステータスがアカウント数に置き換えられたら、ポリシー名を選択して、アカウントとリソースの準拠ステータスを調べることができます。詳細については、「AWS Firewall Manager ポリシーのコンプライアンス情報の表示」を参照してください。

Firewall Manager Fortigate CNF ポリシーの詳細については、「Fortigate Cloud Native Firewall (CNF) as a Service ポリシー」を参照してください。

AWS Firewall Manager ポリシーの使用

AWS Firewall Manager には、次のタイプのポリシーが用意されています。ポリシータイプごとに、を定義します。

- AWS WAF policy – Firewall Manager は、 AWS WAF および AWS WAF Classic ポリシーをサポートしています。どちらのバージョンでも、ポリシーによって保護されるリソースを定義します。
  - AWS WAF ポリシータイプは、ウェブ ACL で最初と最後に実行するルールグループのセットを受け取ります。次に、ウェブ ACL を適用するアカウントで、アカウント所有者は 2 つのセット間で実行するルールとルールグループを追加できます。
  - AWS WAF Classic ポリシータイプは、ウェブ ACL で単一のルールグループを実行します。
- Shield Advanced ポリシー – このポリシータイプは、指定したリソースタイプに対して組織全体で Shield Advanced 保護を適用します。
- Amazon VPC セキュリティグループポリシー – このポリシータイプでは、組織全体で使用されているセキュリティグループを制御し、組織全体でベースラインのルールセットを適用できます。
- Amazon VPC ネットワークアクセスコントロールリスト (ACL) ポリシー – このポリシータイプでは、組織全体で使用されているネットワーク ACLs を制御し、組織全体でネットワーク ACLsのベースラインセットを適用できます。
- Network Firewall ポリシー — このポリシータイプは、組織の VPC に AWS Network Firewall 保護を適用します。 VPCs
- Amazon Route 53 Resolver DNS Firewall ポリシー – このポリシーは、DNS Firewall 保護を組織の VPC に適用します。
- サードパーティーのファイアウォールポリシー – このポリシータイプは、サードパーティーのファイアウォール保護を適用します。サードパーティーのファイアウォールは、 AWS Marketplace コンソールの AWS Marketplace コンソールからサブスクリプションで利用できます。
  - Palo Alto Networks Cloud NGFW ポリシー – このポリシータイプは、Palo Alto Networks Cloud Next Generation Firewall (NGFW) 保護と Palo Alto Networks Cloud NGFW ルールスタックを組織の VPCs に適用します。
  - Fortigate Cloud Native Firewall (CNF) as a Service ポリシー – このポリシータイプは、Fortigate Cloud Native Firewall (CNF) をサービス保護として適用します。Fortigate CNF は、ゼロデイ脅威をブロックし、業界をリードする高度な脅威防止、スマートなウェブアプリケーションファイアウォール (WAF) 、API によってクラウドインフラストラクチャを保護するクラウド中心のソリューションです。

Firewall Manager ポリシーは、個々のポリシータイプに固有です。アカウント間で複数のポリシータイプを適用する場合は、複数のポリシーを作成できます。タイプごとに複数のポリシーを作成できます。

で作成した組織に新しいアカウントを追加すると AWS Organizations、Firewall Manager はポリシーの範囲内にあるそのアカウントのリソースにポリシーを自動的に適用します。

AWS Firewall Manager ポリシーの一般的な設定

AWS Firewall Manager マネージドポリシーには、いくつかの一般的な設定と動作があります。すべてについて、名前を指定してポリシーの範囲を定義し、リソースのタグ付けを使用してポリシーの範囲を制御できます。修復処置を行わずに準拠していないアカウントとリソースを表示するか、非準拠リソースを自動的に修復するかを選択できます。

ポリシーの範囲については、「AWS Firewall Manager ポリシーの範囲」を参照してください。

AWS Firewall Manager ポリシーの作成

ポリシーを作成するステップは、ポリシータイプによって異なります。必ず、必要なポリシーのタイプに対応する手順を使用してください。

> **Important**
>
> AWS Firewall Manager は Amazon Route 53 または をサポートしていません AWS Global Accelerator。Shield Advanced を使用してこれらのリソースを保護する場合、Firewall Manager ポリシーは使用できません。代わりに、「AWS Shield AdvancedAWS リソースへの保護の追加」の手順に従ってください。

トピック

- の AWS Firewall Manager ポリシーの作成 AWS WAF
- AWS WAF Classic の AWS Firewall Manager ポリシーの作成
- の AWS Firewall Manager ポリシーの作成 AWS Shield Advanced
- AWS Firewall Manager 共通セキュリティグループポリシーの作成
- AWS Firewall Manager コンテンツ監査セキュリティグループポリシーの作成
- AWS Firewall Manager 使用状況監査セキュリティグループポリシーの作成
- ネットワーク ACL ポリシーの作成 AWS Firewall Manager

- の AWS Firewall Manager ポリシーの作成 AWS Network Firewall
- Amazon Route 53 Resolver DNS Firewall の AWS Firewall Manager ポリシーの作成
- Palo Alto Networks Cloud NGFW の AWS Firewall Manager ポリシーの作成
- Fortigate Cloud Native Firewall (CNF) as a Service の AWS Firewall Manager ポリシーの作成

の AWS Firewall Manager ポリシーの作成 AWS WAF

Firewall Manager AWS WAF ポリシーでは、 AWS と AWS Marketplace 販売者がユーザーに代わって作成および管理するマネージドルールグループを使用できます。独自のルールグループを作成して使用することもできます。ルールグループの詳細については、「AWS WAF ルールグループ」を参照してください。

独自のルールグループを使用する場合は、Firewall Manager AWS WAF ポリシーを作成する前にそれらのグループを作成します。ガイダンスについては、「独自のルールグループの管理」を参照してください。個々のカスタムルールを使用するには、独自のルールグループを定義し、その中にルールを定義してから、ポリシーでそのルールグループを使用する必要があります。

Firewall Manager AWS WAF ポリシーの詳細については、「」を参照してくださいAWS WAF ポリシー。

の Firewall Manager ポリシーを作成するには AWS WAF （ コンソール ）

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. [Create policy] (ポリシーの作成) を選択します。
4. [Policy type] (ポリシータイプ) では [AWS WAF] を選択します。
5. リージョン で、 を選択します AWS リージョン。Amazon CloudFront ディストリビューションを保護するには、グローバル を選択します。

複数のリージョン (CloudFront ディストリビューション以外) のリソースを保護するには、リージョンごとに個別の Firewall Manager ポリシーを作成する必要があります。

6. [Next] (次へ) を選択します。
7. [Policy name] (ポリシー名) で、わかりやすい名前を入力します。Firewall Manager は、管理するウェブ ACL の名前にポリシー名を含めます。ウェブ ACL 名の後に、`FMManagedWebACLV2-`、ここに入力するポリシー名、`-`、およびウェブ ACL 作成タイムスタンプ (UTC ミリ秒) が続きます。例えば、`FMManagedWebACLV2-MyWAFPolicyName-1621880374078` です。
8. [ウェブリクエストボディの検査] では、オプションで本文のサイズ制限を変更してください。価格に関する考慮事項など、ボディ検査のサイズ制限については、「AWS WAF デベロッパーガイド」の「本文検査のサイズ制限の管理」を参照してください。
9. ポリシールール で、ウェブ ACL で最初と最後の AWS WAF 評価を行うルールグループを追加します。 AWS WAF マネージドルールグループのバージョニングを使用するには、バージョニングを有効にする に切り替えます。各アカウントマネージャーは、最初のルールグループと最後のルールグループの間にルールとルールグループを追加できます。の Firewall Manager ポリシーで AWS WAF ルールグループを使用する方法の詳細については AWS WAF、「」を参照してくださいAWS WAF ポリシー。

   (オプション)ウェブ ACL によるルールグループの使用方法をカスタマイズするには、[編集] を選択します。一般的なカスタマイズ設定は次のとおりです。

   - マネージドルールのグループの場合は、一部またはすべてのルールのルールアクションをオーバーライドします。ルールにオーバーライドアクションを定義しない場合、評価にはルールグループ内で定義されているルールアクションが使用されます。このオプションの詳細については、「AWS WAF デベロッパーガイド」の「ルールグループのアクションオーバーライドオプション」を参照してください。
   - 一部のマネージドルールグループは追加の設定が必要です。マネージドルールグループのプロバイダーのドキュメントを参照してください。 AWS マネージドルールのルールグループに固有の情報については、「 AWS WAF デベロッパーガイドAWS のマネージドルール AWS WAF」の「」を参照してください。

   設定が完了したら、[Save rule] (ルールを保存) を選択します。
10. ウェブ ACL のデフォルトアクションを設定します。これは、ウェブリクエストがウェブ ACL のルールのいずれにも一致しない場合に AWS WAF が実行するアクションです。[Allow] (許可) アクションでカスタムヘッダーを追加することや、[Block] (ブロック) アクションのカスタムレ

スポンスを追加することができます。デフォルトのウェブ ACL アクションの詳細については、「ウェブ ACL のデフォルトアクション」を参照してください。カスタムウェブリクエストを設定する方法については、「AWS WAFのカスタマイズされたウェブリクエストとレスポンス」を参照してください。

11. [ログ記録設定] で、[ログ記録を有効にする] を選択してログ記録をオンにします。ログ記録は、ウェブ ACL で分析されるトラフィックに関する詳細情報を提供します。[ログの出力先] を選択し、設定したログ記録の送信先を選択します。名前が `aws-waf-logs-` で始まるログ記録先を選択する必要があります。 AWS WAF ログ記録の送信先の設定については、「」を参照してくださいAWS WAF ポリシーのログ記録の設定。
12. (オプション) 特定のフィールドとその値がログに含まれることを希望しない場合には、このフィールドをマスキングします。マスキングするフィールドを選び、[Add] (追加) を選択します。必要に応じて手順を繰り返し、追加のフィールドをマスキングします。マスキングされたフィールドは、ログに `REDACTED` と表示されます。例えば、[URI] フィールドをマスキングすると、ログの [URI] フィールドは `REDACTED` となります。
13. (オプション) すべてのリクエストをログに送信しない場合は、フィルタリング条件と動作を追加します。[Filter logs] (ログをフィルタリング) で、適用する各フィルターについて [Add filter] (フィルターを追加) を選択し、次にフィルター基準を選択して、基準に一致するリクエストを保持するかドロップするかを指定します。フィルターの追加が完了したら、必要に応じて、[Default logging behavior] (デフォルトのログ記録動作) を変更します。詳細については、「AWS WAF デベロッパーガイド」の 「ウェブ ACL ログ記録設定」を参照してください。
14. [Token domain list] (トークンドメインリスト) を定義して、保護されたアプリケーション間でトークンの共有を有効にできます。トークンは、 CAPTCHAアクションと Challengeアクション、および AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) と AWS WAF Bot Control に AWS Managed Rules ルールグループを使用するときに実装するアプリケーション統合 SDKs によって使用されます。

    パブリックサフィックスは許可されません。たとえば、`gov.au` または `co.uk` をトークンドメインとして使用することはできません。

    デフォルトでは、 は保護されたリソースのドメインに対してのみトークン AWS WAF を受け入れます。このリストにトークンドメインを追加すると、 はリスト内のすべてのドメインと、関連付けられたリソースのドメインのトークン AWS WAF を受け入れます。詳細については、「AWS WAF デベロッパーガイド」の 「AWS WAF ウェブ ACL トークンドメインリストの設定」を参照してください。

    ウェブ ACL の CAPTCHA およびチャレンジのイミュニティ時間を変更できるのは、既存のウェブ ACL を編集するときのみです。これらの設定は、Firewall Manager の [ポリシーの詳

細] ページで確認できます。これらの設定については、「タイムスタンプの有効期限： AWS WAF トークンのイミュニティ時間」を参照してください。既存のポリシー内で [関連付けの設定]、[CAPTCHA]、[チャレンジ]、または [トークンのドメインリスト] 設定を更新すると、Firewall Manager はローカルのウェブ ACL を新しい値で上書きします。ポリシーの [関連付けの設定]、[CAPTCHA]、[チャレンジ]、または [トークンのドメインリスト] 設定を更新していない場合は、ローカルのウェブ ACL の値は変更されません。このオプションの詳細については、「AWS WAF デベロッパーガイド」の「CAPTCHAChallengeの および AWS WAF」を参照してください。

15. 関連付けられていないウェブ ACL を Firewall Manager に管理させたい場合は、[ウェブ ACL 管理]で、[関連付けられていないウェブ ACL の管理]を有効にしてください。このオプションを使用すると、Firewall Manager は、少なくとも 1 つのリソースがウェブ ACL を使用する場合のみ、ポリシー範囲内のアカウントにウェブ ACL を作成します。アカウントがポリシーの対象になるといつでも、少なくとも 1 つのリソースがウェブ ACL を使用する場合に、Firewall Manager はアカウントにウェブ ACL を自動的に作成します。このオプションを有効にすると、Firewall Manager はアカウント内の関連付けられていないウェブ ACL に 1 回だけクリーンアップを実行します。このクリーンアッププロセスには、数時間かかることがあります。Firewall Manager がウェブ ACL を作成した後、リソースがポリシー範囲から外れた場合、Firewall Manager はそのリソースとウェブ ACL の関連付けを解除しますが、関連付けられていないウェブ ACL はクリーンアップしません。Firewall Manager は、関連付けられていないウェブ ACL の管理を、ポリシーで初めて有効にした場合のみ、関連付けられていないウェブ ACL をクリーンアップします。

16. 組織内の該当する各アカウントにウェブ ACL を作成したいが、まだリソースにウェブ ACL を適用しない場合は、[ポリシーアクション] で、[ポリシールールに準拠していないリソースを特定するが、自動修復しない] を選択し、[関連付けられていないウェブ ACL の管理] を選択します。これらのオプションは後で変更できます。

    代わりに、ポリシーを既存の範囲内のリソースに自動的に適用する場合は、[Auto remediate any noncompliant resources] (準拠していないリソースを自動修復する) を選択します。[関連付けられていないウェブ ACL の管理]が無効になっている場合、[準拠していないリソースを自動修復する]のオプションで、組織内の該当する各アカウントにウェブ ACL を作成し、そのウェブ ACL をアカウント内のリソースに関連付けます。[関連付けられていないウェブ ACL の管理]が有効になっている場合は、[準拠していないリソースを自動修復する]のオプションは、ウェブ ACL への関連付けの対象になるリソースを持つアカウントでのみウェブ ACL を作成して関連付けます。

    [Auto remediate any noncompliant resources] (準拠していないリソースを自動修復) を選択すると、別のアクティブな Firewall Manager ポリシーによって管理されていないウェブ ACL に対し

て、範囲内のリソースから既存のウェブ ACL の関連付けを削除することもできます。このオプションを選択した場合、Firewall Manager は、まずポリシーのウェブ ACL をリソースに関連付けてから、以前の関連付けを削除します。リソースに、別のアクティブな Firewall Manager ポリシーによって管理されている別のウェブ ACL との関連付けがある場合、この選択はその関連付けには影響しません。

17. [Next] (次へ) を選択します。
18. [AWS アカウント this policy applies to] (このポリシーが適用される) で、次のようにオプションを選択します。

    - 組織内のすべてのアカウントにポリシーを適用する場合は、デフォルトの選択のままにし、AWS 組織 のすべてのアカウントを含めます。
    - ポリシーを特定の AWS Organizations 組織単位 (OUs) 内の特定のアカウントにのみ適用する場合は、指定したアカウントと組織単位のみを含める を選択し、含めるアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。
    - 特定のアカウントや AWS Organizations の組織単位 (OU) 以外のすべてにポリシーを適用する場合は、[Exclude the specified accounts and organizational units, and include all others] (指定されたアカウントと組織単位を除外し、他のすべてを含める) を選択して、除外するアカウントと OU を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。

    選択できるオプションは 1 つのみです。

    ポリシーを適用すると、Firewall Manager は新しいアカウントを設定と照合して自動的に評価します。例えば、特定のアカウントのみを含めると、Firewall Manager は新しいアカウントにポリシーを適用しません。別の例として、OU を含めた場合、OU またはその子である OU にアカウントを追加すると、Firewall Manager は新しいアカウントにポリシーを自動的に適用します。
19. [Resource type] (リソースタイプ) で、保護するリソースのタイプを選択します。
20. リソース では、指定したタグでリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

    複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。

21. [次へ] をクリックします。
22. ポリシータグ には、Firewall Manager ポリシーリソースに追加する識別タグを追加します。タグの詳細については、「タグエディタの使用」を参照してください。
23. [Next] (次へ) を選択します。
24. 新しいポリシー設定を確認し、調整が必要なページに戻ります。

ポリシーが完成したら、[Create policy] (ポリシーの作成) を選択します。[AWS Firewall Manager ポリシー] ペインにポリシーが一覧表示されます。おそらく、アカウントの見出しの下に「保留中」と表示され、自動修復設定のステータスを示します。ポリシーの作成には数分かかることがあります。[Pending] (保留中) ステータスがアカウント数に置き換えられたら、ポリシー名を選択して、アカウントとリソースの準拠ステータスを調べることができます。詳細については、「AWS Firewall Manager ポリシーのコンプライアンス情報の表示」を参照してください。

AWS WAF Classic の AWS Firewall Manager ポリシーの作成

AWS WAF Classic の Firewall Manager ポリシーを作成するには (コンソール)

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

> **Note**
>
> Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. [Create policy] (ポリシーの作成) を選択します。
4. [Policy type (ポリシータイプ)] で、[AWS WAF Classic] を選択します。

5. ポリシーに追加する AWS WAF Classic ルールグループを既に作成している場合は、 AWS Firewall Manager ポリシーの作成 を選択し、既存のルールグループ を追加します。新しいルールグループを作成する場合は、[Create a Firewall Manager policy and add a new rule group] (Firewall Manager ポリシーを作成して新しいルールグループを追加する) を選択します。
6. リージョン で、 を選択します AWS リージョン。Amazon CloudFront リソースを保護するには、グローバル を選択します。

   複数のリージョン (リソース以外) の CloudFront リソースを保護するには、リージョンごとに個別の Firewall Manager ポリシーを作成する必要があります。
7. [Next] (次へ) を選択します。
8. ルールグループを作成する場合は、AWS WAF クラシックルールグループの作成 の手順に従います。ルールグループを作成したら、次のステップに進みます。
9. ポリシー名を入力します。
10. 既存のルールグループを追加する場合は、ドロップダウンメニューを使用して追加するルールグループを選択し、[Add rule group] (ルールグループの追加) を選択します。
11. ポリシーには、[Action set by rule group] (ルールグループによって設定されたアクション) と [Count] (カウント) の 2 つのアクションがあります。ポリシーをテストする場合は、アクションを [Count] (カウント) に設定します。このアクションは、ルールグループのルールで指定されたブロックアクションを上書きします。つまり、ポリシーのアクションが [Count] (カウント) に設定されている場合、リクエストはカウントされ、ブロックされません。逆に、ポリシーのアクションを [Action set by rule group] (ルールグループによって設定されたアクション) に設定すると、ルールグループルールのアクションが使用されます。適切なアクションを選択します。
12. [Next] (次へ) を選択します。
13. [AWS アカウント this policy applies to] (このポリシーが適用される) で、次のようにオプションを選択します。

    - 組織内のすべてのアカウントにポリシーを適用する場合は、デフォルトの選択のままにし、AWS 組織 のすべてのアカウントを含めます。
    - 特定の AWS Organizations 組織単位 (OUs) にある特定のアカウントまたはアカウントにのみポリシーを適用する場合は、指定されたアカウントと組織単位のみを含める を選択し、含めるアカウントと OUsを追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。
    - 特定のアカウントまたは AWS Organizations 組織単位 (OUs) 以外のすべてのアカウントにポリシーを適用する場合は、指定したアカウントと組織単位を除外し、その他すべての を含め

て、除外するアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。

選択できるオプションは 1 つのみです。

ポリシーを適用すると、Firewall Manager は新しいアカウントを設定と照合して自動的に評価します。例えば、特定のアカウントのみを含めると、Firewall Manager は新しいアカウントにポリシーを適用しません。別の例として、OU を含めた場合、OU またはその子である OU にアカウントを追加すると、Firewall Manager は新しいアカウントにポリシーを自動的に適用します。

14. 保護するリソースのタイプを選択します。
15. リソース では、指定したタグを持つリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

    複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

    リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。
16. 既存のリソースにポリシーを自動的に適用する場合は、[Create and apply this policy to existing and new resources] (既存および新規のリソースにこのポリシーを作成して適用する) を選択します。

    このオプションは、 AWS 組織内の各関連アカウントにウェブ ACL を作成し、アカウント内のリソースにウェブ ACL を関連付けます。このオプションは、前述の基準 (リソースタイプとタグ) に一致するすべての新しいリソースにもポリシーを適用します。また、[Create policy but do not apply the policy to existing or new resources] (ポリシーを作成するが既存および新規のリソースにポリシーを適用しない) を選択する場合は、Firewall Manager により組織内の各関連アカウントにウェブ ACL が作成されますが、ウェブ ACL はいずれのリソースにも適用されません。ポリシーは後でリソースに適用する必要があります。適切なオプションを選択します。
17. [Replace existing associated web ACLs] (既存の関連付けられたウェブ ACL を置換) では、範囲内のリソースに対して現在定義されているウェブ ACL の関連付けをすべて削除し、このポリシーで作成しているウェブ ACL への関連付けに置き換えることができます。デフォルトでは、Firewall Manager は、新しいウェブ ACL の関連付けを追加する前に既存のウェブ ACL の関連付けを削除しません。既存の関連付けを削除する場合は、このオプションを選択します。

18. [Next] (次へ) を選択します。
19. 新しいポリシーを確認します。設定を変更するには、[Edit] (編集) を選択します。ポリシーが完成したら、[Create and apply policy] (ポリシーの作成と適用) を選択します。

の AWS Firewall Manager ポリシーの作成 AWS Shield Advanced

Shield Advanced の Firewall Manager ポリシーを作成するには (コンソール)

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. [Create policy] (ポリシーの作成) を選択します。
4. [Policy type] (ポリシータイプ) で、[Shield Advanced] を選択します。

   Shield Advanced ポリシーを作成するには、Shield Advanced をサブスクライブする必要があります。登録されていない場合は、登録するよう求められます。サブスクリプションの費用については、「AWS Shield Advanced の料金」を参照してください。
5. リージョン で、 を選択します AWS リージョン。Amazon CloudFront ディストリビューションを保護するには、グローバル を選択します。

   [Global] (グローバル) 以外のリージョンを選択する場合、複数のリージョンでリソースを保護するには、各リージョン用に個別の Firewall Manager ポリシーを作成する必要があります。
6. [Next] (次へ) を選択します。
7. [Name] (名前) で、わかりやすい名前を入力します。
8. [Global] (グローバル) リージョンポリシーの場合のみ、Shield Advanced アプリケーションレイヤー DDoS 自動緩和を管理するかどうかを選択できます。Shield Advanced 機能については、「Shield Advanced アプリケーションレイヤー DDoS 自動緩和」を参照してください。

自動緩和を有効または無効にすることを選択でき、あるいは無視することを選択できます。無視することを選択した場合、Firewall Manager は、Shield Advanced 保護のために自動緩和をまったく管理しません。これらのポリシーのオプションの詳細については、「アプリケーションレイヤー DDoS 自動緩和」を参照してください 。

9. 関連付けられていないウェブ ACL を Firewall Manager に管理させたい場合は、[ウェブ ACL 管理]で、[関連付けられていないウェブ ACL の管理]を有効にしてください。このオプションを使用すると、Firewall Manager は、少なくとも 1 つのリソースがウェブ ACL を使用する場合のみ、ポリシー範囲内のアカウントにウェブ ACL を作成します。アカウントがポリシーの対象になるといつでも、少なくとも 1 つのリソースがウェブ ACL を使用する場合に、Firewall Manager はアカウントにウェブ ACL を自動的に作成します。このオプションを有効にすると、Firewall Manager はアカウント内の関連付けられていないウェブ ACL に 1 回だけクリーンアップを実行します。このクリーンアッププロセスには、数時間かかることがあります。Firewall Manager がウェブ ACL を作成した後、リソースがポリシーの範囲から外れても、Firewall Manager はそのリソースとウェブ ACL との関連付けを解除しません。ウェブ ACL を 1 回限りのクリーンアップに含めるには、まずリソースとウェブ ACL の関連付けを手動で解除してから、[関連付けられていないウェブ ACL の管理]を有効にする必要があります。

10. [Policy action] (ポリシーアクション) では、準拠していないリソースを自動的に修復しないオプションを使用してポリシーを作成することをお勧めします。自動修復を無効にすると、新しいポリシーを適用する前にその効果を評価できます。変更が適切であることを確認したら、ポリシーを編集し、ポリシーアクションを変更して、自動修復を有効にします。

    代わりに、ポリシーを既存の範囲内のリソースに自動的に適用する場合は、[Auto remediate any noncompliant resources] (準拠していないリソースを自動修復する) を選択します。このオプションは、 AWS 組織内の該当する各アカウントとアカウント内の該当する各リソースに Shield Advanced 保護を適用します。

    グローバルリージョンポリシーでのみ、非準拠のリソース を自動修正することを選択した場合、Firewall Manager で既存の AWS WAF Classic ウェブ ACL 関連付けを、最新バージョン AWS WAF (v2) を使用して作成されたウェブ ACLsへの新しい関連付けに自動的に置き換えることもできます。これを選択すると、Firewall Manager は、ポリシー用にまだウェブ ACL を持っていない範囲内のアカウントに新しい空のウェブ ACL を作成した後、以前のバージョンのウェブ ACL との関連付けを削除し、最新バージョンのウェブ ACL との新しい関連付けを作成します。このオプションの詳細については、「AWS WAF クラシックウェブ ACL を最新バージョンのウェブ ACL に置き換えます。」をご参照ください。

11. [Next] (次へ) を選択します。

12. [AWS アカウント this policy applies to] (このポリシーが適用される) で、次のようにオプションを選択します。

    - 組織内のすべてのアカウントにポリシーを適用する場合は、デフォルトの [Include all accounts under my AWS organization] (自分の組織の下にあるすべてのアカウントを含める) を選択したままにします。
    - ポリシーを特定の AWS Organizations 組織単位 (OUs) にある特定のアカウントにのみ適用する場合は、指定したアカウントと組織単位のみを含める を選択し、含めるアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。
    - 特定のアカウントまたは AWS Organizations 組織単位 (OUs) を除くすべてのアカウントにポリシーを適用する場合は、指定したアカウントと組織単位を除外し、その他すべての を含めて、除外するアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。

    選択できるオプションは 1 つのみです。

    ポリシーを適用すると、Firewall Manager は新しいアカウントを設定と照合して自動的に評価します。例えば、特定のアカウントのみを含めると、Firewall Manager は新しいアカウントにポリシーを適用しません。別の例として、OU を含めた場合、OU またはその子である OU にアカウントを追加すると、Firewall Manager は新しいアカウントにポリシーを自動的に適用します。

13. 保護するリソースのタイプを選択します。

    Firewall Manager は、Amazon Route 53 または AWS Global Acceleratorをサポートしていません。これらのサービスからリソースを保護するために Shield Advanced を使用する必要がある場合、Firewall Manager ポリシーを使用することはできません。代わりに、「AWS Shield AdvancedAWS リソースへの保護の追加」の Shield Advanced のガイダンスに従ってください。

14. リソース では、指定したタグでリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

    複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。

15. [次へ] をクリックします。
16. ポリシータグ には、Firewall Manager ポリシーリソースに追加する識別タグを追加します。タグの詳細については、「タグエディタの使用」を参照してください。
17. [Next] (次へ) を選択します。
18. 新しいポリシー設定を確認し、調整が必要なページに戻ります。

    ポリシーが完成したら、[Create policy] (ポリシーの作成) を選択します。[AWS Firewall Manager ポリシー] ペインにポリシーが一覧表示されます。おそらく、アカウントの見出しの下に「保留中」と表示され、自動修復設定のステータスを示します。ポリシーの作成には数分かかることがあります。[Pending] (保留中) ステータスがアカウント数に置き換えられたら、ポリシー名を選択して、アカウントとリソースの準拠ステータスを調べることができます。詳細については、「AWS Firewall Manager ポリシーのコンプライアンス情報の表示」を参照してください。

AWS Firewall Manager 共通セキュリティグループポリシーの作成

共通セキュリティグループポリシーの仕組みの詳細については、「共通セキュリティグループポリシー」を参照してください。

共通セキュリティグループポリシーを作成するには、ポリシーのプライマリとして使用するセキュリティグループが Firewall Manager 管理者アカウントに既に作成されている必要があります。セキュリティグループは、Amazon Virtual Private Cloud (Amazon VPC) または Amazon Elastic Compute Cloud (Amazon EC2) を通じて管理できます。詳細については、「Amazon VPC ユーザーガイド」の「セキュリティグループの操作」を参照してください。

共通セキュリティグループポリシー (コンソール) を作成するには

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

> **Note**
>
> Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. [Create policy] (ポリシーの作成) を選択します。
4. [Policy type] (ポリシータイプ) で、[Security group] (セキュリティグループ) を選択します。
5. [Security group policy type] (セキュリティグループポリシータイプ) で、[Common security groups] (共通セキュリティグループ) を選択します。
6. リージョン で、 を選択します AWS リージョン。
7. [Next] (次へ) を選択します。
8. [Policy name](ポリシー名) で、フレンドリ名を入力します。
9. [Policy rules] (ポリシールール) で、次の操作を行います。

   a. ルールオプションから、セキュリティグループルールとポリシーの範囲内にあるリソースに対して適用する制限を選択します。[Distribute tags from the primary security group to the security groups created by this policy] (プライマリセキュリティグループからこのポリシーによって作成されたセキュリティグループにタグを配布) を選択した場合は、[Identify and report when the security groups created by this policy become non-compliant] (このポリシーによって作成されたセキュリティグループが非準拠になったときに識別して報告) も選択する必要があります。

   > **Important**
   >
   > Firewall Manager は、 AWS サービスによって追加されたシステムタグをレプリカセキュリティグループに配布しません。システムタグは `aws:` プレフィックスで始まります。また、ポリシーに組織のタグポリシーと矛盾するタグがある場合は、Firewall Manager が既存のセキュリティグループでのタグ更新や、新しいセキュリティグループの作成を行うことはありません。タグポリシーの詳細については、「 ユーザーガイド」の「タグポリシー AWS Organizations 」を参照してください。

[プライマリセキュリティグループからのセキュリティグループの参照をこのポリシーによって作成されたセキュリティグループに配布する] を選択した場合、Firewall Manager は Amazon VPC にアクティブなピア接続がある場合にのみセキュリティグループの参照を配布します。このオプションの詳細については、「ポリシールールの設定」を参照してください。

b. プライマリセキュリティグループ で、セキュリティグループの追加 を選択し、使用するセキュリティグループを選択します。Firewall Manager は、Firewall Manager 管理者アカウントのすべての Amazon VPC インスタンスのセキュリティグループのリストを入力します。

   デフォルトでは、ポリシーあたりのプライマリセキュリティグループの最大数は 3 です。この設定についての情報は、「AWS Firewall Manager クォータ」を参照してください。

c. [Policy action] (ポリシーアクション) では、自動的に修復されないオプションを使用してポリシーを作成することをお勧めします。これにより、新しいポリシーを適用する前にその効果を評価できます。変更が適切であることを確認したら、ポリシーを編集し、ポリシーアクションを変更して、準拠していないリソースの自動修復を有効にします。

10. [Next] (次へ) を選択します。
11. [AWS アカウント this policy applies to] (このポリシーが適用される) で、次のようにオプションを選択します。

    - 組織内のすべてのアカウントにポリシーを適用する場合は、デフォルトの選択のままにし、AWS 組織 のすべてのアカウントを含めます。
    - ポリシーを特定の AWS Organizations 組織単位 (OUs) 内の特定のアカウントにのみ適用する場合は、指定したアカウントと組織単位のみを含める を選択し、含めるアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。
    - 特定のアカウントまたは AWS Organizations 組織単位 (OUs) 以外のすべてのアカウントにポリシーを適用する場合は、指定したアカウントと組織単位を除外し、その他すべての を含めて、除外するアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。

    選択できるオプションは 1 つのみです。

    ポリシーを適用すると、Firewall Manager は新しいアカウントを設定と照合して自動的に評価します。例えば、特定のアカウントのみを含めると、Firewall Manager は新しいアカウントにポリ

シーを適用しません。別の例として、OU を含めた場合、OU またはその子である OU にアカウントを追加すると、Firewall Manager は新しいアカウントにポリシーを自動的に適用します。

12. [Resource type] (リソースタイプ) で、保護するリソースのタイプを選択します。

    [EC2 instance] (EC2 インスタンス) を選択した場合は、各 Amazon EC2 インスタンスのすべての Elastic Network Interface を含めるか、デフォルトインターフェイスのみを含めるかを選択できます。範囲内の Amazon EC2 インスタンスに複数の Elastic Network Interface がある場合、すべてのインターフェイスを含めるオプションを選択すると、Firewall Manager はすべてのインターフェイスにポリシーを適用できます。自動修復を有効にすると、Firewall Manager が Amazon EC2 インスタンス内の一部の Elastic Network Interface にポリシーを適用できない場合、そのインスタンスは非準拠としてマークされます。

13. リソース では、指定したタグでリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

    複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

    リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。

14. [Shared VPC resources] (共有 VPC リソース) の場合、アカウントが所有する VPC に加えて、共有 VPC 内のリソースにポリシーを適用する場合は、[Include resources from shared VPCs] (共有 VPC からのリソースを含める) を選択します。

15. [Next] (次へ) を選択します。

16. ポリシー設定を見直して目的の設定になっていることを確認し、[Create policy] (ポリシーの作成) を選択します。

Firewall Manager は、範囲内のアカウントに含まれるすべての Amazon VPC インスタンスに、アカウントごとにサポートされる Amazon VPC の最大クォータまで、プライマリセキュリティグループのレプリカを作成します。Firewall Manager は、レプリカセキュリティグループを、範囲内の各アカウント用のポリシーの範囲内にあるリソースに関連付けます。このポリシーの仕組みの詳細については、「共通セキュリティグループポリシー」を参照してください。

AWS Firewall Manager コンテンツ監査セキュリティグループポリシーの作成

コンテンツ監査セキュリティグループポリシーの仕組みの詳細については、「コンテンツ監査セキュリティグループポリシー」を参照してください。

コンテンツ監査ポリシーの設定によっては、Firewall Manager がテンプレートとして使用するための監査セキュリティグループを指定する必要があります。例えば、どのセキュリティグループでも許可しないすべてのルールを含む監査セキュリティグループがあるとします。ポリシーでこれらの監査セキュリティグループを使用するには、Firewall Manager 管理者アカウントを使用してこれらの監査セキュリティグループを作成する必要があります。セキュリティグループは、Amazon Virtual Private Cloud (Amazon VPC) または Amazon Elastic Compute Cloud (Amazon EC2) を通じて管理できます。詳細については、「Amazon VPC ユーザーガイド」の「セキュリティグループの操作」を参照してください。

コンテンツ監査セキュリティグループポリシーを作成するには (コンソール)

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. [Create policy] (ポリシーの作成) を選択します。
4. [Policy type] (ポリシータイプ) で、[Security group] (セキュリティグループ) を選択します。
5. [Security group policy type] (セキュリティグループポリシータイプ) で、[Auditing and enforcement of security group rules] (セキュリティグループルールの監査と適用) を選択します。
6. リージョン で、 を選択します AWS リージョン。
7. [Next] (次へ) を選択します。
8. [Policy name](ポリシー名) で、フレンドリ名を入力します。
9. [Policy rules] (ポリシールール) で、使用するマネージドポリシールールオプションまたはカスタムポリシールールオプションを選択します。

a. [Configure managed audit policy rules] (マネージド監査ポリシールールを設定) で、次の手順を実行します。

   i. [Configure security group rules to audit] (監査するセキュリティグループルールを設定) で、監査ポリシーを適用するセキュリティグループルールの種類を選択します。

   ii. セキュリティグループ内のプロトコル、ポート、CIDR 範囲設定に基づく監査ルールなどを実行する場合は、[過度に許容されるセキュリティグループルールを監査] を選択し、必要なオプションを選択します。

   [ルールですべてのトラフィックを許可する] を選択すると、カスタムアプリケーションリストを指定して、監査するアプリケーションを指定できます。カスタムアプリケーションリスト、およびポリシーでのアプリケーションリストの使用方法については、「マネージドリスト」および「マネージドリストの使用」を参照してください。

   プロトコルリストを使用する選択では、既存のリストを使用したり、新しいリストを作成したりできます。プロトコルリスト、およびポリシーでのアプリケーションリストの使用方法については、「マネージドリスト」および「マネージドリストの使用」を参照してください。

   iii. 予約済みまたは予約されていない CIDR 範囲へのアクセスに基づいて高リスクを監査する場合は、[高リスクアプリケーションを監査する] を選択し、必要なオプションを選択します。

   [ローカル CIDR 範囲のみにアクセスできるアプリケーション] と [パブリック CIDR 範囲を使用できるアプリケーション] の選択は相互に排他的です。いずれのポリシーでも、選択できるのは最大 1 つです。

   アプリケーションリストを使用する選択では、既存のリストを使用したり、新しいリストを作成したりできます。アプリケーションリスト、およびポリシーでのアプリケーションリストの使用方法については、「マネージドリスト」および「マネージドリストの使用」を参照してください。

   iv. [Overrides] (上書き) 設定を使用して、ポリシー内の他の設定を明示的に上書きします。ポリシーに設定した他のオプションに準拠しているかどうかにかかわらず、特定のセキュリティグループルールを常に許可するか常に拒否するかを選択できます。

   このオプションでは、許可されたルールまたは拒否されたルールテンプレートとして監査セキュリティグループを指定します。[Audit security groups] (監査セキュリティグループ) で、[Add audit security groups] (監査セキュリティグループを追加) を選択し

てから、使用するセキュリティグループを選択します。Firewall Manager は、Firewall Manager 管理者アカウント内におけるすべての Amazon VPC インスタンスからの監査セキュリティグループのリストを設定します。ポリシーの監査セキュリティグループ数のデフォルト最大クォータは 1 です。クォータを引き上げる方法については、「AWS Firewall Manager クォータ」を参照してください。

b. [Configure custom policy rules] (カスタムポリシールールを設定) で、次の手順を実行します。

i. ルールオプションから、監査セキュリティグループで定義されたルールのみを許可するか、すべてのルールを拒否するかを選択します。この選択の詳細については、「コンテンツ監査セキュリティグループポリシー」を参照してください。

ii. [Audit security groups] (監査セキュリティグループ) で、[Add audit security groups] (監査セキュリティグループを追加) を選択してから、使用するセキュリティグループを選択します。Firewall Manager は、Firewall Manager 管理者アカウント内におけるすべての Amazon VPC インスタンスからの監査セキュリティグループのリストを設定します。ポリシーの監査セキュリティグループ数のデフォルト最大クォータは 1 です。クォータを引き上げる方法については、「AWS Firewall Manager クォータ」を参照してください。

iii. [Policy action] (ポリシーアクション) では、自動的に修復されないオプションを使用してポリシーを作成する必要があります。これにより、新しいポリシーを適用する前にその効果を評価できます。変更が適切であることを確認したら、ポリシーを編集し、ポリシーアクションを変更して、準拠していないリソースの自動修復を有効にします。

10. [Next] (次へ) を選択します。

11. [AWS アカウント this policy applies to] (このポリシーが適用される) で、次のようにオプションを選択します。

- 組織内のすべてのアカウントにポリシーを適用する場合は、デフォルトの選択のままにし、AWS 組織 のすべてのアカウントを含めます。
- 特定の AWS Organizations 組織単位 (OUs) にある特定のアカウントまたはアカウントにのみポリシーを適用する場合は、指定されたアカウントと組織単位のみを含める を選択し、含めるアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。
- 特定のアカウントまたは AWS Organizations 組織単位 (OUs) を除くすべてのアカウントにポリシーを適用する場合は、指定したアカウントと組織単位を除外し、その他すべての を含めて、除外するアカウントと OUs を追加します。OU を指定する方法は、OU およびその子で

ある OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。

選択できるオプションは 1 つのみです。

ポリシーを適用すると、Firewall Manager は新しいアカウントを設定と照合して自動的に評価します。例えば、特定のアカウントのみを含めると、Firewall Manager は新しいアカウントにポリシーを適用しません。別の例として、OU を含めた場合、OU またはその子である OU にアカウントを追加すると、Firewall Manager は新しいアカウントにポリシーを自動的に適用します。

12. [Resource type] (リソースタイプ) で、保護するリソースのタイプを選択します。
13. リソース では、指定したタグでリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

    複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

    リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。
14. [Next] (次へ) を選択します。
15. ポリシー設定を見直して目的の設定になっていることを確認し、[Create policy] (ポリシーの作成) を選択します。

Firewall Manager は、ポリシールールの設定に従って、監査セキュリティグループを AWS 組織内の範囲内セキュリティグループと比較します。ポリシーのステータスは、 AWS Firewall Manager ポリシーコンソールで確認できます。ポリシーを作成したら、ポリシーを編集して自動修復を有効にし、監査セキュリティグループポリシーを有効にすることができます。このポリシーの仕組みの詳細については、「コンテンツ監査セキュリティグループポリシー」を参照してください。

AWS Firewall Manager 使用状況監査セキュリティグループポリシーの作成

使用状況監査セキュリティグループポリシーの仕組みの詳細については、「使用状況監査セキュリティグループポリシー」を参照してください。

使用状況監査セキュリティグループポリシーを作成するには (コンソール)

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. [Create policy] (ポリシーの作成) を選択します。
4. [Policy type] (ポリシータイプ) で、[Security group] (セキュリティグループ) を選択します。
5. [セキュリティグループポリシータイプ]で、[未使用および冗長セキュリティグループの監査とクリーンアップ]を選択します。
6. リージョン で、 を選択します AWS リージョン。
7. [Next] (次へ) を選択します。
8. [Policy name](ポリシー名) で、フレンドリ名を入力します。
9. [Policy rules] (ポリシールール) で、使用可能なオプションのいずれかまたは両方を選択します。

   - [このポリシーの範囲内のセキュリティグループは少なくとも 1 つのリソースによって使用される必要があります] を選択した場合、Firewall Manager は、未使用と判断されるセキュリティグループを削除します。このルールを有効にすると、Firewall Manager はポリシーの保存時に最後にルールを実行します。

     Firewall Manager が使用状況と修復のタイミングを判断する方法の詳細については、「」を参照してください使用状況監査セキュリティグループポリシー。

     > **Note**
     >
     > この使用状況監査セキュリティグループポリシータイプを使用する場合は、対象範囲内のセキュリティグループの関連付けステータスを短時間で複数回変更しないでください。これにより、Firewall Manager が対応するイベントを見逃す可能性があります。

デフォルトでは、Firewall Manager は、セキュリティグループが使用できなくなるとすぐに、このポリシールールに準拠していないと見なします。オプションで、セキュリティグループが非準拠と見なされるまでに、未使用のセキュリティグループが存在することができる分数を最大 525,600 分 (365 日) まで指定できます。この設定を使用して、新しいセキュリティグループをリソースに関連付ける時間を確保できます。

> **Important**
>
> デフォルト値の 0 以外の分数を指定する場合は、 で間接的な関係を有効にする必要があります AWS Config。そうしないと、使用状況監査セキュリティグループポリシーは意図したとおりに機能しません。の間接的な関係については AWS Config、「 AWS Config デベロッパーガイド」の「 の間接的な関係 AWS Config」を参照してください。

- [このポリシーの範囲内のセキュリティグループは一意である必要があります] を選択した場合、Firewall Manager は、冗長なセキュリティグループを統合し、1 つのセキュリティグループのみがリソースに関連付けられるようにします。これを選択すると、Firewall Manager により、ポリシーの保存時に最初に実行されます。

10. [Policy action] (ポリシーアクション) では、自動的に修復されないオプションを使用してポリシーを作成することをお勧めします。これにより、新しいポリシーを適用する前にその効果を評価できます。変更が適切であることを確認したら、ポリシーを編集し、ポリシーアクションを変更して、準拠していないリソースの自動修復を有効にします。
11. [Next] (次へ) を選択します。
12. [AWS アカウント this policy applies to] (このポリシーが適用される) で、次のようにオプションを選択します。

    - 組織内のすべてのアカウントにポリシーを適用する場合は、デフォルトの選択のままにし、AWS 組織 のすべてのアカウントを含めます。
    - ポリシーを特定の AWS Organizations 組織単位 (OUs) 内の特定のアカウントにのみ適用する場合は、指定されたアカウントと組織単位のみを含める を選択し、含めるアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。
    - 特定のアカウントまたは AWS Organizations 組織単位 (OUs) を除くすべてのアカウントにポリシーを適用する場合は、指定されたアカウントと組織単位を除外し、その他すべての を含めて、除外するアカウントと OUs を追加します。OU を指定する方法は、OU およびその子

である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。

選択できるオプションは 1 つのみです。

ポリシーを適用すると、Firewall Manager は新しいアカウントを設定と照合して自動的に評価します。例えば、特定のアカウントのみを含めると、Firewall Manager は新しいアカウントにポリシーを適用しません。別の例として、OU を含めた場合、OU またはその子である OU にアカウントを追加すると、Firewall Manager は新しいアカウントにポリシーを自動的に適用します。

13. リソース では、指定したタグでリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

    複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

    リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。

14. [Next] (次へ) を選択します。

15. ポリシーの範囲から Firewall Manager 管理者アカウントを除外していない場合、Firewall Manager によりこれを行うよう求められます。これにより、セキュリティグループは Firewall Manager 管理者アカウントに残ります。このアカウントは、手動コントロール下で、共通セキュリティグループポリシーおよび監査セキュリティグループポリシーに使用します。このダイアログで目的のオプションを選択します。

16. ポリシー設定を見直して目的の設定になっていることを確認し、[Create policy] (ポリシーの作成) を選択します。

一意のセキュリティグループにする必要があることを選択した場合、Firewall Manager は各範囲内 Amazon VPC インスタンスで冗長セキュリティグループをスキャンします。その後、各セキュリティグループを少なくとも 1 つのリソースで使用するように選択した場合、Firewall Manager は、ルールで指定された時間 (単位: 分)、未使用のままのセキュリティグループをスキャンします。ポリシーのステータスは、 AWS Firewall Manager ポリシーコンソールで確認できます。このポリシーの仕組みの詳細については、「使用状況監査セキュリティグループポリシー」を参照してください。

ネットワーク ACL ポリシーの作成 AWS Firewall Manager

ネットワーク ACL ポリシーの仕組みについては、「」を参照してくださいネットワーク ACL ポリシー。

ネットワーク ACL ポリシーを作成するには、Amazon VPC サブネットで使用するネットワーク ACL を定義する方法を知っている必要があります。詳細については、「Amazon VPC ユーザーガイド」のACLs を使用してサブネットへのトラフィックを制御する」および「ネットワーク ACLs」を参照してください。

ネットワーク ACL ポリシーを作成するには (コンソール)

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. [Create policy] (ポリシーの作成) を選択します。
4. ポリシータイプ で、ネットワーク ACL を選択します。
5. リージョン で、 を選択します AWS リージョン。
6. [Next] (次へ) を選択します。
7. [Policy name] (ポリシー名) で、わかりやすい名前を入力します。
8. ポリシールール で、Firewall Manager が管理するネットワーク ACLs で常に実行するルールを定義します。ネットワーク ACLsインバウンドトラフィックとアウトバウンドトラフィックをモニタリングして処理するため、ポリシーでは、両方向のルールを定義します。

   どちらの方向でも、常に最初に実行するルールと最後に実行するルールを定義します。Firewall Manager が管理するネットワーク ACLs では、アカウント所有者は、これらの最初と最後のルールの間で実行するカスタムルールを定義できます。

9. ポリシーアクション で、非準拠のサブネットとネットワーク ACLs を識別したいが、まだ修正アクションを実行しない場合は、ポリシールールに準拠していないが、 を自動修正しないリソースの特定を選択します。これらのオプションは後で変更できます。

   代わりに、既存の範囲内のサブネットにポリシーを自動的に適用する場合は、非準拠のリソースを自動修正する を選択します。このオプションでは、ポリシールールのトラフィック処理動作がネットワーク ACL にあるカスタムルールと競合した場合に修復を強制するかどうかも指定します。Firewall Manager は、強制的な修復を行うかどうかにかかわらず、コンプライアンス違反で競合するルールを報告します。

10. [Next] (次へ) を選択します。

11. [AWS アカウント this policy applies to] (このポリシーが適用される) で、次のようにオプションを選択します。

    - 組織内のすべてのアカウントにポリシーを適用する場合は、デフォルトの選択のままにし、AWS 組織 のすべてのアカウントを含めます。
    - 特定の AWS Organizations 組織単位 (OUs) にある特定のアカウントまたはアカウントにのみポリシーを適用する場合は、指定されたアカウントと組織単位のみを含める を選択し、含めるアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。
    - 特定のアカウントまたは AWS Organizations 組織単位 (OUs) を除くすべてのアカウントにポリシーを適用する場合は、指定したアカウントと組織単位を除外し、その他すべての を含めて、除外するアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。

    選択できるオプションは 1 つのみです。

    ポリシーを適用すると、Firewall Manager は新しいアカウントを設定と照合して自動的に評価します。例えば、特定のアカウントのみを含める場合、Firewall Manager はポリシーを別の新しいアカウントに適用しません。別の例として、OU を含めた場合、OU またはその子である OU にアカウントを追加すると、Firewall Manager は新しいアカウントにポリシーを自動的に適用します。

12. リソースタイプ の場合、設定はサブネット に固定されます。

13. リソース では、指定したタグを持つリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

    複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

    リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。
14. [Next] (次へ) を選択します。
15. ポリシー設定を見直して目的の設定になっていることを確認し、[Create policy] (ポリシーの作成) を選択します。

Firewall Manager はポリシーを作成し、設定に従ってスコープ内のネットワーク ACLs のモニタリングと管理を開始します。このポリシーの仕組みの詳細については、「ネットワーク ACL ポリシー」を参照してください。

の AWS Firewall Manager ポリシーの作成 AWS Network Firewall

Firewall Manager の Network Firewall ポリシーでは、 AWS Network Firewallで管理するルールグループを使用します。ルールグループの管理については、「Network Firewall デベロッパーガイド」の「AWS Network Firewall ルールグループ」を参照してください。

Firewall Manager の Network Firewall ポリシーについては、「AWS Network Firewall ポリシー」を参照してください。

の Firewall Manager ポリシーを作成するには AWS Network Firewall （ コンソール ）

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. [Create policy] (ポリシーの作成) を選択します。
4. [Policy type] (ポリシータイプ) では [AWS Network Firewall] を選択します。
5. [Firewall management type] (ファイアウォールの管理タイプ) で、Firewall Manager にポリシーのファイアウォールをどのように管理させるか選択します。次のオプションから選択します。

