

# Qualys Web Application Scanning

# Starting the Lab Tutorial

Navigate to the URL provided in the lab tutorial supplement to start the tutorial for a topic:



**Lab 1: WAS KnowledgeBase**  
<https://ior.ad/7Aty>



The screenshot shows the Qualys VMDR interface. At the top, there's a navigation bar with links like DASHBOARD, VULNERABILITIES, PRIORITIZATION, SCANS, REPORTS, REMEDIATION, ASSETS (which is currently selected), KNOWLEDGEBASE, and USERS. A red callout bubble labeled '1' points to the ASSETS link. Below the navigation, a large circular graphic represents the 'Asset Management' cycle, divided into segments like 'Discovering & Identifying Assets', 'Prioritizing Threats', and 'Detecting & Deploying Missing Patches'. The main content area displays a tutorial titled 'RSBP - 01 - Add Host Assets and Launch a Scan', which is described as having 59 steps and taking 9 minutes. A large red callout bubble labeled '2' points to the maximize window button in the browser toolbar. A red callout bubble labeled '3' points to a prominent 'Start' button at the bottom of the tutorial panel.

## Agenda

Qualys Web Application Scanning

### Session One:

- WAS Overview
- Basic Web Application Setup
  - Basic Info
  - Crawl Settings
  - Default Scan Settings
- Discovery Scan (Crawling)
- Advanced Web App Setup

### Session Two:

- Additional Configurations
- Web App Testing (Vulnerability Scan)
- WAS Reporting
- Tags and Users
- WAS Integrations



# Web Application Scanning Overview

# WAS Overview

## Automated Testing (Fault Injection)

- Submit “specially crafted” characters
- Observe the server’s response
- This represents 80 – 85% of Web app vulnerabilities

## Manual Testing (BURP Integration)

- Automated tools effectively detect Web application bugs (SQL execution inside user input)
- Human beings are much better at discovering program design flaws

# What Do Automated Tools Miss?

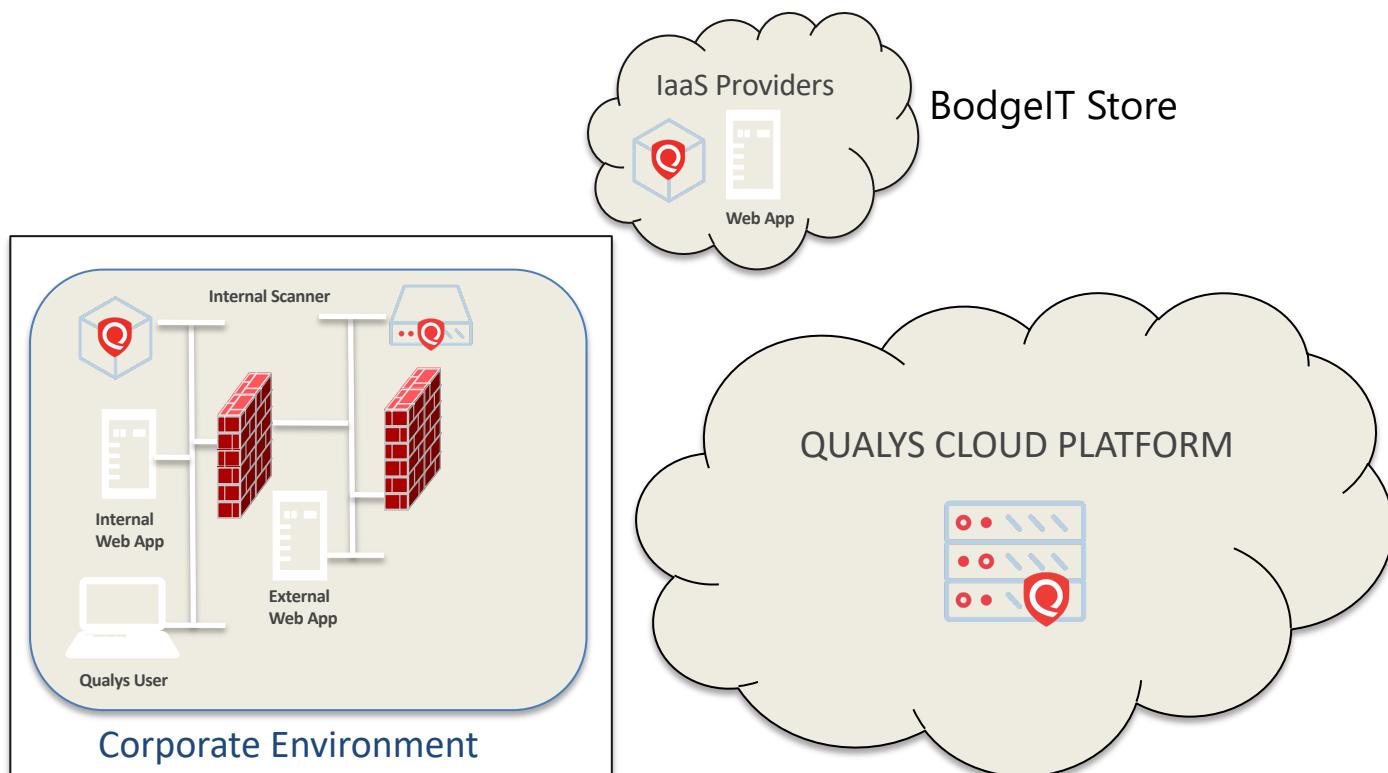
These typically require manual testing and detection:

- **Design Flaws & Logic Errors:** Manually enter URL into browser's address field to force access to an unauthorized page (i.e., point of authentication vs. point of authorization attack).



- **Permission Errors:** Public file share that has employee payroll or medical records.

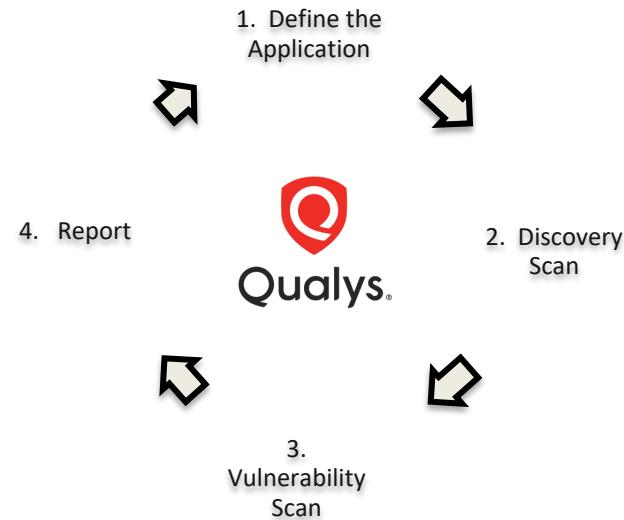
# Qualys Cloud Platform



# WAS Workflow

The workflow for analyzing a Web application involves five steps:

1. Define Web Application
2. Perform Discovery Scan (i.e., Crawl)
3. Perform Vulnerability Scan
4. Create Vulnerability Report
5. Fix Vulnerabilities





# KnowledgeBase & Search Lists

# What does Qualys WAS check?

The screenshot shows the Qualys Web Application Scanning interface. The top navigation bar includes links for Dashboard, Web Applications, Scans, Detections, Reports, Configuration, and KnowledgeBase. The KnowledgeBase tab is currently selected. On the left, there's a sidebar with 'KnowledgeBase' and 'Search Results' sections, along with 'Filter Results' and 'Identification' dropdowns. Below the sidebar is a 'Category' section with a 'Web Application' dropdown. The main content area displays a table of vulnerabilities with columns for QID, Name, Information, Category, and Severity. The table lists several vulnerabilities, all categorized under 'Web Application' and marked as 'Information' level.

	QID	Name	Information	Category	Severity
<input type="checkbox"/>	150125	File Upload Form Found	<span>Info</span>	Web Application	<span>Low</span>
<input type="checkbox"/>	150252	Telerik Web UI Cryptographic Security Bypass Vulnerability	<span>Info</span> + <span>Fix</span> <span>Details</span>	Web Application	<span>Medium</span>
<input type="checkbox"/>	150261	Subresource Integrity (SRI) Not Implemented	<span>Info</span>	Web Application	<span>Low</span>
<input type="checkbox"/>	150280	Wordpress Plugin Sitemap Stored XSS Vulnerability	<span>Info</span>	Web Application	<span>Medium</span>
<input type="checkbox"/>	150287	Ruby on Rails File Content Disclosure Vulnerability	<span>Info</span> + <span>Fix</span> <span>Details</span>	Web Application	<span>Medium</span>
<input type="checkbox"/>	150300	HTTP Request Smuggling	<span>Info</span>	Web Application	<span>Low</span>

- WAS QIDs fall under the “Web Application” category in the KnowledgeBase.
- WAS QIDs begin with 150xxx.

# Lab Tutorials

Please follow **pages 3 – 5** from the Lab Tutorial Supplement

- Lab 1 – KnowledgeBase, p. 4

3 min.



# Web App QIDs

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.

# Search List

## User-defined list of QIDs:

- **Dynamic** - Automatically updated based on list criteria.
- **Static** - Must be updated manually.

**Scan and report exclusively on the vulnerabilities in a Search List.**

## Examples:

- Run a scan just for SQLi.
- Build a report for only XSS.

Search List Creation

Turn help tips: On | Off | Lau

Enter search criteria

Criteria

Category:  Web Application (highlighted with a red arrow)

