[AWS Black Belt Online Seminar] AWS WAF アップデート

サービスカットシリーズ

Solutions Architect
岡　豊

2020/03/24

AWS 公式 Webinar
https://amzn.to/JPWebinar

過去資料
https://amzn.to/JPArchive

自己紹介

岡　豊（おか　ゆたか）

所属：
アマゾン ウェブ サービス ジャパン
Edge サービス担当ソリューションアーキテクト

好きなAWSのサービス：
Amazon CloudFront
AWS WAF
AWS Certificate Manager

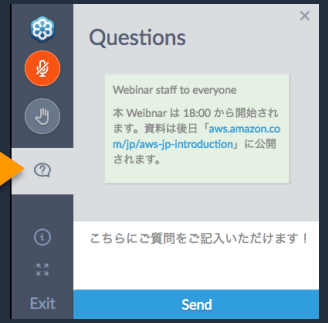
AWS Black Belt Online Seminar とは

「サービス別」「ソリューション別」「業種別」のそれぞれのテーマに分かれて、アマゾン ウェブ サービス ジャパン株式会社が主催するオンラインセミナーシリーズです。

質問を投げることができます！

- 書き込んだ質問は、主催者にしか見えません
- 今後のロードマップに関するご質問は
  お答えできませんのでご了承下さい

① 吹き出しをクリック
② 質問を入力
③ Sendをクリック



Twitter ハッシュタグは以下をご利用ください
#awsblackbelt

内容についての注意点

- 本資料では2020年03月24日時点のサービス内容および価格についてご説明しています。最新の情報はAWS公式ウェブサイト(http://aws.amazon.com)にてご確認ください。

- 資料作成には十分注意しておりますが、資料内の価格とAWS公式ウェブサイト記載の価格に相違があった場合、AWS公式ウェブサイトの価格を優先とさせていただきます。

- 価格は税抜表記となっています。日本居住者のお客様には別途消費税をご請求させていただきます。

- AWS does not offer binding price quotes. AWS pricing is publicly available and is subject to change in accordance with the AWS Customer Agreement available at http://aws.amazon.com/agreement/. Any pricing information included in this document is provided only as an estimate of usage charges for AWS services based on certain information that you have provided. Monthly charges will be based on your actual use of AWS services, and may vary from the estimates provided.

本セッションの目的

2019 年にAWS WAFが大幅にアップデートされました

AWS WAFを以前からご利用の方には、そのアップデートで何が変わったのか、また、AWS WAFをまだお使いでない方には、AWS WAFの全体像を把握いただき、何が出来るのかをご理解いただく

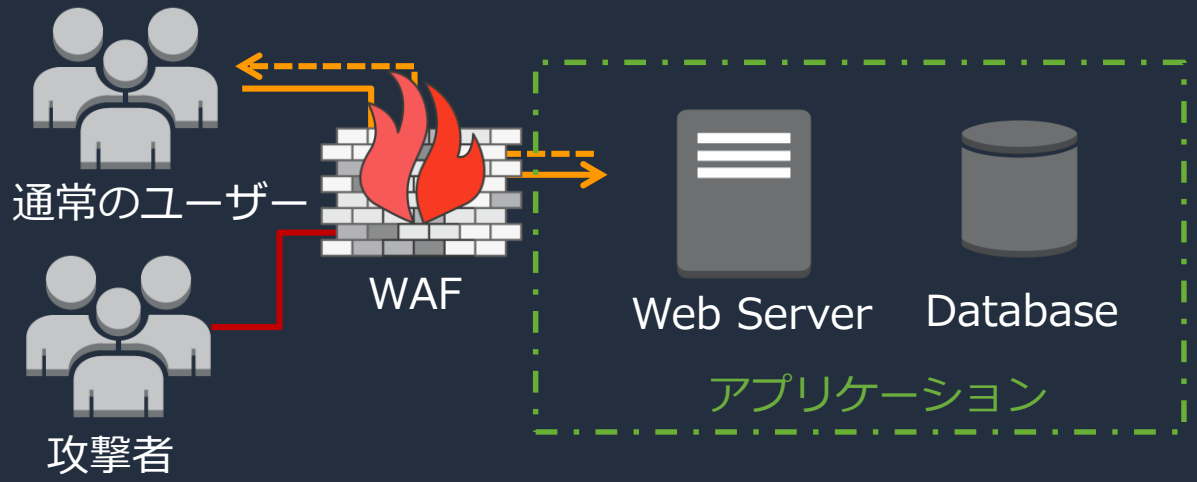
本日のアジェンダ

1. Web Application Firewall(WAF)とは
2. AWS WAFの機能
3. アップデートによる主な変更点
4. AWS WAFの詳細
5. AWS Managed Rules for AWS WAF
6. Web ACL作成の流れ
7. ログとモニタリング
8. AWS WAF利用のための考慮事項
9. 料金体系
10. まとめ

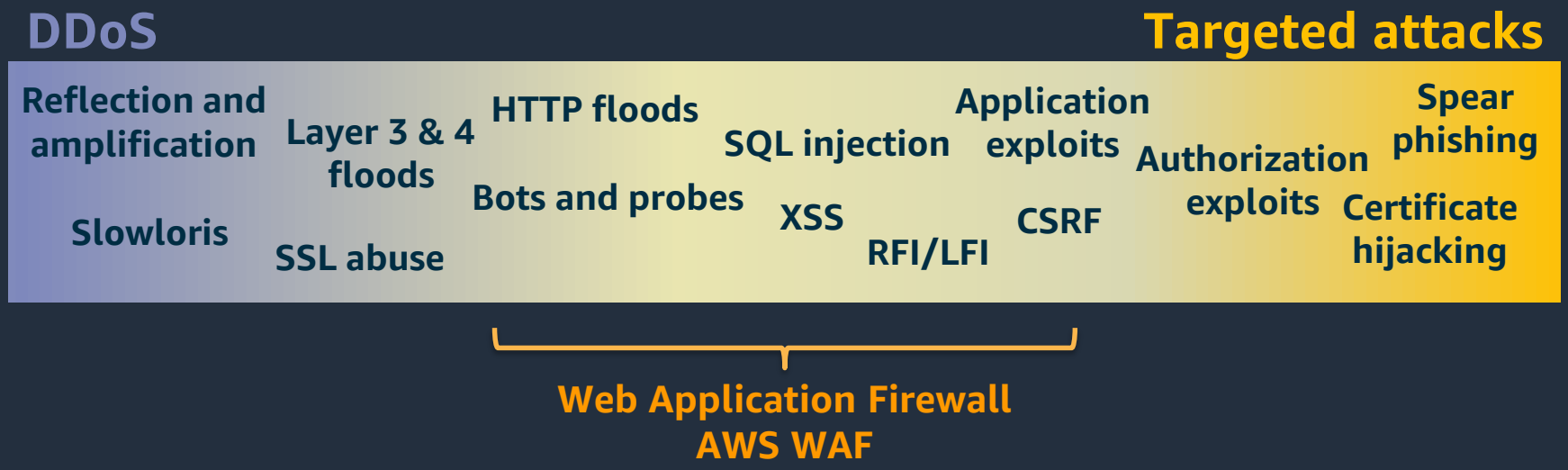
Web Application Firewall (WAF)とは

ウェブアプリケーションの通信内容を検査し、不正なアクセスを遮断するルールセットを持つセキュリティ対策

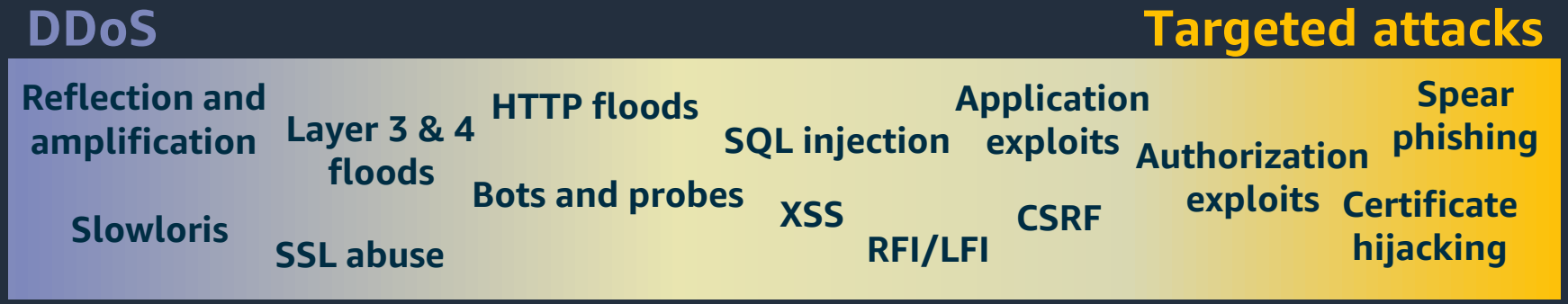
ウェブアプリケーションの脆弱性を悪用した攻撃などからウェブアプリケーションを保護することが目的



様々な攻撃手法



様々な攻撃手法



- Amazon CloudFront
- Elastic Load Balancing
- AWS Shield Standard
- AWS Shield Advanced

Web Application Firewall
AWS WAF

- Amazon Inspector
- Amazon Macie
- Amazon Certificate Manager (ACM)
- AWS Marketplace: IDS/IPS, anti-malware

ウェブアプリケーションの欠陥に対する攻撃を軽減

- WAFは根本的な欠陥を修正するものではなく、それらを悪用する能力を制限
- 様々なHTTPのリクエストパターンを検証
- 攻撃の変化に対応するためにルール構成を迅速に変更する機能が必要

本日のアジェンダ

1. Web Application Firewall(WAF)とは
2. AWS WAFの機能
3. アップデートによる主な変更点
4. AWS WAFの詳細
5. AWS Managed Rules for AWS WAF
6. Web ACL作成の流れ
7. ログとモニタリング
8. AWS WAF利用のための考慮事項
9. 料金体系
10. まとめ

AWS WAFが提供する機能

悪意のあるリクエストのブロック

- SQLインジェクション
- クロスサイトスクリプト
- AWS、またはパートナー提供のマネージドルール

カスタムルールに基づいたWebトラフィックのフィルタ

- Rate-based rules
- IP & Geo-IP filters
- 正規表現パターン、文字列
- サイズ制限
- アクション：許可/拒否

モニタリングとチューニング

- Amazon CloudWatch
- メトリクス/アラーム
- サンプルログ
- Full logs
- カウントアクションモード（検知モード）

AWS WAFの特徴

容易なデプロイ

迅速なインシデントレスポンス

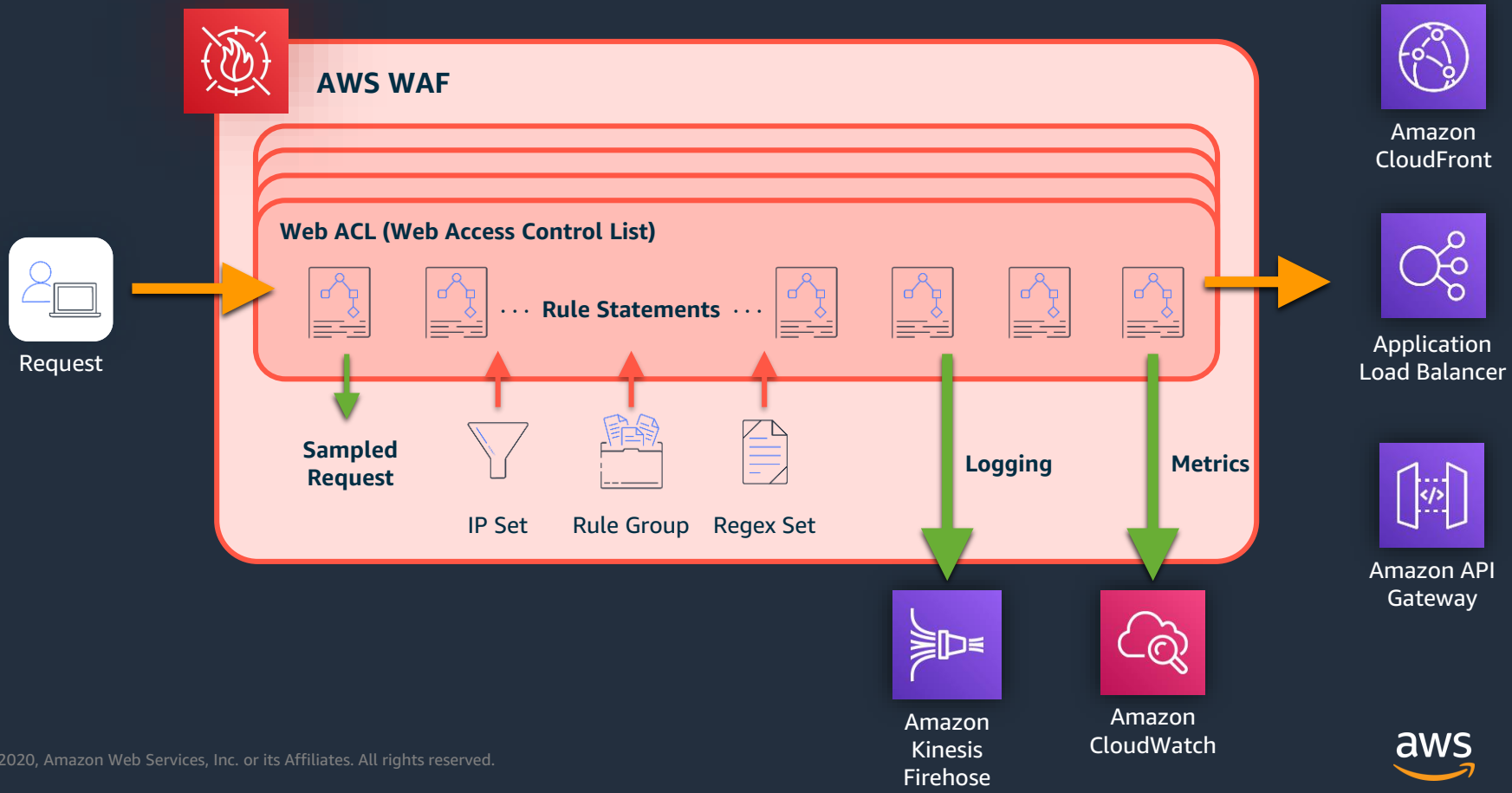
フルAPIサポート

セキュリティオートメーション

廉価

マネージドルール

AWS WAFの全体像



AWS WAF 機能拡張の歴史

- **2015年10月 AWS WAF リリース**
- 2015年12月 CloudFormationに対応

- 2017年5月 AWS WAF on ALBがAWS CloudTrailに対応
- 2017年5月 IP setの上限を1000から10000へ増加
- 2017年5月 AWS WAF on ALBがAWS CloudFormationに対応
- 2017年6月 HIPAA コンプライアンスプログラムの対象となる
- 2017年6月 レートベースのルールに対応
- 2017年10月 正規表現のルールに対応
- 2017年10月 地域別の制御に対応
- 2017年11月 パートナーマネージドルールに対応