   - [Distributed] (分散型) を使用すると、Firewall Manager は、ポリシーの範囲内の各 VPC でファイアウォールエンドポイントを作成および維持します。
   - [Centralized] (集約型) を使用すると、Firewall Manager は、単一の検査 VPC でエンドポイントを作成および維持します。
   - [Import existing firewalls] (既存のファイアウォールのインポート)- Firewall Manager は、リソースセットを使用して Network Firewall から既存のファイアウォールをインポートします。リソースセットの詳細については、「Firewall Manager でのリソースセットの操作」を参照してください。

6. リージョン で、 を選択します AWS リージョン。複数のリージョンのリソースを保護するには、各リージョンに別々の ポリシーを作成する必要があります。
7. [Next] (次へ) を選択します。
8. [Policy name] (ポリシー名) で、わかりやすい名前を入力します。Firewall Manager は、Network Firewall のファイアウォールおよび作成するファイアウォールポリシーの名前にポリシー名を含めます。
9. [AWS Network Firewall policy configuration] (ポリシー設定) では、Network Firewall の場合と同じようにファイアウォールポリシーを設定します。ステートレスルールグループおよびステートフルルールグループを追加し、ポリシーのデフォルトアクションを指定します。オプションで、ポリシーのステートフルルール評価順序とデフォルトアクションを設定し、ログ記録設定を行うことができます。Network Firewall のファイアウォールポリシーの管理については、「AWS Network Firewall デベロッパーガイド」の「AWS Network Firewall ファイアウォールポリシー」を参照してください。

   Firewall Manager の Network Firewall ポリシーを作成すると、Firewall Manager は、範囲内にあるアカウント用にファイアウォールポリシーを作成します。個々のアカウントマネージャーは、ファイアウォールポリシーにルールグループを追加できますが、ここで指定する設定を変更することはできません。
10. [Next] (次へ) を選択します。
11. 前のステップで選択した [Firewall management type] (ファイアウォール管理タイプ) に応じて、次のいずれかを実行します。

- [分散型] ファイアウォール管理タイプを使用している場合、[AWS Firewall Manager エンドポイント設定] 内の [ファイアウォールのエンドポイントの場所] で、以下のオプションのいずれかを選択します。
  - [Custom endpoint configuration] (カスタムエンドポイント設定) - Firewall Manager は、指定したアベイラビリティーゾーンに、ポリシー範囲内の各 VPC に対してファイアウォールを作成します。各ファイアウォールには、少なくとも 1 つのファイアウォールエンドポイントが含まれています。
    - [Availability Zones] (アベイラビリティーゾーン) で、ファイアウォールエンドポイントを作成するアベイラビリティーゾーンを選択します。アベイラビリティーゾーンは、[Availability Zone name] (アベイラビリティーゾーン名) または [Availability Zone ID] (アベイラビリティーゾーン ID) で選択できます。
    - Firewall Manager が VPC のファイアウォールサブネットに使用する CIDR ブロックを指定する場合、そのすべては /28 CIDR ブロックである必要があります。1 行に 1 つのブロックを入力します。これらを省略すると、Firewall Manager は、VPC で使用可能な IP アドレスから選択します。

      > **Note**
      >
      > 自動修復は AWS Firewall Manager Network Firewall ポリシーに対して自動的に行われるため、ここで自動修復を選択しないオプションは表示されません。

  - エンドポイントの自動設定 - Firewall Manager は、VPC 内のパブリックサブネットを持つアベイラビリティーゾーンにファイアウォールエンドポイントを自動的に作成します。
    - [Firewall endpoints] (ファイアウォールエンドポイント) の設定では、Firewall Manager によるファイアウォールエンドポイントの管理方法を指定します。高可用性を実現するために、複数のエンドポイントを使用することをお勧めします。
- このポリシーに [集約型] ファイアウォール管理タイプを使用している場合は、[AWS Firewall Manager エンドポイント設定] 内の [インスペクション VPC の設定] で、検査 VPC の所有者の AWS アカウント ID と検査 VPC の VPC ID を入力します。
  - [Availability Zones] (アベイラビリティーゾーン) で、ファイアウォールエンドポイントを作成するアベイラビリティーゾーンを選択します。アベイラビリティーゾーンは、[Availability Zone name] (アベイラビリティーゾーン名) または [Availability Zone ID] (アベイラビリティーゾーン ID) で選択できます。
  - Firewall Manager が VPC のファイアウォールサブネットに使用する CIDR ブロックを指定する場合、そのすべては /28 CIDR ブロックである必要があります。1 行に 1 つのブロック

を入力します。これらを省略すると、Firewall Manager は、VPC で使用可能な IP アドレスから選択します。

> **Note**
>
> 自動修復は AWS Firewall Manager Network Firewall ポリシーに対して自動的に行われるため、ここで自動修復を選択しないオプションは表示されません。

- [Import existing firewalls] (既存のファイアウォールのインポート) のファイアウォール管理タイプを使用している場合は、[Resource sets] (リソースセット) で 1 つ以上のリソースセットを追加します。リソースセットは、このポリシーで一元管理したい組織のアカウントが所有する既存の Network Firewall を定義します。リソースセットをポリシーに追加するには、まずコンソールまたは PutResourceSet API を使用してリソースセットを作成する必要があります。リソースセットの詳細については、「Firewall Manager でのリソースセットの操作」を参照してください。Network Firewall から既存のファイアウォールをインポートする方法の詳細については、「既存のファイアウォールのインポート」を参照してください。

12. [次へ] をクリックします。
13. ポリシーで分散型ファイアウォール管理タイプを使用している場合は、[Route management] (ルート管理) で、Firewall Manager がそれぞれのファイアウォールエンドポイントを経由してルーティングする必要のあるトラフィックをモニタリングおよびアラートするかどうかを選択します。

> **Note**
>
> [Monitor] (モニタリング) を選択した場合、後日設定を [Off] (オフ) に変更することはできません。モニタリングは、ポリシーを削除するまで継続します。

14. [Traffic type] (トラフィックタイプ) で、ファイアウォール検査のためにトラフィックをルーティングするトラフィックエンドポイントをオプションで追加します。
15. [Allow required cross-AZ traffic] (必要なクロス AZ トラフィックを許可) で、このオプションを有効にすると、Firewall Manager は、独自のファイアウォールエンドポイントを持たないアベイラビリティーゾーンの場合、検査のためにアベイラビリティーゾーンからトラフィックを送信する準拠ルーティングとして扱います。エンドポイントを持つアベイラビリティーゾーンでは、常に独自のトラフィックを検査する必要があります。
16. [Next] (次へ) を選択します。

17. [Policy scope] (ポリシーの範囲) の [AWS アカウント this policy applies to] (このポリシーが適用される) で、次のようにオプションを選択します。

    - 組織内のすべてのアカウントにポリシーを適用する場合は、デフォルトの選択のままにし、AWS 組織 のすべてのアカウントを含めます。
    - 特定の AWS Organizations 組織単位 (OUs) にある特定のアカウントまたはアカウントにのみポリシーを適用する場合は、指定されたアカウントと組織単位のみを含める を選択し、含めるアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。
    - 特定のアカウントまたは AWS Organizations 組織単位 (OUs) を除くすべてのアカウントにポリシーを適用する場合は、指定されたアカウントと組織単位を除外し、その他すべての を含めて、除外するアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。

    選択できるオプションは 1 つのみです。

    ポリシーを適用すると、Firewall Manager は新しいアカウントを設定と照合して自動的に評価します。例えば、特定のアカウントのみを含めると、Firewall Manager は新しいアカウントにポリシーを適用しません。別の例として、OU を含めた場合、OU またはその子である OU にアカウントを追加すると、Firewall Manager は新しいアカウントにポリシーを自動的に適用します。

18. Network Firewall ポリシーの [Resource type] (リソースタイプ) は [VPC] です。

19. リソース では、指定したタグでリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

    複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

    リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。

20. [次へ] をクリックします。

21. ポリシータグ には、Firewall Manager ポリシーリソースに追加する識別タグを追加します。タグの詳細については、「タグエディタの使用」を参照してください。

22. [Next] (次へ) を選択します。
23. 新しいポリシー設定を確認し、調整が必要なページに戻ります。

    ポリシーが完成したら、[Create policy] (ポリシーの作成) を選択します。[AWS Firewall Manager ポリシー] ペインにポリシーが一覧表示されます。おそらく、アカウントの見出しの下に保留中と表示され、自動修復設定のステータスを示します。ポリシーの作成には数分かかることがあります。[Pending] (保留中) ステータスがアカウント数に置き換えられたら、ポリシー名を選択して、アカウントとリソースの準拠ステータスを調べることができます。詳細については、「AWS Firewall Manager ポリシーのコンプライアンス情報の表示」を参照してください。

Amazon Route 53 Resolver DNS Firewall の AWS Firewall Manager ポリシーの作成

Firewall Manager の DNS Firewall ポリシーでは、Amazon Route 53 Resolver DNS Firewall で管理するルールグループを使用します。ルールグループの管理については、「Amazon Route 53 デベロッパーガイド」の「DNS Firewall でのルールグループおよびルールの管理」を参照してください。

Firewall Manager の DNS Firewall ポリシーについては、「Amazon Route 53 Resolver DNS Firewall ポリシー」を参照してください。

Amazon Route 53 Resolver DNS Firewall の Firewall Manager ポリシーを作成するには (コンソール)

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. [Create policy] (ポリシーの作成) を選択します。
4. [Policy type] (ポリシータイプ) で、[Amazon Route 53 Resolver DNS Firewall] を選択します。
5. リージョン で、 を選択します AWS リージョン。複数のリージョンのリソースを保護するには、各リージョンに別々の ポリシーを作成する必要があります。
6. [Next] (次へ) を選択します。

7. [Policy name] (ポリシー名) で、わかりやすい名前を入力します。
8. ポリシー設定で、VPC のルールグループの関連付けの中で DNS Firewall が最初と最後に評価するルールグループを追加します。ポリシーには最大 2 つのルールグループを追加できます。

   Firewall Manager の DNS Firewall ポリシーを作成すると、Firewall Manager は、指定した関連付けの優先順位を使用して、範囲内の VPC とアカウントのルールグループの関連付けを作成します。個々のアカウントマネージャーは、最初の関連付けと最後の関連付けの間にルールグループの関連付けを追加できますが、お客様がここで定義する関連付けを変更することはできません。詳細については、「Amazon Route 53 Resolver DNS Firewall ポリシー」を参照してください。
9. [Next] (次へ) を選択します。
10. [AWS アカウント this policy applies to] (このポリシーが適用される) で、次のようにオプションを選択します。

    - 組織内のすべてのアカウントにポリシーを適用する場合は、デフォルトの選択のままにし、AWS 組織 のすべてのアカウントを含めます。
    - 特定の AWS Organizations 組織単位 (OUs) にある特定のアカウントまたはアカウントにのみポリシーを適用する場合は、指定されたアカウントと組織単位のみを含める を選択し、含めるアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。
    - 特定のアカウントまたは AWS Organizations 組織単位 (OUs) を除くすべてのアカウントにポリシーを適用する場合は、指定したアカウントと組織単位を除外し、その他すべての を含めて、除外するアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。

    選択できるオプションは 1 つのみです。

    ポリシーを適用すると、Firewall Manager は新しいアカウントを設定と照合して自動的に評価します。例えば、特定のアカウントのみを含めると、Firewall Manager は新しいアカウントにポリシーを適用しません。別の例として、OU を含めた場合、OU またはその子である OU にアカウントを追加すると、Firewall Manager は新しいアカウントにポリシーを自動的に適用します。
11. DNS Firewall ポリシーの [Resource type] (リソースタイプ) は [VPC] です。

12. リソース では、指定したタグを持つリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

    複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

    リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。

13. [次へ] をクリックします。

14. ポリシータグ には、Firewall Manager ポリシーリソースに追加する識別タグを追加します。タグの詳細については、「タグエディタの使用」を参照してください。

15. [Next] (次へ) を選択します。

16. 新しいポリシー設定を確認し、調整が必要なページに戻ります。

    ポリシーが完成したら、[Create policy] (ポリシーの作成) を選択します。[AWS Firewall Manager ポリシー] ペインにポリシーが一覧表示されます。おそらく、アカウントの見出しの下に「保留中」と表示され、自動修復設定のステータスを示します。ポリシーの作成には数分かかることがあります。[Pending] (保留中) ステータスがアカウント数に置き換えられたら、ポリシー名を選択して、アカウントとリソースの準拠ステータスを調べることができます。詳細については、「AWS Firewall Manager ポリシーのコンプライアンス情報の表示」を参照してください。

Palo Alto Networks Cloud NGFW の AWS Firewall Manager ポリシーの作成

Palo Alto Networks Cloud Next Generation Firewall (Palo Alto Networks Cloud NGFW) の Firewall Manager ポリシーでは、Firewall Manager を使用して Palo Alto Networks Cloud NGFW リソースをデプロイし、NGFW ルールスタックをすべての AWS アカウントで一元的に管理します。

Firewall Manager Palo Alto Networks Cloud NGFW ポリシーの詳細については、「Palo Alto Networks Cloud NGFW ポリシー」を参照してください。Firewall Manager 用に Palo Alto Networks Cloud NGFW を設定および管理する方法については、Palo Alto Networks Palo Alto Networks Cloud NGFW on AWS のドキュメントを参照してください。

前提条件

AWS Firewall Managerのアカウントを準備するには、いくつかの必須のステップがあります。それらのステップは、AWS Firewall Manager 前提条件 で説明されています。次のステップに進む前に、すべての前提条件を満たしてください。

Palo Alto Networks Cloud NGFW の Firewall Manager ポリシーを作成するには (コンソール)

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. [Create policy] (ポリシーの作成) を選択します。
4. [Policy type] (ポリシータイプ) で、[Palo Alto Networks Cloud NGFW] を選択します。 AWS Marketplace で Palo Alto Networks Cloud NGFW サービスをまだサブスクライブしていない場合は、まずサブスクライブする必要があります。 AWS Marketplace でサブスクライブするには、AWS Marketplace の詳細を表示 を選択します。
5. [Deployment model] (デプロイモデル) で、[Distributed model] (分散モデル) または [Centralized model] (集約型モデル) のいずれかを選択します。デプロイモデルによって、Firewall Managerがポリシーのエンドポイントを管理する方法が決まります。分散モデルでは、Firewall Managerは、ポリシーの範囲内の各 VPC にファイアウォールエンドポイントを維持します。集約型モデルでは、Firewall Manager は検査 VPC に単一のエンドポイントを維持します。
6. リージョン で、 を選択します AWS リージョン。複数のリージョンのリソースを保護するには、各リージョンに別々の ポリシーを作成する必要があります。
7. [Next] (次へ) を選択します。
8. [Policy name] (ポリシー名) で、わかりやすい名前を入力します。
9. ポリシー設定で、このポリシーに関連付ける Palo Alto Networks Cloud NGFW ファイアウォールポリシーを選択します。Palo Alto Networks Cloud NGFW ファイアウォールポリシーの一覧には、Palo Alto Networks Cloud NGFW テナントに関連付けられているすべての Cloud NGFW ファイアウォールポリシーが含まれています。Palo Alto Networks Cloud NGFW ファイアウォー

ルポリシーの作成と管理については、「デプロイガイド」の「Palo Alto Networks Cloud NGFW for トピック AWSAWS Firewall Manager」の「Deploy Palo Alto Networks Cloud NGFW for AWS 」を参照してください。

10. Palo Alto Networks Cloud NGFW ログ記録 - オプションで、ポリシーに記録する Palo Alto Networks Cloud NGFW ログタイプ (複数可) を選択します。Palo Alto Networks Cloud NGFW ログタイプの詳細については、「デプロイガイド」の「Palo Alto Networks Cloud NGFW のログ記録の設定 AWS」を参照してください。 AWS

    [log destination] (ログの宛先) で、Firewall Manager がログを書き込む場合を指定します。

11. [Next] (次へ) を選択します。

12. [Configure third-party firewall endpoint] (サードパーティーのファイアウォールエンドポイントを設定) で、ファイアウォールエンドポイントの作成に分散デプロイモデルと集約型デプロイモデルのいずれを使用しているかに応じて、次のいずれかを実行します。

    - このポリシーに分散デプロイモデルを使用している場合は、[Availability Zones] (アベイラビリティーゾーン) で、ファイアウォールエンドポイントを作成するアベイラビリティーゾーンを選択します。アベイラビリティーゾーンは、[Availability Zone name] (アベイラビリティーゾーン名) または [Availability Zone ID] (アベイラビリティーゾーン ID) で選択できます。
    - このポリシーに集約型デプロイモデルを使用している場合は、[Inspection VPC configuration] (検査 VPC 設定) の [AWS Firewall Manager endpoint configuration] (エンドポイント設定) で、検査 VPC の所有者の AWS アカウント ID と検査 VPC の VPC ID を入力します。
        - [Availability Zones] (アベイラビリティーゾーン) で、ファイアウォールエンドポイントを作成するアベイラビリティーゾーンを選択します。アベイラビリティーゾーンは、[Availability Zone name] (アベイラビリティーゾーン名) または [Availability Zone ID] (アベイラビリティーゾーン ID) で選択できます。

13. Firewall Manager が VPC のファイアウォールサブネットに使用する CIDR ブロックを指定する場合、そのすべては /28 CIDR ブロックである必要があります。1 行に 1 つのブロックを入力します。これらを省略すると、Firewall Manager は、VPC で使用可能な IP アドレスから選択します。

    > **Note**
    >
    > 自動修復は AWS Firewall Manager Network Firewall ポリシーに対して自動的に行われるため、ここで自動修復を選択しないオプションは表示されません。

14. [Next] (次へ) を選択します。

15. [Policy scope] (ポリシーの範囲) の [AWS アカウント this policy applies to] (このポリシーが適用される) で、次のようにオプションを選択します。

    - 組織内のすべてのアカウントにポリシーを適用する場合は、デフォルトの選択のままにし、AWS 組織 のすべてのアカウントを含めます。
    - ポリシーを特定の AWS Organizations 組織単位 (OUs) にある特定のアカウントにのみ適用する場合は、指定したアカウントと組織単位のみを含める を選択し、含めるアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。
    - 特定のアカウントまたは AWS Organizations 組織単位 (OUs) 以外のすべてのアカウントにポリシーを適用する場合は、指定したアカウントと組織単位を除外し、その他すべての を含めて、除外するアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。

    選択できるオプションは 1 つのみです。

    ポリシーを適用すると、Firewall Manager は新しいアカウントを設定と照合して自動的に評価します。例えば、特定のアカウントのみを含めると、Firewall Manager は新しいアカウントにポリシーを適用しません。別の例として、OU を含めた場合、OU またはその子である OU にアカウントを追加すると、Firewall Manager は新しいアカウントにポリシーを自動的に適用します。

16. Network Firewall ポリシーの [Resource type] (リソースタイプ) は [VPC] です。

17. リソース では、指定したタグでリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

    複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

    リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。

18. [Grant cross-account access] (クロスアカウントアクセスを付与) で、[Download AWS CloudFormation template] (テンプレートをダウンロード) を選択します。これにより、 AWS CloudFormation スタックの作成に使用できる AWS CloudFormation テンプレートがダウンロードされます。このスタックは、Palo Alto Networks Cloud NGFW リソースを管理するた

めのクロスアカウントアクセス許可を Firewall Manager に付与する AWS Identity and Access Management ロールを作成します。スタックの詳細については、「AWS CloudFormation ユーザーガイド」の「StackSets の操作」を参照してください。

19. [Next] (次へ) を選択します。
20. ポリシータグ には、Firewall Manager ポリシーリソースに追加する識別タグを追加します。タグの詳細については、「タグエディタの使用」を参照してください。
21. [Next] (次へ) を選択します。
22. 新しいポリシー設定を確認し、調整が必要なページに戻ります。

    ポリシーが完成したら、[Create policy] (ポリシーの作成) を選択します。[AWS Firewall Manager ポリシー] ペインにポリシーが一覧表示されます。おそらく、アカウントの見出しの下に「保留中」と表示され、自動修復設定のステータスを示します。ポリシーの作成には数分かかることがあります。[Pending] (保留中) ステータスがアカウント数に置き換えられたら、ポリシー名を選択して、アカウントとリソースの準拠ステータスを調べることができます。詳細については、「AWS Firewall Manager ポリシーのコンプライアンス情報の表示」を参照してください。

Fortigate Cloud Native Firewall (CNF) as a Service の AWS Firewall Manager ポリシーの作成

Fortigate CNF の Firewall Manager ポリシーを使用すると、Firewall Manager を使用して、すべての AWS アカウントで Fortigate CNF リソースをデプロイおよび管理できます。

Firewall Manager Fortigate CNF ポリシーについては、「Fortigate Cloud Native Firewall (CNF) as a Service ポリシー」を参照してください。Fortigate CNF を Firewall Manager で使用するための設定について詳しくは、「Fortinet のドキュメント」を参照してください。

前提条件

AWS Firewall Managerのアカウントを準備するには、いくつかの必須のステップがあります。それらのステップは、AWS Firewall Manager 前提条件 で説明されています。次のステップに進む前に、すべての前提条件を満たしてください。

Fortigate CNF の Firewall Manager ポリシーを作成するには (コンソール)

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/

fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

> **Note**
>
> Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. [Create policy] (ポリシーの作成) を選択します。
4. [Policy type] (ポリシータイプ) には、[Fortigate Cloud Native Firewall (CNF) as a Service] (サービスとしての Fortigate Cloud ネイティブファイアウォール (CNF)) を選択してください。AWS Marketplace で Fortigate CNF サービスをまだサブスクライブしていない場合は、まずサブスクライブする必要があります。 AWS Marketplace でサブスクライブするには、 AWS Marketplace の詳細を表示 を選択します。
5. [Deployment model] (デプロイモデル) で、[Distributed model] (分散モデル) または [Centralized model] (集約型モデル) のいずれかを選択します。デプロイモデルによって、Firewall Manager がポリシーのエンドポイントを管理する方法が決まります。分散モデルでは、Firewall Manager は、ポリシーの範囲内の各 VPC にファイアウォールエンドポイントを維持します。集約型モデルでは、Firewall Manager は検査 VPC に単一のエンドポイントを維持します。
6. リージョン で、 を選択します AWS リージョン。複数のリージョンのリソースを保護するには、各リージョンに別々の ポリシーを作成する必要があります。
7. [Next] (次へ) を選択します。
8. [Policy name] (ポリシー名) で、わかりやすい名前を入力します。
9. ポリシー設定で、このポリシーに関連付ける Fortigate CNF ファイアウォールポリシーを選択します。Fortigate CNF ファイアウォールポリシーのリストには、Fortigate CNF テナントに関連付けられているすべての CNF ファイアウォールポリシーが含まれています。Fortigate CNF テナントの作成と管理については、「Fortinet のドキュメント」を参照してください。
10. [Next] (次へ) を選択します。
11. [Configure third-party firewall endpoint] (サードパーティーのファイアウォールエンドポイントを設定) で、ファイアウォールエンドポイントの作成に分散デプロイモデルと集約型デプロイモデルのいずれを使用しているかに応じて、次のいずれかを実行します。

    - このポリシーに分散デプロイモデルを使用している場合は、[Availability Zones] (アベイラビリティーゾーン) で、ファイアウォールエンドポイントを作成するアベイラビリティーゾーン

を選択します。アベイラビリティーゾーンは、[Availability Zone name] (アベイラビリティーゾーン名) または [Availability Zone ID] (アベイラビリティーゾーン ID) で選択できます。

- このポリシーに集約型デプロイモデルを使用している場合は、[Inspection VPC configuration] (検査 VPC 設定) の [AWS Firewall Manager endpoint configuration] (エンドポイント設定) で、検査 VPC の所有者の AWS アカウント ID と検査 VPC の VPC ID を入力します。
  - [Availability Zones] (アベイラビリティーゾーン) で、ファイアウォールエンドポイントを作成するアベイラビリティーゾーンを選択します。アベイラビリティーゾーンは、[Availability Zone name] (アベイラビリティーゾーン名) または [Availability Zone ID] (アベイラビリティーゾーン ID) で選択できます。

12. Firewall Manager が VPC のファイアウォールサブネットに使用する CIDR ブロックを指定する場合、そのすべては /28 CIDR ブロックである必要があります。1 行に 1 つのブロックを入力します。これらを省略すると、Firewall Manager は、VPC で使用可能な IP アドレスから選択します。

> **Note**
>
> 自動修復は AWS Firewall Manager Network Firewall ポリシーに対して自動的に行われるため、ここで自動修復を選択しないオプションは表示されません。

13. [Next] (次へ) を選択します。
14. [Policy scope] (ポリシーの範囲) の [AWS アカウント this policy applies to] (このポリシーが適用される) で、次のようにオプションを選択します。

    - 組織内のすべてのアカウントにポリシーを適用する場合は、デフォルトの選択のままにし、AWS 組織 のすべてのアカウントを含めます。
    - 特定の AWS Organizations 組織単位 (OUs) にある特定のアカウントまたはアカウントにのみポリシーを適用する場合は、指定されたアカウントと組織単位のみを含める を選択し、含めるアカウントと OUsを追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。
    - 特定のアカウントまたは AWS Organizations 組織単位 (OUs) を除くすべてのアカウントにポリシーを適用する場合は、指定したアカウントと組織単位を除外し、その他すべての を含めて、除外するアカウントと OUs を追加します。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。

選択できるオプションは 1 つのみです。

ポリシーを適用すると、Firewall Manager は新しいアカウントを設定と照合して自動的に評価します。例えば、特定のアカウントのみを含めると、Firewall Manager は新しいアカウントにポリシーを適用しません。別の例として、OU を含めた場合、OU またはその子である OU にアカウントを追加すると、Firewall Manager は新しいアカウントにポリシーを自動的に適用します。

15. Network Firewall ポリシーの [Resource type] (リソースタイプ) は [VPC] です。
16. リソース では、指定したタグを持つリソースを含めるか除外するかによって、タグ付けを使用してポリシーの範囲を絞り込むことができます。包含または除外は使用できますが、両方を使用することはできません。タグの詳細については、「タグエディタの使用」を参照してください。

    複数のタグを入力した場合、含めるまたは除外するリソースにはそれらのすべてのタグが付いている必要があります。

    リソースタグには NULL 以外の値のみを含めることができます。タグの値を省略すると、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。
17. [Grant cross-account access] (クロスアカウントアクセスを付与) で、[Download AWS CloudFormation template] (テンプレートをダウンロード) を選択します。これにより、 AWS CloudFormation スタックの作成に使用できる AWS CloudFormation テンプレートがダウンロードされます。このスタックは、Fortigate CNF リソースを管理するためのクロスアカウントアクセス許可を Firewall Manager に付与する AWS Identity and Access Management ロールを作成します。スタックの詳細については、「AWS CloudFormation ユーザーガイド」の「StackSets の操作」を参照してください。スタックを作成するには、Fortigate CNF ポータルのアカウント ID が必要です。
18. [次へ] をクリックします。
19. ポリシータグ には、Firewall Manager ポリシーリソースに追加する識別タグを追加します。タグの詳細については、「タグエディタの使用」を参照してください。
20. [Next] (次へ) を選択します。
21. 新しいポリシー設定を確認し、調整が必要なページに戻ります。

    ポリシーが完成したら、[Create policy] (ポリシーの作成) を選択します。[AWS Firewall Manager ポリシー] ペインにポリシーが一覧表示されます。おそらく、アカウントの見出しの下に「保留中」と表示され、自動修復設定のステータスを示します。ポリシーの作成には数分かかることがあります。[Pending] (保留中) ステータスがアカウント数に置き換えられたら、ポリ

シー名を選択して、アカウントとリソースの準拠ステータスを調べることができます。詳細については、「AWS Firewall Manager ポリシーのコンプライアンス情報の表示」を参照してください。

AWS Firewall Manager ポリシーを削除する

次のステップを実行して、Firewall Manager ポリシーを削除できます。

ポリシーを削除するには (コンソール)

1. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
2. 削除するポリシーの横にあるオプションを選択します。
3. [Delete] (削除) を選択します。

> **Note**
>
> Firewall Manager 共通セキュリティグループポリシーを削除する場合、ポリシーのレプリカセキュリティグループを削除するには、ポリシーによって作成されたリソースをクリーンアップするオプションを選択します。それ以外の場合、プライマリが削除されても、レプリカはそのまま残り、各 Amazon VPC インスタンスで手動管理が必要です。

> **Important**
>
> Firewall Manager Shield Advanced ポリシーを削除すると、ポリシーは削除されますが、アカウントは Shield Advanced にサブスクライブされたままとなります。

AWS Firewall Manager ポリシーの範囲

ポリシーの範囲は、ポリシーが適用される場所を定義するものです。一元管理ポリシーは、 の組織内のすべてのアカウントとリソース AWS Organizations、またはアカウントとリソースのサブセットに適用できます。ポリシーの範囲の設定方法については、「AWS Firewall Manager ポリシーの作成」を参照してください。

のポリシースコープオプション AWS Firewall Manager

組織に新しいアカウントまたはリソースを追加すると、Firewall Manager は、各ポリシーの設定に対して自動的に評価し、これらの設定に基づいてポリシーを適用します。例えば、指定したリストのアカウント番号以外のすべてのアカウントにポリシーを適用するように選択できます。また、リスト内のすべてのタグを持つリソースにのみポリシーを適用することを選択することもできます。

AWS アカウント 範囲内

ポリシーの AWS アカウント 影響を受ける を定義するために指定する設定によって、ポリシーを適用する AWS 組織内のアカウントが決まります。次のいずれかの方法でポリシーを適用できます。

- 組織のすべてのアカウントに適用
- 含めたアカウント番号と AWS Organizations の組織単位 (OU) の特定のリストにのみ適用
- 除外したアカウント番号と AWS Organizations の組織単位 (OU) の特定のリストを除くすべてに適用

の詳細については AWS Organizations、「 AWS Organizations ユーザーガイド」を参照してください。

範囲内のリソース

範囲内のアカウントの設定と同様に、リソースに指定した設定によって、ポリシーを適用する範囲内のリソースタイプが決まります。次のいずれかを選択できます。

- すべてのリソース
- 指定したすべてのタグを持つリソース
- 指定したすべてのタグを持つリソースを除くすべてのリソース

NULL 以外の値を持つリソースタグのみを指定できます。値に何も指定しない場合、Firewall Manager は空の文字列値「」でタグを保存します。リソースタグは、同じキーと同じ値を持つタグとのみ一致します。

リソースのタグ付けの詳細については、「タグエディタの使用」を参照してください。

でのポリシー範囲の管理 AWS Firewall Manager

ポリシーが設定されると、Firewall Manager はポリシーを継続的に管理し、ポリシーの範囲に従って、新しい AWS アカウント およびリソースが追加されるときにそれらを適用します。

Firewall Manager が AWS アカウント および リソースを管理する方法

アカウントまたはリソースが何らかの理由で範囲外になった場合は、ポリシースコープを離れるリソースから保護を自動的に削除するチェックボックスを選択しない限り、保護を自動的に削除したり、Firewall Manager が管理するリソースを削除したり AWS Firewall Manager しないでください。

> **Note**
>
> オプション ポリシーの範囲を離れるリソースから自動的に保護を削除する は、 AWS Shield Advanced または AWS WAF Classic ポリシーでは使用できません。

このチェックボックスをオンにすると、 AWS Firewall Manager は、Firewall Manager がアカウントがポリシーの範囲を離れるときに管理するリソースを自動的にクリーンアップするように に指示します。例えば、カスタマーリソースがポリシーの範囲から外れた場合、Firewall Manager は、Firewall Manager で管理するウェブ ACL と保護されたカスタマーリソースとの関連付けを解除します。

カスタマーリソースがポリシーの範囲外になったときに保護から削除する必要があるリソースを決定するために、Firewall Manager は次のガイドラインに従います。

- デフォルトの動作
  - 関連付けられた AWS Config マネージドルールが削除されます。この動作は、チェックボックスとは無関係です。
  - リソースを含まない、関連付けられた AWS WAF ウェブアクセスコントロールリスト (ウェブ ACLsはすべて削除されます。この動作は、チェックボックスとは無関係です。
  - 範囲外となった保護されたリソースは、関連付けられ、保護されたままになります。例えば、ウェブ ACL に関連付けられた API Gateway からの Application Load Balancer または API は、ウェブ ACL に関連付けられたままになり、保護は維持されます。
- [Automatically remove protections from resources that leave the policy scope] (ポリシーの範囲を外れるリソースから保護を自動的に削除) のチェックボックスをオンにすると、次のようになります。
  - 関連付けられた AWS Config マネージドルールが削除されます。この動作は、チェックボックスとは無関係です。
  - リソースを含まない、関連付けられた AWS WAF ウェブアクセスコントロールリスト (ウェブ ACLsはすべて削除されます。この動作は、チェックボックスとは無関係です。

- 範囲外の保護されたリソースは、ポリシーの範囲外になると、Firewall Manager の保護から自動的に関連付けが解除され、削除されます。例えば、セキュリティグループポリシーの場合、Elastic Inference アクセラレーターまたは Amazon EC2 インスタンスは、ポリシーのスコープを離れると、レプリケートされたセキュリティグループから自動的に関連付けが解除されます。レプリケートされたセキュリティグループとそのリソースは、自動的に保護から削除されます。

マネージドリスト

マネージドアプリケーションおよびプロトコルリストにより、 AWS Firewall Manager コンテンツ監査セキュリティグループポリシーの設定と管理が合理化されます。ポリシーが許可または禁止するプロトコルとアプリケーションを定義するために、マネージドリストを使用します。コンテンツ監査セキュリティグループポリシーの詳細については、「コンテンツ監査セキュリティグループポリシー」を参照してください。

コンテンツ監査セキュリティグループポリシーでは、次のタイプのマネージドリストを使用できます。

- Firewall Manager のアプリケーションリストとプロトコルリスト - Firewall Manager は、これらのリストを管理します。
  - アプリケーションリストには、一般ユーザーに対して許可または拒否される必要がある一般的に使用されるアプリケーションを記述する `FMS-Default-Public-Access-Apps-Allowed` と `FMS-Default-Public-Access-Apps-Denied` が含まれます。
  - プロトコルリストには、一般ユーザーに対して許可される必要がある一般的に使用されるプロトコルリストである `FMS-Default-Protocols-Allowed` が含まれます。Firewall Manager が管理するリストはすべて使用できますが、編集や削除はできません。
- カスタムアプリケーションリストとプロトコルリスト - これらのリストを管理できます。必要な設定を使用して、どちらのタイプのリストも作成できます。独自のカスタムマネージドリストを完全に制御でき、必要に応じて作成、編集、削除できます。

> **Note**
>
> 現在、Firewall Manager は、カスタムマネージドリストが削除されると、そのカスタムマネージドリストへの参照をチェックしません。つまり、アクティブなポリシーによって使用されている場合でも、カスタムマネージドアプリケーションリストまたはプロトコルリストを削除できます。これにより、ポリシーが機能しなくなる可能性があります。アプリ

ケーションリストまたはプロトコルリストは、アクティブなポリシーによって参照されていないことを検証した後にのみ削除します。

AWS 管理対象リストはリソースです。カスタムマネージドリストにタグを付けることができます。Firewall Manager のマネージドリストにタグを付けることはできません。

マネージドリストのバージョニング

カスタムマネージドリストにはバージョンがありません。カスタムリストを編集すると、そのリストを参照するポリシーは、更新されたリストを自動的に使用します。

Firewall Manager のマネージドリストはバージョニングされています。Firewall Manager サービスチームは、セキュリティのベストプラクティスをリストに適用するために、必要に応じて新しいバージョンを公開します。

ポリシーで Firewall Manager マネージドリストを使用する場合、次のようにバージョニング戦略を選択します。

- 利用可能な最新バージョン – リスト用に明示的なバージョン設定を指定しない場合、ポリシーは自動的に最新バージョンを使用します。コンソールで使用できる唯一のオプションです。
- 明示的なバージョン - リストのバージョンを指定すると、ポリシーでそのバージョンが使用されます。ポリシーは、バージョン設定を変更するまで、指定したバージョンにロックされたままとなります。バージョンを指定するには、CLI またはいずれかの SDK など、コンソールの外部でポリシーを定義する必要があります。

リストのバージョン設定の選択の詳細については、「コンテンツ監査セキュリティグループポリシーでのマネージドリストの使用」を参照してください。

コンテンツ監査セキュリティグループポリシーでのマネージドリストの使用

コンテンツ監査セキュリティグループポリシーを作成するときに、マネージド監査ポリシールールを使用するように選択できます。このオプションの一部の設定では、マネージドアプリケーションリストまたはプロトコルリストが必要です。これらの設定の例には、セキュリティグループルールで許可されるプロトコルや、アプリケーションがインターネットにアクセスできるプロトコルが含まれます。

マネージドリストを使用する各ポリシー設定には、次の制限が適用されます。

- Firewall Manager のマネージドリストは、任意の設定について最大 1 個指定できます。デフォルトでは、最大 1 個のカスタムリストを指定できます。カスタムリストの上限はソフトクォータなので、その制限の引き上げをリクエストできます。詳細については、「AWS Firewall Manager クォータ」を参照してください。
- コンソールで、Firewall Manager のマネージドリストを選択すると、バージョンを指定できません。ポリシーは、常に最新バージョンのリストを使用します。バージョンを指定するには、CLI またはいずれかの SDK など、コンソールの外部でポリシーを定義する必要があります。Firewall Manager マネージドリストのバージョニングについては、「マネージドリストのバージョニング」を参照してください。

コンソールを通じたコンテンツ監査セキュリティグループポリシーの作成については、「コンテンツ監査セキュリティグループポリシーの作成」を参照してください。

カスタムマネージドアプリケーションリストの作成

カスタムマネージドアプリケーションリストを作成するには

1. Firewall Manager AWS Management Console 管理者アカウントを使用してにサインインし、で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Application lists] (アプリケーションリスト) を選択します。
3. [Application lists] (アプリケーションリスト) ページで、[Create application list] (アプリケーションリストを作成) を選択します。
4. [Create application list] (アプリケーションリストを作成) ページで、リストに名前を付けます。プレフィックス `fms-` は Firewall Manager 用に予約されているため、使用しないでください。
5. プロトコルとポート番号を指定するか、[Type] (タイプ) ドロップダウンからアプリケーションを選択して、アプリケーションを指定します。アプリケーションの仕様に名前を付けます。
6. 必要に応じて [Add another] (別のものを追加) を選択し、リストが完成するまでアプリケーションに関する情報を入力します。

7. (オプション) リストにタグを適用します。
8. [Save] (保存) を選択してリストを保存し、[Application lists] (アプリケーションリスト) ページに戻ります。

カスタムマネージドプロトコルリストの作成

カスタムマネージドプロトコルリストを作成するには

1. Firewall Manager AWS Management Console 管理者アカウントを使用してにサインインし、で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Protocol lists] (プロトコルリスト) を選択します。
3. [Protocol lists] (プロトコルリスト) ページで、[Create protocol list] (プロトコルリストを作成) を選択します。
4. プロトコルリスト作成ページで、リストに名前を付けます。プレフィックス `fms-` は Firewall Manager 用に予約されているため、使用しないでください。
5. プロトコルを指定します。
6. 必要に応じて [Add another] (別のものを追加) を選択し、リストが完成するまでプロトコルに関する情報を入力します。
7. (オプション) リストにタグを適用します。
8. [Save] (保存) を選択してリストを保存し、[Protocol lists] (プロトコルリスト) ページに戻ります。

マネージドリストの表示

アプリケーションリストまたはプロトコルリストを表示するには

1. Firewall Manager AWS Management Console 管理者アカウントを使用してにサインインし、で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall

Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

> **Note**
>
> Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Application lists] (アプリケーションリスト) または [Protocol lists] (プロトコルリスト) を選択します。

   ページには、使用可能な、選択したタイプのリストがすべて表示されます。Firewall Manager ManagedListが管理するリストの列には Y が付いています。

3. リストの詳細を表示するには、その名前を選択します。詳細ページには、リストのコンテンツとタグが表示されます。

   Firewall Manager のマネージドリストの場合、[Version] (バージョン) ドロップダウンを選択して、使用可能なバージョンを確認することもできます。

カスタムマネージドリストの削除

カスタムマネージドリストを削除できます。Firewall Manager が管理するリストを編集または削除することはできません。

> **Note**
>
> 現在、Firewall Manager は、カスタムマネージドリストを削除すると、そのカスタムマネージドリストへの参照をチェックしません。つまり、アクティブなポリシーによって使用されている場合でも、カスタムマネージドアプリケーションリストまたはプロトコルリストを削除できます。これにより、ポリシーが機能しなくなる可能性があります。アプリケーションリストまたはプロトコルリストは、アクティブなポリシーによって参照されていないことを確認した後にのみ削除してください。

カスタムマネージドアプリケーションまたはプロトコルリストを削除するには

1. Firewall Manager AWS Management Console 管理者アカウントを使用してにサインインし、で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall

Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

> **Note**
>
> Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. 次の操作を実行して、削除するリストがいずれの監査セキュリティグループポリシーでも使用中ではないことを確認します。

   a. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。

   b. [AWS Firewall Manager policies] (ポリシー) ページで、監査セキュリティグループを選択して編集し、削除するカスタムリストへの参照をすべて削除します。

      監査セキュリティグループポリシーで使用されているカスタムマネージドリストを削除すると、そのリストを使用しているポリシーが機能しなくなる可能性があります。

3. ナビゲーションペインで、削除するリストのタイプに応じて、[Application lists] (アプリケーションリスト) または [Protocol lists] (プロトコルリスト) を選択します。
4. リストページで、削除するカスタムリストを選択し、[Delete] (削除) を選択します。

AWS WAF ポリシー

Firewall Manager AWS WAF ポリシーでは、リソース全体で使用する AWS WAF ルールグループを指定します。ポリシーを適用すると、ポリシー内のウェブ ACL 管理の設定内容に応じて、Firewall Manager がポリシー範囲内のアカウントにウェブ ACL を作成します。ポリシーで作成されたウェブ ACL では、個々のアカウントマネージャーは、Firewall Manager を通じて定義したルールグループに加えて、ルールとルールグループを追加できます。

Firewall Manager はウェブ ACL をどのように管理するか

Firewall Manager ACLs の管理設定を構成する方法、または API のデータ型で 設定に基づいてウェブ ACLs を作成します。 `optimizeUnassociatedWebACL` SecurityServicePolicyData

[関連付けられていないウェブ ACL の管理] を有効にした場合、Firewall Manager は、少なくとも 1 つのリソースがウェブ ACL を使用する場合のみ、ポリシー範囲内のアカウントにウェブ ACL を作成します。アカウントがポリシーの対象になるといつでも、少なくとも 1 つのリソースがウェブ ACL を使用する場合に、Firewall Manager はアカウントにウェブ ACL を自動的に作成します。関

連付けられていないウェブ ACL の管理を有効にすると、Firewall Manager はアカウント内の関連付けられていないウェブ ACL に 1 回だけクリーンアップを実行します。クリーンアップ中、Firewall Manager は、作成後に変更したウェブ ACL をすべてスキップします。例えば、ルールグループをウェブ ACL に追加したり、その設定を変更したりした場合などです。このクリーンアッププロセスには、数時間かかることがあります。Firewall Manager がウェブ ACL を作成した後、リソースがポリシー範囲から外れた場合、Firewall Manager はそのリソースとウェブ ACL の関連付けを解除しますが、関連付けられていないウェブ ACL はクリーンアップしません。Firewall Manager は、関連付けられていないウェブ ACL の管理を、ポリシーで初めて有効にした場合のみ、関連付けられていないウェブ ACL をクリーンアップします。

このオプションを有効にしない場合、Firewall Manager は関連付けられていないウェブ ACL を管理せず、ポリシーの範囲内にある各アカウントにウェブ ACL を自動的に作成します。

サンプリングと CloudWatch メトリクス

AWS Firewall Manager は、 AWS WAF ポリシー用に作成するウェブ ACLs とルールグループのサンプリングと Amazon CloudWatch メトリクスを有効にします。

ウェブ ACL の命名構造

Firewall Manager は、ポリシーのウェブ ACL を作成するときに、ウェブ ACL に `FMManagedWebACLV2-`*`policy name`*`-`*`timestamp`* という名前を付けます。タイムスタンプは UTC (ミリ秒) です。例えば、`FMManagedWebACLV2-MyWAFPolicyName-1621880374078` です。

> **Note**
>
> 高度な自動アプリケーションレイヤー DDoS 緩和で設定されたリソースが AWS WAF ポリシーの範囲内にある場合、Firewall Manager は AWS WAF ポリシーによって作成されたウェブ ACL をリソースに関連付けることができません。

AWS WAF ポリシー内のルールグループ

Firewall Manager AWS WAF ポリシーによって管理されるウェブ ACLs には、3 つのルールセットが含まれています。これらのセットにより、ウェブ ACL 内のルールおよびルールグループの優先順位は上がります。

- Firewall Manager AWS WAF ポリシーで定義した最初のルールグループは、最初にこれらのルールグループ AWS WAF を評価します。

- アカウントマネージャーがウェブ ACL で定義したルールおよびルールグループ。 AWS WAF は次にアカウントマネージドルールまたはルールグループを評価します。
- Firewall Manager AWS WAF ポリシーで定義した最後のルールグループは、これらのルールグループを最後に AWS WAF 評価します。

これらのルールの各セット内で、 はセット内の優先順位設定に従って、通常どおりルールとルールグループ AWS WAF を評価します。

ポリシーの最初と最後のルールグループセットでは、ルールグループのみを追加できます。マネージドルールグループを使用できます。マネージドルールグループは、 AWS マネージドルールと AWS Marketplace 販売者がユーザーに代わって作成および維持します。独自のルールグループを管理して使用することもできます。これらのすべてのオプションの詳細については、「AWS WAF ルールグループ」を参照してください。

独自のルールグループを使用する場合は、Firewall Manager AWS WAF ポリシーを作成する前にそれらのグループを作成します。ガイダンスについては、「独自のルールグループの管理」を参照してください。個々のカスタムルールを使用するには、独自のルールグループを定義し、その中にルールを定義してから、ポリシーでそのルールグループを使用する必要があります。

Firewall Manager で管理する最初と最後の AWS WAF ルールグループの名前は`POSTFMManaged-`、それぞれ `PREFMManaged-`または で始まり、その後に Firewall Manager ポリシー名とルールグループ作成タイムスタンプが UTC ミリ秒単位で続きます。例えば `PREFMManaged-MyWAFPolicyName-1621880555123` です。

がウェブリクエスト AWS WAF を評価する方法については、「」を参照してくださいウェブ ACL ルールおよびルールグループの評価。

Firewall Manager AWS WAF ポリシーを作成する手順については、「」を参照してくださいの AWS Firewall Manager ポリシーの作成 AWS WAF。

Firewall Manager は、 AWS WAF ポリシーに定義したルールグループのサンプリングと Amazon CloudWatch メトリクスを有効にします。

個々のアカウント所有者は、ポリシーのマネージドウェブ ACL に追加するルールまたはルールグループのメトリクスとサンプリング設定を完全に制御できます。

AWS WAF ポリシーのログ記録の設定

AWS WAF ポリシーの集中ログ記録を有効にして、組織内のウェブ ACL によって分析されるトラフィックに関する詳細情報を取得できます。ログの情報には、 が AWS リソースからリクエストを

AWS WAF 受信した時間、リクエストに関する詳細情報、および各リクエストがすべての範囲内アカウントから一致したルールのアクションが含まれます。ログは、Amazon Data Firehose データストリームまたは Amazon Simple Storage Service (S3) バケットに送信できます。 AWS WAF ログ記録の詳細については、「 AWS WAF デベロッパーガイドAWS WAF ウェブ ACL トラフィックのログ記録」の「」を参照してください。

> **Note**
>
> AWS Firewall Manager は AWS WAF 、 Classic ではなく AWS WAFV2、 に対してこのオプションをサポートします。

トピック

- ログ記録の送信先
- ログ作成の有効化
- ログ記録の無効化

ログ記録の送信先

このセクションでは、 AWS WAF ポリシーログの送信先として選択できるログ記録先について説明します。各セクションでは、送信先の種類のログを設定するためのガイダンスと、送信先の種類に固有の動作に関する情報を提供します。ログ記録の送信先を設定したら、Firewall Manager AWS WAF ポリシーにその仕様を指定して、ログ記録を開始できます。

Firewall Manager では、ログ記録設定を作成した後でログの失敗を可視化することはできません。ログ配信が意図したとおりに機能していることを確認するのはユーザーの責任になります。

> **Note**
>
> Firewall Manager は、組織のメンバーアカウントの既存のログ記録設定を変更しません。

トピック

- Amazon Data Firehose データストリーム
- Amazon Simple Storage Service バケット

Amazon Data Firehose データストリーム

このトピックでは、ウェブ ACL トラフィックログを Amazon Data Firehose データストリームに送信するための情報を提供します。

Amazon Data Firehose ログ記録を有効にすると、Firewall Manager は、ポリシーのウェブ ACLs から、ストレージ送信先を設定した Amazon Data Firehose にログを送信します。ログ記録を有効にすると、 は Kinesis Data Firehose の HTTPS エンドポイントを介して、設定された各ウェブ ACL のログを、設定されたストレージ宛先に AWS WAF 配信します。それを使用する前に、配信ストリームをテストして、組織のログに対応するのに十分なスループットがあることを確認します。Amazon Kinesis Data Firehose を作成し、保存されたログを確認する方法の詳細については、「Amazon Data Firehose とは」を参照してください。

Kinesis によるログ記録を正常に有効化するには、以下の許可が付与されている必要があります。

- `iam:CreateServiceLinkedRole`
- `firehose:ListDeliveryStreams`
- `wafv2:PutLoggingConfiguration`

AWS WAF ポリシーで Amazon Data Firehose ログ記録の送信先を設定すると、Firewall Manager は、Firewall Manager 管理者アカウントにポリシーのウェブ ACL を次のように作成します。

- Firewall Manager は、アカウントがポリシーの範囲内にあるかどうかにかかわらず、Firewall Manager 管理者アカウントにウェブ ACL を作成します。
- ウェブ ACL でログ記録が有効になり、`FMManagedWebACLV2-Logging`*`policy name`*`-`*`timestamp`* というログ名が付けられます。タイムスタンプは、ウェブ ACL 用にログが有効になった UTC 時間 (ミリ秒) です。例えば、`FMManagedWebACLV2-LoggingMyWAFPolicyName-1621880565180` です。ウェブ ACL には、ルールグループおよび関連するリソースはありません。
- ウェブ ACL には、 AWS WAF 料金ガイドラインに従って課金されます。詳細については、「AWS WAF の料金」を参照してください。
- ポリシーを削除すると、Firewall Manager はウェブ ACL を削除します。