Patch Available

CVE ID

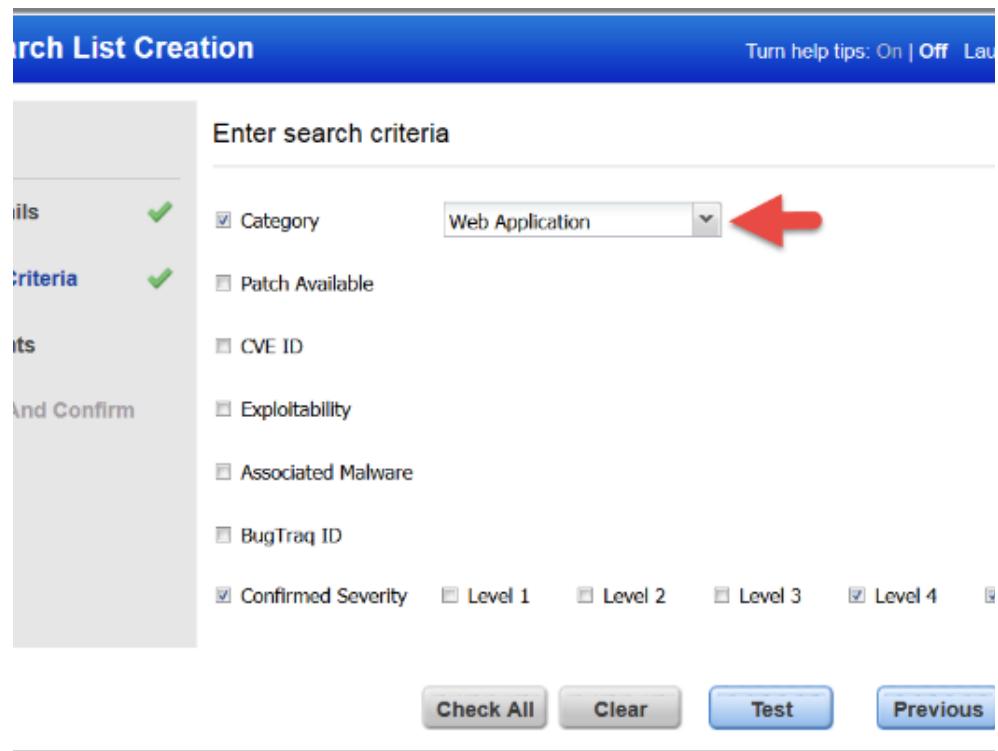
Exploitability

Associated Malware

BugTraq ID

Confirmed Severity:  Level 1  Level 2  Level 3  Level 4

Check All Clear Test Previous



# Lab Tutorials

Please follow **pages 3 – 5** from the Lab Tutorial Supplement

- Lab 2 – Search List, p. 4

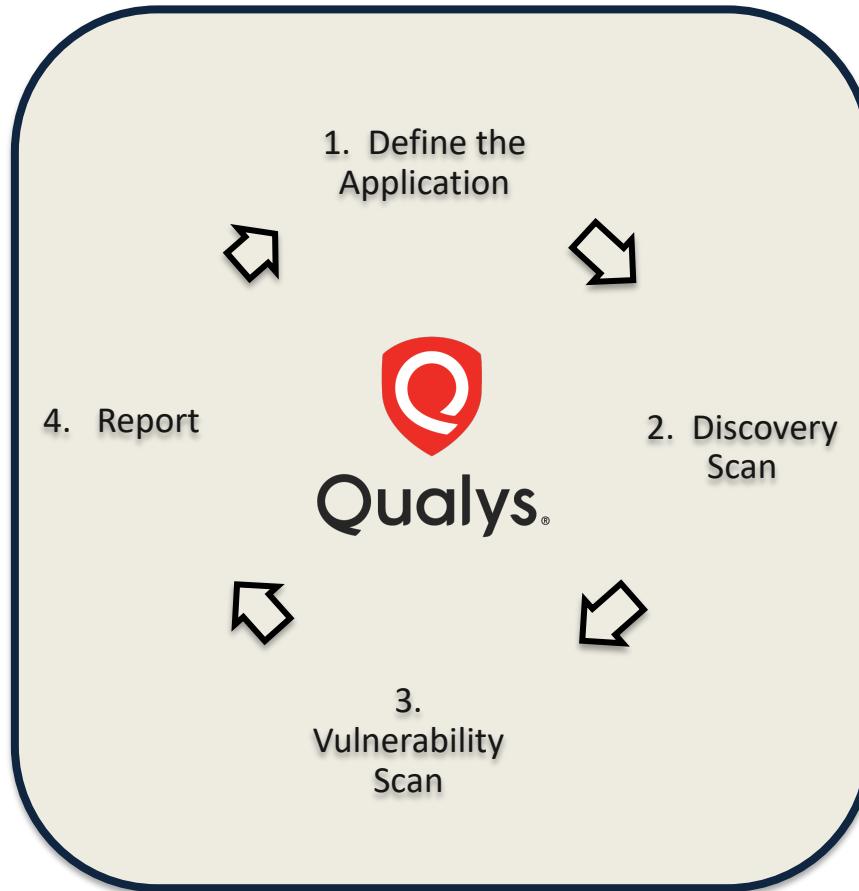
3 min.



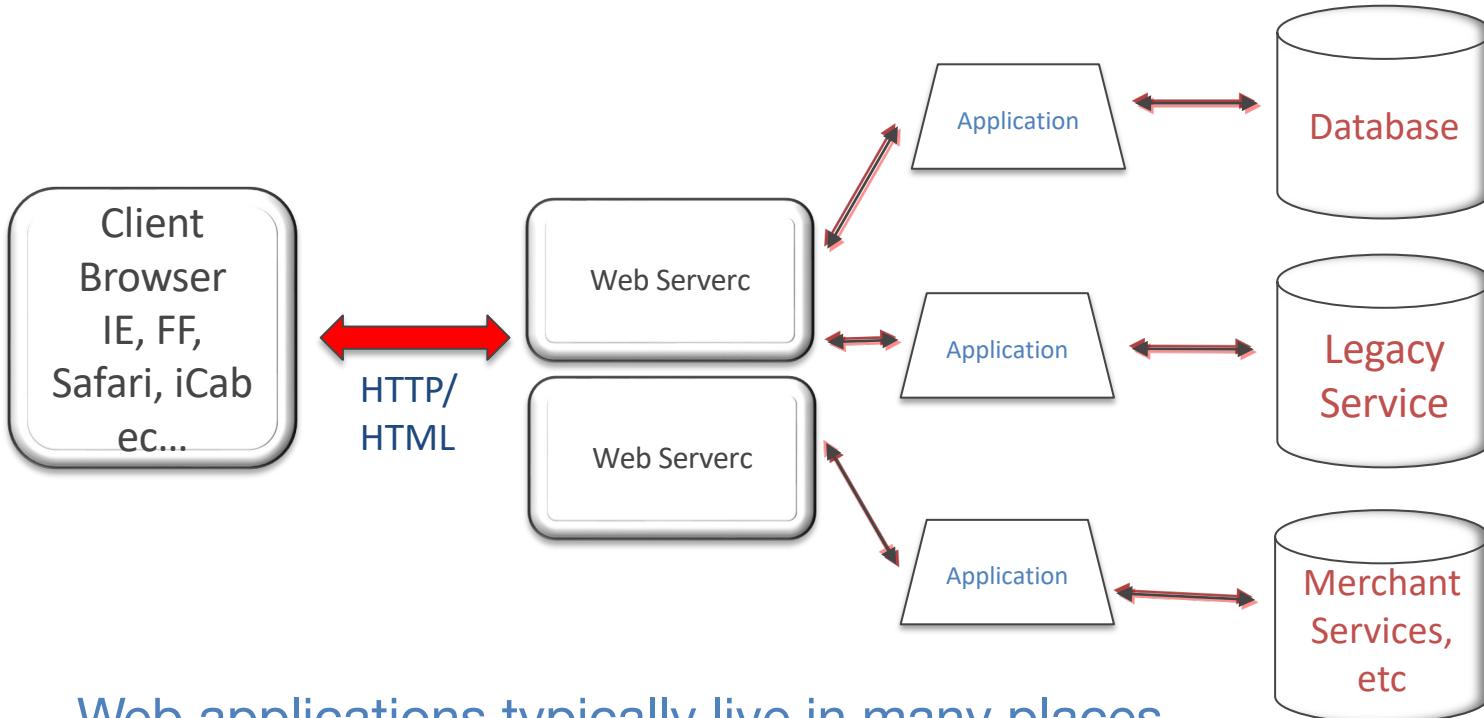


# Basic Web Application Setup

# Qualys WAS Lifecycle



# Multi-tier Web App Architecture



Web applications typically live in many places.

# Defining a Web Application

- **Uniquely identifiable entry point (URL):**  
protocol://hostname[:port]/[path/] file
  - BODGEIT STORE: <http://54.173.177.208:8080/bodgeit/>
  - BANK of QUALYS: <http://demo06.s02.sjc01.qualys.com>
- Uniquely distinguishable application code
- Supports a specific business function.
- Associated with a specific development team.
- **APPLICATION SCOPE (Crawl Scope)**
  - Covers presentation, application logic and database layers.
  - May span multiple hosts, domains, or subdomains.
  - Often designed to meet failover and scalability objectives.

# Crawl Scope Options

The screenshot shows the 'Edit: Web Application' screen in the Qualys Cloud Platform. The left sidebar lists 'STEPS 2/5' with 'Crawl Settings' selected. The main area is titled 'Crawl Settings'. It shows a 'Web Application URI(or Swagger file URL)' set to <http://demo06.s02.sjc01.qualys.com/>. Below it is a 'Crawl Scope' dropdown menu with four options numbered 1 to 4. A red arrow labeled 'Scope Options' points to this menu. A red box highlights a text input field below the dropdown.

Web Application URI(or Swagger file URL)  
<http://demo06.s02.sjc01.qualys.com/>

Crawl Scope

- 1 Limited at or below URL hostname (demo06.s02.sjc01.qualys.com)
- 2 Limit to content located at or below URL subdirectory
- 3 Limit to URL hostname and specified sub-domain
- 4 Limit to URL hostname and specified domains

Explicit URLs to Crawl/ REST paths and Parameters/ SOAP WSDL Location

Explicitly add URLs within the scope, here.

Any URLs explicitly added must fall within the selected Crawl Scope.

# Lab Tutorials

Please follow **pages 6 – 8** in the Lab Tutorial Supplement

- Lab 3 – Bodgeit Store Web App,
- Lab 4 – Standard Login Authentication, p. 6

10 min.



# Basic Info

The screenshot shows the 'Edit: Web Application' page in the Qualys Cloud Platform. The left sidebar displays 'STEPS 1/5' with five steps: 1. Basic Info (selected), 2. Crawl Settings, 3. Default Scan Settings, 4. Additional Configurations, and 5. Review & Confirm. The main content area is titled 'Basic Info'. It contains fields for 'Name \*' (The Bodgeit Store) with 233 characters remaining, 'Owner' (Phil Niegos - CORP-Demo (quays2pc34)), and 'Web Application Url (Swagger file URL) \*' (http:// 54.173.177.208:8080/bodgeit/) with 2020 characters remaining.

The Web Application URL specifies the starting point of Web app scans and contains an IP address or fully qualified domain name (FQDN).

# Crawl Settings

← Edit: Web Application

STEPS 2/5

- 1 Basic Info
- 2 Crawl Settings
- 3 Default Scan Settings
- 4 Additional Configurations
- 5 Review & Confirm

## Crawl Settings

Web Application URI(or Swagger file URL)  
http://54.173.177.208:8080/bodgeit/

Crawl Scope

Limit to content located at or below URL subdirectory

i Scope will be limited to URL subdirectory http://54.173.177.208:8080/bodgeit/, using HTTP or HTTPS and any port. All links starting with http://54.173.177.208:8080/bodgeit/ will be in scope. For example, http://54.173.177.208:8080/bodgeit/ /headlines and https://54.173.177.208:8080/bodgeit/ will be in scope.

Explicit URLs to Crawl/ REST paths and Parameters/ SOAP WSDL Location

Crawl Links

Robots txt file

Do not use robots.txt

Bodgeit Store Scope: Limit to content located at or below URL subdirectory.

http://54.173.177.208:8080/bodgeit/

# Default Scan Settings

← Edit: Web Application

STEPS 3/5

- 1 Basic Info
- 2 Crawl Settings
- 3 Default Scan Settings
- 4 Additional Configurations
- 5 Review & Confirm

## Default Scan Settings

Select Option Profile \*

Initial WAS Options

Select Scanner Appliance

External     Individual     Tags (Scanner Pool)

Lock this scanner appliance for this web application.

Duration ⓘ

Do not Cancel Scan

Crawl Settings

Progressive Scanning ⓘ

Proxy

None

NOTE: If a proxy server is selected, DNS override option will not be applicable

Many of the options specified here in the Default Scan Settings (including the Option Profile) can be adjusted or changed at scan time.



# Option Profile

# Lab Tutorial

Please follow **pages 9 – 11** from the Lab Tutorial Supplement

- Lab 5 – Option Profile, p. 9

5 min.



# Option Profile Scan Parameters

← Edit: Option Profiles

STEPS 2/5

- 1 Profile Details
- 2 Scan Parameter
- 3 Search Criteria
- 4 Comments
- 5 Review And Confirm

## Scan Parameter

Provide details for scan settings.

### General Settings

Define form action URI and form field names. This results in crawling of all forms having same fields but with different action URI.

Form Submission \*

None     Post     Get     Post & Get

Form Crawl Scope

Include form action URI in form uniqueness calculation.

Maximum Links To Crawl \*

8000

User Agent

Mozilla/5.0 (Macintosh; Intel Mac OS X 10\_11\_6) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/11.1.2 Safari/605.1.15

256 characters remaining

Request Parameter Set \*

Initial Parameters

Document Type

Ignore common binary files based on file extensions. ⓘ

## General Settings

# Option Profile Scan Parameters (cont.)

← Edit: Option Profiles

STEPS 2/5

- Profile Details
- Scan Parameter
- Search Criteria
- Comments
- Review And Confirm

**Crawling Options**

Enhanced Crawling ⓘ

Enable SmartScan ⓘ

SmartScan Depth \*

5

**Behavior Settings**

Define the threshold for the timeout and unexpected consecutive errors that would be allowed during a scan. Once the threshold is exceeded, the scan is terminated.