2015 2016 2017

- 2016年1月 HTTP Bodyの検査に対応
- 2016年1月 リクエストサイズの長さ制限の条件に対応
- 2016年3月 Cross-Site Scripting (XSS)ルール条件に対応
- 2016年4月 AWS CloudTrailに対応
- 2016年7月 PCI DSS 3.2 Merchant Level 1にAWS WAFが含まれる
- 2016年10月 IPv6に対応
- 2016年12月 Application Load Balancer (ALB)に対応

※AWS WAF: What's newから抜粋
https://aws.amazon.com/waf/resources/

AWS WAF 機能拡張の歴史(続き)

- 2018年2月 AWS Config が AWS WAF ルールグループをサポート
- 2018年6月 クエリ文字列とのパターンマッチングを強化
- 2018年6月 Non-octet CIDR boundariesをサポート
- 2018年8月 AWS WAF Full logsが新たに利用可能に
- 2018年11月 AWS API Gatewayに対応

- 2020年3月 AWS マネージドルールにAnonymous IP リスト追加

- 2019年8月 レートベースのルールに低しきい値を設定可能に
- 2019年11月 AWS Managed Rules for AWS WAFリリース
- 2019年11月 AWS WAF APIがアップデート(APIv2)
- 2019年12月 AWS WAF Full Logsを拡張

※AWS WAF: What's newから抜粋
https://aws.amazon.com/waf/resources/

本日のアジェンダ

1. Web Application Firewall(WAF)とは
2. AWS WAFの機能
3. アップデートによる主な変更点
4. AWS WAFの詳細
5. AWS Managed Rules for AWS WAF
6. Web ACL作成の流れ
7. ログとモニタリング
8. AWS WAF利用のための考慮事項
9. 料金体系
10. まとめ

従来のAWS WAFはAWS WAF Classicへ名称変更

2019 年11 月末のアップデートにより、従来のAWS WAFはAWS WAF Classicへ名称を変更しました。AWSマネージメントコンソールの左のメニューから切り替え可能です。

AWS WAF 直近の主なアップデート

AWS WAFの新しいAPIの仕組み

- namespace: “wafv2”が付与される
- “waf” や “waf-regional”という区別なく、APIでの操作が可能

ルールの記述方法も拡張

- これまで異なっていたルールタイプ毎の記述方法が統一化
- JSON フォーマットのドキュメントベースのルール記述
- 一つのJSONファイル内にすべてのルールを記載し、UpdateWebACL を呼ぶだけで反映

WAF Capacity Unit (WCU)を新たに利用

- Web ACLあたりの上限10 ルールの制限は撤廃
- その他諸々の制限の緩和 (例:フィルタ数の上限など)

aws

AWS WAF 直近の主なアップデート

コンソール画面の変更
- コンソールをシンプル化し、より簡単に利用できるように

ルール記述の柔軟化
- OR 、複数のトランスフォーム、CIDR表記

ビルトインのマネージドルール: AWS Managed Rules for AWS WAF
- AWS Threat Research Team が作成及びメンテナンスを実施
- OWASP Top 10 や IP reputation listなどが含まれる

AWS WAF Full logsの拡張
- 誤検知発生時のトラブルシューティングに有用となる新しいログフィールドの追加

AWS WAFとAWS WAF Classicの主な違い

機能	WAF Classic	新 AWS WAF
AWSマネージドルールグループ	No	Yes
AWS パートナーマネージドルールグループ	Yes	Yes
Web ACL毎のルール上限	10	WAF Capacity Unit (WCU)の上限 1500 WCU
Web ACL毎のルールグループ数の上限	2	WAF Capacity Unit (WCU)の上限 1500 WCU
ルールの論理条件	AND and NOT	AND, OR, and NOT
API の利用	CloudFrontとALB,API GWで異なるネームスペースを指定 (namespace “waf”) (namespace “waf-regional”)	ネームスペースは“WAFv2”に統一
IP setのCIDRサポート	/8, /16-/32	/1-/32

本日のアジェンダ

1. Web Application Firewall(WAF)とは
2. AWS WAFの機能
3. アップデートによる主な変更点
4. AWS WAFの詳細
5. AWS Managed Rules for AWS WAF
6. Web ACL作成の流れ
7. ログとモニタリング
8. AWS WAF利用のための考慮事項
9. 料金体系
10. まとめ

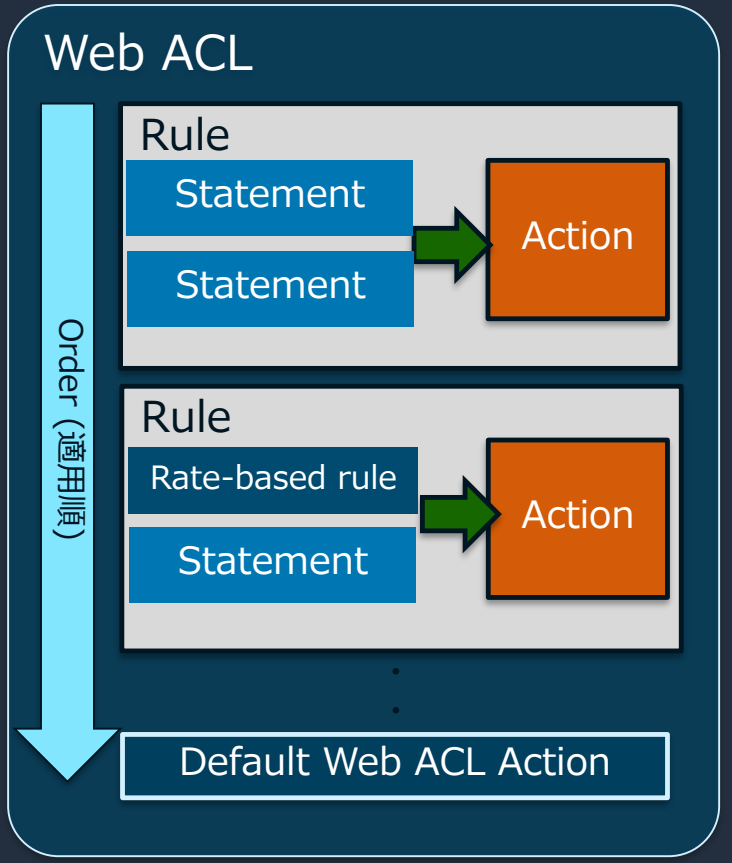
AWS WAFのコンポーネント

- Web ACL (Web Access Control List)
- Rules
- Statement
- Rate-based rule
- Actions
- Managed Rule
- Rule Group
- AWSリソース （CloudFront, ALB, API Gateway）
- レポート
- AWS WAF Full logs

Web ACL

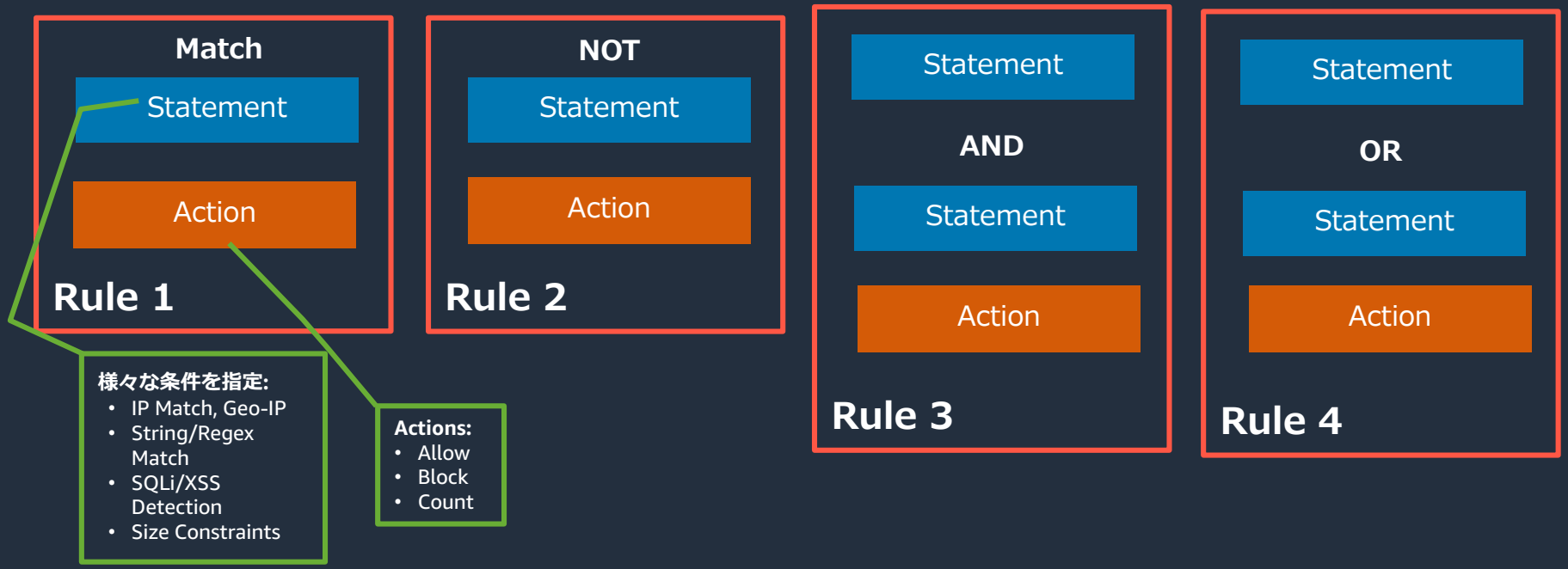
Web ACL を作成し、ルールを追加して攻撃に対する戦略を定義します。ルール内のステートメントにより、リクエストを検査するための条件が定義され、アクションによって条件に一致したリクエストの処理方法が指定されます。

- Web Capacity Unit(WCU)がWeb ACLで設定できるルールの上限となります。(WCUについては後述)
- Web ACL内のルールの優先度を指定可能です。
- Web ACL全体のデフォルトActionを指定(どのルールにも一致しない場合のアクション）します。



Ruleの中身

リクエストに対する検査方法と、リクエストが検査条件に一致した場合の処理を定義



ステートメントで定義可能な条件の種類

一致条件	内容
リクエスト発生国の一致	リクエストの送信元の国を検査
送信元IPアドレスの一致	IPアドレスがマッチしているかどうかのリストを設定（CIDRで設定）
サイズの制約	サイズの制約にマッチしているかどうか
SQL injection	SQLインジェクション攻撃かどうか
XSS injection	クロスサイトスクリプティング攻撃かどうか
文字列の一致	文字列がマッチしているかどうか Header / HTTP Method / Query String / URI / Body 以下のMatchに対応 Contains / Exact / Begins with / Ends with / Contains word
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

リクエストの検査をする対象コンポーネントの種類

コンポーネント	内容
リクエストヘッダー	特定のリクエストヘッダー。このオプションでは、[Header type (ヘッダータイプ)] フィールドでヘッダーの名前（User-Agent、Referer など）も選択
HTTPメソッド	HTTP メソッドを検査する
クエリ文字列	URL 内で "? "文字の後に続く部分 (ある場合）XSS一致条件の場合は、[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)] を選択することを推奨
単一クエリパラメータ	指定されたパラメータの値を検査する
すべてのクエリパラメータ	クエリ文字列に含まれるすべてのパラメータの値を検査する
URI	URL 内でリソースを識別する部分 (例: images/daily-ad.jpg)
Body	リクエストヘッダーの直後に続くリクエストボディの部分。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る

利用可能なテキスト変換の種類

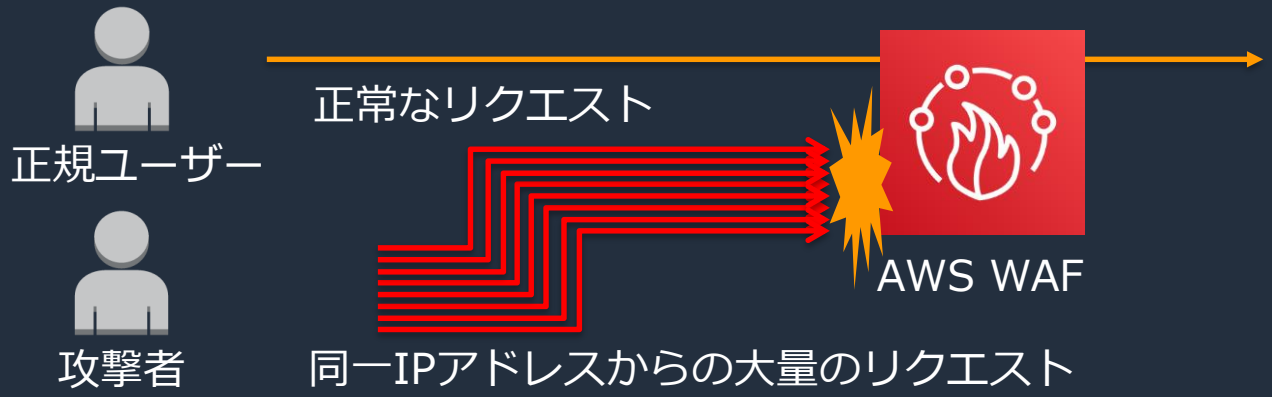
テキスト変換	内容
None	変換せず、そのままリクエストを検査
Convert to lowercase	大文字の A-Z を小文字の a-z に変換
HTML decode	HTML エンコードされた文字をエンコードされていない文字に置き換え
空白の正規化	複数のスペースを 1 つのスペースに置換、以下の文字を空白文字（10 進数の 32）に置き換え ¥f、フォームフィード、10 進数の 12 ¥t、タブ、10 進数の 9 ¥n、改行、10 進数の 10 ¥r、キャリッジリターン、10 進数の 13 ¥v、垂直タブ、10 進数の 11 改行なしスペース、10 進数 160
Simplify command line	攻撃者がオペレーティングシステムのコマンドを挿入した際、異常なフォーマットを使用して、コマンドの一部またはすべてを偽装するのを防ぐ
URL decode	URL エンコードされたリクエストをデコード

論理ルールステートメントの種類

論理ステートメント	内容
ANDロジック	ステートメントを AND ロジックと組み合わせる (API ステートメント – AndStatement)
NOTロジック	ステートメントの結果を否定 (API ステートメント – NotStatement)
ORロジック	ステートメントを OR ロジックと組み合わせる (API ステートメント – OrStatement)