サービスにリンクされたロールおよび `iam:CreateServiceLinkedRole` 許可の詳細については、「のサービスにリンクされたロールの使用 AWS WAF」を参照してください。

配信ストリームの作成の詳細については、「Amazon Data Firehose 配信ストリームの作成」を参照してください。

Amazon Simple Storage Service バケット

このトピックは、ウェブ ACL トラフィックログの Amazon S3 バケットへの送信に関する情報を提供します。

ログ記録の出力先として選択するバケットは、Firewall Manager の管理者アカウントが所有している必要があります。ログ記録用に Amazon S3 バケットを作成する際の要件、およびバケット命名における要件の情報については、「AWS WAF デベロッパーガイド」の「Amazon Simple Storage Service」を参照してください。

結果整合性

Amazon S3 ログ記録先で設定された AWS WAF ポリシーを変更すると、Firewall Manager はバケットポリシーを更新して、ログ記録に必要なアクセス許可を追加します。その場合、Firewall Manager は Amazon Simple Storage Service が従う last-writer-wins セマンティクスモデルとデータ整合性モデルに従います。Firewall Manager コンソールまたは PutPolicy API を使用して Amazon S3 の送信先に複数のポリシーを同時に更新すると、一部のアクセス許可が保存されない場合があります。Amazon S3 におけるデータ整合性モデルの詳細については、「Amazon Simple Storage Service ユーザーガイド」の「 Amazon S3 のデータ整合性モデル」を参照してください。

Amazon S3 に対しログを発行するためのアクセス許可

AWS WAF ポリシーで Amazon S3 バケットのウェブ ACL トラフィックログ記録を設定するには、次のアクセス許可設定が必要です。Amazon S3 をログ記録の出力先として設定すると、Firewall Manager は、サービスがバケットに対しログを公開することを許可するために、これらのアクセス許可を Amazon S3 バケットに自動的にアタッチします。ログ記録と Firewall Manager リソースへのよりきめ細かいアクセスを管理したい場合は、自分でこれらの許可を設定できます。アクセス許可の管理については、「IAM ユーザーガイド」の「AWS リソースのアクセス管理」を参照してください。 AWS WAF 管理ポリシーの詳細については、「」を参照してくださいAWS の マネージドポリシー AWS WAF。

```
{
    "Version": "2012-10-17",
    "Id": "AWSLogDeliveryForFirewallManager",
    "Statement": [
        {
            "Sid": "AWSLogDeliveryAclCheckFMS",
            "Effect": "Allow",
```

```
            "Principal": {
                "Service": "delivery.logs.amazonaws.com"
            },
            "Action": "s3:GetBucketAcl",
            "Resource": "arn:aws:s3:::aws-waf-DOC-EXAMPLE-BUCKET"
        },
        {
            "Sid": "AWSLogDeliveryWriteFMS",
            "Effect": "Allow",
            "Principal": {
                "Service": "delivery.logs.amazonaws.com"
            },
            "Action": "s3:PutObject",
            "Resource": "arn:aws:s3:::aws-waf-logs-DOC-EXAMPLE-BUCKET/policy-id/
AWSLogs/*",
            "Condition": {
                "StringEquals": {
                    "s3:x-amz-acl": "bucket-owner-full-control"
                }
            }
        }
    ]
}
```

サービス間での混乱した代理問題を防ぐためには、aws:SourceArn および aws:SourceAccount グローバル条件コンテキストキーを、バケットのポリシーに追加できます。これらのキーを追加する場合は、ログの出力先を設定する際に Firewall Manager により作成されたポリシーを変更します。あるいは、よりきめ細かな制御が必要な場合には、独自のポリシーを作成することができます。これらの条件をログ記録出力先のポリシーに追加した場合、Firewall Manager は、混乱した代理に関する保護の検証またはモニタリングを行いません。混乱した代理問題の詳細については、「IAM ユーザーガイド」の「混乱する代理問題」を参照してください。

sourceAccount および sourceArn のプロパティを追加すると、バケットポリシーのサイズが増加します。sourceAccount および sourceArn プロパティから成る長いリストを追加する場合は、Amazon S3 のバケットポリシーのサイズが上限を超えないように注意してください。

次の例で、 ロールポリシーで aws:SourceArn および aws:SourceAccount のグローバル条件コンテキストを使用して、混乱した代理問題を防止する方法を示します。を組織内のメンバーIDs *member-account-id*に置き換えます。

```
{
   "Version":"2012-10-17",
```

```
  "Id":"AWSLogDeliveryForFirewallManager",
  "Statement":[
     {
        "Sid":"AWSLogDeliveryAclCheckFMS",
        "Effect":"Allow",
        "Principal":{
           "Service":"delivery.logs.amazonaws.com"
        },
        "Action":"s3:GetBucketAcl",
        "Resource":"arn:aws:s3:::aws-waf-logs-DOC-EXAMPLE-BUCKET",
        "Condition":{
           "StringEquals":{
              "aws:SourceAccount":[
                 "member-account-id",
                 "member-account-id"
              ]
           },
           "ArnLike":{
              "aws:SourceArn":[
                 "arn:aws:logs:*:member-account-id:*",
                 "arn:aws:logs:*:member-account-id:*"
              ]
           }
        }
     },
     {
        "Sid":"AWSLogDeliveryWriteFMS",
        "Effect":"Allow",
        "Principal":{
           "Service":"delivery.logs.amazonaws.com"
        },
        "Action":"s3:PutObject",
        "Resource":"arn:aws:s3:::aws-waf-logs-DOC-EXAMPLE-BUCKET/policy-id/AWSLogs/*",
        "Condition":{
           "StringEquals":{
              "s3:x-amz-acl":"bucket-owner-full-control",
              "aws:SourceAccount":[
                   "member-account-id",
                 "member-account-id"
              ]
           },
           "ArnLike":{
              "aws:SourceArn":[
                 "arn:aws:logs:*:member-account-id-1:*",
```

```
                "arn:aws:logs:*:member-account-id-2:*"
            ]
        }
    }
  }
]
}
```

Amazon S3 バケットのサーバー側の暗号化

Amazon S3 サーバー側の暗号化を有効にするか、S3 バケットで AWS Key Management Service カスタマーマネージドキーを使用できます。 AWS WAF ログに Amazon S3 バケットでデフォルトの Amazon S3 暗号化を使用する場合は、特別なアクションを実行する必要はありません。ただし、お客様が用意した暗号化キーを使用して保管中の Amazon S3 データを暗号化する場合は、 AWS Key Management Service キーポリシーに次のアクセス許可ステートメントを追加する必要があります。

```
{
            "Sid": "Allow Logs Delivery to use the key",
            "Effect": "Allow",
            "Principal": {
                "Service": "delivery.logs.amazonaws.com"
            },
            "Action": [
                "kms:Encrypt",
                "kms:Decrypt",
                "kms:ReEncrypt*",
                "kms:GenerateDataKey*",
                "kms:DescribeKey"
            ],
            "Resource": "*"
}
```

Amazon S3 でお客様が提供する暗号化キーの使用の詳細については、「Amazon Simple Storage Service ユーザーガイド」の「お客様が指定したキーによるサーバー側の暗号化 (SSE-C) の使用」を参照してください。

ログ作成の有効化

次の手順では、Firewall Manager コンソールで AWS WAF ポリシーのログ記録を有効にする方法について説明します。

AWS WAF ポリシーのログ記録を有効にするには

1. ログ記録を有効にする前に、次のようにログ記録の出力先リソースを設定する必要があります。

   - Amazon Kinesis Data Streams - Firewall Manager 管理者アカウントを使用して Amazon Data Firehose を作成します。プレフィックス `aws-waf-logs-` で始まる名前を使用します。例えば、`aws-waf-logs-firewall-manager-central` です。自分が操作しているリージョン内で、PUT ソースを使用して Data Firehose を作成します。Amazon のログをキャプチャする場合は CloudFront、米国東部 (バージニア北部) に Firehose を作成します。それを使用する前に、配信ストリームをテストして、組織のログに対応するのに十分なスループットがあることを確認します。詳細については、「Creating an Amazon Data Firehose Delivery Stream」を参照してください。
   - Amazon Simple Storage Service バケット – 「AWS WAF デベロッパーガイド」の「Amazon Simple Storage Service」トピックにあるガイドラインに従って、Amazon S3 バケットを作成します。また、Amazon S3 に対しログを発行するためのアクセス許可 に一覧されているアクセス許可を使用して、Amazon S3 バケットを設定する必要があります。

2. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

3. ナビゲーションペインで、[Security Policies] (セキュリティポリシー) を選択します。
4. ログ記録を有効にする AWS WAF ポリシーを選択します。 AWS WAF ログ記録の詳細については、「AWS WAF ウェブ ACL トラフィックのログ記録」を参照してください。
5. [Policy details] (ポリシーの詳細) タブの [Policy rules] (ポリシールール) セクションで、[Edit] (編集) を選択します。
6. [ログ記録設定] で、[ログ記録を有効にする] を選択してログ記録をオンにします。ログ記録は、ウェブ ACL で分析されるトラフィックに関する詳細情報を提供します。[ログの出力先] を選択し、設定したログ記録の送信先を選択します。名前が `aws-waf-logs-` で始まるログ記録先を選択する必要があります。 AWS WAF ログ記録の送信先の設定については、「」を参照してくださいAWS WAF ポリシーのログ記録の設定。

7. (オプション) 特定のフィールドとその値がログに含まれることを希望しない場合には、このフィールドをマスキングします。マスキングするフィールドを選び、[Add] (追加) を選択します。必要に応じて手順を繰り返し、追加のフィールドをマスキングします。マスキングされたフィールドは、ログに REDACTED と表示されます。例えば、[URI] フィールドをマスキングすると、ログの [URI] フィールドは REDACTED となります。
8. (オプション) すべてのリクエストをログに送信しない場合は、フィルタリング条件と動作を追加します。[Filter logs] (ログをフィルタリング) で、適用する各フィルターについて [Add filter] (フィルターを追加) を選択し、次にフィルター基準を選択して、基準に一致するリクエストを保持するかドロップするかを指定します。フィルターの追加が完了したら、必要に応じて、[Default logging behavior] (デフォルトのログ記録動作) を変更します。詳細については、「AWS WAF デベロッパーガイド」の 「ウェブ ACL ログ記録設定」を参照してください。
9. [次へ] をクリックします。
10. 設定を確認し、[Save] (保存) を選択してポリシーに対する変更を保存します。

ログ記録の無効化

次の手順では、Firewall Manager コンソールで AWS WAF ポリシーのログ記録を無効にする方法について説明します。

AWS WAF ポリシーのログ記録を無効にするには

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security Policies] (セキュリティポリシー) を選択します。
3. ログ記録を無効にする AWS WAF ポリシーを選択します。
4. [Policy details] (ポリシーの詳細) タブの [Policy rules] (ポリシールール) セクションで、[Edit] (編集) を選択します。
5. [Logging configuration status] (ログ記録設定ステータス) で、[Disabled] (無効) を選択します。

6. [Next] (次へ) を選択します。
7. 設定を確認し、[Save] (保存) を選択してポリシーに対する変更を保存します。

AWS Shield Advanced ポリシー

Firewall Manager AWS Shield ポリシーでは、保護するリソースを選択します。auto 修復を有効にしてポリシーを適用すると、まだウェブ ACL に関連付けられていないスコープ内のリソースごとに、Firewall Manager AWS WAF AWS WAF は空のウェブ ACL を関連付けます。空のウェブ ACL は、Shield によるモニタリングの目的のために使用されます。その後、他のウェブ ACL をリソースに関連付けると、Firewall Manager は、空のウェブ ACL の関連付けを削除します。

> **Note**
>
> AWS WAF ポリシーの範囲内のリソースが、アプリケーション層の自動DDoS軽減が設定されたShield Advancedポリシーの範囲に入ると、Firewall Manager は、ポリシーによって作成されたウェブ ACLを関連付けた後にのみShield Advanced保護を適用します。 AWS WAF

Shield ポリシーで関連付けられていないウェブ ACL AWS Firewall Manager を管理する方法

Firewall Manager が関連付けられていないウェブ ACL を自動的に管理するかどうかは、ポリシーの [関連付けられていないウェブ ACL の管理] 設定または API `optimizeUnassociatedWebACLs` SecurityServicePolicyDataのデータタイプの設定で設定できます。[関連付けられていないウェブ ACL の管理]をポリシーで有効にした場合、Firewall Manager は、少なくとも 1 つのリソースがウェブ ACL を使用する場合のみ、ポリシー範囲内のアカウントにウェブ ACL を作成します。アカウントがポリシーの対象になるといつでも、少なくとも 1 つのリソースがウェブ ACL を使用する場合に、Firewall Manager はアカウントにウェブ ACL を自動的に作成します。

関連付けられていないウェブ ACL の管理を有効にすると、Firewall Manager はアカウント内の関連付けられていないウェブ ACL に 1 回だけクリーンアップを実行します。このクリーンアッププロセスには、数時間かかることがあります。Firewall Manager がウェブ ACL を作成した後、リソースがポリシーの範囲から外れても、Firewall Manager はそのリソースとウェブ ACL との関連付けを解除しません。Firewall Manager にウェブ ACL をクリーンアップさせたい場合は、最初にウェブ ACL から手動でリソースの関連付けを解除してから、ポリシーの[関連付けられていないウェブ ACL の管理] オプションを有効にする必要があります。

このオプションを有効にしない場合、Firewall Manager は関連付けられていないウェブ ACL を管理せず、ポリシーの範囲内にある各アカウントにウェブ ACL を自動的に作成します。

SShield AWS Firewall Manager ポリシーの範囲変更を管理する方法

ポリシースコープ設定の変更、リソースのタグの変更、組織からのアカウントの削除など、さまざまな変更により、アカウントとリソースは AWS Firewall Manager Shield Advanced ポリシーの範囲外になる可能性があります。ポリシーの範囲設定の一般情報については、「AWS Firewall Manager ポリシーの範囲」を参照してください。

AWS Firewall Manager Shield Advancedポリシーでは、アカウントまたはリソースが範囲外になると、Firewall Manager はアカウントまたはリソースの監視を停止します。

アカウントが組織から削除されて範囲外になった場合でも、そのアカウントは引き続き Shield Advanced にサブスクライブされます。アカウントは一括請求ファミリーのメンバーではなくなるため、アカウントには按分計算での Shield Advanced サブスクリプション料金が発生します。一方、範囲外になったが、組織に残っているアカウントについては、追加料金が発生しません。

リソースが範囲外になった場合でも、そのリソースは Shield Advanced によって引き続き保護され、Shield Advanced データ転送料金が引き続き発生します。

アプリケーションレイヤー DDoS 自動緩和

Shield アドバンスドポリシーを Amazon CloudFront ディストリビューションまたはアプリケーションロードバランサーに適用する場合、ポリシーで Shield アドバンスド自動アプリケーションレイヤー DDoS 軽減を設定することができます。

Shield Advanced 自動緩和については、「Shield Advanced アプリケーションレイヤー DDoS 自動緩和」を参照してください。

Shield Advanced アプリケーションレイヤー DDoS 自動緩和には、次の要件があります。

- アプリケーションレイヤーの自動DDoS軽減は、 CloudFront Amazonディストリビューションとアプリケーションロードバランサーでのみ機能します。

  Amazon CloudFront ディストリビューションに Shield アドバンスドポリシーを適用する場合、グローバルリージョン用に作成する Shield アドバンスドポリシーでこのオプションを選択できます。Application Load Balancer に保護を適用する場合、Firewall Manager がサポートする任意のリージョンにポリシーを適用できます。
- アプリケーションレイヤーの DDoS の自動軽減は、最新バージョンの (v2) を使用して作成されたウェブ ACL でのみ機能します。 AWS WAF

そのため、 AWS WAF 従来のウェブ ACL を使用するポリシーがある場合は、そのポリシーを最新バージョンのを自動的に使用する新しいポリシーに置き換えるか、Firewall Manager に既存のポリシー用に新しいバージョンの Web ACL を作成してもらい、それらを使用するように切り替える必要があります。 AWS WAFオプションの詳細については、「AWS WAF クラシックウェブ ACL を最新バージョンのウェブ ACL に置き換えます。」を参照してください。

自動緩和設定

Firewall Manager Shield Advanced ポリシーのアプリケーションレイヤー DDoS 自動緩和オプションは、ポリシーの範囲内のアカウントおよびリソースに Shield Advanced 自動緩和機能を適用します。Shield Advanced 機能の詳細については、「Shield Advanced アプリケーションレイヤー DDoS 自動緩和」を参照してください。

Firewall Manager CloudFront でポリシーの対象となるディストリビューションまたはアプリケーションロードバランサーの自動緩和を有効または無効にするか、Shield Advancedの自動緩和設定をポリシーに無視させるかを選択できます。

- [有効化] - 自動緩和を有効にする場合は、一致するウェブリクエストを Shield Advanced 緩和ルールがカウントまたはブロックするかも指定します。Firewall Manager は、自動緩和が有効になっていないか、ポリシーに指定したルールアクションと一致しないルールアクションを使用している場合に、範囲内のリソースを非準拠としてマークします。自動修復のポリシーを設定すると、Firewall Manager は、必要に応じて非準拠のリソースを更新します。
- [無効化] - 自動緩和を無効にすることを選択した場合、Firewall Manager は、自動緩和が有効になっている範囲内のリソースを非準拠としてマークします。自動修復のポリシーを設定すると、Firewall Manager は、必要に応じて非準拠のリソースを更新します。
- [無視] – 自動緩和を無視することを選択した場合、Firewall Manager は、ポリシーの修復アクティビティを実行するときに Shield ポリシーの自動緩和設定を考慮しません。この設定では、Shield Advanced を通じて自動緩和を有効または無効にできます。Firewall Manager によってこれらの設定を上書きされることはありません。この設定は、Shield Advanced を通じて管理される Classic Load Balancers や Elastic IP リソースには適用されません。Shield Advanced は現在、これらのリソースの L7 自動緩和をサポートしていないためです。

AWS WAF クラシックウェブ ACL を最新バージョンのウェブ ACL に置き換えます。

アプリケーションレイヤーの DDoS の自動軽減は、最新バージョンの AWS WAF (v2) を使用して作成されたウェブ ACL でのみ機能します。

Shield Advanced ポリシーのウェブ ACL バージョンを確認するには、「Shield AWS WAF アドバンスドポリシーで使用されているバージョンの確認」を参照してください。

Shield Advancedポリシーで自動緩和機能を使用したい場合で、 AWS WAF ポリシーが現在クラシックウェブACLを使用している場合は、新しいShield アドバンスドポリシーを作成して現在のポリシーに置き換えるか、このセクションで説明されているオプションを使用して以前のバージョンのウェブACLを現在のSShield アドバンスドポリシー内の新しい (v2) ウェブ ACLに置き換えることができます。新しいポリシーでは、常に最新バージョンのを使用してウェブ ACL が作成されます。 AWS WAFポリシー全体を置き換えた場合、ポリシーを削除すると、Firewall Manager で以前のバージョンのウェブ ACL もすべて削除できます。このセクションの残りの部分では、既存のポリシー内のウェブ ACL を置き換えるためのオプションについて説明します。

Amazon CloudFront リソースの既存の Shield Advanced ポリシーを変更すると、Firewall Manager は、v2 ウェブ ACL をまだ持っていないスコープ内のアカウントで、ポリシー用の新しい空 AWS WAF (v2) ウェブ ACL を自動的に作成できます。Firewall Manager が新しいウェブ ACL を作成するとき、 AWS WAF ポリシーに同じアカウントにすでにクラシックウェブ ACL がある場合、Firewall Manager は新しいバージョンのウェブ ACL を既存のウェブ ACL と同じデフォルトアクション設定で構成します。 AWS WAF 既存のクラシックウェブ ACL がない場合、Firewall Manager は新しいウェブ ACL Allow にデフォルトアクションを設定します。Firewall Manager が新しいウェブ ACL を作成した後、 AWS WAF コンソールで必要に応じてカスタマイズできます。

次のポリシー設定オプションのいずれかを選択すると、Firewall Manager は、まだそれらを持たない範囲内のアカウント用に新しい (v2)ウェブ ACL を作成します。

- アプリケーションレイヤー DDoS 自動緩和を有効または無効にする場合。この選択のみの場合、Firewall Manager が新しいウェブ ACL を作成するだけであり、ポリシーの範囲内リソース上の既存の AWS WAF Classicウェブ ACL の関連付けを置き換えることはありません。
- 自動修復のポリシーアクションを選択し、 AWS WAF クラシックウェブ ACL を AWS WAF (v2) ウェブ ACL に置き換えるオプションを選択すると、アプリケーションレイヤー DDoS 自動緩和の設定の選択内容にかかわらず、以前のバージョンのウェブ ACL を置き換えることを選択できます。

  置換オプションを選択すると、Firewall Manager は必要に応じて新しいバージョンのウェブ ACL を作成し、ポリシーの範囲内のリソースについて次の操作を実行します。
  - リソースが他のアクティブな Firewall Manager ポリシーからウェブ ACL に関連付けられている場合、Firewall Manager は関連付けのみを残します。
  - それ以外の場合、Firewall Manager AWS WAF はクラシックウェブ ACL との関連付けをすべて削除し、リソースをポリシーの AWS WAF (v2) ウェブ ACL に関連付けます。

必要に応じて、Firewall Manager で以前のバージョンのウェブ ACL を新しいバージョンのウェブ ACL に置き換えることを選択できます。以前にポリシーの AWS WAF Classicウェブ ACL をカスタマイズしたことがある場合は、Firewall Manager に置換ステップを実行させることを選択する前に、新しいバージョンのウェブ ACL を同等の設定に更新できます。

ポリシーのウェブ ACL のいずれのバージョンにも、同じバージョンのコンソールまたは Classic のコンソールからアクセスできます。 AWS WAF AWS WAF

Firewall Manager は、ポリシー自体を削除するまで、 AWS WAF 置き換えられたクラシックウェブ ACL を削除しません。 AWS WAF クラシック Web ACL がポリシーで使用されなくなったら、必要に応じて削除できます。

Shield AWS WAF アドバンスドポリシーで使用されているバージョンの確認

Firewall Manager Shield Advancedポリシーがどのバージョンを使用しているかは、 AWS WAF AWS Config ポリシーのサービスにリンクされたルール内のパラメータキーを確認することで確認できます。 AWS WAF 使用中のバージョンが最新の場合、パラメータキーにはとが含まれますpolicyId。webAclArn以前のバージョンの AWS WAF Classic の場合、webAclIdパラメータキーにはとが含まれますresourceTypes。

AWS Config このルールには、ポリシーが対象範囲内のリソースで現在使用しているウェブ ACL のキーのみが一覧表示されます。

Firewall Manager Shield AWS WAF アドバンスドポリシーがどのバージョンを使用しているかを確認するには

1. Shield Advanced ポリシーのポリシー ID を取得します。

    a. Firewall Manager AWS Management Console 管理者アカウントを使用してにサインインし、でFirewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

    b. ナビゲーションペインで、[Security Policies] (セキュリティポリシー) を選択します。

    c. ポリシーのリージョンを選択します。 CloudFront ディストリビューションの場合はですGlobal。

    d. 必要なポリシーを見つけて、その [Policy ID] (ポリシー ID) の値をコピーします。

        ポリシー ID の例: 1111111-2222-3333-4444-a55aa5aaa555。

2. ポリシー ID AWS Config を文字列に追加してポリシーのルール名を作成します。`FMManagedShieldConfigRule`

   AWS Config ルール名の例:. `FMManagedShieldConfigRule1111111-2222-3333-4444-a55aa5aaa555`

3. AWS Config 関連するルールのパラメータで、`policyId`およびという名前のキーを検索しますwebAclArn。

   a. https://console.aws.amazon.com/config/ AWS Config でコンソールを開きます。
   b. ナビゲーションペインで [Rules (ルール)] を選択します。
   c. リストでFirewall Manager AWS Config ポリシーのルール名を見つけて選択します。ルールのページが開きます。
   d. [Rule details] (ルールの詳細) の [Parameters] (パラメータ) セクションで、キーを探します。`policyId` と `webAclArn` という名前のキーが見つかった場合、ポリシーは最新バージョンの AWS WAFを使用して作成されたウェブ ACL を使用します。`webAclId`and という名前のキーが見つかった場合`resourceTypes`、そのポリシーは以前のバージョンの AWS WAF Classic を使用して作成された Web ACL を使用します。

セキュリティグループポリシー

AWS Firewall Manager セキュリティグループポリシーを使用して、 で組織の Amazon Virtual Private Cloud セキュリティグループを管理できます AWS Organizations。一元管理されたセキュリティグループポリシーは、組織全体、またはアカウントとリソースの特定のサブセットに適用できます。さらに、監査および使用状況セキュリティグループポリシーを使用して、組織内で使用中のセキュリティグループポリシーをモニタリングおよび管理することもできます。

Firewall Manager は、ポリシーを継続的に維持し、組織全体で追加または更新されるたびにアカウントとリソースに適用します。の詳細については AWS Organizations、「 AWS Organizations ユーザーガイド」を参照してください。

Amazon Virtual Private Cloud セキュリティグループの詳細については、「Amazon VPC ユーザーガイド」の「VPC のセキュリティグループ」を参照してください。

Firewall Manager セキュリティグループポリシーを使用して、 AWS 組織全体で次の操作を実行できます。

- 指定したアカウントとリソースに共通セキュリティグループを適用します。
- セキュリティグループルールを監査し、準拠していないルールを見つけて修復します。

- セキュリティグループの使用状況を監査し、未使用および冗長なセキュリティグループをクリーンアップします。

このセクションでは、Firewall Manager セキュリティグループポリシーの仕組みについて説明し、使用する際のガイダンスを提供します。セキュリティグループポリシーを作成する手順については、「AWS Firewall Manager ポリシーの作成」を参照してください。

共通セキュリティグループポリシー

Firewall Manager では、共通セキュリティグループポリシーを使用して、組織全体のアカウントおよびリソースに対するセキュリティグループの一元管理された関連付けが提供されます。組織内でポリシーを適用する場所と方法を指定します。

次のリソースタイプに共通セキュリティグループポリシーを適用できます。

- Amazon Elastic Compute Cloud (Amazon EC2) インスタンス
- Elastic Network Interface
- Application Load Balancer
- Classic Load Balancer

コンソールを使用して共通セキュリティグループポリシーを作成する方法については、「共通セキュリティグループポリシーの作成」を参照してください。

共有 VPC

共通セキュリティグループポリシーのポリシーの範囲設定で、共有 VPC を含めるよう選択できます。この選択肢には、別のアカウントによって所有され、範囲内のアカウントと共有される VPC が含まれます。範囲内のアカウントが所有する VPC は、常に含まれます。共有 VPC の詳細については、「Amazon VPC ユーザーガイド」の「共有 VPC の使用」を参照してください。

共有 VPC を含めるには、次の注意事項が適用されます。これらに加えて、セキュリティグループポリシーの注意点と制限事項 のセキュリティグループポリシーに関する一般的な注意事項も適用されます。

- Firewall Manager は、範囲内の各アカウントの VPC にプライマリセキュリティグループをレプリケートします。共有 VPC の場合、Firewall Manager は VPC が共有されている範囲内のアカウントごとに、プライマリセキュリティグループを 1 回 レプリケートします。これにより、単一の共有 VPC に複数のレプリカが作成される可能性があります。

- 新しい共有 VPC を作成する場合、ポリシーの範囲内にある VPC に少なくとも 1 つのリソースを作成するまで、Firewall Manager セキュリティグループポリシーの詳細にその共有VPC は表示されません。
- 共有 VPC を有効にしたポリシーで共有 VPC を無効にすると、共有 VPC において、Firewall Manager は、リソースに関連付けられていないレプリカセキュリティグループを削除します。Firewall Manager は、残りのレプリカセキュリティグループをそのままの状態にしますが、それらの管理を停止します。これらの残りのセキュリティグループを削除するには、各共有 VPC インスタンスで手動で管理する必要があります。

プライマリセキュリティグループ

共通セキュリティグループポリシーごとに、1 つ以上のプライマリセキュリティグループ AWS Firewall Manager を指定します。

- プライマリセキュリティグループは、Firewall Manager 管理者アカウントによって作成される必要があり、アカウント内の任意の Amazon VPC インスタンスに存在できます。
- プライマリセキュリティグループは、Amazon Virtual Private Cloud (Amazon VPC) または Amazon Elastic Compute Cloud (Amazon EC2) を通じて管理します。詳細については、「Amazon VPC ユーザーガイド」の「セキュリティグループの操作」を参照してください。
- Firewall Manager セキュリティグループポリシーのプライマリとして、1 つ以上のセキュリティグループを指定できます。デフォルトでは、ポリシー内で許可されているセキュリティグループの数は 1 つですが、リクエストを送信して増やすことができます。詳細については、「AWS Firewall Manager クォータ」を参照してください。

ポリシールールの設定

共通セキュリティグループポリシーのセキュリティグループとリソースに対して、次の変更管理動作の 1 つ以上を選択できます。

- ローカルユーザーがレプリカセキュリティグループに対して行った変更を特定し、レポートします。
- ポリシーの範囲内にある AWS リソースから他のセキュリティグループの関連付けを解除します。
- プライマリグループからレプリカセキュリティグループにタグを配布します。

> **Important**
>
> Firewall Manager は、 AWS サービスによって追加されたシステムタグをレプリカセキュリティグループに配布しません。システムタグは aws: プレフィックスで始まります。また、ポリシーに組織のタグポリシーと矛盾するタグがある場合は、Firewall Manager が既存のセキュリティグループでのタグ更新や、新しいセキュリティグループの作成を行うことはありません。タグポリシーの詳細については、「 ユーザーガイド」の「タグポリシー AWS Organizations 」を参照してください。

- プライマリグループからレプリカセキュリティグループにセキュリティグループの参照を配布します。

  これにより、指定したセキュリティグループの VPC に関連するインスタンスに対して、スコープ内の全リソースで共通のセキュリティグループの参照ルールを簡単に確立することができます。このオプションを有効にすると、Firewall Manager は、セキュリティグループが Amazon Virtual Private Cloud のピアセキュリティグループを参照する場合にのみ、セキュリティグループ参照を伝播します。レプリカセキュリティグループがピアセキュリティグループを正しく参照していない場合、Firewall Manager はこれらのレプリケートされたセキュリティグループを非準拠としてマークします。Amazon VPC でピアセキュリティグループを参照する方法については、「Amazon VPC ピアリングガイド」の「ピアセキュリティグループを参照するようにセキュリティグループを更新する」を参照してください。 https://docs.aws.amazon.com/vpc/latest/peering/

  このオプションを有効にしない場合、Firewall Manager はセキュリティグループ参照をレプリカセキュリティグループに伝達しません。Amazon VPC での VPC ピアリングの詳細については、「Amazon VPC ピアリングガイド」を参照してください。

ポリシーの作成と管理

共通セキュリティグループポリシーを作成すると、Firewall Manager はポリシーの範囲内のすべての Amazon VPC インスタンスにプライマリセキュリティグループをレプリケートし、レプリケートされたセキュリティグループをポリシーの範囲内にあるアカウントおよびリソースに関連付けます。プライマリセキュリティグループを変更すると、Firewall Manager はその変更をレプリカに伝達します。

共通セキュリティグループポリシーを削除する場合、ポリシーによって作成されたリソースをクリーンアップするかどうかを選択できます。Firewall Manager の共通セキュリティグループの場合、これらのリソースはレプリカセキュリティグループです。ポリシーの削除後に個々のレプリカを手動で管

理する場合を除き、クリーンアップオプションを選択してください。ほとんどの場合、クリーンアップオプションを選択するのが最も簡単な方法です。

レプリカの管理方法

Amazon VPC インスタンス内のレプリカセキュリティグループは、他の Amazon VPC セキュリティグループと同様に管理されます。詳細については、「Amazon VPC ユーザーガイド」の「VPC のセキュリティグループ」を参照してください。

コンテンツ監査セキュリティグループポリシー

AWS Firewall Manager コンテンツ監査セキュリティグループポリシーを使用して、組織のセキュリティグループで使用されているルールを監査し、ポリシーアクションを適用します。コンテンツ監査セキュリティグループポリシーは、ポリシーで定義した範囲に従って、 AWS 組織内で使用されているすべてのお客様が作成したセキュリティグループに適用されます。

コンソールを使用してコンテンツ監査セキュリティグループポリシーを作成する方法については、「コンテンツ監査セキュリティグループポリシーの作成」を参照してください。

ポリシーの範囲リソースタイプ

次のリソースタイプにコンテンツ監査セキュリティグループポリシーを適用できます。

- Amazon Elastic Compute Cloud (Amazon EC2) インスタンス
- Elastic Network Interface
- Amazon VPC セキュリティグループ

セキュリティグループは、明示的に範囲内にある場合、または範囲内のリソースに関連付けられている場合、ポリシーの範囲で考慮されます。

ポリシールールのオプション

コンテンツ監査ポリシーごとに、マネージドポリシールールまたはカスタムポリシールールのいずれかを使用できますが、両方は使用できません。

- マネージドポリシールール - マネージドルールを含むポリシーでは、アプリケーションとプロトコルのリストを使用して、Firewall Manager が監査して準拠または非準拠としてマークされるルールを制御できます。Firewall Manager によって管理されているリストを使用できます。独自のアプリケーションリストとプロトコルリストを作成および使用することもできます。これらのリストタイ

プおよびカスタムリストの管理オプションの詳細については、「マネージドリスト」を参照してください。

- カスタムポリシールール - カスタムポリシールールを含むポリシーでは、既存のセキュリティグループをポリシーの監査セキュリティグループとして指定します。監査セキュリティグループルールは、Firewall Manager が監査して準拠または非準拠としてマークされるルールを定義するテンプレートとして使用できます。

監査セキュリティグループ

ポリシーで監査セキュリティグループを使用するには、Firewall Manager 管理者アカウントを使用して監査セキュリティグループを作成する必要があります。セキュリティグループは、Amazon Virtual Private Cloud (Amazon VPC) または Amazon Elastic Compute Cloud (Amazon EC2) を通じて管理できます。詳細については、「Amazon VPC ユーザーガイド」の「セキュリティグループの操作」を参照してください。

コンテンツ監査セキュリティグループポリシーに使用するセキュリティグループは、Firewall Manager によって、ポリシーの範囲内にあるセキュリティグループの比較参照としてのみ使用されます。Firewall Manager は、組織内のどのリソースにも関連付けません。

監査セキュリティグループでルールを定義する方法は、ポリシールール設定での選択によって異なります。

- マネージドポリシールール – マネージドポリシールール設定では、監査セキュリティグループを使用して、ポリシー内の他の設定を上書きし、許可または拒否しなければ別のコンプライアンスに関する結果を発生させ得るルールを明示的に許可または拒否します。
  - 監査セキュリティグループで定義されているルールを常に許可することを選択した場合、監査セキュリティグループで定義されているルールと一致するルールは、他のポリシー設定にかかわらず、ポリシーに準拠しているものとみなされます。
  - 監査セキュリティグループで定義されているルールを常に拒否することを選択した場合、監査セキュリティグループで定義されているルールと一致するルールは、他のポリシー設定にかかわらず、ポリシーに非準拠であるものとみなされます。
- カスタムポリシールール: カスタムポリシールール設定について、監査セキュリティグループは、範囲内のセキュリティグループルールで許容できるものと許容できないものの例を提示します。
  - ルールの使用を許可する場合は、すべての範囲内セキュリティグループに、ポリシーの監査セキュリティグループルールの許可範囲内にあるルールのみを含める必要があります。この場合、ポリシーのセキュリティグループルールは、許容できる操作の例を示します。

- ルールの使用を拒否する場合は、すべての範囲内セキュリティグループに、ポリシーの監査セキュリティグループルールの許可範囲内にないルールのみを含める必要があります。この場合、ポリシーのセキュリティグループは、許容できない処理の例を示します。

ポリシーの作成と管理

監査セキュリティグループポリシーを作成するときは、自動修復を無効にする必要があります。自動修復を有効にする前に、ポリシー作成の影響を確認することをお勧めします。予想される影響を確認したら、ポリシーを編集して自動修復を有効にできます。自動修復が有効な場合、Firewall Manager は範囲内セキュリティグループで非準拠のルールを更新または削除します。

監査セキュリティグループポリシーの影響を受けるセキュリティグループ

組織内のすべてのユーザー定義セキュリティグループは、監査セキュリティグループポリシーの範囲に含める資格があります。

レプリカセキュリティグループはユーザー定義ではないため、監査セキュリティグループポリシーの範囲に直接適用することはできません。ただし、ポリシーの自動修復アクティビティの結果として更新できます。共通セキュリティグループポリシーのプライマリセキュリティグループはユーザー定義であり、監査セキュリティグループポリシーの範囲内に含めることができます。監査セキュリティグループポリシーがプライマリセキュリティグループに変更を加えた場合、Firewall Manager はそれらの変更をレプリカに自動的に伝達します。

使用状況監査セキュリティグループポリシー

AWS Firewall Manager 使用状況監査セキュリティグループポリシーを使用して、組織で未使用および冗長なセキュリティグループがないかをモニタリングし、オプションでクリーンアップを実行します。このポリシーの自動修復を有効にすると、Firewall Manager は次の処理を行います。

1. 冗長なセキュリティグループを統合します (このオプションを選択した場合)。
2. 未使用のセキュリティグループを削除します (このオプションを選択した場合)。

次のリソースタイプに使用状況の監査セキュリティグループポリシーを適用できます。

- Amazon VPC セキュリティグループ

コンソールを使用して使用状況の監査セキュリティグループポリシーを作成する方法については、「使用状況監査セキュリティグループポリシーの作成」を参照してください。

Firewall Manager が冗長セキュリティグループを検出して修正する方法

セキュリティグループを冗長とみなすには、まったく同じルールセットを持ち、同じ Amazon VPC インスタンスに存在している必要があります。

冗長なセキュリティグループのセットを修復するため、Firewall Manager は、保持するセット内のセキュリティグループの 1 つを選択し、セット内の他のセキュリティグループに関連付けられているすべてのリソースに関連付けます。その後、Firewall Manager は、関連付けられたリソースから他のセキュリティグループの関連付けを解除し、使用されていない状態にします。

> **Note**
>
> 未使用のセキュリティグループも削除するように選択した場合、Firewall Manager は次にその処理を実行します。これにより、冗長セットに含まれるセキュリティグループが削除される可能性があります。

Firewall Manager が未使用のセキュリティグループを検出して修正する方法

Firewall Manager は、次の両方に当てはまる場合、セキュリティグループが未使用であると見なします。

- セキュリティグループは、Amazon EC2 インスタンスまたは Amazon EC2 Elastic Network Interface では使用されません。
- Firewall Manager は、ポリシールール期間で指定された分数内に設定項目を受信していません。

ポリシールールの期間はデフォルト設定の 0 分ですが、時間を最大 365 日 (525,600 分) に増やして、新しいセキュリティグループをリソースに関連付ける時間を確保できます。

> **Important**
>
> デフォルト値の 0 以外の分数を指定する場合は、 で間接的な関係を有効にする必要があります AWS Config。そうしないと、使用状況監査セキュリティグループポリシーは意図したとおりに機能しません。の間接的な関係については AWS Config、「 AWS Config デベロッパーガイド」の「 の間接的な関係 AWS Config」を参照してください。

Firewall Manager は、可能であれば、ルール設定に従ってアカウントから削除することで、未使用のセキュリティグループを修正します。Firewall Manager がセキュリティグループを削除できない場

合、ポリシーに準拠していないとマークされます。Firewall Manager は、別のセキュリティグループによって参照されているセキュリティグループを削除することはできません。

修復のタイミングは、デフォルトの期間設定とカスタム設定のどちらを使用するかによって異なります。

- 期間を 0 に設定、デフォルト – この設定では、Amazon EC2 インスタンスまたは Elastic Network Interface でセキュリティグループが使用されなくなると、セキュリティグループは直ちに未使用と見なされます。

  このゼロ期間設定では、Firewall Manager はセキュリティグループを直ちに修正します。

- 0 より大きい期間 – この設定では、Amazon EC2 インスタンスまたは Elastic Network Interface によって使用されておらず、Firewall Manager が指定された分数内に設定項目を受信していない場合、セキュリティグループは未使用と見なされます。

  ゼロ以外の期間設定の場合、Firewall Manager は、セキュリティグループが 24 時間未使用の状態のままになった後にセキュリティグループを修正します。

デフォルトのアカウント指定

コンソールで使用状況に関する監査セキュリティグループポリシーを作成すると、Firewall Manager は自動的に [Exclude the specified accounts and include all others] (指定されたアカウントを除外し、他のすべてを含める) を選択します。その後、サービスによって、除外するリストに Firewall Manager 管理者アカウントが配置されます。これは推奨されるアプローチであり、Firewall Manager 管理者アカウントに属するセキュリティグループを手動で管理できます。

セキュリティグループポリシーのベストプラクティス

このセクションでは、 AWS Firewall Managerを使用してセキュリティグループを管理するための推奨事項を示します。

Firewall Manager 管理者アカウントを除外する

ポリシーの範囲を設定する場合は、Firewall Manager 管理者アカウントを除外します。コンソールを使用して使用状況監査セキュリティグループポリシーを作成する場合、これがデフォルトのオプションです。

自動修復を無効にした状態で開始する

コンテンツまたは使用状況監査セキュリティグループポリシーの場合は、自動修復を無効にした状態で始めます。ポリシーの詳細情報を確認し、自動修復による影響を判断します。変更が適切であることを確認したら、ポリシーを編集して自動修復を有効にします。

外部ソースを使用してセキュリティグループを管理する場合は、競合を回避する

Firewall Manager 以外のツールまたはサービスを使用してセキュリティグループを管理する場合は、Firewall Manager の設定と外部ソースの設定との競合を避けるように注意してください。自動修復を使用して、設定が競合する場合は、両側のリソースを消費する、競合する修復のサイクルを作成できます。

例えば、 AWS リソースのセットのセキュリティグループを維持するために別のサービスを設定し、同じリソースの一部またはすべてに対して異なるセキュリティグループを維持するように Firewall Manager ポリシーを設定するとします。他のセキュリティグループを範囲内のリソースに関連付けることを禁止するようにどちらかの側を設定すると、その側ではもう一方の側によって維持されているセキュリティグループの関連付けが削除されます。両側がこのように設定されている場合、競合する関連付けの解除と関連付けのサイクルになる可能性があります。

さらに、Firewall Manager 監査ポリシーを作成して、他のサービスのセキュリティグループ設定と競合するセキュリティグループ設定を強制するとします。Firewall Manager 監査ポリシーによって適用された修復によって、そのセキュリティグループを更新または削除し、他のサービスのコンプライアンスから解除できます。モニタリングして、検出した問題を自動的に修復するように他のサービスが設定されている場合、セキュリティグループを再作成または更新して、また Firewall Manager 監査ポリシーへのコンプライアンスを解除します。Firewall Manager 監査ポリシーに自動修復が設定されている場合、また外部セキュリティグループなどを更新または削除します。

このような競合を回避するには、Firewall Manager と外部ソースの間で相互に排他的な設定を作成します。

タグ付けを使用して、Firewall Manager ポリシーによる自動修復から外部のセキュリティグループを除外できます。これを行うには、外部ソースによって管理されるセキュリティグループまたはその他のリソースに 1 つ以上のタグを追加します。その後、Firewall Manager ポリシーの範囲を定義するときに、リソース仕様で、追加した 1 つまたは複数のタグを持つリソースを除外します。

同様に、外部のツールまたはサービスでは、Firewall Manager が管理するセキュリティグループを管理アクティビティまたは監査アクティビティから除外します。Firewall Manager リソースをインポートしないか、Firewall Manager 固有のタグ付けを使用して外部管理から除外します。

使用状況監査セキュリティグループポリシーのベストプラクティス

使用状況監査セキュリティグループポリシーを使用する場合は、次のガイドラインに従ってください。

- 15 分以内など、セキュリティグループの関連付けステータスを短時間で複数回変更することは避けてください。これにより、Firewall Manager は対応するイベントの一部またはすべてを見逃す可能性があります。例えば、セキュリティグループを Elastic Network Interface にすばやく関連付けたり、関連付けを解除したりしないでください。

セキュリティグループポリシーの注意点と制限事項

このセクションでは、Firewall Manager セキュリティグループポリシーを使用する際の注意事項と制限事項を示します。

- Fargate サービスタイプを使用して作成された Amazon EC2 Elastic Network Interface のセキュリティグループの更新はサポートされていません。ただし、 Amazon EC2 サービスタイプで Amazon ECS Elastic Network Interface のセキュリティグループを更新することはできます。
- Firewall Manager は、Amazon Relational Database Service によって作成された Amazon EC2 Elastic Network Interface のセキュリティグループをサポートしていません。
- Amazon ECS Elastic Network Interface の更新は、ローリング更新 (Amazon ECS) デプロイコントローラーを使用する Amazon ECS サービスでのみ可能です。CODE_DEPLOY や外部コントローラーなどの他の Amazon ECS デプロイコントローラーでは、Firewall Manager は現在 Elastic Network Interface を更新できません。
- Amazon EC2 Elastic Network Interface のセキュリティグループでは、セキュリティグループに対する変更は Firewall Manager にすぐには表示されません。Firewall Manager は通常、数時間以内に変更を検出しますが、検出は最長で 6 時間遅延する可能性があります。
- Firewall Manager は、Network Load Balancer の Elastic Network Interface のセキュリティグループの更新をサポートしていません。
- 一般的なセキュリティグループポリシーでは、共有 VPC が後でアカウントとの共有を解除された場合、Firewall Manager はアカウント内のレプリカセキュリティグループを削除しません。
- 使用状況監査セキュリティグループポリシーでは、カスタム遅延時間設定ですべてのスコープが同じである複数のポリシーを作成すると、コンプライアンス検出結果を含む最初のポリシーが検出結果をレポートするポリシーになります。

セキュリティグループポリシーのユースケース

AWS Firewall Manager 一般的なセキュリティグループポリシーを使用して、Amazon VPC インスタンス間の通信のホストファイアウォール設定を自動化できます。このセクションでは、標準 Amazon VPC アーキテクチャを一覧表示し、Firewall Manager 共通セキュリティグループポリシーを使用して各アーキテクチャを保護する方法について説明します。これらのセキュリティグループポリシーを使用すると、統合されたルールセットを適用してさまざまなアカウントのリソースを選択し、Amazon Elastic Compute Cloud および Amazon VPC のアカウント単位の設定を回避できます。

Firewall Manager 共通セキュリティグループポリシーでは、別の Amazon VPC のインスタンスとの通信に必要な EC2 Elastic Network Interface のみにタグ付けできます。同じ Amazon VPC 内の他のインスタンスの方がセキュリティが高く、分離されています。

ユースケース: Application Load Balancer および Classic Load Balancer に対するリクエストのモニタリングと制御

Firewall Manager 共通セキュリティグループポリシーを使用して、範囲内のロードバランサーが処理すべきリクエストを定義できます。これは、Firewall Manager コンソールで設定できます。セキュリティグループのインバウンドルールに準拠したリクエストのみがロードバランサーに到達でき、そのロードバランサーはアウトバウンドルールを満たすリクエストのみを配信します。

ユースケース: インターネットにアクセス可能、パブリック Amazon VPC

Amazon VPC 共通セキュリティグループポリシーを使用して、インバウンドポート 443 のみを許可するなど、パブリック Amazon VPC を保護できます。これは、パブリック VPC のインバウンド HTTPS トラフィックのみを許可することと同じです。VPC 内のパブリックリソースにタグ付けし (「PublicVPC」など)、そのタグを持つリソースのみに Firewall Manager ポリシーの範囲を設定できます。Firewall Manager は、これらのリソースにポリシーを自動的に適用します。

ユースケース: パブリックおよびプライベート Amazon VPC インスタンス

パブリックリソースには、前のインターネットにアクセス可能なパブリック Amazon VPC インスタンスのユースケースで推奨されているのと同じ共通セキュリティグループポリシーを使用できます。2 つ目の共通セキュリティグループポリシーを使用して、パブリックリソースとプライベートリソース間の通信を制限できます。パブリックおよびプライベート Amazon VPC インスタンスのリソースに「」のようなタグPublicPrivateを付けて、2 番目のポリシーを適用します。3 つ目のポリシーを使用して、プライベートリソースと他の企業またはプライベート Amazon VPC インスタンスとの間で許可される通信を定義できます。このポリシーの場合、プライベートリソースで別の識別タグを使用できます。

ユースケース: ハブアンドスポーク Amazon VPC インスタンス

共通セキュリティグループポリシーを使用して、ハブ Amazon VPC インスタンスとスポーク Amazon VPC インスタンス間の通信を定義できます。2 つ目のポリシーを使用して、各スポーク Amazon VPC インスタンスからハブ Amazon VPC インスタンスへの通信を定義できます。

ユースケース: Amazon EC2 インスタンスのデフォルトネットワークインターフェイス

共通セキュリティグループポリシーを使用して、内部 SSH やパッチ/OS 更新サービスなどの標準通信のみを許可し、その他の安全でない通信を禁止することができます。

ユースケース: オープン許可を持つリソースを特定する

監査セキュリティグループポリシーを使用して、パブリック IP アドレスと通信する許可を持つ組織内のすべてのリソース、またはサードパーティベンダーに属する IP アドレスを持つリソースを特定できます。

Amazon VPC ネットワークアクセスコントロールリスト (ACL) ポリシー

このセクションでは、 AWS Firewall Manager ネットワーク ACL ポリシーの仕組みについて説明し、その使用に関するガイダンスを提供します。コンソールを使用してネットワーク ACL ポリシーを作成するガイダンスについては、「」を参照してくださいネットワーク ACL ポリシーの作成。

Amazon VPC ネットワークアクセスコントロールリスト (ACLs「Amazon VPC ユーザーガイド」の「ネットワーク ACLs」を参照してください。

Firewall Manager のネットワーク ACL ポリシーを使用して、 の組織の Amazon Virtual Private Cloud (Amazon VPC) ネットワークアクセスコントロールリスト (ACLsを管理できます AWS Organizations。ポリシーのネットワーク ACL ルール設定と、設定を適用するアカウントとサブネットを定義します。Firewall Manager は、組織全体で追加または更新されるアカウントとサブネットにポリシー設定を継続的に適用します。ポリシーの範囲と の詳細については AWS Organizations、AWS Firewall Manager ポリシーの範囲「」および「 ユーザーガイドAWS Organizations」を参照してください。

Firewall Manager ネットワーク ACL ポリシーを定義する場合、名前やスコープなどの標準の Firewall Manager ポリシー設定に加えて、以下を指定します。