Timeout Error Threshold

100

Unexpected Error Threshold

500

**Performance Settings**

Provide performance settings to define the intensity of web application scans.

Pre-Defined    Custom

Scan Intensity

Medium (# of HTTP Threads: 5)

**Bruteforcing Settings**

Select password bruteforcing option to check vulnerabilities related to password-cracking techniques

Use password bruteforcing

## Crawling, Behavior, Performance, and Bruteforcing

# Option Profile Search Criteria

The screenshot shows the 'Add New: Option Profiles' interface, specifically the 'Search Criteria' step (Step 3 of 5). The left sidebar lists steps 1 through 5: Profile Details, Scan Parameter, Search Criteria, Comments, and Review And Confirm. The main area is titled 'Search Criteria' and contains instructions: 'Provide criteria for search during the web application scan.' Below this is the 'Detection Scope' section, which includes a dropdown menu set to 'Core'. A checkbox for 'Include additional XSS payloads (may significantly increase scan time)' is present. A tooltip for 'Core QIDs' provides information: 'View list of Core QIDs' and 'Note: All Information Gathered QIDs are included in scan detection scope when you set the scope to Core.' Further down are sections for 'Sensitive Content' (checkboxes for Credit Card Numbers, Social Security Numbers (US), and Custom Contents) and 'Keyword URL Search' (checkbox for Keyword Search).

← Add New: Option Profiles

STEPS 3/5

- 1 Profile Details
- 2 Scan Parameter
- 3 Search Criteria
- 4 Comments
- 5 Review And Confirm

## Search Criteria

Provide criteria for search during the web application scan.

### Detection Scope

Select the scope of detections for the web application scan with this profile. Specify if the scan should perform a full assessment for all WAS detections, or if the scan shall focus on the specific WAS detections/vulnerabilities.

**Detection \***

Core

Include additional XSS payloads (may significantly increase scan time).

**View list of Core QIDs**  
Note: All Information Gathered QIDs are included in scan detection scope when you set the scope to Core.

### Sensitive Content

Credit Card Numbers

Social Security Numbers (US)

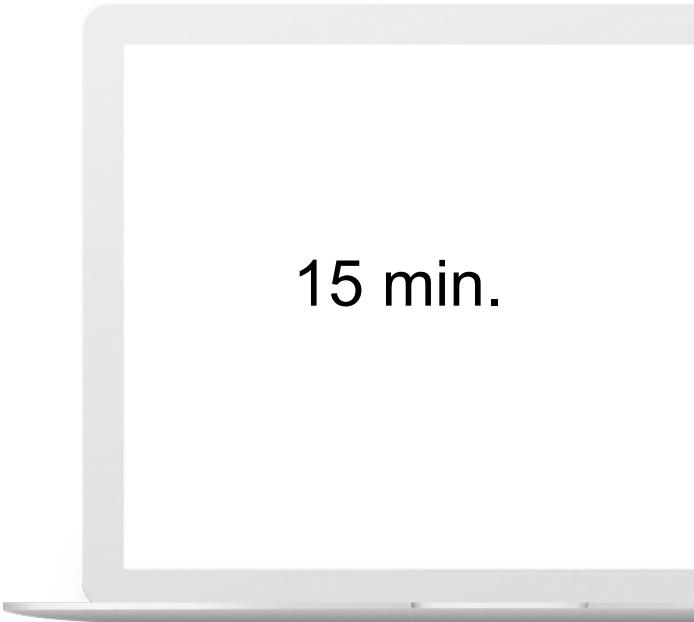
Custom Contents

### Keyword URL Search

Keyword Search

Detection Scope, Sensitive Content, and Keyword URL Search

# Session Break

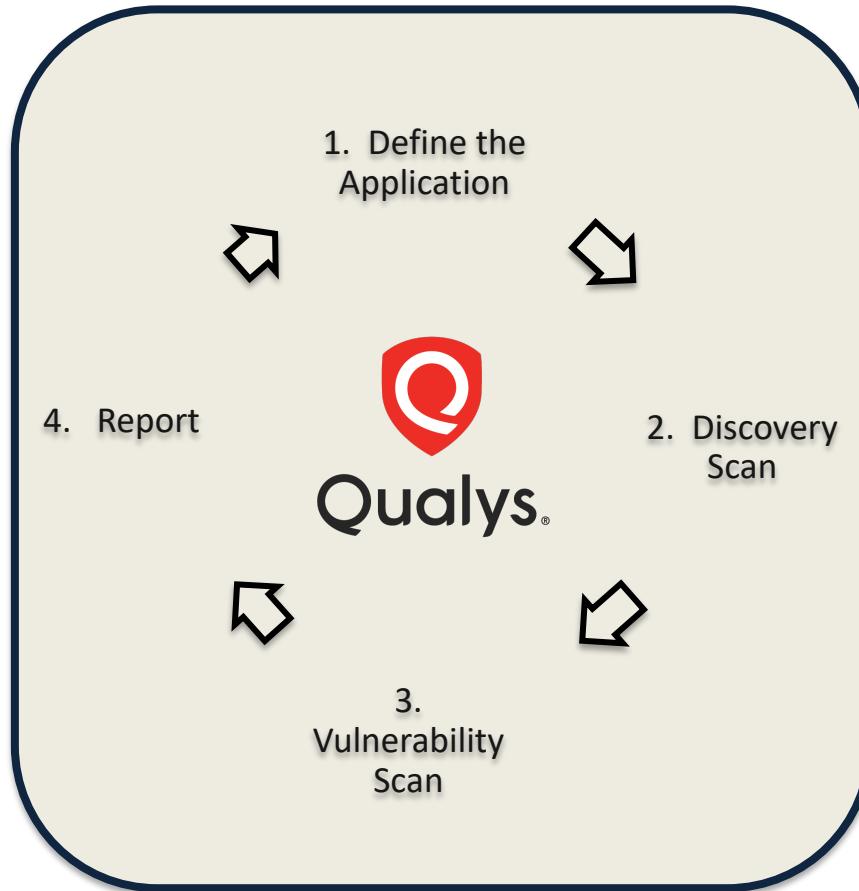


15 min.



# Discovery Scan (Crawling)

# Qualys WAS Lifecycle



# Discovery

1. Scan begins at starting URL identified in the application definition.
2. Using the Scope Options identified in the Web app's Crawl Settings, the scan follows links to discover pages and content.
3. Configuration data is collected from the target app and its host.
4. Vulnerability testing is not performed.

# The Crawl

## HTML-based links

```
<li class="heading underline"><a href="/company/newsroom/news-releases/">Newsroom</a></li>
<li><a href="/company/newsroom/news-releases/">News Releases</a></li>
<li><a href="https://community.qualys.com/blogs" target="_blank">Qualys Blogs</a></li>
<li><a href="/company/newsroom/media-coverage/">Media Coverage</a></li>
<li><a href="/research/security-alerts/">Security Alerts</a></li>
...
```

## Links via JavaScript

```
<script src="/asset/vendor/modernizr/border-image.js"></script>
<script src="//ajax.aspnetcdn.com/ajax/jquery/jquery-1.7.2.min.js"></script>
<script src="//dldejaj6dcqv24.cloudfront.net/vendors/jquery.tools-1.2.7.min.gz.js"></script>
<script src="//dldejaj6dcqv24.cloudfront.net/vendors/jquery.easing-1.3.min.gz.js"></script>
<script src="/asset/script/main.js"></script>
<script type="text/javascript" src="/asset/script/homepage.js"></script>
```

# QID 150009 - Links Crawled

**Information Gathered Details**

150009 Links Crawled

Finding #	272985* (125111062)	Web Application	BodgeIT Store
Group	Information Gathered	Authentication	Not Used
CWE	-	Detection Date	21 May 2015 9:39AM GMT-0500
OWASP	-		
WASC	-		

Details Show

Results

Duration of crawl phase (seconds): 162.00  
Number of links: 31  
(THIS NUMBER EXCLUDES FORM REQUESTS AND LINKS RE-REQUESTED DURING AUTHENTICATION.)

Export results to an unformatted text file.

Export...

http://ec2-54-84-232-118.compute-1.amazonaws.com:8080/bodgeit/  
http://ec2-54-84-232-118.compute-1.amazonaws.com:8080/bodgeit/about.jsp  
http://ec2-54-84-232-118.compute-1.amazonaws.com:8080/bodgeit/admin.jsp  
http://ec2-54-84-232-118.compute-1.amazonaws.com:8080/bodgeit/advanced.jsp  
http://ec2-54-84-232-118.compute-1.amazonaws.com:8080/bodgeit/basket.jsp  
http://ec2-54-84-232-118.compute-1.amazonaws.com:8080/bodgeit/contact.jsp  
http://ec2-54-84-232-118.compute-1.amazonaws.com:8080/bodgeit/home.jsp  
http://ec2-54-84-232-118.compute-1.amazonaws.com:8080/bodgeit/login.jsp  
http://ec2-54-84-232-118.compute-1.amazonaws.com:8080/bodgeit/product.jsp?nrnid=10  
http://ec2-54-84-232-118.compute-1.amazonaws.com:8080/bodgeit/pr...  
http://ec2-54-84-232-118.compu...

Download results to a formatted ASCII text file.

Contents are too large to be displayed in your browser. Please [download them instead.](#)

# Explicit URLs to Crawl

- Specify URLs you want the service to crawl
- Useful for pages not linked to other pages in the application

The screenshot shows the 'Edit Mode' configuration page for a web application scan. The left sidebar lists various settings: Asset Details, Application Details (selected), Scan Settings, Crawl Settings, Redundant Links, Authentication, Crawl Exclusion Lists, Advanced Options, Malware Monitoring, Comments, and Action Log. The main area is titled 'Tell us about the web application you want to scan'. It includes a 'Target Definition' section with a 'Web Application URL' field containing 'https://demo06.s02.sjc01.qualys.com:443/'. Below it is a 'Crawl Scope\*' section with a dropdown menu set to 'Limit at or below URL hostname (demo06.s02.sjc01.qualys.co)'. A detailed description explains the scope limitation. The 'Explicit URLs to Crawl / REST Paths and Parameters / SOAP WSDL Location' section contains two URLs: 'https://demo06.s02.sjc01.qualys.com/aade3aEfjafae.htm' and 'https://demo06.s02.sjc01.qualys.com/webservices/wsdl'. A red callout box highlights this section with the text: 'Non-linked URLs or Web Services (must be consistent with selected scope)'. At the bottom, there's a 'Burp Log File' section with a note about uploading log files and a 'Upload Burp Log File' button.

# QID 150115 - Authentication Form Found

Authentication form found at: http://demo06.s02.sjc01.qualys.com/  
Action uri: http://demo06.s02.sjc01.qualys.com/  
Fields: **login, password, action, submit**

## Standard HTTP Login Form



The screenshot shows a standard HTTP login form with the following fields:

- Username: (?)
- Password: (?)
- Remember Me
- Login
- Cancel

Below the form, the HTML source code is displayed, highlighting the login form structure:

```
<div id="jive-userbar">
  <div id="jive-userbar-login">
    <form action="$http://community.qualys.com/cs_login" method="post" name="loginform" class="sammy-app-1336162184090-12->">
      <span class="jive-userbar-login-welcome" id="jiveLoginWelcome" style="display: none; ">...</span>
      <span class="jive-userbar-login-form" id="jiveLoginForm" style=">
        <span class="jive-userbar-login-username">
          <label for="login-username">
            Username: </label>
            <a href="forgot-username!input.jspa" title="I forgot my username ">(?)</a>
            <input type="text" name="username" size="20" maxlength="150" value tabindex="1" id="login-username">
          </span>
        <span class="jive-userbar-login-password">
          <label for="login-password">
            Password: </label>
            <a href="emailPasswordToken!input.jspa" title="I forgot my password ">(?)</a>
            <input type="password" name="password" size="20" maxlength="150" value tabindex="2" id="login-password">
          </span>
```

# When does a Discovery Scan End?

Crawl stops when:

- Max number of links threshold is met

Maximum crawl requests (the total number of links and forms to follow)\*

8000

- No new links are discovered
- Scan time-out is reached

Cancel scan after



after

2



hours

\*Default max. scan time = 24 hrs.

