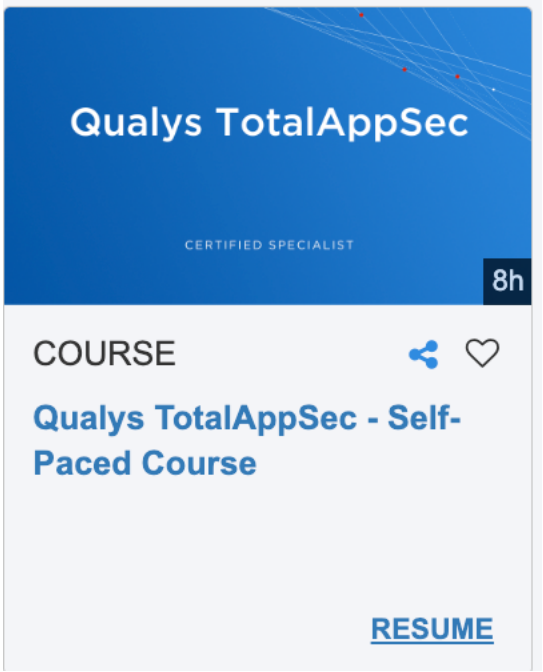
2026 年 1 月

Qualys TotalAppSec トレーニングガイド

はじめに

このコースでは、環境内のデータを正確にを維持し、一貫性と信頼性の高いレポートを生成するためのベストプラクティスを学びます。また、組織内の様々なステークホルダー向けのレポートを生成するための、様々な戦略とツールの活用方法も学びます。

下記のフリートレニング **Qualys TotalAppSec** にエントリし、この資料とともにコースを完了する事を推奨します。
https://www.qualys.com/training/



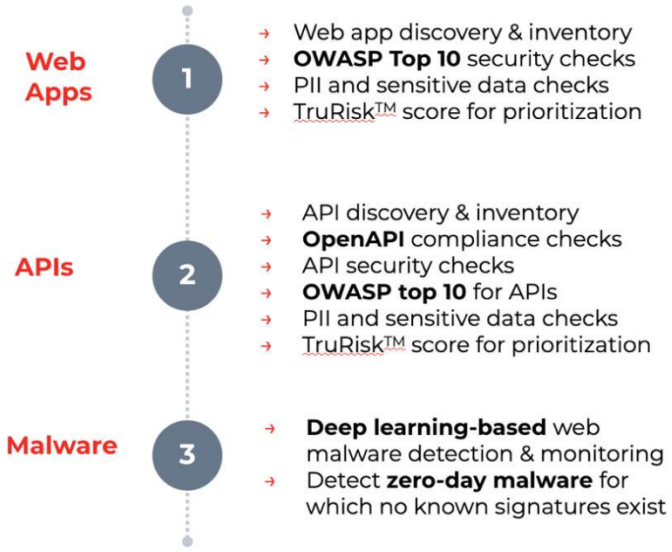
アジェンダ

このレッスンでは、以下のトピックについて演習やビデオを見ながら説明してゆきます。各セクションの後にショートテストがあります。

TotalAppSec 概要

TotalAppSec とは何ですか?

TotalAppSec は、**Web アプリケーション スキャン、API スキャン、Web マルウェア検出**を組み合わせた統合アプリケーション セキュリティ プラットフォームであり、詳細な可視性、インテリジェントなテスト、優先順位付けされたリスク修復を通じて、組織がクラウドベースのアプリケーションを保護できるようにします。



イントロダクション

このトレーニング ビデオでは、TotalAppSec の概要について学習します。TotalAppSec とは何か、なぜ重要なのか、TotalAppSec が克服する主な課題、コア機能の詳細などについて説明します。

Total Appsec の概要

- **Qualys が提供するクラウドファースト組織向けの統合型アプリケーションセキュリティプラットフォーム。**
- **3 つの主要機能を統合：**
  - Web アプリケーションスキャン：**OWASP Top 10 脆弱性検出、PII 露出確認、リスクスコアリング。**
  - API セキュリティ：**API 自動検出、OpenAPI コンプライアンス評価、脆弱性スキャン。**
  - Web マルウェア検出：**ディープラーニングでゼロデイ脅威を含むマルウェアを防御。**

提供価値

- **アプリや API の深い可視性。**
- **スマートでコンテキストに応じたテスト。**
- **優先度付き修復で重要な問題を迅速に解決。**

現実的な課題

- **Web アプリや API の集中管理が困難（特にマルチクラウド環境）。**
- **API テストが不完全でコンプライアンスチェック不足。**
- **複数ツール使用によるワークフロー分断。**
- **マルウェア検出がサイロ化され、対応が遅れる。**
- **ビジネスに基づくリスク優先度がなく、アラート疲れや未解決脆弱性が発生。**

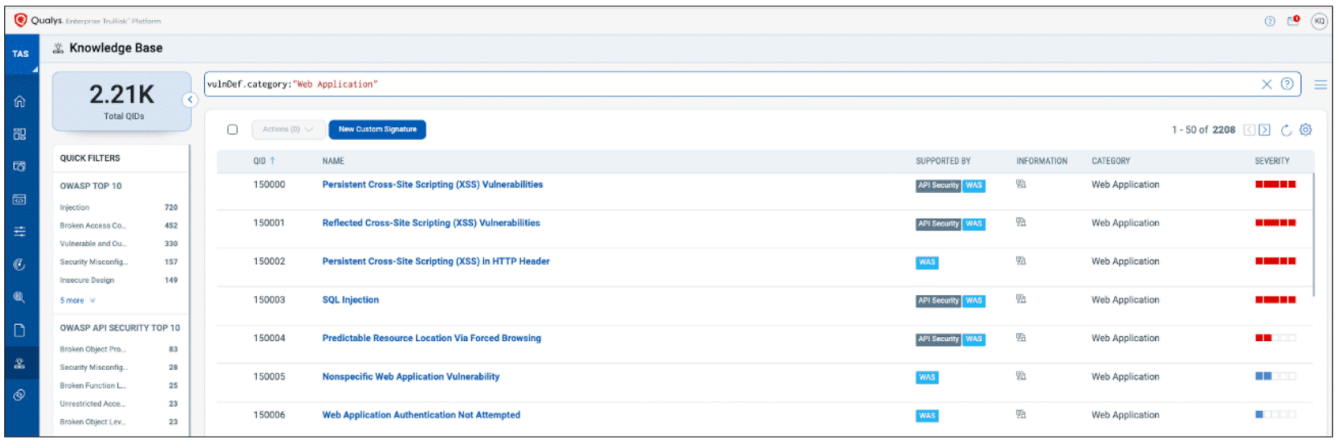
Total Appsec の解決策

- **クラウド全体で包括的なアプリ・API 発見。**
- **OWASP Top 10 脆弱性、マルウェア、PII 露出、API 仕様コンプライアンスを評価。**
- **リスクスコアリングと統合ビューで迅速な修復。**
- **Mulesoft、Apigee、Swagger などとの API 統合で開発・セキュリティ連携を強化。**

まとめ

- **モダンなクラウドネイティブ環境におけるアプリケーションセキュリティを効率化。**
- **統合された可視性、自動スキャン、深いインサイト、リスク優先度付けを 1 つのソリューションで提供。**

ナレッジベースとカスタムシグネチャ



ナレッジベースとは何か、そしてそれに含まれる内容について

- **「Knowledge Base」タブでは、産業オートメーション環境で特定できる脆弱性に関する詳細情報を提供します。**
- **表示される主要な属性:**
  - **QID(脆弱性識別子)**
  - **脆弱性タイトル**
  - **重大度評価**
  - **CVE 識別子**
  - **検出日や変更日**

ナレッジベースの機能

- **脆弱性を管理・探索する方法:**
  - 検索:**Qualys Query Language(QQL)を使用**
  - フィルター:**左側ペインで簡単に絞り込み**
- クイックアクションメニュー:
  - **脆弱性詳細の表示**
  - **重大度の編集・再設定**
  - **脆弱性の無視またはアクティブ化**
- アクションメニュー:
  - **複数脆弱性への一括操作**
- 検索アクションメニュー:
  - **最近の検索の表示**
  - **カスタムクエリの保存**
  - **保存した検索履歴の管理**

脆弱性スキャンの 3 つのタイプ

1. 確認済み脆弱性(confirmed vulnerabilities)
   - **実際にセキュリティ上の脅威をもたらすことが確認された脆弱性**
   - **スキャン中に高い信頼性で検出**

2. 潜在的脆弱性(potential vulnerabilities)
    - **脆弱性の可能性を示すが、手動検証が必要**
    - **Web アプリケーションスキャンでフラグ付け**
3. 機密情報検出(sensitive content)
    - **JavaScript ファイル内のパスワードや API キーなどの露出データを識別**

重大度評価

- **QUALYS では 1～5 のレーティングを使用**
    - 5:**最も重大で即時対応が必要**
- **潜在的脆弱性と確認済み脆弱性の両方に適用**

4. 収集された情報(information gathered)

- **脆弱性ではなく、スキャン中に収集された有用なデータ**
    - **HTTP ヘッダー**
    - **サーバーバナー**
    - **クロールされたリンク数**
- **重大度は通常 1～3**

Confirmed Vulnerabilities

Severity	Level	Description
	Minimal	Basic information disclosure (e.g. web server type, programming language) might enable intruders to discover other vulnerabilities, but lack of this information does not make the vulnerability harder to find.
	Medium	Intruders may be able to collect sensitive information about the application platform, such as the precise version of software used. With this information, intruders can easily exploit known vulnerabilities specific to software versions. Other types of sensitive information might disclose a few lines of source code or hidden directories.

Potential Vulnerabilities

Severity	Level	Description
	Minimal	Presence of this vulnerability is indicative of basic information disclosure (e.g. web server type, programming language) and might enable intruders to discover other vulnerabilities. For example in this scenario, information such as web server type, programming language, passwords or file path references can be disclosed.
	Medium	Presence of this vulnerability is indicative of basic information disclosure (e.g. web server type, programming language) and might enable intruders to discover other vulnerabilities. For example version of software or session data can be disclosed, which could be used to exploit.

Sensitive Content

Severity	Level	Description
	Minimal	Sensitive content was found in the web server response. During our scan of the site form(s) were found with field(s) for credit card number or social security number. This information disclosure could result in a confidentiality breach and could be a target for intruders. For this reason we recommend caution.
	Medium	Sensitive content was found in the web server response. Specifically our service found a certain sensitive content pattern (defined in the option profile). This information disclosure could result in a confidentiality breach and could be a target for intruders. For this reason we recommend caution.

Information Gathered

Severity	Level	Description
	Minimal	Intruders may be able to retrieve sensitive information related to the web application platform.
	Medium	Intruders may be able to retrieve sensitive information related to internal functionality or business logic of the web application.
	Serious	Intruders may be able to detect highly sensitive data, such as personally identifiable information (PII) about other users of the web application.

評価基準

Web Application Security Consortium - **オープンソースのベストプラクティスやセキュリティ基準を作成。**
WASC Threat Classification(脅威分類)で、各種攻撃手法や脆弱性を体系的に整理している。
Common Weakness Enumeration(CWE) **-ソフトウェアやハードウェアに存在する セキュリティ上の弱点（脆弱性の種類）を体系的に分類・整理した国際的な標準リスト。共有脆弱性タイプ一覧とも呼ばれている。**

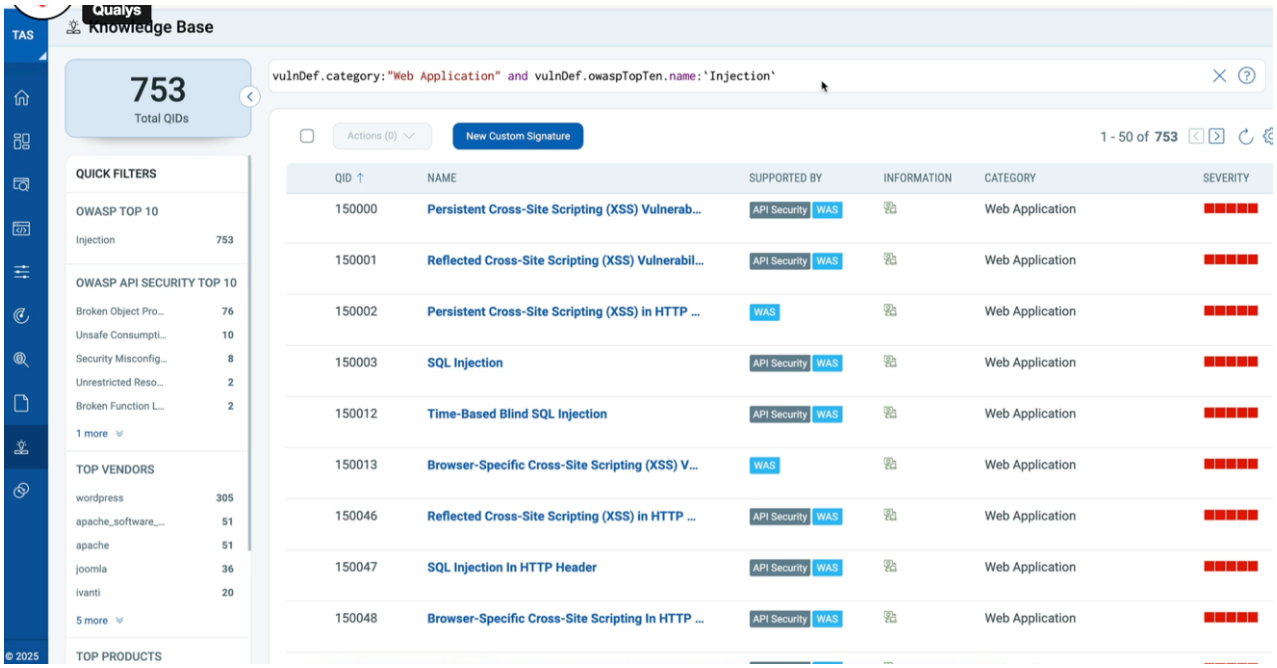
OWASP（Open Web Application Security Project）- **Web アプリケーションにおける最も重大な脆弱性リストを定期的に発表しており、世界中の企業や開発プロジェクトでセキュリティ基準として採用されている。**

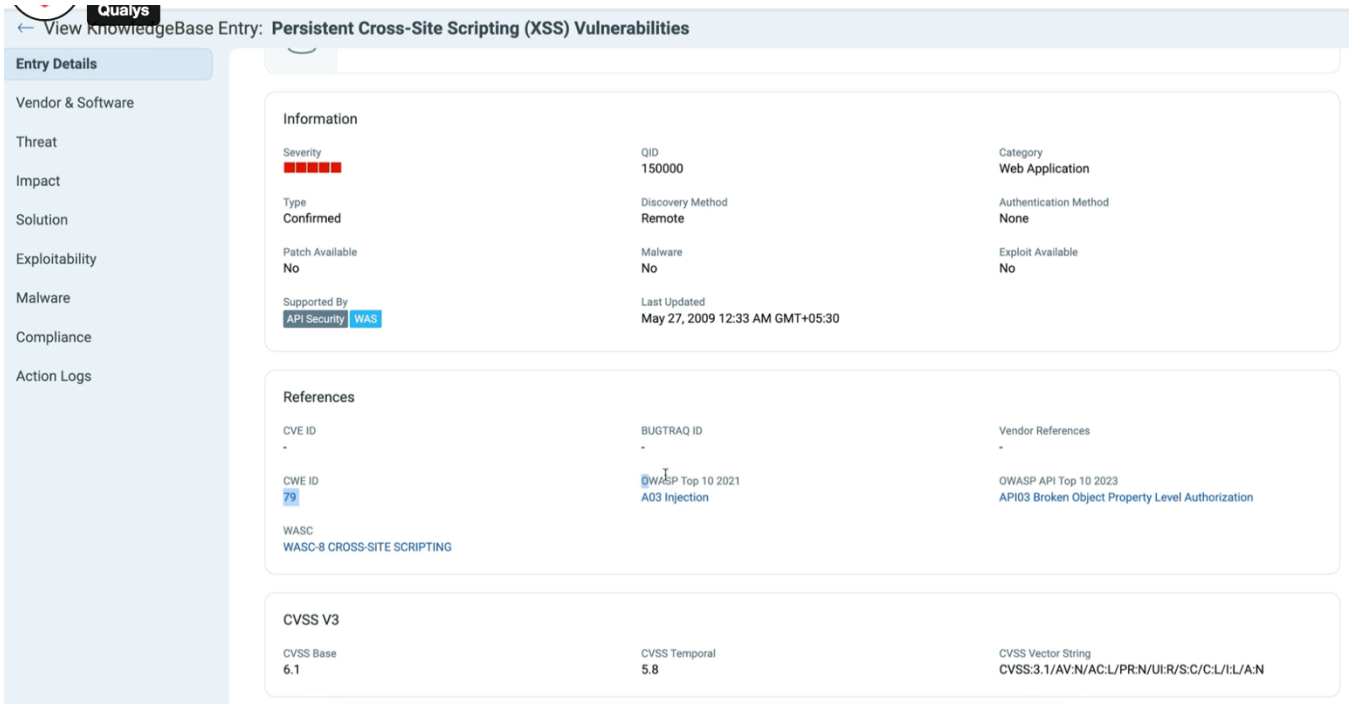
デモンストレーション概要

- **「WAS(Total App Sec)」モジュールのランディングページから「Knowledge Base」タブへ移動**
- **QUALYS サブスクリプションアカウントに関連する QID の総数を確認**
- フィルタリング方法：
  - **クイックフィルターを使用**
  - **Qualys Query Language で検索トークンを記述**
- 検索トークン例：
  - **vulnerability definition カテゴリ (vulnDef.category:Web Application)**
- 詳細確認：
  - **QID の情報（名前、リモートアクセス可否、重大度）**
  - **CVSS バージョン 3 のスコア**
  - **脅威の詳細、影響、解決策**

検索クエリの保存

- **画面右端のアイコンで検索クエリを保存**
- **保存した検索結果を管理可能**



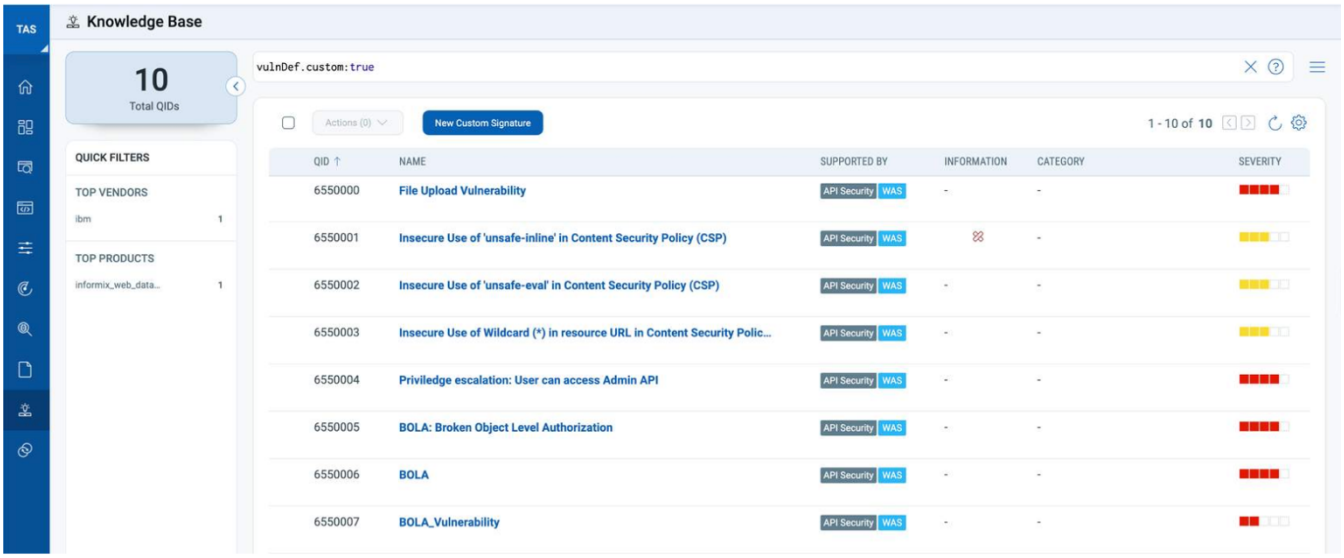


カスタムシグネチャ

Qualys TotalAppSec はカスタムシグネチャをサポートするようになり、組織は独自の検出ルールを作成して追加できるようになりました。 これにより、組み込みのシグネチャと並行したカスタム セキュリティ テストが可能になり、柔軟性が向上し、特定のセキュリティ要件との整合性が向上し、脆弱性管理が改善されます。

カスタム署名の使用例:

- **特定のニーズに合わせた新しい署名の作成**
- **侵入型ペイロードを使用したテストの追加**
- **既存の Qualys TotalAppSec チェックの変更または強化**
- **環境固有のテストにカスタムペイロードを使用する**
- **侵入テスト中に発見された問題の検出を自動化する**



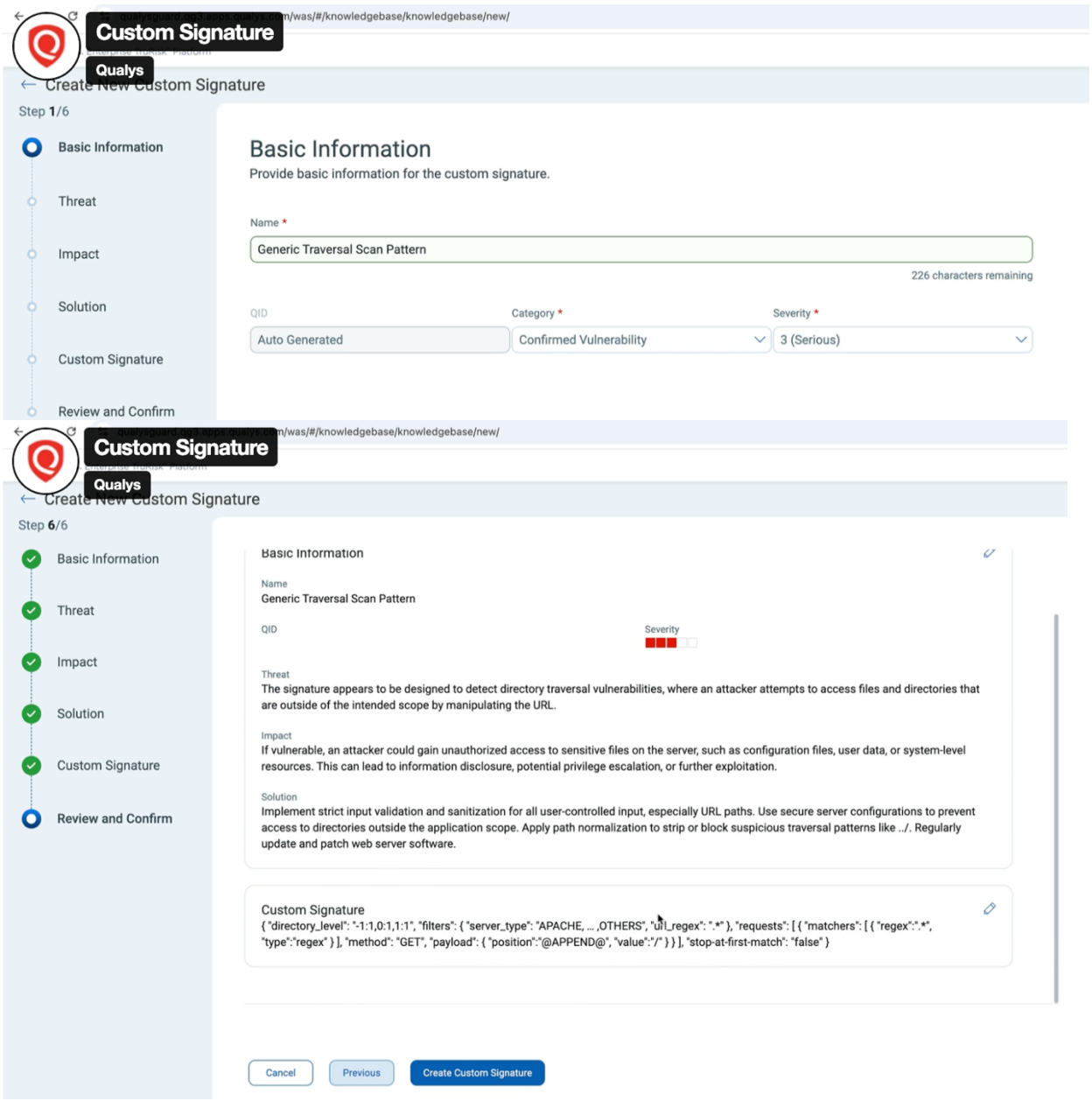
このトレーニング ビデオでは、カスタム署名を作成する方法、必要な手順、効果的な実装のためのベスト プラクティスについて学習します。

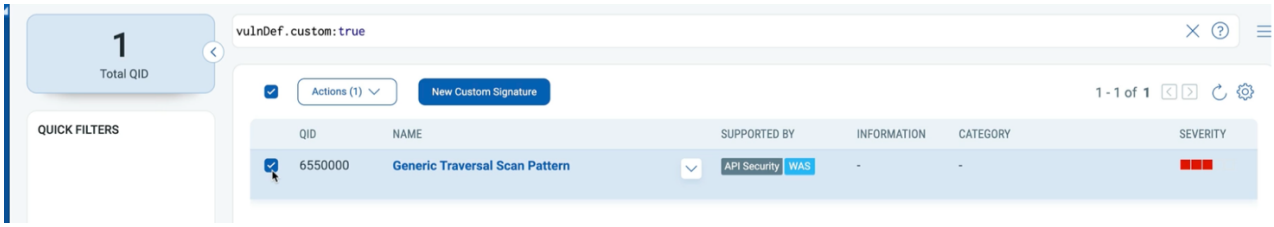
具体例

Apache などの Web サーバーに対し、URL 末尾にスラッシュを追加してレスポンスを分析することで、ディレクトリトラバーサルや不適切なリダイレクトを検出。