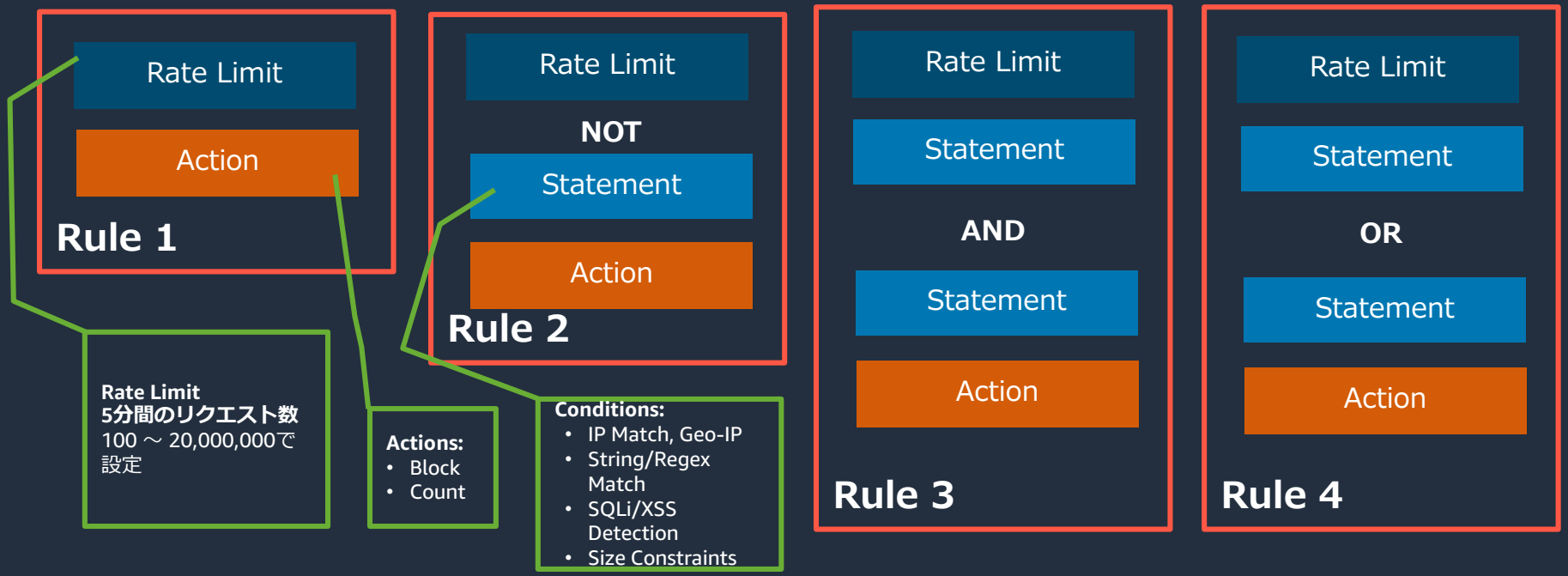
レートベースのルール

- 5 分間あたりの同一IPアドレスからのリクエスト数が設定された閾値を超過したら、Block/Countする。
- 5 分間あたりの閾値の設定範囲は、100 ～ 20,000,000 で設定
- ルール作成時にRate-based Ruleを選択
- 全てのリクエストを対象にするか、ステートメント内の条件に一致したリクエストだけを対象にするかを選択可能



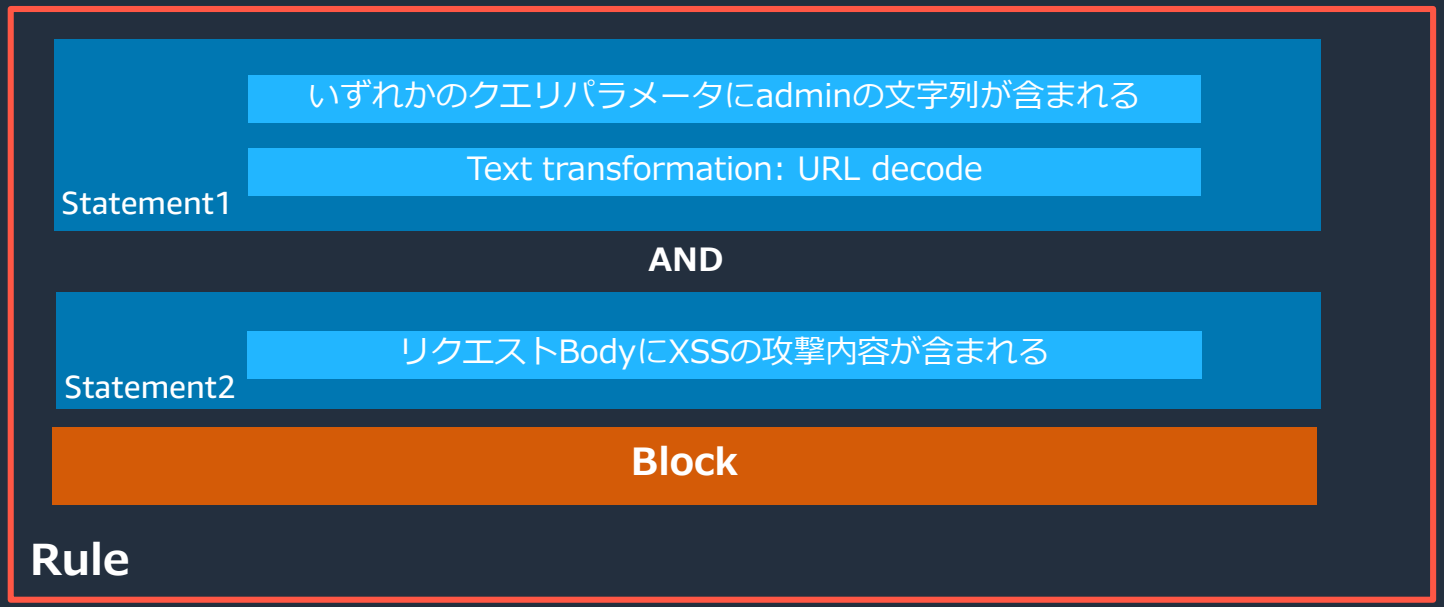
Ruleの中身 (Rate-based ruleのパターン)

リクエストレートの制限を適用する条件を指定　(All, NOT, OR, AND)



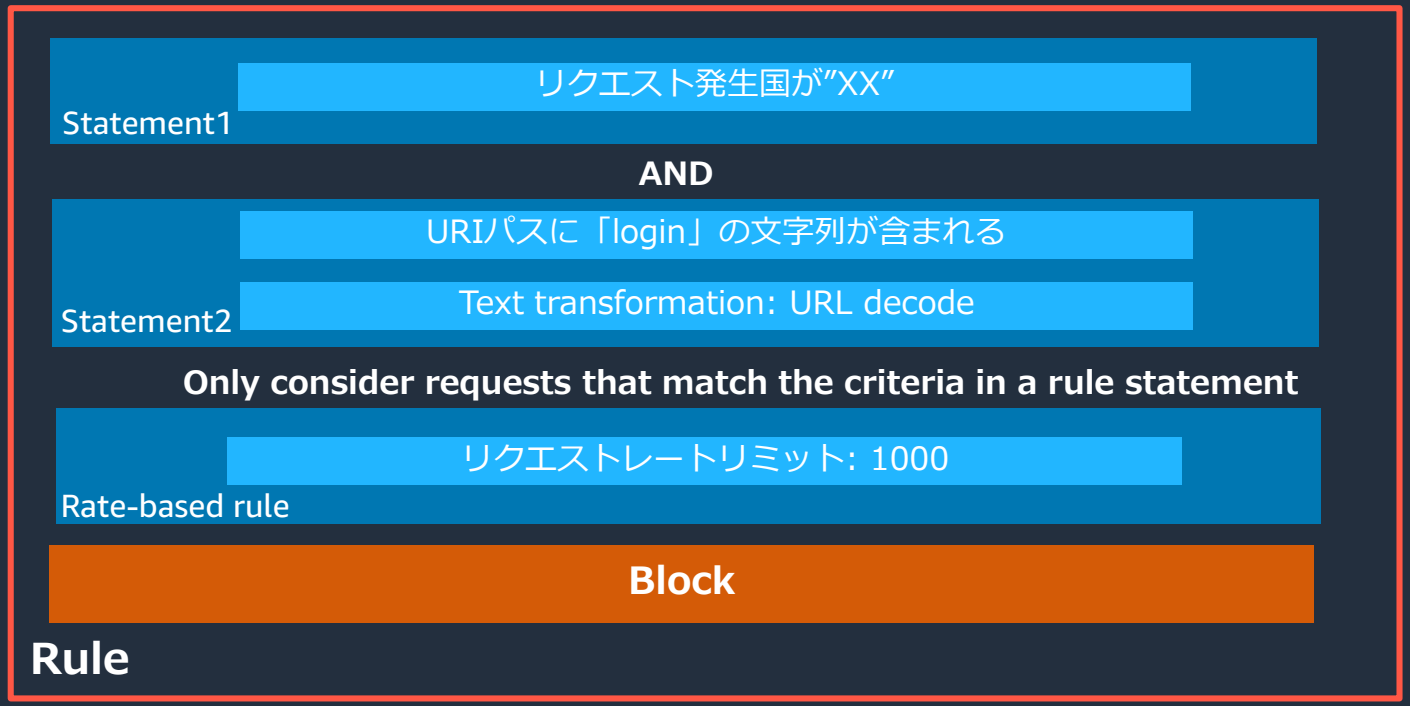
ルール設定例 1

クエリパラメータにadminの文字列を含むリクエストに対し、リクエストBodyがクロスサイトスクリプティング攻撃かどうかをチェックし、ブロック



ルール設定例２

特定の国からのリクエストのURIパスに「login」を含む場合において、リクエストのレートが1,000 を超えたらブロックする

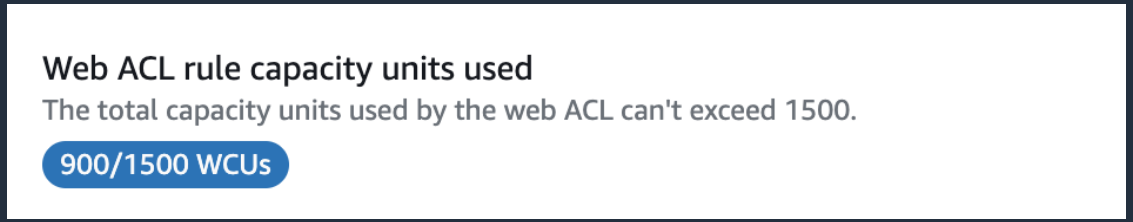


WAF Capacity Unit (WCU)とは

WAFのルールに対する処理コストの考え方。それぞれのルールの中身に応じて処理コストを計上、その合計がWeb ACLの Capacity を超えない範囲でルールを登録可能

注意点：

- Web ACLのWCUの上限は 1500 となる
- 文字列のマッチ条件の処理内容によって、WCUの使用量が異なる
- 正規表現のルール数の上限はこれまで通り10 個まで (Regex pattern sets)
- Web ACLの画面に合計利用WCUの値が表示される



Web ACLで消費されるWCUリソース

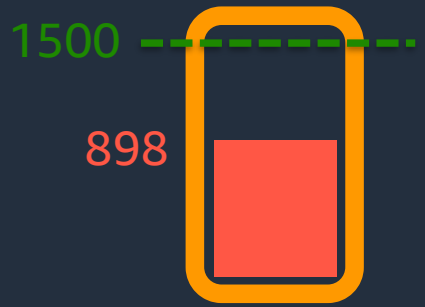
より複雑なルールはより多くのWCUを利用

Match Statement	Description	WCUs
IP Set Match	リクエストの送信元を一連の IP アドレスおよびアドレス範囲と比較	1
Regex Pattern Set	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	25 per pattern set
Size Constraint	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	1
SQLi Attack	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	20
String Match	指定されたリクエストコンポーネントと文字列を比較	Depends on the type of match
XSS Scripting Attack	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	40

ルールグループも作成が可能（含まれるルールの合算WCUが消費される）

Web ACLにおけるWCUの計算例

- 文字列マッチルール
    - Bodyに対する 10 個の文字列フィルタ→ 10 x 10 WCU = 100 WCU
    - Bodyに対する空白の正規化 → 10 WCU
- SQLi 検出ルール
    - URL, cookie header, と bodyに適用 → 3 x 20 WCU = 60 WCU
- 正規表現ルール
    - 10 個の正規表現を含むパターンセット → 1 x 25 WCU = 25 WCU
- 3 つのIPセットルール
    - 3 つの IPセット (10,000 IPまで指定可能) → 3 x 1 = 3 WCU
- AWS Managed Rules
    - Core ruleset → 700 WCU



合計 WCU :898 WCU

WCU 残:1,500 - 898 = 602 WCU

ルールグループ

- 個別のルール以外にも複数のルールを組み合わせたルールグループも自分で作ることが可能
- Web ACL内でグループ単位で優先度を設定したり、グループ内の個別ルールのアクションをオーバーライドすることが可能
- ルールグループ作成時にWCUの値を設定
    - 将来的にルールグループをメンテナンスしていくときにWCUの上限を超えないように事前に予約するための機能
    - 含まれるルール自体はWCU22 だがルールグループの設定は 200 などに設定できる（包含しているものより大きい値が設定可能）
    - 1 回設定したら変えられない

本日のアジェンダ

1. Web Application Firewall(WAF)とは
2. AWS WAFの機能
3. アップデートによる主な変更点
4. AWS WAFの詳細
5. AWS Managed Rules for AWS WAF
6. Web ACL作成の流れ
7. ログとモニタリング
8. AWS WAF利用のための考慮事項
9. 料金体系
10. まとめ

AWS Managed Rules for AWS WAF (AMR)とは

AWS WAFに組み込み可能なビルトインルールセット

- AWS Threat Research Team (TRT)が作成及びメンテナンスを実施
- OWASP Top 10 をメインに対応を実施

追加費用は不要

- WAF Capacity Unit (WCU)は消費するがそれ以外の追加費用は不要

Category	Ruleset	Description
CRS	Core Ruleset	Based on OWASP Top 10
EXR	Admin Protection	Blocks common administrative access
EXR	SQL DB	Predefined SQL injection detection
EXR	Linux	Linux based path traversal attempts
EXR	Known Bad Inputs	Well known bad request indicators
EXR	PHP	PHP specific exploits
EXR	WordPress	WordPress specific exploits
EXR	Posix	Posix based path traversal attempts
EXR	Windows	Windows based path traversal attempts
IP List	AWS IP Reputation List	Blocks IP that is known to have bot activities
IP List	Anonymous IP list	VPN, proxies, Tor nodes, and hosting providers.