# “No Web Service”

Scan will return “No Web Service” status if the scanner:

- Cannot get a DNS lookup on the site
- Cannot reach the target because of routing
- Cannot get a web service to respond to a GET request

# Lab Tutorials

Please follow **pages 12 - 14** in the Lab Tutorial Supplement

- Lab 6 – Web App Scanning, p. 12

5 min.



# Web Application Sitemap

- View pages crawled
- Create new Web apps
- Add URLs to Black List
- Add URLs to White List

The screenshot shows the Qualys Web Application Sitemap interface for a Catalog Web Application. The main window displays a list of URLs under the 'Link' tab. A context menu is open over the URL 'boq', with the option 'Create Web Application' highlighted. A red box and a red callout bubble with the text 'Download web app links' point to the 'Create Web Application' button. The interface includes filters for Crawled (28), Rejected (1), External (2), Vulnerabilities (44), and Sensitive Contents (0). On the right, there are sections for Folder Information, Children Information, and Assessment Details, which includes a pie chart of total vulnerabilities by level.

Web Application Sitemap: Catalog Web Application: demo06.s02.sjc01.qualys.com, Port 443

Use the filters below to alter list view for this application sitemap.

Page view filters: C Crawled 28, R Rejected 1, E External 2, Vulnerabilities 44, Sensitive Contents 0

Link in view: demo06.s02.sjc01.qualys.com:443

Actions (1) Export Sitemap 1 - 8 of 8

Link Info. Children Info.

Link

- ..
- admin
- boq** (selected)
- icons
- includes
- phpMyAdmin
- ?account=business
- ?account=personal

Quick Actions:  
Create Web Application  
Add To Black List  
Add To White List

Folder Information:  
Folder: https://demo06.s02.sjc01.qualys.com:443/boq/  
Status: Crawled  
Vulnerabilities: 1  
Sensitive Content: 0

Children Information:  
Pages Crawled 24, Vulnerabilities 26

Assessment Details:  
Total Vulnerabilities 26  
2 Level 5, 0 Level 4, 9 Level 3, 6 Level 2, 9 Level 1

Crawling Details

# Lab Tutorials

Please follow **page 15** in the Lab Tutorial Supplement

- Lab 7 – Sitemap, p. 15

3 min.



# Basic Web App Setup Review

## 1. Basic Info

- Web application name and URL
- Custom attributes and Asset Tags

## 2. Crawl Settings

- Crawl Scope
- Crawl Links ([Selenium scripts](#))

## 3. Default Scan Settings

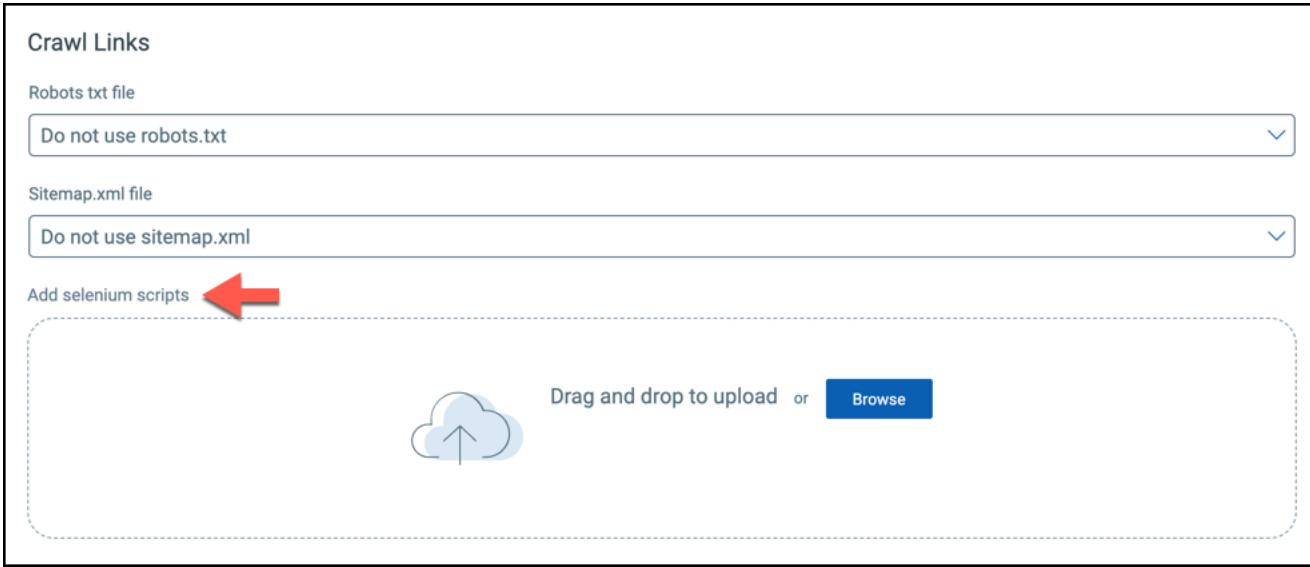
- Option Profile ([Form Uniqueness](#), [Enhanced Crawling](#), [SmartScan](#))
- [Progressive Scanning](#)

## 4. Additional Configurations



# Advanced Web App Setup

# Crawl Settings: Selenium Scripts



The screenshot shows the 'Crawl Links' section of a configuration interface. It includes fields for 'Robots txt file' (set to 'Do not use robots.txt') and 'Sitemap.xml file' (set to 'Do not use sitemap.xml'). Below these is a section labeled 'Add selenium scripts' with a red arrow pointing to it. This section contains a dashed rectangular area with a cloud icon and an upward arrow, a 'Browse' button, and the text 'Drag and drop to upload or'.

Add a script created with Qualys Browser Recorder.

# Lab Tutorials

Please follow **pages 18 – 19** from the Lab Tutorial Supplement

- Lab 8 – QBR Crawl Script, p. 18

5 min.



# Option Profile: Form Crawl Scope

Form Crawl Scope

Include form action URI in form uniqueness calculation.

When enabled, form uniqueness is calculated using the form action URI in addition to form field names.

First name:

Last name:

Click the "Submit" button and the form-data will be sent to a page on the server called "action\_page.php".

```
<form action="/action_page.php" method="get">
  <label for="fname">First name:</label>
  <input type="text" id="fname" name="fname"><br><br>
  <label for="lname">Last name:</label>
  <input type="text" id="lname" name="lname"><br><br>
  <input type="submit" value="Submit">
</form>
```

# Option Profile: Enhanced Crawling

## Crawling Options

Enhanced Crawling

When enabled we will attempt to load and render individual directories. If unique content is found, we'll begin crawling from there to improve scan coverage.

If the following link is found during crawling: <https://www.example.com/foo/abc/xyz/register.php>

Step 1 : We make a request <https://www.example.com/foo/abc/xyz/>

Step 2 : Remove the directory "[xyz/](#)" from the URL and crawl <https://www.example.com/foo/abc/>

Step 3 : Further remove the directory "[abc/](#)" from the URL and crawl, <https://www.example.com/foo/>

Objects found during this process will get added to the crawl queue thus improving the scan coverage.

# Option Profile: SmartScan

- Used for enhanced AJAX or Single Page Applications (SPA)
- Supports sites using AngularJS and bootstrap
- View QID 150148 “AJAX links crawled” to verify SmartScan is working

## SmartScan Support

When enabled we'll perform advanced scanning, using enhanced AJAX/SPA deep crawling and vulnerability testing, for a number of actions per page. This option is recommended for scanning sites with advanced frameworks and technologies.

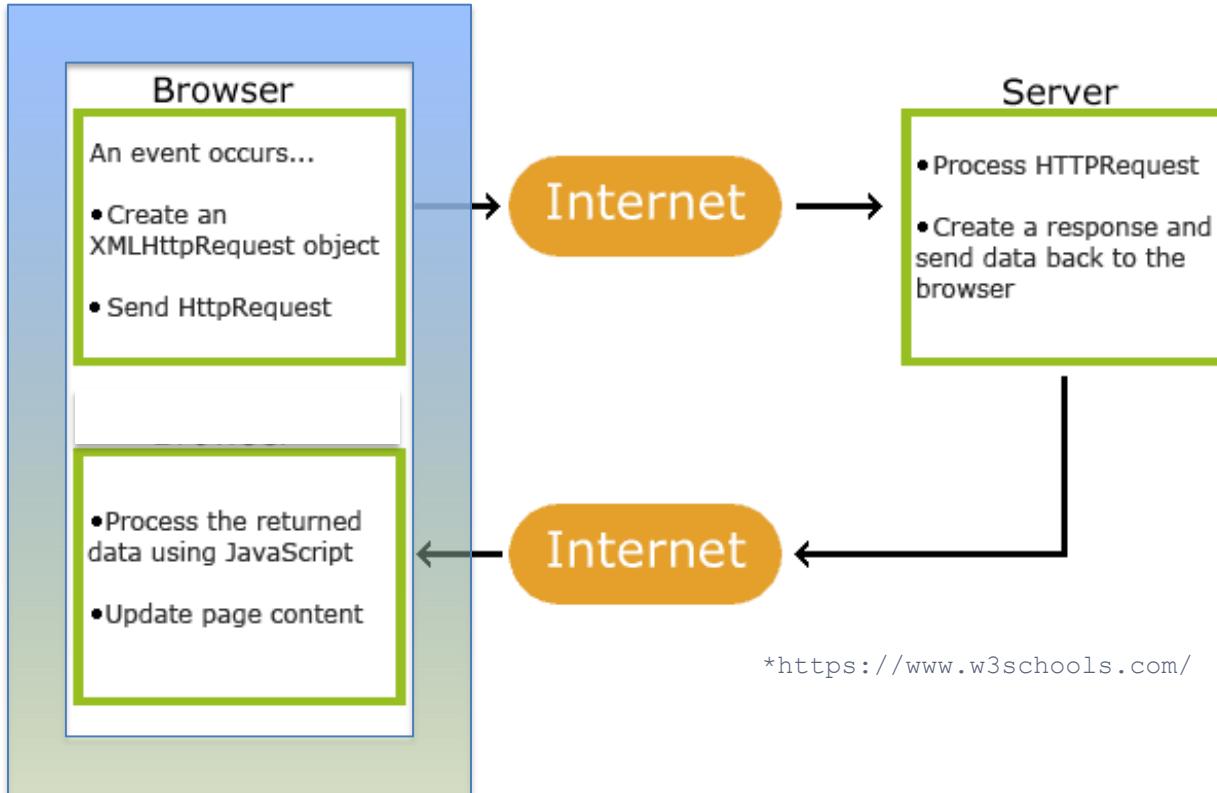
Enable SmartScan Support

You can customize the number of actions that can be tested per page. Note the higher the number you set, the longer the scan duration.

SmartScan Depth\*

5

# How AJAX Works



- Read data from a Web server - after the page has loaded
- Update a Web page without reloading the page
- Send data to a Web server - in the background



# Progressive Scanning

# Progressive Scanning

← Add New: Web Application

STEPS 3/5

- 1 Basic Info
- 2 Crawl Settings
- 3 Default Scan Settings
- 4 Additional Configurations
- 5 Review & Confirm

## Default Scan Settings

Select Option Profile \*

Select option(s)

Select Scanner Appliance

External    Individual    Tags (Scanner Pool)

Lock this scanner appliance for this web application.