カスタムシグネチャ作成手順

1. 「Create New Signature」ボタンをクリック。
2. 基本情報入力（名前、カテゴリ、重大度）。
3. 脆弱性の脅威説明、影響、解決策を記載。
4. 検出ロジック（JSON 形式）を入力。
5. 内容確認後「Create custom signature」をクリック。


Custom Signature
Qualys
Create New Custom Signature
Step 1/6
Basic Information
Threat
Impact
Solution
Custom Signature
Review and Confirm
Basic Information
Provide basic information for the custom signature.
Name *
Generic Traversal Scan Pattern
226 characters remaining
QID
Category *
Severity *
Auto Generated
Confirmed Vulnerability
3 (Serious)
Custom Signature
Qualys
Create New Custom Signature
Step 6/6
Basic Information
Threat
Impact
Solution
Custom Signature
Review and Confirm
Basic Information
Name
Generic Traversal Scan Pattern
QID
Severity
Threat
The signature appears to be designed to detect directory traversal vulnerabilities, where an attacker attempts to access files and directories that are outside of the intended scope by manipulating the URL.
Impact
If vulnerable, an attacker could gain unauthorized access to sensitive files on the server, such as configuration files, user data, or system-level resources. This can lead to information disclosure, potential privilege escalation, or further exploitation.
Solution
Implement strict input validation and sanitization for all user-controlled input, especially URL paths. Use secure server configurations to prevent access to directories outside the application scope. Apply path normalization to strip or block suspicious traversal patterns like ../. Regularly update and patch web server software.
Custom Signature
{ "directory_level": "-1:1,0:1,1:1", "filters": { "server_type": "APACHE, ... ,OTHERS", "url_regex": ".*" }, "requests": [{ "matchers": [{ "regex":".*", "type":"regex" }], "method": "GET", "payload": { "position":"@APPEND@", "value":"/" } }], "stop-at-first-match": "false" }
Cancel
Previous
Create Custom Signature


1
Total QID
QUICK FILTERS
vulnDef.custom:true
Actions (1)
New Custom Signature
1 - 1 of 1
QID
NAME
SUPPORTED BY
INFORMATION
CATEGORY
SEVERITY
6550000
Generic Traversal Scan Pattern
API Security
WAS

管理面の注意点

- カスタムシグネチャ作成にはマネージャーユーザー権限が必要。
- 作成後はナレッジベースで検索・管理可能（例：検索トークンでフィルタリング）。

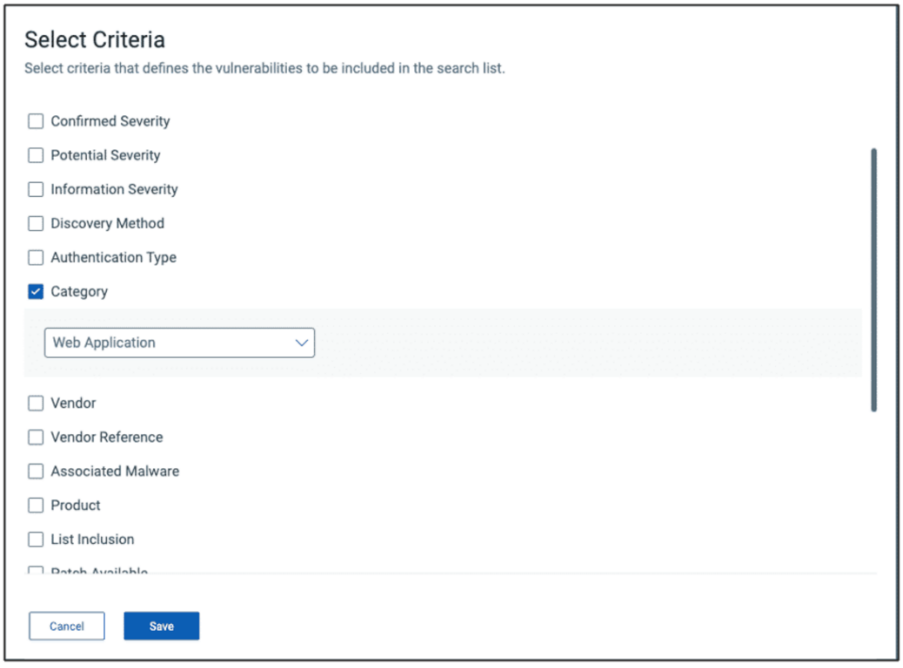
Search Lists

このレッスンでは、静的検索リスト(Static Search List)と動的検索リスト(Dynamic Search List)を作成し、Web アプリケーションスキャンで効果的に適用する方法を学びます。ガイド付きラボとシミュレーションを通して、これらのリストを設定してスキャン動作とレポートをカスタマイズする実践的な経験を積むことができます。

サーチリストは、スキャンおよびレポート中に脆弱性をフィルタリングするために使用される QID (Qualys ID) のユーザー定義のコレクションです。

静的検索リスト – ユーザーが QID を明示的に追加する、手動で管理されるリスト。
動的検索リスト – 事前定義された検索条件に基づいて自動的に更新されます。

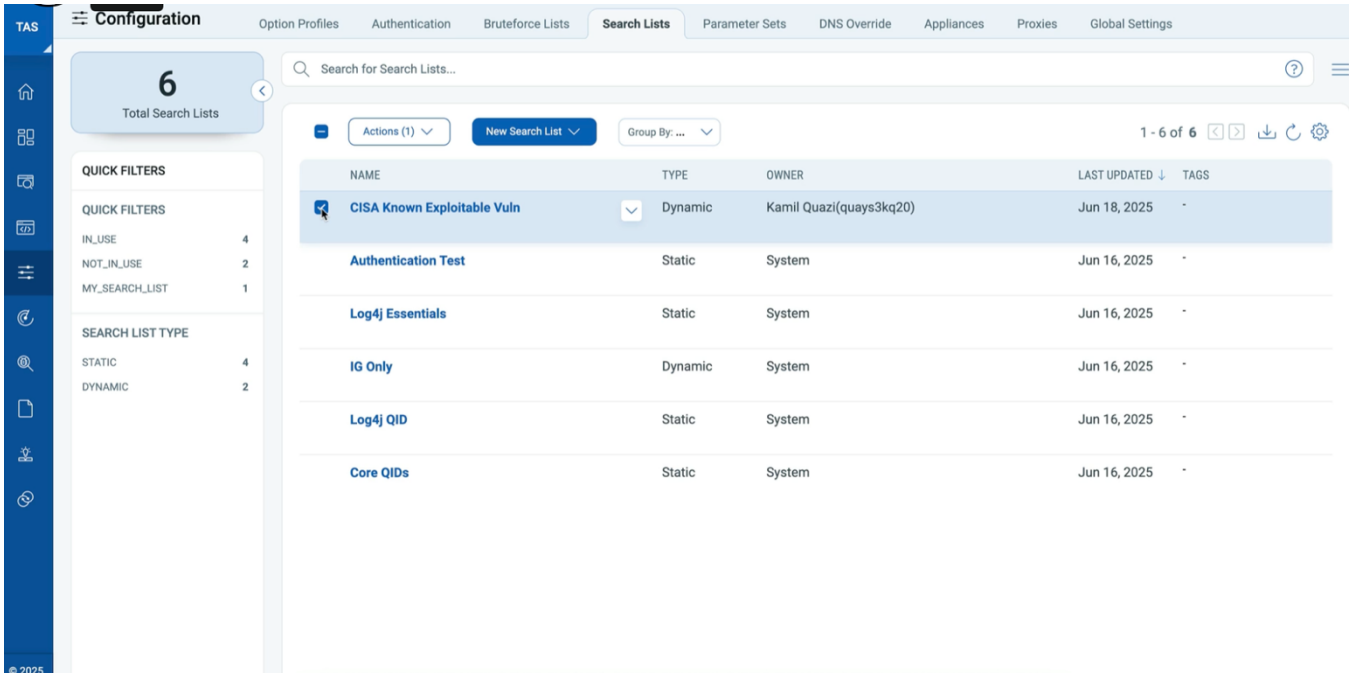
例:
SQLi または Log4j のみのスキャンを実行します。
XSS のみのレポートを作成します。



- **メリット**
  動的・静的検索リストを使うことで、スキャンやレポートをセキュリティ優先度に合わせて柔軟に設計可能。

- **デモ手順（動的検索リストの作成）**

1. 「Configuration」→「Search List」へ移動。
2. 「New Search List」ボタンをクリック。
3. 名前を入力（例：CISA 既知のエクスプロイト対象脆弱性）。
4. 検索条件を設定：
    - カテゴリを「Web Application」に設定。
    - 「Exploitability」で「CISA Known Exploitable Vulnerabilities」を選択。
5. 保存後、テストスキャンで条件に一致する脆弱性（例：111 件）を確認。
6. コメント追加、内容確認後「Create Search List」ボタンで作成完了。
7. 作成した検索リストはスキャンやレポートに利用可能。



ディスカバリ

1. クロスプロダクト機能の概要

最初に説明されているのは、Qualys Total Appsec の「クロスプロダクト機能」です。
この機能は、クラウド環境にある Web アプリケーションや API を自動的に検出するための仕組みです。これにより、管理者は自社のクラウド構成に存在するアプリや API を把握しやすくなり、セキュリティ対策を強化できます。

2. API ゲートウェイとの統合

次に、Mulesoft や Azure API Management などの API ゲートウェイと統合する事ができます。
この統合により、Swagger ファイルや公開されているエンドポイントを自動的に検出できます。Swagger は API 仕様を記述するための標準フォーマットなので、これを検出することで API の構造や利用可能なエンドポイントを把握できます。

3. Swagger API ディスカバリー機能

この機能は、サブスクリプション内の Web アプリをスキャンし、Swagger や OpenAPI 仕様を検出します。
検出された仕様は構造化され、分析しやすくなります。さらに、Qualys Connector との連携により、Web アプリケーションスキャンがクラウド構成と統合され、自動検出とカタログ化が可能になります。

4. 検出後の処理

検出された Web アプリや API は、脆弱性スキャンの対象としてサブスクリプションに追加できます。これにより、セキュリティカバレッジが広がり、潜在的なリスクを早期に発見できます。

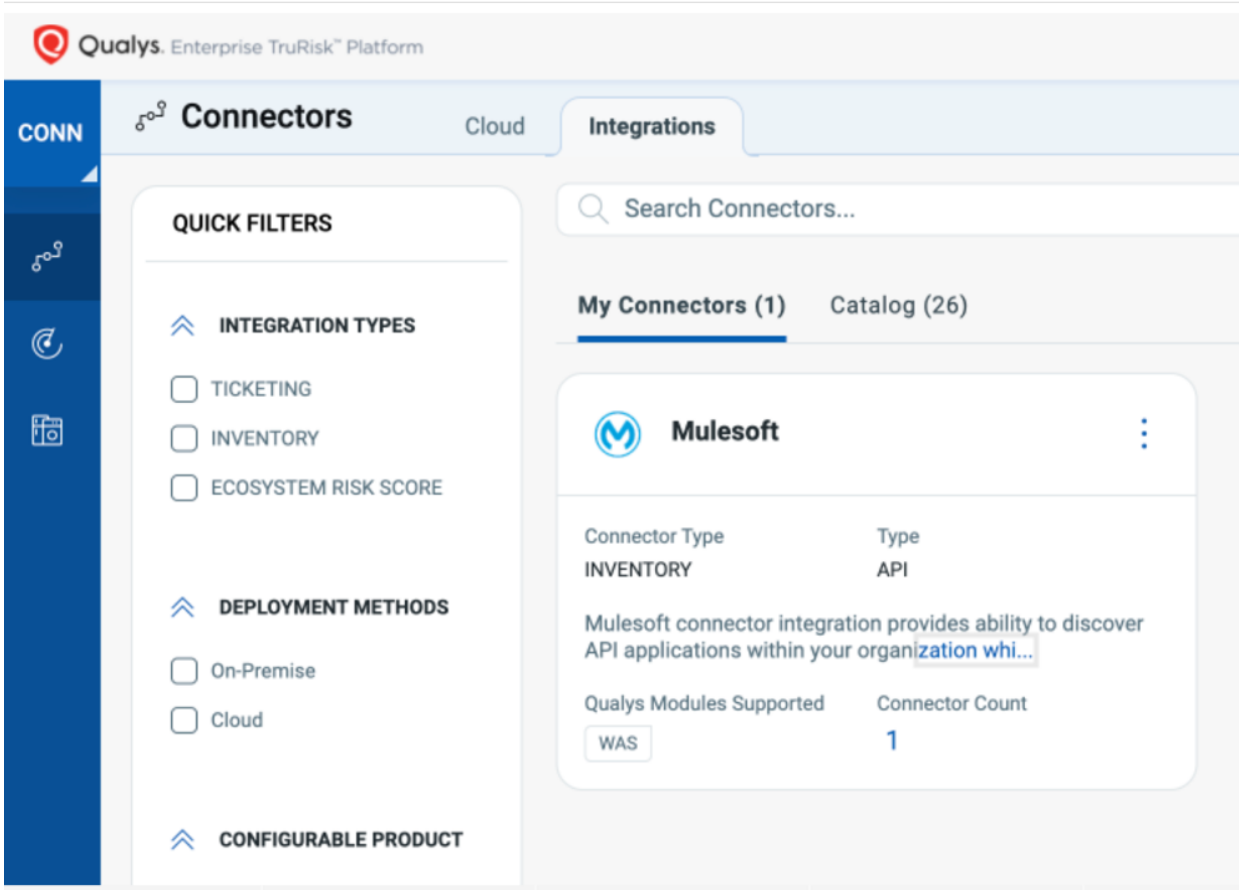
5. Discovery Sources タブ

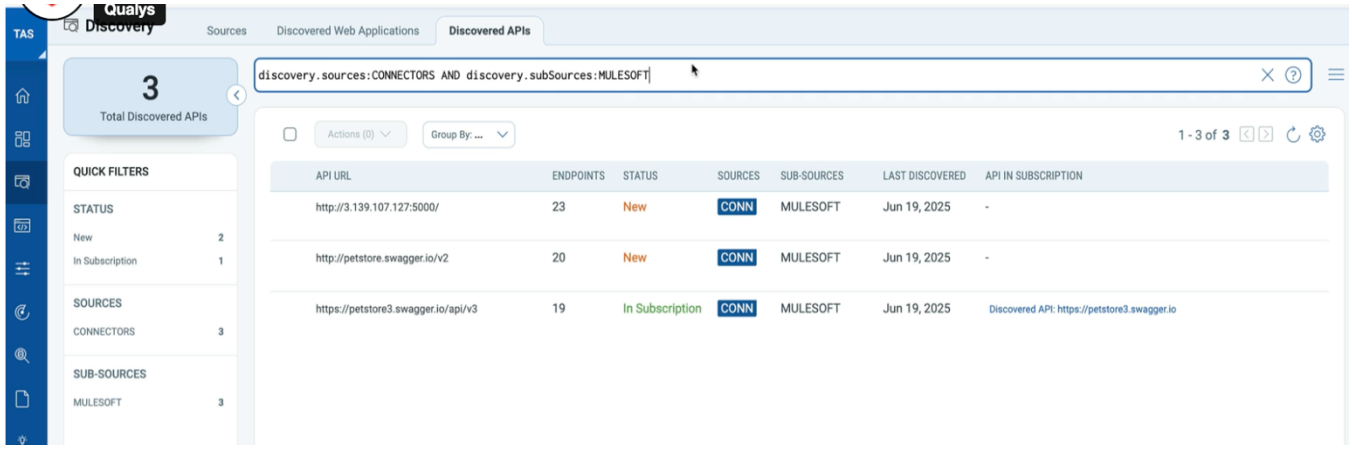
ここでは、AWS コネクタなどのクラウド連携機能について説明されています。
Discovery Sources タブで利用可能なコネクタを確認することで、クラウドインフラ全体のアプリケーションを効率的に追跡・保護できます。

6. Mulesoft API コネクタの設定手順

ドキュメント後半は、Mulesoft API コネクタの設定方法をステップごとに説明しています。

- **Create Connector** → **Create connection** をクリックして設定開始。
- 必要情報を入力：
    - コネクタ名と説明
    - データモデル選択（API Discovery）





認証情報（Token URL、ユーザー名、パスワード）

1. データモデルと変換マップ

Mulesoft API コネクタは、標準のデータモデルマッピングと変換マップを提供します。これにより、データのインポートやエクスポート時に正しく変換されます。

2. プロファイル作成

次に、コネクタの状態や実行スケジュールを決めるプロファイルを作成します。
スケジュールは実施頻度やタイムゾーン、実施日時を指定できます。

9. コネクタの有効化と確認

設定後、コネクタが Inactive の場合は、アクションボタンから Activate して接続を確認します。

10. API ディスカバリーの実行

最後に、Discover APIs をクリックすると、自動生成されたクエリで API 一覧が表示されます。
検出された API に対して、必要なアクション（追加、スキャンなど）を選択できます。

基本的なアプリケーション設定

導入

このレッスンでは、基本情報の入力、クロール範囲オプションの構成、デフォルトのスキャン設定の識別、Web アプリケーションの作成など、Web アプリケーションを定義する方法について説明します。

Web アプリケーションスキャン

Web アプリケーション スキャン（WAS）は、Web アプリケーションのクロールとテストを自動化して脆弱性を検出するクラウドベースのソリューションです。
WAS は、包括的、正確、かつスケーラブルな Web セキュリティを提供し、組織が Web アプリケーション内の脆弱性を評価、追跡、対処することを可能にします。WAS は、Qualys Enterprise TruRisk ™プラットフォームを基盤としています。
WAS には、マルウェア感染がないか Web サイトを積極的に監視し、ブラックリストへの登録を防止してブランドの評判を守るために Web サイトの所有者に警告を送信する高度なスキャンテクノロジーが搭載されています。
WAS は、動的アプリケーションセキュリティテスト（DAST）ツールです。

Web アプリケーションスキャンテスト手法

自動テスト（フォールトインジェクション）

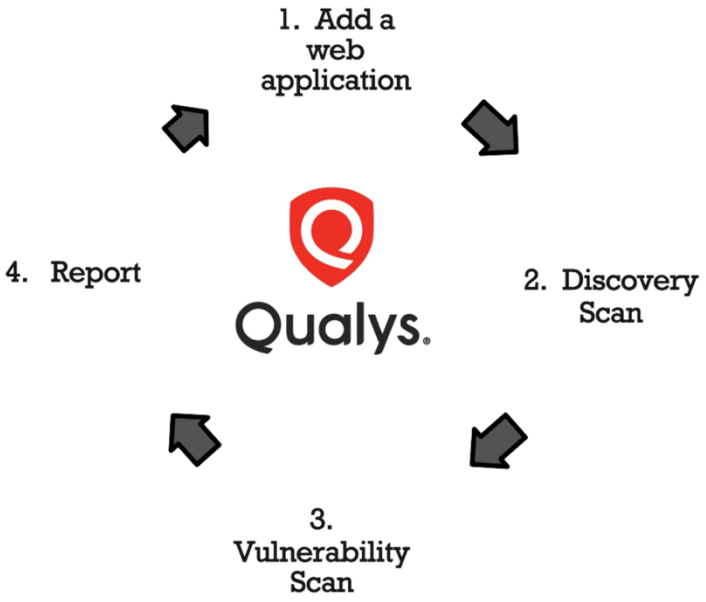
「特別に作られた」キャラクターを提出する
サーバーの応答を観察する
これは Web アプリの脆弱性の 80～85%を占めています。

手動テスト（BURP 統合）

自動化ツールは、Web アプリケーションのバグ (ユーザー入力内の SQL 実行) を効果的に検出します。
Qualys は Burp Suite Professional と連携し、ユーザーが手動テスト結果をアップロードし、自動スキャンデータを充実させてより包括的なセキュリティ評価、すなわちウェブアプリケーションの分析プロセスを実現できるようにしています。

WAS ワークフロー

右図の通り、まずは、分析を開始するために必要なコンテキストをウェブアプリケーションとして、プラットフォームに追加することから始めます。その後、ディスカバリースキャンやクロールスキャンがアプリケーション構造をマッピングします。見落とされがちなページ、リンク、エンドポイントの特定をします。次に脆弱性スキャンが始まります。このフェーズでは、アプリケーション全体の攻撃面全体で共通するセキュリティ上の欠陥を積極的にテストします。そして最後に、システム全体を詳細な脆弱性レポートにまとめ、チームが修正に必要な可視性を提供します。この効率的なプロセスにより、潜在的な脅威が悪用される前に早期に発見されます。



新規追加: Web アプリケーション

脆弱性、マルウェア、機密コンテンツなどのセキュリティ リスクをスキャンする Web アプリケーションを追加します。

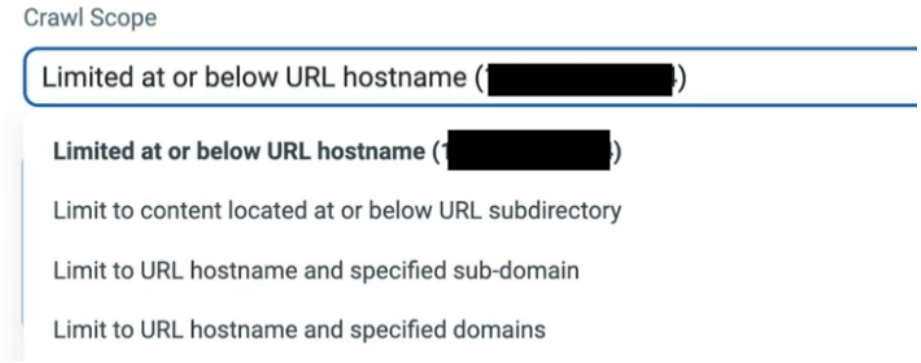
- 基本情報 - Web アプリケーションの基本情報を提供します。
- クロール設定 - Web アプリケーション設定で選択したクロール範囲によって、スキャンの対象範囲が決まります。
- デフォルトのスキャン設定 - Web アプリケーションのデフォルトのオプション プロファイル、スキャナ アプライアンス、およびその他のスキャン設定を構成できます。
- 追加構成 - 認証レコード、ヘッダー インジェクション、パス ファジングなどの高度なスキャン設定をカスタマイズして、脆弱性の検出を強化します。
- 確認と確定 - 定義されたすべての設定を表示し、設定を編集または更新できます。

基本的なウェブアプリケーションのセットアップ

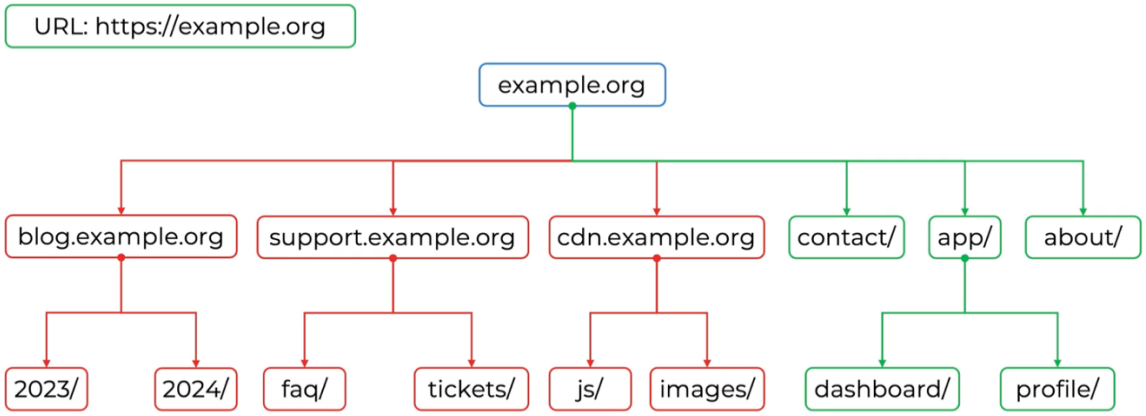
このトレーニングビデオでは、Basic Information、Crawl Settings、Default Settings の設定から基本的なウェブアプリケーションのセットアップ方法を学びます。

クロールスコープ

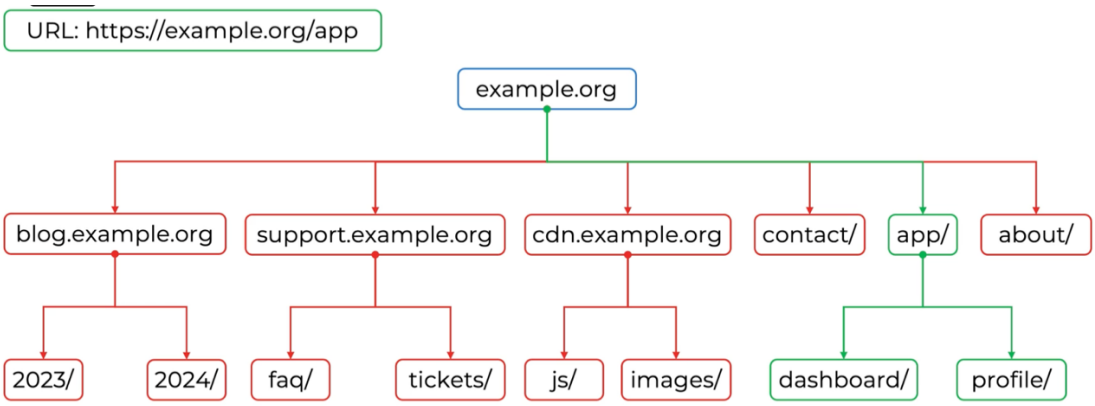
下記の通りクロールスコープには 4 つの選択肢があり、これにより評価中にスキャナーがアプリケーションのどの部分を探索するかが決まります。