AWS Managed Rules for AWS WAF ルールー覧

（2020/3/24現在）

Name	Description	Capacity Unit
Admin protection	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る。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軽減したい場合に有効	100
Amazon IP reputation list	このグループには、Amazon の内部脅威インテリジェンスに基づくIPレビュテーションリストが含まれる。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有効。これらの IP アドレスをブロックすることで、ボットを軽減し、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軽減できる。	25
Anonymous IP list	このグループには、匿名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ためのルールが含まれる。 これには、VPN、プロキシ、Torノード、およびホスティングプロバイダからの要求が含まれる。アプリケーションからアイデンティティを隠そうとしている可能性のあるビューアを除外する場合に有効。	50
Core rule set	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る。これらは、OWASP の出版物や多くの共通脆弱性識別子 (CVE) に記載されているものを含め、幅広い脆弱性の悪用から保護するのに有用。すべての AWS WAF ユースケースでこのルールグループを使用することの検討を推奨	700
Known bad inputs	既知の不正な入力ルールグループには、無効且つ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る。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軽減可能	200
Linux operating system	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る。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。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る。また、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る。	200

AWS マネージドルール ルールグループ のリスト

https://docs.aws.amazon.com/ja_jp/waf/latest/developerguide/aws-managed-rule-groups-list.html

aws

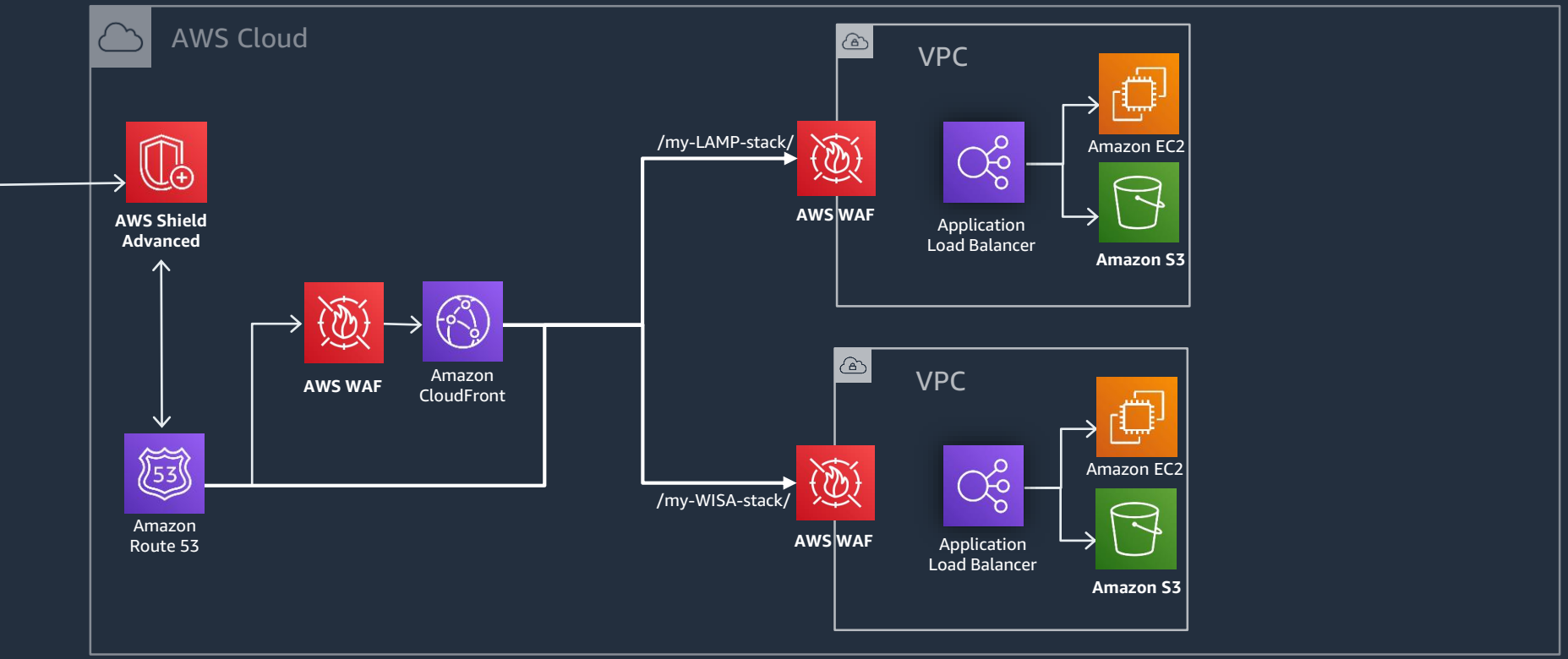
AWS Managed Rules for AWS WAF ルールー覧

（2020/3/24現在）

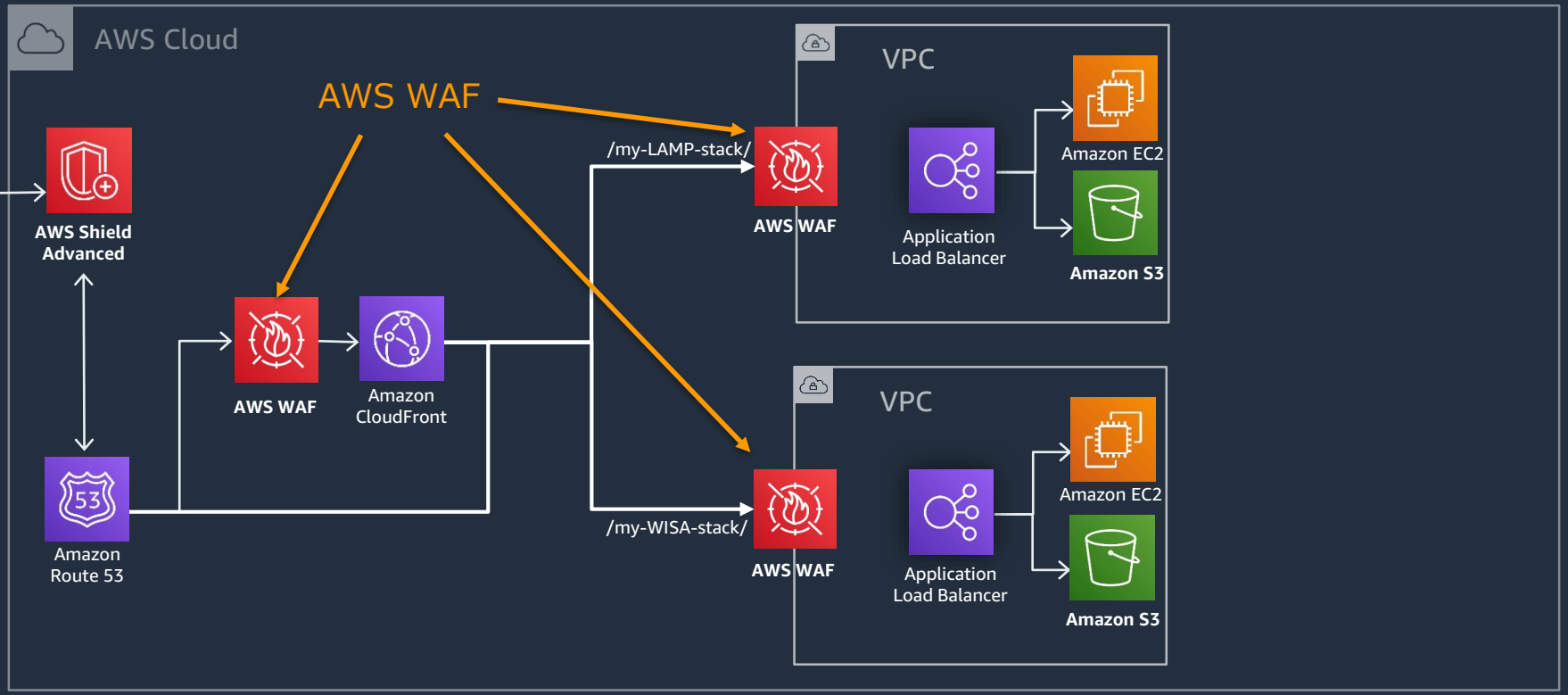
Name	Description	Capacity Unit
PHP application	PHP アプリケーションルールグループには、安全でない PHP 関数のインジェクションなど、PHP プログラミング言語の使用に固有の脆弱性の悪用に関連するリクエストパターンをブロックするルールが含まれる。これにより、攻撃者が許可されていないコードまたはコマンドをリモートで実行できる脆弱性の悪用を防ぐ。アプリケーションが連結するサーバーに PHP がインストールされている場合は、このルールグループを使用することの検討を推奨	100
POSIX operating system	POSIX オペレーティングシステムルールグループには、ローカルファイルインクルージョン (LFI) 攻撃を含む 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る。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可能。アプリケーションの一部が、Linux、AIX、HP-UX、macOS、Solaris、FreeBSD、OpenBSD、その他多くのものを含む POSIX または POSIX と同等の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使用することの検討を推奨。	100
Windows operating system	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る。これにより、攻撃者が不正なコマンドを実行したり、悪意のあるコードを実行したりする脆弱性の悪用を防ぐことが可能。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使用することの検討を推奨。	200
SQL database	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る。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可能。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使用することの検討を推奨。	200
WordPress application	WordPress アプリケーショングループには、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レル。WordPress を実行している場合は、このルールグループを使用することの検討を推奨。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る。	100