Duration ⓘ

Do not Cancel Scan

Crawl Settings

Progressive Scanning ⓘ

Proxy

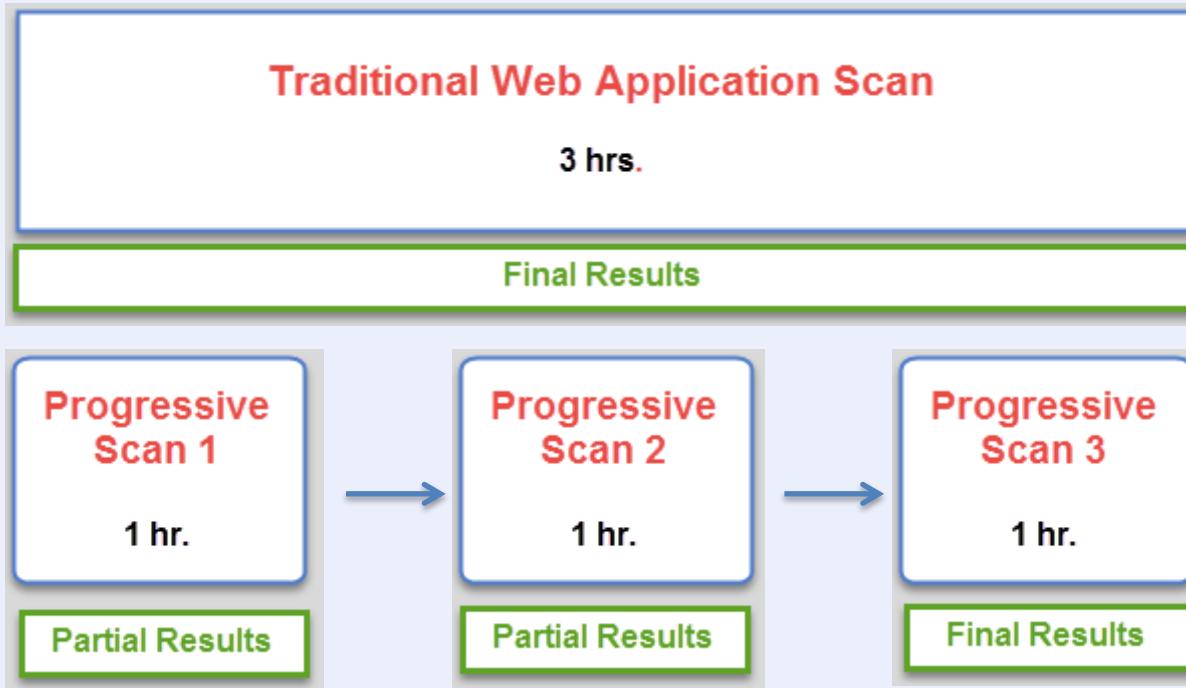
Select option(s)



# Progressive Scanning Behavior

- Remedies small or insufficient scanning windows, by expanding testing coverage, over time (scans are performed in multiple, progressive stages).
- Each successive scan builds upon the information obtained from the previous scan.
- Web app's history or findings are updated in stages, with each scan.
- Prioritizes pages not previously crawled and new functionality.

# Progressive Scan Example



Works best with Frequently Scheduled Scans

# Schedule a Progressive Scan

The screenshot shows the 'Schedule Vulnerability Scan Creation' interface. A red box highlights the title bar 'Schedule Vulnerability Scan Creation'. A red arrow points from the title bar to the top-left corner of the main content area. The left sidebar shows 'Step 3 of 6' with items 1 through 6, where items 1, 2, 3, and 5 have green checkmarks, while 4 and 6 are grayed out. The main content area is titled 'Configure scan settings' and contains a 'DNS Override' section. Below it is a sub-modal window titled 'Schedule Vulnerability Scan Creation' with its own sidebar showing 'Step 4 of 6' (items 1 through 5) and a green checkmark next to item 1. This sub-modal has a 'Configure task start date and occurrence' section. It includes a 'Recurrence' field set to 'Daily' (highlighted by a red arrow), a 'Duration' section with a 'Cancel Scan After' field set to '1 hour' (also highlighted by a red arrow), and a note about canceling the scan after N hours or a certain time. At the bottom of the sub-modal are 'Previous' and 'Continue' buttons.

Schedule Vulnerability Scan Creation

Turn help tips: On | Off Launch help ×

Step 3 of 6

- 1 Task details ✓
- 2 Target ✓
- 3 Settings ✓
- 4 Scheduling
- 5 Notification
- 6 Review And Confirm

Configure scan settings

DNS Override

Schedule Vulnerability Scan Creation

Turn help tips: On | Off Launch help ×

Step 4 of 6

- 1 Task details ✓
- 2 Target ✓
- 3 Settings ✓
- 4 Scheduling ✓
- 5 Notification

Configure task start date and occurrence

Recurrence

(\*) REQUIRED FIELDS

Mode\* Daily

Recurrence

Ends after 1 occurrences

Every 1 days

Duration

Cancel the scan after N hours or at a certain time. By default the scan will run until it completes, or the maximum scan time is reached.

Cancel Option

Cancel Scan After 1 hour

Cancel Previous Continue

## Agenda

### Qualys Web Application Scanning

## Session Two:

- Additional Configurations
- Web App Testing
- WAS Reporting
- Tags and Users
- WAS Integrations



# Additional Configurations

# Additional Configurations

← Edit: Web Application

STEPS 4/5

- 1 Basic Info
- 2 Crawl Settings
- 3 Default Scan Settings
- 4 Additional Configurations
- 5 Review & Confirm

## Additional Configurations

**Authentication Records**  
Select one or more authentication records to be used for scanning this web application. Each record will define one or more authentication methods (Basic, Server, NTLM).

**Header Injection**  
This is intended for situations where a workaround is needed for complex authentication schemes or to impersonate a web browser.

**API Endpoint Definition**  
Select any one non-swagger based API, if this field is empty the value will be considered as null

**Set up Exclusion Lists**  
Global exclusions can be configured as global settings. Choose whether to use global exclusions and add more exclusions for this web app if you like.

**Default Dns Override**  
Select one or more DNS override records with mappings you'd like to use by default when scanning this web application.

**Redundant Links**  
Specify links in the web applications for which contents are the same and because of which scan may spend too much time crawling and assessing these URLs. Links shall be specified as regular expressions so that you can specify an expression to match a list of links.

**Path Fuzzing Rules**  
Path fuzzing rules allow the scanner to interpret URI path components as application parameters when your web application uses a URL rewrite. Path fuzzing rules tell the scanner the path components to be tested. The scanner will fuzz the URI path components only if you define the path fuzzing rules.

**Form Training**  
Provide a list of form field values to be used for submitting HTML Forms during crawling.

**Malware Monitoring**  
By enabling Malware Monitoring on this web application, you will allow QualysGuard to perform a regular scan for all malware on your external web site. The application owner will receive an email notification when malicious software is detected. Note Malware Monitoring is available for external sites only.

- Authentication Records
- Header Injection
- API Endpoint Definition
- Set up Exclusion Lists
- Default DNS Override
- Redundant Links
- Path Fuzzing Rules
- Form Training
- Malware Monitoring

# Authentication

Best Practice: perform authenticated scans using non-privilege accounts.

## Form Records

- Standard Login
- Custom
- Selenium Script - Qualys Browser Recorder

## Server Records

- Basic
- Digest
- NTLM

Information Gathered Details

150009 Links Crawled

Results

Duration of crawl phase (seconds): 186.00  
Number of links: 32  
(This number excludes form requests and links re-requested during authentication)

<http://54.84.232.118:8080/bodgeit/>  
<http://54.84.232.118:8080/bodgeit/about.jsp>  
<http://54.84.232.118:8080/bodgeit/admin.jsp>   
<http://54.84.232.118:8080/bodgeit/advanced.jsp>  
<http://54.84.232.118:8080/bodgeit/basket.jsp>  
<http://54.84.232.118:8080/bodgeit/contact.jsp>  
<http://54.84.232.118:8080/bodgeit/home.jsp>  
<http://54.84.232.118:8080/bodgeit/login.jsp>  
<http://54.84.232.118:8080/bodgeit/password.jsp>  
<http://54.84.232.118:8080/bodgeit/product.jsp?pr>

# Lab Tutorials

Please follow **pages 18 – 19** from the Lab Tutorial Supplement

- Lab 9 – QBR Authentication Script, p. 19

5 min.



# Header Injection

## Header Injection



This is intended for situations where a workaround is needed for complex authentication schemes or to impersonate a web browser.

Example: Cookie: ASP.NET\_SessionId=yw13b045nq1zluvxp4vi4o55; .ASPXFORMSAU

131072 characters remaining

**Header injection** – HTTP headers injected by our scanning service to facilitate the web application scan. This option is commonly used to provide a workaround for complex authentication schemes.



# API Endpoint Testing

# API Testing

API Endpoint Definition

Select any one non-swagger based API, if this field is empty the value will be considered as null

1 Postman Collection       2 Burp Proxy Capture       3 Swagger/OpenAPI File

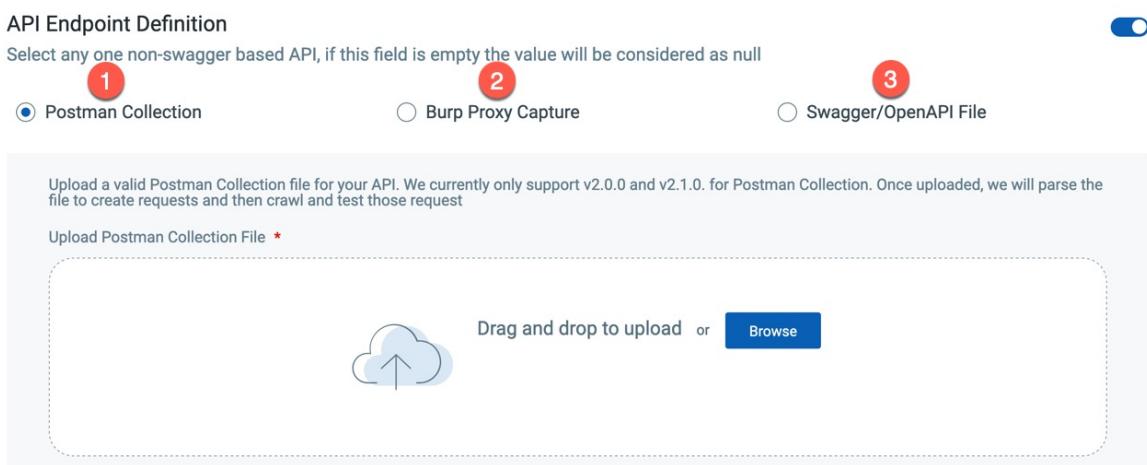
4 Turn off API endpoint definition

Upload a valid Postman Collection file for your API. We currently only support v2.0.0 and v2.1.0. for Postman Collection. Once uploaded, we will parse the file to create requests and then crawl and test those request

Upload Postman Collection File \*

Drag and drop to upload or

Upload Postman Environment Variables File



- APIs are vulnerable to many of the same attacks used against Web applications.
- You can specify API endpoints using:
  1. Swagger file
  2. Postman Collection
  3. Burp Proxy Capture

# Swagger-Based REST APIs

- Qualys WAS can identify “Swagger-based” REST API endpoints using a Swagger file.
- The Swagger file provides all the details about the APIs and how to invoke them.  
(Supported format: JSON)

**API Endpoint Definition**

Select any one non-swagger based API, if this field is empty the value will be considered as null