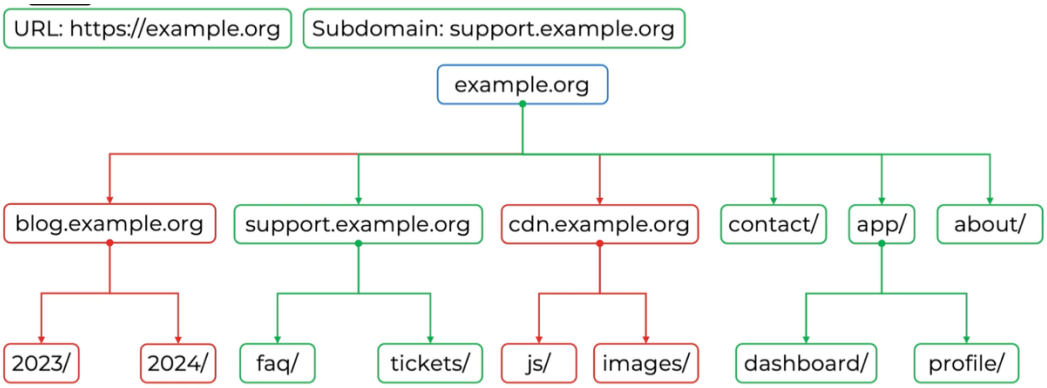
1. **Limited at or below URL hostname**
   次の例では、緑で囲ったディレクトリはクロールしますが、赤で囲ったサブドメイン配下はクロールしません。



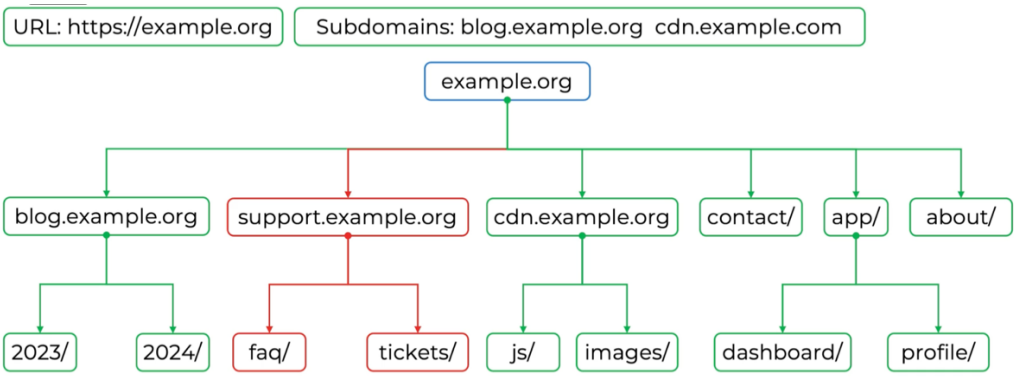
2. **Limit to content located at or below URL subdirectory**
   次の例では、緑で囲った、指定したサブディレクトリ配下はクロールしますが、それ以外はクロールしません。



3. **Limit to URL hostname and specified sub-domain**
   次の例では、緑で囲ったドメイン名と指定したサブドメインはクロールしますが、他のサブドメインはクロールしません。



4. **Limit to URL hostname and specified domains**
   次の例では、緑で囲ったドメイン名と指定したドメイン及びサブドメインはクロールしますが、他のドメインやサブドメインはクロールしません。

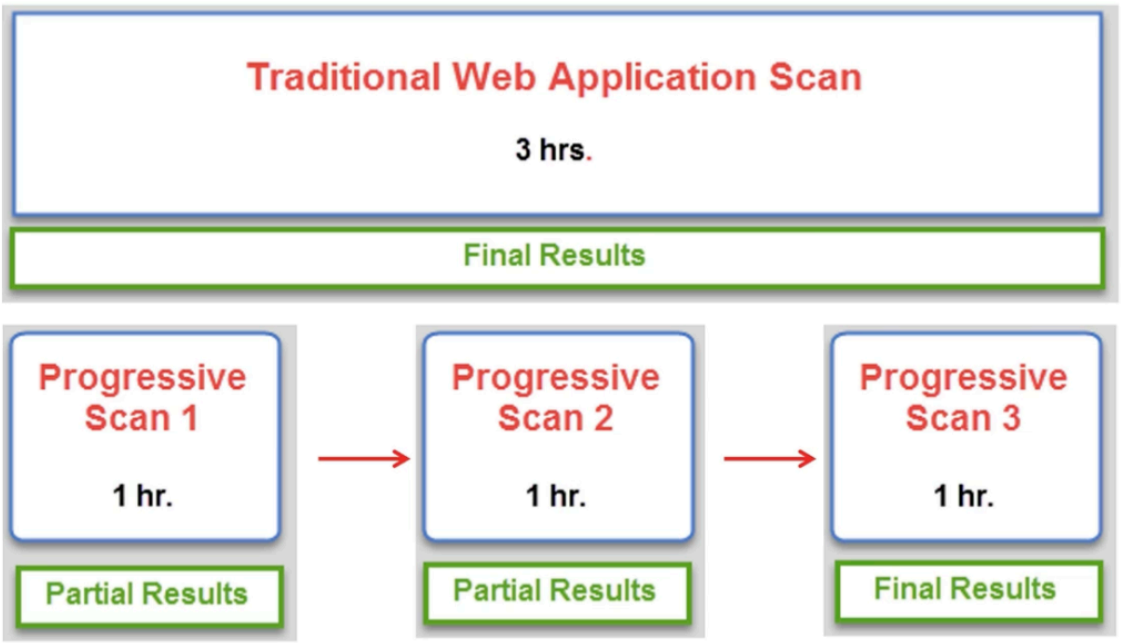


Selenium スクリプト

Selenium スクリプトは、従来のクローラーSelenium とは異なり、**動的なやり取りや JavaScript レンダリングを必要とするウェブアプリケーションのクロールに使用**されます。重いウェブサイト上で実際のユーザーの動作を模倣して JavaScript を扱う方法です。例えば、あなたのウェブアプリケーションで、ユーザーが動的にウェブアプリケーションとインタラクションする必要がある新しい機能を追加しました。その場合は、Selenium スクリプトを利用することができます。Chrome 拡張機能 である Qualys Browser Recorder で、ユーザーがウェブアプリケーションとやり取りする手順を実際に模倣できますのでダウンロードしてください。

プログレッシブスキャン

プログレッシブスキャンが有効になると、**複数のスキャンサイクルを組み合わせて短いスキャンウィンドウを効率的に活用**できます。例を挙げて説明しましょう。Qualys WAS には 1 回のスキャンで最大 8,000 リンクもしくは 24 時間という制限がありますが、プログレッシブスキャンを使うと途中で終了しても続きから再開できるため、大規模なアプリを数回のスキャンで確実に完了できます。



Works best with Frequently Scheduled Scans

WAS:Option Profile

導入

このレッスンでは、Web アプリケーション スキャン中に適用されるプロファイルの詳細、スキャン パラメータ、検索条件の設定など、オプション プロファイルを構成する方法について説明します。

WAS: オプションプロファイル

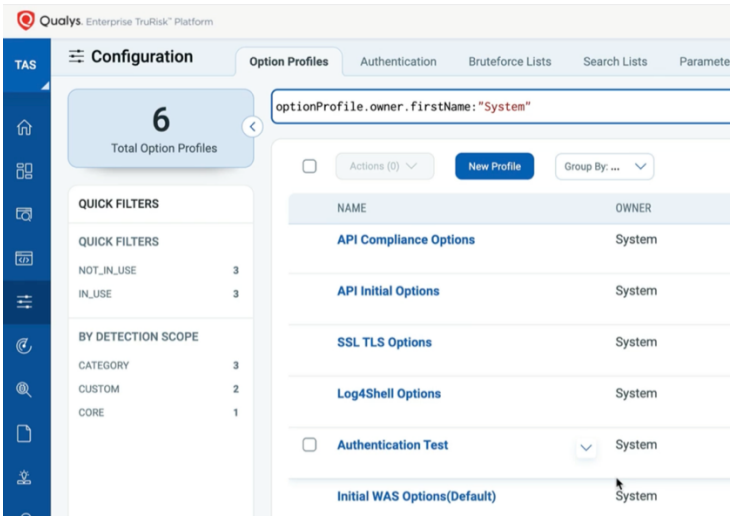
Web アプリケーション スキャン (WAS) のオプション プロファイルは、スキャン パラメータ、検出テスト、エラーしきい値を含むスキャンのブループリントです。

プロファイルの詳細:オプション プロファイルに適用する名前やタグなど、オプション プロファイルの基本的な詳細を指定します。
スキャン パラメーター:フォームの送信、クロール オプション、動作/パフォーマンス設定など、Web アプリケーションのスキャン中に使用されるパラメーターを選択します。
検索条件:検出範囲、機密コンテンツ、キーワード URL 検索など、Web アプリケーションのスキャン中に適用する検索条件を選択します。

オプションプロファイル

このトレーニング ビデオでは、基本的な Web アプリケーションのスキャン オプション プロファイルを設定する方法を学習します。オプションプロファイルは設計図の役割を果たします。スキャナーの挙動、何を探すか、どれだけ深くスキャンするかを定義します。ウェブアプリケーションにおいては、これらの構成を理解することが効率的かつ効果的なスキャンを実行する鍵となります。

TAS > Configuration > Option Profiles タブを開き、optionProfile.owner.firstName: “System”と入力するとシステムが作成したプロファイルがリストを見ることができます。



オプションプロファイルの設定項目

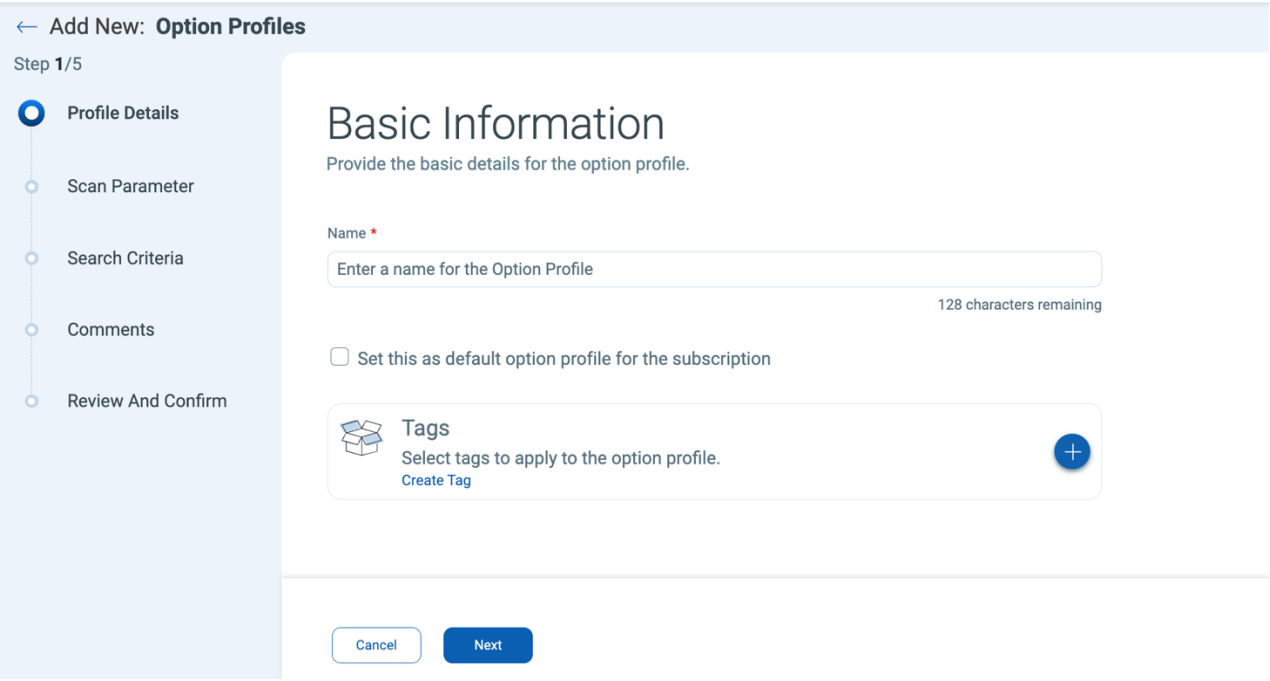
プロファイルの詳細(Profile Details): オプションプロファイルの名前やタグなど、オプションプロファイルの基本情報を入力します。

スキャンパラメータ(Scan Parameter): フォーム送信、クロールオプション、動作/パフォーマンス設定など、Web アプリケーションのスキャン中に使用するパラメータを選択します。

検索条件(Search Criteria): 検出範囲、機密コンテンツ、キーワード URL 検索など、Web アプリケーションのスキャン中に適用される検索条件を選択します。

オプションプロフィールを作成したい場合は、**新しいプロフィール**ボタンをクリックします。

Step 1/5 オプションプロファイルの基本情報： オプションプロファイルのカスタム名前を入力します。次に、このオプションプロファイルをサブスクリプション全体のデフォルトに設定できるチェックボックスがあります。オプションプロファイルにタグを付けて、機能や場所、または提供したい条件に従って整理できます。



Step 2/5 スキャンパラメータ: From Submission:

1. None（フォームを送信しない）

例：予期しない変更（アカウント削除・取引実行など）を避けたい 本番や機密性の高い環境に最適

2. POST のみ送信

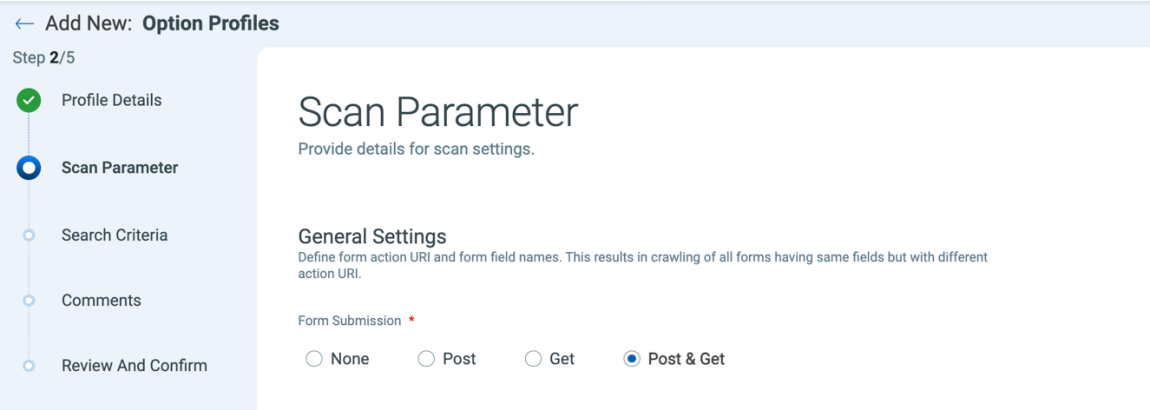
POST はログイン・登録・決済など、データをサーバーに送る操作に使われる
例：銀行アプリのログインフォームをテストしたい場合に有効

3. GET のみ送信

GET メソッドのフォーム（URL にデータを付与するタイプ）のみ送信
例：商品検索ページだけクロールしたいとき、不要なデータ変更を避けられる

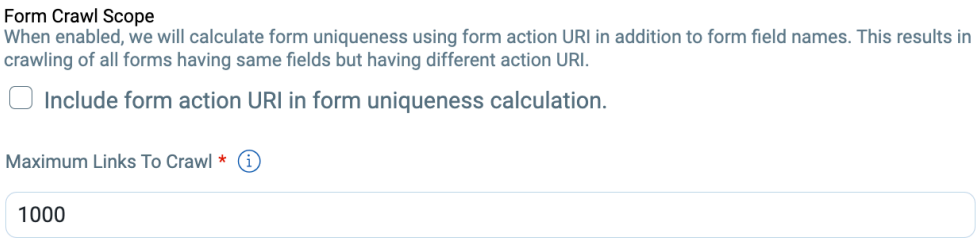
4. POST と GET（両方送信）

テストやステージング環境など、すべてのフォーム動作を確認したい場合に最適
例：医療ポータルのテスト環境で、ログイン・登録・検索をすべてスキャンしたいケース



フォームクロール範囲(Form Crawl Scope)

こちらは、クロールできるリンクの上限値を設定します。 スキャン中にクロールするリンクやフォームの最大数を設定でき、1 つのウェブアプリケーションあたり **8000 リンクに制限**できます。クロールできるリンクの最大数が 8000 リンクを超える場合は、ぜひご注意ください。その後、スキャンが急に終了し、部分的なスキャン結果になります。



ユーザーエージェント(User Agent)

スキャナーは通常「**汎用ユーザーエージェント**」を使いますが、アプリによってはユーザーエージェントによってコンテンツを出し分けたり、アクセスをブロックする場合があります。

例えば **モバイル専用の Web アプリ**では、デスクトップ用ユーザーエージェントのままだと画面が読み込まれず、空白ページやリダイレクトが返されてスキャンが進まないことがあります。

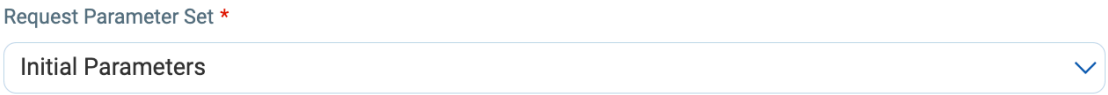
この問題を避けるために、スキャナーのユーザーエージェントを **モバイルブラウザに偽装させる**ことができます。
例：「iPhone のブラウザです」とアプリに伝えることで、モバイル専用コンテンツを正しく読み込めるようになります。

尚、Web アプリケーションのユーザーエージェント文字列を確認するには、任意のブラウザから Web アプリケーションにアクセスし、リクエストヘッダーからユーザーエージェント文字列をコピーしてください。

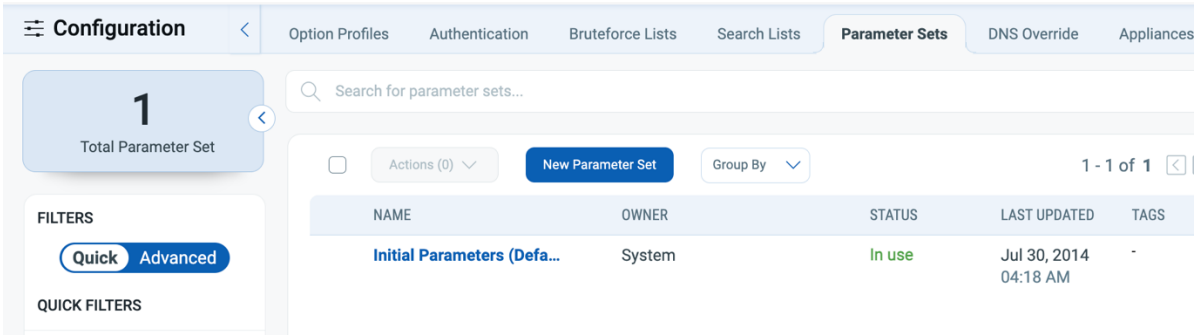
Custom Parameter:

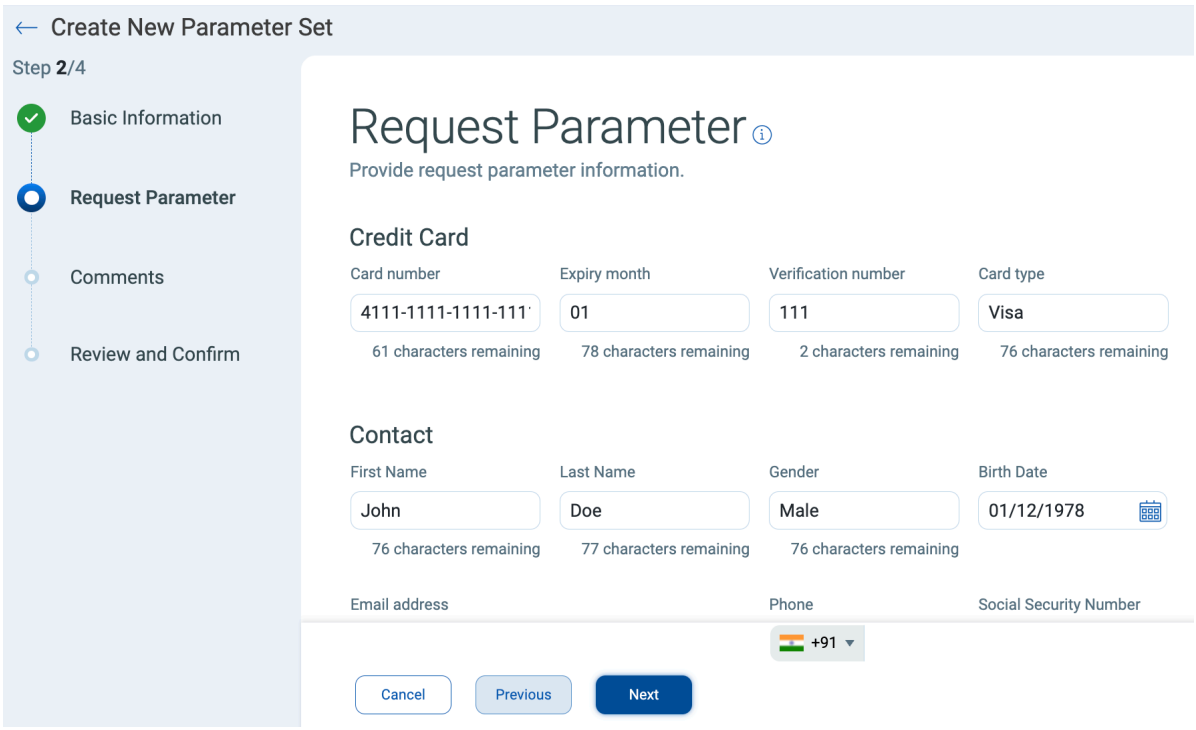
さらに、アプリによっては**入力値に応じて異なる応答を返す**ことがあります。
（例：国選択のドロップダウンで特定の値を選んだときだけ表示される情報など）

このような場合は、**カスタムパラメータ値**を設定することで、スキャナーが条件付き応答を見逃さないようにできます。



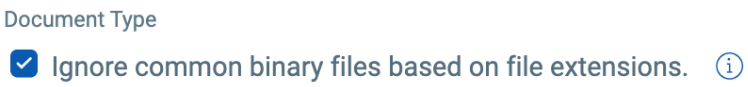
カスタムパラメータは以下の通り TAS > Configuration > Parameter Sets にて作成できます。





Document Type:

スキャン設定の「Document Type(文書の種類)」では、**デフォルトでバイナリファイルを無視する**仕組みになっています。
理由：目的は「ウェブアプリのスキャン」であり、関連する文書ファイルそのものを読む必要がないため。
ただし、必要であれば チェックを外してバイナリも含めて処理可能です。
その場合は注意が必要で、**スキャン時間が大幅に長くなる**ことがあります。



強化クロール（Enhanced Crawl）：

隠れたリンクや見落としたリンクを見つけるための機能です。 この機能を有効にすると URL をディレクトリ単位で切り詰め（ディレクトリチョッピング）、上位ディレクトリを順にクロールして**新しいリンクを発見**します。 これにより脆弱性につながる潜在的なリンクをより多く探索する事ができます。

例えば、クロール中に以下のリンクが見つかったとします。

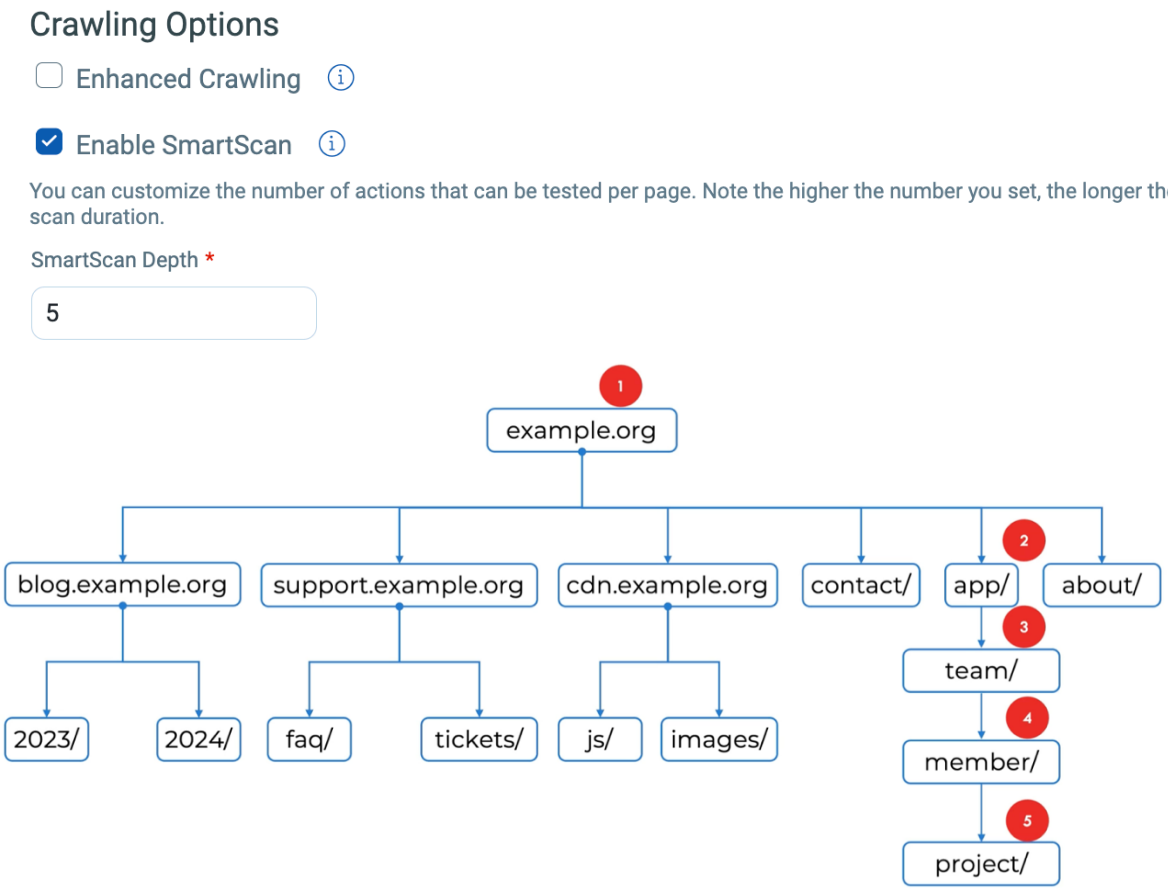
https://www.example.com/foo/abc/xyz/register.php