aws

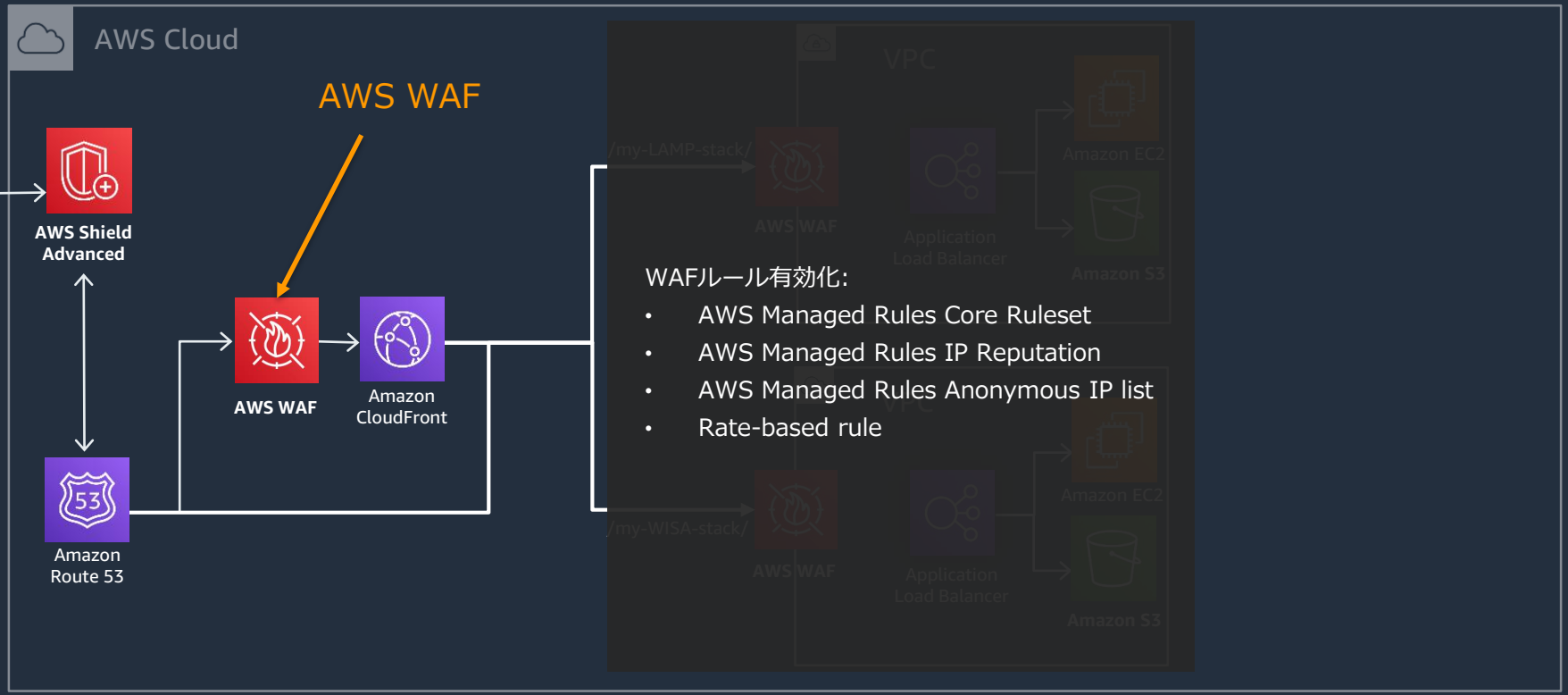
ユースケース: AWS Managed Rulesを利用した多段防御



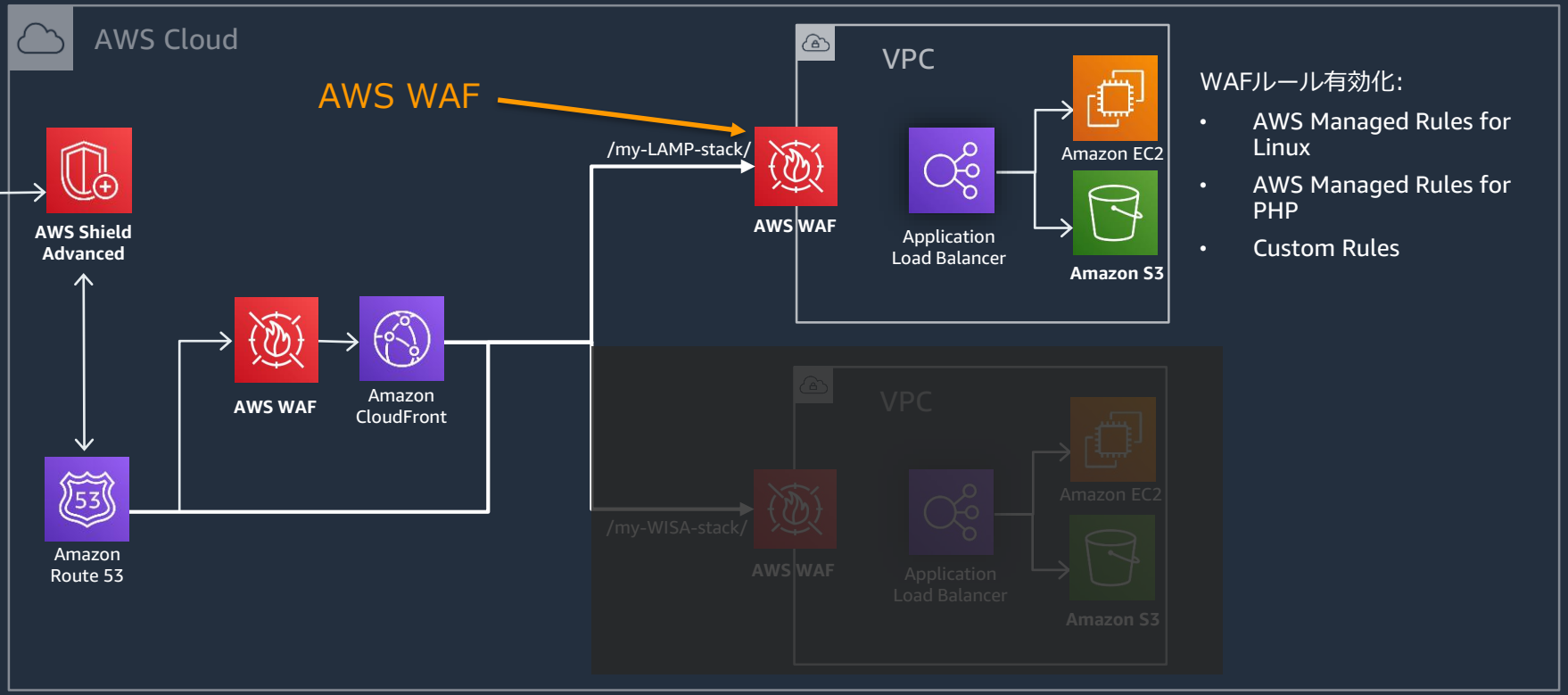
ユースケース: AWS Managed Rulesを利用した多段防御



ユースケース: AWS Managed Rulesを利用した多段防御



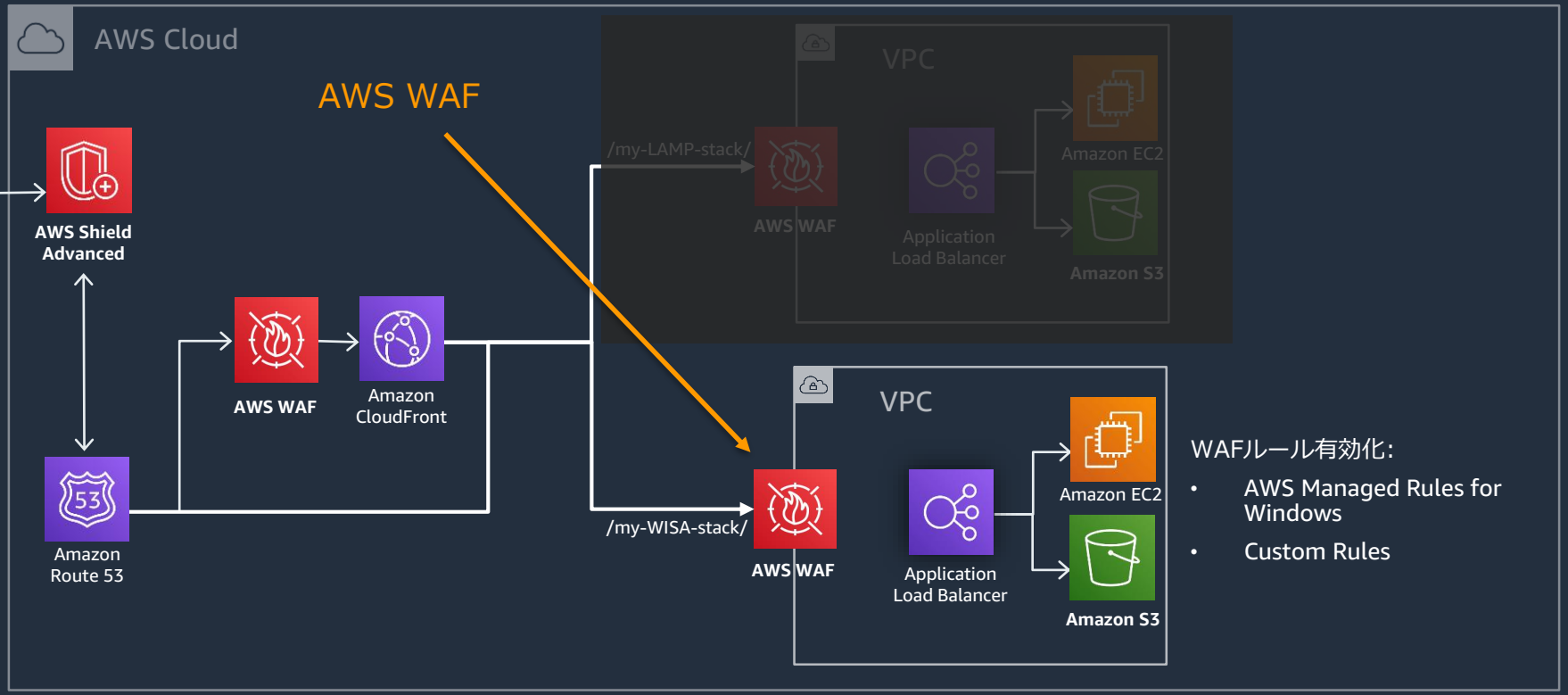
ユースケース: AWS Managed Rulesを利用した多段防御



WAFルール有効化:

- AWS Managed Rules for Linux
- AWS Managed Rules for PHP
- Custom Rules

ユースケース: AWS Managed Rulesを利用した多段防御



WAFルール有効化:

- AWS Managed Rules for Windows
- Custom Rules

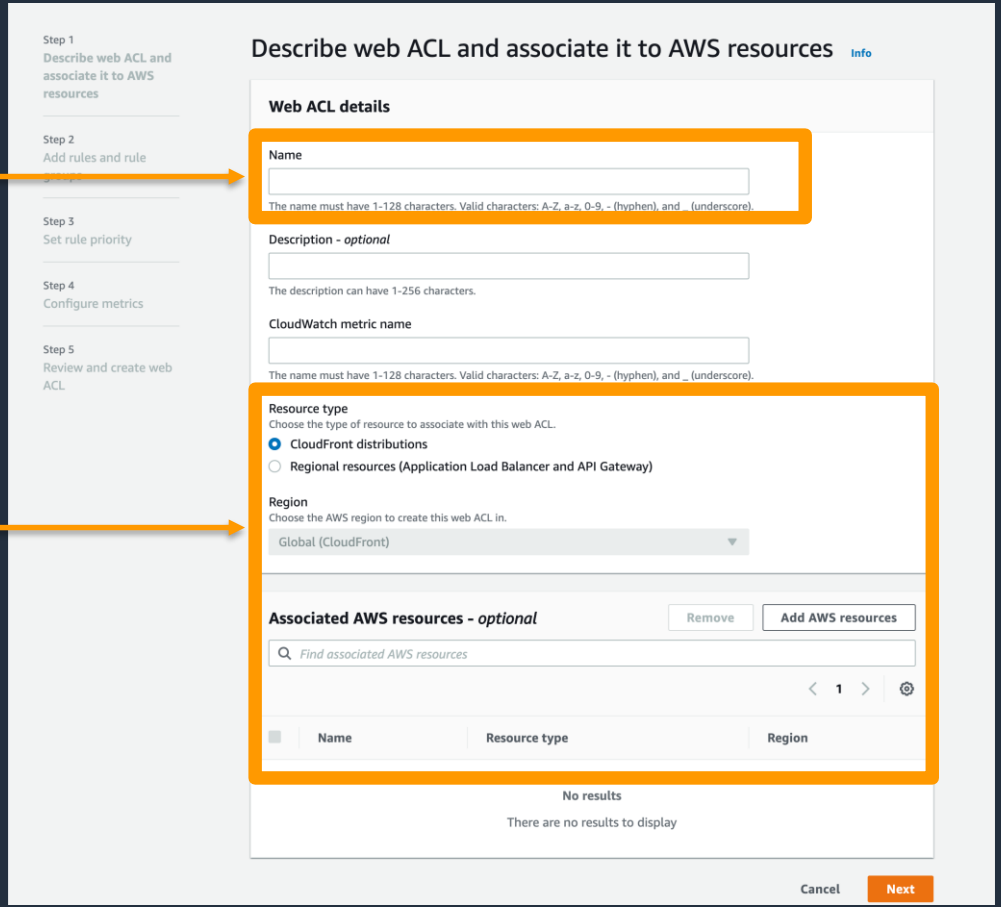
本日のアジェンダ

1. Web Application Firewall(WAF)とは
2. AWS WAFの機能
3. アップデートによる主な変更点
4. AWS WAFの詳細
5. AWS Managed Rules for AWS WAF
6. Web ACL作成の流れ
7. ログとモニタリング
8. AWS WAF利用のための考慮事項
9. 料金体系