Postman Collection     Burp Proxy Capture     Swagger/OpenAPI File

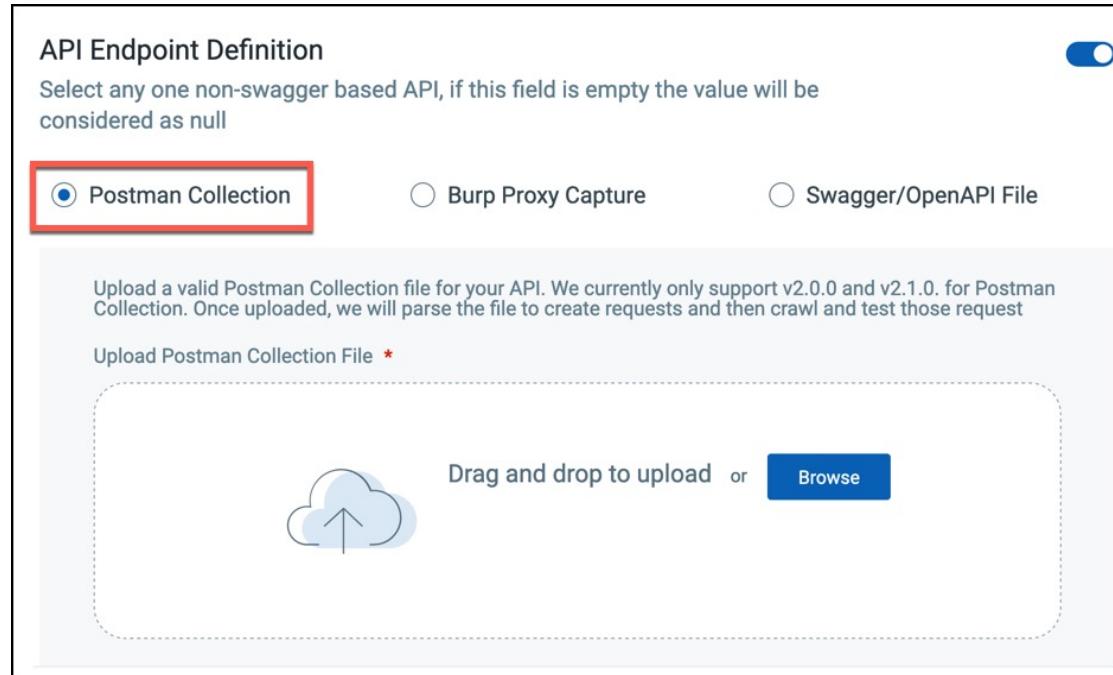
You have the option to upload a Swagger/OpenAPI File for your API. We currently support version 2.0 and 3.0 Once uploaded we will parse it to create requests and then crawl and test those requests.

Upload Swagger/OpenAPI File



Drag and drop to upload or [Browse](#)

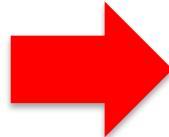
# API Endpoint Definition Using Postman



- If you are using Postman to create and document APIs, Postman collections can be uploaded to pass API endpoint definitions to the WAS scanning engine.

# API Endpoint Definition Using Burp

- Qualys WAS can also discover REST API endpoints using a Burp proxy capture.
- Burp proxy can be configured to capture all requests to the REST API services.
- The captured request can be saved and exported in the form of XML file.



- Documentation and resources
  - <https://www.qualys.com/docs/qualys-was-crawling-rest-services.pdf>

# API Endpoint Definition Using Burp

- XML file will contain all the information about the endpoints which should be tested in Qualys WAS.

**API Endpoint Definition**

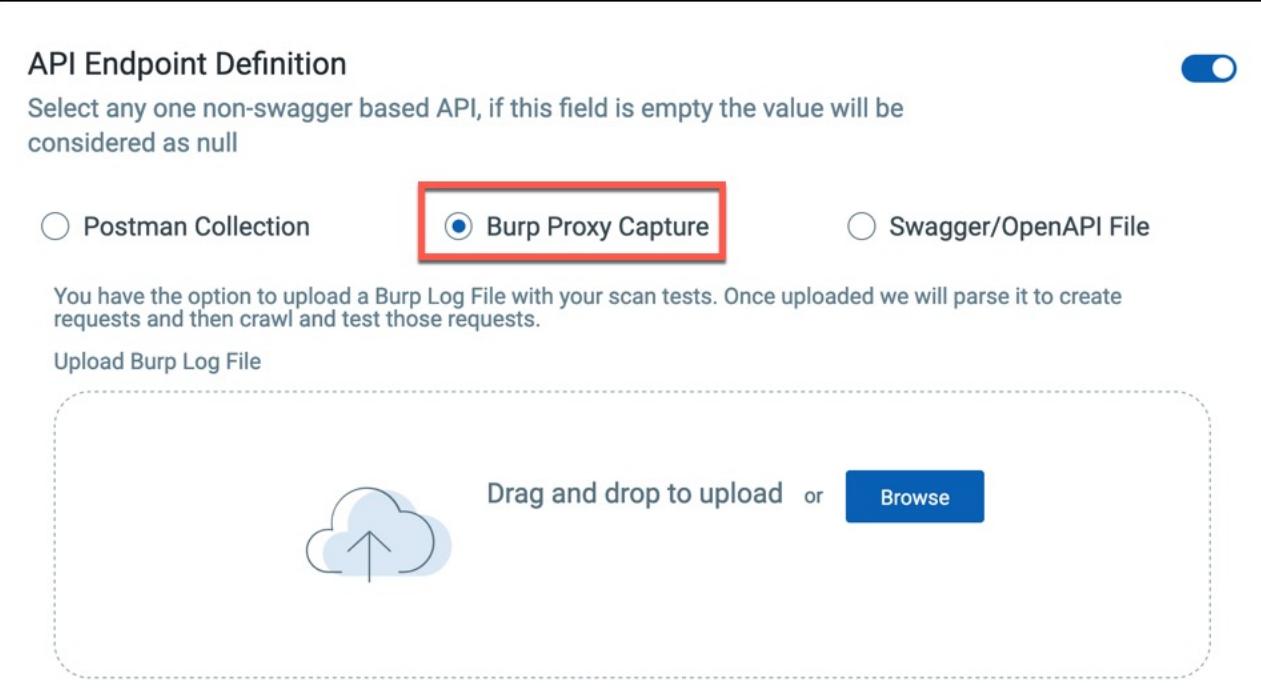
Select any one non-swagger based API, if this field is empty the value will be considered as null

Postman Collection       Burp Proxy Capture       Swagger/OpenAPI File

You have the option to upload a Burp Log File with your scan tests. Once uploaded we will parse it to create requests and then crawl and test those requests.

Upload Burp Log File

Drag and drop to upload or



# Explicit URLs for REST & SOAP

Target Definition	(*) REQUIRED FIELDS
Web Application URL (or Swagger file URL) <code>http://demo06.s02.sjc01.qualys.com/</code>	
Crawl Scope*	
<b>Limit at or below URL hostname (demo06.s02.sjc01.qualys.co)</b>	<input type="button" value="▼"/>
Scope will be limited to the hostname within the URL: <code>http://demo06.s02.sjc01.qualys.com/</code> , using HTTP or HTTPS and any port. All links discovered on the <code>demo06.s02.sjc01.qualys.com</code> domain will be in scope. For example, all links discovered in <code>http://demo06.s02.sjc01.qualys.com/support/</code> and <code>https://demo06.s02.sjc01.qualys.com:8080/logout/</code> will be in scope. Links outside the <code>demo06.s02.sjc01.qualys.com</code> domain are not in scope. This means, for example, links like <code>http://demo06.s02.sjc01.qualys.com</code> and <code>http://cdn.demo06.s02.sjc01.qualys.com</code> will not be in scope.	
Explicit URLs to Crawl / REST Paths and Parameters / SOAP WSDL Location	
<code>https://demo06.s02.sjc01.qualys.com/aade3aEfjafae.htm</code> <code>https://demo06.s02.sjc01.qualys.com/webservices/wsdl</code>	

- You can manually add the REST Paths and Parameters, to the “Explicit URLs to Crawl” field.
- Qualys WAS also supports basic security testing of SOAP based web services that have a Web Service Description Language (WSDL) file added to the “Explicit URLs to Crawl” field.
- Both “REST Paths and Parameters” and “SOAP WSDL files” should fall within the crawl scope.



# Setup Exclusion Lists

# Setup Exclusion Lists

## URLs

Global exclusions can be configured as global settings. Choose whether to use global exclusions and add more exclusions for this web app if you like.

**Allow List** 1

Set up a allow list to allow links to be scanned even if a exclude list would normally block it. If you define a allow list and no exclude list, then a default exclude list equivalent to "block all URLs" is assumed.

URLs  
 Regular Expressions

**Exclude List** 2

Set up a exclude list to prevent those URLs or their sub-directories from being scanned. Any link that matches a black list entry will not be scanned unless it also matches a white list entry.

URLs  
 Regular Expressions

**POST data exclude List** 3

Set up a list of regular expressions to block any form submission for POST requests with body that matches any of these entries.

URLs

**Logout Regular Expressions** 4

Set up a regular expression to identify the logout link. A matching link will not be crawled or scanned.

URLs

**Parameters** 5

Select one or more parameter exclusion records you'd like to use by default when scanning this web application.

Is Regex	Type	Parameter Value	Add
NO	ANY		<button>Add</button>



# Default DNS Override

# Default DNS Override

New DNS Override Settings

Step 2 of 3

Tell us the DNS settings you'd like to use

DNS Mappings (\*) REQUIRED FIELDS

Define the mappings you prefer to use for scanning.

Hostname or FQDN	IP Address	
example: my.host.com	example: 10.10.10.10	<a href="#">+ Add another</a>
www.yourwebapp.com	10.10.10.1	<a href="#">Remove</a>

Cancel Previous Continue

Quickly override the current DNS entry for any WAS Web app, without the hassle of modifying DNS records or HOSTS files.



# Redundant Links

# Redundant Links

- What if different “dynamically” generated links all go to the same Web page?

**Redundant Links** 

Specify links in the web applications for which contents are the same and because of which scan may spend too much time crawling and assessing these URLs. Links shall be specified as regular expressions so that you can specify an expression to match a list of links.

http://www.myshop.com/products/prod\_[1-100].html 

- Specify redundant link patterns to avoid crawling and scanning the same pages (multiple times).



# Path Fuzzing Rules

# Path Fuzzing Rules

- For testing Web app technologies that support **URL Rewriting**, such as ASP and .Net MVC.
- Example: Let us consider news web site:
  - `http://www.abc.com/issue/17/section/sports/article/28`
  - `http://www.abc.com/search.php?issue=17&section=sports&article=28`
- The Path Fuzzing Rule is:
  - `http://www.abc.com/issue/{issue}/section/{section}/article/{article}`
  - Qualys Scanner Appliance will perform fault injection tests (SQLi, XSS, etc...) within the curly braces.



# Form Training

# Form Training

- Configure preset values for common form fields.
- Similar outcome as Qualys Browser Recorder or Initial Parameters.

**New Field**

Name	Value
<input type="text" value="Product"/>	<input type="text" value="Widget"/>
249 characters remaining	2042 characters remaining

**New Masked Field**

Name	Value	Confirm Value
<input type="text" value="SSN"/>	<input type="text" value="....."/>	<input type="text" value="....."/>
253 characters remaining	2039 characters remaining	2038 characters remaining



# Malware Monitoring

# Malware Monitoring

**Malware Monitoring**

By enabling Malware Monitoring on this web application, you will allow QualysGuard to perform a regular scan for all malware on your external web site. The application owner will receive an email notification when malicious software is detected. Note Malware Monitoring is available for external sites only.

**Recurrence**

Mode  
Weekly

Schedule Ends After  
52 Occurrence

Scan to be scheduled Every  
1 Weeks

On Days

Monday  Tuesday  Wednesday  Thursday  Friday  Saturday  Sunday

**Launch Information**

Start Date \*  
11/15/2022 

Start \*  
12:00 am 

Select Timezone \*  
(GMT -06:00) Central Standard Time (CST America/Chicago)

Send Notification

Enable Qualys Malware Scans for external facing Web apps.