拡張クロールが有効になっている場合、まず **https://www.example.com/foo/abc/xyz** にリクエストを送信します。

次に、URL からディレクトリ「xyz/」を削除し、**https://www.example.com/foo/abc/** をクロールします。 その後、さらに「abc/」を削除し、**https://www.example.com/foo/** をクロールします。

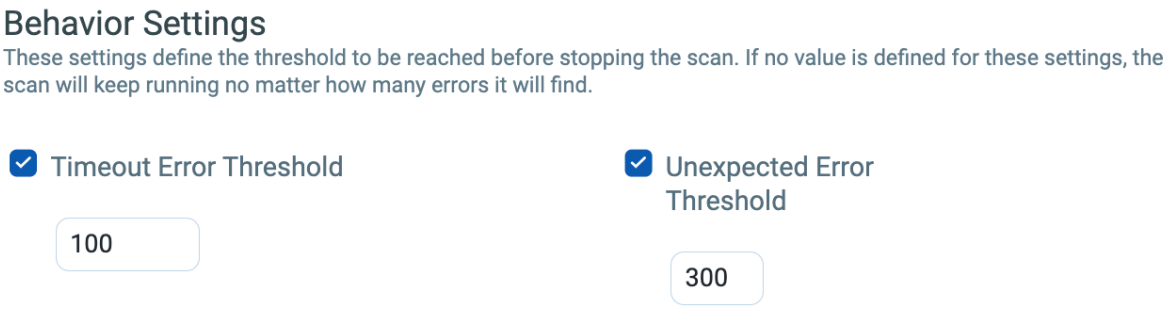
スマートスキャン(Enabled SmartScan）と AJAX 深度

強化クロール後に有効化される高度なクロール機能です。この機能を有効化するとフレームワーク依存のアプリ（Ajax など）をより深くテストで来ます。スキャン深度とは、「トップページから何階層先までリンクを辿るか」という設定で、例：深度 5 → 最大 5 段階先までクロールできます。



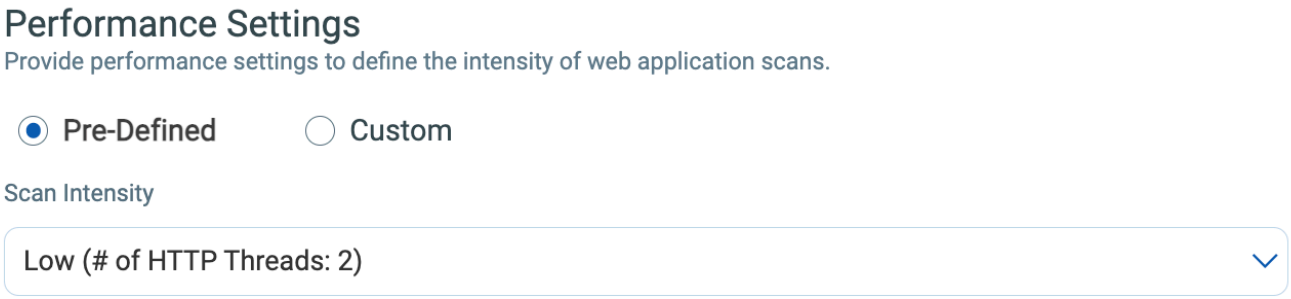
スキャン動作（タイムアウト・エラー閾値）

- **タイムアウト閾値(Timeout Error Threshold)**：スキャンを止めるまでに許容される最大タイムアウト回数。
- **予期せぬエラー閾値(Unexpected Error Threshold)**：許容されるエラー回数。
  → どちらも無駄なリソース消費を抑える仕組み。



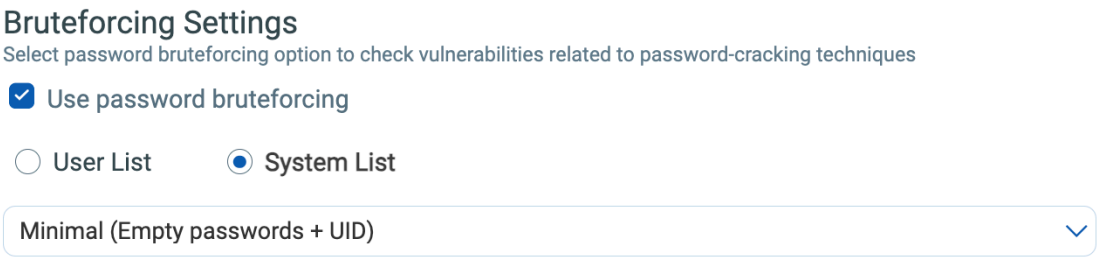
パフォーマンス設定

- **スキャン強度**をネットワーク環境に合わせて調整可能（最大～最小）。
- カスタム設定では、**リクエスト間のスレッド数や遅延調整**ができ、本番環境へ負荷をかけすぎないようにバランスを取れる。



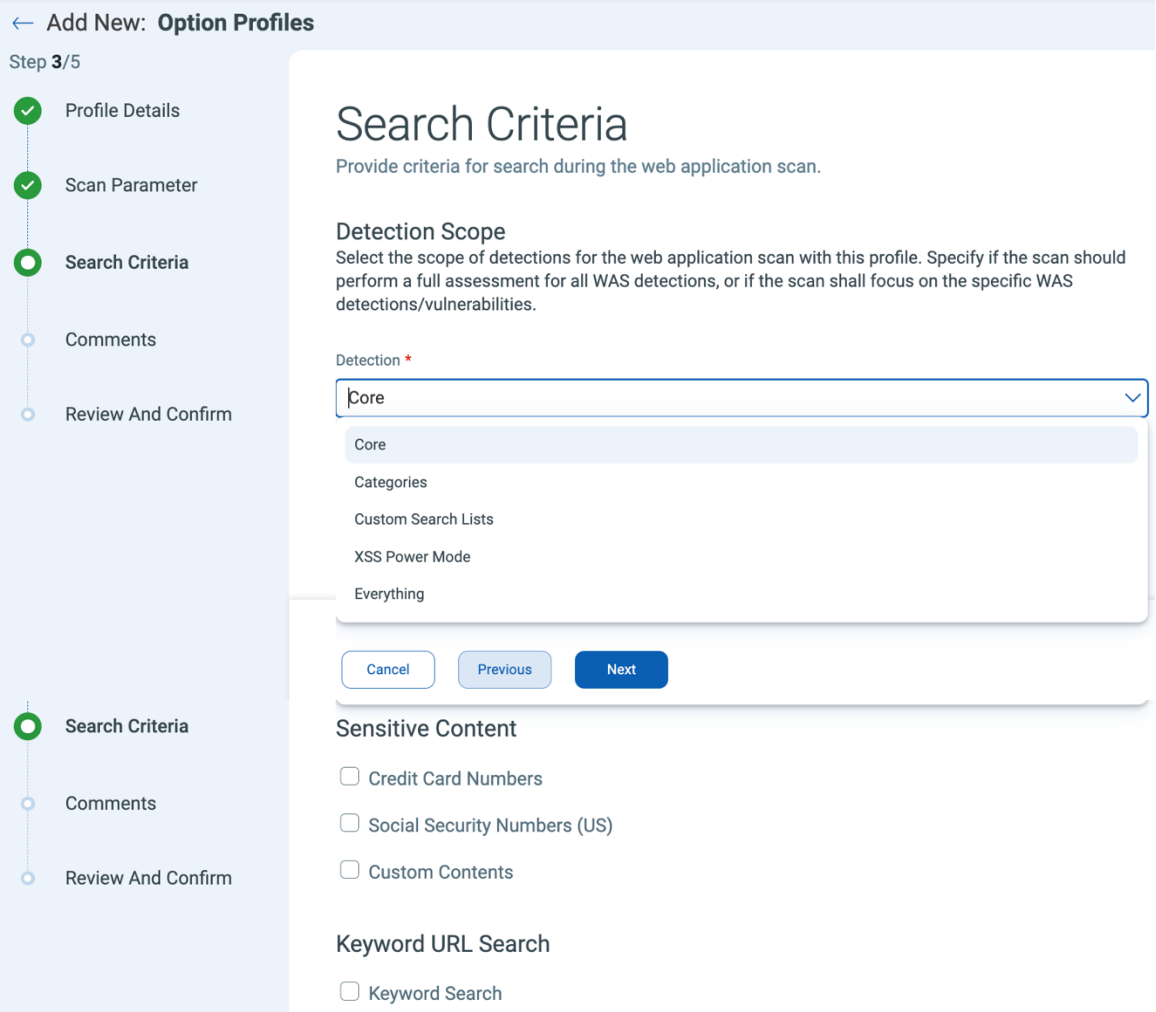
Bruteforcing 設定

パスワードクラッキング技術に関連する脆弱性をチェックするには、「**パスワードブルートフォースを使用する**」チェックボックスをオンにします。デフォルトでは、「パスワードブルートフォースを使用する」チェックボックスと「最小」オプションのシステムリストが選択されています。



Step 3/5 Search Criteria(検索条件):

検出範囲、機密コンテンツ、キーワード URL 検索など、Web アプリケーションのスキャン中に適用される検索条件を選択します。



Qualys Web Application Scanning：追加設定および認証レコード構成について

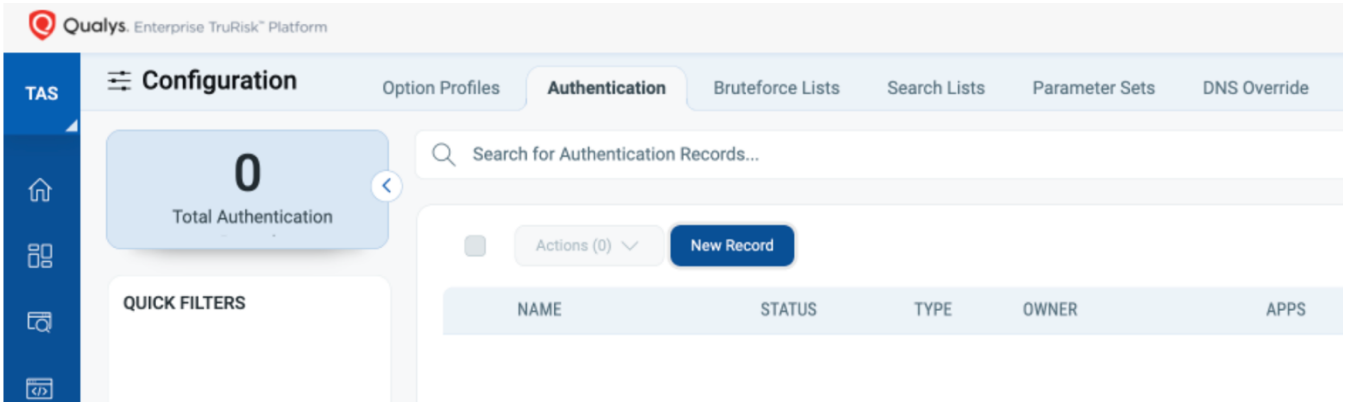
本章では、基本的な Web アプリケーション設定に続き、認証レコードの構成方法について解説します。Qualys Web Application Scanning（WAS）では、複数の認証方式に対応しており、これらを適切に設定することで、認証後の領域を含めたより包括的な脆弱性評価が可能になります。

WAS:追加構成

認証記録
ヘッダー注入
API エンドポイント定義

除外リストの設定
デフォルトの DNS オーバーライド
冗長リンク
パスファジングルール
フォームトレーニング
マルウェア監視

認証レコードの作成は TAS > Configuration > Authentication > New Record ボタンを押します。



認証方式の概要

1. フォーム認証（Form Authentication）

最も一般的な Web アプリケーションの認証方式であり、HTML フォームを利用したログイン手法です。標準的なユーザー名・パスワード形式に加え、非標準的なフィールドが存在する場合は、カスタムフォームを用いて入力項目を定義できます。また、複数ステップを含む複雑なログインや JavaScript 主導のログイン処理が必要な場合は、**Selenium スクリプト**を利用してログインプロセスを自動化できます。

1.1 HTML フォーム認証の設定

ログインページがアプリケーションと同一ドメイン上にある場合に最適です。
設定時には以下を指定します。

- ログイン URL
- テストアカウントのユーザー名・パスワード

ログインフォームが別ドメイン上にある場合は、スキャン時に到達できるよう**「明示的にクロールする URL」**に追加する必要があります。

1.2 カスタムフォーム認証

標準以外の入力項目（例：トークン、ドロップダウン、隠しフィールド等）を持つ場合に使用します。複数の認証情報セットを定義することも可能です。

1.3 Selenium スクリプト認証

以下のような複雑なログインフローに最適です。

- 複数ページにわたるログイン
- CAPTCHA 回避（ただし CAPTCHA の有無によっては自動化困難）
- JavaScript が動的に描画するページでのログイン

Selenium IDE を使用してログイン操作を記録し、生成されたスクリプトをアップロードします。また、ログイン成功を識別するための「マッチ式（Match Expression）」も指定します。

1.4. スキャン結果の確認

認証の成功・失敗は以下の QID で確認できます。

- **QID 1507**：フォーム認証のステータス
- **QID 15010**：Selenium スクリプト認証の成否（どのページで失敗したかも特定可能）

2. OAuth2 認証

OAuth 2.0 は、現代的な Web アプリケーションや API で広く利用されるトークンベース認証方式です。対応しているグラントタイプ（認可方式）は以下の通りです。

Authorization Code：**サーバーサイドアプリケーション向け。認可コードを利用してトークンを安全に取得。**
例：社内人事ポータルが Google Workspace と連携し、認可コードからアクセストークンを取得。
Implicit：**ブラウザベース、クライアントサイドアプリ向け。クライアントシークレットを保持できない環境で利用。**
例：ダッシュボードに埋め込まれた JavaScript SPA が一時的にユーザーデータを取得。
Client Credentials：**マシン間通信など、ユーザー関与のないバックエンド処理に最適。**
例：自動ツールが API へ定期的に接続し、ログや設定情報を取得。
Resource Owner Password Credentials：**ユーザー名・パスワードを直接アプリに渡す方式。信頼できるアプリのみで使用。**
例：デスクトップアプリがブラウザを挟めない環境で直接ユーザー資格情報を受け取り認証。

3. サーバーレコード認証

Basic 認証：資格情報はエンコードのみで暗号化されません。古い Web サーバーなどで利用されることがあります。
Digest 認証：Basic 認証に比べ、ハッシュ化によりセキュリティが強化されています。
NTLM 認証：主に Windows ベースの企業環境で使用される方式です。

4. ベストプラクティス:非特権アカウントで認証スキャンを行う

認証スキャンには **非特権アカウントの利用** が推奨されます。理由は以下の通りです。

最小権限の原則に基づき、必要最低限のアクセス権のみを付与することで、操作リスクを低減します。
特権アカウントではデータ変更・削除や機密情報アクセスが可能なため、スキャン中に意図せず状態を変更するリスクがあります。
本番環境に対する影響を最小化できます。

Header Injection

ヘッダーインジェクション設定は、スキャナーが Web アプリケーションへアクセスする際に **カスタム HTTP ヘッダー を送信**する必要がある場合に使用します。

この機能は次のようなケースで特に有効です：

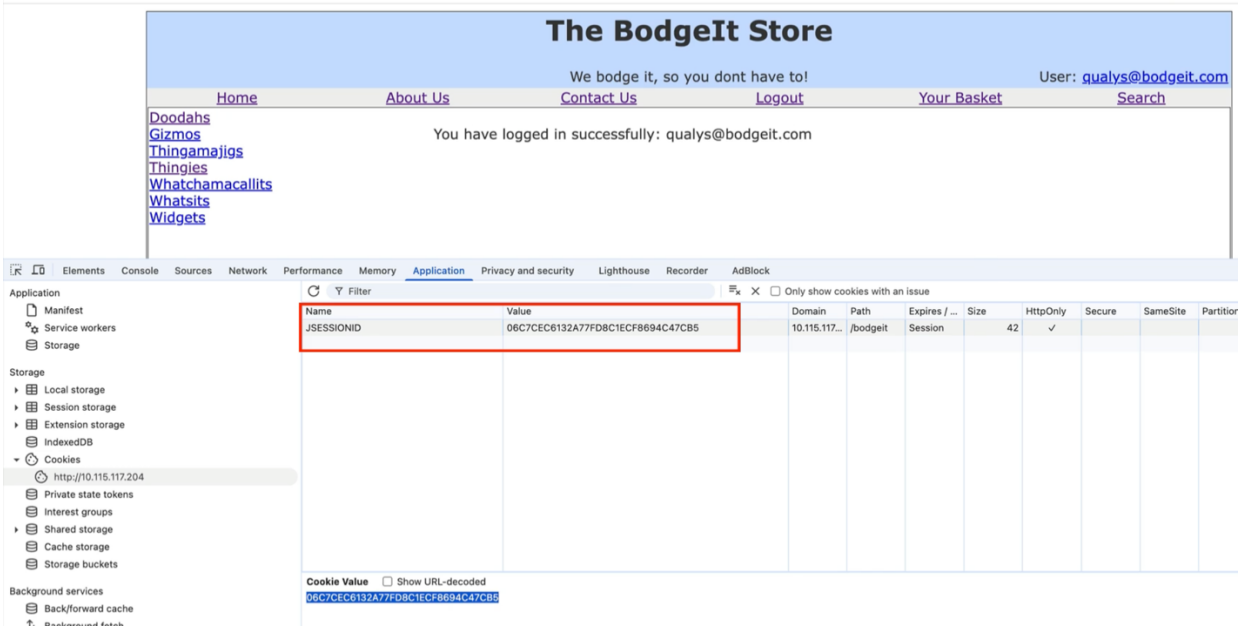
- セッションベースのアクセストークンが必要なアプリケーション
- ログイン時にブラウザ同等の振る舞いを必要とするアプリケーション

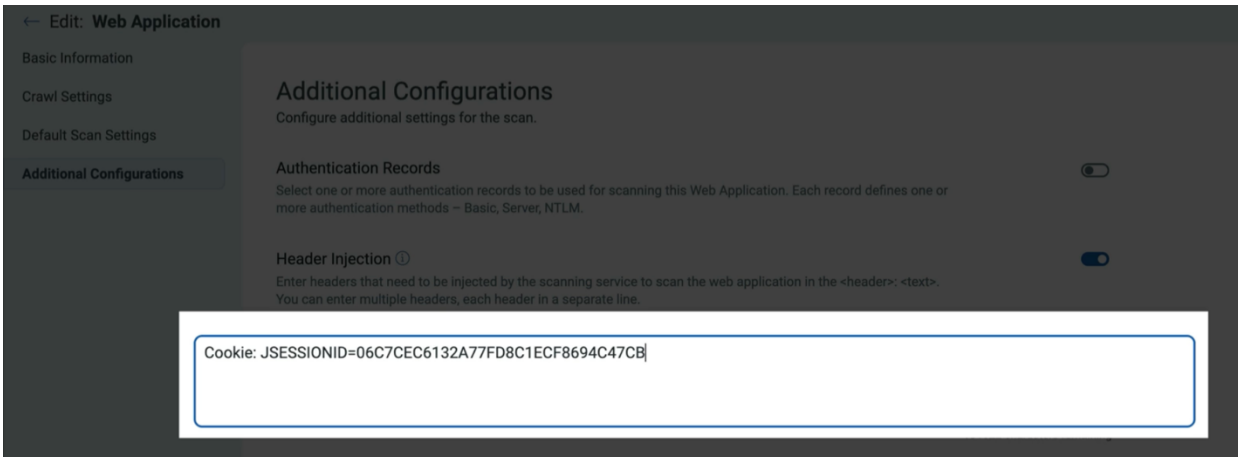
例 1: 複雑なログイン (例: CAPTCHA を使用) を回避するには、次のようなセッション Cookie を挿入します:
Cookie: mwf_login=2-e3b930b2cf6549d0351346d3cf56e9ae

例 2：Budget Store アプリケーションの場合
アプリケーションがログイン後に Session ID を発行する場合、以下の手順でその値を取得します。

1. Web アプリケーションへ手動でログイン
2. ブラウザでデベロッパーツールを開く
3. **Applications（または Storage）タブ → Cookies** を選択
4. Session ID（例：`session_id`）を確認
5. 該当値をコピーし、ヘッダーインジェクション欄へ次の形式で入力

Cookie: session_id=<取得した値>





API Endpoint Definition

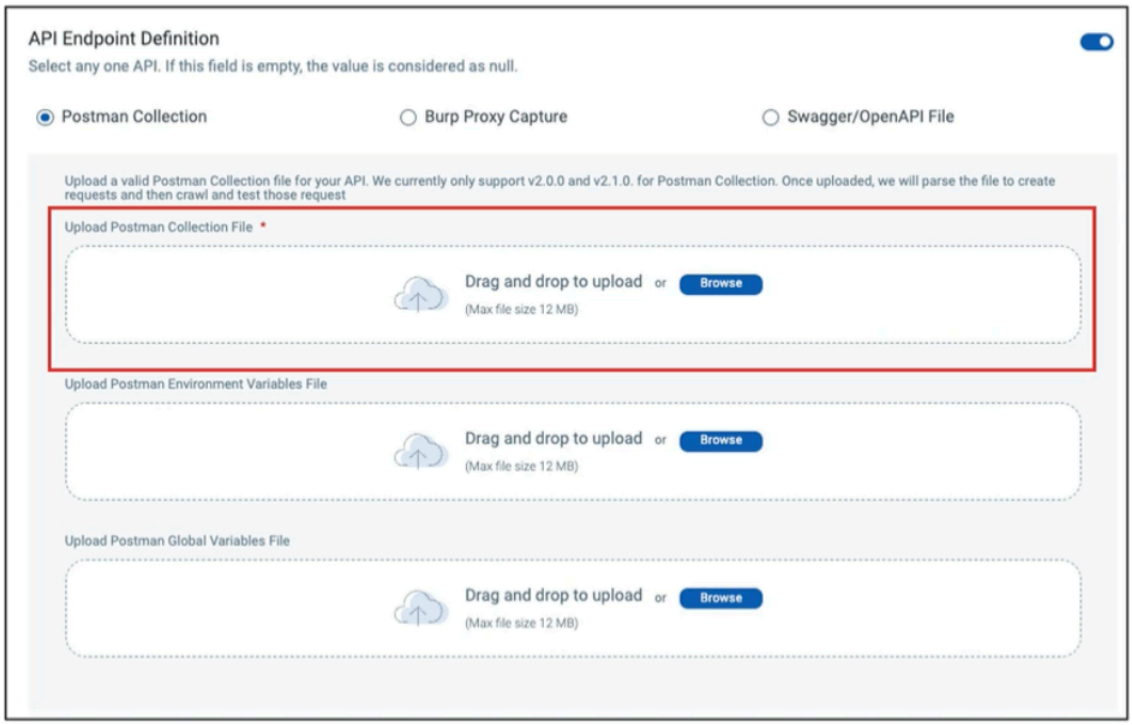
Web アプリケーション同様、API も HTTP 上で動作するため様々な攻撃に晒されます。
近年のアプリケーションは API に強く依存する傾向があり、攻撃対象となりやすいため、適切なスキャン設定が重要です。
Qualys WAS では、API エンドポイントを以下の方法で定義できます。

2.1 Postman コレクションのアップロード

Postman は API の設計・テストに広く利用されるツールです。
Postman コレクションには以下の情報が含まれます：

- エンドポイント URL
- HTTP メソッド
- 認証情報
- 必要なパラメータ

既存のコレクションを **JSON 形式でエクスポート**し、そのまま Qualys に取り込むことで、開発チームが日常的に使用している API テスト資産を再利用できます。



2.2 Web プロキシ（Burp Suite）経由のキャプチャ

Burp Suite は、ブラウザと Web アプリケーション間の HTTP/HTTPS 通信をインターセプト・解析 するためのツールです。

ブラウザまたは Postman を使って API と通信すると、Burp Suite は以下の情報を記録します：

- エンドポイント URL
- HTTP メソッド
- HTTP ヘッダー
- パラメータ
- 認証トークン

  など

キャプチャ結果は **XML 形式でエクスポート**でき、Qualys にアップロードすることで、API の構造を正しく認識したうえで体系的な脆弱性検査を実施できます。
Swagger や Postman ファイルが存在しない場合に特に有効な方法です。

2.3 Swagger / OpenAPI ファイルの利用

Swagger（OpenAPI Specification）は、REST API を構造化されたマシンリーダブルな形式で定義する標準仕様です。
Qualys WAS は以下に対応しています：

- **Swagger 2.0**
- **OpenAPI（JSON / YAML 形式）**

ファイルがスキャナーから参照可能な場所に配置されていれば、自動的にエンドポイントを解析し、脆弱性検査を実行します。

まとめ

ヘッダーインジェクションや API エンドポイント定義は、Web アプリケーションおよび API を包括的にスキャンするための重要な設定です。
適切に構成することで、認証後の領域や API エコシステム全体に対するセキュリティ可視性を大きく向上できます。

Qualys Web Application Scanning：除外リスト設定と DNS Override の構成方法

本トレーニングでは、スキャンをより安全かつ効率的に実行するための **除外リスト設定（Exclusion Lists）** と、テスト／ステージング環境向けに利用できる **DNS Override レコード の設定方法**をご紹介します。