- インバウンドトラフィックとアウトバウンドトラフィックの処理に関する最初と最後のルール。Firewall Manager は、ポリシーの範囲内にあるネットワーク ACLs にこれらの存在と順序を強

制するか、コンプライアンス違反を報告します。個々のアカウントは、ポリシーの最初と最後のルールの間に実行するカスタムルールを作成できます。

- 修復時に修復を強制するかどうかによって、ネットワーク ACL のルール間でトラフィック管理の競合が発生します。これは、ポリシーに対して修復が有効になっている場合にのみ適用されます。

Firewall Manager のネットワーク ACL ルールとタグ付け

このセクションでは、ネットワーク ACL ポリシールールの仕様と、Firewall Manager によって管理されるネットワーク ACLs について説明します。

マネージドネットワーク ACL でのタグ付け

Firewall Manager は、マネージドネットワーク ACL に の値を持つ`FMManaged`タグを付けます`true`。Firewall Manager は、このタグ設定を持つネットワーク ACLs に対してのみ修復を実行します。

ポリシーで定義するルール

ネットワーク ACL ポリシー仕様では、インバウンドトラフィックに対して最初と最後に実行するルールと、アウトバウンドトラフィックに対して最初と最後に実行するルールを定義します。

デフォルトでは、ポリシー内の最初と最後のルールを任意に組み合わせて使用するために、最大 5 つのインバウンドルールを定義できます。同様に、最大 5 つのアウトバウンドルールを定義できます。これらの制限の詳細については、「」を参照してくださいソフトクォータ。ネットワーク ACLs「Amazon VPC ユーザーガイド」の「ネットワーク ACLs」を参照してください。

ポリシールールにルール番号を割り当てません。代わりに、ルールを評価する順序で指定します。Firewall Manager はその順序を使用して、管理するネットワーク ACLs にルール番号を割り当てます。

これ以外にも、Amazon VPC を介してネットワーク ACL のルールを管理する場合と同様に、ポリシーのネットワーク ACL ルールの仕様を管理します。Amazon VPC でのネットワーク ACL 管理の詳細については、「Amazon VPC ユーザーガイド」の「ネットワーク ACLs」および「ネットワーク ACLs」を参照してください。

マネージドネットワーク ACL のルール

Firewall Manager は、個々のアカウントマネージャーが定義するカスタムルールの前後にポリシーの最初と最後のルールを配置することで、管理するネットワーク ACL にルールを設定します。Firewall

Manager は、カスタムルールの順序を保持します。ネットワーク ACLsは、最も低い番号のルールから評価されます。

Firewall Manager が最初にネットワーク ACL を作成するときに、次の番号でルールを定義します。

- 最初のルール: 1、2、... – Firewall Manager ネットワーク ACL ポリシーでユーザーが定義します。

  Firewall Manager は、1 から始まるルール番号を 1 の増分で割り当て、ポリシー仕様で順序付けしたルールを順序付けします。
- カスタムルール: 5,000、5,100... – Amazon VPC を通じて個々のアカウントマネージャーによって管理されます。

  Firewall Manager は、5,000 から始まり、後続のルールごとに 100 ずつ増加する番号をこれらのルールに割り当てます。
- 最後のルール: ... 32,765、32,766 – Firewall Manager ネットワーク ACL ポリシーでユーザーが定義します。

  Firewall Manager は、ポリシー仕様で順序付けされたルールとともに、1 刻みで可能な最大数 32766 で終わるルール番号を割り当てます。

ネットワーク ACL の初期化後、Firewall Manager は個々のアカウントがマネージドネットワーク ACLsで行う変更を制御しません。個々のアカウントは、コンプライアンスを破ることなくネットワーク ACL を変更でき、カスタムルールがポリシーの最初と最後のルールの間で番号が付けられたままであり、最初と最後のルールが指定された順序を維持している場合に限ります。ベストプラクティスとして、カスタムルールを管理するときは、このセクションで説明されている番号付けに従ってください。

Firewall Manager がサブネットのネットワーク ACL 管理を開始する方法

Firewall Manager は、サブネットを Firewall Manager が作成し、 を FMManagedに設定してタグ付けしたネットワーク ACL に関連付けると、サブネットのネットワーク ACL の管理を開始しますtrue。

ネットワーク ACL ポリシーに準拠するには、サブネットのネットワーク ACL が、ポリシーの最初のルールをポリシーで指定された順序で最初に配置し、最後のルールを最後に配置し、順序を付けて配置し、その他のカスタムルールを中間に配置する必要があります。これらの要件は、サブネットが既に関連付けられているアンマネージドネットワーク ACL またはマネージドネットワーク ACL によって満たされます。

Firewall Manager がアンマネージドネットワーク ACL に関連付けられているサブネットにネットワーク ACL ポリシーを適用すると、Firewall Manager は実行可能なオプションが識別されると停止し、次の順序でチェックします。

1. 関連付けられたネットワーク ACL がすでに準拠している — 現在サブネットに関連付けられているネットワーク ACL が準拠している場合、Firewall Manager はその関連付けをそのままにして、サブネットのネットワーク ACL 管理を開始しません。

   Firewall Manager は、所有していないネットワーク ACL を変更または管理しませんが、準拠している限り、Firewall Manager はそれをそのままにして、ポリシーコンプライアンスをモニタリングします。
2. 準拠のマネージドネットワーク ACL が利用可能 – Firewall Manager が、必要な設定に準拠するネットワーク ACL を既に管理している場合、これはオプションです。修復が有効になっている場合、Firewall Manager はサブネットを関連付けます。修復が無効になっている場合、Firewall Manager はサブネットを非準拠としてマークし、修復オプションとしてネットワーク ACL の関連付けを置き換えます。
3. 新しい準拠マネージドネットワーク ACL の作成 – 修復が有効になっている場合、Firewall Manager は新しいネットワーク ACL を作成し、サブネットに関連付けます。それ以外の場合、Firewall Manager はサブネットに非準拠のマークを付け、新しいネットワーク ACL を作成し、ネットワーク ACL の関連付けを置き換える修復オプションを提供します。

これらのステップが失敗した場合、Firewall Manager はサブネットのコンプライアンス違反を報告します。

Firewall Manager は、サブネットが最初に対象範囲に入ったとき、およびサブネットのアンマネージドネットワーク ACL がコンプライアンス違反になったときに、これらのステップに従います。

Firewall Manager が非準拠のマネージドネットワーク ACLs

このセクションでは、Firewall Manager がマネージドネットワーク ACLs がポリシーに準拠していない場合にそれらを修正する方法について説明します。Firewall Manager は、 `FMManaged` タグを に設定して、マネージドネットワーク ACLs のみを修復します`true`。Firewall Manager によって管理されていないネットワーク ACLs「」を参照してください初期ネットワーク ACL 管理。

修復は、最初、カスタム、最後のルールの相対的な場所を復元し、最初と最後のルールの順序を復元します。修復中、Firewall Manager は必ずしもネットワーク ACL の初期化で使用するルール番号にルールを移動するとは限りません。これらのルールカテゴリの初期番号設定と説明については、「」を参照してください初期ネットワーク ACL 管理。

準拠ルールとルールの順序を確立するには、Firewall Manager がネットワーク ACL 内でルールを移動する必要がある場合があります。Firewall Manager は、既存の準拠ルールの順序を可能な限り維持することで、ネットワーク ACL の保護を維持します。例えば、ルールを新しい場所に一時的に複製し、元のルールを順番に削除して、プロセス中に相対的な場所を保持する場合があります。

このアプローチでは設定を保護しますが、中間ルールにはネットワーク ACL にスペースも必要です。Firewall Manager がネットワーク ACL のルールの制限に達すると、修復が停止します。この場合、ネットワーク ACL はコンプライアンス違反のままになり、Firewall Manager はその理由を報告します。

アカウントが Firewall Manager によって管理されるネットワーク ACL にカスタムルールを追加し、それらのルールが Firewall Manager の修復を妨げる場合、Firewall Manager はネットワーク ACL 上の修復アクティビティを停止し、競合を報告します。

強制修復

ポリシーの自動修復を選択した場合は、最初のルールと最後のルールのどちらを強制的に修復するかも指定します。

Firewall Manager は、カスタムルールとポリシールール間のトラフィック処理に競合が発生した場合、対応する強制修復設定を参照します。強制修復が有効になっている場合、Firewall Manager は競合にかかわらず修復を適用します。このオプションが有効になっていない場合、Firewall Manager は修復を停止します。いずれの場合も、Firewall Manager はルールの競合を報告し、修復オプションを提供します。

ルール数の要件と制限

修復中、Firewall Manager は、ルールを一時的に複製して、提供する保護を変更せずにルールを移動することがあります。

インバウンドルールまたはアウトバウンドルールの場合、Firewall Manager が修復を実行するために必要とするルールの最大数は次のとおりです。

```
2 * (the number of rules defined in the policy for the traffic direction)
+
the number of custom rules defined in the network ACL for the traffic direction
```

ネットワーク ACLs とネットワーク ACL ポリシーは、変更可能なルール制限によって制限されます。Firewall Manager が修復作業で制限に達すると、修復の試行を停止し、コンプライアンス違反を報告します。

Firewall Manager が修復アクティビティを実行するためのスペースを確保するために、制限の引き上げをリクエストできます。または、ポリシーまたはネットワーク ACL の設定を変更して、使用するルールの数を減らすこともできます。

ネットワーク ACL の制限の詳細については、「Amazon VPC ユーザーガイド」のACLsの Amazon VPC クォータ」を参照してください。

修復が失敗した場合

ネットワーク ACL の更新中に、何らかの理由で Firewall Manager を停止する必要がある場合、Firewall Manager は変更をロールバックせず、代わりにネットワーク ACL を中間状態のままにします。FMManaged タグが に設定されているネットワーク ACL に重複するルールが表示された場合true、Firewall Manager はおそらく修正中です。変更が一定期間部分的に完了する可能性がありますが、Firewall Manager が修復を行うアプローチにより、トラフィックが中断されたり、関連するサブネットの保護が低下したりすることはありません。

Firewall Manager は、コンプライアンス違反ACLs を完全に修復しない場合、関連するサブネットのコンプライアンス違反を報告し、可能な修復オプションを提案します。

修復が失敗した後に再試行する

ほとんどの場合、Firewall Manager がネットワーク ACL に対する修復変更を完了できなかった場合、最終的には変更を再試行します。

ただし、修復がネットワーク ACL ルール数の制限または VPC ネットワーク ACL 数の制限に達した場合は例外です。Firewall Manager は、リソースを制限設定 AWS を超える修復アクティビティを実行できません。このような場合、続行するにはカウントを減らすか、制限を増やす必要があります。制限の詳細については、「Amazon VPC ユーザーガイド」のACLs の Amazon VPC クォータ」を参照してください。

Firewall Manager ネットワーク ACL コンプライアンスレポート

Firewall Manager は、範囲内のサブネットにアタッチされているすべてのネットワーク ACLs のコンプライアンスをモニタリングして報告します。

一般的に、ルールの順序が正しくない場合や、ポリシールールとカスタムルール間のトラフィック処理動作の競合などの状況では、コンプライアンス違反が発生します。コンプライアンス違反レポートには、コンプライアンス違反と修復オプションが含まれます。

Firewall Manager は、ネットワーク ACL ポリシーのコンプライアンス違反を、他のポリシータイプと同じ方法で報告します。コンプライアンスレポートの詳細については、「」を参照してくださいAWS Firewall Manager ポリシーのコンプライアンス情報の表示。

ポリシー更新中のコンプライアンス違反

ネットワーク ACL ポリシーを変更した後、Firewall Manager がポリシーの範囲内にあるネットワーク ACLs を更新するまで、Firewall Manager はそれらのネットワーク ACLs非準拠としてマークします。Firewall Manager は、ネットワーク ACLs が厳密に言うとコンプライアンスに準拠している場合でも、これを行います。

例えば、ポリシー仕様からルールを削除しても、範囲内のネットワーク ACLs にはまだ追加のルールがあるにもかかわらず、そのルール定義はポリシーに準拠している可能性があります。ただし、追加のルールは Firewall Manager が管理しているルールの一部であるため、Firewall Manager はそれらを現在のポリシー設定の違反と見なします。これは、Firewall Manager が Firewall Manager マネージドネットワーク ACLs。

Firewall Manager ネットワーク ACL ポリシーを使用するためのベストプラクティス

このセクションでは、Firewall Manager のネットワーク ACL ポリシーとマネージドネットワーク ACLs。

Firewall Manager によって管理されるネットワーク ACLs を特定するには、 **FMManaged** タグを参照してください。

Firewall Manager が管理するネットワーク ACLs のFMManagedタグは に設定されていますtrue。このタグを使用すると、独自のカスタムネットワーク ACLs を Firewall Manager で管理しているものから区別できます。

ネットワーク ACL の**FMManaged**タグの値を変更しない

Firewall Manager は、このタグを使用して、ネットワーク ACL で管理ステータスを設定および決定します。

Firewall Manager マネージドネットワーク ACLs

サブネットと Firewall Manager によって管理されるネットワーク ACLs との関連付けを手動で変更しないでください。そうすることで、Firewall Manager がこれらのサブネットの保護を管理する機能を無効にできます。Firewall Manager によって管理されるネットワーク ACLs は、 のFMManagedタグ設定を探すことで識別できますtrue。

Firewall Manager ポリシー管理からサブネットを削除するには、Firewall Manager ポリシースコープ設定を使用してサブネットを除外します。例えば、サブネットにタグを付け、そのタグをポリシースコープから除外できます。詳細については、「AWS Firewall Manager ポリシーの範囲」を参照してください。

マネージドネットワーク ACL を更新するときは、Firewall Manager によって管理されるルールを変更しないでください。

Firewall Manager によって管理されるネットワーク ACL では、「」で説明されている番号付けスキームに従って、カスタムルールをポリシールールから分離したままにしますFirewall Manager のネットワーク ACL ルールとタグ付け。5,000 ~ 32,000 の数値を持つルールのみを追加または変更します。

アカウントの制限に対してルールを追加しすぎないようにする

ネットワーク ACL の修復中、Firewall Manager は通常、ネットワーク ACL ルールの数を一時的に増やします。コンプライアンス違反の問題を回避するには、使用しているルールに十分なスペースがあることを確認してください。詳細については、「Firewall Manager が非準拠のマネージドネットワーク ACLs」を参照してください。

自動修復を無効にした状態で開始する

自動修復を無効にしてから、ポリシーの詳細情報を確認して、自動修復が与える影響を判断します。変更が適切であることを確認したら、ポリシーを編集して自動修復を有効にします。

Firewall Manager のネットワーク ACL ポリシーに関する注意事項

このセクションでは、Firewall Manager ネットワーク ACL ポリシーを使用する際の注意事項と制限事項を示します。

- 他のポリシーよりも更新時間が遅い – Firewall Manager は、通常、Amazon EC2 ネットワーク ACL APIs がリクエストを処理できる速度が制限されているため、他の Firewall Manager ポリシーよりもネットワーク ACL ポリシーとポリシーの変更をより遅く適用します。ポリシーの変更は、他の Firewall Manager ポリシーと同様の変更よりも時間がかかる場合があります。特に、ポリシーを初めて追加する場合です。
- 初期サブネット保護のために、Firewall Manager は古いポリシーを優先します。これは、Firewall Manager ネットワーク ACL ポリシーによってまだ保護されていないサブネットにのみ適用されます。サブネットが同時に複数のネットワーク ACL ポリシーの範囲に入る場合、Firewall Manager は最も古いポリシーを使用してサブネットを保護します。

- ポリシーがサブネットの保護を停止する理由 – サブネットのネットワーク ACL を管理するポリシーは、次のいずれかが発生するまで管理を保持します。
  - サブネットはポリシーの範囲外になります。
  - ポリシーは削除されます。
  - サブネットの関連付けは、別の Firewall Manager ポリシーによって管理され、サブネットが範囲内にあるネットワーク ACL に手動で変更します。

Firewall Manager ネットワーク ACL ポリシーの削除

Firewall Manager ネットワーク ACL ポリシーを削除すると、Firewall Manager は、ポリシー`false`で管理しているすべてのネットワーク ACLs の`FMManaged`タグ値を に変更します。

さらに、ポリシーによって作成されたリソースをクリーンアップするかどうかを選択できます。クリーンアップを選択すると、Firewall Manager は次のステップを順番に試行します。

1. 関連付けを元の に戻す – Firewall Manager は、Firewall Manager が管理を開始する前に、関連付けられているネットワーク ACL にサブネットを関連付けようとします。
2. ネットワーク ACL から最初と最後のルールを削除する – 関連付けを変更できない場合、Firewall Manager はポリシーの最初と最後のルールを削除しようとし、サブネットに関連付けられているネットワーク ACL にカスタムルールのみを残します。
3. ルールや関連付けに何もしない — 上記のいずれも実行できない場合、Firewall Manager はネットワーク ACL とその関連付けをそのまま残します。

クリーンアップオプションを選択しない場合は、ポリシーが削除された後、各ネットワーク ACL を手動で管理する必要があります。ほとんどの場合、クリーンアップオプションを選択するのが最も簡単な方法です。

AWS Network Firewall ポリシー

AWS Firewall Manager Network Firewall ポリシーを使用して、 で組織全体の Amazon Virtual Private Cloud VPCs AWS Network Firewall ファイアウォールを管理できます AWS Organizations。一元管理されたファイアウォールを組織全体に適用することも、アカウントと VPC の選択したサブセットに適用することもできます。

Network Firewall は、VPC 内のパブリックサブネットのために、ネットワークトラフィックフィルタリング保護を提供します。Firewall Manager は、ポリシーで定義されているファイアウォール管

理タイプに基づいてファイアウォールを作成および管理します。Firewall Manager は以下のファイアウォール管理モデルを提供します。

- 分散型 - ポリシーの範囲内にある各アカウントおよび VPC について、Firewall Manager は Network Firewall ファイアウォールを作成し、ファイアウォールエンドポイントを VPC サブネットにデプロイして、ネットワークトラフィックをフィルタリングします。
- 集約型 - Firewall Manager は、単一の Amazon VPC に単一の Network Firewall のファイアウォールを作成します。
- 既存のファイアウォールのインポート - Firewall Manager は、既存のファイアウォールを、単一の Firewall Manager ポリシーにインポートして管理します。ポリシーで管理されているインポートされたファイアウォールに追加のルールを適用して、ファイアウォールがセキュリティ基準を満たすようにすることが可能です。

> **Note**
>
> Firewall Manager の Network Firewall ポリシーは、組織全体における VPC のための Network Firewall 保護を管理するために使用する Firewall Manager ポリシーです。
> Network Firewall 保護は、ファイアウォールポリシーと呼ばれる Network Firewall サービスのリソースで指定されます。

Network Firewall の使用については、「AWS Network Firewall デベロッパーガイド」を参照してください。

次のセクションでは、Firewall Manager の Network Firewall ポリシーを使用するための要件と、ポリシーの仕組みについて説明します。ポリシーの作成手順については、「の AWS Firewall Manager ポリシーの作成 AWS Network Firewall」を参照してください。

リソース共有を有効にする必要があります。

Network Firewall ポリシーは、組織内のアカウント全体で Network Firewall ルールグループを共有します。これを機能させるには、 AWS Organizations用にリソース共有を有効にする必要があります。リソース共有を有効にする方法については、「Network Firewall ポリシーと DNS Firewall ポリシーのリソース共有」を参照してください。

Network Firewall ルールグループを定義する必要があります。

新しい Network Firewall ポリシーを指定するときは、 AWS Network Firewall を直接使用する場合と同じようにファイアウォールポリシーを定義します。追加するステートレスルールグループ、デ

フォルトのステートレスアクション、およびステートフルルールグループを指定します。ルールグループをポリシーに含めるには、そのルールグループが Firewall Manager 管理者アカウントに既に存在している必要があります。Network Firewall ルールグループの作成については、「AWS Network Firewall ルールグループ」を参照してください。

Firewall Manager がファイアウォールエンドポイントを作成する方法

ポリシーの Firewall 管理タイプによって、Firewall が Firewall Manager を作成する方法が決まります。ポリシーでは、[Distributed] (分散型) ファイアウォールや [Centralized] (集約型) ファイアウォールを作成することも、[import existing firewalls] (既存のファイアウォールのインポート) を選択することもできます。

- 分散型 - 分散デプロイモデルでは、Firewall Manager は、ポリシー範囲内にある各 VPC のためにエンドポイントを作成します。ファイアウォールエンドポイントを作成するアベイラビリティーゾーンを指定してエンドポイントの場所をカスタマイズすることも、Firewall Manager に、パブリックサブネットを使用してアベイラビリティーゾーンにエンドポイントを自動的に作成させることもできます。アベイラビリティーゾーンを手動で選択する場合は、アベイラビリティーゾーンごとに許可される CIDR のセットを制限するオプションがあります。Firewall Manager でエンドポイントを自動的に作成する場合は、サービスが VPC 内で単一のエンドポイントを作成するか、複数のファイアウォールエンドポイントを作成するかを指定する必要もあります。
  - 複数のファイアウォールエンドポイントの場合、Firewall Manager は、各アベイラビリティーゾーンにファイアウォールエンドポイントをデプロイします。ここには、インターネットゲートウェイ、または Firewall Manager が作成したファイアウォールエンドポイントルートがルートテーブル内に存在するサブネットがあります。これは、Network Firewall ポリシーのデフォルトのオプションです。
  - 単一のファイアウォールエンドポイントの場合、Firewall Manager は、インターネットゲートウェイルートを持つサブネット内の単一のアベイラビリティーゾーンにファイアウォールエンドポイントをデプロイします。このオプションでは、他のゾーンのトラフィックは、ファイアウォールによるフィルタリングの対象となるためにゾーン境界を越える必要があります。

    > **Note**
    >
    > これらのオプションの両方で、IPv4/prefixlist ルートを含むルートテーブルに関連付けられたサブネットが必要です。Firewall Manager は、他のリソースをチェックしません。

- 集約型 - 集約型デプロイモデルを使用すると、Firewall Manager は、検査 VPC 内に 1 つ以上のファイアウォールエンドポイントを作成します。検査 VPC は、Firewall Manager がエンドポイントを起動する中央 VPC です。集約型デプロイモデルを使用する場合は、ファイアウォールエンド

ポイントを作成するアベイラビリティーゾーンも指定します。ポリシーを作成した後で検査 VPC を変更することはできません。別の検査 VPC を使用するには、新しいポリシーを作成する必要があります。

- 既存のファイアウォールのインポート - 既存のファイアウォールをインポートする場合、ポリシーに 1 つ以上のリソースセットを追加して、ポリシーで管理するファイアウォールを選択します。リソースセットは、組織内のアカウントによって管理されるリソース (この場合は Network Firewall 内の既存のファイアウォール) の集合体です。ポリシーでリソースセットを使用する前に、まずリソースセットを作成する必要があります。Firewall Manager のリソースセットについては、「Firewall Manager でのリソースセットの操作」を参照してください。

  インポートされたファイアウォールを使用する場合は、以下の点に留意してください。

  - インポートしたファイアウォールが非対応になった場合、Firewall Manager は次の場合を除いて、自動的に違反の解決を試みます。
    - Firewall Manager と Network Firewall ポリシーのステートフルまたはステートレスのデフォルトアクションが一致しない場合。
    - インポートしたファイアウォールのファイアウォールポリシー内のルールグループが、Firewall Manager ポリシーのルールグループと同じ優先度を持つ場合。
    - インポートしたファイアウォールが、ポリシーのリソースセットに含まれていないファイアウォールに関連付けられたファイアウォールポリシーを使用している場合。これは、ファイアウォールに設定できるファイアウォールポリシーは 1 つだけですが、1 つのファイアウォールポリシーを複数のファイアウォールに関連付けることができるため発生する可能性があります。
    - インポートされたファイアウォールのファイアウォールポリシーに属する既存のルールグループのうち、Firewall Manager ポリシーでも指定されているものに、異なる優先順位が割り当てられる場合。
  - ポリシーでリソースクリーンアップを有効にすると、Firewall Manager は FMS インポートポリシーに含まれているルールグループを、リソースセットの範囲内のファイアウォールから削除します。
  - Firewall Manager の既存のファイアウォールのインポート管理タイプで管理されているファイアウォールは、一度に 1 つのポリシーによってのみ管理できます。同じリソースセットが複数のインポートネットワークファイアウォールポリシーに追加された場合、リソースセット内のファイアウォールは、リソースセットが追加された最初のポリシーによって管理され、2 番目のポリシーでは無視されます。

- 現在、Firewall Manager では、例外ポリシー設定をストリーミングしません。例外ポリシーのストリーミングの詳細については、「AWS Network Firewall デベロッパーガイド」の「例外ポリシーのストリーム」を参照してください。

分散型または集約型のファイアウォール管理を使用しているポリシーのアベイラビリティーゾーンのリストを変更すると、Firewall Manager は、過去に作成されたが、現在はポリシーの範囲に含まれていないエンドポイントのクリーンアップを試みます。Firewall Manager は、範囲外のエンドポイントを参照するルートテーブルのルートがない場合にのみ、エンドポイントを削除します。Firewall Manager は、これらのエンドポイントを削除できないことが判明した場合、ファイアウォールサブネットを非準拠としてマークし、安全に削除できるようになるまでエンドポイントの削除を試行し続けます。

Firewall Manager がファイアウォールサブネットを管理する方法

ファイアウォールサブネットは、ネットワークトラフィックをフィルタリングするファイアウォールエンドポイントのために Firewall Manager が作成する VPC サブネットです。各ファイアウォールエンドポイントは、専用 VPC サブネットでデプロイされる必要があります。Firewall Manager は、ポリシーの範囲内にある各 VPC に少なくとも 1 つのファイアウォールサブネットを作成します。

自動エンドポイント設定で分散型デプロイモデルを使用するポリシーの場合、Firewall Manager は、インターネットゲートウェイルートを持つサブネット、または Firewall Manager がポリシー用に作成したファイアウォールエンドポイントへのルートを持つサブネットを持つアベイラビリティーゾーンにのみ、ファイアウォールサブネットを作成します。詳細については、「Amazon VPC ユーザーガイド」の「VPC とサブネット」を参照してください。

Firewall Manager がファイアウォールエンドポイントを作成するアベイラビリティーゾーンを指定する分散型または集約型モデルのいずれかを使用するポリシーの場合、Firewall Manager は、アベイラビリティーゾーンに他のリソースがあるかどうかにかかわらず、それらの特定のアベイラビリティーゾーンにエンドポイントを作成します。

Network Firewall ポリシーを最初に定義する際に、Firewall Manager が範囲内の各 VPC 内のファイアウォールサブネットを管理する方法を指定します。この選択を後で変更することはできません。

自動エンドポイント設定で分散型デプロイモデルを使用するポリシーの場合、次のオプションから選択できます。

- パブリックサブネットを持つすべてのアベイラビリティーゾーンにファイアウォールサブネットをデプロイします。これがデフォルトの動作です。これにより、トラフィックフィルタリング保護の高可用性を実現できます。

- 1 つのアベイラビリティーゾーンに 1 つのファイアウォールサブネットをデプロイします。この選択により、Firewall Manager は、最もパブリックなサブネットを持つ VPC 内のゾーンを特定し、そこにファイアウォールサブネットを作成します。単一のファイアウォールエンドポイントは、VPC のすべてのネットワークトラフィックをフィルタリングします。これにより、ファイアウォールのコストは削減できますが、可用性は高くなく、他のゾーンからのトラフィックがフィルタリング対象となるためにはゾーン境界を越える必要があります。

カスタムエンドポイント設定または集約型デプロイモデルで分散デプロイモデルを使用するポリシーの場合、Firewall Manager は、ポリシーの範囲内にある指定されたアベイラビリティーゾーンにサブネットを作成します。

Firewall Manager がファイアウォールサブネットに使用する VPC CIDR ブロックを提供することも、ファイアウォールエンドポイントアドレスの選択の決定を Firewall Manager に任せることもできます。

- CIDR ブロックを指定しない場合、Firewall Manager は、使用可能な IP アドレスを VPC にクエリします。
- CIDR ブロックのリストを指定すると、Firewall Manager は、指定した CIDR ブロック内の新しいサブネットのみを検索します。/28 CIDR ブロックを使用する必要があります。Firewall Manager が作成する各ファイアウォールサブネットについて、CIDR ブロックリストをたどり、アベイラビリティーゾーンと VPC に適用可能で、かつ、利用可能なアドレスを持つ最初に見つかったリストを使用します。Firewall Manager が VPC 内のオープンスペースを見つけることができない場合 (制限の有無にかかわらず)、サービスは VPC 内にファイアウォールを作成しません。

Firewall Manager がアベイラビリティーゾーンで必要なファイアウォールサブネットを作成できない場合、そのサブネットをポリシーに非準拠であるものとしてマークします。ゾーンがこの状態にある間、別のゾーンのエンドポイントによるフィルタリングの対象となるためには、ゾーンのトラフィックがゾーン境界を越える必要があります。これは、単一のファイアウォールサブネットのシナリオに似ています。

Firewall Manager が Network Firewall リソースを管理する方法

Firewall Manager でポリシーを定義するときは、標準の AWS Network Firewall ファイアウォールポリシーのネットワークトラフィックフィルタリング動作を指定します。ステートレスおよびステートフル Network Firewall ルールグループを追加し、ステートレスルールと一致しないパケットのデフォルトアクションを指定します。でのファイアウォールポリシーの操作については AWS Network Firewall、AWS Network Firewall 「 ファイアウォールポリシー」を参照してください。

分散型および集約型ポリシーの場合、Network Firewall ポリシーを保存すると、Firewall Manager は、ポリシーの範囲内にある各 VPC にファイアウォールとファイアウォールポリシーを作成します。Firewall Manager は、次の値を連結して、これらの Network Firewall リソースの名前を指定します。

- `FMManagedNetworkFirewall` または `FMManagedNetworkFirewallPolicy` のいずれかの固定文字列 (リソースタイプによる)。
- Firewall Manager ポリシー名。これは、ポリシーの作成時に割り当てる名前です。
- Firewall Manager ポリシー ID。これは Firewall Manager ポリシーの AWS リソース ID です。
- Amazon VPC ID。これは、Firewall Manager がファイアウォールとファイアウォールポリシーを作成する VPC の AWS リソース ID です。

Firewall Manager によって管理されるファイアウォールの名前の例を次に示します。

```
FMManagedNetworkFirewallEXAMPLENameEXAMPLEFirewallManagerPolicyIdEXAMPLEVPCId
```

ファイアウォールポリシー名の例を次に示します。

```
FMManagedNetworkFirewallPolicyEXAMPLENameEXAMPLEFirewallManagerPolicyIdEXAMPLEVPCId
```

ポリシーを作成した後、VPC のメンバーアカウントは、ファイアウォールポリシー設定またはルールグループを上書きすることはできませんが、Firewall Manager が作成したファイアウォールポリシーにルールグループを追加することはできます。

Firewall Manager がポリシーの VPC ルートテーブルを管理およびモニタリングする方法

> **Note**
>
> ルートテーブル管理は、集約型デプロイモデルを使用するポリシーでは現在サポートされていません。

Firewall Manager がファイアウォールエンドポイントを作成すると、それらの VPC ルートテーブルも作成されます。ただし、Firewall Manager は VPC ルートテーブルを管理しません。ネットワークトラフィックを Firewall Manager によって作成されたファイアウォールエンドポイントに送信するように VPC ルートテーブルを設定する必要があります。Amazon VPC イングレスルーティングの拡張機能を使用して、新しいファイアウォールエンドポイントを通じてトラフィックをルーティングす

るようにルーティングテーブルを変更します。変更によって、保護するサブネットと外部の場所の間にファイアウォールエンドポイントを挿入する必要があります。実行が必要となる正確なルーティングは、アーキテクチャとそのコンポーネントによって異なります。

現在、Firewall Manager では、ファイアウォールをバイパスするインターネットゲートウェイ宛てのトラフィックについて、VPC ルートテーブルのルートをモニタリングできます。Firewall Manager は、NAT ゲートウェイなどの他のターゲットゲートウェイをサポートしていません。

VPC のルートテーブルの管理については、「Amazon Virtual Private Cloud ユーザーガイド」の「Managing route tables for your VPC」(VPC のルートテーブルの管理) を参照してください。Network Firewall のためのルートテーブル管理に関する情報については、「AWS Network Firewall デベロッパーガイド」の「AWS Network Firewallのためのルートテーブル設定」を参照してください。

ポリシーのモニタリングを有効にすると、Firewall Manager は VPC ルート設定を継続的にモニタリングし、その VPC のファイアウォール検査をバイパスするトラフィックについて警告します。サブネットにファイアウォールエンドポイントルートがある場合、Firewall Manager は次のルートを検索します。

- Network Firewall エンドポイントにトラフィックを送信するルート。
- Network Firewall エンドポイントからインターネットゲートウェイにトラフィックを転送するルート。
- インターネットゲートウェイから Network Firewall エンドポイントへのインバウンドルート。
- ファイアウォールサブネットからのルート。

サブネットに Network Firewall ルートがあるが、Network Firewall とインターネットゲートウェイのルートテーブルに非対称ルーティングがある場合、Firewall Manager はサブネットを非準拠としてレポートします。また、Firewall Manager は、Firewall Manager が作成したファイアウォールルートテーブルおよびサブネットのルートテーブルでインターネットゲートウェイへのルートを検出し、それらを非準拠として報告します。Network Firewall サブネットルートテーブルおよびインターネットゲートウェイルートテーブル内の追加ルートも、非準拠として報告されます。Firewall Manager は、違反タイプに応じて、ルート設定を準拠状態にするための修復アクションを提案します。Firewall Manager は、すべてのケースで提案を提供するわけではありません。例えば、お客様のサブネットに、Firewall Manager の外部で作成されたファイアウォールエンドポイントがある場合、Firewall Manager は修復アクションを提案しません。

デフォルトでは、Firewall Manager は、アベイラビリティーゾーンの境界を越えるトラフィックを検査のために非準拠としてマークします。ただし、VPC 内に単一のエンドポイントを自動的に作成す

るように選択した場合、Firewall Manager は、アベイラビリティーゾーンの境界を越えるトラフィックを非準拠としてマークしません。

カスタムエンドポイント設定で分散デプロイモデルを使用するポリシーの場合、あるアベイラビリティーゾーンからのものであって、アベイラビリティーゾーンの境界を越え、ファイアウォールエンドポイントがないトラフィックを、準拠または非準拠のいずれとしてマークするかを選択できます。

> **Note**
>
> - Firewall Manager は、IPv6 ルートやプレフィックスリストルートなど、IPv4 以外のルートに対する修復アクションを提案しません。
> - `DisassociateRouteTable` API コールを使用して行われた呼び出しは、検出に最大で 12 時間かかる場合があります。
> - Firewall Manager は、ファイアウォールエンドポイントを含むサブネットの Network Firewall ルートテーブルを作成します。Firewall Manager は、このルートテーブルに有効なインターネットゲートウェイと VPC のデフォルトルートだけが含まれていると仮定します。このルートテーブル内の追加ルートまたは無効なルートは、非準拠とみなされます。

Firewall Manager ポリシーを設定するときに、[Monitor] (モニタリング) モードを選択すると、Firewall Manager はリソースの違反とリソースに関する修復の詳細を提供します。これらの推奨される修復アクションを使用して、ルートテーブル内のルートの問題を修正できます。[Off] (オフ) モードを選択した場合、Firewall Manager はルートテーブルのコンテンツをモニタリングしません。このオプションでは、VPC ルートテーブルを自ら管理できます。これらのリソース違反の詳細については、「AWS Firewall Manager ポリシーのコンプライアンス情報の表示」を参照してください。

> **Warning**
>
> ポリシーの作成時に AWS Network Firewall ルート設定で Monitor を選択した場合、そのポリシーで Monitor をオフにすることはできません。ただし、[Off] (オフ) を選択すると、後で有効にすることができます。

AWS Network Firewall ポリシーのログ記録の設定

Network Firewall ポリシーの集中ログ記録を有効にして、組織内のトラフィックに関する詳細情報を取得できます。フローログ記録を選択してネットワークトラフィックフローをキャプチャするか、ア

ラートログ記録を選択して DROP または ALERT に設定されたルールアクションを持つルールに一致するトラフィックをレポートできます。 AWS Network Firewall ログ記録の詳細については、「AWS Network Firewall デベロッパーガイド」の「AWS Network Firewallからのネットワークトラフィックのログ記録」を参照してください。

ポリシーの Network Firewall ファイアウォールから Amazon S3 バケットにログを送信します。ログ記録を有効にすると、 は、ファイアウォール設定を更新して、予約された AWS Firewall Manager プレフィックス を使用して選択した Amazon S3 バケットにログを配信することで、設定された各 Network Firewall のログを AWS Network Firewall 配信します`<policy-name>-<policy-id>`。

> **Note**
>
> このプレフィックスは、Firewall Manager によってログ記録設定が追加されたのか、またはアカウント所有者によって追加されたのかを判断するために、Firewall Manager によって使用されます。アカウント所有者が独自のカスタムログ記録に予約済みプレフィックスを使用しようとすると、Firewall Manager ポリシーのログ記録設定によって上書きされます。

Amazon S3 バケットを作成し、保存されているログを確認する方法の詳細については、「Amazon Simple Storage Service ユーザーガイド」の「Amazon S3 とは」を参照してください。

ログ記録を有効にするには、次の要件を満たす必要があります。

- Firewall Manager ポリシーで指定する Amazon S3 が存在している必要があります。
- アクセス許可を持っている必要があります。
  - `logs:CreateLogDelivery`
  - `s3:GetBucketPolicy`
  - `s3:PutBucketPolicy`
- ログ記録の送信先である Amazon S3 バケットが に保存されているキーによるサーバー側の暗号化を使用している場合は AWS Key Management Service、Firewall Manager が CloudWatch Logs ロググループにログを記録できるように、 AWS KMS カスタマーマネージドキーに次のポリシーを追加する必要があります。

```
{
    "Effect": "Allow",
    "Principal": {
        "Service": "delivery.logs.amazonaws.com"
```

```
    },
    "Action": [
        "kms:Encrypt*",
        "kms:Decrypt*",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:Describe*"
    ],
    "Resource": "*"
}
```

AWS Network Firewall 一元ログ記録に使用できるのは、Firewall Manager 管理者アカウントのバケットのみであることに注意してください。

Network Firewall ポリシーで集中ログ記録を有効にすると、Firewall Manager はアカウントに対して次のアクションを実行します。

- Firewall Manager は、選択した S3 バケットの許可を更新して、ログ配信を許可します。
- Firewall Manager は、ポリシーの範囲内にある各メンバーアカウントのために、S3 バケットにディレクトリを作成します。各アカウントのログは `<bucket-name>/<policy-name>-<policy-id>/AWSLogs/<account-id>` にあります。

Network Firewall ポリシーのログ記録を有効にするには

1. Firewall Manager 管理者アカウントを使用して Amazon S3 バケットを作成します。詳細については、「Amazon Simple Storage Service ユーザーガイド」の「バケットの作成」を参照してください。
2. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

3. ナビゲーションペインで、[Security Policies] (セキュリティポリシー) を選択します。

4. ログ記録を有効にする Network Firewall ポリシーを選択します。 AWS Network Firewall ログ記録の詳細については、「 デベロッパーガイド」の「 からのネットワークトラフィックのログ記録 AWS Network Firewall」を参照してください。 AWS Network Firewall
5. [Policy details] (ポリシーの詳細) タブの [Policy rules] (ポリシールール) セクションで、[Edit] (編集) を選択します。
6. ログを有効にして集約するには、[Logging configuration] (ログ記録の設定) で 1 つ以上のオプションを選択します。
   - フローログを有効化および集約する
   - アラートログを有効化および集約する
7. ログの配信先とする Amazon S3 バケットを選択します。有効にするログタイプごとにバケットを選択する必要があります。両方のログタイプに同じバケットを使用できます。
8. (オプション) カスタムメンバーアカウントで作成されたログ記録をポリシーのログ記録設定に置き換える場合は、[Override existing logging configuration] (既存のログ設定を上書き) を選択します。
9. [Next] (次へ) を選択します。
10. 設定を確認し、[Save] (保存) を選択してポリシーに対する変更を保存します。

Network Firewall ポリシーのログ記録を無効にするには

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security Policies] (セキュリティポリシー) を選択します。
3. ログ記録を無効にする Network Firewall ポリシーを選択します。
4. [Policy details] (ポリシーの詳細) タブの [Policy rules] (ポリシールール) セクションで、[Edit] (編集) を選択します。

5. [Logging configuration status] (ログ記録設定のステータス) で、[Enable and aggregate flow logs] (フローログを有効にして集約) および [Enable and aggregate alert logs] (アラートログを有効にして集約) の選択を解除します (選択されている場合)。
6. [Next] (次へ) を選択します。
7. 設定を確認し、[Save] (保存) を選択してポリシーに対する変更を保存します。

Amazon Route 53 Resolver DNS Firewall ポリシー

AWS Firewall Manager DNS Firewall ポリシーを使用して、 の組織全体で Amazon Route 53 Resolver DNS Firewall ルールグループと Amazon Virtual Private Cloud VPCs 間の関連付けを管理できます AWS Organizations。一元管理されたルールグループを組織全体に適用することも、アカウントと VPC の選択したサブセットに適用することもできます。

DNS Firewall は、VPC のアウトバウンド DNS トラフィックのフィルタリングと規制を提供します。DNS Firewall のルールグループで再利用可能なフィルタリングルールのコレクションを作成し、そのルールグループを VPC に関連付けます。Firewall Manager ポリシーを適用すると、ポリシーの範囲内にある各アカウントおよび VPC について、Firewall Manager は、Firewall Manager ポリシーで指定した関連付けの優先順位の設定を使用して、ポリシー内の各 DNS Firewall ルールグループと、ポリシーの範囲内にある各 VPC の間の関連付けを作成します。

DNS Firewall の使用方法については、「Amazon Route 53 デベロッパーガイド」の「Amazon Route 53 Resolver DNS Firewall」を参照してください。

次のセクションでは、Firewall Manager の DNS Firewall ポリシーを使用するための要件と、ポリシーの仕組みについて説明します。ポリシーの作成手順については、「Amazon Route 53 Resolver DNS Firewall の AWS Firewall Manager ポリシーの作成」を参照してください。

リソース共有を有効にする必要があります。

DNS Firewall ポリシーは、組織内のアカウント全体で DNS Firewall ルールグループを共有します。これを機能させるには、 でリソース共有が有効になっている必要があります AWS Organizations。リソース共有を有効にする方法については、「Network Firewall ポリシーと DNS Firewall ポリシーのリソース共有」を参照してください。

DNS Firewall ルールグループを定義する必要があります。

新しい DNS Firewall ポリシーを指定する際に、Amazon Route 53 Resolver DNS Firewall を直接使用する場合と同じようにルールグループを定義します。ルールグループをポリシーに含めるには、そ

のルールグループが Firewall Manager 管理者アカウントに既に存在している必要があります。DNS Firewall ルールグループの作成については、「DNS Firewall のルールグループとルール」を参照してください。

優先順位が最低のルールグループと最高のルールグループの関連付けを定義します

Firewall Manager の DNS Firewall ポリシーを通じて管理する DNS Firewall ルールグループの関連付けには、VPC について、優先度が最低の関連付けと優先度が最高の関連付けが含まれます。ポリシー設定では、これらは最初と最後のルールグループとして表示されます。

DNS Firewall は、VPC の DNS トラフィックを次の順序でフィルタリングします。

1. Firewall Manager の DNS Firewall ポリシーで定義されている最初のルールグループ。有効な値は 1 ~ 99 です。
2. DNS Firewall を通じて個々のアカウントマネージャーによって関連付けられた DNS Firewall ルールグループ。
3. Firewall Manager の DNS Firewall ポリシーで定義されている最後のルールグループ。有効な値は 9,901 ~ 10,000 です。

ルールグループの削除

Firewall Manager DNS Firewall ポリシーからルールグループを削除するには、次の手順を実行する必要があります。

1. ルールグループを Firewall Manager DNS Firewall ポリシーから削除します。
2. でルールグループの共有を解除します AWS Resource Access Manager。自己所有のルールグループを共有解除するには、リソース共有からルールグループを削除する必要があります。これは、 AWS RAM コンソールまたは AWS CLI を使用して実行できます。リソースの共有解除ついては、「AWS RAM ユーザーガイド」の「AWS RAM内のリソース共有を更新する」を参照してください。
3. DNS Firewall コンソールまたは AWS CLI を使用してルールグループを削除します。

Firewall Manager が作成するルールグループの関連付けの名前を付ける方法

DNS Firewall ポリシーを保存するときに自動修復を有効にした場合、Firewall Manager は、ポリシーで指定したルールグループとポリシーの範囲内にある VPC の間に DNS Firewall の関連付けを作成します。Firewall Manager は、次の値を連結することにより、これらの関連付けに名前を付けます。

- 固定文字列、FMManaged_。
- Firewall Manager ポリシー ID。これは Firewall Manager ポリシーの AWS リソース ID です。

Firewall Manager によって管理されるファイアウォールの名前の例を次に示します。

```
FMManaged_EXAMPLEDNSFirewallPolicyId
```

ポリシーを作成した後、VPC のアカウント所有者がファイアウォールポリシー設定またはルールグループの関連付けを上書きすると、Firewall Manager は、ポリシーを非準拠としてマークし、是正措置の提案を試みます。アカウント所有者は、DNS Firewall ポリシーの範囲内の VPC に他の DNS Firewall ルールグループを関連付けることができます。個々のアカウント所有者によって作成された関連付けには、最初と最後のルールグループの関連付けの間で優先順位の設定が必要です。

Palo Alto Networks Cloud NGFW ポリシー

パロアルトネットワークスのクラウド次世代ファイアウォール (NGFW) は、ポリシーに使用できるサードパーティのファイアウォールサービスです。 AWS Firewall Manager Firewall Manager 用のパロアルトネットワークスクラウドNGFWを使用すると、パロアルトネットワークスクラウドのNGFWリソースとルールスタックを作成し、すべてのアカウントに一元的に展開できます。 AWS

Firewall Manager でパロアルトネットワークスのクラウド NGFW を使用するには、まずMarketplaceでパロアルトネットワークスクラウド NGFW 従量課金制サービスに登録する必要があります。AWS サブスクライブ後、Palo Alto Networks Cloud NGFW サービスで一連のステップを実行して、アカウントと Cloud NGFW 設定を構成します。次に、Firewall Manager ークラウド FMS ポリシーを作成して、Organizations 内のすべてのアカウントにパロアルトネットワークスのクラウドNGFWリソースとルールを一元的に展開および管理します。 AWS

Firewall Manager ポリシーを作成する手順については、「Palo Alto Networks Cloud NGFW の AWS Firewall Manager ポリシーの作成」を参照してください。Firewall Manager 用に Palo Alto Networks Cloud NGFW を設定および管理する方法については、Palo Alto Networks Palo Alto Networks Cloud NGFW on AWS のドキュメントを参照してください。

Fortigate Cloud Native Firewall (CNF) as a Service ポリシー

Fortigate Cloud Native Firewall（ CNF ）as a Serviceは、ポリシーに使用できるサードパーティのファイアウォールサービスです。 AWS Firewall Manager Fortigate CNF は、クラウドネットワークの保護とセキュリティポリシーの管理を容易にする次世代のファイアウォールサービスで

す。Fortigate CNF for Firewall Manager を使用すると、Fortigate CNFのリソースとポリシーセットを作成し、すべてのアカウントに一元的に導入できます。 AWS

Firewall Manager でFortigate CNFを使用するには、まずMarketplace でFortigate Cloudネイティブファイアウォール（CNF）をサービスとして登録する必要があります。 AWS サブスクライブ後、Fortigate CNF サービスで一連の手順を実行し、グローバルポリシーセットやその他の設定を構成します。次に、Firewall Manager ポリシーを作成して、Organizations 内のすべてのアカウントに Fortigate CNFリソースを一元的に展開して管理します。 AWS

Fortigate CNF Firewall Manager ポリシーを作成する手順については、「Fortigate Cloud Native Firewall (CNF) as a Service の AWS Firewall Manager ポリシーの作成」を参照してください。Firewall Manager で使用するために Fortigate CNF を設定および管理する方法については、Fortigate CNF のドキュメントを参照してください。

Network Firewall ポリシーと DNS Firewall ポリシーのリソース共有

Firewall Manager のNetwork Firewall と DNS ファイアウォールのポリシーを管理するには、 AWS Organizations in とのリソース共有を有効にする必要があります AWS Resource Access Manager。これにより、Firewall Manager は、これらのポリシータイプを作成するときに、アカウント全体で保護をデプロイできます。

リソース共有を有効にするには、「AWS Resource Access Manager ユーザーガイド」の「AWS Organizations内でリソース共有を有効にする」の手順に従ってください。

リソース共有に関する問題

リソース共有を有効にしている場合や、リソース共有を必要とするFirewall Manager ポリシーで作業しているときに、リソース共有で問題が発生する可能性があります。 AWS RAM

これらの問題の例には次が含まれます。

- 指示に従って共有を有効にすると、 AWS RAM コンソールの [共有の有効化] AWS Organizations オプションがグレー表示になり、選択できなくなります。
- リソース共有を必要とするポリシーで Firewall Manager を使用している場合、ポリシーは非準拠としてマークされ、リソース共有または AWS RAM が有効になっていないことを示すメッセージが表示されます。

リソースの共有で問題が発生した場合は、次の手順を使用してリソース共有の有効化を試みます。

再試行して、リソース共有を有効にする

- 再試行し、次のいずれかのオプションを使用して共有を有効にします。
  - (オプション) AWS RAM コンソールで、AWS Resource Access Manager ユーザーガイドの「共有を有効にする」 AWS Organizationsの指示に従います。
  - (オプション) AWS RAM API を使用して、を呼び出しますEnableSharingWithAwsOrganization。のドキュメントを参照してくださいEnableSharingWithAwsOrganization。

Firewall Manager でのリソースセットの操作

AWS Firewall Manager リソースセットは、Firewall Manager ポリシーでグループ化して管理できるファイアウォールなどのリソースのコレクションです。リソースセットを使用すると、組織内のメンバーは、ポリシーで管理するリソースをきめ細かく制御できます。リソースセットを使用するには、コンソールまたは PutResourceSet API を使用してリソースセットを作成し、そのリソースセットを Firewall Manager ポリシーに追加します。

次のリソースタイプとセキュリティポリシータイプのリソースセットを作成および管理できます。

リソースタイプ	Firewall Manager のセキュリティポリシータイプ
AWS Network Firewall - ファイアウォール	Network Firewall ポリシー - リソースセットを使用して、Network Firewall から既存のファイアウォールをインポートします。Network Firewall ポリシーでリソースセットを使用する方法については、の AWS Firewall Manager ポリシーの作成 AWS Network Firewall 手順の「既存のファイアウォールのインポート」を参照してください。