Web ACL名の設定と対象リソースの選択

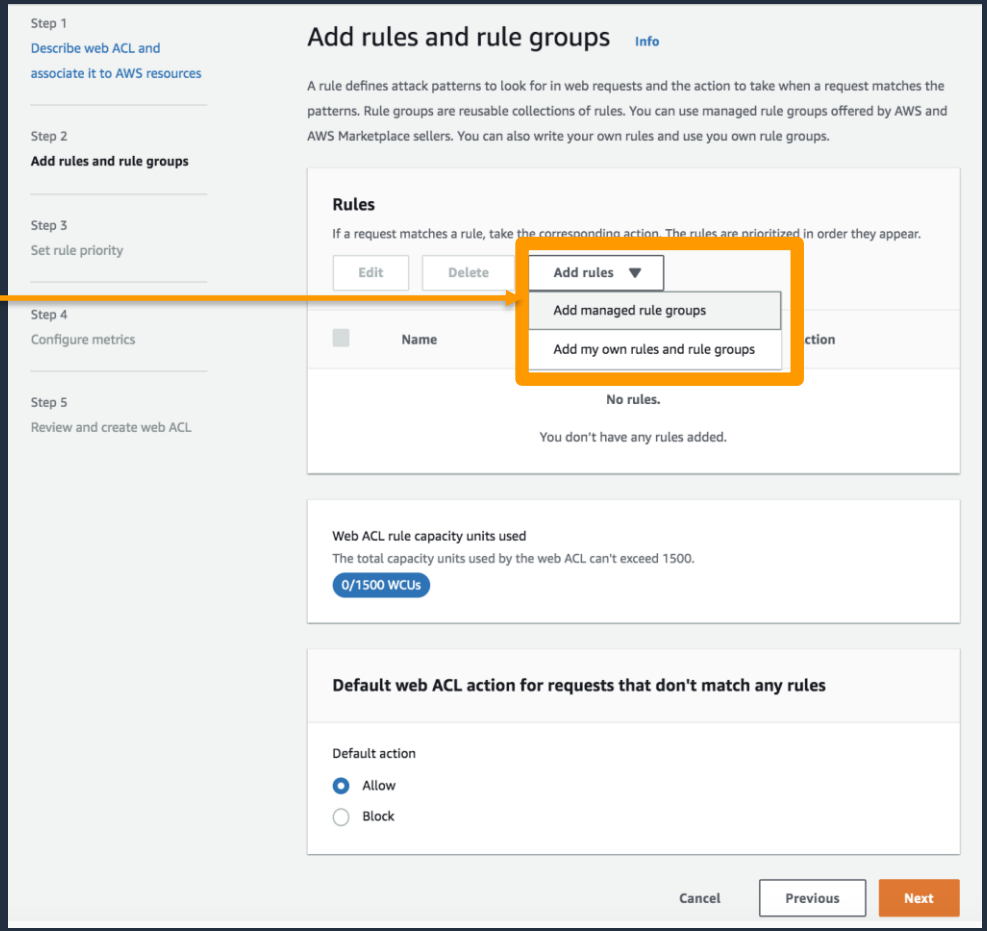
Web ACLの名称の設定

AWS WAFを適用するリソースの選択



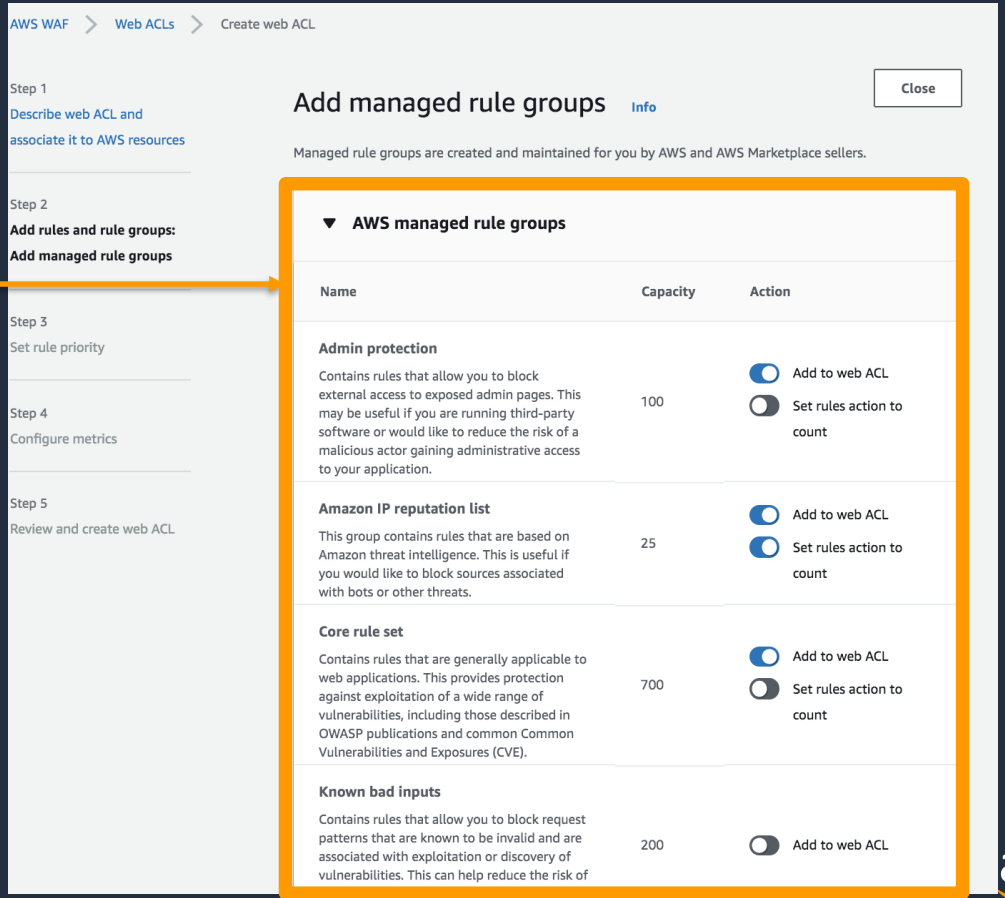
利用するルールの選択

使用するルールの選択する（ルール設定の子ウィザードが起動）



AWSマネージドルールの追加

Managed Rule Group内で
利用するルールを選択

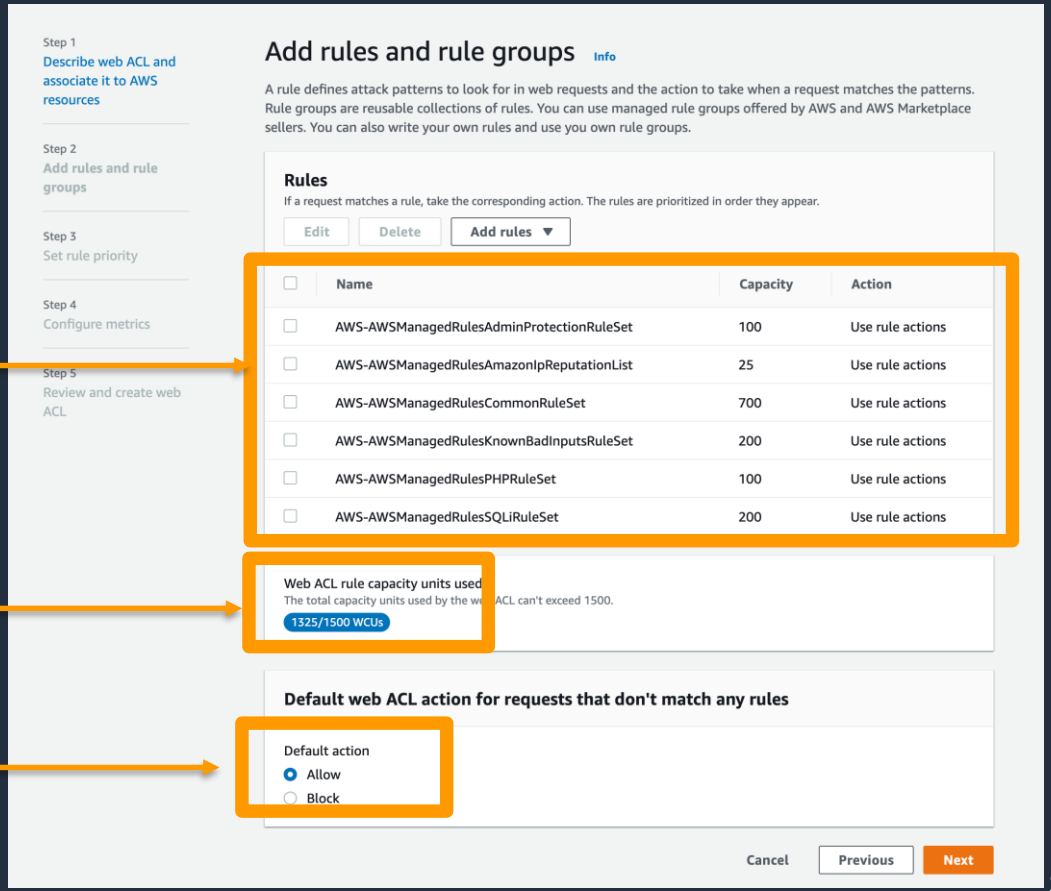


デフォルトアクションの選択

選択されたルールを確認

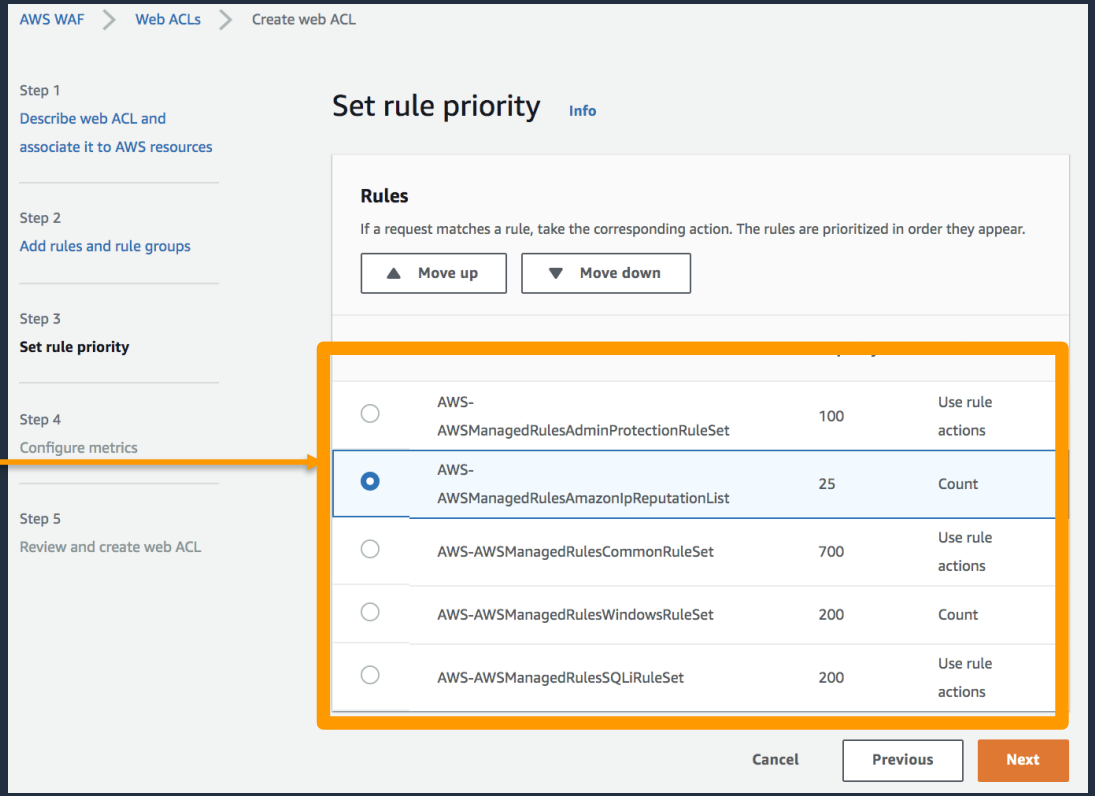
WCUの合計を確認

デフォルトアクションを
許可（Allow）か拒否
（Block）ションを選択



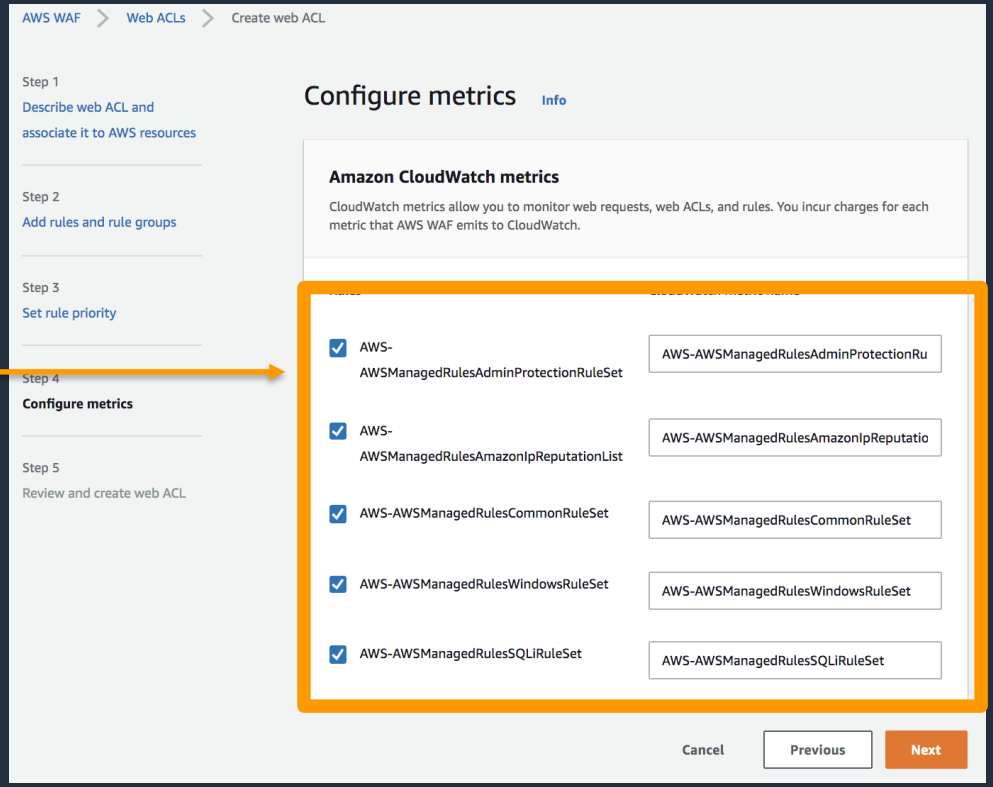
ルールの優先度の設定

ルールの優先度（適用順序）を指定



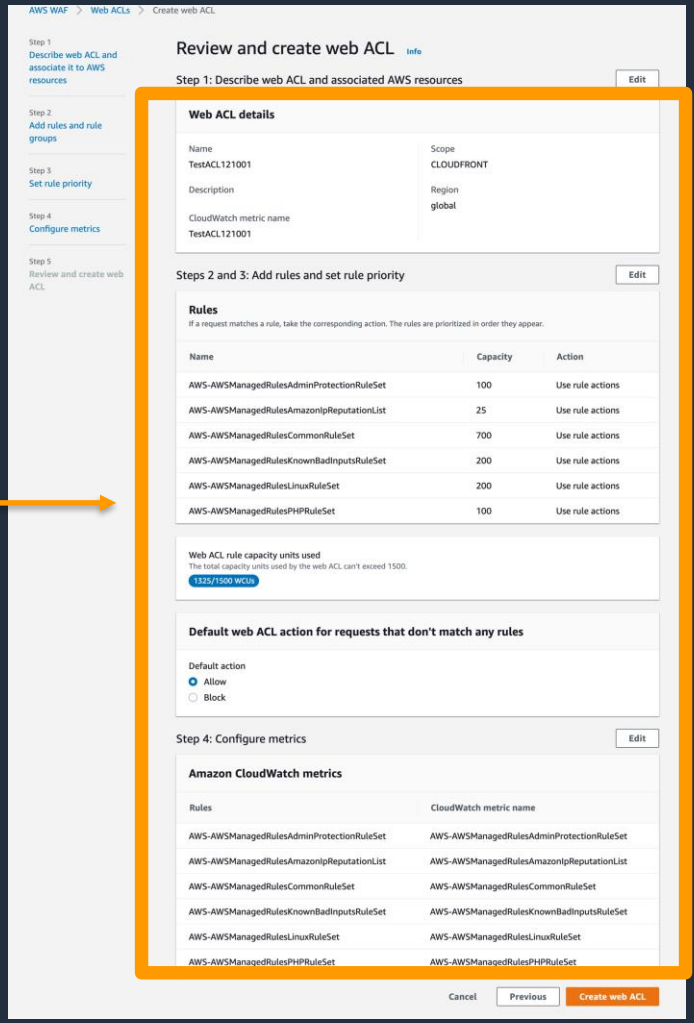
CloudWatchメトリクスの設定

CloudWatchメトリクス名を指定（デフォルトはルール名がそのまま適用される）



設定レビューとWeb ALC作成

設定した内容をレビューし、
Web ACLを作成



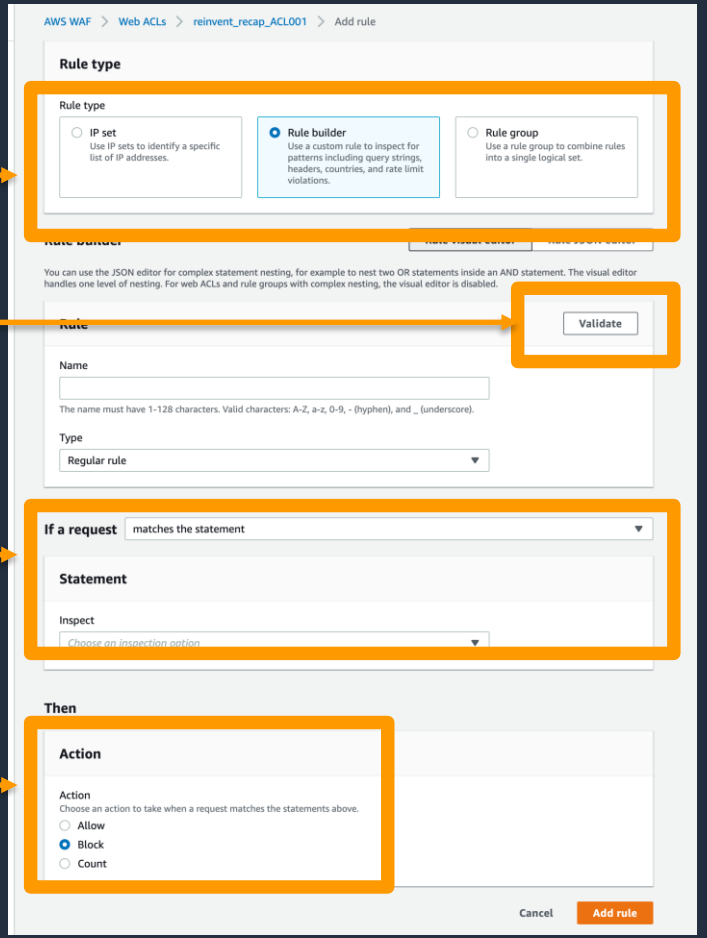
自分で作成したルールを追加する

IPセットルール、ルールビルダー、ルールグループから選択

ルールの整合性チェック

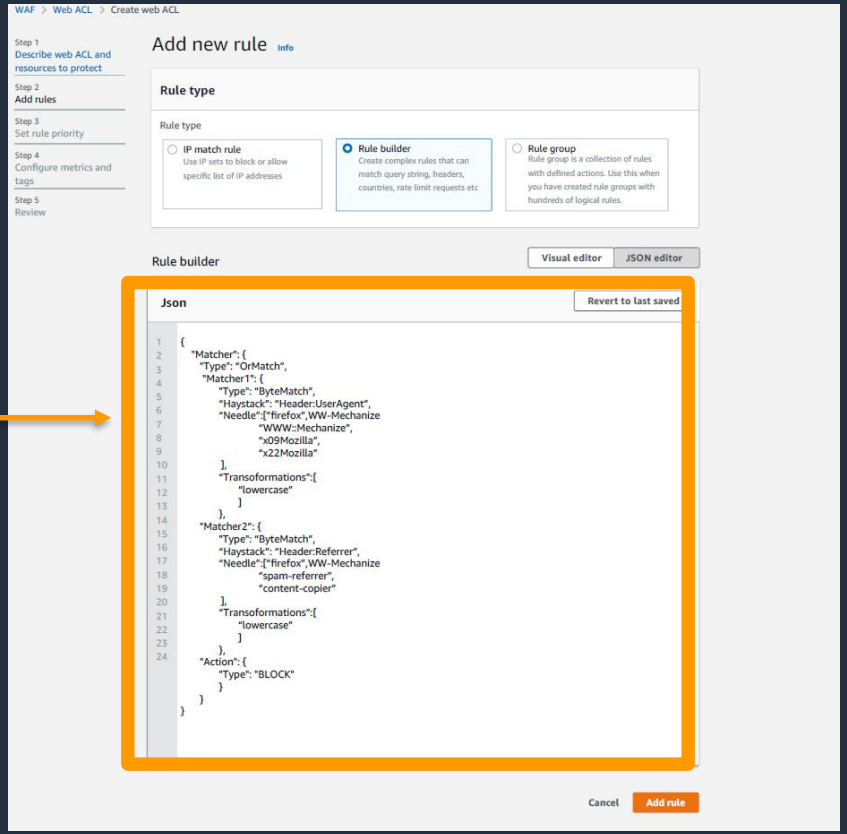
ステートメントを設定

ルールのアクションを選択



自分で作成したルールを追加する(JSON editor)