1. 除外リスト（Exclusion Lists）

除外リストは、スキャンの対象範囲を細かく制御し、**意図しない領域へのアクセスやアプリケーションへの影響を防ぐ**ための重要な設定です。

1.1 Allow List（許可リスト）

Allow List（ホワイトリスト）は、**「スキャナーがアクセスを許可された URL のみをスキャンする」**
という厳密な制御を行うためのリストです。
これは、プライベートイベントの“ゲストリスト”のようなもので、ここに記載された URL 以外はすべて無視されます。
推奨利用例：

- **スキャン範囲を厳密に限定したい場合**
- **影響のある領域を明確に制御したいテスト環境**

1.2 Exclude List（除外リスト）

Exclude List（ブラックリスト）は、**スキャン対象から除外したい領域を設定**します。
例：

- 管理者向けページ（/admin）
- 決済ページ（/payment）
- 機密性が高く、スキャンによる影響が懸念されるパス

Allow List にも登録されている場合を除き、これらの URL にはスキャナーはアクセスしません。

1.3 POST Data Exclude List（POST データ除外リスト）

たとえば「お問い合わせフォーム」など、POST 操作によりメール送信が行われる箇所をスキャン対象に含めてしまうと、

- 実際にメールが大量送信される
- アプリケーションに不要な負荷がかかる

といった問題が発生する可能性があります。
これらのフォーム送信を回避するため、該当 URL を POST Data Exclude List に登録します。

1.4 Logout Regular Expression（ログアウト正規表現）

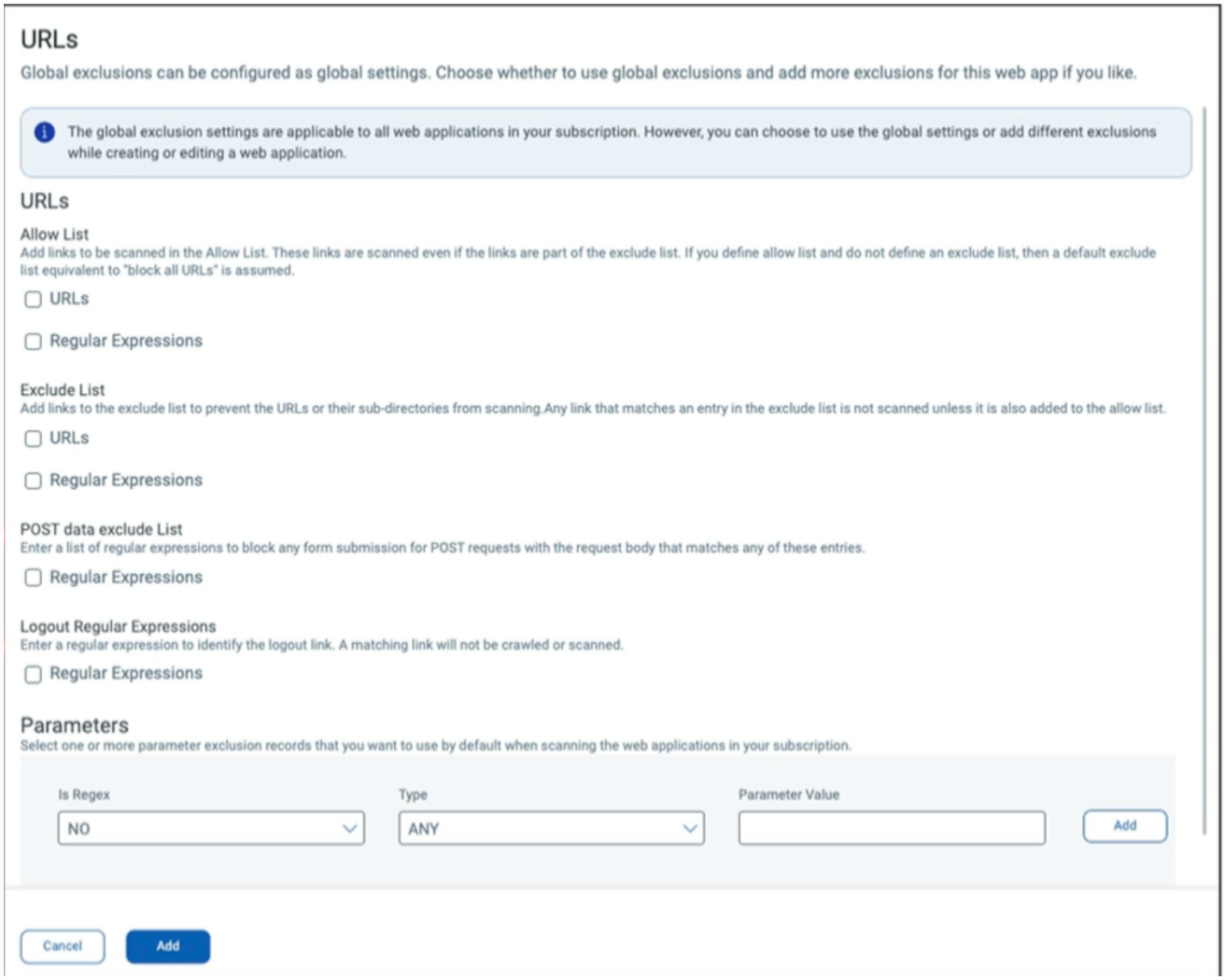
スキャナーが誤ってログアウトリンクをクリックすると、認証スキャンが途中で失敗する可能性があります。
ログアウトに該当する文字列パターン（例：logout, signout）を正規表現で指定することで、スキャナーがそのリンクを避けるよう制御できます。

1.5 Parameter Exclusion List（パラメータ除外リスト）

特定のパラメータ（例：トークン、機密データ、セッション情報）をスキャン対象から除外するための設定です。
以下の種類を選択可能です：

- URL パラメータ
- POST データ
- Cookie

正規表現にも対応しており、特定パターンを一括で除外できます。



＜まとめ＞

除外リストを適切に設定することで：

- 高リスク領域への意図しないアクセスを防止
- フォーム送信など副作用のある操作を回避
- スキャンを効率化し、必要な範囲に集中

といったメリットが得られます。

2. DNS Override レコード

次に、テスト環境やステージング環境のスキャンに便利な DNS Override について解説します。

2.1 DNS Override とは

通常、スキャナーは標準 DNS を使用してドメイン名を解決します。たとえば budget.com をスキャンすると、DNS が返す IP に接続します。しかし、次のようなケースでは DNS Override が必要になります：

- **ステージング環境には DNS レコードがない**
- **テスト／QA／本番で同じドメイン名だが IP が異なる**
- **内部ネットワーク上でのみアクセス可能な環境をスキャンしたい**

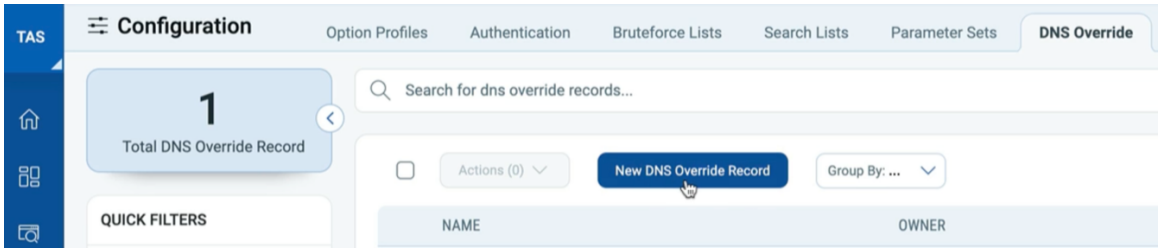
DNS Override を使用すれば、**ドメイン名 → 特定の IP アドレス** を手動でマッピングし、スキャナーに正しい接続先を明示できます。

2.2 DNS Override の設定手順

1. **Configurations** タブへ移動
2. **DNS Override → New DNS Override Record** を選択
3. レコード名を入力
    - ホスト名（例：staging.budget.com）
    - 解決させたい IP アドレス

    を設定
4. レコード作成後、Web アプリケーション設定の Additional Configurations 内で、当該 DNS Override を有効化

これにより、スキャナーは DNS を参照せず、手動で指定した IP へ直接アクセスします。



3. まとめ

除外リストと DNS Override を活用することで、Qualys WAS のスキャンはより：

- 正確に
- 安全に
- 効率的に

実施できるようになります。

スキャン最適化のための Redundant Link と Path Fuzzing Rules の活用

本セクションでは、Web アプリケーションスキャンを効率化し、不要なクロールを排除するための **Redundant Link（冗長リンク） と Path Fuzzing Rules（パスファジングルール） の設定**について解説します。

1. Redundant Link（冗長リンク）の設定

EC サイトなどでは、同一のコンテンツに対して多数の異なる URL が生成されることが一般的です。
たとえば、商品カタログにおいて：

- 画像ごと
- カラーバリエーションごと
- 表示形式ごと

に異なる URL が生成されることがありますが、これらは 最終的に同一の商品ページへ到達するケース が多く見られます。
このような URL をすべてクロールすると、次のような影響が出ます：

- スキャン時間の増大
- 必要な領域への到達が遅延
- 無駄なリソース消費

そこで、**冗長リンクのパターンを定義し、最大許可件数（Max Occurrences） を設定する**ことで、同一パターンに該当する URL のクロールを制御し、効率的なスキャンが可能になります。
これにより、主要なページを見逃すことなく、スキャンの集中度と効率性を大幅に向上させることができます。

2. Path Fuzzing Rules（パスファジングルール）

Path Fuzzing Rules は、**URL リライト方式を採用する Web アプリケーション** に対して効果的な機能です。
従来型の URL は、以下のようにクエリパラメータによって可変値が渡されます：

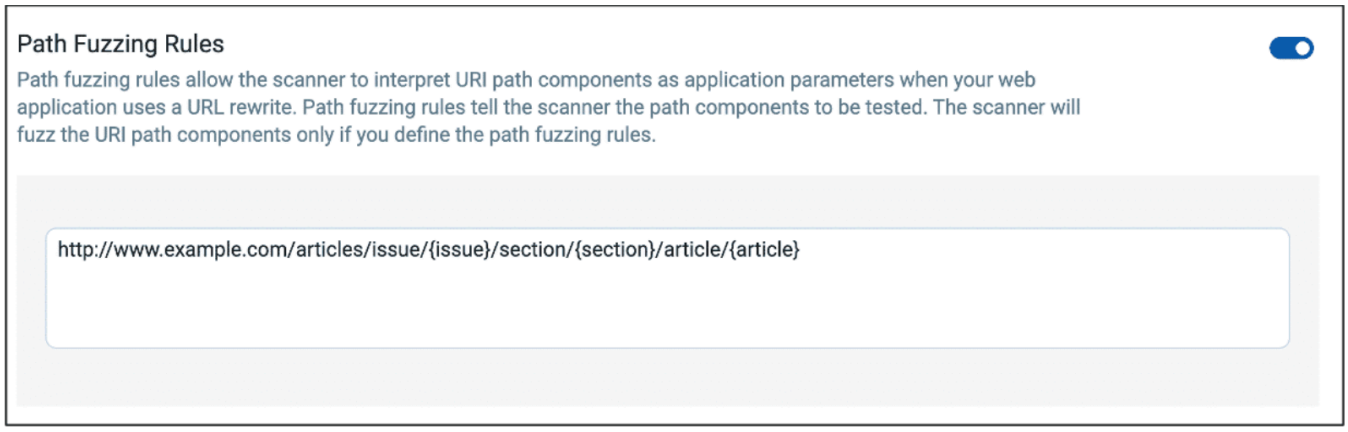
```
/issue?section=sports&article=28
```

この場合：

section と article がパラメータ名

sports と 28 がパラメータ値

であり、スキャナーはこれらの値が動的であることを理解し、適切なペイロード注入を実施します。



2.1 URL リライト形式の場合

一方で、次のような URL リライト構造では、パラメータがパスに埋め込まれています：

/issue/sports/article/28

この形式では、どの部分がパラメータ名で、どこが動的値なのかをスキャナーが自動的に判断することが困難です。

2.2 パス内の動的値を定義する方法

この課題を解決するために、次のように 動的部分を波括弧（{}）で囲む 定義を行います：

`/issue/{sports}/article/{28}`

このように定義することで、スキャナーは：

- 動的パラメータの位置と範囲を正確に識別
- 適切なペイロード注入を実施

できるようになり、リライト形式の URL に対しても完全な脆弱性評価が可能となります。

まとめ

- **Redundant Link**
  → 重複する構造の URL パターンを制御し、スキャン効率を向上させる
- **Path Fuzzing Rules**
  → URL リライトを使用するアプリケーションに対し、動的パラメータを明示して正確なテストを実行する

これらの設定を活用することで、スキャン時間の最適化と網羅性の両立が実現できます。

フォームトレーニング（Form Training）とマルウェア監視（Malware Monitoring）について

本トレーニングでは、Web アプリケーションのセキュリティスキャンを強化する 2 つの重要な機能である フォームトレーニング と マルウェア監視 について解説します。

1. フォームトレーニング（Form Training）

フォームトレーニングは、**Web クロール時に使用されるフォーム入力値**を事前に定義するための機能です。
スキャン対象のアプリケーションが 特定の形式の入力値（例：4 桁 PIN）を要求する場合 や、必須項目が存在するフォーム に特に有効です。
この機能を利用することで、スキャン中のフォーム送信がより正確かつ完全に実行され、実際のユーザー操作に近い動作が再現できます。

主な機能

- フィールド名と値の個別設定
  例:
  - pin=1234
  - zipcode=10001
- グローバルフィールドの設定
  特定のフォームに依存せず、名称が一致するフィールドに対して自動的に値を適用します。

これらは、Qualys Browser Recorder や初期パラメータ設定で実現できる動作と同等の効果を持ちながら、より簡潔な構成で実施できます。

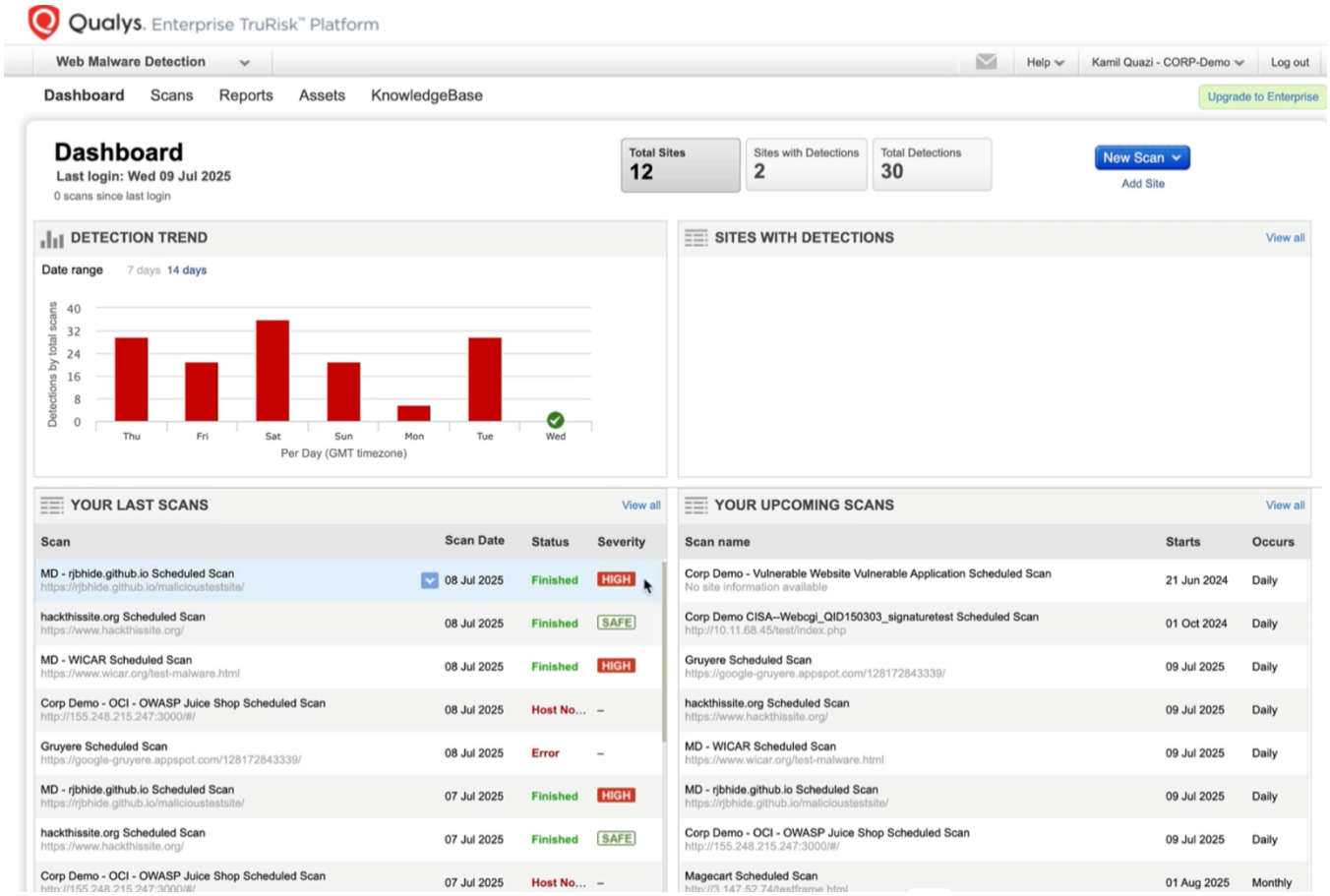
2. マルウェア監視（Malware Monitoring）

マルウェア監視は、外部向け Web サイトを継続的に安全な状態に保つための機能です。 外部公開サイトに対して 定期的なマルウェアチェックを行いたい場合 に最適です。

動作の概要

1. 機能を有効化すると、**数時間以内に初回スキャンが自動実行**されます。
2. 以降は、ほぼ同時刻に **毎日自動スキャン** が実行されます。
3. スケジュール情報や検出結果は **Malware Detection アプリケーション**から確認できます。

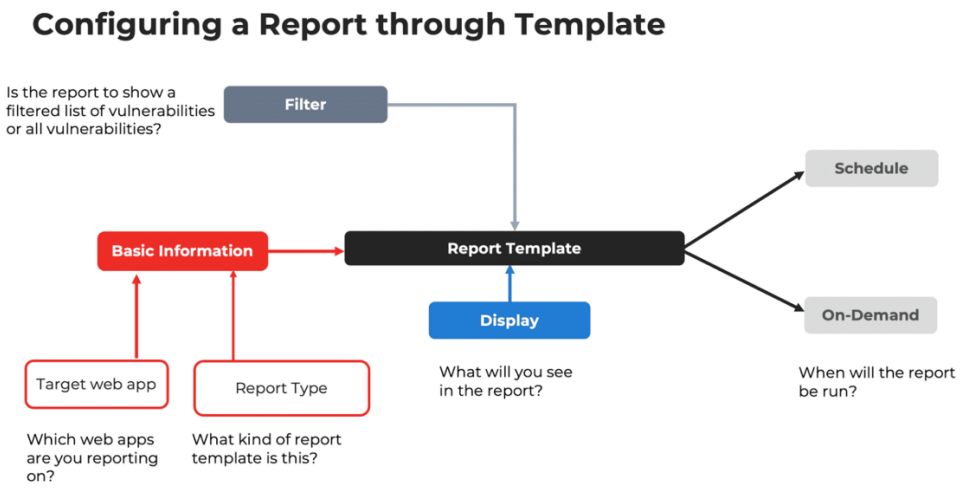
これにより、感染ページや脅威の詳細を迅速に把握できます。



レポート

導入

このレッスンでは、組み込みのレポートテンプレートを使用してレポートを設定する方法を学びます。また、新しいレポートテンプレートの作成、タグを使用したレポートの実行、レポートのスケジュール設定についても学習します。

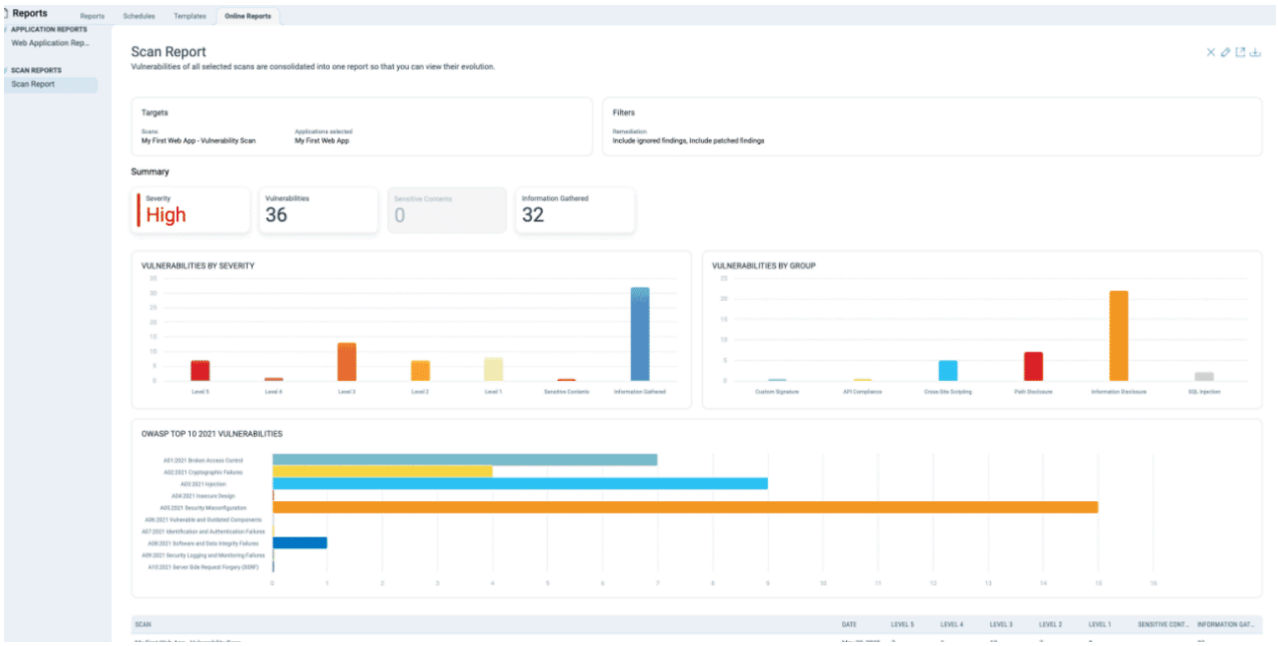


1.1 スキャンレポート

特定のスキャン実行結果に基づくレポート であり、指定した日時のスキャンに限定した脆弱性検出内容を表示します。過去の脆弱性履歴は含まれません。

利用場面

- 特定スキャンの詳細を確認したい場合
- スキャン動作のトラブルシューティング
- 検出内容の検証
- 特定期間中のスキャン範囲の妥当性確認



Web アプリケーションレポート

選択した Web アプリケーションに対し 実行されたすべてのスキャン結果を集約したレポート です。
単独スキャンではなく、複数スキャンをまたぐ全体傾向を把握できます。

単一の Web アプリケーションで実行されたすべてのスキャンを結合します。
脆弱性の履歴とステータスが含まれます (新規、アクティブ、再開、修正済み)。

利用場面

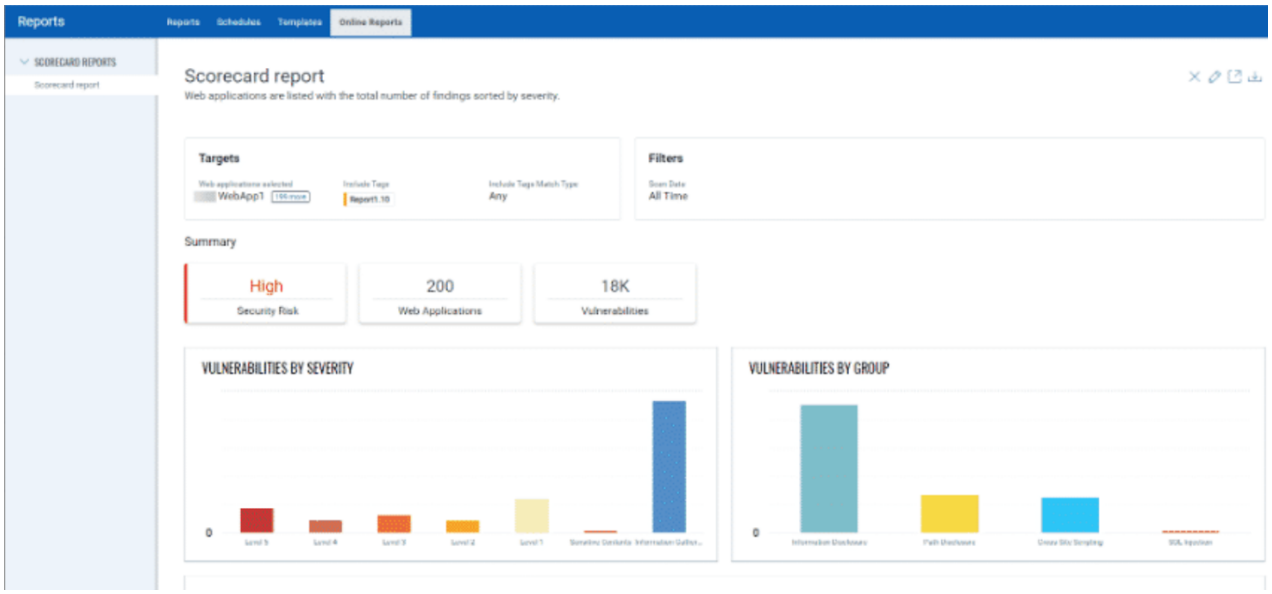
- 脆弱性の長期的な推移を確認
- 再発している問題の特定
- セキュリティレビュー資料の作成
- 全体的なセキュリティ状態の可視化