次のセクションでは、リソースセットの作成と削除の要件について説明します。

トピック

- Firewall Manager でリソースセットを操作するときの考慮事項
- リソースセットの作成
- リソースセットの削除

Firewall Manager でリソースセットを操作するときの考慮事項

リソースセットを使用するときは、次の考慮事項に注意してください。

存在しないリソースへの参照

リソースセットにリソースを追加する際は、Amazon リソースネーム (ARN) を使用してリソースへの参照を作成します。Firewall Manager は Amazon リソースネーム (ARN) が正しい形式であるかど

うかを検証するものの、参照先リソースが存在するかどうかは確認しません。リソースが存在しないのに ARN の検証を通過した場合、Firewall Manager はリソースセットにリソース参照を含めます。同じ ARN を持つ新しいリソースが後で作成された場合、Firewall Manager はリソースセットの関連ポリシーのルールグループを新しいリソースに適用します。

削除されたリソース

リソースセット内のリソースが削除されても、リソースへの参照は、Firewall Manager 管理者によって削除されるまでリソースセット内に残ります。

AWS Organizations 組織を離れるメンバーアカウントが所有するリソース

メンバーアカウントが組織を離れると、そのメンバーアカウントが所有するリソースへの参照はすべてリソースセットに残りますが、リソースセットが関連付けられているポリシーでは管理されなくなります。

複数のポリシーへの関連付け

リソースセットは複数のポリシーに関連付けることができますが、すべてのポリシータイプが同じリソースを管理する複数のポリシーをサポートしているわけではありません。サポートされていないシナリオについては、特定のポリシータイプのドキュメントを参照してください。

リソースセットの作成

リソースセットを作成するには (コンソール)

1. Firewall Manager AWS Management Console 管理者アカウントを使用してにサインインし、で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

   > **Note**
   >
   > Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Resources sets] (リソースセット) を選択します。
3. [Create resource set] (リソースセットの作成) を選択します。
4. [Resource set name] (リソースセット名) には、わかりやすい名前を入力します。

5. (オプション) リソースセットの [Description] (説明) を入力します。
6. [次へ] を選択します。
7. [リソースを選択] で [AWS アカウント ID] を選択し、[リソースを選択] を選択して、このアカウントが所有および管理するリソースをリソースセットに追加します。リソースを選択したら、[Add] (追加) を選択してリソースをリソースセットに追加します。
8. [次へ] を選択します。
9. [Resource set tags] (リソースセットタグ) の場合は、リソースセットに必要な識別タグを追加します。タグの詳細については、「タグエディタの使用」を参照してください。
10. [Next] (次へ) を選択します。
11. 新しいリソースセットを確認します。変更するには、変更する部分で [Edit] (編集) を選択します。これにより、作成ウィザードの対応するステップに戻ります。設定が適切であることを確認したら、[Create resource set] (リソースセットの作成) を選択します。

リソースセットの削除

リソースセットを削除する前に、そのリソースセットを使用するすべてのポリシーからリソースセットの関連付けを解除する必要があります。コンソールまたは PutPolicy API を使用して、ポリシーの詳細ページでリソースグループの関連付けを解除できます。

リソースセットを削除するには (コンソール)

1. ナビゲーションペインで、[Resources sets] (リソースセット) を選択します。
2. 削除するリソースセットの横にあるオプションを選択します。
3. [削除] をクリックします。

AWS Firewall Manager ポリシーのコンプライアンス情報の表示

このセクションでは、 AWS Firewall Manager ポリシーの範囲内にあるアカウントとリソースのコンプライアンスステータスを表示するガイダンスを提供します。クラウドのセキュリティとコンプライアンス AWS を維持するために で実施されているコントロールについては、「」を参照してくださいFirewall Manager のコンプライアンス検証。

> **Note**
>
> Firewall Manager がポリシーコンプライアンスをモニタリングするには、保護されたリソースの設定変更を継続的に記録 AWS Config する必要があります。 AWS Config 設定では、記録頻度をデフォルト設定である連続 に設定する必要があります。

> **Note**
>
> 保護されたリソースで適切なコンプライアンス状態を維持するには、Firewall Manager 保護の状態を自動または手動で繰り返し変更しないでください。Firewall Manager は、 からの情報 AWS Config を使用して、リソース設定の変更を検出します。変更が十分に迅速に適用されると、 AWS Config は変更の一部を追跡できなくなり、Firewall Manager のコンプライアンスまたは修復状態に関する情報が失われる可能性があります。
> Firewall Manager で保護しているリソースのコンプライアンスまたは修復ステータスが正しくない場合は、まず Firewall Manager の保護を変更またはリセットするプロセスを実行していないことを確認し、 で関連する設定ルールを再評価してリソース AWS Config の追跡を更新します AWS Config。

すべての AWS Firewall Manager ポリシーについて、ポリシーの範囲内にあるアカウントとリソースのコンプライアンスステータスを表示できます。ポリシーの設定がアカウントまたはリソースの設定で反映されている場合、そのアカウントまたはリソースは、Firewall Manager ポリシーに準拠しています。各ポリシータイプには独自のコンプライアンス要件があります。これは、ポリシーを定義するときにチューニングできます。一部のポリシーでは、セキュリティリスクをより良く理解および管理するのに役立てるために、範囲内のリソースの詳細な違反情報を表示することもできます。

ポリシーのコンプライアンス情報を表示するには

1. Firewall Manager 管理者アカウント AWS Management Console を使用して にサインインし、 で Firewall Manager コンソールを開きますhttps://console.aws.amazon.com/wafv2/fmsv2。Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

> **Note**
>
> Firewall Manager 管理者アカウントの設定については、「AWS Firewall Manager 前提条件」を参照してください。

2. ナビゲーションペインで、[Security policies] (セキュリティポリシー) を選択します。
3. ポリシーを選択します。ポリシーページの [Accounts and resources] (アカウントとリソース) タブで、Firewall Manager は、ポリシーの範囲内にあるアカウントと範囲外にあるアカウントでグループ化された、組織内のアカウントを一覧表示します。

   [Accounts within policy scope] (ポリシーの範囲内のアカウント) ペインには、各アカウントのコンプライアンスステータスが一覧表示されます。[Compliant] (準拠) ステータスは、ポリシーがアカウントのすべての範囲内のリソースに正常に適用されたことを示します。[Noncompliant] (非準拠) ステータスは、ポリシーがアカウントの 1 つ以上の範囲内のリソースに適用されていないことを示します。
4. 非準拠のアカウントを選択します。アカウントページで、Firewall Manager は、準拠していない各リソースの ID と種類、およびリソースがポリシーに違反している理由の一覧を表示します。

> **Note**
>
> リソースタイプ `AWS::EC2::NetworkInterface` (ENI) および `AWS::EC2::Instance` の場合、Firewall Manager は限られた数の非準拠リソースを表示する場合があります。その他の非準拠のリソースを一覧表示するには、対象のアカウントについて最初に表示されるリソースを修正します。

5. Firewall Manager ポリシータイプがコンテンツ監査セキュリティグループポリシーである場合、リソースの詳細な違反情報にアクセスできます。

   違反の詳細を表示するには、リソースを選択します。

> **Note**
>
> 詳細なリソース違反ページが追加される前に、Firewall Manager が非準拠であると判断したリソースには、違反の詳細がない場合があります。

リソースページに、Firewall Manager は、リソースタイプに応じて、違反に関する具体的な詳細を一覧表示します。

- **`AWS::EC2::NetworkInterface`** (ENI) – Firewall Manager は、リソースが準拠していないセキュリティグループに関する情報を表示します。セキュリティグループを選択すると、その詳細が表示されます。
- **`AWS::EC2::Instance`** - Firewall Manager は、非準拠の EC2 インスタンスにアタッチされている ENI を表示します。また、リソースが準拠していないセキュリティグループに関する情報を表示します。セキュリティグループを選択すると、その詳細が表示されます。
- **`AWS::EC2::SecurityGroup`** - Firewall Manager は、次の違反の詳細を表示します。
  - [Noncompliant security group rule] (非準拠のセキュリティグループルール) – 違反しているルール (プロトコル、ポート範囲、IP CIDR 範囲、説明など)。
  - [Referenced rule] (参照されるルール) – 非準拠のセキュリティグループのルールが違反する監査セキュリティグループのルールとその詳細。
  - [Violation reasons] (違反の理由) – 違反の検出結果の説明。
  - [Remediation action] (修復アクション) – 実行する推奨アクション。Firewall Manager が安全な修復アクションを決定できない場合、このフィールドは空白です。
- **`AWS::EC2::Subnet`** — これはネットワーク ACL および Network Firewall ポリシーに使用されます。

  Firewall Manager は、サブネット ID、VPC ID、アベイラビリティーゾーンを表示します。該当する場合、Firewall Manager には違反に関する追加情報が含まれます。違反の説明のコンポーネントには、リソースの予想される状態の説明、現在の非準拠状態、および使用可能な場合は、不一致の原因の説明が含まれます。

  Network Firewall 違反

  - [management violations] (ルート管理違反) – モニタリングモードを使用する Network Firewall ポリシーの場合、Firewall Manager は、基本的なサブネット情報に加えて、サブネット、インターネットゲートウェイ、および Network Firewall のサブネットルートテーブルの想定ルートと実際のルートを表示します。Firewall Manager は、実際のルートがルートテーブルの想定されるルートと一致しない場合、違反がある旨のアラートを発信します。
  - [Remediation actions for route management violations] (ルート管理違反の修正アクション) – モニタリングモードを使用する Network Firewall ポリシーの場合、Firewall Manager は、違反のあるルート設定に対して可能な修正アクションを提案します。

例えば、サブネットがファイアウォールエンドポイントを介してトラフィックを送信すると予想されるが、現在のサブネットがインターネットゲートウェイに直接トラフィックを送信しているとします。これは、ルート管理違反です。この場合の推奨される修復は、順序付けられたアクションのリストである場合があります。1 つ目の推奨事項は、必要なルートを Network Firewall サブネットのルートテーブルに追加して、発信トラフィックをインターネットゲートウェイに転送し、VPC 内の宛先の着信トラフィックを `local` に転送することです。2 つ目の推奨事項は、サブネットのルートテーブル内のインターネットゲートウェイルートまたは無効な Network Firewall ルートを置き換えて、発信トラフィックをファイアウォールエンドポイントに送信することです。3 つ目の推奨事項は、受信トラフィックをファイアウォールエンドポイントに転送するために、インターネットゲートウェイのルートテーブルに必要なルートを追加することです。

- **`AWS::EC2::InternetGateway`** - これは、モニターモードが有効になっている Network Firewall ポリシーに使用されます。
  - [Route management violations] (ルート管理違反) – インターネットゲートウェイがルートテーブルに関連付けられていない場合、またはインターネットゲートウェイのルートテーブルに無効なルートがある場合、インターネットゲートウェイは非準拠です。
  - [Remediation actions for route management violations] (ルート管理違反の修正アクション) – Firewall Manager は、ルート管理違反を修正するための可能な修正アクションを提案します。

  Example 1 — ルート管理違反と修復の提案

  インターネットゲートウェイはルートテーブルに関連付けられていません。推奨される修復アクションは、順序付けられたアクションのリストである場合があります。最初のアクションは、ルートテーブルを作成することです。2 つ目のアクションは、ルートテーブルをインターネットゲートウェイに関連付けることです。3 つ目のアクションは、必要なルートをインターネットゲートウェイルートテーブルに追加することです。

  Example 2 — ルート管理違反と修復の提案

  インターネットゲートウェイは有効なルートテーブルに関連付けられていますが、ルートが正しく設定されていません。推奨される修復は、順序付けられたアクションのリストである場合があります。最初の提案は、無効なルートを削除することです。2 つ目は、必要なルートをインターネットゲートウェイルートテーブルに追加することです。

- **`AWS::NetworkFirewall::FirewallPolicy`** - これは、Network Firewall ポリシーに使用されます。Firewall Manager は、非準拠になるように変更された Network Firewall のファイア

ウォールポリシーに関する情報を表示します。この情報は、予想されるファイアウォールポリシーと、カスタマーアカウントで見つかったポリシーを提供するものであるため、ステートレスルールグループ名とステートフルルールグループ名および優先度の設定、カスタムアクション名、ならびにデフォルトのステートレスアクションの設定を比較できます。違反の説明のコンポーネントには、リソースの予想される状態の説明、現在の非準拠状態、および使用可能な場合は、不一致の原因の説明が含まれます。

- **`AWS::EC2::VPC`** - これは DNS Firewall ポリシーに使用されます。Firewall Manager は、Firewall Manager の DNS Firewall ポリシーの範囲内にあり、ポリシーに準拠していない VPC に関する情報を表示します。提供される情報には、VPC に関連付けられることが予想されるルールグループおよび実際のルールグループが含まれます。違反の説明のコンポーネントには、リソースの予想される状態の説明、現在の非準拠状態、および使用可能な場合は、不一致の原因の説明が含まれます。

AWS Firewall Manager 検出結果

AWS Firewall Manager は、コンプライアンス違反のリソースと検出した攻撃の検出結果を作成し、に送信します AWS Security Hub。Security Hub の検出結果については、「AWS Security Hubでの検索」を参照してください。

Security Hub と Firewall Manager を使用する場合、Firewall Manager は、検出結果を自動的に Security Hub に送信します。Security Hub の使用を開始する方法については、AWS Security Hub ユーザーガイドの「AWS Security Hubの設定」を参照してください。

> **Note**
>
> Firewall Manager は、管理対象のポリシーとモニタリング対象のリソースの結果のみを更新します。
> Firewall Manager は、以下の検出結果を解決しません。
>
> - 削除されたポリシー。
> - 削除されたリソース。
> - タグの変更やポリシー定義の変更など、Firewall Manager ポリシーの範囲外になったリソース。

Firewall Manager の検出結果を表示する方法

Security Hub で Firewall Manager の検出結果を表示するには、「Working with Findings in Security Hub」(Security Hub で検出結果を使用する) にあるガイダンスに従い、次の設定を使用してフィルターを作成します。

- [Product Name] (製品名) に設定された属性。
- [EQUALS] に設定された演算子。
- `Firewall Manager` に設定された値。この設定では、大文字と小文字が区別されます。

これを無効にすることはできますか?

Security Hub コンソールを使用して、 AWS Firewall Manager 検出結果と Security Hub の統合を無効にすることができます。ナビゲーションバーで [Integrations] (統合) を選択し、Firewall Manager ペインで [Disable Integration] (統合の無効化) を選択します。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。

AWS Firewall Manager 検出結果タイプ

- AWS WAF ポリシー調査結果
- AWS Shield Advanced ポリシー調査結果
- セキュリティグループ共通ポリシーの検出結果
- セキュリティグループコンテンツ監査ポリシーの検出結果
- セキュリティグループ使用状況監査ポリシーの検出結果
- Amazon Route 53 Resolver DNS Firewall ポリシーの検出結果

AWS WAF ポリシー調査結果

Firewall Manager AWS WAF ポリシーを使用して、 AWS WAF 内のリソースにルールグループを適用できます AWS Organizations。詳細については、「AWS Firewall Manager ポリシーの使用」を参照してください。

Firewall Manager マネージドウェブ ACL がリソースにありません。

AWS リソースには、Firewall Manager AWS Firewall Manager ポリシーに従って管理対象ウェブ ACL が関連付けられていない。ポリシーの Firewall Manager 修復を有効にして、これを修正できます。

- 重要度 - 80
- ステータスの設定 - PASSED/FAILED

- 更新 - Firewall Manager が修復アクションを実行すると、検出結果が更新され、重要度が HIGH から INFORMATIONAL に引き下げられます。修復を実行した場合、Firewall Manager は検出結果を更新しません。

Firewall Manager 管理ウェブ ACL に正しく設定されていないルールグループがあります。

Firewall Manager によって管理されるウェブ ACL のルールグループが、Firewall Manager ポリシーに従って正しく設定されていません。これは、ウェブ ACL に、ポリシーに必要なルールグループがないことを意味します。ポリシーの Firewall Manager 修復を有効にして、これを修正できます。

- 重要度 - 80
- ステータスの設定 - PASSED/FAILED
- 更新 - Firewall Manager が修復アクションを実行すると、検出結果が更新され、重要度が HIGH から INFORMATIONAL に引き下げられます。修復を実行した場合、Firewall Manager は検出結果を更新しません。

AWS Shield Advanced ポリシー調査結果

AWS Shield Advanced ポリシーの詳細については、を参照してくださいセキュリティグループポリシー。

Shield Advanced 保護がリソースにありません。

Firewall Manager のポリシーによると、Shield Advanced AWS による保護が必要なリソースにはそれがありません。ポリシーの Firewall Manager 修復を有効にして、リソースの保護を有効にできます。

- 重要度 - 60
- ステータスの設定 - PASSED/FAILED
- 更新 - Firewall Manager が修復アクションを実行すると、検出結果が更新され、重要度が HIGH から INFORMATIONAL に引き下げられます。修復を実行した場合、Firewall Manager は検出結果を更新しません。

Shield Advanced は、モニタリング対象のリソースに対する攻撃を検出しました。

Shield アドバンスドは、 AWS 保護されたリソースへの攻撃を検出しました。ポリシーで Firewall Manager の修復を有効にできます。

- 重要度 - 70
- ステータス設定 - なし
- 更新 — Firewall Manager はこの検出結果を更新しません。

セキュリティグループ共通ポリシーの検出結果

セキュリティグループ共通ポリシーの詳細については、「セキュリティグループポリシー」を参照してください。

リソースのセキュリティグループが正しく設定されていません。

Firewall Manager は、Firewall Manager ポリシーに従って、必要な Firewall Manager マネージドセキュリティグループの関連付けがないリソースを特定しました。ポリシーで Firewall Manager 修復を有効にできます。これにより、ポリシー設定に従って関連付けが作成されます。

- 重要度 - 70
- ステータスの設定 - PASSED/FAILED
- 更新 — Firewall Manager はこの検出結果を更新します。

Firewall Manager レプリカセキュリティグループがプライマリセキュリティグループと同期されていません。

Firewall Manager レプリカセキュリティグループが、共通セキュリティグループポリシーに従って、プライマリセキュリティグループと同期されていません。ポリシーで Firewall Manager 修復を有効にして、レプリカセキュリティグループをプライマリと同期させることができます。

- 重要度 - 80
- ステータスの設定 - PASSED/FAILED
- 更新 — Firewall Manager はこの検出結果を更新します。

セキュリティグループコンテンツ監査ポリシーの検出結果

セキュリティグループコンテンツ監査ポリシーの詳細については、「セキュリティグループポリシー」を参照してください。

セキュリティグループは、コンテンツ監査セキュリティグループに準拠していません。

Firewall Manager セキュリティグループコンテンツ監査ポリシーが、非準拠のセキュリティグループを特定しました。これは、コンテンツ監査ポリシーの範囲内にあり、ポリシーとその監査セキュリティグループで定義された設定に準拠していない、お客様が作成したセキュリティグループです。ポリシーで Firewall Manager 修復を有効にできます。これにより、非準拠のセキュリティグループが変更され、コンプライアンス準拠状態になります。

- 重要度 - 70
- ステータスの設定 - PASSED/FAILED
- 更新 — Firewall Manager はこの検出結果を更新します。

セキュリティグループ使用状況監査ポリシーの検出結果

セキュリティグループ使用状況監査ポリシーの詳細については、「セキュリティグループポリシー」を参照してください。

Firewall Manager により、冗長セキュリティグループが検出されました。

Firewall Manager セキュリティグループ使用状況監査により、冗長セキュリティグループが特定されました。これは、同じ Amazon Virtual Private Cloud インスタンス内の別のセキュリティグループと同じルールが設定されたセキュリティグループです。使用状況監査ポリシーで Firewall Manager 自動修復を有効にできます。これにより、冗長セキュリティグループが 1 つのセキュリティグループで置き換えられます。

- 重要度 - 30
- ステータス設定 - なし
- 更新 — Firewall Manager はこの検出結果を更新しません。

Firewall Manager により、未使用のセキュリティグループが検出されました。

Firewall Manager セキュリティグループ使用状況監査により、未使用セキュリティグループが特定されました。これは、Firewall Manager の共通セキュリティグループポリシーによって参照されていないセキュリティグループです。使用状況監査ポリシーで Firewall Manager 自動修復を有効にできます。これにより、未使用のセキュリティグループが削除されます。

- 重要度 - 30
- ステータス設定 - なし
- 更新 — Firewall Manager はこの検出結果を更新しません。

Amazon Route 53 Resolver DNS Firewall ポリシーの検出結果

DNS Firewall ポリシーの詳細については、「Amazon Route 53 Resolver DNS Firewall ポリシー」を参照してください。

DNS Firewall 保護がリソースにありません

VPC に、Firewall Manager の DNS Firewall ポリシーで定義されている DNS Firewall ルールグループの関連付けがありません。この検出結果には、ポリシーで指定されたルールグループが一覧表示されます。

- 重要度 - 80

AWS Firewall Manager サービスの使用におけるセキュリティ

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からメリットを得られます。

> **Note**
>
> このセクションでは、Firewall Manager Network Firewall ポリシーや AWS セキュリティグループポリシーなど、 サービスとその AWS リソースを使用する AWS Firewall Manager ための標準的なセキュリティガイダンスを提供します。
> Firewall Manager を使用して AWS リソースを保護する方法については、Firewall Manager ガイドの残りの部分を参照してください。

セキュリティは、 AWS とユーザーの間で共有される責任です。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して説明しています。

- クラウドのセキュリティ – AWS は、 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責任を担います AWS クラウド。また、 は、お客様が安全に使用できるサービス AWS も提供します。セキュリティの有効性は、AWS コンプライアンスプログラムの一環として、サードパーティーの審査機関によって定期的にテストおよび検証されています。Firewall Manager に適用されるコンプライアンスプログラムについては、「コンプライアンスプログラムによる対象範囲内の AWS のサービス」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、組織の要件、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、Firewall Manager の使用時に責任共有モデルがどのように適用されるかを理解するために役立ちます。次のトピックでは、セキュリティおよびコンプライアンスの目的を達成するために Firewall Manager を設定する方法を示します。また、Firewall Manager リソースのモニタリングや保護に役立つ他の AWS のサービスの使用方法についても説明します。

トピック

Firewall Manager でのデータ保護

AWS のデータ保護には、https://aws.amazon.com/compliance/shared-responsibility-model/、（責任分担モデル）が適用されます AWS Firewall Manager。このモデルで説明したように、 AWS は、AWS クラウドすべてを稼働させ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。お客様は、このインフラストラクチャでホストされてい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、「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。欧州でのデータ保護の詳細については、「AWS セキュリティブログ」に投稿された「AWS 責任共有モデルおよび GDPR」のブログ記事を参照してください。

データ保護のため、 AWS アカウント 認証情報を保護し、 AWS IAM Identity Center 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。こうすると、それぞれのジョブを遂行するために必要なアクセス許可のみを各ユーザーに付与できます。また、以下の方法でデータを保護することをお勧めします。

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用してリソースと通信します。 AWS TLS 1.2、できれば TLS 1.3 が必要です。

- を使用して API とユーザーアクティビティのロギングを設定します。 AWS CloudTrail
- AWS 暗号化ソリューションと、 AWS のサービスその中に含まれるデフォルトのセキュリティコントロールをすべて使用してください。
- Amazon Macie などの高度なマネージド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 を介してアクセスするときに FIPS 140-2 で検証された暗号モジュールが必要な場合は、FIPS エンドポイントを使用してください。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) 140-2」を参照してください。

お客様の E メールアドレスなどの機密情報やセンシティブ情報は、タグや名前フィールドなどの自由形式のフィールドに配置しないことを強くお勧めします。これには、Firewall Manager やその他のコンソール、API AWS CLI、または AWS SDK AWS のサービス を使用して作業する場合も含まれます。名前に使用する自由記述のテキストフィールドやタグに入力したデータは、課金や診断ログに使用される場合があります。外部サーバーへの URL を提供する場合は、そのサーバーへのリクエストを検証するための認証情報を URL に含めないように強くお勧めします。

Firewall Manager のエンティティ (ポリシーなど) は、中国 (北京) や中国 (寧夏) など、暗号化が利用できない特定のリージョンを除き、保管時に暗号化されます。リージョンごとに一意の暗号化キーが使用されます。

の Identity and Access Management AWS Firewall Manager

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 (サインインについて) 認証し、Firewall Manager リソースの使用について誰を認可 (アクセス許可を付与) するのかを制御します。IAM は、追加料金なしで AWS のサービス 使用できる です。

トピック

- 対象者
- 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
- が IAM と AWS Firewall Manager 連携する方法
- のアイデンティティベースのポリシーの例 AWS Firewall Manager
- AWS の マネージドポリシー AWS Firewall Manager

- AWS Firewall Manager ID とアクセスのトラブルシューティング
- Firewall Manager のサービスにリンクされたロールの使用
- サービス間の混乱した代理の防止

対象者

AWS Identity and Access Management (IAM) の使用方法は、Firewall Manager で行う作業によって異なります。

サービスユーザー – ジョブを実行するために Firewall Manager サービスを使用するユーザーには、必要な認証情報とアクセス許可を管理者が付与します。さらに多くの Firewall Manager の機能を使用して作業を行う際には、追加のアクセス許可が必要になる場合があります。アクセスの管理方法を理解しておくと、管理者に適切な許可をリクエストするうえで役立ちます。アクセスが不可能な Firewall Manager の機能がある場合は、「AWS Shield ID 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。

サービス管理者 – 社内で Firewall Manager のリソースを担当しているユーザーには、通常、Firewall Manager への完全なアクセス権が付与されます。Firewall Manager 機能やリソースに対するアクセス権を、サービスを利用しているユーザーの中の誰に付与するかを決めるのは、管理者の仕事です。その後、IAM 管理者にリクエストを送信して、サービスユーザーの権限を変更する必要があります。このページの情報を点検して、IAM の基本概念を理解してください。社内で Firewall Manager と IAM をどのように使用できるかの詳細については、「と IAM の AWS Shield 連携方法」を参照してください。

IAM 管理者 – 管理者は、Firewall Manager へのアクセスを管理するポリシーの作成方法について、詳しく理解しておく必要があります。IAM で使用できる、Firewall Manager でのアイデンティティベースのポリシーの例は、「AWS Shieldのアイデンティティベースのポリシーの例」で確認してください。

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。として、IAM ユーザーとして AWS アカウントのルートユーザー、または IAM ロールを引き受けて認証 (にサインイン AWS) される必要があります。

ID ソースを介して提供された認証情報を使用して、フェデレーティッド ID AWS として にサインインできます。 AWS IAM Identity Center （ IAM Identity Center) ユーザー、会社のシングルサインオン認証、Google または Facebook の認証情報は、フェデレーティッド ID の例です。フェデレー

ティッド ID としてサインインする場合、IAM ロールを使用して、前もって管理者により ID フェデレーションが設定されています。フェデレーション AWS を使用して にアクセスすると、間接的にロールを引き受けることになります。

ユーザーのタイプに応じて、 AWS Management Console または AWS アクセスポータルにサインインできます。へのサインインの詳細については AWS、「 ユーザーガイド」の「 にサインインする方法 AWS アカウントAWS サインイン 」を参照してください。

AWS プログラムで にアクセスする場合、 は Software Development Kit (SDK) とコマンドラインインターフェイス (CLI) AWS を提供し、認証情報を使用してリクエストに暗号で署名します。 AWS ツールを使用しない場合は、リクエストに自分で署名する必要があります。推奨される方法を使用してリクエストを自分で署名する方法の詳細については、IAM ユーザーガイドの API AWS リクエストの署名を参照してください。

使用する認証方法を問わず、追加セキュリティ情報の提供をリクエストされる場合もあります。例えば、 AWS では、多要素認証 (MFA) を使用してアカウントのセキュリティを向上させることをお勧めします。詳細については、「AWS IAM Identity Center ユーザーガイド」の「Multi-factor authentication」(多要素認証) および「IAM ユーザーガイド」の「AWSでの多要素認証 (MFA) の使用」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、アカウント内のすべての 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 1 つのサインインアイデンティティから始めます。この ID は AWS アカウント ルートユーザーと呼ばれ、アカウントの作成に使用した E メールアドレスとパスワードでサインインすることでアクセスできます。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。ルートユーザーの認証情報は保護し、ルートユーザーでしか実行できないタスクを実行するときに使用します。ルートユーザーとしてサインインする必要があるタスクの完全なリストについては、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。

フェデレーティッドアイデンティティ

ベストプラクティスとして、管理者アクセスを必要とするユーザーを含む人間のユーザーに、一時的な認証情報を使用して にアクセスするための ID プロバイダーとのフェデレーションの使用を要求 AWS のサービス します。

フェデレーティッド ID は、エンタープライズユーザーディレクトリ、ウェブ ID プロバイダー、AWS Directory Service、アイデンティティセンターディレクトリのユーザー、または ID ソースを通じて提供された認証情報 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

ティッド ID が にアクセスすると AWS アカウント、ロールを引き受け、ロールは一時的な認証情報を提供します。

アクセスを一元管理する場合は、 AWS IAM Identity Centerを使用することをお勧めします。IAM Identity Center でユーザーとグループを作成することも、独自の ID ソース内のユーザーとグループのセットに接続して同期して、すべての AWS アカウント とアプリケーションで使用することもできます。IAM Identity Center の詳細については、「AWS IAM Identity Center ユーザーガイド」の「What is IAM Identity Center?」(IAM Identity Center 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、単一のユーザーまたはアプリケーションに対して特定のアクセス許可 AWS アカウント を持つ 内のアイデンティティです。可能であれば、パスワードやアクセスキーなどの長期的な認証情報を保有する IAM ユーザーを作成する代わりに、一時的な認証情報を使用することをお勧めします。ただし、IAM ユーザーでの長期的な認証情報が必要な特定のユースケースがある場合は、アクセスキーをローテーションすることをお勧めします。詳細については、IAM ユーザーガイドの長期的な認証情報を必要とするユースケースのためにアクセスキーを定期的にローテーションするを参照してください。

IAM グループは、IAM ユーザーの集団を指定するアイデンティティです。グループとしてサインインすることはできません。グループを使用して、複数のユーザーに対して一度に権限を指定できます。多数のユーザーグループがある場合、グループを使用することで権限の管理が容易になります。例えば、IAMAdminsという名前のグループを設定して、そのグループにIAM リソースを管理する許可を与えることができます。

ユーザーは、ロールとは異なります。ユーザーは 1 人の人または 1 つのアプリケーションに一意に関連付けられますが、ロールはそれを必要とする任意の人が引き受けるようになっています。ユーザーには永続的な長期の認証情報がありますが、ロールでは一時的な認証情報が提供されます。詳細については、「IAM ユーザーガイド」の「IAM ユーザー (ロールではなく) の作成が適している場合」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 AWS アカウント を持つ 内のアイデンティティです。これは IAM ユーザーに似ていますが、特定のユーザーには関連付けられていません。ロール を切り替える AWS Management Console ことで、 で IAM ロールを一時的に引き受けることができます。ロールを引き受けるには、 または AWS API AWS CLI オペレーションを呼び出すか、カスタム URL を使用します。ロールを使用する方法の詳細については、「IAM ユーザーガイド」の「IAM ロールの使用」を参照してください。

IAM ロールと一時的な認証情報は、次の状況で役立ちます:

- フェデレーションユーザーアクセス – フェデレーティッド ID に許可を割り当てるには、ロールを作成してそのロールの許可を定義します。フェデレーティッド ID が認証されると、その ID はロールに関連付けられ、ロールで定義されている許可が付与されます。フェデレーションの詳細については、「IAM ユーザーガイド」の「Creating a role for a third-party Identity Provider」(サードパーティーアイデンティティプロバイダー向けロールの作成) を参照してください。IAM Identity Center を使用する場合は、許可セットを設定します。アイデンティティが認証後にアクセスできるものを制御するため、IAM Identity Center は、権限セットを IAM のロールに関連付けます。アクセス許可セットの詳細については、「AWS IAM Identity Center ユーザーガイド」の「アクセス許可セット」を参照してください。
- 一時的な IAM ユーザー権限 - IAM ユーザーまたはロールは、特定のタスクに対して複数の異なる権限を一時的に IAM ロールで引き受けることができます。
- クロスアカウントアクセス - IAM ロールを使用して、自分のアカウントのリソースにアクセスすることを、別のアカウントの人物 (信頼済みプリンシパル) に許可できます。クロスアカウントアクセス権を付与する主な方法は、ロールを使用することです。ただし、一部の では AWS のサービス、 (ロールをプロキシとして使用する代わりに) ポリシーをリソースに直接アタッチできます。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。
- クロスサービスアクセス — 一部の は、他の の機能 AWS のサービス を使用します AWS のサービス。例えば、あるサービスで呼び出しを行うと、通常そのサービスによって Amazon EC2 でアプリケーションが実行されたり、Amazon S3 にオブジェクトが保存されたりします。サービスでは、呼び出し元プリンシパルの許可、サービスロール、またはサービスリンクロールを使用してこれを行う場合があります。
  - 転送アクセスセッション (FAS) – IAM ユーザーまたはロールを使用して でアクションを実行する場合 AWS、ユーザーはプリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストリクエストリクエストと組み合わせて使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。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。
  - サービスロール - 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 のサービスにアクセス許可を委任するロールの作成」を参照してください。

- サービスにリンクされたロール – 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、サービスにリンクされた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。

- Amazon EC2 で実行されているアプリケーション – IAM ロールを使用して、EC2 インスタンスで実行され、 AWS CLI または AWS API リクエストを行うアプリケーションの一時的な認証情報を管理できます。これは、EC2 インスタンス内でのアクセスキーの保存に推奨されます。 AWS ロールを EC2 インスタンスに割り当て、そのすべてのアプリケーションで使用できるようにするには、インスタンスにアタッチされたインスタンスプロファイルを作成します。インスタンスプロファイルにはロールが含まれ、EC2 インスタンスで実行されるプログラムは一時的な認証情報を取得できます。詳細については、IAM ユーザーガイドのAmazon EC2 インスタンスで実行されるアプリケーションに IAM ロールを使用して許可を付与するを参照してください。

IAM ロールと IAM ユーザーのどちらを使用するかについては、IAM ユーザーガイドの(IAM ユーザーではなく) IAM ロールをいつ作成したら良いのか?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、 AWS ID またはリソースにアタッチします。ポリシーは AWS 、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ときにアクセス許可を定義する のオブジェクトです。 は、プリンシパル (ユーザー、ルートユーザー、またはロールセッション) 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。ポリシーでの権限により、リクエストが許可されるか拒否されるかが決まります。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構造と内容の詳細については、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 AWS JSON 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んなリソースにどんな条件でアクションを実行できるかということです。

デフォルトでは、ユーザーやロールに権限はありません。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。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き継ぐことができます。

IAM ポリシーは、オペレーションの実行方法を問わず、アクションの許可を定義します。例えば、`iam:GetRole`アクションを許可するポリシーがあるとします。そのポリシーを持つユーザーは、 AWS Management Console、、 AWS CLIまたは AWS API からロール情報を取得できます。

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ポリシーは、IAM ユーザーグループ、ユーザーのグループ、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 ポリシーの作成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、さらにインラインポリシーまたはマネージドポリシーに分類できます。インラインポリシーは、単一のユーザー、グループ、またはロールに直接埋め込まれています。管理ポリシーは、 内の複数のユーザー、グループ、ロールにアタッチできるスタンドアロンポリシーです AWS アカウント。管理ポリシーには、 AWS 管理ポリシーとカスタマー管理ポリシーが含まれます。マネージドポリシーまたは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 ユーザーガイドのマネージドポリシーとインラインポリシーの比較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。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制御できます。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は、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。

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。

アクセスコントロールリスト (ACL)

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。

Amazon S3、 AWS WAF、および Amazon VPC は、ACLs。ACL の詳細については、Amazon Simple Storage Service デベロッパ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。

その他のポリシータイプ

AWS は、一般的ではない追加のポリシータイプをサポートします。これらのポリシータイプでは、より一般的なポリシータイプで付与された最大の権限を設定できます。

- アクセス許可の境界 - アクセス許可の境界は、アイデンティティベースのポリシーによって IAM エンティティ (IAM ユーザーまたはロール) に付与できる権限の上限を設定する高度な機能です。エンティティにアクセス許可の境界を設定できます。結果として得られる権限は、エンティティのアイデンティティベースポリシーとそのアクセス許可の境界の共通部分になります。`Principal` フィールドでユーザーまたはロールを指定するリソースベースのポリシーでは、アクセス許可の境界は制限されません。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。アクセス許可の境界の詳細については、IAM ユーザーガイドのIAM エンティティのアクセス許可の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 (SCPs) – SCPs は、 の組織または組織単位 (OU) に対する最大アクセス許可を指定する JSON ポリシーです AWS Organizations。 AWS Organizations は、 AWS アカウント ビジネスが所有する複数の をグループ化して一元管理するサービスです。組織内のすべての機能を有効にすると、サービスコントロールポリシー (SCP) を一部またはすべてのアカウントに適用できます。SCP は、各 を含むメンバーアカウントのエンティティのアクセス許可を制限します AWS アカウントのルートユーザー。Organizations と SCP の詳細については、AWS Organizations ユーザーガイドの「SCP の仕組み」を参照してください。
- セッションポリシー - セッションポリシーは、ロールまたはフェデレーションユーザーの一時的なセッションをプログラムで作成する際にパラメータとして渡す高度なポリシーです。結果としてセッションの権限は、ユーザーまたはロールのアイデンティティベースポリシーとセッションポリシーの共通部分になります。また、リソースベースのポリシーから権限が派生する場合もあります。これらのポリシーのいずれかを明示的に拒否した場合、権限は無効になります。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権限を理解するのがさらに難しくなります。複数のポリシータイプが関係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。

が IAM と AWS Firewall Manager 連携する方法

IAM を使用して Firewall Manager へのアクセスを管理する前に、Firewall Manager で使用できる IAM の機能を把握してください。

で使用できる IAM の機能 AWS Firewall Manager

IAM 機能	Firewall Manager 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	はい
ポリシー条件キー (サポート固有)	いいえ
ACL	なし
ABAC (ポリシー内のタグ)	はい
一時的な認証情報	あり
転送アクセスセッション (FAS)	あり
サービスロール	部分的
サービスリンクロール	あり

Firewall Manager 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を把握するには、「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。

Firewall Manager 用 ID ベースのポリシー

アイデンティティベースポリシーをサポートする	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 ポリシーの作成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。プリンシパルは、それが添付されているユーザーまたはロールに適用されるため、アイデンティティベースのポリシーでは指定できません。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。

Firewall Manager でのアイデンティティベースのポリシーの例については、「のアイデンティティベースのポリシーの例 AWS Firewall Manager」を参照してください。

Firewall Manager のための ID ベースのポリシー例

Firewall Manager でのアイデンティティベースのポリシーの例については、「のアイデンティティベースのポリシーの例 AWS Firewall Manager」を参照してください。

Firewall Manager 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制御できます。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は、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、アカウント全体、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。リソースベースのポリシーにクロスアカウントのプリンシパルを追加しても、信頼関係は半分しか確立されない点に注意してください。プリンシパルとリソースが異なる にある場合 AWS アカウント、信頼されたアカウントの

IAM 管理者は、プリンシパルエンティティ (ユーザーまたはロール) にリソースへのアクセス許可も付与する必要があります。IAM 管理者は、アイデンティティベースのポリシーをエンティティにアタッチすることで権限を付与します。ただし、リソースベースのポリシーで、同じアカウントのプリンシパルへのアクセス権が付与されている場合は、アイデンティティベースのポリシーをさらに付与する必要はありません。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。

Firewall Manager のポリシーアクション

ポリシーアクションに対するサポート	あり

管理者は AWS JSON 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んなリソースにどんな条件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。ポリシーアクションの名前は通常、関連付けられた AWS API オペレーションと同じです。一致する API オペレーションのない許可のみのアクションなど、いくつかの例外があります。また、ポリシーに複数のアクションが必要なオペレーションもあります。これらの追加アクションは、依存アクションと呼ばれます。

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権限を付与するポリシーで使用されます。

Firewall Manager アクションのリストを確認するには、「サービス認証リファレンス」の「AWS Firewall Managerで定義されるアクション」を参照してください。

Firewall Manager のポリシーアクションには、次のプレフィックスを付加します。

```
fms
```

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": [
      "fms:action1",
      "fms:action2"
         ]
```

ワイルドカード (*) を使用して複数アクションを指定できます。例えば、Describe という単語で始まるすべてのアクションを指定するには、次のアクションを含めます。

```
"Action": "fms:Describe*"
```

Firewall Manager でのアイデンティティベースのポリシーの例については、「のアイデンティティベースのポリシーの例 AWS Firewall Manager」を参照してください。

Firewall Manager のポリシーリソース

ポリシーリソースに対するサポート	あり

管理者は AWS JSON 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どのような条件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。ステートメントには、Resource または NotResource 要素を含める必要があります。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。これは、リソースレベルの許可と呼ばれる特定のリソースタイプをサポートするアクションに対して実行できます。

オペレーションのリスト化など、リソースレベルの権限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": "*"
```

Firewall Manager でのリソースタイプとその ARN のリストを確認するには、「サービス認証リファレンス」の「AWS Firewall Managerで定義されるリソースタイプ」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は、「AWS Firewall Managerで定義されるアクション」を参照してください。

Firewall Manager でのアイデンティティベースのポリシーの例については、「のアイデンティティベースのポリシーの例 AWS Firewall Manager」を参照してください。

Firewall Manager 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	なし

管理者は AWS JSON 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んなリソースにどんな条件でアクションを実行できるかということです。

Condition 要素 (または Condition ブロック) を使用すると、ステートメントが有効な条件を指定できます。Condition 要素はオプションです。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することで、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。

1 つのステートメントに複数の Condition 要素を指定する場合、または 1 つの Condition 要素に複数のキーを指定する場合、 AWS では AND 論理演算子を使用してそれらを評価します。1 つの条件キーに複数の値を指定すると、 は論理ORオペレーションを使用して条件 AWS を評価します。ステートメントの権限が付与される前にすべての条件が満たされる必要があります。

条件を指定する際にプレースホルダー変数も使用できます。例えば IAM ユーザーに、IAM ユーザー名がタグ付けされている場合のみリソースにアクセスできる権限を付与することができます。詳細については、IAM ユーザーガイドのIAM ポリシーの要素: 変数およびタグを参照してください。

AWS は、グローバル条件キーとサービス固有の条件キーをサポートします。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。

Firewall Manager の条件キーのリストを確認するには、「サービス認証リファレンス」 の「AWS Firewall Managerの条件キー」を参照してください。条件キーを使用できるアクションとリソースについては、「 で定義されるアクション AWS Firewall Manager」を参照してください。

Firewall Manager でのアイデンティティベースのポリシーの例については、「のアイデンティティベースのポリシーの例 AWS Firewall Manager」を参照してください。

Firewall Manager の 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許可を持つかをコントロールします。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。

Firewall Manager を使用する ABAC

ABAC のサポート (ポリシー内のタグ)	はい

属性ベースのアクセス制御 (ABAC) は、属性に基づいてアクセス許可を定義するアクセス許可戦略です。では AWS、これらの属性はタグ と呼ばれます。タグは、IAM エンティティ (ユーザーまたはロール) および多くの AWS リソースにアタッチできます。エンティティとリソースのタグ付けは、ABAC の最初の手順です。その後、プリンシパルのタグがアクセスしようとしているリソースのタグと一致した場合にオペレーションを許可するように ABAC ポリシーをします。

ABAC は、急成長する環境やポリシー管理が煩雑になる状況で役立ちます。

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 とは?を参照してください。ABAC をセットアップするステップ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に基づくアクセスコントロール (ABAC) を使用するを参照してください。

Firewall Manager 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

一部の AWS のサービス は、一時的な認証情報を使用してサインインすると機能しません。一時的な認証情報 AWS のサービス を使用する などの詳細については、IAM ユーザーガイドのAWS のサービス「IAM と連携する 」を参照してください。

ユーザー名とパスワード以外の AWS Management Console 方法で にサインインする場合、一時的な認証情報を使用します。例えば、会社の Single Sign-On (SSO) リンク AWS を使用して にアクセスすると、そのプロセスによって一時的な認証情報が自動的に作成されます。また、ユーザーとしてコンソールにサインインしてからロールを切り替える場合も、一時的な認証情報が自動的に作成されます。ロールの切り替えに関する詳細については、IAM ユーザーガイドのロールへの切り替え (コンソール)を参照してください。

一時的な認証情報は、 AWS CLI または AWS API を使用して手動で作成できます。その後、これらの一時的な認証情報を使用して . AWS recommends にアクセスできます AWS。これは、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することを推奨しています。詳細については、IAM の一時的セキュリティ認証情報を参照してください。

転送アクセスセッション (Firewall Manager)

転送アクセスセッション (FAS) をサポート	あり

IAM ユーザーまたはロールを使用して でアクションを実行すると AWS、プリンシパルと見なされます。一部のサービスを使用する際に、アクションを実行することで、別のサービスの別のアクションがトリガーされることがあります。FAS は、 を呼び出すプリンシパルのアクセス許可を AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストのリクエストと組み合わせて使用します。FAS リクエストは、サービスが他の AWS のサービス またはリソースとのやり取りを完了する必要があるリクエストを受け取った場合にのみ行われます。この場合、両方のアクションを実行するためのアクセス許可が必要です。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。

Firewall Manager のサービスロール

サービスロールのサポート	部分的

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 のサービスにアクセス許可を委任するロールの作成」を参照してください。

> **Warning**
>
> サービスロール向けの許可を変更すると、Firewall Manager の機能が破損する可能性があります。Firewall Manager 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。

Firewall Manager での IAM ロールの選択

Firewall Manager で *PutNotificationChannel* API アクションを使用するには、サービスがユーザーに代わって Amazon SNS メッセージを発行できるように、Firewall Manager が Amazon SNS にアクセスすることを許可するロールを選択する必要があります。詳細については、 API リファレンスPutNotificationChannelの「」を参照してください。 AWS Firewall Manager

SNS トピックでのアクセス許可設定の例を次に示します。このポリシーを独自のカスタムロールで使用するには、Amazon リソースネーム (ARN) AWSServiceRoleForFMS を SnsRoleName のARN に置き換えます。

```
{
  "Sid": "AWSFirewallManagerSNSPolicy",
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::account ID:role/aws-service-role/
fms.amazonaws.com/AWSServiceRoleForFMS"
  },
  "Action": "sns:Publish",
  "Resource": "SNS topic ARN"
}
```

Firewall Manager のアクションとリソースの詳細については、 AWS Identity and Access Management 「 で定義されるアクション」ガイドトピックを参照してください。 AWS Firewall Manager

Firewall Manager の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、サービスにリンクされた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。

サービスリンク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。表の中から、[Service-linked role] (サービスにリンクされたロール) 列に Yes と記載されたサービスを見つけます。サービスリンク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。

のアイデンティティベースのポリシーの例 AWS Firewall Manager

デフォルトでは、ユーザーおよびロールに Firewall Manager のリソースを作成または変更する許可が付与されていません。また、、 AWS Command Line Interface （ AWS CLI ） AWS Management Console、または AWS API を使用してタスクを実行することはできません。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。その後、管理者はロールに IAM ポリシーを追加し、ユーザーはロールを引き受けることができます。

これらサンプルの 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 ポリシーの作成」を参照してください。

Firewall Manager 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、「サービス認証リファレンス」の「AWS Firewall Manager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

- ポリシーのベストプラクティス
- Firewall Manager コンソールの使用
- 自分の権限の表示をユーザーに許可する
- Firewall Manager のセキュリティグループに読み取りアクセス権を付与する

ポリシーのベストプラクティス

ID ベースのポリシーは、アカウント内で誰かが Firewall Manager のリソースを作成、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を実行すると、 AWS アカウントに料金が発生する可能性があります。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を付与するには、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して、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「AWS ジョブ機能の管理ポリシー」を参照してください。
- 最小特権を適用する – IAM ポリシーで権限を設定するときは、タスクの実行に必要な権限のみを付与します。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、最小特権権限とも呼ばれています。IAM を使用して権限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 でのポリシーと権限」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。例えば、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、 などの特定の を介してサービスアクションが使用される場合に 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 AWS CloudFormation。詳細については、『IAM ユーザーガイド』の [IAM JSON policy elements: Condition] (IAM JSON ポリシー要素：条件) を参照してください。

- IAM Access Analyzer 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 Access Analyzer 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 ポリシーの検証」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、セキュリティを強化するために MFA を有効にします。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 保護 API アクセスの設定」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。

Firewall Manager コンソールの使用

AWS Firewall Manager コンソールにアクセスするには、最小限のアクセス許可のセットが必要です。これらのアクセス許可により、 AWS アカウント内の Firewall Manager リソースの詳細をリスト化および表示できます。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 Firewall Manager 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 Firewall Manager *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「ユーザーへのアクセス許可の追加」を参照してください。

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": "*"
        }
    ]
}
```

Firewall Manager のセキュリティグループに読み取りアクセス権を付与する

Firewall Manager は、ユーザーにクロスアカウントでのリソースアクセスを許可しますが、クロスアカウントでのリソース保護を作成することは許可しません。リソースの保護は、それらのリソースを所有するアカウント内からのみ作成できます。

すべてのリソースの fms:Get 、fms:List、および ec2:DescribeSecurityGroups アクションへの、アクセス許可を付与するポリシーの例を次に示します。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "fms:Get*",
                "fms:List*",
                "ec2:DescribeSecurityGroups"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ]
}
```

AWS の マネージドポリシー AWS Firewall Manager

AWS 管理ポリシーは、 によって作成および管理されるスタンドアロンポリシーです AWS。 AWS 管理ポリシーは、多くの一般的なユースケースに対するアクセス許可を付与するように設計されているため、ユーザー、グループ、ロールへのアクセス許可の割り当てを開始できます。

AWS 管理ポリシーは、すべての AWS お客様が使用できるため、特定のユースケースに対して最小特権のアクセス許可を付与しない場合があることに注意してください。ユースケース別にカスタマーマネージドポリシーを定義して、マネージドポリシーを絞り込むことをお勧めします。

AWS 管理ポリシーで定義されているアクセス許可は変更できません。が AWS 管理ポリシーで定義されたアクセス許可 AWS を更新すると、ポリシーがアタッチされているすべてのプリンシパル ID (ユーザー、グループ、ロール) が更新されます。 は、新しい AWS のサービス が起動されたとき、または既存のサービスで新しい API AWS オペレーションが使用可能になったときに、 AWS 管理ポリシーを更新する可能性が最も高くなります。

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 マネージドポリシー: **AWSFMAdminFullAccess**