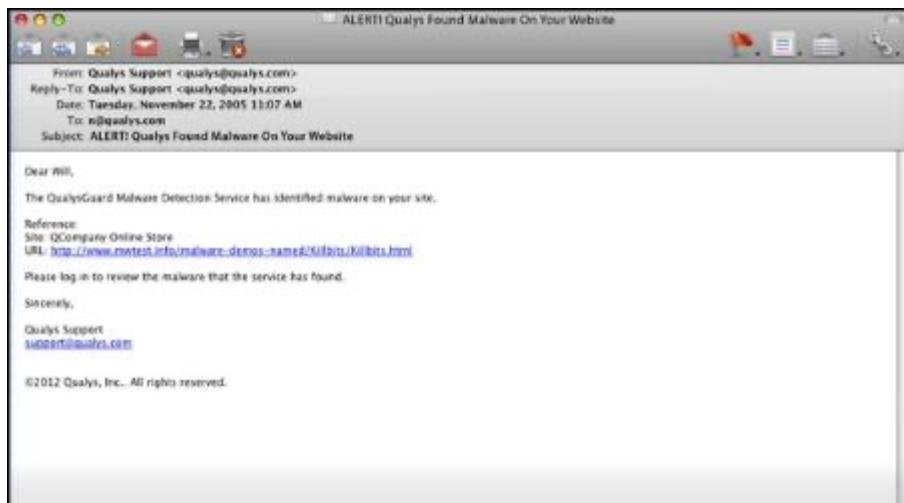
# Malware Detection

Web Application URL



Qualys MDS Virtual  
Machine Farm

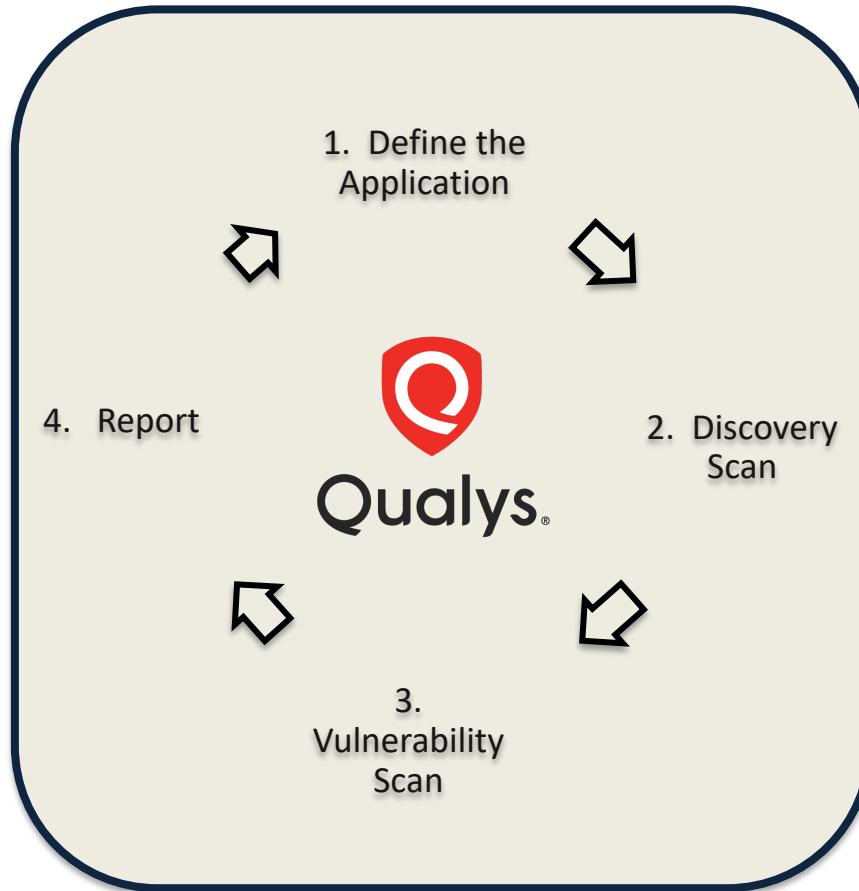
- ✓ MDS scans targeted Web app (staying within its defined scope).
- ✓ MDS provides both signature-based and behavioral analysis (i.e., “**zero-day**” detections) .
- ✓ Alerts are sent via email if/when Malware is found.





# Web Application Testing

# Qualys WAS Lifecycle



# Detection Scope

## Detection Scope

Select the scope of detections for the web application scan with this profile. Specify if the scan should perform a full assessment for all WAS detections, or if the scan shall focus on the specific WAS detections/vulnerabilities.

### Detection \*

Everything



Core

Categories

Custom Search Lists

XSS Power Mode

Everything

Include additional XSS payloads (may significantly increase scan time).

i Note: Including everything will cause longer scan times.

Assessment tests are performed according to the “Detection Scope” settings configured in the Web app’s Option Profile.

# WASC



WASC www.webappsec.org divides Web vulnerabilities into six categories

Authentication

Authorization

Client-side Attacks

Command Execution

Information Disclosure

Logical Attacks

## Attacks

[Abuse of Functionality](#)

[Brute Force](#)

[Buffer Overflow](#)

[Content Spoofing](#)

[Credential/Session Prediction](#)

[Cross-Site Scripting](#)

[Cross-Site Request Forgery](#)

[Denial of Service](#)

[Fingerprinting](#)

[Format String](#)

[HTTP Response Smuggling](#)

[HTTP Response Splitting](#)

[HTTP Request Smuggling](#)

[HTTP Request Splitting](#)

[Integer Overflows](#)

[LDAP Injection](#)

[Mail Command Injection](#)

[Null Byte Injection](#)

[OS Commanding](#)

[Path Traversal](#)

[Predictable Resource Location](#)

[Remote File Inclusion \(RFI\)](#)

[Routing Detour](#)

[Session Fixation](#)

[SOAP Array Abuse](#)

[SSI Injection](#)

[SQL Injection](#)

[URL Redirector Abuse](#)

[XPath Injection](#)

[XML Attribute Blowup](#)

[XML External Entities](#)

[XML Entity Expansion](#)

[XML Injection](#)

[XQuery Injection](#)



# Common Weakness Enumeration

A Community-Developed List of Software & Hardware Weakness Types

Rank	ID	Name	Score
[1]	<a href="#">CWE-119</a>	Improper Restriction of Operations within the Bounds of a Memory Buffer	75.56
[2]	<a href="#">CWE-79</a>	Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')	45.69
[3]	<a href="#">CWE-20</a>	Improper Input Validation	43.61
[4]	<a href="#">CWE-200</a>	Information Exposure	32.12
[5]	<a href="#">CWE-125</a>	Out-of-bounds Read	26.53
[6]	<a href="#">CWE-89</a>	Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')	24.54
[7]	<a href="#">CWE-416</a>	Use After Free	17.94
[8]	<a href="#">CWE-190</a>	Integer Overflow or Wraparound	17.35
[9]	<a href="#">CWE-352</a>	Cross-Site Request Forgery (CSRF)	15.54
[10]	<a href="#">CWE-22</a>	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')	14.10
[11]	<a href="#">CWE-78</a>	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')	11.47
[12]	<a href="#">CWE-787</a>	Out-of-bounds Write	11.08
[13]	<a href="#">CWE-287</a>	Improper Authentication	10.78
[14]	<a href="#">CWE-476</a>	NULL Pointer Dereference	9.74
[15]	<a href="#">CWE-732</a>	Incorrect Permission Assignment for Critical Resource	6.33
[16]	<a href="#">CWE-434</a>	Unrestricted Upload of File with Dangerous Type	5.50
[17]	<a href="#">CWE-611</a>	Improper Restriction of XML External Entity Reference	5.48
[18]	<a href="#">CWE-94</a>	Improper Control of Generation of Code ('Code Injection')	5.36
[19]	<a href="#">CWE-798</a>	Use of Hard-coded Credentials	5.12
[20]	<a href="#">CWE-400</a>	Uncontrolled Resource Consumption	5.04
[21]	<a href="#">CWE-772</a>	Missing Release of Resource after Effective Lifetime	5.04
[22]	<a href="#">CWE-426</a>	Untrusted Search Path	4.40
[23]	<a href="#">CWE-502</a>	Deserialization of Untrusted Data	4.30
[24]	<a href="#">CWE-269</a>	Improper Privilege Management	4.23
[25]	<a href="#">CWE-295</a>	Improper Certificate Validation	4.06