スコアカードレポート

スコアカード レポートには、選択した Web アプリケーションの結果がインタラクティブなビューで表示されます。
スコアカード レポートには、1 つ以上のタグに関連付けられた Web アプリケーションのセキュリティ ステータスが表示されます。
デフォルトでは、スコアカードレポートには脆弱性が検出されたすべてのウェブアプリケーションが含まれます。タグを選択することで、特定のウェブアプリケーションにレポートを絞り込むことができます。

利用場面

- 経営層（例：CSO）に向けた簡潔なセキュリティ状況報告
- 複数アプリケーション全体のリスク状況の把握
- プレゼンテーション用資料としての活用

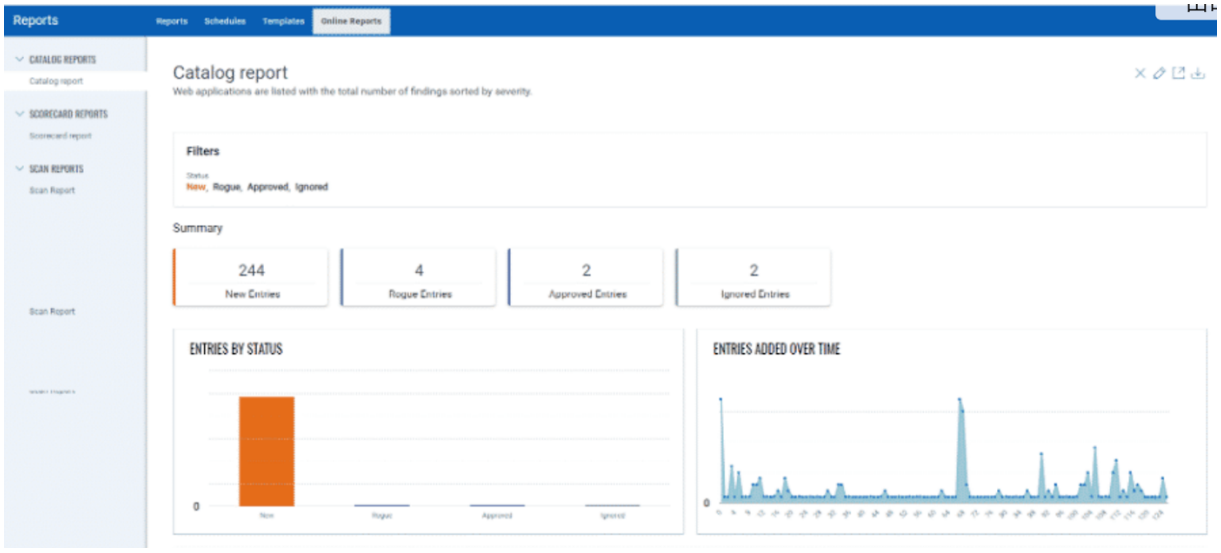


カタログレポート

資産管理や露出状況の把握に役立つ、**マップスキャン・脆弱性スキャンから発見された Web サーバーの一覧** を表示します。

利用場面

- 環境内で検出された Web サーバーの棚卸し
- 発見ベースでの資産可視化
- 新規スキャン後のサーバー一覧作成



2. レポート生成手順（デモ）

以下では、Scan Report を例に具体的な生成手順を解説します。

2.1 レポートテンプレートの確認

1. Total AppSec モジュールのホームページから **Reports** タブへ移動
2. **Templates** をクリック
    - Qualys ではレポート種別に応じたテンプレートが標準提供されています
3. 必要に応じて **New Template** からカスタムテンプレートを作成可能

この例では Scan Report テンプレートを使用します。

2.2 レポートの生成

方法 1：テンプレートから直接実行

- テンプレートの **Quick Actions** メニュー → **Run** を選択

方法 2：レポートメニューから作成

1. **Reports** → **New Report** をクリック
2. レポートタイプに **Scan Report** を選択
3. デフォルトのテンプレートが自動で選択される
4. **Next** をクリックし、対象スキャンを選択

スキャンの指定方法

- **Scan Job を直接指定**
- **Web アプリケーションを指定（最新スキャンを自動反映）**

今回は個別スキャンを選択し、一覧から対象スキャンを追加します。

5. 設定内容を確認後、**Create Report** をクリック
   → レポートが生成されます。

3. レポート内容の確認

生成された Scan Report では以下の項目を確認できます。

3.1 Appendix：Scan Details（スキャン詳細）

- スキャン所要時間
- 使用された Option Profile の設定内容

3.2 表示フィルターの編集

画面右上の **Edit Filters → Display** から、レポートに含めるグラフや情報を追加できます。

例:

- もっとも脆弱性の多い URL（Top 10 Vulnerable URLs）を追加

3.3 クロール結果の確認

Information Gathered → Scan Diagnostics

- スキャナーがクロールした URL 一覧
- 正しくクロールされているかの検証
- エクスポートも可能

3.4 脆弱性詳細（Vulnerability Details）

脆弱性（例：Cross-Site Scripting）を開くと：

- 脅威内容
- 影響
- 解決策
- 検出時のペイロードと Web アプリのレスポンス（View Details）

を確認できます。
これにより、脆弱性の実在性を精度高く判断できます。

3.5 ダウンロード形式

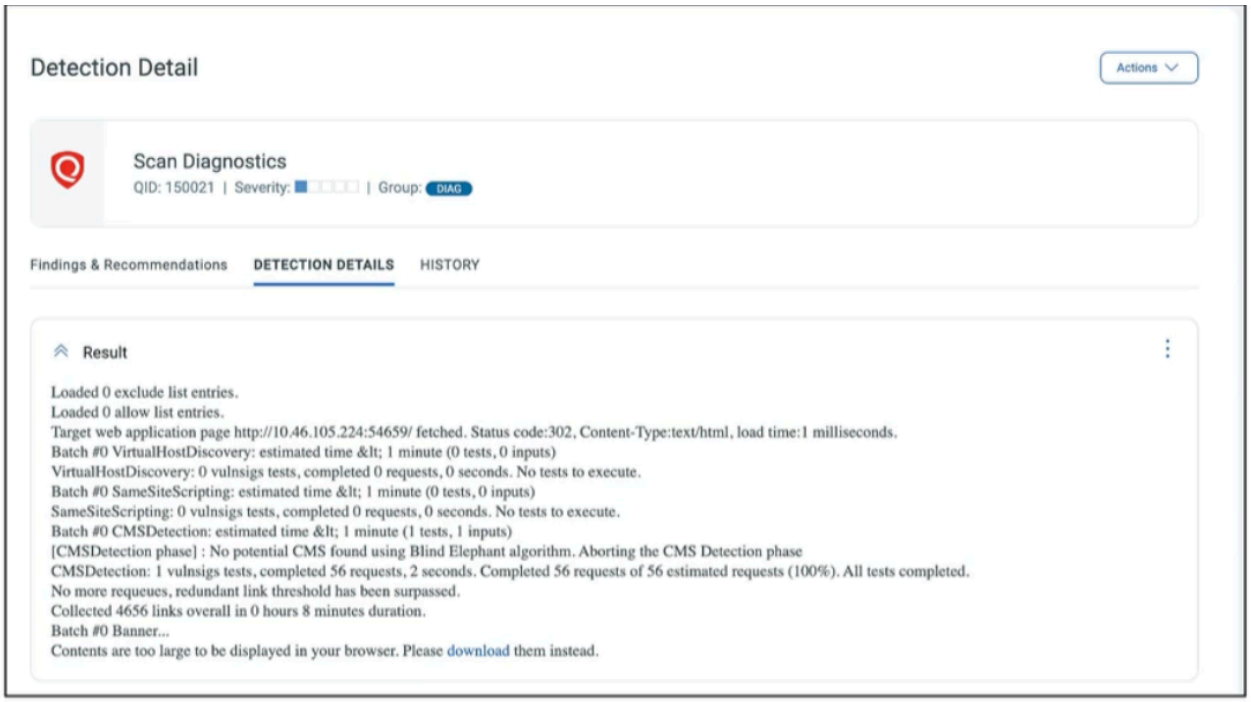
右上の Download メニューより、レポートを以下形式で取得可能：

- PDF
- CSV
- HTML
- XML
  など

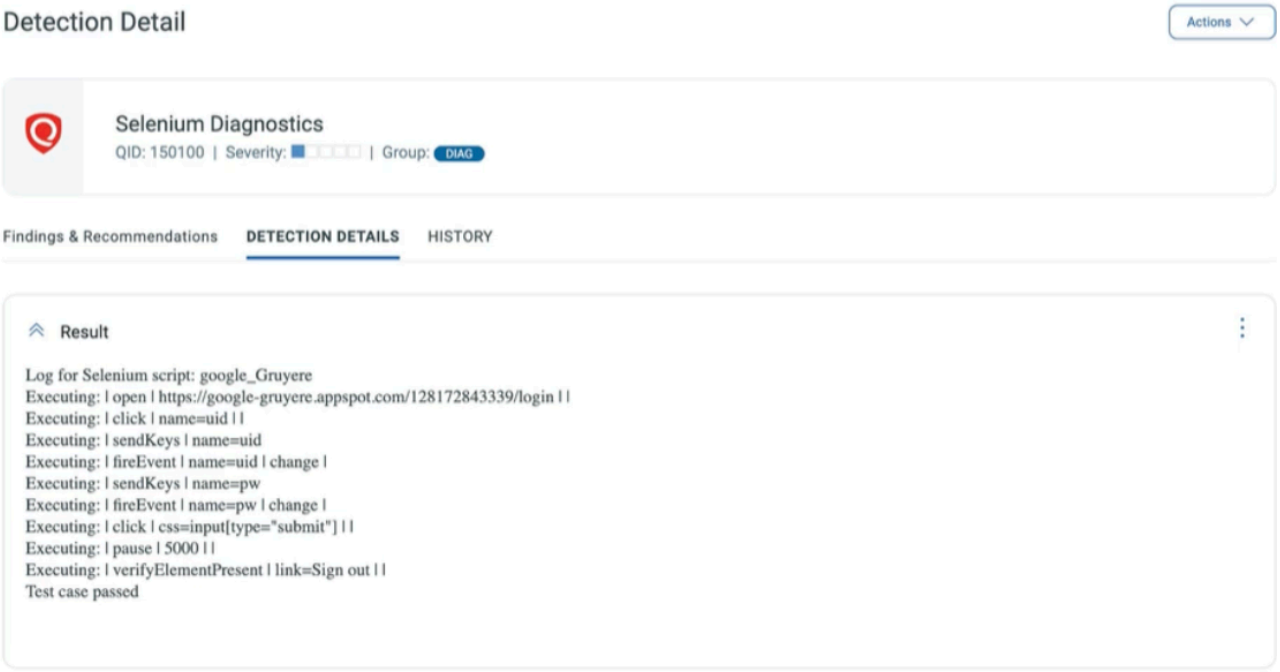
Web Application Report

Web Application Scanning の主要なレポート機能の一つである Web Application Report は、複数の Web アプリケーションにおけるセキュリティ状況を統合的に確認できるレポートです。
単一のアプリケーションに対して実施されたすべてのスキャン結果を集約し、脆弱性の推移を時系列で把握することができます。
レポートには、各脆弱性のライフサイクル（**New、Active、Re-opened、Fixed**）が履歴として表示され、現状と過去の両面から優先度を判断できます。また、レポートにはスキャン診断情報も含まれます。これらは前回の動画で紹介した「追加設定」で設定できる項目です。

特に注目すべき診断情報として **QID 150021** があり、スキャン中のスキャナーアプライアンスの動作やパフォーマンスに関する詳細な技術情報を提供します。



さらに、**QID 3150100** は Selenium スクリプトを使用する場合に有用で、レポート内で“passed”や“failed”などのキーワードを検索することで、テスト結果を素早く把握し、オートメーションフローのトラブルシューティングに役立ちます。



Web Application Report の作成デモ

ここからは、カスタム Web Application Report テンプレートを作成し、実際にレポートを生成する手順を説明します。

1. **テンプレート基本情報の入力**
   まず、テンプレート名などの基本情報を入力します。
2. **レポートから除外する情報の設定**
   既定では、レポートに作成者のユーザー名が表示されます。
   表示を避けたい場合は、「レポートから除外」をチェックします。
3. **外部統合ソースの選択**
   Qualys の検出結果だけでなく、Burp Suite や Bugcrowd での検出結果もレポートに含めることができます。これにより、複数ソースの脆弱性情報を統合できます。
4. **脆弱性ステータスフィルター**
   既定では「新規・アクティブ・再検出」の オープン状態の脆弱性 が選択されます。
   修正済み脆弱性のみを確認したい場合は、専用のテンプレートを作成することを推奨します。
   レポートを簡潔かつ実用的に保つため、既定ではオープン脆弱性に限定されています。
5. **サーチリストフィルター**
   特定のサーチリストの脆弱性のみを含めたい場合に選択します。
   適用しない場合は、すべての検出結果が含まれます。
6. **URL フィルター**
   特定の URL やアプリケーションのセクションに限定してレポートを作成できます。
   重要度の高いパスを重点的に分析したい場合に有効です。
7. **リメディエーションフィルター**
   無視済みやパッチ適用済みの検出結果を含めるかどうかを選択できます。
   これにより、意図的に除外した項目や修正済み項目も含めた包括的なレポートを作成できます。
8. **表示オプション**
   最も脆弱性の多い URL を強調表示し、リスクの高い領域を可視化できます。
9. **グルーピング／ソート条件**
   URL、深刻度、カテゴリなど、希望する基準でレポートの結果を整理できます。

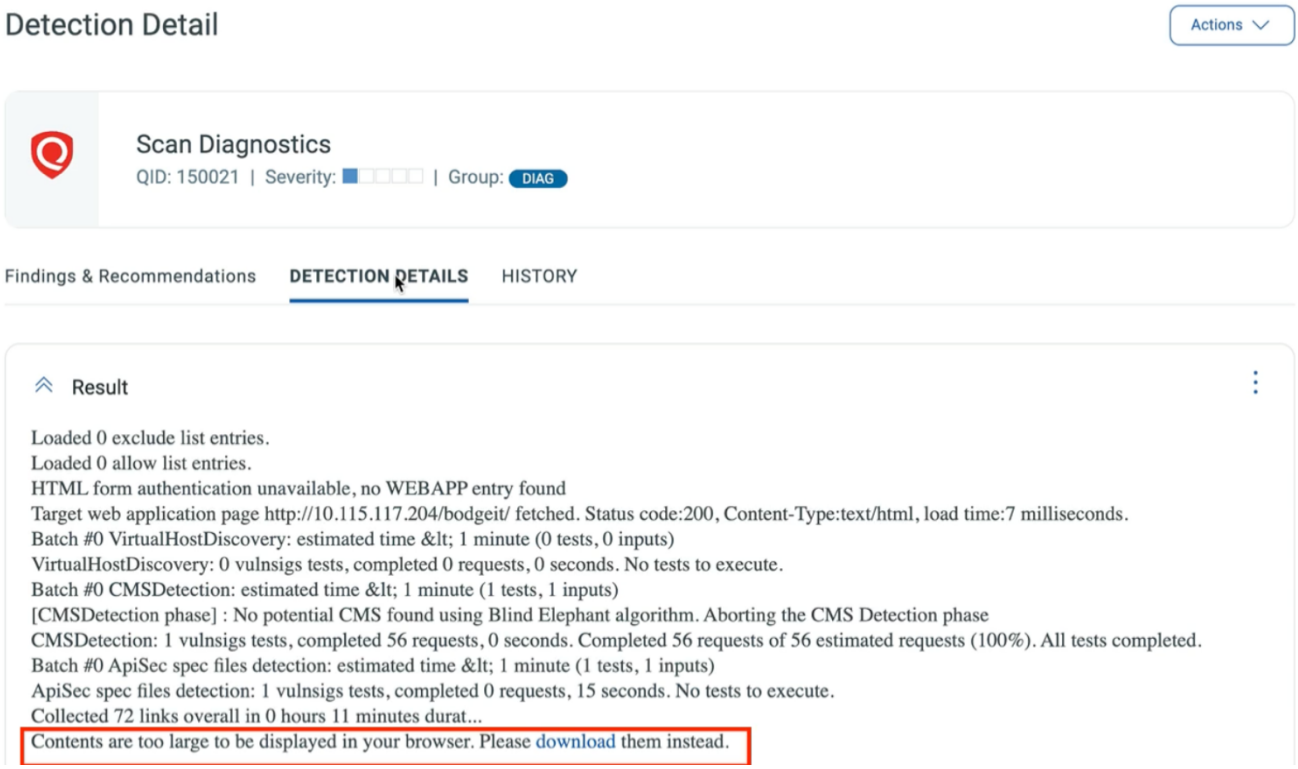
設定内容を確認後、テンプレートを保存すると、カスタムレポートを生成できるようになります。

レポートの生成

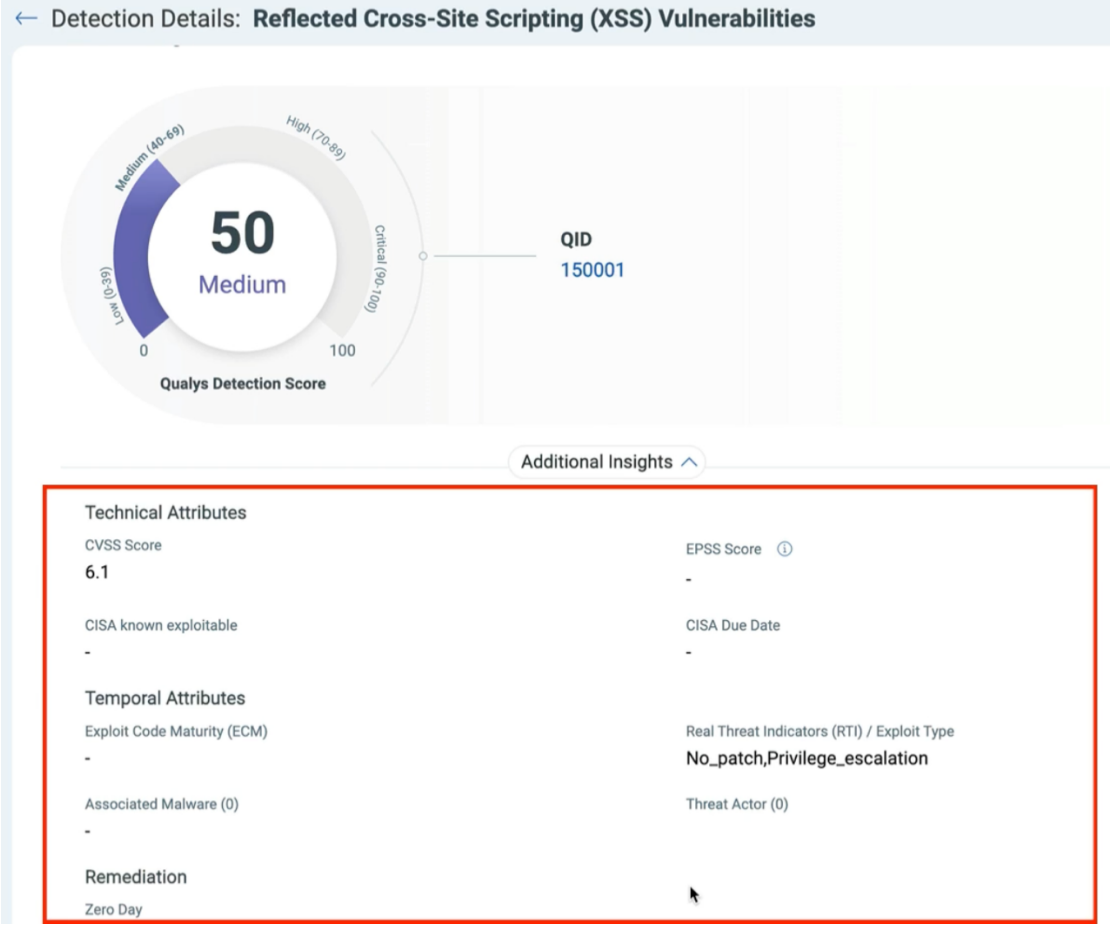
1. **Quick Actions メニューから「Render Report」を選択**
2. **レポート対象の選択**
   - タグを使って動的に Web アプリケーションを選択
   - または対象アプリケーションを手動で選択

3. **設定内容の確認**
   テンプレート名、脆弱性ステータス、サーチリスト、URL フィルターなどが表示されます。

結果セクションでは、**QID 150021** によるスキャナーの詳細な挙動情報を確認でき、トラブルシューティングやスキャン信頼性向上に役立ちます。



また、たとえば **クロスサイトスクリプティング** を検索すると、詳細な検出情報と **Qualys Detection Score** が確認できます。このスコアは複数のスレットインテリジェンスを基に算出され、リスクの優先度判断を支援します。



レポートから直接アクションを実行することも可能で、たとえば既知のリスクである場合は「無視」設定できます。
最後に、レポートをエクスポートする際は、上部のダウンロードボタンから暗号化された PDF を生成し、パスワードを設定して保護できます。
以上が、Qualys における Web Application Report の生成・確認・エクスポートの手順です。

API セキュリティ

導入

このレッスンでは、TotalAppSec が脆弱性を特定し、コンプライアンスチェックを実行することで API セキュリティを実現する仕組みを学びます。また、完全な API スキャンの実行方法、コンプライアンスチェックの検出結果の確認方法、特定された問題に対する推奨ソリューションの分析方法も学習します。

よくある課題：API セキュリティの複雑性

例として、マイクロサービスアーキテクチャを採用し、アカウント、決済、在庫管理といった主要サービスを支える 150 以上の API を運用する EC 企業を考えてみます。
これらの API は社内利用だけでなく、顧客や外部パートナーにも提供されています。そのため、セキュリティチームには API を確実に「発見」「評価」「監視」する仕組みが不可欠です。
しかし、実際には次のような課題が存在します。

- **すべての API が把握できていない**（未文書化 API、いわゆるシャドー API が存在）
- **制御の不統一**により、PII などの機密データが露出する可能性がある
- **OpenAPI 仕様の順守が統一されていない**
- **セキュリティテストが本番環境に及ばず**、リアルタイム脅威を検出できない
- 開発者が **修正手順を把握しづらく、対応が遅れがち**

TotalAppSec が提供する解決策

これらの課題に対し、Qualys TotalAppSec は次のように効果を発揮します。

1. API の完全な自動発見

未文書化・外部公開・未登録の API も含め、すべてを自動的に検出します。
これにより、セキュリティチームは シャドー API や管理外 API も含めた完全な可視化 を得られます。
EC 企業の例では、150 の API をチーム別に手作業で管理する必要がなくなりました。

2. OpenAPI/Swagger 仕様に基づく構造検証

すべての API を OpenAPI v3 準拠かどうか検証し、Swagger 定義ファイルの正確性・完全性もチェックします。
これにより、開発チーム間の表記揺れや仕様の不一致を防ぎ、API 文書化・デプロイ方法が統一されます。

3. 本番環境も対象にした継続的なセキュリティ評価

TotalAppSec は、本番サービスに影響を与えずに パッシブ／アクティブ双方のスキャン を実施します。
OWASP API Security Top 10 に示される実践的脅威（例：Broken Object Level Authorization、不適切な認可設定、機密データ露出）を対象に監査します。
たとえば、パートナー向け決済 API にアクセス制御の弱さがあれば、侵害前に検知・修正できます。

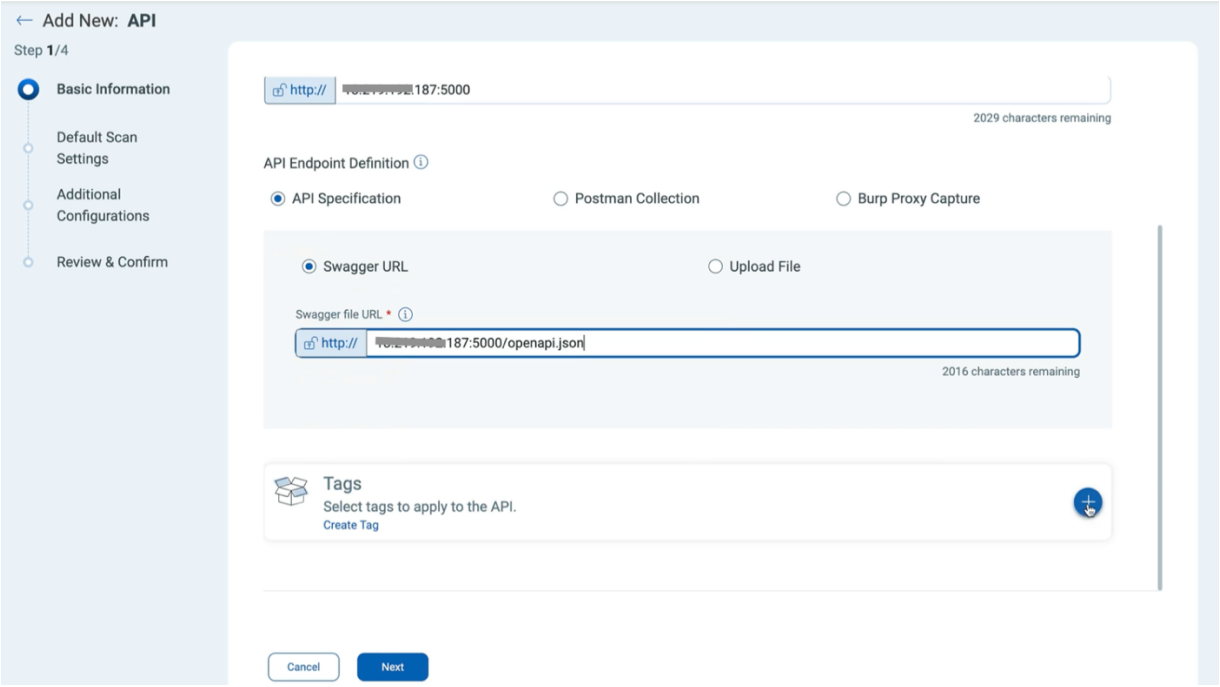
4. TrueRisk による優先順位付け

検出項目はすべて同等ではありません。TotalAppSec は ビジネスインパクト と 攻撃される可能性 の両面からリスクを評価します。これにより、開発者は重要度の高い脆弱性から効率よく対応できます。
例として、150 の API に発生した軽微な問題すべてに対応するのではなく、
「顧客データを露出しうる重要な API」 から優先して修正できます。