AWSFMAdminFullAccess AWS 管理ポリシーを使用して、管理者がすべての Firewall Manager ポリシータイプを含む AWS Firewall Manager リソースにアクセスできるようにします。このポリシーには、 AWS Firewall Managerで Amazon Simple Notification Service 通知を設定するためのアクセス許可は含まれません。Amazon Simple Notification Service のアクセスを設定する方法については、「Setting up access for Amazon Simple Notification Service」(Amazon Simple Notification Service のアクセスをセットアップする) を参照してください。

ポリシーのリストと詳細については、「」の「IAM コンソール」を参照してくださいAWSFMAdminFullAccess。このセクションの残りの部分では、ポリシー設定の概要を説明します。

アクセス許可ステートメント

このポリシーは、一連のアクセス許可に基づくステートメントにグループ化されます。

- AWS Firewall Manager ポリシーリソース - すべての Firewall Manager ポリシータイプを含む AWS Firewall Manager、 のリソースへの完全な管理アクセス許可を許可します。
- Amazon Simple Storage Service への AWS WAF ログの書き込み - Firewall Manager が Amazon S3 で AWS WAF ログの書き込みと読み取りを行うことを許可します。
- サービスにリンクされたロールの作成 – 管理者がサービスにリンクされたロールを作成できるようにします。これにより、Firewall Manager はユーザーに代わって他の サービスのリソースにアクセスできます。このアクセス許可では、Firewall Manager が使用するサービスリンクロールのみを作成できます。Firewall Manager がサービスにリンクされたロールを使用する方法については、「」を参照してくださいFirewall Manager のサービスにリンクされたロールの使用。
- AWS Organizations — 管理者による AWS Organizationsの組織への Firewall Manager の使用を許可します。で Firewall Manager の信頼されたアクセスを有効にすると AWS Organizations、管理者アカウントのメンバーは組織全体の結果を表示できます。 AWS Organizations で を使用する方法については AWS Firewall Manager、「 ユーザーガイドAWS Organizations 」の「 を他の AWS のサービスで使用するAWS Organizations 」を参照してください。

アクセス許可カテゴリ

以下は、ポリシー内のアクセス許可のタイプと、それらが提供するアクセス許可の一覧です。

- fms – AWS Firewall Manager リソースを使用します。
- waf および waf-regional — Classic ポリシーを使用します AWS WAF 。

- elasticloadbalancing — AWS WAF ウェブ ACLsto に関連付けます。
- firehose – AWS WAF ログに関する情報を表示します。
- organizations – AWS Organizations リソースを使用します。
- shield – ポリシーの AWS Shield サブスクリプション状態を表示します。
- route53resolver — VPCs 用 Route 53 プライベート DNS VPCs ポリシーで、VPC 用 Route 53 プライベート DNS ルールグループを使用します。
- wafv2 – AWS WAFV2 ポリシーを使用します。
- network-firewall – AWS Network Firewall ポリシーを使用します。
- ec2 – ポリシーのアベイラビリティーゾーンとリージョンを表示します。
- s3 – AWS WAF ログに関する情報を表示します。

AWS マネージドポリシー: **FMSServiceRolePolicy**

このポリシーにより、 AWS Firewall Manager は Firewall Manager および統合サービスでユーザーに代わって AWS リソースを管理できます。このポリシーは、AWSServiceRoleForFMS サービスにリンクされたロールにアタッチされます。サービスにリンクされたロールの詳細については、「Firewall Manager 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。

ポリシーの詳細については、FMSServiceRolePolicy の IAM コンソールを参照してください。

AWS マネージドポリシー： AWSFMAdminReadOnlyAccess

すべての AWS Firewall Manager リソースへの読み取り専用アクセスを許可します。

ポリシーのリストと詳細については、「」の「IAM コンソール」を参照してくださいAWSFMAdminReadOnlyAccess。このセクションの残りの部分では、ポリシー設定の概要を説明します。

アクセス許可カテゴリ

以下に、ポリシー内のアクセス許可のタイプと、アクセス許可が読み取り専用アクセスを許可する情報を示します。

- fms - AWS Firewall Manager リソース。
- waf および waf-regional — AWS WAF クラシックポリシー。
- firehose – AWS WAF ログ。
- organizations - AWS 組織リソース。

- `shield` – AWS Shield ポリシー。
- `route53resolver` – Route 53 Private DNS for VPCsポリシー内の VPCs ルールグループの Route 53 Private DNS。
- `wafv2` – で利用可能な AWS WAFV2 ルールグループと AWS マネージドルールのルールグループ AWS WAFV2。
- `network-firewall` – AWS Network Firewall ルールグループとルールグループのメタデータ。
- `ec2` – AWS Network Firewall ポリシーアベイラビリティーゾーンとリージョン 。
- `s3` – AWS WAF ログ。

AWS マネージドポリシー： AWSFMMemberReadOnlyAccess

AWS Firewall Manager メンバーリソースへの読み取り専用アクセスを許可します。
ポリシーのリストと詳細については、「」の「IAM コンソール」を参照してくださ
いAWSFMMemberReadOnlyAccess。

AWS マネージドポリシーに対する Firewall Manager の更新

Firewall Manager の AWS マネージドポリシーの更新に関する詳細を、このサービスがこれらの変更の追跡を開始した以降の分について表示します。このページへの変更に関する自動アラートについては、Firewall Manager のドキュメント履歴ページ (ドキュメント履歴) で 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
FMSServiceRolePolicy – 更新されたポリシー	ネットワーク ACLs。 IAM コンソールで更新されたポリシーを参照してください: FMS ServiceRolePolicy。	2024-04-22
FMSServiceRolePolicy – 更新されたポリシー	Firewall Manager が、指定された AWS Config ルールが準拠しているかどうかを記述で	2023-04-21

変更	説明	日付
	きるようにするアクセス許可を追加しました。 IAM コンソールで更新されたポリシーを参照してください: FMS ServiceRolePolicy。	
FMSServiceRolePolicy – 更新されたポリシー	Firewall Manager が Amazon EC2 インスタンスとネットワークインターフェイスの属性を記述できるようにするアクセス許可が追加されました。 IAM コンソールで更新されたポリシーを参照してください: FMS ServiceRolePolicy。	2022-11-15
AWSFMAdminReadOnlyAccess - ポリシーを更新	AWS WAFV2、Shield、Network Firewall、DNS Firewall、Amazon VPC セキュリティグループ、ポリシーをサポートするアクセス許可を追加しました。 IAM コンソールで更新されたポリシーを参照してください: AWSFMAdminReadOnlyAccess。	2022-11-02

変更	説明	日付
AWSFMAdminFullAccess - ポリシーを更新	AWS WAFV2、Shield、Network Firewall、DNS Firewall、Amazon VPC セキュリティグループ、ポリシーをサポートするアクセス許可を追加しました。Amazon SNS のアクセス許可を削除しました。 IAM コンソールで更新されたポリシーを参照してください: AWSFMAdminFullAccess。	2022-10-21
FMSServiceRolePolicy – AWS Firewall Manager サードパーティーのファイアウォールポリシーに対する新しいアクセス許可	この変更により、Firewall Manager は、サードパーティーのファイアウォールポリシーに関連付けられた Amazon EC2 VPC エンドポイントを作成および削除できます。	2022-03-30
FMSServiceRolePolicy – AWS Network Firewall ポリシーの新しいアクセス許可	Network Firewall ポリシーのファイアウォールのデプロイをサポートするための新しい許可を追加しました。新しい許可により、ポリシーの範囲内にあるアカウントのアベイラビリティーゾーンに関する情報を取得できます。	2022-02-16

変更	説明	日付
FMSServiceRolePolicy – AWS Shield ポリシーの新しいアクセス許可	AWS WAF リージョンリソースと AWS WAF グローバルリソースのタグを取得するための新しいアクセス許可を追加しました。リソース ARN ACLs を取得する AWS WAF リージョンのアクセス許可を追加しました。Shield アプリケーションレイヤー DDoS 自動緩和をサポートするための許可を追加しました。	2022-01-07
FMSServiceRolePolicy – AWS Shield ポリシーの新しいアクセス許可	Elastic Load Balancing リソース用にタグを取得するための新しい許可を追加しました。	2021-11-18
FMSServiceRolePolicy – セキュリティグループと AWS Network Firewall ポリシーの新しいアクセス許可	AWS Network Firewall ポリシーの集中ログ記録を有効にする新しいアクセス許可を追加しました。さらに、 がセキュリティグループポリシーのリソースを AWS Firewall Manager クエリする方法に影響を与える Config サービスへの変更をサポートするために、読み取り専用の Amazon EC2 アクセス許可が追加されました。	2021-09-29

変更	説明	日付
FMSServiceRolePolicy – AWS WAF リソースの ARN 形式	AWS WAF リソースの ARN 形式を標準化するように FMSServiceRolePolicy を更新しました。更新された ARN 形式は arn:aws:waf:*:*:* と arn:aws:waf-regional:*:*:* です。	2021-08-12
FMSServiceRolePolicy - 中国の他のリージョン	AWS Firewall Manager はFMSServiceRolePolicy 、中国の BJS および ZHY リージョンで を有効にしました。	2021-08-12
FMSServiceRolePolicy - 既存のポリシーに対する更新	が Amazon Route 53 Resolver DNS Firewall を管理 AWS Firewall Manager できるようにする新しいアクセス許可を追加しました。 この変更により、Firewall Manager は Amazon Route 53 Resolver DNS Firewall の関連付けを設定できるようになります。これにより、Firewall Manager を使用して、 AWS Organizationsの組織全体で VPC のために DNS Firewall 保護を提供できます。	2021-03-17
Firewall Manager が変更の追跡を開始	Firewall Manager が AWS マネージドポリシーの変更の追跡を開始しました。	2021-03-02

AWS Firewall Manager ID とアクセスのトラブルシューティング

次の情報は、Firewall Manager と IAM の使用に伴って発生する可能性がある、一般的な問題を診断したり修復したりする際に役立ちます。

トピック

Firewall Manager でのアクション実行が承認されない

「I am not authorized to perform an action in Amazon Bedrock」というエラーが表示された場合、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。

次の例は、`mateojackson` という IAM ユーザーがコンソールを使用して架空の *`my-example-widget`* リソースに関する詳細を表示しようとしたとき、架空の `fms:`*`GetWidget`* アクセス許可がない場合に発生するエラーを示しています。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 fms:GetWidget on resource: my-example-widget
```

この場合、`fms:`*`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 管理者に問い合わせてください。サインイン資格情報を提供した担当者が管理者です。

私にはiam を実行する権限がありません:PassRole

`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 Firewall Manager にロールを渡せるようにする必要があります。

新しいサービスロールやサービスにリンクされたロールを作成する代わりに、 AWS のサービス 既存のロールをそのサービスに渡すことができるものもあります。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。

`marymajor` という IAM ユーザーが、コンソールを使用して Firewall Manager でアクションを実行しようとした際に発生するエラーの例を次に示します。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権限が必要です。Mary には、ロールをサービスに渡す権限がありません。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

この場合、メアリー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、管理者に問い合わせてください。 AWS サインイン資格情報を提供した担当者が管理者です。

自分以外のユーザーがFirewall Manager AWS アカウント のリソースにアクセスできるようにしたい

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、以下を参照してください:

- これらの機能を Firewall Manager でサポートしているかどうかを確認するには、「と IAM の AWS Shield 連携方法」 を参照してください。
- AWS アカウント 所有しているリソース全体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「AWS アカウント 所有する別の 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第三者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「AWS アカウント 第三者が所有するリソースへのアクセスの提供」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でのロールとリソースベースのポリシーの使用の違いの詳細については、「IAM ユーザーガイド」の「IAM ロールとリソースベースのポリシーとの相違点」を参照してください。

Firewall Manager のサービスにリンクされたロールの使用

AWS Firewall Manager 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は、Firewall Manager に直接リンクされたー

意のタイプの IAM ロールです。サービスにリンクされたロールは、Firewall Manager によって事前定義されており、ユーザーに代わってサービスから他の AWS の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ことで、必要な許可を手動で追加する必要がなくなるため、Firewall Manager の設定が簡単になります。Firewall Manager は、サービスにリンクされたロールの許可を定義します。特に定義されている場合を除き、Firewall Manager のみがそのロールを引き受けることができます。定義された許可には、信頼ポリシーと許可ポリシーが含まれます。こ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。

サービスにリンクされたロールを削除するには、まずそのロールの関連リソースを削除します。これにより、リソースに対する許可が誤って削除されることがなくなり、Firewall Manager のリソースは保護されます。

サービスにリンクされたロールをサポートする他のサービスについては、「IAM と連携するAWS のサービス」を参照し、[Service-Linked Role] (サービスにリンクされたロール) 列で [Yes] (はい) のあるサービスを探してください。そのサービスに関する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、リンクが設定されている [Yes] (はい) を選択します。

Firewall Manager でのサービスにリンクされたロールの許可

AWS Firewall Manager は、サービスにリンクされたロール名 AWSServiceRoleForFMS を使用して、Firewall Manager がユーザーに代わって AWS サービスを呼び出し、ファイアウォールポリシーと AWS Organizations アカウントリソースを管理できるようにします。このポリシーは、 AWS マネージドロール にアタッチされますAWSServiceRoleForFMS。マネージドロールの詳細については、「AWS マネージドポリシー: FMSServiceRolePolicy」を参照してください。

AWSServiceRoleForFMS サービスにリンクされたロールは、サービスを信頼してロール を引き受けますfms.amazonaws.com。

ロールの許可ポリシーは、指定したリソースに対して以下のアクションを完了することを Firewall Manager に許可します。

- waf - アカウントの AWS WAF Classic ウェブ ACLsルールグループのアクセス許可、およびウェブ ACLs関連付けを管理します。
- ec2 – Elastic Network Interface と Amazon EC2 インスタンスで、セキュリティグループを管理します。Amazon VPC ACLs を管理します。
- vpc – Amazon VPC 内のサブネット、ルートテーブル、タグ、エンドポイントを管理します。

- `wafv2` - アカウントの AWS WAF ウェブ ACLs、ルールグループのアクセス許可、およびウェブ ACLs関連付けを管理します。
- `cloudfront` - CloudFront ディストリビューションを保護するためACLs を作成します。
- `config` - アカウントで Firewall Manager が所有する AWS Config ルールを管理します。
- `iam` - このサービスにリンクされたロールを管理し、 および Shield AWS WAF ポリシーのログ記録を設定する場合は、必要な AWS WAF および Shield サービスにリンクされたロールを作成します。
- `organization` - Firewall Manager が所有するサービスにリンクされたロールを作成して、Firewall Manager が使用する AWS Organizations リソースを管理します。
- `shield` - アカウント内のリソース AWS Shield の保護と L7 緩和設定を管理します。
- `ram` - DNS Firewall ルールグループと Network Firewall ルールグループの AWS RAM リソース共有を管理します。
- `network-firewall` - アカウント内の Firewall Manager が所有する AWS Network Firewall リソースと依存する Amazon VPC リソースを管理します。
- `route53resolver` – アカウント内で、Firewall Manager が所有する DNS Firewall の関連付けを管理します。

IAM コンソールでポリシーの全文を参照してください: FMS ServiceRolePolicy。

サービスリンクロールの作成、編集、削除を IAM エンティティ (ユーザー、グループ、ロールなど) に許可するには、アクセス許可を設定する必要があります。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許可」を参照してください。

Firewall Manager のサービスにリンクされたロールの作成

サービスリンクロールを手動で作成する必要はありません。で Firewall Manager の口グ記録を有効にするか AWS Management Console、Firewall Manager CLI または Firewall Manager API で`PutLoggingConfiguration`リクエストを行うと、Firewall Manager によってサービスにリンクされたロールが作成されます。

口グ記録を有効化するためには、`iam:CreateServiceLinkedRole` 許可が必要です。

このサービスにリンクされたロールを削除した後で再度作成する必要が生じた場合は、同じ方法でアカウントにロールを再作成できます。Firewall Manager 口グ記録を有効にすると、Firewall Manager は、ユーザーのためにサービスにリンクされたロールを再作成します。

Firewall Manager のサービスにリンクされたロールの編集

Firewall Manager では、 AWSServiceRoleForFMS サービスにリンクされたロールを編集することはできません。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「サービスにリンクされたロールの編集」を参照してください。

Firewall Manager 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。そうすることで、積極的にモニタリングまたは保守されていない未使用のエンティティを排除できます。ただし、手動で削除する前に、サービスにリンクされたロールのリソースをクリーンアップする必要があります。

> **Note**
>
> リソースを削除する際に、Firewall Manager のサービスでそのロールが使用されている場合、削除は失敗することがあります。失敗した場合は、数分待ってから操作を再試行してください。

IAM を使用して、サービスにリンクされたロールを削除するには

IAM コンソール、IAM CLI、または IAM API を使用して、サービスにリンクされたロールを削除します AWSServiceRoleForFMS。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。

Firewall Manager サービスにリンクされたロールをサポートするリージョン

Firewall Manager では、このサービスが利用可能な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ています。詳細については、「Firewall Manager エンドポイントとクォータ」を参照してください。

サービス間の混乱した代理の防止

混乱した代理問題は、アクションを実行するためのアクセス許可を持たないエンティティが、より特権のあるエンティティにアクションの実行を強制できてしまう場合に生じる、セキュリティ上の問題です。では AWS、サービス間のなりすましにより、混乱した代理人問題が発生する可能性があります。サービス間でのなりすましは、1 つのサービス (呼び出し元サービス) が、別のサービス (呼び出

し対象サービス) を呼び出すときに発生する可能性があります。呼び出し元サービスは、本来ならアクセスすることが許可されるべきではない方法でその許可を使用して、別のお客様のリソースに対する処理を実行するように操作される場合があります。これを防ぐために、 AWS には、アカウント内のリソースへのアクセス権が付与されたサービスプリンシパルですべてのサービスのデータを保護するために役立つツールが用意されています。

aws:SourceArnaws:SourceAccountリソースポリシーではグローバル条件コンテキストキーとグローバル条件コンテキストキーを使用して、 AWS Firewall Manager リソースに別のサービスを付与する権限を制限することをおすすめします。クロスサービスアクセスにリソースを 1 つだけ関連付けたい場合は、aws:SourceArn を使用します。そのアカウント内のリソースをクロスサービスの使用に関連付けることを許可する場合は、aws:SourceAccount を使用します。

混乱した代理問題から保護するための最も効果的な方法は、リソースの完全な ARN を指定して、aws:SourceArn グローバル条件コンテキストキーを使用することです。リソースの完全な ARN が不明な場合や、複数のリソースを指定する場合には、グローバルコンテキスト条件キー aws:SourceArn で、ARN の未知部分を示すためにワイルドカード文字 (*) を使用します。例えば、arn:aws:fms:*:*account-id*:* です。

aws:SourceArn の値に Amazon S3 バケット ARN などのアカウント ID が含まれていない場合は、両方のグローバル条件コンテキストキーを使用して、アクセス許可を制限する必要があります。

aws:SourceArn AWS Firewall Manager AWS の値は管理者のアカウントでなければなりません。

次の例では、Firewall Manager で aws:SourceArn グローバル条件コンテキストキーを使用して、混乱した代理問題を防止する方法を示しています。

次の例では、Firewall Manager ロールの信頼ポリシーで aws:SourceArn グローバル条件コンテキストキーを使用して、混乱した代理問題を防止する方法を示しています。*[Region]* (リージョン) および *[Account-ID]* (アカウント ID) をユーザー自身の情報に置き換えます。

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Sid": "ConfusedDeputyPreventionExamplePolicy",
    "Effect": "Allow",
    "Principal": {
      "Service": "servicename.amazonaws.com"
    },
    "Action": "sts:AssumeRole",
    "Condition": {
      "ArnLike": {
```

```
        "aws:SourceArn": [
          "arn:aws:fms:Region:account-id:${*}",
          "arn:aws:fms:Region:account-id:policy/*"]
      },
      "StringEquals": {
        "aws:SourceAccount": "account-id"
      }
    }
  }
}
```

Firewall Manager でのログ記録とモニタリング

監視は、Firewall Manager AWS とソリューションの信頼性、可用性、パフォーマンスを維持する上で重要な部分です。マルチポイント障害が発生した場合により簡単にデバッグできるように、 AWS ソリューションのすべての部分から監視データを収集する必要があります。 AWS には、Firewall Manager リソースを監視し、発生する可能性のあるイベントに対応するためのツールがいくつか用意されています。

Amazon CloudWatch アラーム

CloudWatch アラームを使用すると、指定した期間にわたって 1 つのメトリクスを監視できます。メトリックスが特定のしきい値を超えると、Amazon SNS CloudWatch AWS Auto Scaling トピックまたはポリシーに通知を送信します。詳細については、「Amazon によるモニタリング CloudWatch」を参照してください。

AWS CloudTrail ログ

CloudTrail Firewall Manager でユーザー、ロール、 AWS またはサービスが実行したアクションの記録を提供します。によって収集された情報を使用して CloudTrail、Firewall Manager に対して行われた要求、要求が行われたIPアドレス、要求の実行者、要求の実行日時、およびその他の詳細を判断できます。詳細については、「での AWS CloudTrail API コールのログ記録」を参照してください。

Firewall Manager のコンプライアンス検証

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、コンプライアンスプログラムAWS のサービス による対象範囲内のコンプライアンスプログラムを参照し、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、「 でのレポートのダウンロード AWS Artifact」の」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。 は、コンプライアンスに役立つ以下のリソース AWS を提供しています。

- セキュリティとコンプライアンスのクイックスタートガイド – これらのデプロイガイドでは、アーキテクチャ上の考慮事項について説明し、セキュリティとコンプライアンスに重点を置いたベースライン環境 AWS を にデプロイする手順について説明します。
- アマゾン ウェブ サービスにおける HIPAA セキュリティとコンプライアンスのアーキテクチャ — このホワイトペーパーでは、企業が AWS を使用して HIPAA 対象アプリケーションを作成する方法について説明します。

  > **Note**
  >
  > すべて AWS のサービス HIPAA の対象となるわけではありません。詳細については、「HIPAA 対応サービスのリファレンス」を参照してください。

- AWS コンプライアンスリソース – このワークブックとガイドのコレクションは、お客様の業界や地域に適用される場合があります。
- AWS カスタマーコンプライアンスガイド — コンプライアンスの観点から責任共有モデルを理解します。このガイドでは、ガイダンスを保護し AWS のサービス 、複数のフレームワーク (米国国立標準技術研究所 (NIST)、Payment Card Industry Security Standards Council (PCI)、国際標準化機構 (ISO) を含む) のセキュリティコントロールにマッピングするためのベストプラクティスをまとめています。
- 「 デベロッパーガイド」の「 ルールによるリソースの評価」 – この AWS Config サービスは、リソース設定が社内プラクティス、業界ガイドライン、および規制にどの程度準拠しているかを評価します。 AWS Config
- AWS Security Hub – これにより AWS のサービス 、 内のセキュリティ状態を包括的に確認できます AWS。Security Hub では、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対するコンプライアンスをチェックします。サポートされているサービスとコントロールのリストについては、「Security Hub のコントロールリファレンス」を参照してください。

- Amazon GuardDuty – これにより AWS アカウント、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環境を監視することで、、、ワークロード、コンテナ、データに対する潜在的な脅威 AWS のサービス を検出します。 GuardDuty は、特定のコンプライアンスフレームワークで義務付けられている侵入検知要件を満たすことで、PCI DSS などのさまざまなコンプライアンス要件への対応に役立ちます。
- AWS Audit Manager – これにより AWS のサービス 、 AWS 使用状況を継続的に監査し、リスクの管理方法と規制や業界標準への準拠を簡素化できます。

Firewall Manager 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 アベイラビリティーゾーンを中心に構築されています。 AWS リージョン 物理的に分離された複数のアベイラビリティーゾーンを提供し、低レイテンシー、高スループット、冗長性の高いネットワークで接続します。アベイラビリティーゾーンでは、アベイラビリティーゾーン間で中断せずに、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、耐障害性、およびスケーラビリティが優れています。

AWS リージョン およびアベイラビリティーゾーンの詳細については、「グローバルインフラストラクチャ」を参照してください。AWS

AWS Firewall Manager内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として、 AWS Firewall Manager AWS グローバルなネットワークセキュリティによって保護されています。 AWS AWS セキュリティサービスとインフラストラクチャの保護方法については、「AWS Cloud Security」を参照してください。 AWS 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設計するには、「Security Pillar AWS Well-Architected Framework におけるインフラストラクチャ保護」を参照してください。

AWS 公開されている API 呼び出しを使用して、ネットワーク経由で Firewall Manager にアクセスします。クライアントは以下をサポートする必要があります:

- Transport Layer Security (TLS)。TLS 1.2、できれば TLS 1.3 が必要です。

- DHE (Ephemeral Diffie-Hellman) や ECDHE (Elliptic Curve Ephemeral Diffie-Hellman) などの Perfect Forward Secrecy (PFS) を使用した暗号スイート。これらのモードは、Java 7 以降など、ほとんどの最新システムでサポートされています。

また、リクエストには、アクセスキー ID と、IAM プリンシパルに関連付けられているシークレットアクセスキーを使用して署名する必要があります。または、AWS Security Token ServiceAWS STS を使用して、一時的なセキュリティ認証情報を生成し、リクエストに署名することもできます。

AWS Firewall Manager クォータ

AWS Firewall Manager には、次のクォータ (以前は制限と呼ばれていました) が適用されます。

AWS Firewall Manager には、クォータを増やすことができるデフォルトのクォータと固定クォータがあります。

Firewall Manager によって管理されるセキュリティグループポリシーとネットワーク ACL ポリシーには、標準の Amazon VPC クォータが適用されます。詳細については、「Amazon VPC ユーザーガイド」の「Amazon VPC クォータ」を参照してください。

各 Firewall Manager の Network Firewall ポリシーは、関連付けられたファイアウォールポリシーとそのルールグループを使用して Network Firewall ファイアウォールを作成します。これらの Network Firewall リソースは、「Network Firewall デベロッパーガイド」の「AWS Network Firewall クォータ」にリストされているクォータを前提とします。

ソフトクォータ

AWS Firewall Manager には、リージョンあたりのエンティティ数に対するデフォルトのクォータがあります。このクォータの引き上げをリクエストできます。

すべてのポリシータイプ

リソース	リージョンあたりのデフォルトのクォータ
の組織あたりのアカウント AWS Organizations	可変。アカウントに送信された招待はこのクォータに対してカウントされます。

リソース	リージョンあたりのデフォルトのクォータ
	招待されたアカウントが拒否された場合、管理アウントが招待をキャンセルした場合、または招待状の有効期限が切れた場合は、カウントが返されます。
AWS Organizationsの組織ごとの Firewall Manager ポリシー。	50。リージョンの指定 `Global` および `US East (N. Virginia) Region` は同じリージョンを参照しているため、この制限は 2 つのポリシーを組み合わせた合計に適用されます。
Firewall Manager ポリシーあたりの範囲内の組織単位	20
Firewall Manager ポリシーの範囲内にあるアカウント (個々のアカウントを明示的に含めたり除外したりする場合)。	200
Firewall Manager ポリシーの範囲内にあるアカウント (個々のアカウントを明示的に含めたり除外したりしない場合)。	2,500
Firewall Manager ポリシーごとのリソースを含む、または除外するタグ。	8
アカウントあたりのリソースセットの数。	20
リソースセットあたりのリソースの数。	100

リソース	リージョンあたりのデフォルトのクォータ
Firewall Manager ポリシーあたりのリソースセットの数。	5

AWS WAF ポリシー

リソース	リージョンあたりのデフォルトのクォータ
AWS WAF Firewall Manager 管理者アカウントあたりの ルールグループ。	100
AWS WAF Firewall Manager 管理者アカウントあたりのクラシックルールグループ。	10
AWS WAF ポリシーあたりのルールグループ。	50

共通セキュリティグループポリシー

リソース	リージョンあたりのデフォルトのクォータ。
ポリシーごとのプライマリセキュリティグループ。	3
共有 VPC を含む、アカウントあたりのポリシーごとの範囲内の Amazon VPC インスタンス。	100

コンテンツ監査セキュリティグループポリシー

リソース	リージョンあたりのデフォルトのクォータ
ポリシーごとの監査セキュリティグループ。	1
アプリケーションリストあたりのアプリケーション。	50
すべてのトラフィックを許可するルール用のカスタムマネージドアプリケーションリスト。	1
ポリシールールごとのカスタムマネージドアプリケーションリスト。	1
アカウントあたりのカスタムマネージドアプリケーションリスト。	10
プロトコルリストあたりのプロトコル。	5
ポリシー内における任意の設定のカスタムマネージドプロトコルリスト。	1
アカウントあたりのカスタムマネージドプロトコルリスト。	10

ネットワーク ACL ポリシー

リソース	リージョンあたりのデフォルトのクォータ
最初または最後のルールに使用されるネットワーク ACL ポリシーあたりのインバウンドルールの数。例えば、5 つの最初と最後のインバウンドルール、または 2 つの最初と最後の 3 つのインバウンドルールを持つことができますが、4 つの最初と最後のインバウンドルールを持つことはできません。	5
最初または最後のルールに使用されるネットワーク ACL ポリシーあたりのアウトバウンドルールの数。例えば、5 つの最初と最後のアウトバウンドルール、または 2 つの最初と最後のアウトバウンドルールを持つ	5

リソース	リージョンあたりのデフォルトのクォータ
ことができますが、4 つの最初と最後のアウトバウンドルールを持つことはできません。	

DNS Firewall ポリシー

リソース	リージョンあたりのデフォルトのクォータ
DNS Firewall ポリシーあたりの DNS Firewall ルールグループ。	2

ハードクォータ

に関連する以下のリージョンごとのクォータは変更 AWS Firewall Manager できません。

すべてのポリシータイプ

リソース	リージョンあたりのクォータ
AWS Organizations 組織に含めることができる Firewall Manager 管理者の最大数。1 人の必須デフォルト管理者に加え、最大 9 人の Firewall Manager 管理者を持つことができます。	10

AWS WAF ポリシー

リソース	リージョンあたりのクォータ
AWS WAF ポリシー内のルールグループの合計ウェブ ACL キャパシティーユニット (WCU)。	5,000

AWS WAF クラシックポリシー

リソース	リージョンあたりのクォータ
AWS WAF ポリシーあたりのクラシックルールグループ。	2: お客様が作成した 1 つのルールグループと 1 つの AWS Marketplace ルールグループ。
AWS WAF Firewall Manager Classic ルールグループあたりの AWS WAF Classic ルール。	10

セキュリティグループコンテンツ監査ポリシー

リソース	リージョンあたりのクォータ
ポリシー内の任意の設定に関する Firewall Manager のマネージドアプリケーションリスト。	1
ポリシー内の任意の設定に関する Firewall Manager のマネージドプロトコルリスト。	1

Network Firewall ポリシー

リソース	リージョンあたりのクォータ
1 つのポリシーについて自動的に修正できる VPC の数。	1,000
1 つのポリシーに指定できる IPV4 CIDR の数。	50

監視 AWS WAF、 AWS Firewall Manager、および AWS Shield Advanced

モニタリングは、サービスの信頼性、可用性、パフォーマンスを維持するうえで重要な部分です。

> **Note**
>
> Shield Advanced リソースをモニタリングし、Shield Advanced を使用して発生する可能性のある DDoS イベントを特定する方法については、「AWS Shield」を参照してください。

これらのサービスのモニタリングを開始する前に、次の質問に対する回答を反映したモニタリング計画を作成する必要があります。

- どのような目的でモニタリングしますか?
- どのリソースをモニタリングしますか?
- どのくらいの頻度でこれらのリソースをモニタリングしますか?
- どのモニタリングツールを利用しますか?
- 誰がモニタリングタスクを実行しますか?
- 問題が発生したときに誰が通知を受け取りますか?

次のステップでは、さまざまなタイミングと負荷条件でパフォーマンスを測定することにより、お客様の環境で通常のパフォーマンスのベースラインを確定します。監視中は AWS WAF、Firewall Manager、Shield Advanced、および関連サービスが過去の監視データを保存して、現在のパフォーマンスデータと比較したり、通常のパフォーマンスパターンやパフォーマンスの異常を特定したり、問題に対処する方法を考案したりできるようにします。

というのも AWS WAF、ベースラインを確立するには、少なくとも以下の項目を監視する必要があります。

- 許可されたウェブリクエストの数
- ブロックされたウェブリクエストの数

トピック

- モニタリングツール

- Amazon によるモニタリング CloudWatch
- での AWS CloudTrail API コールのログ記録

モニタリングツール

AWS AWS WAF とを監視するために使用できるさまざまなツールが用意されています AWS Shield Advanced。これらのツールの中には、自動モニタリングを設定できるものもあれば、手動操作を必要とするものもあります。モニタリングタスクをできるだけ自動化することをお勧めします。

自動モニタリングツール

以下の自動監視ツールを使用して AWS WAF 、 AWS Shield Advanced 何か問題が発生した場合は監視したり報告したりできます。

- ウェブ ACL トラフィック概要ダッシュボード — AWS WAF コンソールのウェブ ACL のページに移動して [トラフィック概要] タブを開くと、ウェブ ACL が評価するウェブトラフィックの概要にアクセスできます。

  トラフィック概要ダッシュボードには、 AWS WAF アプリケーションのウェブトラフィックを評価する際に収集される Amazon CloudWatch メトリックスの概要がほぼリアルタイムで表示されます。すべてのウェブトラフィックと、インテリジェント脅威の軽減ルールグループによって評価されたトラフィックの概要を確認できます。

  詳細については、「ウェブ ACL トラフィック概要ダッシュボード」を参照するか、コンソールのダッシュボードに移動してください。
- Amazon CloudWatch Alarms — 指定した期間にわたって 1 つのメトリックスを監視し、複数の期間にわたって特定のしきい値を基準としたメトリクスの値に基づいて 1 つ以上のアクションを実行します。アクションは、Amazon Simple Notification Service (Amazon SNS) のトピックまたは Amazon EC2 Auto Scaling のポリシーに送信される通知です。アラームは、状態が持続的に変化した場合にのみアクションを呼び出します。 CloudWatch アラームは、単に特定の状態にあるからといってアクションを起動するわけではありません。状態が変化し、指定された期間にわたって維持されている必要があります。詳細については、「 CloudFront使用によるアクティビティの監視」を参照してください。 CloudWatch

  > **Note**
  >
  > CloudWatch ではメトリクスとアラームは有効になっていません。 AWS Firewall Manager

で説明されているように、 AWS WAF アドバンスドメトリクスの監視とShield CloudWatch に使用できるだけでなくAmazon によるモニタリング CloudWatch、保護対象リソースのアクティビティの監視にも使用できます。 CloudWatch 詳細については、次を参照してください。

- 『Amazon CloudFront 開発者ガイド』 CloudFront CloudWatchでのアクティビティのモニタリング
- 「API Gateway デベロッパーガイド」の「Logging and monitoring in Amazon API Gateway」(Amazon API Gateway でログ記録とモニタリング)
- CloudWatch 『Elastic Load Balancing ユーザーガイド』のApplication Load Balancer のメトリックス
- 「AWS AppSync デベロッパーガイド」の「Monitoring and Logging」(モニタリングとログ記録)
- 「Amazon Cognito デベロッパーガイド」の「Logging and monitoring in Amazon Cognito」(Amazon Cognito でログ記録とモニタリング)
- Logs にストリーミングされた App Runner CloudWatch ログの表示と、『開発者ガイド』CloudWatchで報告された App Runner サービスメトリクスの表示AWS App Runner

- Amazon CloudWatch Logs — AWS CloudTrail またはその他のソースからのログファイルを監視、保存、アクセスできます。詳細については、「Amazon CloudWatch ログとは」を参照してください。 。
- Amazon CloudWatch Events — AWS サービスを自動化し、システムイベントに自動的に対応します。 AWS CloudWatch サービスからのイベントはほぼリアルタイムでイベントに配信され、作成したルールにイベントが一致したときに実行する自動アクションを指定できます。詳細については、「Amazon CloudWatch イベントとは」を参照してください。
- AWS CloudTrail ログモニタリング — アカウント間でのログファイルの共有、 CloudTrail ログファイルを CloudWatch Logs に送信してリアルタイムで監視し、Java でログ処理アプリケーションを作成し、配信後にログファイルが変更されていないことを検証します。 CloudTrail詳細についてはでの AWS CloudTrail API コールのログ記録、『AWS CloudTrail ユーザーガイド』の「CloudTrail ログファイルの操作」を参照してください。
- AWS Config— AWS リソース同士の関係や過去の構成など、 AWS アカウント内のリソースの設定を確認して、構成や関係が時間の経過とともにどのように変化するかを確認できます。

手動モニタリングツール

AWS Shield Advanced 監視のもう 1 つの重要な点は AWS WAF 、 CloudWatch アラームでカバーされない項目を手動で監視することです。 AWS WAF、Shield Advanced CloudWatch、 AWS

Management Console およびその他のダッシュボードを表示して、 AWS 環境の状態を確認できます。ウェブ ACL とルールのログファイルも確認することをお勧めします。

- たとえば、 AWS WAF ダッシュボードを表示するには:
  - AWS WAF Web ACL ページの [Requests] タブに、リクエストの合計数と作成した各ルールに一致するリクエストのグラフが表示されます。詳細については、「ウェブリクエストのサンプルの表示」を参照してください。
- CloudWatch 次の項目に関するホームページを表示します。
  - 現在のアラームとステータス
  - アラームとリソースのグラフ
  - サービスのヘルスステータス

  さらに、 CloudWatch を使用して次の操作を実行できます。
  - 重視するサービスをモニタリングするためのカスタマイズしたダッシュボードを作成する。
  - メトリクスデータをグラフ化して、問題をトラブルシューティングして、傾向を確認する。
  - AWS すべてのリソースメトリクスを検索して参照できます。
  - 問題があることを通知するアラームを作成および編集する。

Amazon によるモニタリング CloudWatch

Amazon を使用してウェブリクエスト、ウェブ ACL、ルールをモニタリングできます。Amazon は CloudWatch、 AWS WAF AWS Shield Advanced 読み取り可能でほぼリアルタイムのメトリックスからの未加工データを収集して処理します。Amazon CloudWatch の統計情報を使用して、ウェブアプリケーションやサービスのパフォーマンスを把握できます。詳細については、「Amazon CloudWatch CloudWatch ユーザーガイドの内容」を参照してください。

> **Note**
>
> CloudWatch Firewall Manager のメトリックとアラームは有効になっていません。

CloudWatch アラームの状態が変化したときに Amazon SNS メッセージを送信する Amazon アラームを作成できます。アラームは、指定期間にわたって 1 つのメトリクスを監視し、複数期間にわたる指定しきい値との比較結果に基づいて 1 つ以上のアクションを実行します。アクションは、Amazon SNS のトピックまたは Auto Scaling のポリシーに送信される通知です。アラームは、状態が持続的に変化した場合にのみアクションを呼び出します。 CloudWatch アラームが特定の状態

にあるからといってアクションを起動するわけではありません。状態が変化し、指定された期間にわたって維持されている必要があります。

トピック

- メトリクスおよびディメンションの表示
- AWS WAF メトリクスとディメンション
- AWS Shield Advanced 指標
- AWS Firewall Manager 通知

メトリクスおよびディメンションの表示

メトリクスは最初にサービス名前空間ごとにグループ化され、次に各名前空間内のさまざまなディメンションの組み合わせによってグループ化されます。 AWS Firewall Manager メトリクスは記録されません。

- AWS WAF 名前空間は `AWS/WAFV2`
- Shield Advanced の名前空間は `AWS/DDoSProtection` です

> **Note**
>
> AWS WAF 1 分に 1 回メトリクスを報告します。
> Shield Advanced は、イベント中 1 分に 1 回メトリクスをレポートし、イベント以外ではその頻度は少なくなります。

AWS WAF およびのメトリックスを表示するには、次の手順に従います AWS Shield Advanced。

CloudWatch コンソールを使用してメトリクスを表示するには

1. AWS Management Console にサインインし、https://console.aws.amazon.com/cloudwatch/ CloudWatch のコンソールを開きます。
2. 必要に応じて、 AWS リージョンをリソースがあるリージョンに変更してください。には CloudFront、米国東部 (バージニア北部) リージョンを選択します。
3. ナビゲーションペインの [Metrics] (メトリクス) で、[All metrics] (すべてのメトリクス) を選択し、[Browse] (参照) タブでサービスを検索します。

AWS CLI を使用してメトリクスを表示するには

- AWS/WAFV2 の場合、コマンドプロンプトで次のコマンドを使用します。

```
aws cloudwatch list-metrics --namespace "AWS/WAFV2"
```

Shield Advanced の場合、コマンドプロンプトで次のコマンドを使用します。

```
aws cloudwatch list-metrics --namespace "AWS/DDoSProtection"
```

AWS WAF メトリクスとディメンション

AWS WAF は、メトリクスを 1 分に 1 回レポートします。 は、 AWS/WAFV2 名前空間にメトリクスとディメンション AWS WAF を提供します。

コンソールのウェブ ACL のトラフィック概要タブで AWS WAF 、 AWS WAF メトリクスの概要情報を確認できます。詳細については、コンソールに移動するか、「」を参照してくださいウェブ ACL トラフィック概要ダッシュボード。

ウェブ ACLs 、ルール、ルールグループ、およびラベルの次のメトリクスを確認できます。

- ルール – メトリクスはルールアクションによってグループ化されます。例えば、 Count モードでルールをテストすると、その一致はウェブ ACL のCountメトリクスとして一覧表示されます。
- ルールグループ – ルールグループのメトリクスは、ルールグループのメトリクスの下に一覧表示されます。
- 別のアカウントが所有するルールグループ – ルールグループのメトリクスは、通常、ルールグループの所有者にのみ表示されます。ただし、ルールのルールアクションを上書きすると、そのルールのメトリクスがウェブ ACL メトリクスの下に一覧表示されます。さらに、ルールグループによって追加されたラベルは、ウェブ ACL メトリクスに一覧表示されます。

  このカテゴリのルールグループはAWS のマネージドルール AWS WAF、別のアカウントによって共有される AWS Marketplace マネージドルールグループ、他のサービスによって提供されるルールグループ、、および ルールグループです。
- ラベル - 評価中にウェブリクエストに追加されたラベルは、ウェブ ACL ラベルメトリクスに一覧表示されます。ルールとルールグループによって追加されたか、別のアカウントが所有するルール

グループ内のルールによって追加されたかに関係なく、すべてのラベルのメトリクスにアクセスできます。

トピック

ウェブ ACL、ルールグループ、およびルールのメトリクスとディメンション

ウェブ ACL、ルールグループ、およびルールのメトリクス

メトリクス	説明
`AllowedRequests`	許可されたウェブリクエストの数。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum
`BlockedRequests`	ブロックされたウェブリクエストの数。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum
`CountedRequests`	カウントされたウェブリクエストの数。 レポート条件: ゼロ以外の値がある。 カウントされたウェブリクエストは、少なくとも 1 つのルールに一致するリクエストです。リクエストカウントは、通常、テストに使用されます。 有効な統計: Sum
`CaptchaRequests`	CAPTCHA コントロールが適用されたウェブリクエストの数。 レポート条件: ゼロ以外の値がある。

メトリクス	説明
	CAPTCHA ウェブリクエストは、CAPTCHA アクション設定を持つルールに一致するリクエストです。このメトリクスは、有効な CAPTCHA トークンがあるか否かにかかわらず、一致するすべてのリクエストを記録します。 有効な統計: Sum
RequestsWithValidCaptchaToken	CAPTCHA コントロールが適用され、有効な CAPTCHA トークンを持つウェブリクエストの数。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum
CaptchasAttempted	CAPTCHA パズルチャレンジに応答してエンドユーザーから送信されたソリューションの数。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum
CaptchasSolved	パズルを正解し、送信された CAPTCHA パズルソリューションの数。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum

メトリクス	説明
ChallengeRequests	チャレンジコントロールが適用されたウェブリクエストの数。 レポート条件: ゼロ以外の値がある。 チャレンジウェブリクエストは、Challenge アクション設定を持つルールに一致するリクエストです。このメトリクスは、有効なチャレンジトークンがあるか否かにかかわらず、一致するすべてのリクエストを記録します。 有効な統計: Sum
RequestsWithValidChallengeToken	チャレンジコントロールが適用され、有効なチャレンジトークンを持つウェブリクエストの数。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum
PassedRequests	渡されたリクエストの数。これは、ルールグループのルールのいずれとも一致せず、ルールグループ評価を通過するリクエストについてのみ使用されます。 レポート条件: ゼロ以外の値がある。 渡されたリクエストは、ルールグループのいずれのルールにも一致しないリクエストです。 有効な統計: Sum

ウェブ ACL、ルールグループ、およびルールのディメンション

ディメンション	説明
Region	Amazon CloudFront ディストリビューションを除くすべての保護されたリソースタイプに必要です。
Rule	次のいずれかです。 • Rule のメトリクス名。 • すべて。これは、WebACL または RuleGroup 内のすべてのルールを表します。 • Default_Action (WebACL ディメンションと組み合わせた場合のみ)。これは、ウェブ ACL のルールのアクションによって評価が終了されなかったリクエストに割り当てられたアクションを表します。
RuleGroup	RuleGroup のメトリクス名。
WebACL	WebACL のメトリクス名。
Country	リクエストの送信元の国 これは、国際標準化機構 (ISO) 3166 規格の 2 文字の名称です。たとえば、米国は US、ウクライナは UA です。 リクエストに X-Forwarded-For ヘッダーがある場合、 AWS WAF はそのヘッダーを使用してこの設定を決定します。それ以外の場合は、 AWS WAF はクライアント IP の国を使用します。この決定は、MaxMind GeoIP データベースを使用して IPs AWS WAF の場所を決定するためにルールで使用するロジックとは無関係です。
Attack	ウェブ ACL で使用するルールとルールグループに基づいて、リクエストで が AWS WAF 識別した攻撃のタイプ。

ディメンション	説明
	ルールとベースライン AWS マネージドルールグループのルールは、攻撃タイプを識別できます。たとえば、クロスサイトスクリプティング (XSS) ルール一致は XSS 攻撃タイプを識別し、レートベースのルールはボリューム攻撃タイプを識別します。攻撃タイプは、通常、ウェブリクエストの評価を終了したルールのタイプを示します。
`Device`	リクエストを送信したクライアントのデバイスタイプは、ウェブリクエストの `user-agent` ヘッダーから取得されます。
`ManagedRuleGroup`	`ManagedRuleGroup` のメトリクス名。
`ManagedRuleGroupRule`	一致`ManagedRuleGroup` した 内のルール。

ラベルメトリクスとディメンション

ルールおよびウェブ ACL で使用するマネージドルールグループによる評価中に、リクエストに追加されたラベルのメトリクス。詳細については、「ウェブリクエストのラベル」を参照してください。

1 つのウェブリクエストについて、 は最大 100 個のラベルのメトリクス AWS WAF を保存します。ウェブ ACL 評価では、100 個を超えるラベルを適用したり、100 個を超えるラベルと照合したりできますが、メトリクスには最初の 100 個のみが反映されます。

ラベルメトリクス

メトリクス	説明
`AllowedRequests`	アクション設定 Allow を適用したウェブリクエストのラベルの数。ラベルは、ウェブリクエストの評価中のどの時点でも追加できます。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum

メトリクス	説明
BlockedRequests	アクション設定 Block を適用したウェブリクエストのラベルの数。ラベルは、ウェブリクエストの評価中のどの時点でも追加できます。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum
CountedRequests	Count アクション設定を持つルールグループルールによってウェブリクエストに追加されるラベルの数。 このメトリクスは、ルールグループ内のルールについて、ルールグループの所有者のみが使用できます。その他のケースでは、リクエストに適用された終了アクション (Allow または Block など) にカウントラベルメトリクスがまとめられます。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum
CaptchaRequests	終了 CAPTCHA アクションが適用されたウェブリクエストのラベルの数。ラベルは、ウェブリクエストの評価中のどの時点でも追加できます。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum
ChallengeRequests	終了 Challenge アクションが適用されたウェブリクエストのラベルの数。ラベルは、ウェブリクエストの評価中のどの時点でも追加できます。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum

メトリクス	説明
AllowRuleMatch	Allow アクションを使用して、関連付けられたラベルを生成し、リクエスト評価を終了させた、一致したルールの数。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum
BlockRuleMatch	Block アクションを使用して、関連付けられたラベルを生成し、リクエスト評価を終了させた、一致したルールの数。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum
CountRuleMatch	関連付けられたラベルを生成し、Countアクションを適用した、一致したルールの数。 複数のルールが同じラベルとアクションで設定されている場合、1 つのリクエストでこのメトリクスの複数のインスタンスが発生する可能性があります。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum
CaptchaRuleMatch	CAPTCHA アクションを使用して、関連付けられたラベルを生成し、リクエスト評価を終了させた、一致したルールの数。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum

メトリクス	説明
ChallengeRuleMatch	Challenge アクションを使用して、関連付けられたラベルを生成し、リクエスト評価を終了させた、一致したルールの数。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum
CaptchaRuleMatchWithValidToken	関連付けられたラベルを生成し、非終了CAPTCHAアクションを適用した、一致したルールの数。 複数のルールが同じラベルとアクションで設定されている場合、1 つのリクエストでこのメトリクスの複数のインスタンスが発生する可能性があります。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum
ChallengeRuleMatchWithValidToken	関連付けられたラベルを生成し、非終了Challengeアクションを適用した、一致したルールの数。 複数のルールが同じラベルとアクションで設定されている場合、1 つのリクエストでこのメトリクスの複数のインスタンスが発生する可能性があります。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum

ラベルディメンション

ディメンション	説明
Region	Amazon CloudFront ディストリビューションを除くすべての保護されたリソースタイプに必要です。
WebACL	WebACL のメトリクス名。

ディメンション	説明
RuleGroup	RuleGroup のメトリクス名。メトリクス CountedRequests に使用されます。
LabelNamespace	リクエストに追加されたラベルの名前空間プレフィックス。
Label	リクエストに追加されたラベルの名前。
Context	ラベル追加のコンテキストとして機能するマネージドルールグループ。例えば、 などのトークン管理ラベルのコンテキストawswaf:managed:token:accepted は、Bot Control や ATP AWS WAF マネージドルールグループなど、リクエストでトークン管理を使用するマネージドルールグループです。このディメンションはすべてのラベルに適用されるわけではありません。

無料のボット可視性メトリクスおよびディメンション

ウェブ ACL で Bot Control を使用しない場合、 は Bot Control マネージドルールグループをウェブリクエストのサンプリング AWS WAF に適用します。追加料金はかかりません。これにより、保護対象リソースに流入するボットトラフィックを把握できます。Bot Control については、「AWS WAF Bot Control ルールグループ」を参照してください。

無料のボット可視性メトリクス

メトリクス	説明
SampleAllowedRequest	Allow アクションを持つサンプリングされたリクエストの数。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum
SampleBlockedRequest	Block アクションを持つサンプリングされたリクエストの数。