複雑なネスト条件を定義する場合にJSON editorで直接JSON形式でルールを記述可能（Visual editorでは一つのネストのみ対応）



APIによるWebACLの作成と更新

JSON形式でweb ACLを定義し、APIを呼び出す

web ACL: CreateWebACL または UpdateWebACLを利用

- シンタックスのチェックを行い、作成/更新を実施
- エラーが発生した場合は、以下のようなエラーメッセージが出力される
    - 例： WebACL has exceeded WCU capacity, rule statement is illogical

RuleGroup: CreateRuleGroup または UpdateRuleGroup

IP set: CreateIPSet または UpdateIPSet

Regex set: CreateRegexPatternSet または UpdateRegexPatternSet

aws

WAFv2 APIでサポートされるアクション一覧

AssociateWebACL
CreateIPSet
CreateRegexPatternSet
CreateRuleGroup
CreateWebACL
DeleteIPSet
DeleteRegexPatternSet
DeleteRuleGroup
DeleteWebACL
DisassociateWebACL
GetIPSet
GetManagedRuleSet
GetRateBasedStatementManagedKeys
GetRegexPatternSet
GetRuleGroup
GetWebACL
GetWebACLForResource
ListIPSets
ListManagedRuleSets
ListRegexPatternSets
ListResourcesForWebACL
ListRuleGroups
ListWebACLs
UpdateIPSet
UpdateRegexPatternSet
UpdateRuleGroup
UpdateWebACL

AWS WAFV2
APIリファレンス
https://docs.aws.amazon.com/waf/latest/APIReference/API_Operations.html

※ CLIで利用する場合は単語に-を付与
例） ListIPSets → list-ip-sets
aws wafv2 list-ip-sets --scope=REGIONAL

APIによるWeb ACLの作成例

```
// web ACLの作成例（作成するが関連付けはしない/別API）
{
  “Name”: “foo”,          // web ACLの名前
  “Scope”: “CLOUDFRONT”, // web ACLのリージョンまたはCloudFront（Global）
  “DefaultAction”: {      // web ACLでルールがトリガーされなかった場合のデフォルト動作
    "Block": {}           // 1つ選ぶ
    "Allow": {}
  },
  “Description”: “foo”, // web ACLのDescription
  “Rules”: [ // ルール記載ブロック
    {
      <Rule #1>
    }, {
      <Rule #2>
    }
  ],
  “VisibilityConfig”: { // CloudWatch とSampledRequests の設定
    "SampledRequestsEnabled": true,
    "CloudWatchMetricsEnabled": true,
    "MetricName": "TestMetric"
  }
}
```


aws

ルールの記述例

```
// conditionを組み合わせてルールを作成する
"Rules": [
  {
    "Name": "foo",
    “Priority”: 1, // トリガーされる順序（プライオリティ）
    "Statements": {
      <Condition> // conditions (SQLInjectionを検知した場合、など)を定義
    },
    "Action": {
      "Block": {}
      "Allow": {}
      "Count": {}
    },
   "VisibilityConfig": {// CloudWatch とSampledRequests の設定
     "SampledRequestsEnabled": true,
     "CloudWatchMetricsEnabled": true,
     "MetricName": "TestMetric"
    }
  }
  // 必要な内容を継続して記載
]
```

ステートメントの記述例 (XSS and SQLi)

```
"Statement": {
  "OrStatement": {
    "Statements": [{
      "XssMatchStatement": {
        "FieldToMatch": {
          "QueryString": {}
        },
        "TextTransformations": [
          {"Priority": 1, "Type": "URL_DECODE”},
          {"Priority": 2, "Type": ”LOWERCASE”},
        ]
      }
    },
    {
      "SqliMatchStatement": {
        "FieldToMatch": {
          "Body": {}
        },
        "TextTransformations": [
          {"Priority": 1, "Type": "HTML_ENTITY_DECODE”},
          {"Priority": 2, "Type": ”NONE_COMPRESS_WHITE_SPACE”}
        ]
      }
```

テキスト変換の種類:
NONE
COMPRESS_WHITE_SPACE
HTML_ENTITY_DECODE
LOWERCASE
CMD_LINE
URL_DECODE

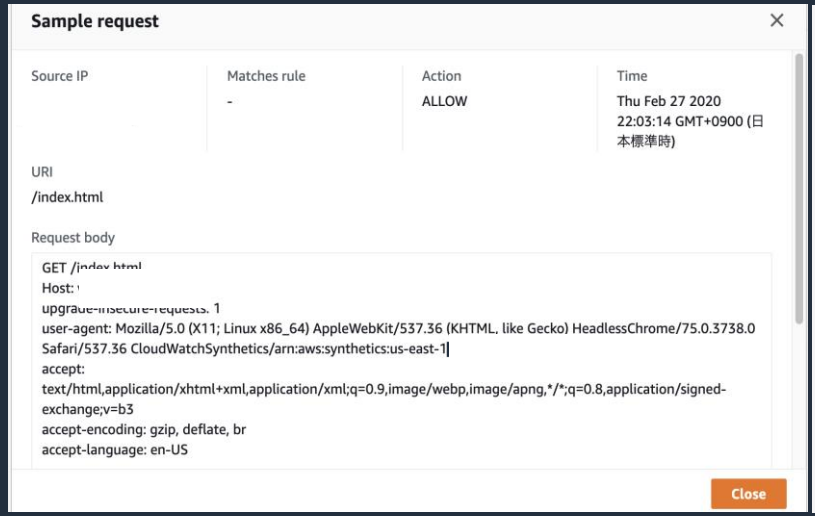
aws

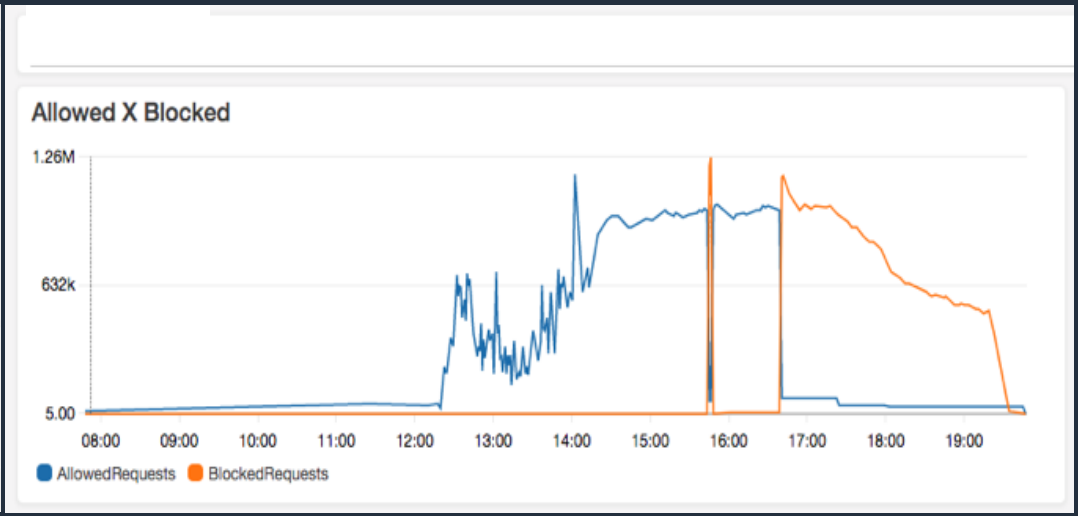
本日のアジェンダ

1. Web Application Firewall(WAF)とは
2. AWS WAFの機能
3. アップデートによる主な変更点
4. AWS WAFの詳細
5. AWS Managed Rule for AWS WAF
6. Web ACL作成の流れ
7. ログとモニタリング
8. AWS WAF利用のための考慮事項
9. 料金体系
10. まとめ

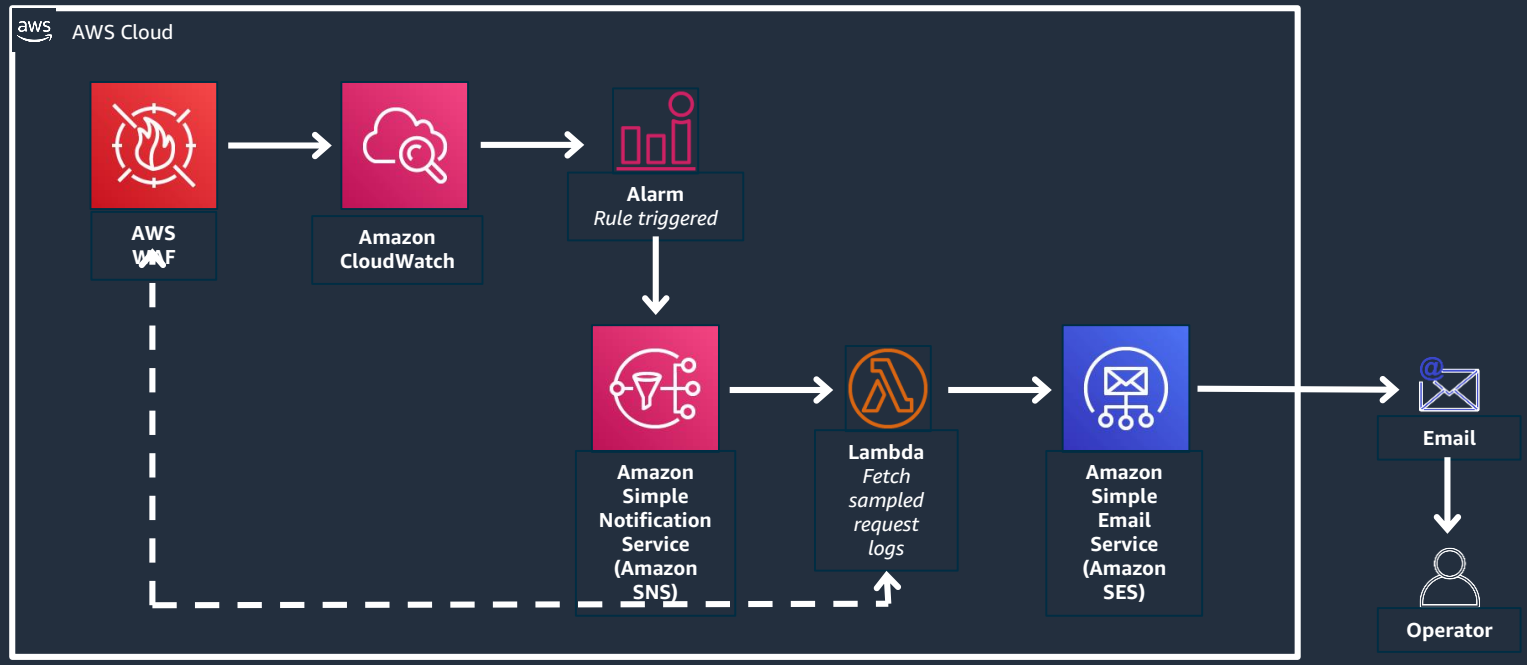
モニタリング

- Amazon CloudWatchを利用してAWS WAFのメトリクスをモニタリング、アラートの設定が可能
- サンプルリクエストからリクエスト内容の詳細(ソースIP、URI、リクエストBodyなど）を確認可能





参考: Amazon CloudWatchによるインシデントレポート

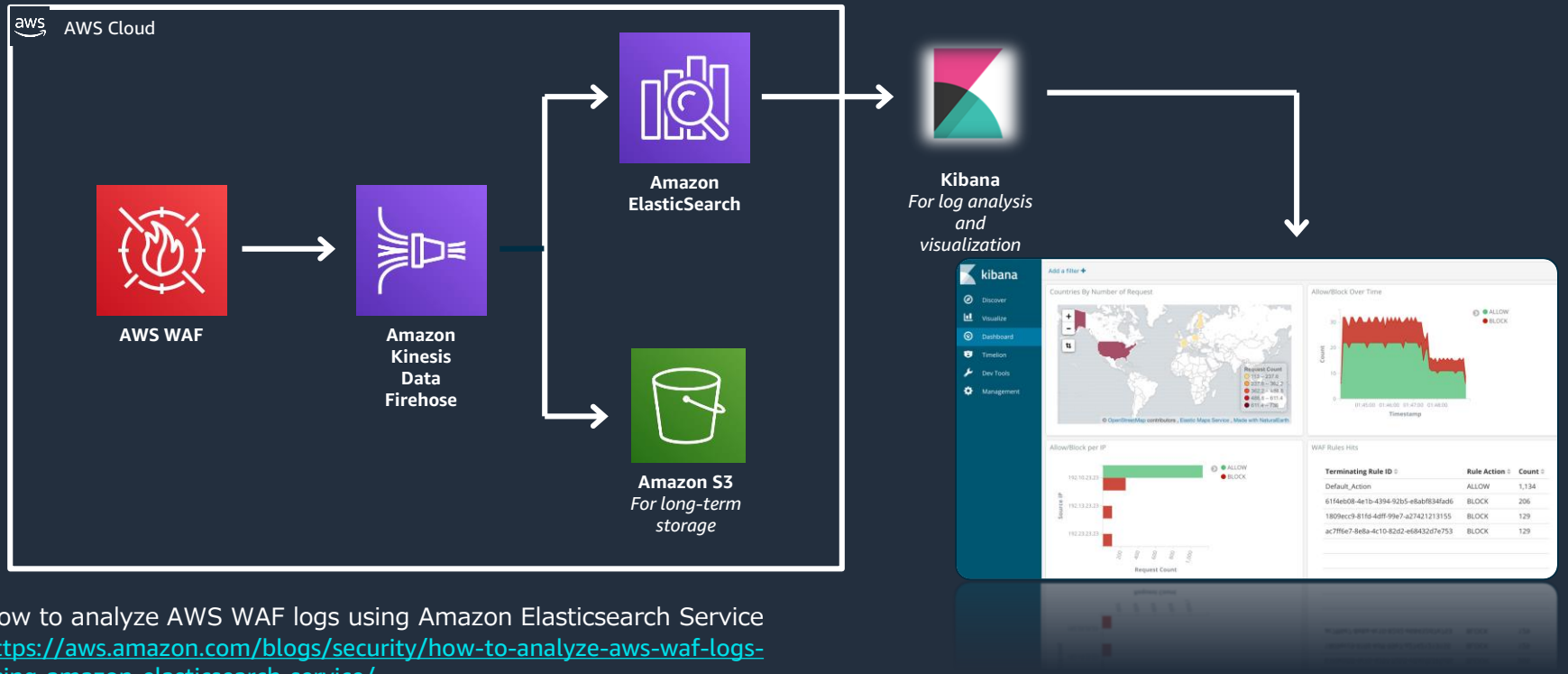


AWS WAF Full logs

- Kinesis Data Firehoseを介して、選択した宛先にJSON形式でログを送出
- 記録されたすべてのリクエストには、一致したリクエストヘッダーとRuleIDが含まれる
- Cookieや認証ヘッダなど、センシティブな情報をログから除外可能
- リクエストのどこで(ヘッダ/クエリ/Body) どの値がルールに合致してブロックされたかを確認可能


```
"terminatingRuleId": "STMTest_SQLi_XSS",
  "terminatingRuleType": "REGULAR",
  "action": "BLOCK",
  "terminatingRuleMatchDetails": [
    {
      "conditionType": "SQL_INJECTION",
      "location": "HEADER",
      "matchedData": [
        "10",
        "AND",
        "1"
      ]
    }
```