API の追加と評価の手順デモ

ここからは、実際に API を TotalAppSec に追加し、評価を行う手順を紹介します。

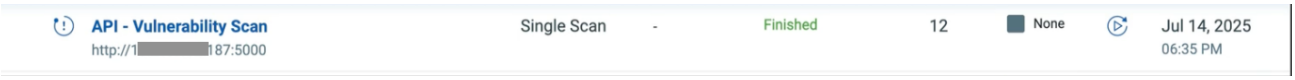
1. **ホーム画面から Applications → APIs を選択**
2. **New API をクリック**
    - 名前、URL、エンドポイント定義を入力
    - 例では Swagger ファイルの URL を使用（他の方法も選択可能）
3. **タグの割り当て**
    - 事前に作成した API タグを選択（タグ作成は別動画で説明）
4. **Default Settings を確認してそのまま Next**
5. **追加設定（認証レコード、ヘッダーインジェクションなど）を必要に応じて設定**
6. **Create API をクリックして登録完了**



脆弱性スキャンの実行

1. Quick Actions → **Vulnerability Scan**
2. スキャンジョブの名前を入力し、対象 API を選択
3. スキャン設定（Option Profile など）を選択
    - ここでは Qualys 既定値を使用
4. Launch Scan をクリックして開始

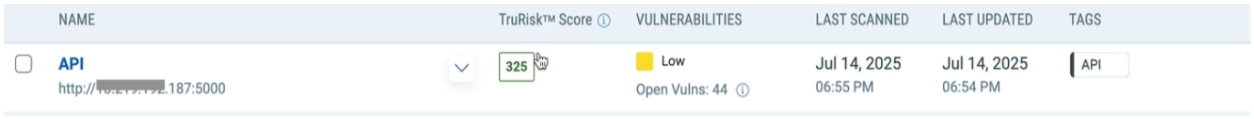
スキャン状況は Scans タブ で確認できます。同じ手順で Compliance Scan も実行可能です。



結果の確認

スキャン完了後、API セクションから結果を確認できます。

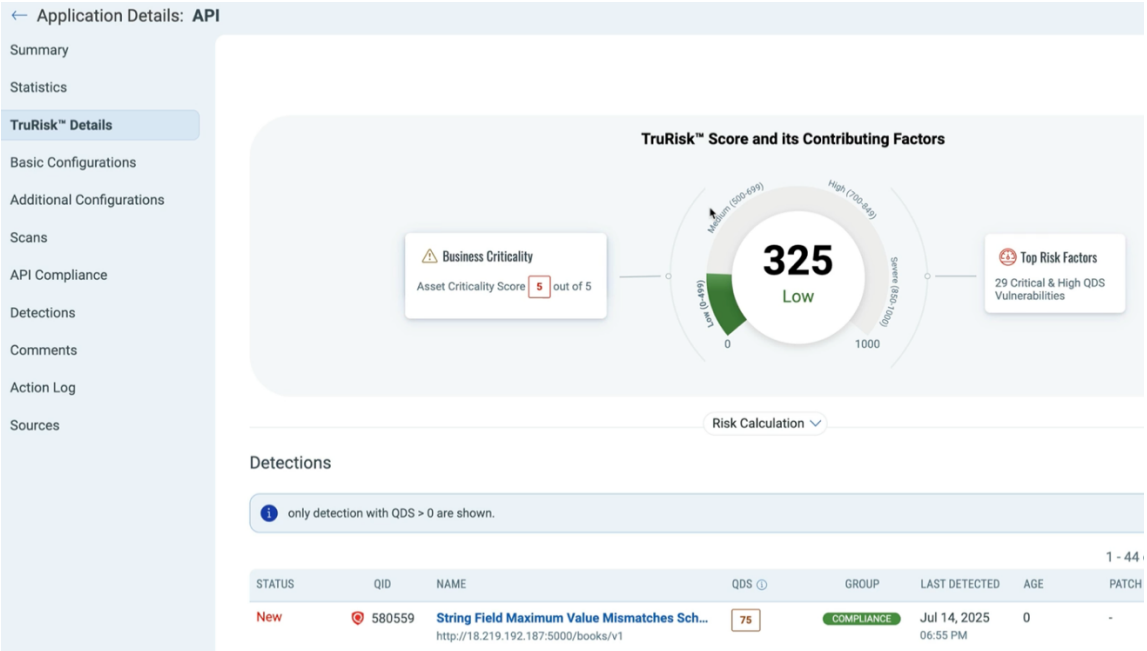
- API の **TrueRisk スコア（例：325）** を確認可能
- Endpoints ボタンから、Swagger 定義に基づく **ユニークエンドポイント一覧** を閲覧
- 各エンドポイントの検出結果へ直接アクセス可能



TrueRisk の詳細

スコアは次の要素で構成されます：

- 資産のビジネスクリティカル度（タグから継承）
- Qualys Detection Score（脅威インテリジェンスに基づく）

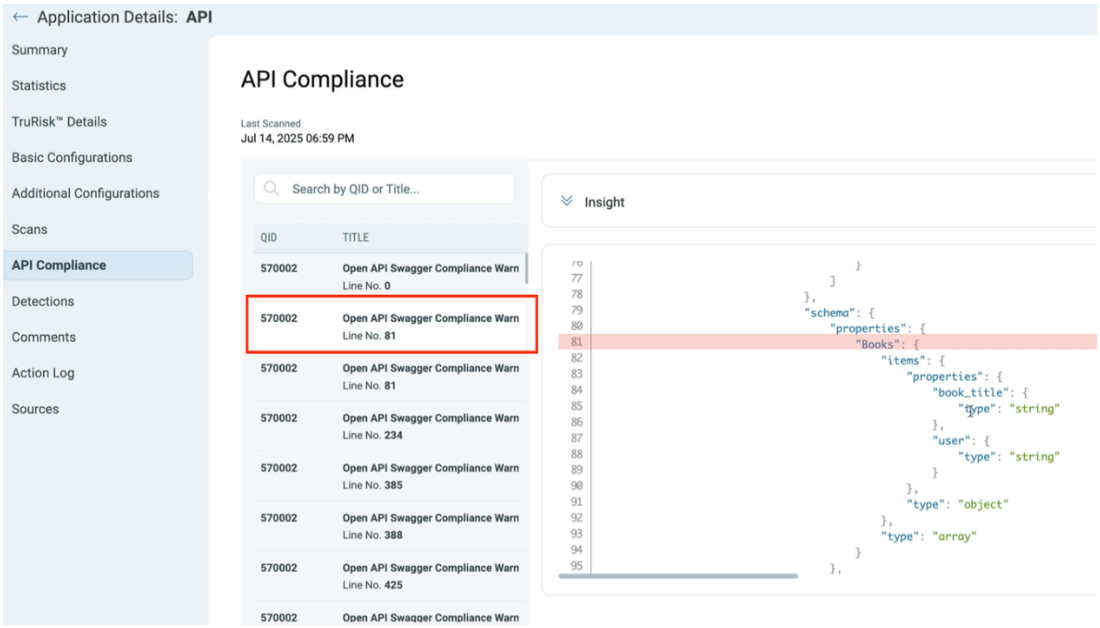


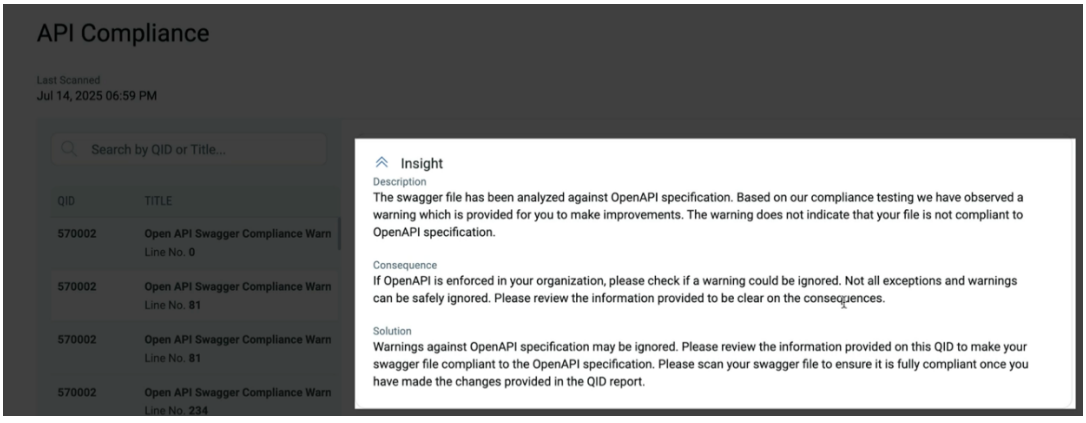
OpenAPI/Swagger の準拠状態

API セクションから、OpenAPI/Swagger の準拠状況を確認できます。

例:
OpenAPI Compliance Warning at line #81

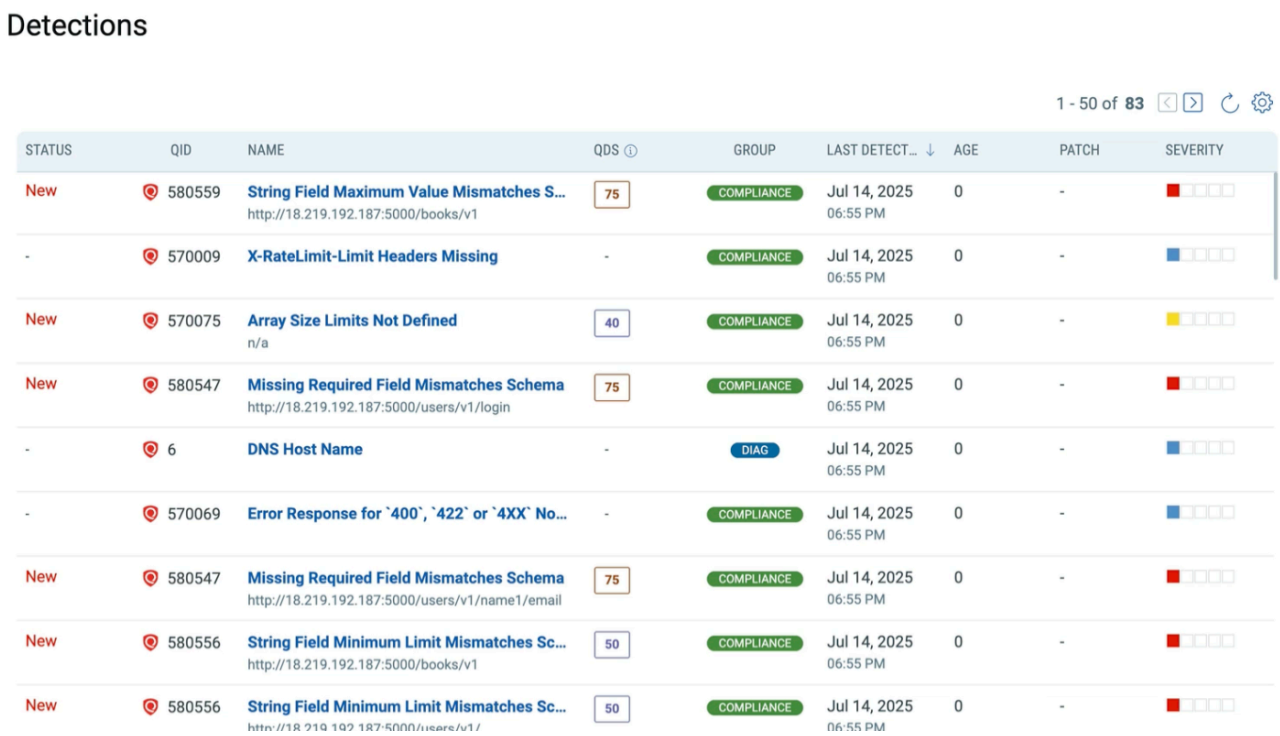
- 警告内容の詳細、影響、推奨対応が表示される
- 仕様には準拠していても、改善点が明示される場合がある
- 組織として OpenAPI 標準を厳格に運用する場合、警告も確認し改善が必要





レポートのダウンロード

検出された QID（脆弱性項目一覧）は、Web Application Report と同様の手順でレポートとしてダウンロードできます。



タグベースのユーザースコープ

タグは、アセットを効果的に整理・管理するための重要な要素であり、手動または定義済みルールを用いた自動化のいずれにも対応しています。これにより、アセットのセグメント化が最適化され、迅速な意思決定を可能にします。
タグはアセットだけのための機能ではありません。ユーザー、Web アプリケーション、API、レポートテンプレート、オプションプロファイルなど、さまざまなオブジェクトの整理にも活用できます。タグを用いることで、一貫性の確保、レポート作成の簡略化、さらにはチーム単位でのアクセス制御にも役立ちます。

それでは、実際の操作手順を見ていきましょう。

タグの作成方法

現在、TotalAppSec モジュールのホーム画面にいます。タグを作成するためには別モジュールへ移動します。

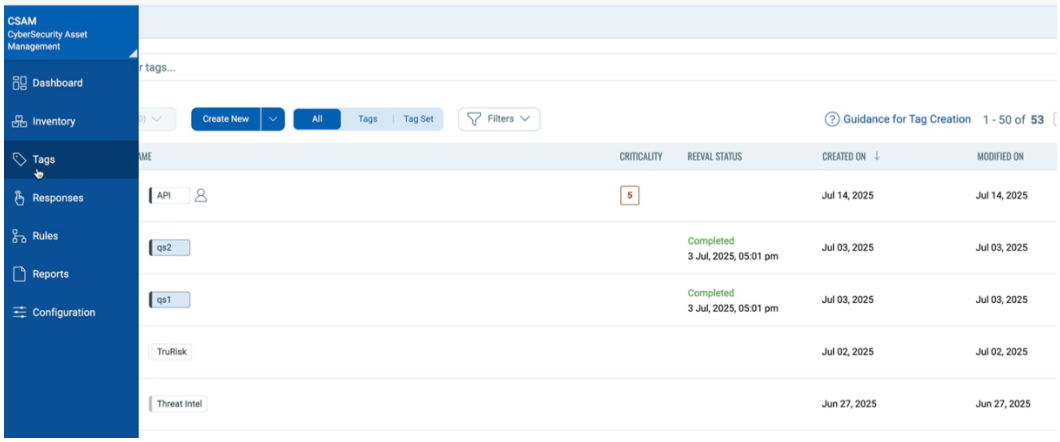
1. **モジュールピッカーを開く**

2. **Asset Management > CyberSecurity Asset Management（CSAM）** を選択
   ※CSAM へのアクセスがない場合は、Global Asset View を使用することも可能です。

CSAM モジュールに移動したら、Tags タブを開くと、サブスクリプションで作成済みのすべてのタグが表示されます。ここには、先ほど API 追加時に使用したタグも含まれています。
また、タグ階層（階層構造）を利用することで、Web アプリケーションや API をより効率的に整理できます。

新しいタグの作成

1. **Create New ボタンをクリック**
2. **タグ名を入力**
   例: total appsec ― このタグに属するアセットが TotalAppSec に関連することを示すため

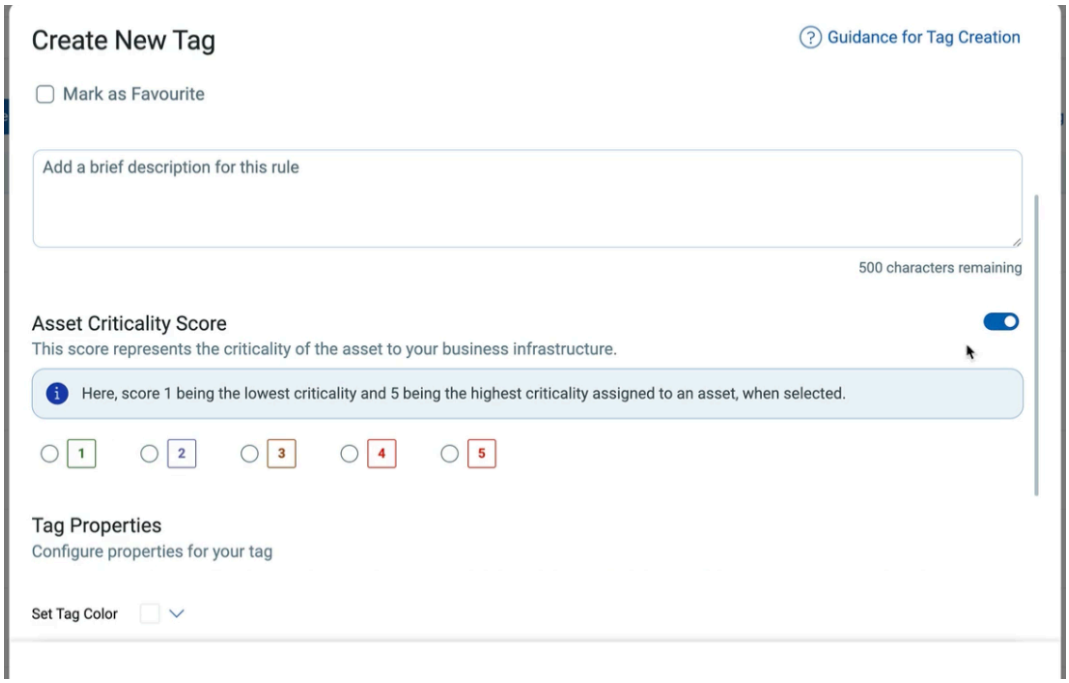


アセットクリティカリティの設定

画面をスクロールすると、「Asset Criticality Score」が表示されます。
これは、アセットのビジネス上の重要度を示します。

- スイッチをオンにすると、**1〜5** のスコアを設定可能
  - **1：最も低い重要度**
  - **5：最も高い重要度（例：本番 Web アプリケーションなど「クラウンジュエル」級）**

スコアを設定しない場合、デフォルトのクリティカリティは **2** となります。
今回の例ではスコアを設定せず進めます。



タグタイプの選択（Static / Dynamic）

タグには 2 種類があります：

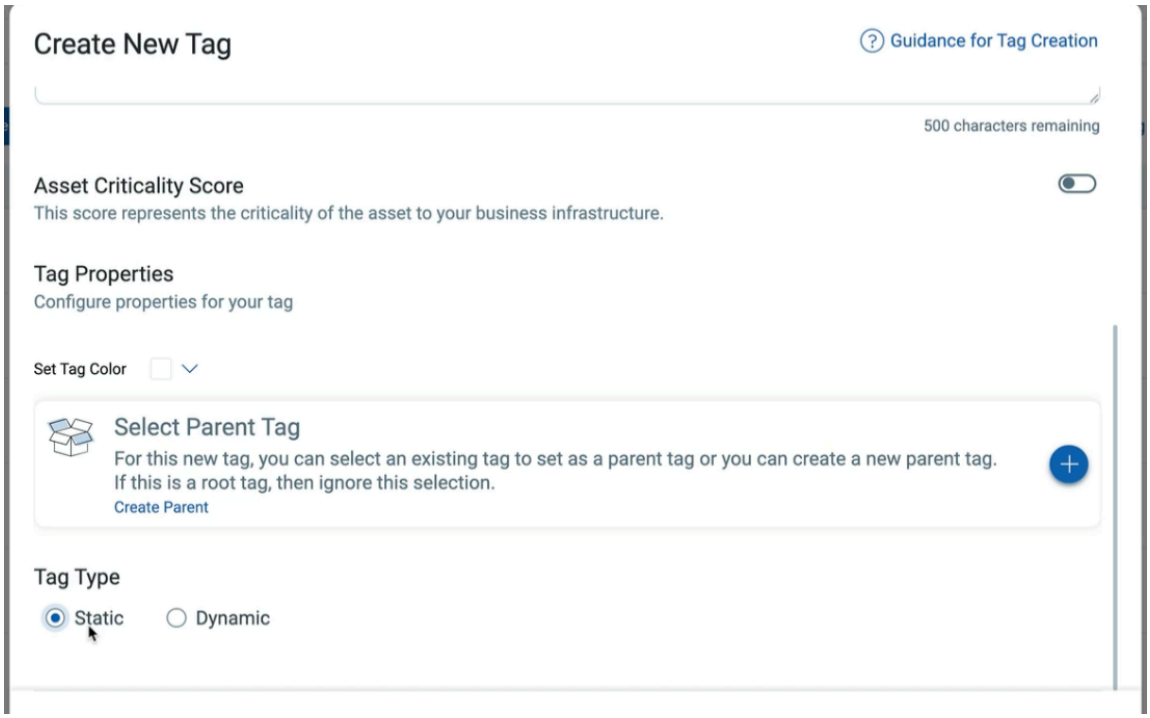
1. Static Tag（静的タグ）

- Web アプリケーションや API を手動でタグに追加する方法
- 親タグの作成時に推奨される方式

2. Dynamic Tag（動的タグ）

- ルールエンジンを使って自動的にタグ付けする方法
- 条件に基づき自動でアセットを分類できる

今回は親タグとして利用するため、Static を選択します。 設定が完了したら Create をクリックし、タグが作成されます。



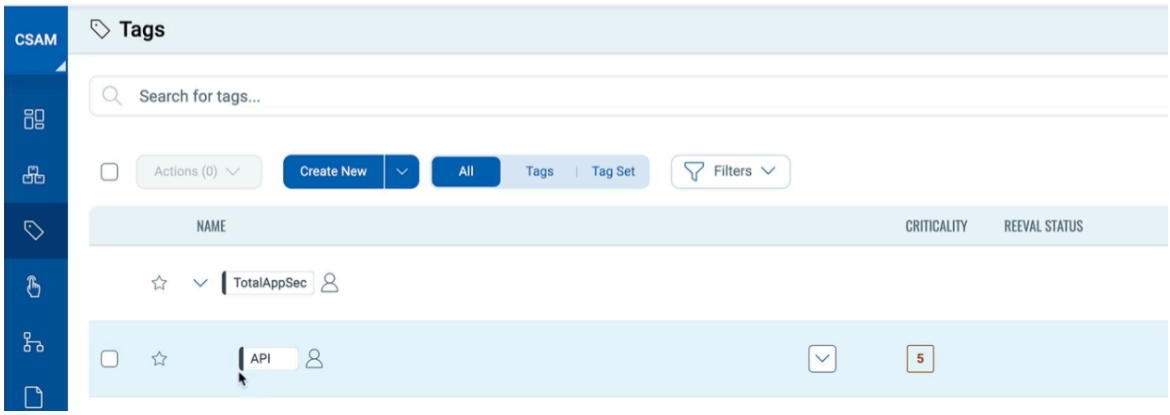
タグ階層の管理

次に、API 用タグを親タグ（total appsec）に紐づけます。

1. 該当タグの **Quick Actions > Edit** をクリック
2. **Parent Tag** を *total appsec* に設定

これで、API タグは total appsec の子タグになりました。
Drop-down から階層を確認すると：

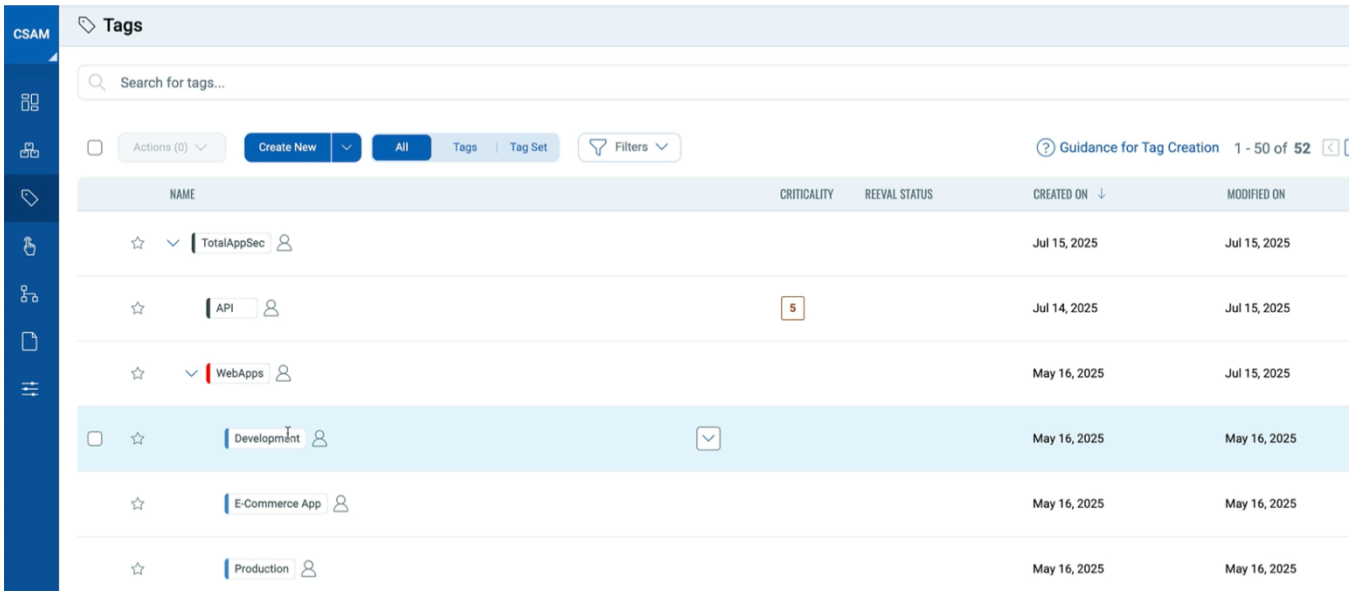
- **total appsec（親）**
  - **API（子）**



さらにスクロールすると、同様の階層構造が Web アプリケーションにも適用されている例が確認できます。
例：

- **Web Applications（親）**
  - development（子）
  - e-commerce（子）
  - production（子）

必要に応じて、それぞれにアセットクリティカリティを設定できます。

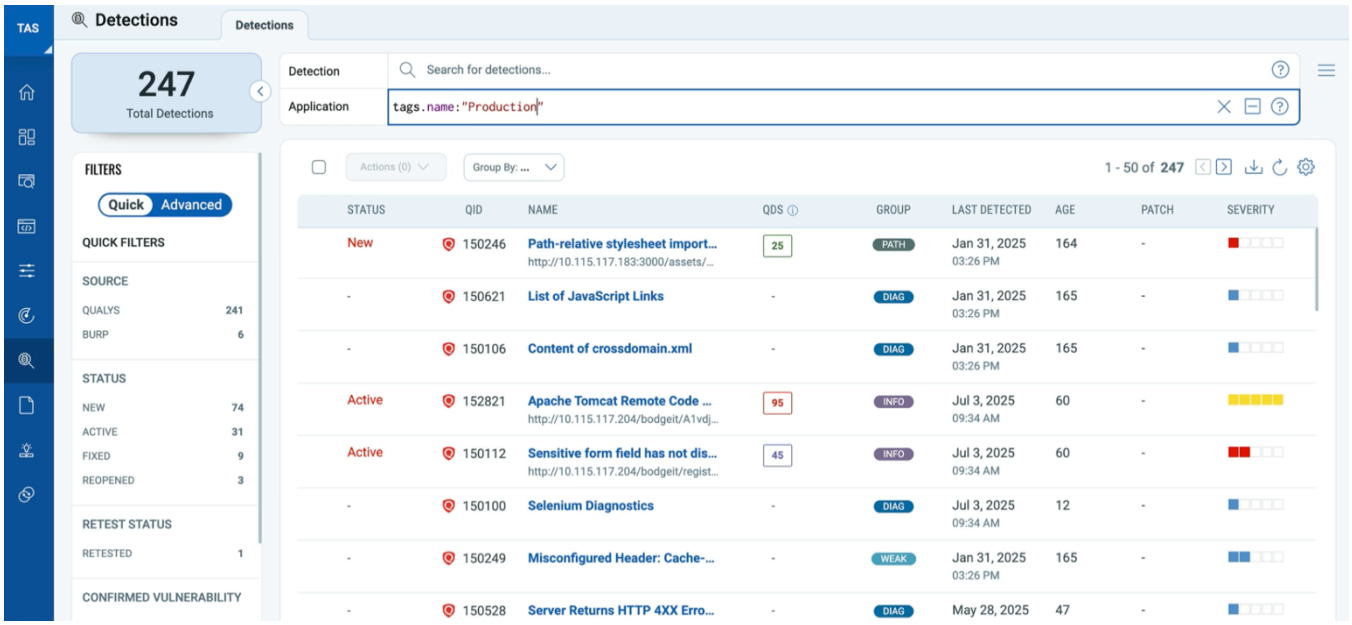


タグ活用のメリット

タグは TotalAppSec 内で大きなメリットをもたらします。

1. **TrueRisk スコアへの直接的な影響** - タグに設定したアセットクリティカリティは、API やアプリケーションの TrueRisk スコアに反映されます。
2. **大規模環境での絞り込みが容易** - アセットが大量にある場合、タグ名を条件にして検索することで結果を効率的にフィルタリングできます。
3. **脆弱性管理の効率化**