メトリクス	説明
	レポート条件: ゼロ以外の値がある。 有効な統計: Sum
SampleCaptchaRequest	CAPTCHA アクションを持つサンプリングされたリクエストの数。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum
SampleChallengeRequest	Challenge アクションを持つサンプリングされたリクエストの数。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum
SampleCountRequest	Count アクションを持つサンプリングされたリクエストの数。 レポート条件: ゼロ以外の値がある。 有効な統計: Sum

無料のボット可視性ディメンション

ディメンション	説明
Region	Amazon CloudFront ディストリビューションを除くすべての保護されたリソースタイプに必要です。
WebACL	WebACL のメトリクス名。
BotCategory	ウェブリクエストのラベルに基づく、検出されたボットカテゴリ名。
VerificationStatus	ウェブリクエストのラベルに基づく、検出されたボット検証ステータス名。

ディメンション	説明
Signal	ウェブリクエストのラベルに基づく、検出されたボットシグナル名。

AWS Shield Advanced 指標

Shield Advancedは、保護対象のすべてのリソースについて、 CloudWatch Amazonの検出、緩和、および上位貢献者メトリックスを公開しています。これらのメトリックスにより、 CloudWatch リソース用のダッシュボードとアラームを作成および設定できるようになるため、リソースの監視能力が向上します。

Shield アドバンスコンソールには、記録される多くのメトリクスの概要が表示されます。詳細については、DDoS イベントの可視性 を参照してください。

アプリケーション層を保護するためにアプリケーション層の DDoS の自動軽減を有効にすると、

メトリクスレポートの場所

Shield Advanced は、次のように、米国東部 (バージニア北部) (us-east-1) リージョンのメトリクスをレポートします。

- グローバルサービスのAmazon CloudFront とAmazon ルート53。
- 保護グループ 保護グループについては、「AWS Shield Advanced 保護グループ」を参照してください。

他のリソースタイプについては、Shield Advanced がリソースのリージョンのメトリクスをレポートします。

メトリックスレポートのタイミング

Shield Advancedは、イベントが発生していないときよりも、DDoSイベントの際に、 CloudWatch AWS リソースのメトリクスをAmazonにレポートする頻度が高くなります。Shield Advanced は、イベント中は 1 分ごとに、およびイベント終了直後に 1 回、メトリクスをレポートします。

イベントが発生していない間、Shield Advanced は 1 日に 1 回、リソースに割り当てられた時間にメトリクスを報告します。この定期レポートでは、メトリクスがアクティブな状態に保たれ、CloudWatch カスタムアラームやダッシュボードで使用できるようになります。

アラームに関する推奨事項

注意が必要な状況を通知するアラームを作成することをお勧めします。まず、保護対象リソースごとにアラームを作成して、DDoSDetected検出メトリックがゼロ以外になったときに報告するという方法もあります。このメトリクスのゼロ以外の値は、DDoS 攻撃が進行中であることを必ずしも意味するわけではありませんが、メトリクスがこの状態にある場合は、リソースのステータスを詳しく調べることをお勧めします。

リクエストフラッドについては、アプリケーションのヘルスやウェブリクエストの量などの要素も考慮する複合チェックのアラームを作成することをお勧めします。さまざまな攻撃ベクトルディメンションのトラフィック量について報告する他の 3 つのメトリクスでアラームを設定できます。アプリケーションの容量を考慮し、トラフィックがアプリケーションの制限に近づいたときにアラームを発信することで、望ましくないノイズを過剰に発生させることなく、必要に応じて通知する一連のルールを作成できます。

トピック

- 検出メトリクス
- 緩和のメトリクス
- 上位の寄稿者のメトリクス

検出メトリクス

Shield アドバンスドは、AWS/DDoSProtection名前空間のメトリクスとディメンションを提供します。

検出メトリクス

メトリクス	説明
DDoSDetected	特定の Amazon リソースネーム (ARN) に対して DDoS イベントが進行中かどうかを示します。 このメトリクスは、イベント中はゼロ以外の値を持ちます。
DDoSAttackBitsPerSecond	特定の Amazon リソースネーム (ARN) の DDoS イベント中に認められたビット数。このメトリクスは、ネットワークおよびトランス

メトリクス	説明
	ポートレイヤー (レイヤー 3 およびレイヤー 4) の DDoS イベントにのみ使用できます。 このメトリクスは、イベント中はゼロ以外の値を持ちます。 単位: ビット
DDoSAttackPacketsPerSecond	特定の Amazon リソースネーム (ARN) の DDoS イベント中に認められたパケット数。このメトリクスは、ネットワークおよびトランスポートレイヤー (レイヤー 3 およびレイヤー 4) の DDoS イベントにのみ使用できます。 このメトリクスは、イベント中はゼロ以外の値を持ちます。 単位: パケット
DDoSAttackRequestsPerSecond	特定の Amazon リソースネーム (ARN) の DDoS イベント中に認められたリクエスト数。このメトリクスは、レイヤー 7 の DDoS イベントのみで使用できます。メトリクスは、特に重要なレイヤー 7 イベントのみについて報告されます。 このメトリクスは、イベント中はゼロ以外の値を持ちます。 単位: リクエスト

Shield Advanced は、他のディメンションなしで DDoSDetected メトリクスを投稿します。残りの検出メトリクスには、次のリストから、攻撃のタイプに対応する AttackVector ディメンションが含まれます。

- ACKFlood
- ChargenReflection

- DNSReflection
- GenericUDPReflection
- MemcachedReflection
- MSSQLReflection
- NetBIOSReflection
- NTPReflection
- PortMapper
- RequestFlood
- RIPReflection
- SNMPReflection
- SSDPReflection
- SYNFlood
- UDPFragment
- UDPTraffic
- UDPReflection

緩和のメトリクス

Shield アドバンスドは、AWS/DDoSProtection名前空間にメトリクスとディメンションを提供します。

緩和のメトリクス

メトリクス	説明
VolumePacketsPerSecond	検出されたイベントに対応してデプロイされた緩和策によってドロップされたか、または渡された 1 秒あたりのパケット数。 単位: パケット

緩和のディメンション

ディメンション	説明
ResourceArn	Amazon リソースネーム (ARN)
MitigationAction	適用された緩和策の結果。想定される値は、Pass または Drop です。

上位の寄稿者のメトリクス

Shield アドバンスドは、AWS/DDoSProtectionネームスペースにメトリクスを提供します。

上位の寄稿者のメトリクス

メトリクス	説明
VolumePacketsPerSecond	上位のコントリビューターの 1 秒あたりのパケット数。 単位: パケット
VolumeBitsPerSecond	上位のコントリビューターの 1 秒あたりのビット数。 単位: ビット

Shield Advanced は、イベントの寄稿者を特徴づけるディメンションの組み合わせによって、上位の寄稿者のメトリクスを投稿します 上位の寄稿者のあらゆるメトリクスについて、次のいずれかのディメンションの組み合わせを使用できます。

- ResourceArn, Protocol
- ResourceArn, Protocol, SourcePort
- ResourceArn, Protocol, DestinationPort
- ResourceArn, Protocol, SourceIp
- ResourceArn, Protocol, SourceAsn
- ResourceArn, TcpFlags

上位の寄稿者のディメンション

ディメンション	説明
ResourceArn	Amazon リソースネーム (ARN)。
Protocol	TCP または UDP のいずれかの IP プロトコル名。
SourcePort	ソース TCP または UDP ポート。
DestinationPort	宛先 TCP または UDP ポート。
SourceIp	送信元 IP アドレス。
SourceAsn	ソース Autonomous System number (ASN)。
TcpFlags	ダッシュ - で区切られた TCP パケットに存在するフラグの組み合わせ。モニタリング対象フラグは、ACK、FIN、RST、SYN です。このディメンション値は、常にアルファベット順にソートされて表示されます。例: ACK-FIN-RST-SYN 、ACK-SYN、FIN-RST。

AWS Firewall Manager 通知

AWS Firewall Manager メトリックスを記録しないため、Firewall Manager 専用の Amazon CloudWatch アラームを作成することはできません。ただし、可能性のある攻撃のアラートを送信する Amazon SNS 通知を設定することができます。Firewall Manager で Amazon SNS 通知を作成するには、「ステップ 4: Amazon SNS CloudWatch 通知とアマゾンアラームを設定する」を参照してください。

での AWS CloudTrail API コールのログ記録

AWS WAF AWS Shield Advanced、、ユーザー、ロール AWS CloudTrail、 AWS Firewall Manager AWS またはサービスによって実行されたアクションの記録を提供するサービスと統合されています。 CloudTrail は、Shield Advanced または Firewall Manager コンソールからの呼び出し、および AWS WAF、Shield Advanced または Firewall Manager API へのコード呼び出しを含む AWS WAF、これらのサービスの API 呼び出しのサブセットをイベントとしてキャプチャします。証跡を作成すると、Shield Advanced、Firewall Manager CloudTrail のイベントなど、Amazon S3 バケットへの

イベントの継続的な配信を有効にできます。 AWS WAFトレイルを設定しなくても、 CloudTrail コンソールの [イベント履歴] に最新のイベントが表示されます。によって収集された情報を使用して CloudTrail、これらのサービスに対して行われた要求、要求の送信元 IP アドレス、要求の実行者、実行日時、その他の詳細情報を確認できます。

設定方法や有効化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

CloudTrail AWS アカウント アカウントを作成すると有効になります。サポートされているイベントアクティビティが Shield Advanced または Firewall Manager AWS WAFで発生すると、 CloudTrail AWS そのアクティビティは他のサービスイベントとともにイベント履歴に記録されます。で最近のイベントを表示、検索、ダウンロードできます AWS アカウント。詳細については、「CloudTrail イベント履歴によるイベントの表示」を参照してください。

Shield Advanced AWS アカウント、Firewall Manager のイベントなど AWS WAF、社内でのイベントの継続的な記録については、トレイルを作成してください。トレイルを使用すると CloudTrail 、Amazon S3 バケットにログファイルを配信できます。デフォルトでは、コンソールで証跡を作成すると、証跡がすべての リージョンに適用されます。トレイルは、 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、 AWS CloudTrail ログに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サービスを設定できます。詳細については、次を参照してください:

- 追跡の作成のための概要
- CloudTrail サポート対象のサービスとインテグレーション
- の Amazon SNS 通知の設定 CloudTrail
- CloudTrail 複数のリージョンからのログファイルの受信、 CloudTrail複数のアカウントからのログファイルの受信

AWS WAF の情報 AWS CloudTrail

すべての AWS WAF アクションは によってログに記録 AWS CloudTrail され、 AWS WAF API リファレンス に記載されています。例えば、、`ListWebACL`、 を呼び出すと`UpdateWebACL`、CloudTrail ログファイルにエントリ`DeleteWebACL`が生成されます。

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、ルートユーザーの認証情報で行われたかどうか

- リクエストが、ロールとフェデレーティッドユーザーのどちらの一時的なセキュリティ認証情報を使用して送信されたか
- 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、CloudTrailuserIdentity Element」を参照してください。

例: AWS WAF ログファイルエントリ

証跡は、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。 AWS CloudTrail ログファイルには 1 つ以上のログエントリが含まれます。イベントは任意ソースからの単一リクエストを表し、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の情報を含みます。 CloudTrail ログファイルはパブリック API コール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。

AWS WAF ウェブ ACL オペレーションの CloudTrail ログエントリの例を次に示します。

の例: の CloudTrail ログエントリ `CreateWebACL`


```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "principalId",
    "arn": "arn:aws:sts::112233445566:assumed-role/Admin",
    "accountId": "112233445566",
    "accessKeyId": "accessKeyId",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "principalId",
        "arn": "arn:aws:iam::112233445566:role/Admin",
        "accountId": "112233445566",
        "userName": "Admin"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2019-11-06T03:43:07Z"
      }
    }
  },
  "eventTime": "2019-11-06T03:44:21Z",
```

```
  "eventSource": "wafv2.amazonaws.com",
  "eventName": "CreateWebACL",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "10.0.0.1",
  "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/78.0.3904.87 Safari/537.36",
  "requestParameters": {
    "name": "foo",
    "scope": "CLOUDFRONT",
    "defaultAction": {
      "block": {}
    },
    "description": "foo",
    "rules": [
      {
        "name": "foo",
        "priority": 1,
        "statement": {
          "geoMatchStatement": {
            "countryCodes": [
              "AF",
              "AF"
            ]
          }
        },
        "action": {
          "block": {}
        },
        "visibilityConfig": {
          "sampledRequestsEnabled": true,
          "cloudWatchMetricsEnabled": true,
          "metricName": "foo"
        }
      }
    ],
    "visibilityConfig": {
      "sampledRequestsEnabled": true,
      "cloudWatchMetricsEnabled": true,
      "metricName": "foo"
    }
  },
  "responseElements": {
    "summary": {
      "name": "foo",
```

```
      "id": "ebbcb976-8d59-4d20-8ca8-4ab2f6b7c07b",
      "description": "foo",
      "lockToken": "67551e73-49d8-4363-be48-244deea72ea9",
      "aRN": "arn:aws:wafv2:us-east-1:112233445566:global/webacl/foo/
ebbcb976-8d59-4d20-8ca8-4ab2f6b7c07b"
    }
  },
  "requestID": "c51521ba-3911-45ca-ba77-43aba50471ca",
  "eventID": "afd1a60a-7d84-417f-bc9c-7116cf029065",
  "eventType": "AwsApiCall",
  "apiVersion": "2019-04-23",
  "recipientAccountId": "112233445566"
}
```

の例: の CloudTrail ログエントリ GetWebACL

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AssumedRole",
    "arn": "arn:aws:sts::112233445566:assumed-role/Admin/admin",
    "accountId": "112233445566",
    "accessKeyId": "accessKeyId",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AssumedRole",
        "arn": "arn:aws:iam::112233445566:role/Admin",
        "accountId": "112233445566",
        "userName": "Admin"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2019-11-06T19:17:20Z"
      }
    }
  },
  "eventTime": "2019-11-06T19:18:28Z",
  "eventSource": "wafv2.amazonaws.com",
  "eventName": "GetWebACL",
  "awsRegion": "us-east-1",
```

```
  "sourceIPAddress": "10.0.0.1",
  "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) AppleWebKit/537.36
 (KHTML, like Gecko) Chrome/78.0.3904.87 Safari/537.36",
  "requestParameters": {
    "name": "foo",
    "scope": "CLOUDFRONT",
    "id": "webacl"
  },
  "responseElements": null,
  "requestID": "f2db4884-4eeb-490c-afe7-67cbb494ce3b",
  "eventID": "7d563cd6-4123-4082-8880-c2d1fda4d90b",
  "readOnly": true,
  "eventType": "AwsApiCall",
  "apiVersion": "2019-04-23",
  "recipientAccountId": "112233445566"
}
```


の例: の CloudTrail ログエントリ UpdateWebACL


```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "principalId",
    "arn": "arn:aws:sts::112233445566:assumed-role/Admin",
    "accountId": "112233445566",
    "accessKeyId": "accessKeyId",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "principalId",
        "arn": "arn:aws:iam::112233445566:role/Admin",
        "accountId": "112233445566",
        "userName": "Admin"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2019-11-06T19:17:20Z"
      }
    }
  },
  "eventTime": "2019-11-06T19:20:56Z",
```

```
  "eventSource": "wafv2.amazonaws.com",
  "eventName": "UpdateWebACL",
  "awsRegion": ""us-east-1,
  "sourceIPAddress": "10.0.0.1",
  "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/78.0.3904.87 Safari/537.36",
  "requestParameters": {
    "name": "foo",
    "scope": "CLOUDFRONT",
    "id": "ebbcb976-8d59-4d20-8ca8-4ab2f6b7c07b",
    "defaultAction": {
      "block": {}
    },
    "description": "foo",
    "rules": [
      {
        "name": "foo",
        "priority": 1,
        "statement": {
          "geoMatchStatement": {
            "countryCodes": [
              "AF"
            ]
          }
        },
        "action": {
          "block": {}
        },
        "visibilityConfig": {
          "sampledRequestsEnabled": true,
          "cloudWatchMetricsEnabled": true,
          "metricName": "foo"
        }
      }
    ],
    "visibilityConfig": {
      "sampledRequestsEnabled": true,
      "cloudWatchMetricsEnabled": true,
      "metricName": "foo"
    },
    "lockToken": "67551e73-49d8-4363-be48-244deea72ea9"
  },
  "responseElements": {
    "nextLockToken": "a6b54c01-7975-4e6d-b7d0-2653cb6e231d"
```

```
  },
  "requestID": "41c96e12-9790-46ab-b145-a230f358f2c2",
  "eventID": "517a10e6-4ca9-4828-af90-a5cff9756594",
  "eventType": "AwsApiCall",
  "apiVersion": "2019-04-23",
  "recipientAccountId": "112233445566"
}
```

の例: の CloudTrail ログエントリ DeleteWebACL

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "principalId",
    "arn": "arn:aws:sts::112233445566:assumed-role/Admin/session-name",
    "accountId": "112233445566",
    "accessKeyId": "accessKeyId",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "principalId",
        "arn": "arn:aws:iam::112233445566:role/Admin",
        "accountId": "112233445566",
        "userName": "Admin"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2019-11-06T19:17:20Z"
      }
    }
  },
  "eventTime": "2019-11-06T19:25:17Z",
  "eventSource": "wafv2.amazonaws.com",
  "eventName": "DeleteWebACL",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "10.0.0.1",
  "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) AppleWebKit/537.36
 (KHTML, like Gecko) Chrome/78.0.3904.87 Safari/537.36",
  "requestParameters": {
    "name": "foo",
    "scope": "CLOUDFRONT",
```

```
    "id": "ebbcb976-8d59-4d20-8ca8-4ab2f6b7c07b",
    "lockToken": "a6b54c01-7975-4e6d-b7d0-2653cb6e231d"
  },
  "responseElements": null,
  "requestID": "71703f89-e139-440c-96d4-9c77f4cd7565",
  "eventID": "2f976624-b6a5-4a09-a8d0-aa3e9f4e5187",
  "eventType": "AwsApiCall",
  "apiVersion": "2019-04-23",
  "recipientAccountId": "112233445566"
}
```

例: AWS WAF 従来のログファイルエントリ

AWS WAF Classic は の以前のバージョンです AWS WAF。詳細については、「AWS WAF クラシック」を参照してください。

ログエントリは、CreateRule、GetRule、UpdateRule、および DeleteRule の各オペレーションを示しています。

```
{
  "Records": [
   {
      "eventVersion": "1.03",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDAIEP4IT4TPDEXAMPLE",
        "arn": "arn:aws:iam::777777777777:user/nate",
        "accountId": "777777777777",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "nate"
      },
      "eventTime": "2016-04-25T21:35:14Z",
      "eventSource": "waf.amazonaws.com",
      "eventName": "CreateRule",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "AWS Internal",
      "userAgent": "console.amazonaws.com",
      "requestParameters": {
        "name": "0923ab32-7229-49f0-a0e3-66c81example",
        "changeToken": "l9434322-8685-4ed2-9c5b-9410bexample",
        "metricName": "0923ab32722949f0a0e366c81example"
      },
      "responseElements": {
```

```
        "rule": {
          "metricName": "0923ab32722949f0a0e366c81example",
          "ruleId": "12132e64-6750-4725-b714-e7544example",
          "predicates": [

          ],
          "name": "0923ab32-7229-49f0-a0e3-66c81example"
        },
        "changeToken": "l9434322-8685-4ed2-9c5b-9410bexample"
      },
      "requestID": "4e6b66f9-d548-11e3-a8a9-73e33example",
      "eventID": "923f4321-d378-4619-9b72-4605bexample",
      "eventType": "AwsApiCall",
      "apiVersion": "2015-08-24",
      "recipientAccountId": "777777777777"
    },
    {
      "eventVersion": "1.03",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "AIDAIEP4IT4TPDEXAMPLE",
        "arn": "arn:aws:iam::777777777777:user/nate",
        "accountId": "777777777777",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "nate"
      },
      "eventTime": "2016-04-25T21:35:22Z",
      "eventSource": "waf.amazonaws.com",
      "eventName": "GetRule",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "AWS Internal",
      "userAgent": "console.amazonaws.com",
      "requestParameters": {
        "ruleId": "723c2943-82dc-4bc1-a29b-c7d73example"
      },
      "responseElements": null,
      "requestID": "8e4f3211-d548-11e3-a8a9-73e33example",
      "eventID": "an236542-d1f9-4639-bb3d-8d2bbexample",
      "eventType": "AwsApiCall",
      "apiVersion": "2015-08-24",
      "recipientAccountId": "777777777777"
    },
    {
      "eventVersion": "1.03",
```

```
    "userIdentity": {
      "type": "IAMUser",
      "principalId": "AIDAIEP4IT4TPDEXAMPLE",
      "arn": "arn:aws:iam::777777777777:user/nate",
      "accountId": "777777777777",
      "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
      "userName": "nate"
    },
    "eventTime": "2016-04-25T21:35:13Z",
    "eventSource": "waf.amazonaws.com",
    "eventName": "UpdateRule",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "console.amazonaws.com",
    "requestParameters": {
      "ruleId": "7237b123-7903-4d9e-8176-9d71dexample",
      "changeToken": "32343a11-35e2-4dab-81d8-6d408example",
      "updates": [
        {
          "predicate": {
            "type": "SizeConstraint",
            "dataId": "9239c032-bbbe-4b80-909b-782c0example",
            "negated": false
          },
          "action": "INSERT"
        }
      ]
    },
    "responseElements": {
      "changeToken": "32343a11-35e2-4dab-81d8-6d408example"
    },
    "requestID": "11918283-0b2d-11e6-9ccc-f9921example",
    "eventID": "00032abc-5bce-4237-a8ee-5f1a9example",
    "eventType": "AwsApiCall",
    "apiVersion": "2015-08-24",
    "recipientAccountId": "777777777777"
  },
  {
    "eventVersion": "1.03",
    "userIdentity": {
      "type": "IAMUser",
      "principalId": "AIDAIEP4IT4TPDEXAMPLE",
      "arn": "arn:aws:iam::777777777777:user/nate",
      "accountId": "777777777777",
```

```
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "userName": "nate"
      },
      "eventTime": "2016-04-25T21:35:28Z",
      "eventSource": "waf.amazonaws.com",
      "eventName": "DeleteRule",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "AWS Internal",
      "userAgent": "console.amazonaws.com",
      "requestParameters": {
        "changeToken": "fd232003-62de-4ea3-853d-52932example",
        "ruleId": "3e3e2d11-fd8b-4333-8b03-1da95example"
      },
      "responseElements": {
        "changeToken": "fd232003-62de-4ea3-853d-52932example"
      },
      "requestID": "b23458a1-0b2d-11e6-9ccc-f9928example",
      "eventID": "a3236565-1a1a-4475-978e-81c12example",
      "eventType": "AwsApiCall",
      "apiVersion": "2015-08-24",
      "recipientAccountId": "777777777777"
    }
  ]
}
```


AWS Shield Advanced 内の情報 CloudTrail

AWS Shield Advanced CloudTrail 次のアクションをイベントとしてログファイルに記録することをサポートします。

- ListAttacks
- DescribeAttack
- CreateProtection
- DescribeProtection
- DeleteProtection
- ListProtections
- CreateSubscription
- DescribeSubscription
- GetSubscriptionState

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、以下を判別するために役立ちます。

- リクエストが、ルートユーザーの認証情報で行われたかどうか
- リクエストがロールまたはフェデレーションユーザーのテンポラリなセキュリティ認証情報を使用して行われたかどうか。
- AWS リクエストが別のサービスによってなされたかどうか。

詳細については、「CloudTrail userIdentity エレメント」を参照してください。

例: Shield Advanced ログファイルエントリ

トレイルは、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 ログファイルには 1 つ以上のログエントリが含まれます。イベントはあらゆるソース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に関する情報、アクションの日時、リクエストパラメータなどが含まれます。 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、特定の順序で表示されることはありません。

次の例は、 CloudTrail DeleteProtectionListProtectionsおよびアクションを示すログエントリを示しています。


```
[
 {
   "eventVersion": "1.05",
   "userIdentity": {
     "type": "IAMUser",
     "principalId": "123456789098765432123 1",
     "arn": "arn:aws:iam::123456789012:user/SampleUser",
     "accountId": "123456789012",
     "accessKeyId": "1AFGDT647FHU83JHFI81H",
     "userName": "SampleUser"
   },
   "eventTime": "2018-01-10T21:31:14Z",
   "eventSource": "shield.amazonaws.com",
   "eventName": "DeleteProtection",
   "awsRegion": "us-east-1",
   "sourceIPAddress": "AWS Internal",
   "userAgent": "aws-cli/1.14.10 Python/3.6.4 Darwin/16.7.0 botocore/1.8.14",
```

```
    "requestParameters": {
      "protectionId": "12345678-5104-46eb-bd03-agh4j8rh3b6n"
    },
    "responseElements": null,
    "requestID": "95bc0042-f64d-11e7-abd1-1babdc7aa857",
    "eventID": "85263bf4-17h4-43bb-b405-fh84jhd8urhg",
    "eventType": "AwsApiCall",
    "apiVersion": "AWSShield_20160616",
    "recipientAccountId": "123456789012"
  },
  {
    "eventVersion": "1.05",
    "userIdentity": {
      "type": "IAMUser",
      "principalId": "123456789098765432123",
      "arn": "arn:aws:iam::123456789012:user/SampleUser",
      "accountId": "123456789012",
      "accessKeyId": "1AFGDT647FHU83JHFI81H",
      "userName": "SampleUser"
    },
    "eventTime": "2018-01-10T21:30:03Z",
    "eventSource": "shield.amazonaws.com",
    "eventName": "ListProtections",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "aws-cli/1.14.10 Python/3.6.4 Darwin/16.7.0 botocore/1.8.14",
    "requestParameters": null,
    "responseElements": null,
    "requestID": "6accca40-f64d-11e7-abd1-1bjfi8urhj47",
    "eventID": "ac0570bd-8dbc-41ac-a2c2-987j90j3h78f",
    "eventType": "AwsApiCall",
    "apiVersion": "AWSShield_20160616",
    "recipientAccountId": "123456789012"
  }
]
```


AWS Firewall Manager 内の情報 CloudTrail

AWS Firewall Manager CloudTrail 次のアクションをイベントとしてログファイルに記録することをサポートします。

- AssociateAdminAccount
- DeleteNotificationChannel

- DeletePolicy
- DisassociateAdminAccount
- PutNotificationChannel
- PutPolicy
- GetAdminAccount
- GetComplianceDetail
- GetNotificationChannel
- GetPolicy
- ListComplianceStatus
- ListPolicies

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、以下を判別するために役立ちます。

- リクエストが、ルートユーザーの認証情報で行われたかどうか
- リクエストがロールまたはフェデレーションユーザーのテンポラリなセキュリティ認証情報を使用して行われたかどうか。
- AWS リクエストが別のサービスによってなされたかどうか。

詳細については、「CloudTrail userIdentity エレメント」を参照してください。

例: Firewall Manager のログファイルエントリ

トレイルは、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 ログファイルには 1 つ以上のログエントリが含まれます。イベントはあらゆるソース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に関する情報、アクションの日時、リクエストパラメータなどが含まれます。 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、特定の順序で表示されることはありません。

次の例は `GetAdminAccount`--> CloudTrail アクションを示すログエントリを示しています。

```
{
                "eventVersion": "1.05",
                "userIdentity": {
                                "type": "AssumedRole",
```

```
                                "principalId": "123456789098765432123",
                                "arn": "arn:aws:sts::123456789012:assumed-role/Admin/
SampleUser",
                                "accountId": "123456789012",
                                "accessKeyId": "1AFGDT647FHU83JHFI81H",
                                "sessionContext": {
                                                "attributes": {
                                                                "mfaAuthenticated":
 "false",
                                                                "creationDate":
 "2018-04-14T02:51:50Z"
                                                               },
                                                "sessionIssuer": {
                                                                "type": "Role",
                                                                "principalId":
 "123456789098765432123",
                                                                "arn":
 "arn:aws:iam::123456789012:role/Admin",
                                                                "accountId":
 "123456789012",
                                                                "userName": "Admin"
                                                                 }
                                                  }
                                },
                "eventTime": "2018-04-14T03:12:35Z",
                "eventSource": "fms.amazonaws.com",
                "eventName": "GetAdminAccount",
                "awsRegion": "us-east-1",
                "sourceIPAddress": "72.21.198.65",
                "userAgent": "console.amazonaws.com",
                "requestParameters": null,
                "responseElements": null,
                "requestID": "ae244f41-3f91-11e8-787b-dfaafef95fc1",
                "eventID": "5769af1e-14b1-4bd1-ba75-f023981d0a4a",
                "eventType": "AwsApiCall",
                "apiVersion": "2018-01-01",
                "recipientAccountId": "123456789012"
    }
```

AWS WAF と AWS Shield Advanced API を使用する

このセクションでは、Shield AWS WAF アドバンスドでのマッチセット、ルール、ウェブ ACL の作成と管理をおよび Shield Advanced API にリクエストする方法と、Shield アドバンスドでのサブスクリプションとプロテクションについて説明します。 AWS WAF さらに、リクエストの構成要素、レスポンスの内容、リクエストの認証方法について説明します。

トピック

AWS SDK を使用する

SDK AWS を提供する言語を使用している場合は、API を思い通りに処理するのではなく、SDK を使用してください。SDK を使用すると、認証が簡単になり、開発環境との統合が容易になり、Shield Advanced AWS WAF コマンドに簡単にアクセスできます。 AWS SDK の詳細については、ツールをダウンロードトピックのを参照してください。サービスを使用するためのアカウントのセットアップ

AWS WAF またはShield アドバンスドへの HTTPS リクエストの実行

AWS WAF およびShield アドバンストリクエストは、RFC 2616で定義されているHTTPSリクエストです。他の HTTP リクエストと同様に、 AWS WAF または Shield Advanced へのリクエストには、リクエストメソッド、URI、リクエストヘッダー、およびリクエスト本文が含まれます。レスポンスには HTTP ステータスコードとレスポンスヘッダーが含まれており、レスポンス本文が含まれている場合もあります。

リクエストの URI

リクエスト URI は常に 1 つのスラッシュ / です。

HTTP ヘッダー

AWS WAF およびShield アドバンスドでは、HTTP リクエストのヘッダーに次の情報が必要です。

Host (必須)

リソースが作成される場所を指定するエンドポイント。エンドポイントの詳細については、「AWS サービスエンドポイント」を参照してください。たとえば、Host CloudFront ディストリビューションのヘッダーの値はですwaf.amazonaws.com:443。 AWS WAF

x-amz-date または日付 (必須)

Authorization ヘッダーに含める署名を作成するときに使用できる日付。ISO 8601 の標準形式に基づいて UTC 時間で日付を指定します (次の例を参照)。

```
x-amz-date: 20151007T174952Z
```

x-amz-date または Date のどちらかを含める必要があります。(一部の HTTP クライアントライブラリでは、Date ヘッダーを設定することができません)。x-amz-dateヘッダーが存在する場合、 AWS WAF Dateリクエストの認証時にどのヘッダーも無視します。

タイムスタンプは、 AWS リクエストを受信したときのシステム時間の 15 分以内である必要があります。このようにしないと、リクエストは RequestExpired エラーコードで失敗し、任意のユーザーがリクエストを再現できなくなります。

Authorization (必須)

リクエスト認証に必要な情報。このヘッダーの作成方法の詳細については、「リクエストの認証」を参照してください。

X-Amz-Target (必須)

AWSWAF_ または AWSShield_、API バージョン (ピリオドなし)、ピリオド (.)、オペレーション名を連結したもの。例えば、以下のようになります。

AWSWAF_20150824.CreateWebACL

Content-Type (条件付き)

コンテンツタイプとして JSON をそのバージョンと共に指定します。例えば、次のようになります。

```
Content-Type: application/x-amz-json-1.1
```

条件: POST リクエストに必要です。

Content-Length (条件付き)

RFC 2616 に基づくメッセージの長さ (ヘッダーなし)。

条件: リクエストボディ自体に情報が含まれる場合、必須です (このヘッダーは、ほとんどのツールキットで自動的に追加されます)。

次に示しているのは、 AWS WAFでウェブ ACL を作成するための HTTP リクエストのヘッダーの例です。

```
POST / HTTP/1.1
Host: waf.amazonaws.com:443
X-Amz-Date: 20151007T174952Z
Authorization: AWS4-HMAC-SHA256
               Credential=AccessKeyID/20151007/us-east-2/waf/aws4_request,
               SignedHeaders=host;x-amz-date;x-amz-target,

 Signature=145b1567ab3c50d929412f28f52c45dbf1e63ec5c66023d232a539a4afd11fd9
X-Amz-Target: AWSWAF_20150824.CreateWebACL
Accept: */*
Content-Type: application/x-amz-json-1.1; charset=UTF-8
Content-Length: 231
Connection: Keep-Alive
```

HTTP リクエストボディ

AWS WAF 多くの Shield アドバンスド API アクションでは、リクエストの本文に JSON 形式のデータを含める必要があります。

次に示しているリクエストの例では、シンプルな JSON ステートメントを使用して、IP アドレス 192.0.2.44 (CIDR 表記では 192.0.2.44/32) を含めるように、IPSet を更新しています。

```
POST / HTTP/1.1
Host: waf.amazonaws.com:443
X-Amz-Date: 20151007T174952Z
Authorization: AWS4-HMAC-SHA256
               Credential=AccessKeyID/20151007/us-east-2/waf/aws4_request,
               SignedHeaders=host;x-amz-date;x-amz-target,

 Signature=145b1567ab3c50d929412f28f52c45dbf1e63ec5c66023d232a539a4afd11fd9
```

```
X-Amz-Target: AWSWAF_20150824.UpdateIPSet
Accept: */*
Content-Type: application/x-amz-json-1.1; charset=UTF-8
Content-Length: 283
Connection: Keep-Alive

{
   "ChangeToken": "d4c4f53b-9c7e-47ce-9140-0ee5ffffffff",
   "IPSetId": "69d4d072-170c-463d-ab82-0643ffffffff",
   "Updates": [
      {
         "Action": "INSERT",
         "IPSetDescriptor": {
            "Type": "IPV4",
            "Value": "192.0.2.44/32"
         }
      }
   ]
}
```

HTTP レスポンス

すべての AWS WAF Shield アドバンスド API アクションには、JSON 形式のデータがレスポンスに含まれます。

HTTP レスポンスの重要なヘッダーと、それらをアプリケーション内で扱う方法 (該当する場合) を示します。

HTTP/1.1

このヘッダーにはステータスコードが続きます。ステータスコード 200 はオペレーションの成功を示します。

型: 文字列

x-amzn-RequestId

AWS WAF または Shield Advanced によって作成され、リクエストを一意に識別する値 (例:)。K2QH8DN0U907N97FNA2GDLL80BVV4KQNSO5AEMVJF66Q9ASUAAJGに問題がある場合は AWS WAF、 AWS この値を使用して問題のトラブルシューティングを行うことができます。

型: 文字列

Content-Length

レスポンス本文の長さ (バイト単位)。

型: 文字列

日付

Shield AWS WAF アドバンスドが応答した日付と時刻。たとえば、2015 年 10 月 7 日 (水) 12:00:00 (GMT) など。

型: 文字列

エラーレスポンス

リクエストの結果がエラーの場合、HTTP レスポンスには次の値が含まれます。

- レスポンス本文としての JSON エラードキュメント
- Content-Type
- 該当する HTTP ステータスコード (3xx、4xx、または 5xx)

次に示しているのは、JSON エラードキュメントの例です。

```
HTTP/1.1 400 Bad Request
x-amzn-RequestId: b0e91dc8-3807-11e2-83c6-5912bf8ad066
x-amzn-ErrorType: ValidationException
Content-Type: application/json
Content-Length: 125
Date: Mon, 26 Nov 2012 20:27:25 GMT

{"message":"1 validation error detected: Value null at 'TargetString' failed to satisfy
 constraint: Member must not be null"}
```

リクエストの認証

SDK AWS を提供する言語を使用する場合は、その SDK を使用することをお勧めします。すべての AWS SDK を使用すると、 AWS WAF または Shield Advanced API を使用する場合と比較して、リクエストに署名するプロセスが大幅に簡略化され、時間を大幅に節約できます。また、SDK は開発環境と容易に統合されるため、関連するコマンドへのアクセスが簡単です。

AWS WAF Shield Advanced では、送信するすべてのリクエストに署名して認証を受ける必要があります。リクエストに署名するには、暗号化ハッシュ関数を使用してデジタル署名を計算します。この関数は入力に基づいてハッシュ値を返します。入力には、リクエストのテキスト、およびシークレットアクセスキーが含まれます。ハッシュ関数から返されるハッシュ値をリクエストに署名として含めます。署名は、リクエストの `Authorization` ヘッダーの一部です。

リクエストを受け取ると、 AWS WAF または Shield Advanced は、リクエストの署名に使用したのと同じハッシュ関数と入力を使用して署名を再計算します。結果の署名がリクエスト内の署名と一致する場合、 AWS WAF または Shield Advanced がリクエストを処理します。一致しない場合、リクエストは拒否されます。

AWS WAF および Shield アドバンスドは、AWS 署名バージョン 4 を使用した認証をサポートしています。署名の計算プロセスは 3 つのタスクに分けることができます。

タスク 1: 正規リクエストを作成する

『https://docs.aws.amazon.com/general/latest/gr/sigv4-create-canonical-request.html』の「Amazon Web Services 全般のリファレンスタスク 1: 署名バージョン 4 の正規リクエストを作成する」で説明されているように、正規形式で HTTP リクエストを作成します。

タスク 2: 署名対象の文字列を作成する

暗号化ハッシュ関数への入力値の 1 つとして使用する文字列を作成します。署名対象の文字列と呼ばれる文字列は、次の値を連結したものです。

- ハッシュアルゴリズムの名前
- リクエスト日
- 認証情報スコープ文字列
- 前のタスクからの正規形式のリクエスト

認証情報スコープの文字列自体は、日付、リージョン、およびサービス情報を結合したものです。

`X-Amz-Credential` パラメータでは、次の内容を指定します。

- リクエストの送信先であるエンドポイントのコード (`us-east-2`)
- サービスの省略形としての `waf`

例:

```
X-Amz-Credential=AKIAIOSFODNN7EXAMPLE/20130501/us-east-2/waf/
aws4_request
```

タスク 3: 署名を作成する

2 つの入力文字列を受け取る暗号化ハッシュ関数を使用して、リクエストの署名を作成します。

- タスク 2 からの署名対象の文字列。
- 派生キー。シークレットアクセスキーから開始し、認証情報スコープの文字列を使用して一連のハッシュベースのメッセージ認証コード (HMAC) を作成することで、派生キーが計算されます。

関連情報

このサービスを利用する際に役立つ関連リソースは次のとおりです。

、 AWS WAF AWS Shield Advanced、には次のリソースがあります AWS Firewall Manager。

- 実装ガイドライン AWS WAF — 既存および新規の Web AWS WAF アプリケーションを保護するための実装に関する最新の推奨事項を記載した技術文書。
- AWS ディスカッションフォーラム — このサービスや他のサービスに関連する技術的な質問を議論するための、コミュニティベースのフォーラムです。 AWS
- AWS WAF ディスカッションフォーラム — 開発者向けのコミュニティベースのフォーラムで、関連する技術的な質問について話し合います。 AWS WAF
- Shield Advanced ディスカッションフォーラム – デベロッパーが Shield Advanced に関連する技術的な質問について話し合うためのコミュニティベースのフォーラム。
- AWS WAF 製品情報 — 機能 AWS WAF、価格などに関する情報を提供する主要な Web ページです。
- Shield Advanced の製品情報 – 機能、料金など、Shield Advanced に関する情報の主要なウェブページ。

次のリソースは Amazon Web Services で利用可能です。

- クラスとワークショップ — 自分のペースで進めることができるラボに加えて、職務ベースのコースや専門コースへのリンク。 AWS スキルを磨き、実践的な経験を積むのに役立ちます。
- AWS デベロッパーセンター — チュートリアルを調べたり、ツールをダウンロードしたり、開発者イベントについて学んだりできます。 AWS
- AWS 開発者ツール — アプリケーションの開発と管理に役立つ開発者ツール、SDK、IDE ツールキット、コマンドラインツールへのリンク。 AWS
- 入門リソースセンター — アプリケーションのセットアップ方法 AWS アカウント、 AWS コミュニティへの参加方法、初めてのアプリケーションの起動方法について説明します。
- ハンズオンチュートリアル — チュートリアルに従って、 step-by-step 初めてのアプリケーションを起動しましょう。 AWS
- AWS ホワイトペーパー — アーキテクチャ、セキュリティ、経済などのトピックを扱い、 AWS ソリューションアーキテクトやその他の技術専門家が作成した技術ホワイトペーパーの包括的なリストへのリンク。 AWS

- AWS Support センター — ケースの作成と管理のハブです。 AWS Support フォーラム、技術的なよくある質問、サービスの状態など、その他の役立つリソースへのリンクも含まれています。 AWS Trusted Advisor
- AWS Support— クラウドでのアプリケーションの構築と実行を支援する AWS Support one-on-one、迅速に対応できるサポートチャネルに関する情報を掲載する主要ウェブページです。
- お問い合わせ - AWS の請求、アカウント、イベント、不正使用、その他の問題などに関するお問い合わせの受付窓口です。
- AWS サイト規約 — 当社の著作権と商標、お客様のアカウント、ライセンス、サイトへのアクセス、およびその他のトピックに関する詳細情報。

ドキュメント履歴

このページでは、このドキュメントの大きな変更点をまとめています。

サービス機能は、サービスが利用可能な AWS リージョンに段階的にロールアウトされることがあります。このドキュメントは、最初のリリースのためにのみ更新されています。リージョンの可用性に関する情報を提供したり、その後のリージョンのロールアウトを発表したりすることはありません。サービス機能のリージョンの可用性と更新に関する通知のサブスクライブについては、「 の最新情報 AWS」を参照してください。

変更	説明	日付
JSON 本文解析の仕組みを明確にする	JSON 本文検査のカバレッジを更新して、 が解析をどのように AWS WAF 処理し、本文がフォールバック動作を解析するかを明確にしました。	2024 年 6 月 25 日
の AWS マネージドルールを更新しました AWS WAF	Linux オペレーティングシステムのルールセットを更新しました。	2024 年 6 月 6 日
AWS WAF マネージドポリシーの変更	`WAFV2LoggingServiceRolePolicy` および を更新`AWSServiceRoleForWAFV2Logging` して、ステートメント IDs (Sid) をアクセス許可設定に追加しました。	2024 年 6 月 3 日
AWS WAF マネージドポリシーの変更の追跡	AWS WAF は、 マネージドポリシー `WAFV2LoggingServiceRolePolicy` とサービスにリンクされたロール の変更の追跡を開始しました`AWSServic`	2024 年 6 月 3 日

	eRoleForWAFV2Logging 。	
の AWS マネージドルールを更新しました AWS WAF	Bot Control、ATP、および ACFP マネージドルールグループがバージョニングされ、他のバージョニング管理ルールと同様に、バージョン更新の SNS AWS 通知が提供されます。	2024 年 5 月 29 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが POSIX オペレーティングシステムのルールグループ AWS WAF を更新しましたAWSManagedRulesUnixRuleSet 。	2024 年 5 月 28 日
CAPTCHA および Challenge アクション	ブラウザクライアントが CAPTCHA パズルとサイレントチャレンジを実行するために HTTPS を必要とすることを明確にしました。	2024 年 5 月 20 日
Amazon Security Lake との統合	Security Lake を使用してウェブ ACL トラフィックデータを収集できるようになりました。詳細については、「Amazon Security Lake ユーザーガイド」の「 AWS のサービスからのデータ収集」を参照してください。	2024 年 5 月 22 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールがコアルールセット (CRS) ルールグループ AWS WAF を更新しました。	2024 年 5 月 21 日

の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが SQLi データベースルールグループ AWS WAF を更新しました。	2024 年 5 月 14 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールは、既知の不正な入力と POSIX オペレーティングシステムのルールグループ AWS WAF を更新しました。	2024 年 5 月 8 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが Windows オペレーティングシステムのルールグループ AWS WAF を更新しました。	2024 年 5 月 3 日
AWS WAF モバイル SDK Android Kotlin コードサンプル	Kotlin ベースの Android 統合のサンプルコードを追加しました。	2024 年 5 月 2 日
AWS WAF メトリクスがディメンションと新しいメトリクスを追加	AWS WAF は、ルールメトリクス`ManagedRuleSetRule` に の新しいディメンションを追加し、ラベルメトリクスの一致ルールアクションに新しいメトリクスを追加しました。	2024 年 5 月 2 日
AWS Firewall Manager がネットワーク ACL ポリシーをサポート	Firewall Manager は、Firewall Manager ネットワーク ACLs ポリシーによる Amazon VPC ネットワークアクセスコントロールリスト (ACL) の管理をサポートするようになりました。	2024 年 4 月 25 日

AWS Firewall Manager セキュリティポリシーの更新	ネットワーク ACL を管理するためのアクセス許可を追加するために を FMSServiceRolePolicy に更新しました。 ACLs	2024 年 4 月 22 日
ヘルスチェックメトリクスリストの更新	ヘルスチェックで一般的に使用されるメトリクスのリストからメトリクスを削除しました。	2024 年 4 月 16 日
Firewall Manager セキュリティグループポリシーの更新	使用状況監査セキュリティグループポリシーを更新し、ドキュメントを改善しました。「使用状況監査ポリシー」セクションと、ベストプラクティスと制限に関するセクションを参照してください。	2024 年 4 月 2 日
Bot Control の例を更新しました	ターゲット検査レベルを示す例を追加し、ベストプラクティスを反映するように既存の例を更新しました。	2024 年 3 月 27 日
ATP の例を更新	レスポンス検査の設定を示す例を追加し、ベストプラクティスを反映するように既存の例を更新しました。	2024 年 3 月 27 日
ACFP の例を更新	レスポンス検査設定を示す例を追加しました。	2024 年 3 月 27 日
Amazon CloudWatch Logs ロググストリームの制限を更新する	AWS WAF CloudWatch Logs ロググストリームへのログの発行に関するウェブごとの ACL 制限はなくなりました。	2024 年 3 月 27 日

AWS Shield Advanced アプリケーションレイヤー (レイヤー 7) 保護	アプリケーションレイヤーの検出と緩和、ウェブ ACL の使用、レートベースのルール、およびアプリケーションレイヤー DDoS 自動緩和に関する一般的なガイダンスとベストプラクティスガイダンスを更新しました。	2024 年 3 月 14 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが IP 評価ルールグループ AWS WAF を更新しました。	2024 年 3 月 13 日
本文検査のサイズ制限の変更	AWS WAF では、一部のリージョンのリソースについて、本文検査サイズの制限が大きくなりました。	2024 年 3 月 7 日
AWS WAF レートベースのルールの設定可能な評価ウィンドウ	レートベースのルールがリクエストをカウントするために使用する時間枠を 1、2、5、または 10 分に設定できるようになりました。デフォルトは 5 で、このリリースより前の唯一のオプションでした。	2024 年 2 月 28 日
CAPTCHAおよび のログ記録情報の拡大 Challenge	最上位レベル`captchaResponse` と`challengeResponse` フィールドには、終了中か非終了かにかかわらず、リクエストに適用されるこれらのアクションの最後の値が入力されるようになりました。以前は、これらのフィールドは終了アクションに対してのみ入力されていました。	2024 年 2 月 22 日

JavaScript CAPTCHA API キー管理	API を使用して CAPTCHA JS AWS WAF APIsキーを削除できるようになりました。	2024 年 2 月 6 日
AWS WAF CAPTCHA パズルオーディオ	CAPTCHA パズルのオーディオバージョンは、複数の言語をサポートするようになりました。	2024 年 2 月 6 日
AWS WAF チャレンジと CAPTCHA トークンのラベル付け	トークン管理で CAPTCHA トークンのラベルが追加され、チャレンジトークンのトークンラベル付けが強化されました。	2023 年 12 月 20 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールは、既知の不正な入力ルールグループ AWS WAF を更新しました。	2023 年 12 月 16 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールは、既知の不正な入力ルールグループ AWS WAF を更新しました。	2023 年 12 月 14 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールがコアルールセット (CRS) ルールグループ AWS WAF を更新しました。	2023 年 12 月 6 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールで、次のルールグループ AWS WAF が更新されました。 AWS WAF Bot Control。	2023 年 12 月 5 日

Firewall Manager の AWS Config 前提条件を更新	の Firewall Manager マネージドロールの代わりにカスタム IAM ロールを使用する場合は AWS Config、アクセス許可ポリシーで AWS Config レコーダーが Firewall Manager リソースを記録できるようにする必要があります。	2023 年 11 月 17 日
AWS WAF コンソールダッシュボード	AWS WAF コンソールでウェブ ACL のすべてのルールとサンプリングされたリクエストを表示するためのガイダンスを修正しました。	2023 年 11 月 17 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが Bot Control ルールグループ AWS WAF を更新しました。	2023 年 11 月 14 日
AWS WAF コンソールに新しいウェブ ACL ダッシュボードがある	AWS WAF コンソールのウェブ ACL ページには、新しいウェブトラフィック概要ダッシュボードがあります。	2023 年 11 月 14 日
ATP マネージドルールグループを更新しました	ルール `VolumetricIpFailedLoginResponseHigh` および `VolumetricSessionFailedLoginResponseHigh` のラベル情報を修正しました。	2023 年 11 月 13 日

更新された ACFP マネージドルールグループ	ルール `VolumetricIPSuccessfulResponse` および `VolumetricSessionSuccessfulResponse` のラベル情報を修正しました。	2023 年 11 月 13 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールがコアルールセット (CRS) ルールグループ AWS WAF を更新しました。	2023 年 11 月 2 日
Shield Advanced アプリケーションレイヤー DDoS 自動緩和	Shield Advanced は、DDoS 攻撃のソースであることが判明している IP アドレスからのリクエストの量を制限するレートベースのルールを自動緩和ルールグループに保持するようになりました。	2023 年 10 月 31 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールがコアルールセット (CRS) ルールグループ AWS WAF を更新しました。	2023 年 10 月 30 日
Bot Control マネージドルールグループは、リクエスト CSP のシグナルラベルを削除しました	Bot Control マネージドルールグループから、クラウドサービスプロバイダー (CSP) を示すシグナルラベルが削除されました。	2023 年 10 月 28 日
リクエスト CSP の Bot Control マネージドルールグループシグナルラベル	Bot Control マネージドルールグループのシグナルラベルには、クラウドサービスプロバイダー (CSP) を示すラベルが含まれています。	2023 年 10 月 27 日