参考：Amazon Elasticsearch and Kibanaによるログ分析



How to analyze AWS WAF logs using Amazon Elasticsearch Service
https://aws.amazon.com/blogs/security/how-to-analyze-aws-waf-logs-using-amazon-elasticsearch-service/

参考: カスタムエラーページ(Block時に任意のページを表示)

WAFのBlock時にカスタムエラーページを表示したい場合、CloudFrontの“Custom Error Response”から任意のエラーページを設定可能

注意点：

- CloudFrontのみ設定可能（CloudFrontのカスタムエラーページの機能を利用）
- オリジンによって返された 403と、AWS WAFが返す 403 を区別できません
- ステータスコード 403 で複数のエラーページを表示できません。
  アプリケーションが出す 403とAWS WAFが出す 403 は同じエラーページを表示することになります。

参考: パートナーマネージドルール

パートナー提供のマネージドルールも利用可能

AWS Marketplaceでパートナー提供のマネージドルールグループをサブスクライブすることで、AWS WAFからそのマネージドルールグループを使用可能

AWS Marketplace
https://aws.amazon.com/marketplace/solutions/security/waf-managed-rules

CSC CYBER SECURITY CLOUD
GEOGUARD
FORTINET
imperva
Trustwave
f5
TREND MICRO

aws

本日のアジェンダ

1. Web Application Firewall(WAF)とは
2. AWS WAFの機能
3. アップデートによる主な変更点
4. AWS WAFの詳細
5. AWS Managed Rules for AWS WAF
6. Web ACL作成の流れ
7. ログとモニタリング
8. AWS WAF利用のための考慮事項
9. 料金体系
10. まとめ

AWS WAF利用のための考慮事項１ - ルールポリシー

ブラックリスト型（ネガティブ）

- デフォルトのアクションは許可
- ルールに合致した不正なリクエストをブロック
- 一般的なWebサイトではこちらが多い

ホワイトリスト型（ポジティブ）

- デフォルトのアクションはブロック
- ルールに合致したリクエストのみを許可
- 全てのリクエストパターンが事前に把握できる場合のみ
- 限られた利用者がアクセスするケース

aws

AWS WAF利用のための考慮事項２ - ルールの運用

セルフサービス

- テンプレートなどを利用してルールを構築
- ログの分析やルールのアップデートを継続実施
- 誤検知が発生した場合は、ルールの見直しを自身で実施

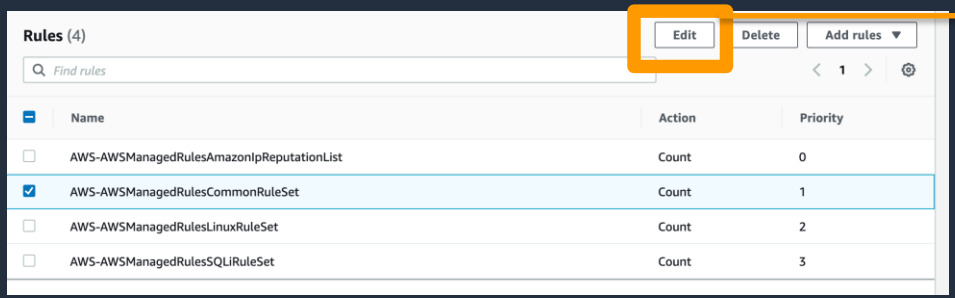
マネージド

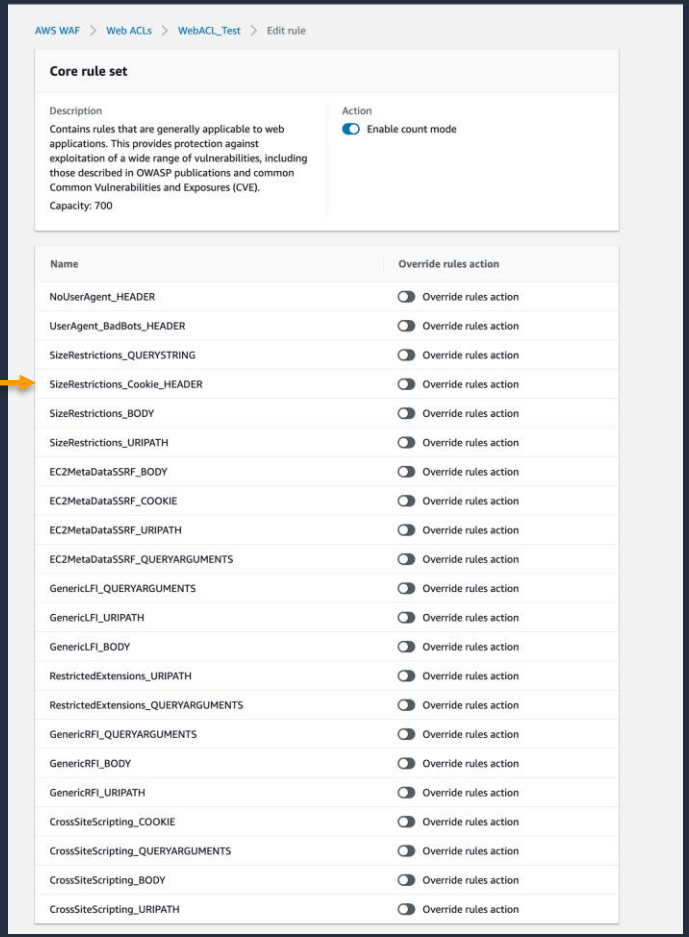
- AWSパートナーのマネージドルールまたは、AWS Managed rules for AWS WAFを利用する
- ルールのアップデートはManaged Ruleの提供ベンダーに任せる
- 誤検知が発生した場合はルールに合致した文字列を正規表現ルールに設定して、部分的にトラフィックを通す。または該当ルールをCountモードへ切り替え

aws

参考

マネージドルール内の個別のルールを
Countモードに切り替える





AWS WAF利用のための考慮事項２ - ルールの運用

セルフサービス

- テンプレートなどを利用してルールを構築
- ログの分析やルールのアップデートを継続実施
- 誤検知が発生した場合は、ルールの見直しを自身で実施

マネージド

- AWSパートナーのマネージドルールまたは、AWS Managed rules for AWS WAFを利用する
- ルールのアップデートはManaged Ruleの提供ベンダーに任せる
- 誤検知が発生した場合はルールに合致した文字列を正規表現ルールに設定して、部分的にトラフィックを通す。または該当ルールをCountモードへ切り替え

aws

AWS WAF利用のための考慮事項３ - 導入ステップ

Countモード

- 開発環境等でテストを実施
- 誤検知が発生しないかサンプルリクエストレポートで確認
- CloudWatchメトリクスでルールを評価
- Full Logを確認

Blockモード

- 誤検知がないことを確認後、Blockモードへ変更
- ルール毎に段階的にBlockモードへ変更することも可能
- サンプルリクエストレポートで確認
- BlockされたリクエストはFull logでWebACLsルールにマッチした内容を確認できる

AWS WAFとの組み合わせ

AWS WAF

AWS Shield Advanced

AWS Firewall Manager

本日のアジェンダ

1. Web Application Firewall(WAF)とは
2. AWS WAFの機能
3. アップデートによる主な変更点
4. AWS WAFの詳細
5. AWS Managed Rules for AWS WAF
6. Web ACL作成の流れ
7. ログとモニタリング
8. AWS WAF利用のための考慮事項
9. 料金体系
10. まとめ

AWS WAFの料金

Web ACL

- $5 /月（1web ACLあたり/1500 WCUを含む）

ルール

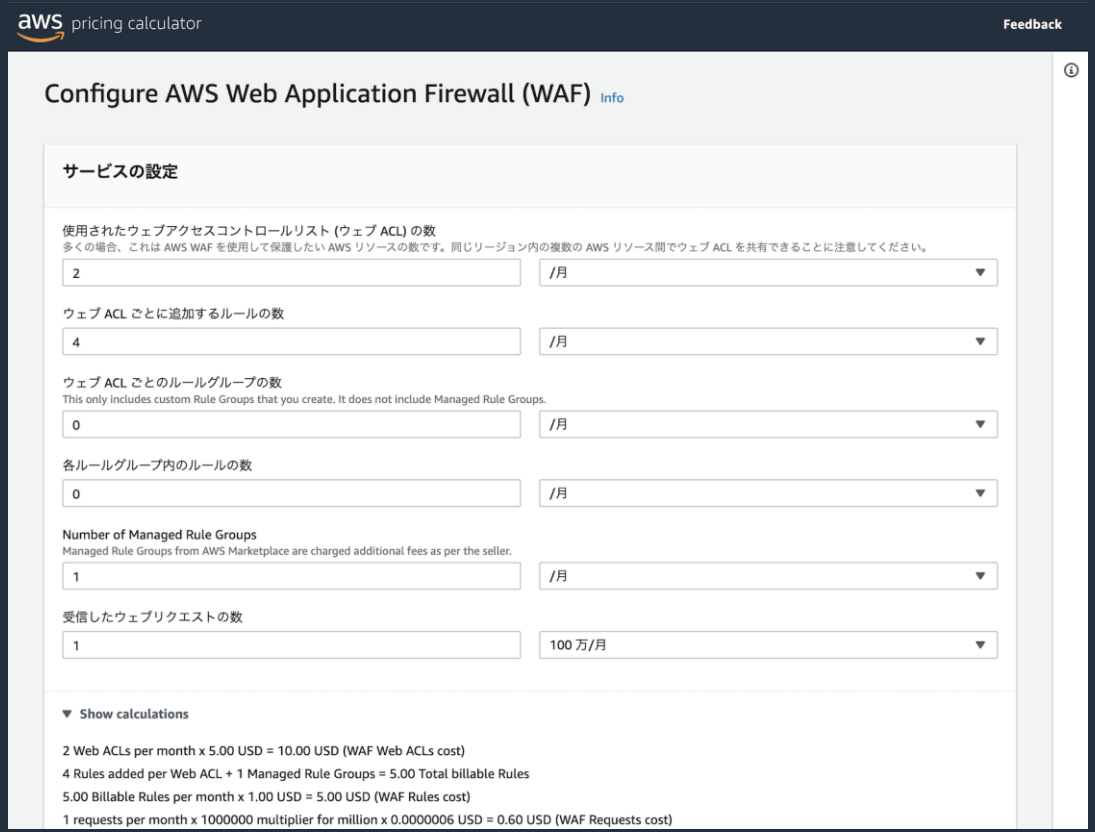
- $1 /月 （1ルールあたり）

リクエスト数

- $0.60 /百万リクエスト

- 全てのリージョンで同一価格

料金計算ツール



料金計算ツール
https://aws.amazon.com/jp/waf/pricing/

本日のアジェンダ

1. Web Application Firewall(WAF)とは
2. AWS WAFの機能
3. アップデートによる主な変更点
4. AWS WAFの詳細
5. AWS Managed Rules for AWS WAF
6. Web ACL作成の流れ
7. ログとモニタリング
8. AWS WAF利用のための考慮事項
9. 料金体系
10. まとめ

まとめ

- マネージドルールにより、実践的なルールセットをすぐに導入可能
- フレキシブルにルールをカスタマイズ可能
- 全ての機能がAPIでコントロール可能
- モニタリングやログにより迅速に攻撃状況を把握可能
- 今すぐ使い始められて、使った分だけの料金

参考資料

- AWS WAF 開発者ガイド
  https://docs.aws.amazon.com/ja_jp/waf/latest/developerguide/waf-chapter.html
- AWS マネージドルール ルールグループ のリスト
  https://docs.aws.amazon.com/ja_jp/waf/latest/developerguide/aws-managed-rule-groups-list.html
- AWS マネージドルール変更履歴
- https://docs.aws.amazon.com/waf/latest/developerguide/aws-managed-rule-groups-changelog.html
- AWS WAF V2 APIリファレンス
  https://docs.aws.amazon.com/waf/latest/APIReference/API_Operations.html
- 料金計算ツール
  https://aws.amazon.com/jp/waf/pricing/

Q&A

お答えできなかったご質問については
AWS Japan Blog 「https://aws.amazon.com/jp/blogs/news/」にて
後日掲載します。

AWS の日本語資料の場所「AWS 資料」で検索

https://amzn.to/JPArchive

AWS Well-Architected 個別技術相談会

毎週”W-A個別技術相談会”を実施中

- AWSのソリューションアーキテクト(SA)に対策などを相談することも可能
- 申込みはイベント告知サイトから

(https://aws.amazon.com/jp/about-aws/events/)

AWS イベント で[検索]

ご視聴ありがとうございました

AWS 公式 Webinar
https://amzn.to/JPWebinar

過去資料
https://amzn.to/JPArchive