Detections タブでは、環境全体の脆弱性が一覧表示されます。
ここでもタグを使えば、例えば 本番環境の脆弱性だけを抽出 することが可能です。

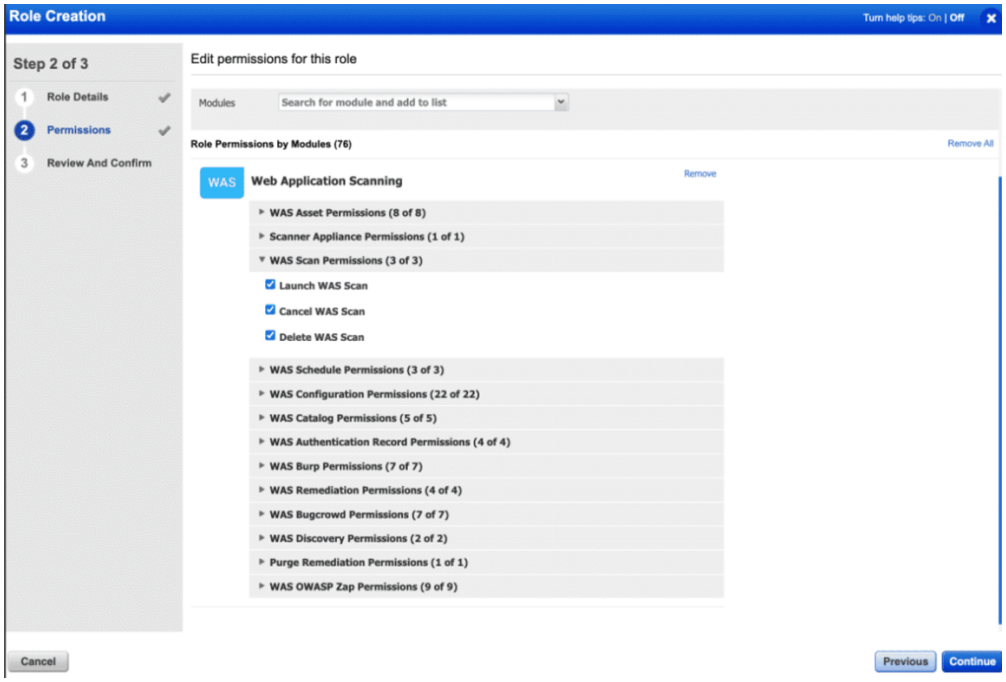


ユーザー管理

このトレーニング ビデオでは、カスタム ロールを作成し、ユーザーのスコープに対してタグを割り当てる方法を学習します。

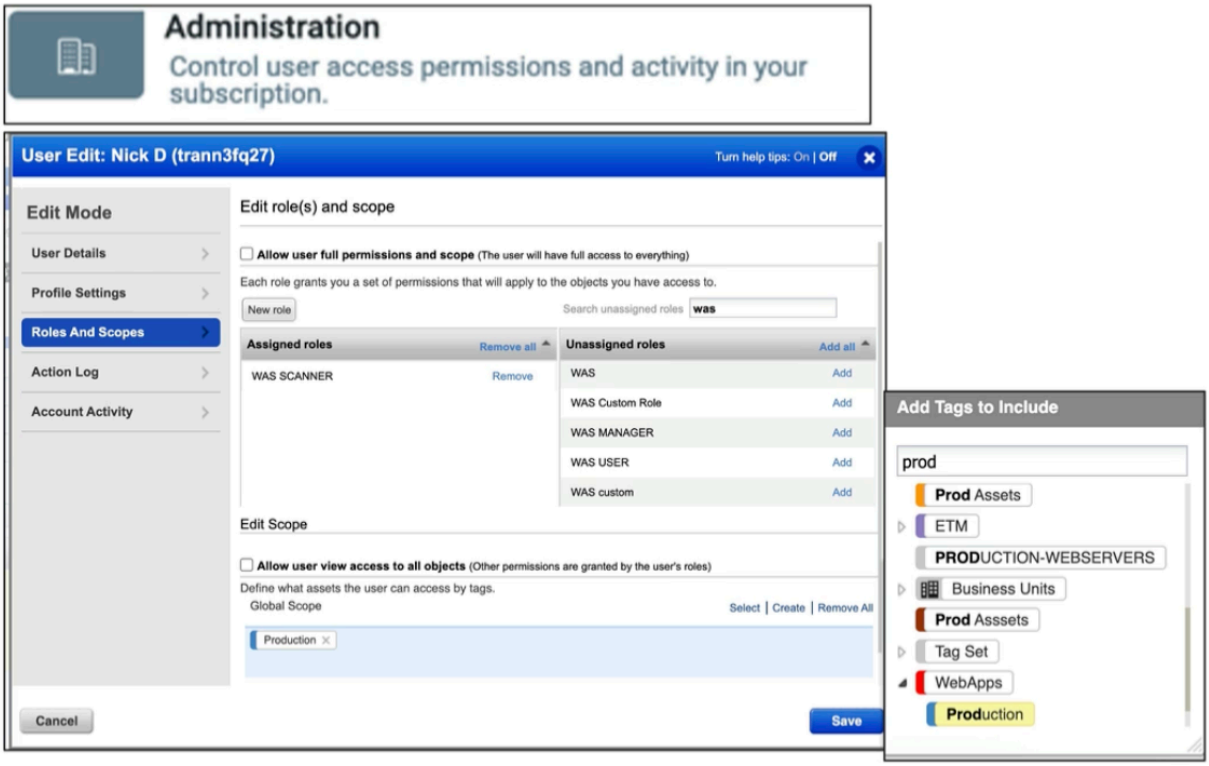
組織全体で安全かつ効率的なアクセス制御を実現するためには、適切なロールと権限を正しく割り当てる管理が重要です。まず、タグやオブジェクトが整理されているかをご確認ください。タグを活用することで、資産に対してきめ細かな権限制御が可能となり、次のステップとして効果的な管理が行えるようになります。
役割に応じたカスタムロールを作成することで、スキャニング、レポート作成、設定業務など、チームごとの責務に沿った権限設定が可能です。ユーザーロールとアセットタグを組み合わせることで、「何ができるか」と「どこにアクセスできるか」を同時に制御する強力なアクセスモデルを構築できます。
注意すべきポイントとして、ユーザーに割り当てるロールの権限は、そのユーザーがアクセス可能なタグに対して必要となる権限を満たしている必要があります。これにより、ユーザーには必要なアクセスのみが付与され、それ以上の権限は与えられません。
それでは、実際の操作デモに移り、この仕組みの活用方法をご紹介します。

ユーザーロール


Role Creation
Turn help tips: On | Off
Step 2 of 3
1 Role Details
2 Permissions
3 Review And Confirm
Edit permissions for this role
Modules
Search for module and add to list
Role Permissions by Modules (76)
Remove All
WAS
Web Application Scanning
Remove
WAS Asset Permissions (8 of 8)
Scanner Appliance Permissions (1 of 1)
WAS Scan Permissions (3 of 3)
Launch WAS Scan
Cancel WAS Scan
Delete WAS Scan
WAS Schedule Permissions (3 of 3)
WAS Configuration Permissions (22 of 22)
WAS Catalog Permissions (5 of 5)
WAS Authentication Record Permissions (4 of 4)
WAS Burp Permissions (7 of 7)
WAS Remediation Permissions (4 of 4)
WAS Bugcrowd Permissions (7 of 7)
WAS Discovery Permissions (2 of 2)
Purge Remediation Permissions (1 of 1)
WAS OWASP Zap Permissions (9 of 9)
Cancel
Previous
Continue

ロールとタグを組み合わせる

ユーザーロールと資産タグを組み合わせてアクセス権限を付与します。ユーザーロールの権限が、ユーザーのスコープ内のすべてのタグに必要な権限以上であることを確認してください。



■ ユーザーの作成

現在、Total AppSec モジュールのホーム画面を開いています。
ユーザーの作成および管理は「Administration Utility」から行います。モジュール一覧の最下部「Utilities」内に Administration セクションがあります。
ここには、現在このサブスクリプションで利用しているすべてのユーザーが表示されます。「Create User」ボタンをクリックし、新規ユーザーを作成します。一般情報を入力し、必須項目をすべて埋めたらユーザーを作成します。
作成後、名前で検索することで、ユーザーが正常に追加されたことを確認できます。

■ カスタムロールの作成

次に、このユーザー向けのカスタムロールを作成します。
今回は「Web Application Scanner」という名称のロールを作成します。
権限設定画面では、UI アクセス権を付与し、Web Application Scanning モジュールへのアクセスを許可します。さまざまな細かな権限が用意されていますが、今回は「アセット権限」のうち Web アプリケーションデータの削除やパージに関する権限を外します。

設定が完了したら「Finish」をクリックします。これでカスタムロールが作成されました。

■ ユーザーへのロール割り当て

作成したカスタムロールを先ほどのユーザーに割り当てます。
ユーザー管理に戻り、対象ユーザーの「Edit」を選択し、「Roles and Scope」タブを開きます。初期状態ではフル権限とフルスコープが割り当てられていますので、チェックを外し、作成したカスタムロールをこのユーザーへ割り当てます。
続いて、「すべてのオブジェクト表示」のチェックを外し、このユーザーには「Production」タグの資産のみアクセスを許可します。

■ 作成したユーザーでの動作確認

最後に、スキャナーユーザーとしてログインします。
Applications タブを開くと、このユーザーには Production タグに属する 1 つの Web アプリケーションのみが表示されていることを確認できます。また、Quick Actions メニューでは、レポート機能やデータパージに関する項目がグレーアウトされ、操作できない状態であることが確認できます。

インテグレーション

導入

このレッスンでは、Qualys TotalAppSec がサポートする統合について学習します。また、脆弱性管理を強化するために、Burp Suite の検出結果を Qualys Web Application Scanning ツールにインポートする方法も学習します。

Burp Suite 統合

Burp Suite と WAS の統合により、Burp Suite スキャナーと WAS スキャナーの両方からの検出結果を一元的に保存できるようになります。
この統合により、複数のユーザーと結合されたスキャン結果を共有できるようになります。現在、Burp Suite バージョン 1.7.24 のインポートをサポートしています。Qualys では、Burp で検出された問題を TAS に簡単にインポートするために、Qualys Burp 拡張機能を試すことをお勧めします。
Burp 拡張機能は、Burp Suite Professional 内の Extender タブにある BApp Store から入手できます。
Qualys Burp 拡張機能の詳細については、こちらの ブログを参照してください。（新しいタブで開きます） 記事


Introducing a Burp Extension for Integration with Qualys Web Application Scanning

Dave Ferguson, Director of Web Application Security, Qualys
March 17, 2025 - 3 min read

3 Share

Table of Contents
- Qualys Burp Extension

Bugcrowd 統合

Web アプリケーションのテストに Bugcrowd ツールキットを使用する場合は、WAS アカウント内で Bugcrowd Scanner の結果を管理できます。

- 統合されたデータは複数のユーザーと共有できます。
- 「統合」の下の「Bugcrowd」タブには、インポートされた Bugcrowd レポートが表示されます。
- 追加の Bugcrowd レポートを直接インポートすることもできます。

スキャン能力を拡張するうえで、外部ツールとの連携（Integration）は非常に重要です。
インテグレーション機能を活用することで、外部ツールで検出された脆弱性情報を一元的に取り込み、ワークフローを効率化し、統合的なセキュリティ運用を実現できます。
ここでは、Qualys が提供する主なインテグレーションオプションをご紹介します。

■ Burp Suite Findings との連携

Qualys は Burp Suite および Qualys スキャナーの結果を一元管理するための連携機能を備えています。
これにより、複数のスキャン結果を統合し、チーム全体で共有することで、脆弱性の全体像をより明確に把握できます。
現時点では Burp Suite バージョン 1.7.24 のインポートをサポートしています。
また、Qualys Burp Extension を利用することで、Burp で検出された問題をスムーズに取り込むことが可能です。
Burp Suite Professional の Extender タブから拡張機能ストアにアクセスし、拡張機能の詳細およびドキュメントリンクを確認できます。

■ Bugcrowd との連携

アプリケーションテストに Bugcrowd を利用している場合、Qualys 上で Bugcrowd スキャナーの検出結果を管理できます。
Integration セクション内の Bugcrowd タブでは、インポート済みレポートの参照、ユーザー間での共有、追加レポートの取り込みなどが可能です。

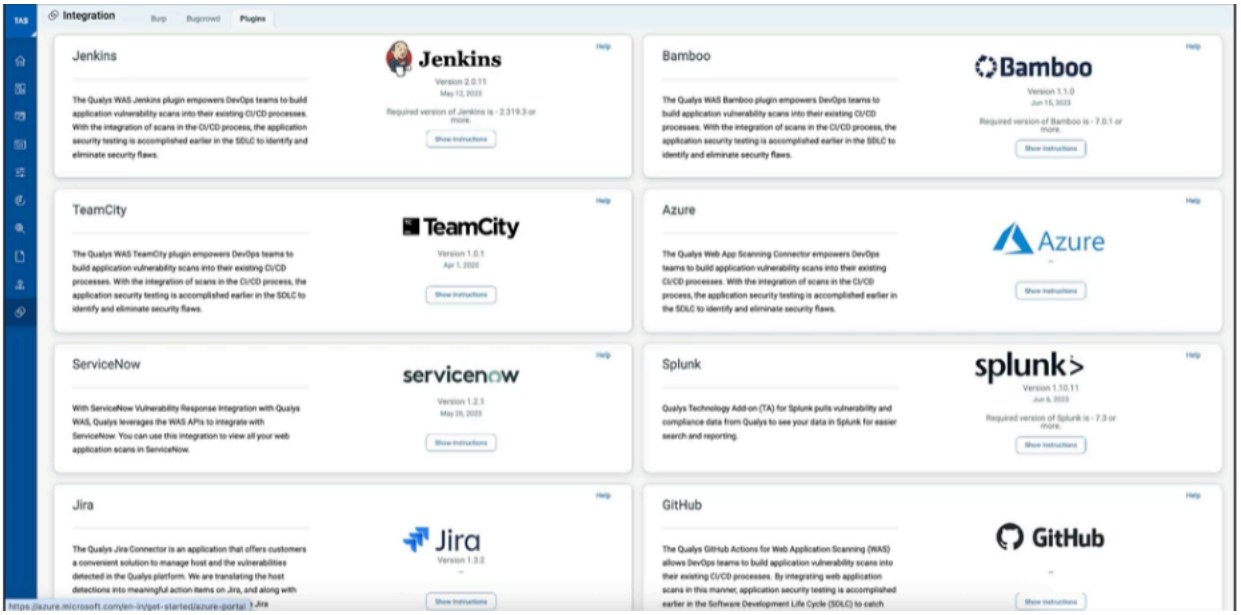
■ プラグインによる連携

Qualys はプラグインを通じてサードパーティツールとの連携にも対応しています。
Integration セクション内の Plugins タブには、現在サポートされているプラグイン一覧が表示されます。
これにより、利用しているツールやワークフローに合わせた柔軟なスキャン環境を構築できます。

■ インテグレーションの活用メリット

これらのインテグレーション機能を利用することで、

- 検出結果の一元管理
- 重複したツール運用の削減
- セキュリティチーム内のコラボレーション強化
  が実現できます。

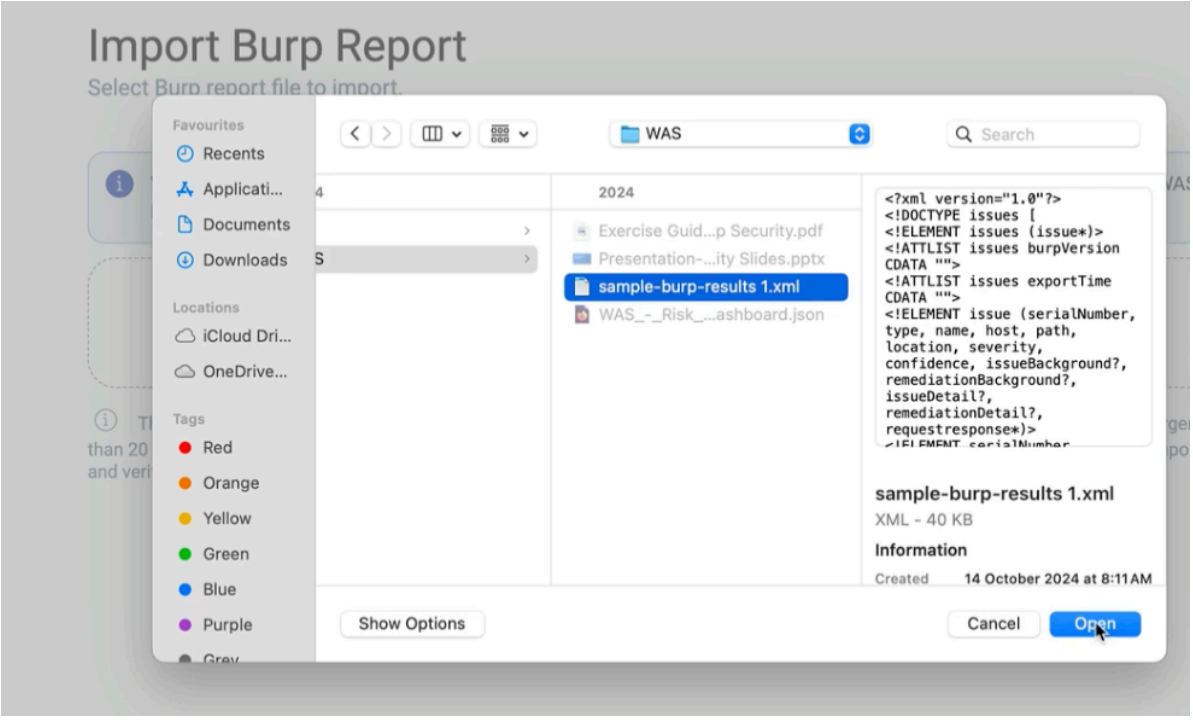


それでは、実際のデモを通じてインテグレーションの操作方法を確認していきます。

デモンストレーション

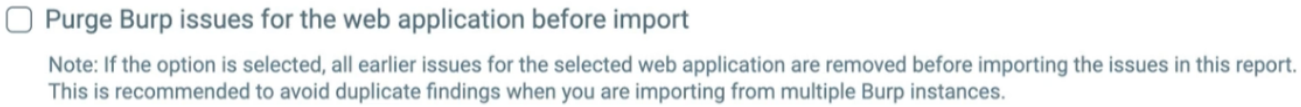
現在、Total AppSec モジュールのホーム画面を開いています。
画面下部にある Integrations セクションへ移動すると、Qualys がサポートする各種インテグレーションを確認できます。
今回は、Burp Suite の検出結果を Qualys のサブスクリプションにインポートしてみます。

1. **Import Burp Report** をクリック
2. ローカルデバイスからレポートファイルを選択



3. 該当する Web アプリケーションを関連付けたい場合は「＋」アイコンからアプリケーションを選択
4. 画面下部の **Import** をクリック

なお、「Purge Burp Issues for the web application before import」を選択すると、対象 Web アプリケーションの既存 Burp 関連の検出結果を削除できます。

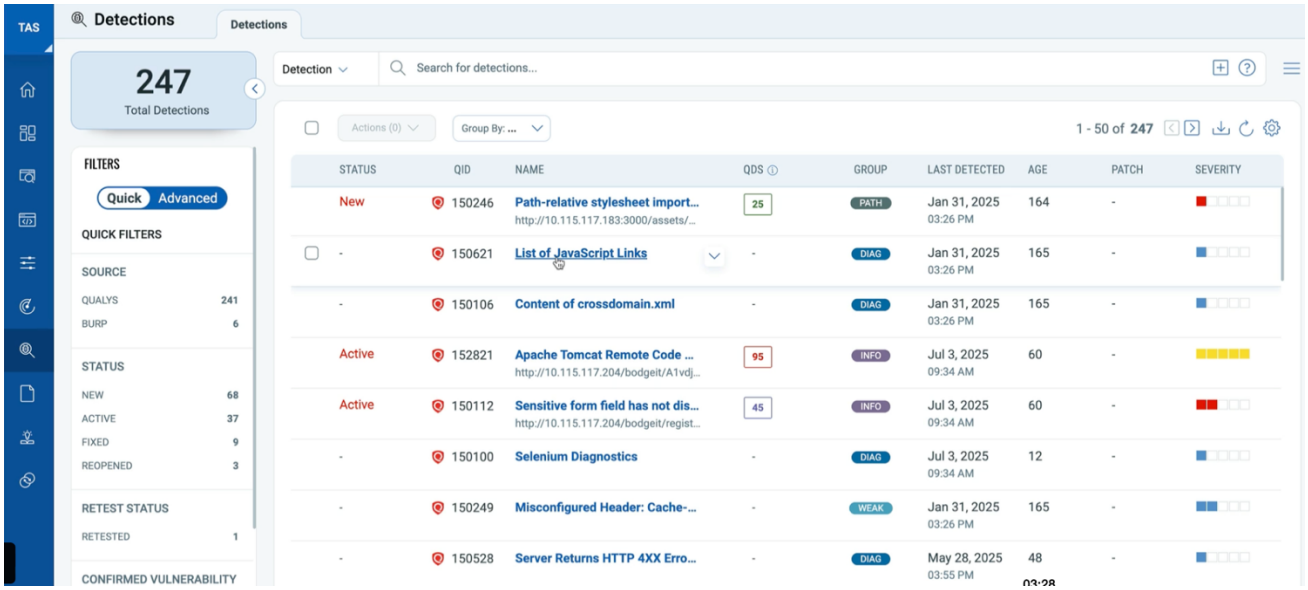


複数環境からレポートを取り込む場合、このオプションを活用することで重複登録を防止できます。
インポートが完了すると、Qualys アカウントに検出結果が追加されます。

検出結果を確認するには：

- **Detections** タブを開く
- 左側のクイックフィルターで **Source: Burp** を選択

これで、インポートした Burp レポートのすべての検出結果を確認できます。



プラグイン

TAS はプラグインを使用した Web アプリケーションのスキャンをサポートしています。「統合」の「プラグイン」タブには、WAS と統合可能なプラグインが表示されます。

参考リンク:

1. Jenkins 用 Qualys Web アプリ スキャン コネクタ（新しいタブで開きます）
2. Bamboo 向け Qualys Web アプリスキャンコネクタ（新しいタブで開きます）
3. TeamCity 向け Qualys Web アプリスキャンコネクタ（新しいタブで開きます）
4. Azure DevOps 拡張機能と Qualys Web App Scanning の統合

以上