AWS WAF IAM アクセス許可情報の更新	ウェブ ACL の関連付けを管理する AWS WAF アクションについて、ポリシーアクションセクションに、各ウェブアプリケーションリソースタイプのアクセス許可要件が一覧表示されるようになりました。	2023 年 10 月 25 日
Firewall Manager による変更されたウェブ ACL の管理	関連付けられていないウェブ ACL の管理を有効にすると、Firewall Manager は変更されたウェブ ACL を未使用リソースの 1 回限りのクリーンアップに含めません。	2023 年 10 月 19 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが POSIX オペレーティングシステムのルールグループ AWS WAF を更新しましたAWSManagedRulesUnixRuleSet 。	2023 年 10 月 12 日
AWS WAF メトリクスにディメンションが追加されました	AWS WAF は、ウェブ ACL メトリクスを表示するための新しいディメンションを追加しました。	2023 年 10 月 12 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールがコアルールセット (CRS) ルールグループ AWS WAF を更新しました。	2023 年 10 月 11 日
AWS WAF モバイル SDK 仕様の更新	storeTokenInCookieStorage 操作を WAFTokenProvider に追加しました。	2023 年 10 月 11 日

の例外デプロイ AWS マネージドルール AWS WAF	AWS の マネージドルールは、既知の不正な入力ルールグループの 2 つの静的バージョンを AWS WAF リリースし、最新の静的バージョンを指すようにデフォルトバージョンを更新しました。	2023 年 10 月 4 日
AWS WAF HTML エンティティデコードテキスト変換	HTML エンティティデコードのテキスト変換機能を拡張しました。	2023 年 10 月 4 日
Firewall Manager のセキュリティグループ共通ポリシーに新しいオプションを追加しました	Firewall Manager は、セキュリティグループの参照をレプリカのセキュリティグループに配布できるようになりました。	2023 年 10 月 3 日
AWS WAF が JA3 フィンガープリントの検査を追加	Amazon CloudFront ディストリビューションと Application Load Balancer について、ウェブリクエストの JA3 フィンガープリントと完全に一致できるようになりました。	2023 年 9 月 26 日
Firewall Manager セキュリティグループのポリシールール設定の更新	Firewall Manager は、プライマリセキュリティグループからレプリカセキュリティグループへのセキュリティグループの参照をサポートするようになりました。	2023 年 9 月 25 日

Shield Advanced アプリケーションレイヤー DDoS 自動緩和を更新しました	Firewall Manager は、アプリケーションレイヤー DDoS 自動緩和が設定された Shield Advanced ポリシーの Application Load Balancer リソースをサポートするようになりました。	2023 年 9 月 14 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールで、次のルールグループ AWS WAF が更新されました。AWS WAF Bot Control。	2023 年 9 月 6 日
AWS WAF ボットコントロール	Bot Control マネージドルールグループのターゲットを絞った保護レベルで、IP アドレス間のトークンの再利用を検出できるようになりました。また、オプションでトラフィック統計の機械学習分析が可能になり、ボット関連のアクティビティも検出できるようになりました。	2023 年 9 月 6 日
AWS WAF モバイル SDK 仕様の更新	`tokenRefreshDelaySec` の最小値、最大値、デフォルト値をそれぞれ 300、600、300 から、88、300、88 に引き下げました。	2023 年 9 月 5 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが AWS WAF Bot Control ルールグループ AWS WAF を更新しました。	2023 年 8 月 30 日

Shield Advanced アプリケーションレイヤー DDoS 自動緩和	を使用して AWS CloudFormation 、アプリケーションレイヤー DDoS 自動緩和で使用するウェブ ACLs を管理するためのガイダンスを追加しました。	2023 年 8 月 30 日
セキュリティグループ監査ポリシーの Firewall Manager の新しいオプション	過度に許容されるルールグループを監査するための新しいオプションが追加され、コンソール手順の説明が改善されました。	2023 年 8 月 29 日
新しい Firewall Manager Shield および AWS WAF ポリシーオプション	および Shield で関連付けられていないウェブ ACLs の管理を有効にする AWS WAF と、Firewall Manager は、ウェブ ACLs が少なくとも 1 つのリソースによって使用される場合にのみ、ポリシー範囲内のアカウントにウェブ ACLs を作成します。	2023 年 8 月 9 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールがコアルールセット (CRS) ルールグループ AWS WAF を更新しました。	2023 年 7 月 26 日
URI パスにおけるレートベースのルールの集約	レートベースのルールのカスタム集約キーに URI パスを指定できるようになりました。	2023 年 7 月 19 日
の新しい AWS WAFポリシールールオプション AWS Firewall Manager	AWS Firewall Manager では、AWS WAF ウェブリクエストボディの検査サイズ制限の設定のサポートが追加されました。	2023 年 7 月 18 日

AWS WAF マネージドポリシーの変更	AWSWAFFullAccessPolicy 、、AWSWAFConsoleFullAccess 、および を更新AWSWAFConsoleReadOnlyAccess してAWSWAFReadOnlyAccess 、 で保護できるリソースタイプに AWS Verified Access を追加しました AWS WAF。	2023 年 6 月 17 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールにルールグループ AWS WAF が追加されましたAWSManagedRulesACFPRuleSet 。	2023 年 6 月 13 日
AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) の更新	正規表現を使用して、ATP マネージドルールグループのログインエンドポイントを指定できるようになりました。	2023 年 6 月 13 日
CAPTCHA JavaScript API の新しい情報	新しいセクションでは、 が CAPTCHA でリクエストに AWS WAF 応答したときにカスタム CAPTCHA パズルを提供する方法について説明します。	2023 年 6 月 13 日
新しい ACFP マネージドルールグループ	新しいルールグループ AWSManagedRulesACFPRuleSet を使用して、アカウント作成の不正な試みを検出およびブロックします。	2023 年 6 月 13 日

新しい AWS WAF Fraud Control Account Creation Fraud Prevention (ACFP)	新しい Fraud Control Account Creation AWS WAF Fraud Prevention (ACFP) マネージドルールグループ を使用して、不正なアカウント作成の試みを検出およびブロックできますAWSManagedRulesACFPRuleSet 。保護された CloudFront ディストリビューションでは、ACFP を使用して、最近失敗したアカウント作成試行が多すぎるクライアントからの新しいアカウント作成の試行をブロックすることもできます。	2023 年 6 月 13 日
AWS WAF マネージドポリシーの変更	AWSWAFFullAccessPolicy 、、AWSWAFConsoleFullAccess 、および を更新AWSWAFConsoleReadOnlyAccess してAWSWAFReadOnlyAccess 、 AWS App Runner サービスのアクセス設定を修正しました。	2023 年 6 月 6 日
Firewall Manager のセキュリティグループポリシーの制限を追加	共有 VPC が後で共有を解除された場合、Firewall Manager は関連するアカウントのレプリカセキュリティグループを削除しません。	2023 年 6 月 2 日
新しい AWS WAF リクエストコンポーネント: Header order	リクエスト内のヘッダー名の順序付けされたリストと照合できるようになりました。	2023 年 5 月 30 日

の AWS マネージドルールを更新しました AWS WAF	Linux オペレーティングシステムのルールセットを更新しました。	2023 年 5 月 22 日
AWS WAF ルールセクションの組織を更新しました	ルールステートメントのリストがステートメントタイプ別にグループ化されるようになりました。	2023 年 5 月 16 日
トピックの移動: レート制限されている IP アドレスの一覧表示	レートベースのルールによってレート制限されている IP アドレスを一覧表示するトピックが、レートベースのルールのトピックに追加されました。	2023 年 5 月 16 日
レートベースのルールの拡張オプション	IP アドレス以外の集約キーに基づくウェブリクエストのレート制限や、キーの組み合わせを使用した集約が可能になりました。また、スコープダウンステートメントに一致するすべてのリクエストをさらに集約せずにレート制限することもできます。	2023 年 5 月 16 日
Firewall Manager のクォータの引き上げ	の組織あたりの Firewall Manager ポリシーの数を 20 AWS Organizations から 50 に増やしました。ポリシーごとのプライマリセキュリティグループの最大数を 1 から 3 に増やしました。WCU の最大数をソフトクォータからハードクォータに変更しました。	2023 年 5 月 5 日

ルールグループあたりの最大 WCU が引き上げられました	サポートに引き上げをリクエストしなくても、ルールグループあたり最大 5,000 WCU を使用できるようになりました。この新しい制限を引き上げることはできません。	2023 年 5 月 1 日
AWS WAF プレフィックス付きの Amazon S3 ログバケットの場所	AWS WAF で Amazon S3 ログバケット名にプレフィックスを使用できるようになりました。	2023 年 5 月 1 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールがコアルールセット (CRS) ルールグループ AWS WAF を更新しました。	2023 年 4 月 28 日
AWS Verified Access インスタンスのサポートが に追加されました AWS WAF	AWS WAF ウェブ ACL を Verified Access インスタンスに関連付けることができるようになりました。この変更は、 の最新バージョンでのみ使用でき AWS WAF 、 AWS WAF Classic では利用できません。	2023 年 4 月 28 日
複数の Firewall Manager 管理者との連携に関する章を改訂しました	組織のファイアウォールリソースを作成および管理する Firewall Manager 管理者を、複数指定できるようになりました。	2023 年 4 月 24 日
AWS Firewall Manager マネージドポリシーの更新	更新済み `FMSServiceRolePolicy` 。	2023 年 4 月 21 日

CAPTCHA の新しい JavaScript クライアントアプリケーション統合	JavaScript クライアントアプリケーションで CAPTCHA パズルの配置と特性をカスタマイズできるようになりました。	2023 年 4 月 20 日
アプリケーションインテグレーションの名前をインテリジェントスレット統合に変更	クライアントアプリケーション統合の既存の機能の名前をインテリジェントな脅威に対応した統合に変更し、 用の新しい CAPTCHA アプリケーション統合と区別しやすくしました JavaScript。	2023 年 4 月 20 日
1,500 WCU を超えるウェブ ACL に適用される変動料金	ウェブ ACL で 1,500 WCU を使用すると追加コストが発生しますが、このコストは、ウェブ ACL WCU の使用量の増減に応じて自動的に調整されます。ウェブ ACL の最大容量は 5,000 WCU です。	2023 年 4 月 11 日
ウェブ ACL あたりのWCU の最大数が引き上げられました	サポートに引き上げをリクエストしなくても、ウェブ ACL あたり最大 5,000 WCU を使用できるようになりました。この新しい制限を引き上げることはできません。	2023 年 4 月 11 日
CloudFront ウェブ ACLs の本文検査サイズ制限	Amazon CloudFront ディストリビューションを保護するウェブ ACLs の場合、ウェブ ACL 設定で本文検査サイズの制限を最大 64 KB まで増やすことができます。	2023 年 4 月 11 日

の本文検査サイズの増加 CloudFront	Amazon CloudFront ディストリビューションの AWS WAF 本文検査の最大サイズ制限が 8 KB から 64 KB に引き上げられました。のデフォルトの検査サイズ制限 CloudFront は 16 KB です。	2023 年 4 月 11 日
の新しい AWS WAFポリシールールオプション AWS Firewall Manager	AWS Firewall Manager は、AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) および AWS WAF Bot Control AWS マネージドルールのルールグループ、Amazon S3 ログ記録先、ルールアクションの上書き、CAPTCHAChallenge ルールアクション、トークンドメインリストのサポートを追加します。	2023 年 4 月 7 日
Firewall Manager が ログ記録の送信先として Amazon S3 AWS WAF バケットをサポート	Amazon S3 バケットを AWS WAF ポリシーのログ記録先として使用できるようになりました。	2023 年 4 月 7 日
AWS WAF マネージドポリシーの変更	AWSWAFFullAccessPolicy 、、AWSWAFConsoleFullAccess 、および を更新AWSWAFConsoleReadOnlyAccess してAWSWAFReadOnlyAccess 、 で保護できるリソースタイプに AWS App Runner サービスを追加しました AWS WAF。	2023 年 3 月 30 日

セキュリティグループポリシー内でのタグの使用に関する警告を追加しました	ポリシーに、組織のタグポリシーと矛盾するタグが存在する場合に、Firewall Manager が既存のセキュリティグループでのタグ更新や、新しいセキュリティグループの作成を行うことはありません。	2023 年 3 月 28 日
サービスロール情報の更新	Firewall Manager でのサービスロールの使用方法を更新しました。	2023 年 3 月 8 日
レートベースのルールによるレート制限処理の方法についての情報を修正しました	スコープダウンステートメントを含むレートベースのルールは、ルールのスコープダウンステートメントと一致するリクエストのみをレート制限します。以前は、レート制限された IP アドレスのすべてのリクエストに対して、この制限が適用される旨を表記していました。	2023 年 3 月 1 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが PHP アプリケーションルールグループ AWS WAF を更新しました。	2023 年 2 月 27 日
AWS App Runner への のサポートを追加 AWS WAF	AWS WAF ウェブ ACL を AWS App Runner サービスに関連付けることができるようになりました。この変更は、の最新バージョンでのみ使用でき AWS WAF 、 AWS WAF Classic では利用できません。	2023 年 2 月 23 日

の IAM ガイダンスを更新しました AWS Firewall Manager	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 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 16 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールがルールグループ AWS WAF を更新しAWSManagedRulesATPRuleSet 、Amazon CloudFront ディストリビューションを保護するウェブ ACLs にログインレスポンス検査を追加しました。	2023 年 2 月 15 日
AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ログインレスポンス検査	保護された CloudFront ディストリビューションでは、ATP を使用して、最近送信したログイン試行の失敗回数が多すぎるクライアントからの新しいログイン試行をブロックできるようになりました。	2023 年 2 月 15 日
の AWS マネージドルールを更新しました AWS WAF	コアルールセットを更新しました。	2023 年 1 月 25 日
インテリジェントな脅威の軽減のためのベストプラクティス	Bot Control、ATP、およびその他のインテリジェントな脅威の軽減機能を実装するためのベストプラクティスを記載したセクションを追加しました。	2023 年 1 月 22 日

HTTP/2 疑似ヘッダーの検査方法	HTTP/2 疑似ヘッダーを対応するウェブリクエストコンポーネントにマップするセクションを追加しました。	2023 年 1 月 20 日
AWS WAF Classic の IAM ガイダンスを更新しました	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 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 1 月 3 日
の IAM ガイダンスを更新しました AWS WAF	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 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 1 月 3 日
の IAM ガイダンスを更新しました AWS Shield	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 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 1 月 3 日
Amazon Route 53 Resolver DNS Firewall ポリシーの更新	Amazon Route 53 Resolver の DNS Firewall ルールグループの削除に関する情報を追加しました。	2022 年 12 月 29 日
の AWS マネージドルールを更新しました AWS WAF	Linux オペレーティングシステムのルールセットを更新しました。	2022 年 12 月 15 日
の AWS マネージドルールを更新しました AWS WAF	コアルールセットを更新しました。	2022 年 12 月 5 日

Firewall Manager は、Fortigate Cloud Native Firewall (CNF) as a Service ポリシーのサポートを追加します	Firewall Manager が Fortigate CNF ポリシーをサポートするようになりました。	2022 年 12 月 2 日
DNS Firewall ポリシー AWS Config の要件を削除しました	DNS Firewall ポリシーでは、リソースタイプ EC2 VPC の Config を有効にするだけでよくなりました。	2022 年 11 月 17 日
AWS Firewall Manager マネージドポリシーの更新	更新済み `FMSServiceRolePolicy` 。	2022 年 11 月 15 日
AWS WAF CAPTCHA パズルの言語オプションの拡張	CAPTCHA パズルは、複数の言語で書かれた説明を提供するようになりました。各オーディオパズル内の説明は、現在も英語でのみ提供されています。	2022 年 11 月 11 日
Firewall Manager のリソースセットの新しいクォータ	リソースセットの新しいクォータを追加しました。	2022 年 11 月 8 日
リソースセットのサポートを追加する	リソースセットを作成して、Firewall Manager ポリシーで管理するリソースをグループ化できます。	2022 年 11 月 8 日
Network Firewall からのファイアウォールのインポートのサポートを追加する	リソースセットを使用して、Network Firewall ポリシーの既存のファイアウォールをインポートして管理できるようになりました。	2022 年 11 月 8 日
AWS Firewall Manager マネージドポリシーの更新	更新済み `AWSFMAdminReadOnlyAccess` 。	2022 年 11 月 2 日

地理一致ステートメントは、国と地域のリクエストにラベルが追加されるようになりました	地理マッチングとラベルマッチングを組み合わせることで、リージョンレベルで地理的なリクエスト元を管理できるようになりました。	2022 年 10 月 31 日
上位レベルのセクションの名前を「マネージド保護」に変更しました	セクションには、マーケティングページに沿った AWS WAF インテリジェントな脅威の軽減という名前が付けられました。	2022 年 10 月 27 日
Bot Control マネージドルールグループのターゲットを絞った新しい保護レベル	Bot Control マネージドルールグループには、高度なボットを検出および軽減するためのターゲットを絞ったルールが追加されました。この保護レベルは追加料金でご利用いただけます。	2022 年 10 月 27 日
AWS WAF トークンに関する新しいセクション	がインテリジェントな脅威の軽減にトークン AWS WAF を使用する方法を理解します。	2022 年 10 月 27 日
Firewall Manager Network Firewall ポリシーの更新に関する重要な注意事項を追加しました	Firewall Manager ポリシーを更新すると、そのポリシーによって作成されたすべての Network Firewall ポリシーが、Firewall Manager ポリシーの Network Firewall ポリシー構成で更新されます。	2022 年 10 月 27 日

ルールグループ内のアクションオーバーライド	ルールグループ内のルールアクションを任意のルールアクション設定にオーバーライドできるようになりました。前の Count アクションオーバーライドと同様に、オーバーライドをルールグループ内のすべてのルールと個々のルールに適用できます。	2022 年 10 月 27 日
AWS WAF 新しいChallenge ルールアクションオプション	リクエストがブラウザから送信されていることを確認するために、Challenge を使用するようにルールを設定できます。	2022 年 10 月 27 日
AWS WAF は、保護された複数のアプリケーション間でトークンを共有できます	ウェブ ACL のトークンドメインリストを設定することで、保護されている複数のアプリケーション間でトークンの使用を有効にできます。	2022 年 10 月 27 日
すべてのヘッダーの仕様は大文字と小文字を区別しない	すべてのヘッダーの仕様を大文字と小文字を区別しないように変更しました。これは単一ヘッダーの動作と一致します。	2022 年 10 月 26 日
AWS Firewall Manager マネージドポリシーの変更	`AWSFMAdminFullAccess` の修正。	2022 年 10 月 21 日
の AWS マネージドルールを更新しました AWS WAF	既知の不正入力ルールグループを更新しました。	2022 年 10 月 20 日
の AWS マネージドルールを更新しました AWS WAF	既知の不正入力ルールグループを更新しました。	2022 年 10 月 5 日

AWS WAF モバイル SDK 仕様の更新	tokenRefreshDelaySec のデフォルト値を 600 (10 分) から 300 (5 分) に引き下げました。	2022 年 9 月 30 日
の AWS マネージドルールを更新しました AWS WAF	POSIX オペレーティングシステム、PHP アプリケーション、 WordPress アプリケーションのルールグループについて、このドキュメントに記載されているラベル名を修正しました。	2022 年 9 月 19 日
の新しい AWS WAFポリシールールオプション AWS Firewall Manager	AWS Firewall Manager では、AWS WAF ポリシーのデフォルトのウェブアクションに対して、カスタマイズされたウェブリクエストとレスポンスがサポートされるようになりました。	2022 年 9 月 9 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールで、IP レピュテーションのルールグループ AWS WAF が更新されました。	2022 年 8 月 30 日
AWS WAF マネージドポリシーの変更	AWSWAFFullAccessPolicy 、、AWSWAFConsoleFullAccess 、および を更新AWSWAFConsoleReadOnlyAccess してAWSWAFReadOnlyAccess 、 で保護できるリソースタイプに Amazon Cognito ユーザープールを追加しました AWS WAF。	2022 年 8 月 25 日

AWS WAF Fraud Control アカウント乗っ取り防止 (ATP)	Amazon CloudFront ディストリビューションで AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) 機能を使用できるようになりました。	2022 年 8 月 24 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが、次のルールグループ AWS WAF を更新しました: 既知の不正な入力。	2022 年 8 月 22 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールで、次のルールグループ AWS WAF が更新されました: `AWSManagedRulesATPRuleSet` 。	2022 年 8 月 11 日
に Amazon Cognito ユーザープールのサポートを追加 AWS WAF	AWS WAF ウェブ ACL を Amazon Cognito ユーザープールに関連付けることができるようになりました。この変更は、 の最新バージョンでのみ使用でき AWS WAF 、 AWS WAF Classic では利用できません。	2022 年 8 月 11 日
バージョニングされた AWS マネージドルールのルールグループのデプロイに関するセクションを追加しました	バージョン管理された AWS マネージドルールのルールグループのデプロイを文書化する新しいセクションを追加しました。このセクションには、リリース候補のデプロイ中におけるデフォルトバージョンの命名方法に関する情報が含まれています。	2022 年 7 月 29 日

Network Firewall ポリシーのログ記録を設定するための要件を更新	暗号化された Amazon S3 バケットをログの送信先として使用する Network Firewall ポリシーの要件を追加しました。	2022 年 7 月 26 日
SQLi ルールステートメントの感度レベルオプション	これで、SQL インジェクションルールステートメントの感度を上げることができます。ただし、デフォルトの感度レベルが LOW の既存のステートメントの動作は変更されません。	2022 年 7 月 15 日
Network Firewall ポリシー設定オプションを追加	Firewall Manager は、Network Firewall のファイアウォールポリシー設定で、ステートフルな評価順序とデフォルトアクションをサポートするようになりました。	2022 年 7 月 14 日
Firewall Manager セキュリティグループのポリシールール設定の更新	Firewall Manager は、プライマリセキュリティグループからレプリカセキュリティグループへのタグ配布をサポートするようになりました。	2022 年 7 月 7 日
AWS Shield ガイドの更新	Shield ガイドの情報を拡充して、Shield がイベントの軽減をどのように実行するかを説明します。	2022 年 6 月 24 日
テストおよびチューニング AWS WAF 保護に関するガイダンスを更新	テストとチューニングの一般的なガイダンス AWS WAF が更新され、最上位のトピックになりました。	2022 年 6 月 20 日

の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールで、次のルールグループ AWS WAF が更新されました: Core ルールセット (CRS)。	2022 年 6 月 9 日
新しい Firewall Manager における混乱した代理に関するガイダンス	Firewall Manager における混乱した代理問題の防止方法に関するガイダンスを追加しました。	2022 年 6 月 1 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールで、次のルールグループ AWS WAF が更新されました: Core ルールセット (CRS)。	2022 年 5 月 24 日
新しい AWS WAF リクエストコンポーネント: Headers および Cookies	これで、ウェブリクエストで cookie を検査でき、単一のヘッダーだけでなく、ウェブリクエスト内のすべてのヘッダーを検査できるようになりました。	2022 年 4 月 29 日
AWS WAF オーバーサイズの本文、ヘッダー、Cookie リクエストコンポーネントの処理	これらのコンポーネントを検査するルール内で、オーバーサイズのリクエスト本文、ヘッダー、Cookie を AWS WAF が処理する方法を指定できるようになりました。これらのコンポーネントを検査する、すでに作成済みのルールは、オーバーサイズ処理の新しい Continue オプションに一致する動作をします。	2022 年 4 月 29 日
AWS WAF Amazon S3 ログポリシーの変更	Amazon S3 ログ許可ポリシーと例を更新しました。	2022 年 4 月 12 日

Application Load Balancer AWS Shield Advanced の でアプリケーションレイヤー DDoS 自動緩和オプションが利用可能に	Shield Advanced は、Application Load Balancer 向けにアプリケーションレイヤー DDoS 自動緩和をサポートするようになりました。これにより、すべてのアプリケーションレイヤー保護のために使用できます。保護されたリソースに対するアプリケーションレイヤー DDoS 攻撃の一部であるウェブリクエストを自動的にカウントまたはブロックするように Shield Advanced を設定できます。	2022 年 4 月 8 日
マネージドルールグループの現在のデフォルトバージョン設定のインジケーターを追加しました	マネージドルールグループのバージョンリストに、現在のデフォルトバージョンが示されるようになりました。	2022 年 4 月 8 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールで、次のルールグループ AWS WAF が更新されました。AWS WAF Bot Control。	2022 年 4 月 6 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが、次のルールグループ AWS WAF を更新しました: 既知の不正な入力。	2022 年 3 月 31 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが、次のルールグループ AWS WAF を更新しました: 既知の不正な入力。	2022 年 3 月 30 日

Firewall Manager が Palo Alto Networks Cloud Next Generation Firewall (NGFW) のサポートを追加	Firewall Manager が Palo Alto Networks Cloud Next Generation Firewall (NGFW) をサポートするようになりました。	2022 年 3 月 30 日
Palo Alto Networks Cloud NGFW のサポートを に追加する AWS Firewall Manager	AWS Firewall Manager が Palo Alto Networks Cloud Next Generation Firewall (NGFW) ポリシーをサポートするようになりました。	2022 年 3 月 30 日
AWS Shield ガイドの更新	Shield ガイドの情報を拡充して、Shield がイベント検出をどのように実行するかを説明し、DDoS に対する耐性が高いアーキテクチャの例を記載しました。	2022 年 3 月 16 日
AWS Shield ガイドの更新	Shield ガイドの情報を拡充し、さまざまなセクションの編成を改善しました。主な変更点については、Shield Response Team (SRT) のサポート、 でのリソース保護 AWS Shield Advanced、DDoS イベントの可視性の各セクションを参照してください。	2022 年 2 月 24 日
Firewall Manager が Network Firewall の集約型デプロイモデルのサポートを開始	分散型および集約型デプロイモデルを使用するポリシーを設定する方法を説明する新しい手順を追加しました。	2022 年 2 月 24 日

Firewall Manager が AWS Network Firewall 一元化されたデプロイモデルのサポートを追加	分散型デプロイモデルまたは集中型デプロイモデルを使用するように AWS Network Firewall ポリシーを設定できるようになりました。分散デプロイモデルでは、Firewall Manager は、ポリシー範囲内の各 VPC にファイアウォールエンドポイントを作成および維持します。集約型デプロイモデルでは、Firewall Manager は、単一の検査 VPC でファイアウォールエンドポイントを作成および維持します。	2022 年 2 月 24 日
AWS WAF マネージドルールグループのバージョニングのサポートを に追加する AWS Firewall Manager	AWS Firewall Manager は、Firewall Manager AWS WAF ポリシーで AWS WAF マネージドルールグループのバージョニングをサポートするようになりました。	2022 年 2 月 18 日
AWS Firewall Manager マネージドポリシーの変更	`FMSServiceRolePolicy` に更新します。	2022 年 2 月 16 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールで、IP 評価リストのルールグループ AWS WAF が更新されました。	2022 年 2 月 15 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールに AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) ルールグループ AWS WAF が追加されました`AWSManagedRulesATPRuleSet` 。	2022 年 2 月 11 日

AWS WAF ガイドの組織の変更	マネージド保護の新しい上位のセクションを追加しました。CAPTCHA セクションをルールから新しいマネージド保護セクションに移動しました。ラベルセクションをルールから独自の上位のセクションに移動しました。	2022 年 2 月 11 日
AWS WAF クライアントアプリケーション統合	AWS WAF JavaScript およびモバイルクライアント APIs を使用して、クライアントアプリケーションをインテリジェントな脅威軽減 AWS マネージドルールのルールグループと統合し、検出を強化します。	2022 年 2 月 11 日
AWS WAF Fraud Control アカウント乗っ取り防止 (ATP)	新しい AWS WAF Fraud Control アカウント乗っ取り防止 (ATP) マネージドルールグループ を使用して、アカウント乗っ取りの試みを検出してブロックできますAWSManagedRulesATPRuleSet 。	2022 年 2 月 11 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが、次のルールグループ AWS WAF を更新しました: 既知の不正な入力。	2022 年 1 月 28 日
AWS WAF マネージドポリシーの変更	AWSWAFFullAccessPolicy および AWSWAFConsoleFullAccess が更新され、ログ記録の許可が修正されました。	2022 年 1 月 11 日

の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールは、コアルールセット (CRS)、SQLi データベースのルールグループ AWS WAF を更新しました。	2022 年 1 月 10 日
Firewall Manager が Shield Advanced アプリケーションレイヤー DDoS 自動緩和をサポート	Amazon CloudFront リソースの Firewall Manager Shield Advanced ポリシーに、アプリケーションレイヤー DDoS 自動緩和のサポートが含まれるようになりました。	2022 年 1 月 7 日
AWS Firewall Manager マネージドポリシーの変更	`FMSServiceRolePolicy` に更新します。	2022 年 1 月 7 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが、次のルールグループ AWS WAF を更新しました: 既知の不正な入力。	2021 年 12 月 17 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが、次のルールグループ AWS WAF を更新しました: 既知の不正な入力。	2021 年 12 月 11 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが、次のルールグループ AWS WAF を更新しました: 既知の不正な入力。	2021 年 12 月 10 日
新しい AWS Shield Advanced サービスにリンクされたロール	アプリケーションレイヤー DDoS 自動緩和機能をサポートするために `AWSServiceRoleForAWSShield` を追加しました。	2021 年 12 月 1 日

新しい AWS Shield 管理ポリシー	アプリケーションレイヤー DDoS 自動緩和機能をサポートするために AWSShieldServiceRolePolicy を追加しました。	2021 年 12 月 1 日
アプリケーションレイヤー DDoS 自動緩和オプションが AWS Shield Advanced で利用可能に CloudFront	Shield Advanced は、Amazon CloudFront ディストリビューションのアプリケーションレイヤー DDoS 自動緩和をサポートするようになりました。ディストリビューションに対する CloudFrontアプリケーションレイヤー DDoS 攻撃の一部であるウェブリクエストを自動的にカウントまたはブロックするように Shield Advanced を設定できます。	2021 年 12 月 1 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールは、コアルールセット (CRS)、Windows オペレーティングシステム、Linux オペレーティングシステム、IP 評価リストのルールグループ AWS WAF を更新しました。	2021 年 11 月 23 日
AWS Firewall Manager マネージドポリシーの変更	FMSServiceRolePolicy に更新します。	2021 年 11 月 18 日

の拡張ログ記録オプション AWS WAF	Amazon CloudWatch Logs ロググループまたは Amazon Simple Storage Service (Amazon S3) バケットにウェブ ACL トラフィックを記録できるようになりました。これらのオプションは、Amazon Data Firehose 配信ストリームへのログ記録の既存のオプションに追加されます。	2021 年 11 月 15 日
AWS WAF マネージドポリシーの変更	`AWSWAFFullAccessPolicy` および `AWSWAFConsoleFullAccess` が更新され、追加のログ記録の宛先をサポートするようになりました。	2021 年 11 月 15 日
AWS WAF 新しいCAPTCHA ルールアクションオプション	ウェブリクエストに対して CAPTCHA を実行し、必要に応じてクライアントに CAPTCHA 問題を送信するようにルールを設定できます。	2021 年 11 月 8 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールがコアルールセット (CRS) ルールグループ AWS WAF を更新しました。	2021 年 10 月 27 日
の AWS マネージドルールを更新しました AWS WAF	すべての AWS マネージドルールのルールグループでラベル付けがサポートされるようになりました。ルールの説明には、ラベルの指定が含まれます。	2021 年 10 月 25 日

Firewall Manager が Network Firewall ログフィルタリングをサポート	AWS Firewall Manager が Network Firewall ポリシーのログフィルタリングをサポートするようになりました。	2021 年 10 月 4 日
AWS Firewall Manager マネージドポリシーの変更	`FMSServiceRolePolicy` に更新します。	2021 年 9 月 29 日
正規表現一致ステートメントを追加しました	ウェブリクエストを 1 つの正規表現と照合できるようになりました。	2021 年 9 月 22 日
ルールグループ内のレートベースの AWS WAF ルール	ルールグループ内でレートベースの AWS WAF ルールを定義できるようになりました。では AWS Firewall Manager、この機能は ポリシーで AWS WAF 完全にサポートされています。	2021 年 9 月 13 日
Firewall Manager が AWS WAF ログフィルタリングをサポート	AWS Firewall Manager で、AWS WAF ポリシーのログフィルタリングがサポートされるようになりました。	2021 年 8 月 31 日
で out-of-scope リソース保護を自動的に削除する AWS Firewall Manager	AWS Firewall Manager では、ポリシーの範囲を離れるリソースから保護を自動的に削除できます。	2021 年 8 月 25 日
AWS Firewall Manager マネージドポリシーの変更	`FMSServiceRolePolicy` に更新します。	2021 年 8 月 12 日
マネージドルールグループにバージョニングを追加しました	マネージドルールグループのプロバイダーは、ルールグループをバージョニングできるようになりました。	2021 年 8 月 9 日

AWS Firewall Manager 管理者要件の変更	Firewall Manager 管理者アカウントとして、組織の管理アカウントを使用できます。これは禁止されていました。	2021 年 8 月 2 日
Firewall Manager のクォータの引き上げ	Firewall Manager ポリシーの範囲内に存在できる Amazon VPC インスタンスの数を 10 から 100 に引き上げました。	2021 年 7 月 28 日
AWS Firewall ManagerAWS Network Firewall ルートテーブルモニタリングのサポート	AWS Firewall Manager は、ルートテーブルのモニタリングをサポートするようになりました。また、ルートの設定が間違っている AWS Network Firewall ポリシーに関する修復アクションの推奨事項をセキュリティ管理者に提供します。	2021 年 7 月 8 日
AWS WAF 追加のテキスト変換オプション	テキスト変換のオプションが拡張され、リクエストを検査する前にウェブリクエストコンポーネントを適用できるようになりました。	2021 年 6 月 24 日
Firewall Manager AWS WAF ポリシーリソースの命名を変更	Firewall Manager が AWS WAF ポリシーに対して管理するウェブ ACLsルールグループ、およびログ記録の命名が変更されました。	2021 年 5 月 26 日

の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールで、IP 評価リストへのラベル付けのサポート AWS WAF が追加され、Amazon IP 評価リストのルール名のサフィックスが削除されました。	2021 年 5 月 4 日
AWS Organizations 委任管理者のサポートを追加	AWS Firewall Manager 管理者アカウントを設定すると、Firewall Manager はアカウントを Firewall Manager の AWS Organizations 委任管理者として指定するようになりました。この変更により、Firewall Manager 管理者アカウントを設定するときに、組織の管理アカウント以外のメンバーアカウントを指定する必要があります。この変更は、既存の設定には影響しません。	2021 年 4 月 30 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールに AWS WAF Bot Control ルールグループ AWS WAF が追加されました。	2021 年 4 月 1 日
ルールグループ内で個別のルールアクションを Count に設定	ルールグループ内の個々のルールアクションを Count に設定できるようになりました。ルールグループレベルにある既存の上書きの情報が修正されました。	2021 年 4 月 1 日

マネージドルールグループのスコープダウンステートメント	レートベースのステートメントと同じ方法で、マネージドルールグループでスコープダウンステートメントを使用できるようになりました。	2021 年 4 月 1 日
ログのフィルタリング	ルールのアクションとラベルに基づいて、ログに記録するウェブ ACL トラフィックをフィルタリングできるようになりました。	2021 年 4 月 1 日
AWS WAF ウェブリクエストの ラベル	一致するウェブリクエストにラベルを追加し、他のルールによって追加されたラベルと照合するルールを設定できます。	2021 年 4 月 1 日
AWS WAF ボットコントロール	Bot Control マネージドルールグループとウェブリクエストのラベル付け、スコープダウンステートメント、ログフィルタリングを組み合わせた新しい AWS WAF Bot Control 機能を使用して、ボットトラフィックをモニタリングおよび制御できます。	2021 年 4 月 1 日
Firewall Manager が Amazon Route 53 Resolver DNS Firewall ポリシーをサポート	AWS Firewall Manager は VPCs の Amazon Route 53 Resolver DNS Firewall アウトバウンド DNS トラフィックフィルタリングの一元管理をサポートします。	2021 年 3 月 31 日

カスタムリクエストとレスポンスの処理	AWS WAF がブロックしないウェブリクエストのカスタムヘッダーを含めることができ、 AWS WAF がブロックするウェブリクエストについてはカスタムレスポンスを送信できます。これは、ウェブ ACL のデフォルトアクションおよびルールアクションの設定で使用できます。	2021 年 3 月 29 日
AWS Firewall Manager マネージドポリシーの変更	`FMSServiceRolePolicy` に更新します。	2021 年 3 月 17 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールは、コアルールセット (CRS)、管理者保護、既知の不正な入力、Linux オペレーティングシステムのルールグループ AWS WAF を更新しました。	2021 年 3 月 3 日
AWS Shield マネージドポリシーの変更の追跡	Shield が AWS マネージドポリシーの変更の追跡を開始しました。	2021 年 3 月 3 日
AWS Firewall Manager マネージドポリシーの変更の追跡	Firewall Manager が AWS マネージドポリシーの変更の追跡を開始しました。	2021 年 3 月 2 日
AWS WAF マネージドポリシーの変更の追跡	AWS WAF が AWS マネージドポリシーの変更の追跡を開始しました。	2021 年 3 月 1 日

ウェブリクエストボディを解析された JSON として検査する	ウェブリクエストボディを解析およびフィルタリングされた JSON として検査するオプションを追加しました。これは、ウェブリクエストボディをプレーンテキストとして検査する既存のオプションに追加されるものです。	2021 年 2 月 12 日
Firewall Manager が AWS Network Firewall ポリシーをサポート	AWS Firewall Manager はVPCsの AWS Network Firewall ネットワークトラフィックフィルタリングの一元管理をサポートします。	2020 年 11 月 17 日
AWS Shield Advanced 保護グループのサポートを追加	保護されたリソースを論理グループにグループ化し、それらの保護をまとめて管理できるようになりました。	2020 年 11 月 13 日
AWS AppSync への のサポートを追加 AWS WAF	AWS WAF ウェブ ACL を AWS AppSync GraphQL API に関連付けることができるようになりました。この変更は、 の最新バージョンでのみ使用でき AWS WAF 、 AWS WAF Classic では利用できません。	2020 年 10 月 1 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが Windows オペレーティングシステムのルールセット AWS WAF を更新しました。	2020 年 9 月 23 日

の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが、PHP アプリケーションと POSIX オペレーティングシステムのルールセット AWS WAF を更新しました。	2020 年 9 月 16 日
更新された AWS Shield コンソール	AWS Shield は、ユーザーエクスペリエンスが向上した新しいコンソールオプションを提供します。このドキュメントのコンソールに関するガイダンスは、新しいコンソールに関するものです。	2020 年 9 月 1 日
Firewall Manager が共通セキュリティグループポリシーを更新	AWS Firewall Manager 共通セキュリティグループポリシーで、コンソールの実装を通じて Application Load Balancer と Classic Load Balancer のリソースタイプがサポートされるようになりました。新しいオプションは、共通ポリシーの [Policy scope] (ポリシーの範囲) 設定で使用できます。	2020 年 8 月 11 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールがコアルールセット AWS WAF を更新しました。	2020 年 8 月 7 日
Firewall Manager が AWS WAF ログ記録設定をサポート	AWS Firewall Manager は、AWS WAF ポリシーの一元化されたログ記録設定をサポートするようになりました。	2020 年 7 月 30 日

ウェブリクエストで IP アドレスの場所を指定する	ウェブリクエストの発信元を使用する代わりに、指定した HTTP ヘッダーからの IP アドレスを使用するオプションを追加しました。代替ヘッダーは通常 `X-Forwarded-For` (XFF) ですが、任意のヘッダー名を指定できます。このオプションは、IP セット一致、geo 一致、およびレートベースのルールカウント集約に使用できます。	2020 年 7 月 9 日
Firewall Manager がコンテンツ監査セキュリティグループポリシーを更新	AWS Firewall Manager には、マネージドルールオプション、マネージドアプリケーションとプロトコルリストを使用する 、リソース違反の詳細など、コンテンツ監査セキュリティグループポリシーの機能を拡張しました。	2020 年 7 月 7 日
Firewall Manager のマネージドリスト	AWS Firewall Manager でマネージドアプリケーションとプロトコルのリストがサポートされるようになりました。Firewall Manager はいくつかのリストを管理します。また、独自のリストを作成して管理できます。	2020 年 7 月 7 日

Firewall Manager は、共通セキュリティグループポリシーで共有 VPC をサポートします	AWS Firewall Manager は、共有 VPCsでの共通セキュリティグループポリシーの使用をサポートするようになりました。これは、範囲内のアカウントが所有する VPC での使用に加えて、行うことができます。	2020 年 5 月 26 日
の AWS マネージドルールを更新しました AWS WAF	の AWS マネージドルールに各ルールのドキュメントを追加しました AWS WAF。	2020 年 5 月 20 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが Linux オペレーティングシステムのルールグループ AWS WAF を更新しました。	2020 年 5 月 19 日
AWS WAF Classic リソースを AWS WAF (v2) に移行するためのサポートを追加	コンソールまたは API を使用して Classic AWS WAF リソースをエクスポートし、最新バージョンの に移行できるようになりました AWS WAF。	2020 年 4 月 27 日

ポリシー範囲内の AWS Organizations 組織単位のサポートを追加する	AWS Firewall Manager では、AWS Organizations 組織単位 (OUsを使用してポリシーの範囲を指定できるようになりました。OU を使用して、特定のアカウントを含めたり除外したりするほかに、範囲にアカウントを含めたり、スコープからアカウントを除外したりできます。OU を指定する方法は、OU およびその子である OU のすべてのアカウント (後から追加される子の OU およびアカウントを含む) を指定する方法と同じです。	2020 年 4 月 6 日
AWS WAF (v2) のサポートを に追加する AWS Firewall Manager	AWS Firewall Manager は、以前のバージョン AWS WAF である AWS WAF Classic に加えて、 の最新バージョンをサポートするようになりました。	2020 年 3 月 31 日
AWS Firewall Manager 共通セキュリティグループポリシーの更新	AWS Firewall Manager 共通セキュリティグループポリシーに、範囲内の Amazon EC2 インスタンス内のすべての Elastic Network Interface にポリシーを適用するオプションが追加されました。ただし、引き続きデフォルトの Elastic Network Interface にのみポリシーを適用することも選択できます。	2020 年 3 月 11 日

の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールにAWSManagedRulesAnonymousIpList ルールグループ AWS WAF が追加されました。	2020 年 3 月 6 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルールが WordPress アプリケーションとAWSManagedRulesCommonRuleSet ルールグループ AWS WAF を更新しました。	2020 年 3 月 3 日
AWS Shield Advanced 保護オプションに Amazon Route 53 ヘルスチェックを追加	Shield Advanced では、Amazon Route 53 ヘルスチェックの関連付けの使用をサポートして、脅威の検出と緩和機能の精度を向上させます。	2020 年 2 月 14 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルール AWS WAF は、SQL データベースルールグループを更新して、メッセージ URI のチェックを追加しました。	2020 年 1 月 23 日
セキュリティグループ使用状況監査ポリシーの Firewall Manager の新しいオプション	Firewall Manager には、セキュリティグループ使用状況監査ポリシーの新しいオプションがあります。セキュリティグループの未使用状態が最低何分続いたら非準拠とみなすかを設定できるようになりました。デフォルトでは、この時間設定はゼロです。	2020 年 1 月 14 日

AWS WAF ポリシーの Firewall Manager の新しいオプション	Firewall Manager には、 AWS WAF ポリシーの新しいオプションがあります。ポリシーの新しいウェブ ACL を関連付ける前に、範囲内のリソースから既存のウェブ ACL の関連付けをすべて削除することを選択できるようになりました。	2020 年 1 月 14 日
の AWS マネージドルールを更新しました AWS WAF	AWS の マネージドルール AWS WAF は、 Core ルールセットと SQL データベースルールグループのルールのテキスト変換を更新しました。	2019 年 12 月 20 日
AWS Firewall Manager と統合 AWS Security Hub	AWS Firewall Manager は、コンプライアンス違反のリソースと攻撃に関する検出結果を作成し、 に送信するようになりました AWS Security Hub。	2019 年 12 月 18 日

AWS WAF バージョン 2 のリリース	デ AWS WAF ベロッパーガイドの新しいバージョン。JSON 形式でウェブ ACL またはルールグループを管理できます。拡張された機能には、論理ルールステートメント、ルールステートメントのネスト、IP アドレスおよびアドレス範囲の完全な CIDR サポートが含まれます。ルールは AWS リソースではなくなり、ウェブ ACL またはルールグループのコンテキストにのみ存在します。既存のお客様の場合、サービスの以前のバージョンは AWS WAF Classic と呼ばれるようになりました。APIs 、 SDKs 、および CLIs、 AWS WAF Classic は命名スキームを保持し、この最新バージョンの AWS WAF は、context に応じて「V2」または「v2」が追加されて参照されます。 AWS WAF Classic で作成された AWS リソース AWS WAF にはアクセスできません。でこれらのリソースを使用するには AWS WAF、移行する必要があります。	2019 年 11 月 25 日
AWS の マネージドルールのルールグループ AWS WAF	AWS マネージドルールのルールグループを追加しました。これらは、 AWS WAF のお客様には無料です。	2019 年 11 月 25 日

AWS Firewall Manager Amazon Virtual Private Cloud セキュリティグループの サポート	Firewall Manager に Amazon VPC セキュリティグループのサポートを追加しました。	2019 年 10 月 10 日
AWS Firewall Manager のサポート AWS Shield Advanced	Firewall Manager に Shield Advanced のサポートを追加しました。	2019 年 3 月 15 日
チュートリアル: 階層型ポリシーの作成	チュートリアル : AWS Firewall Managerでの階層型ポリシーの作成に関するチュートリアルを追加しました。	2019 年 2 月 11 日
ルールグループのルールレベルの制御	ルール AWS Marketplace グループと独自のルールグループから個々のルールを除外できるようになりました。	2018 年 12 月 12 日
AWS Shield AdvancedAWS Global Accelerator 標準アクセラレーターのサポート	Shield Advanced で AWS Global Accelerator 標準アクセラレーターを保護できるようになりました。	2018 年 11 月 26 日
AWS WAF Amazon API Gateway のサポート	AWS WAF はAmazon API Gateway APIsを保護するようになりました。	2018 年 10 月 25 日
拡張 AWS シールドアドバンスト入門ウィザード	新しいウィザードでは、レートベースのルールと Amazon CloudWatch Events を作成できます。	2018 年 8 月 31 日
AWS WAF logging	ログ記録を有効にして、ウェブ ACL で分析されるトラフィックに関する詳細情報を取得します。	2018 年 8 月 31 日

条件内のパラメータのクエリをサポート	条件を作成するときに、特定のパラメータに対するリクエストを検索できるようになりました。	2018 年 6 月 5 日
Shield Advanced 使用開始ウィザード	AWS Shield Advanced にサブスクライブするための新しい合理化されたプロセスを導入しました。	2018 年 6 月 5 日
許可される CIDR 範囲が拡張されました	IP 一致条件を作成するときに、 は IPv4 アドレス範囲 /8、および /16 から /32 までの任意の範囲をサポートする AWS WAF ようになりました。	2018 年 6 月 5 日

2018 年よりも前の更新

次の表は、「AWS WAF デベロッパーガイド」の各リリースでの 2018 年より前の重要な変更を示しています。

変更	API バージョン	説明	リリース日
更新	2016-08-24	AWS Marketplace ルールグループ	2017 年 11 月
更新	2016-08-24	Elastic IP 向けの Shield Advanced サポート	2017 年 11 月
更新	2016-08-24	グローバル脅威ダッシュボード	2017 年 11 月
更新	2016-08-24	DDoS 対応ウェブサイトのチュートリアル	2017 年 10 月

変更	API バージョン	説明	リリース日
更新	2016-08-24	Geo 条件と正規表現条件	2017 年 10 月
更新	2016-08-24	レートベースのルール	2017 年 6 月
更新	2016-08-24	再編成	2017 年 4 月
更新	2016-08-24	DDOS 保護とアプリケーションロードバランサーのサポートに関する情報を追加しました。	2016 年 11 月
新機能	2015-08-24	すべての API 呼び出しを AWS WAF through に記録できるようになりました。 AWS このサービスは AWS CloudTrail、アカウントの API 呼び出しを記録し、ログファイルを S3 バケットに配信します。 CloudTrail ログを使用して、セキュリティ分析、 AWS リソースへの変更の追跡、コンプライアンス監査に役立てることができます。統合することで AWS WAF 、 AWS WAF API に対してどのようなリクエストが行われたか、各リクエストが行われたソース IP アドレス、リクエストの実行者、実行日時などを判断できます。 CloudTrail すでに使用している場合は AWS CloudTrail、 CloudTrail ログに AWS WAF API 呼び出しが表示されるようになります。 CloudTrail アカウントで有効化していない場合は、 CloudTrail から有効化できますAWS Management Console。有効にしても追加料金はかかりませんが CloudTrail、Amazon S3 と Amazon SNS の使用には標準料金が適用されます。	2016 年 4 月 28 日

変更	API バージョン	説明	リリース日
新機能	2015-08-24	これで、クロスサイトスクリプティングまたは XSS と呼ばれる、悪意のあるスクリプトが含まれていると思われるウェブリクエストを許可、ブロック、またはカウントできるようになりました。 AWS WAF 攻撃者は、ウェブアプリケーションの脆弱性を悪用するために、悪意のあるスクリプトをウェブリクエストに挿入することがあります。詳細については、「クロスサイトスクリプティング攻撃ルールステートメント」を参照してください。	2016 年 3 月 29 日
新機能	2015-08-24	このリリースでは、 AWS WAF 以下の機能が追加されました。 • クエリ文字列や URI など、リクエストの特定の部分の長さに基づいてウェブリクエストを許可、ブロック、 AWS WAF またはカウントするように設定できます。詳細については、「サイズ制約ルールステートメント」を参照してください。 • AWS WAF リクエスト本文の内容に基づいてウェブリクエストを許可、ブロック、またはカウントするように設定できます。これは、HTTP リクエストボディとしてウェブサーバーに送信する追加のデータ (フォームからのデータなど) を含むリクエストの部分です。この機能は、文字列一致条件、SQL インジェクション一致条件、上記の新しいサイズ制約条件に適用されます。詳細については、「ウェブリクエストコンポーネントの仕様と処理」を参照してください。	2016 年 1 月 27 日
新機能	2015-08-24	AWS WAF コンソールを使用して、ウェブ ACL CloudFront を関連付けるディストリビューションを選択できるようになりました。詳細については、「ウェブ ACL とディストリビューションの関連付けまたは関連付けの解除」を参照してください。 CloudFront	2015 年 11 月 16 日

変更	API バージョン	説明	リリース日
初回リリース	2015-08-24	これは「AWS WAF デベロッパーガイド」の初回リリースです。	2015 年 10 月 6 日

AWS 用語集

最新の AWS 用語については、「 AWS の用語集 リファレンス」のAWS 「 用語集」を参照してください。

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。