# OWASP

## The 2021 OWASP Top 10



A01  
**Broken Access Control**



A06  
**Vulnerable and Outdated Components**



A02  
**Cryptographic Failures**



A07  
**Identification and Authentication Failures**



A03  
**Injection**



A08  
**Software and Data Integrity Failures**



A04  
**Insecure Design**



A09  
**Security Logging and Monitoring Failures**

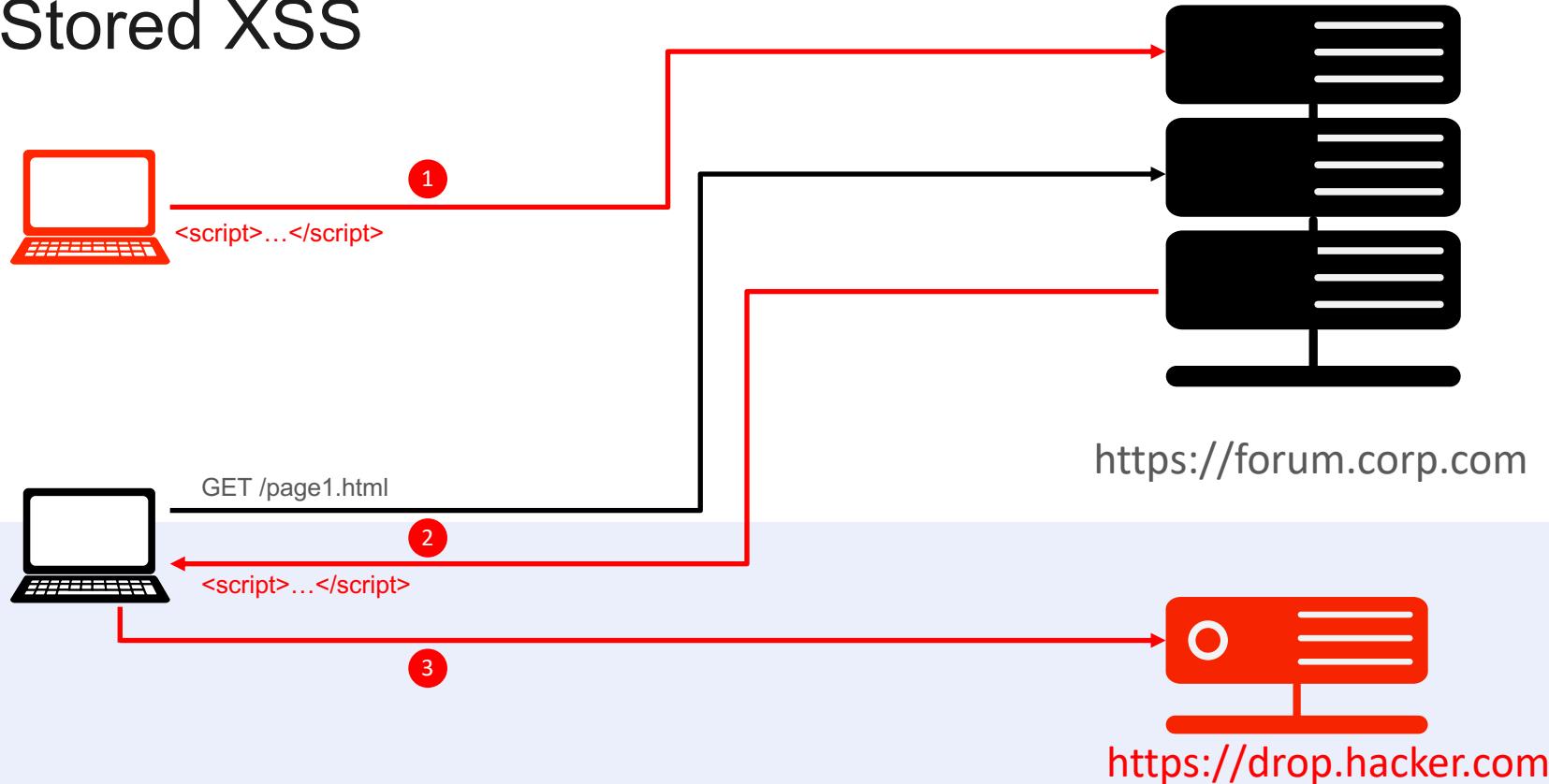


A05  
**Security Misconfiguration**



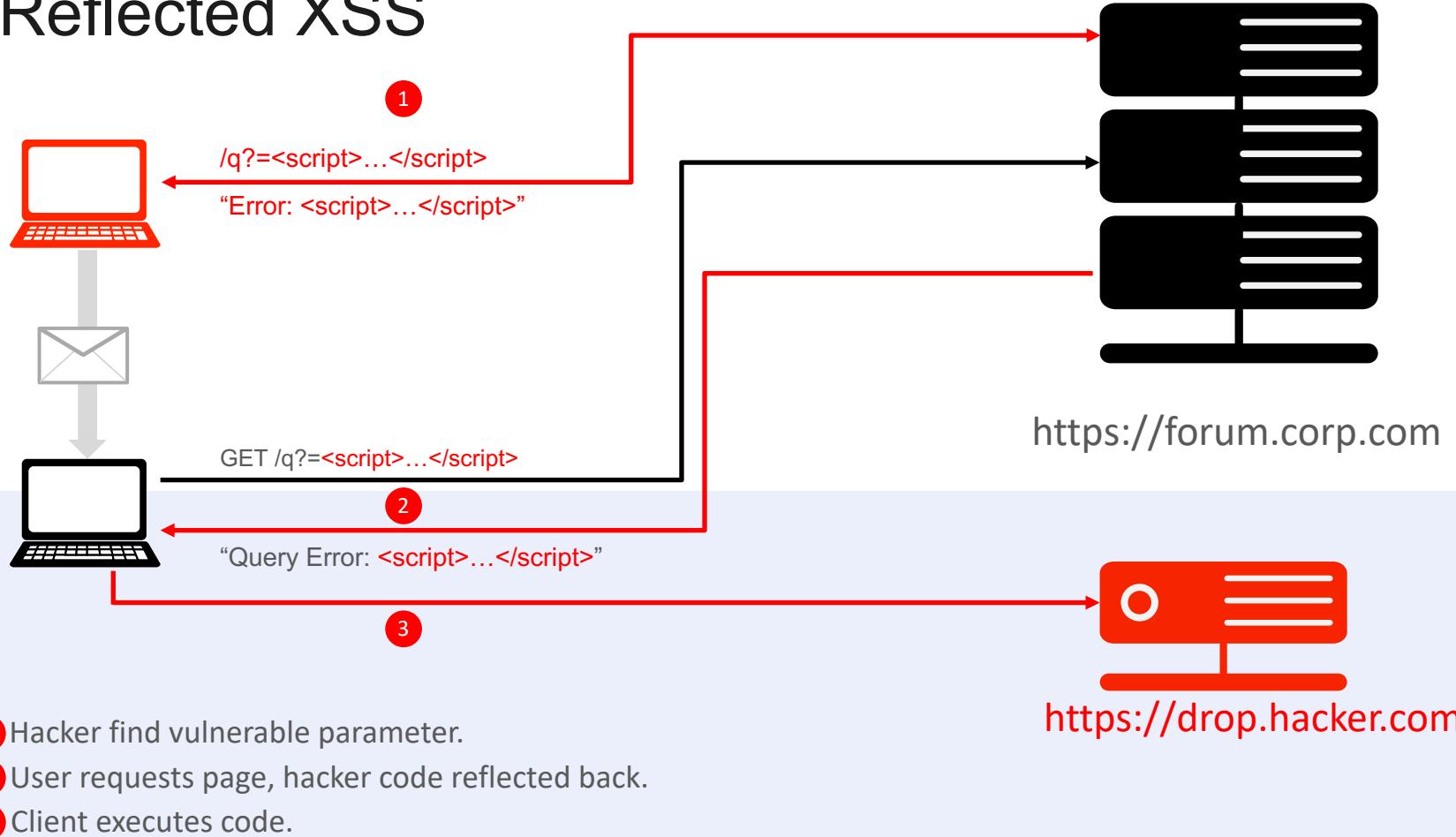
A10  
**Server-Side Request Forgery (SSRF)**

# Stored XSS



- ① Hacker uploads malicious script into Web app's repository or database
- ② User requests page, stored hacker script sent
- ③ Client executes code

# Reflected XSS



① Hacker find vulnerable parameter.

② User requests page, hacker code reflected back.

③ Client executes code.

# BodgeIT XSS Vulnerability

http://54.173.177.208:8080/bodgeit/search.jsp

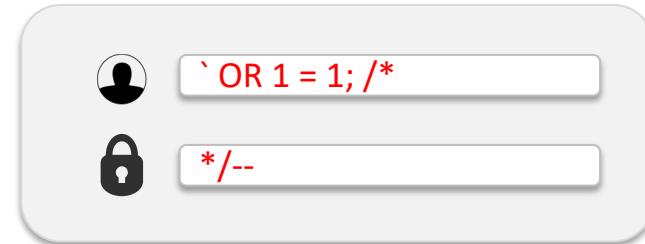
The screenshot shows a web page titled "Search". It has a search bar labeled "Search for" with a placeholder "Search for" and a "Search" button below it. To the right of the search bar is a link "Advanced Search". A large black callout bubble points from the bottom left towards the search bar. Inside the bubble, the following JavaScript code is visible: <script>alert (document.cookie)</script>. This indicates that user input is being reflected directly into the browser's DOM, which is a characteristic of a reflected XSS vulnerability.

- BodgeIT Store “Search” field does not sanitize input.
- Injected scripts are rendered or reflected within the client browser.

# SQL injection



A screenshot of a login interface. It features two input fields: one for 'username' containing 'johndoe' and another for 'password' containing 'qualys'. Each field has a small icon next to it: a user silhouette for the username field and a padlock for the password field.



A screenshot of a login interface. The 'username' field contains the value '` OR 1 = 1; /\*' and the 'password' field contains '\*//--'. Both fields have their respective icons (user silhouette and padlock).

```
select * from users where username='johndoe' and password = 'qualys'
```

```
select * from users where username='` OR 1 = 1; /*' and password = '*//--'
```

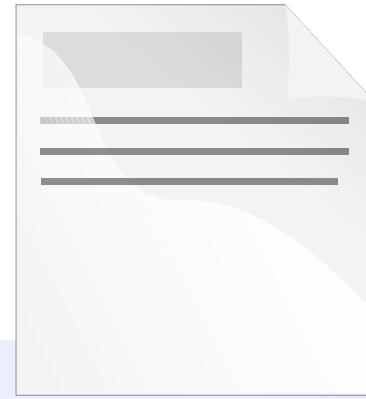
# Blind SQLi

True



GET /article.php?id=1 **and 1 = 1**

False



GET /article.php?id=1 **and 1 = 2**

- By asking the server true and false questions and getting different results we are able to determine if vulnerability.

# BodgeIT SQLi Vulnerability

`http://54.173.177.208:8080/bodgeit/login.jsp`

**Login**

admin@thebodgeitstore.com' OR '1' = '1

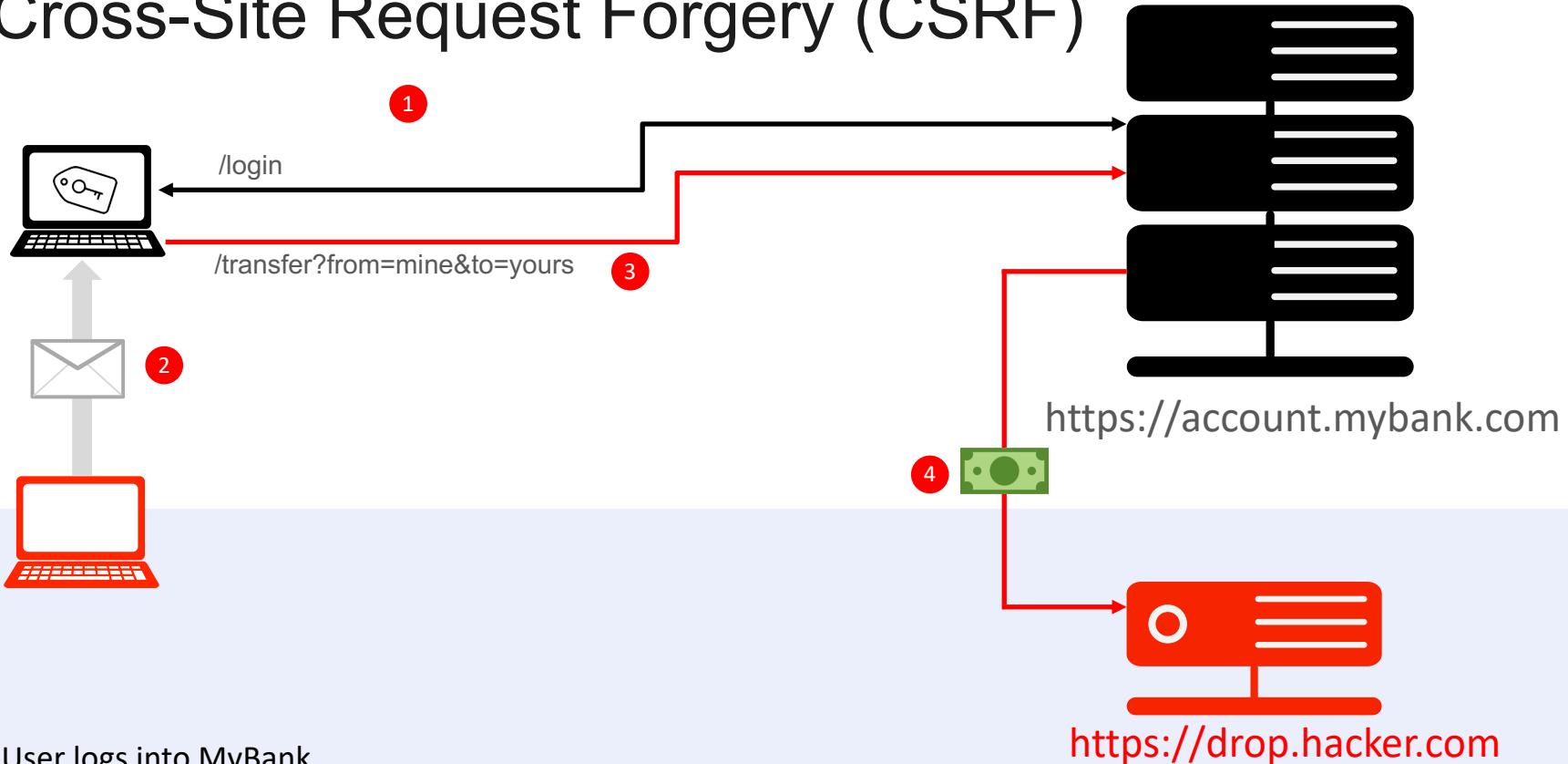
Username:

Password:

If you dont have an account with us then please [Register](#) now for a free account.

The BodgeIT Store "Login" page uses string concatenation to build a SQL request, making it easy to inject a malicious SQL command into the "Username" field.

# Cross-Site Request Forgery (CSRF)



1 User logs into MyBank.

2 User is sent a phishing email with false link.

3 Clicking on the link a request is sent to the banks website using the users authentication token.

4 User money is transferred to the hackers account.

# Web App Report Details

## Vulnerability Details

150012 Blind SQL Injection  
URL: <http://54.173.177.208:8080/bodgeit/login.jsp> Install Patch Ignore Retest Active

Finding	Unique	Patch #	Group	CWE	OWASP	WASC	CVSS V

## Vulnerability Details

150001 Reflected Cross-Site Scripting (XSS) Vulnerabilities  
URL: <http://192.168.1.233:8080/bodgeit/search.jsp> Install Patch Ignore Retest New

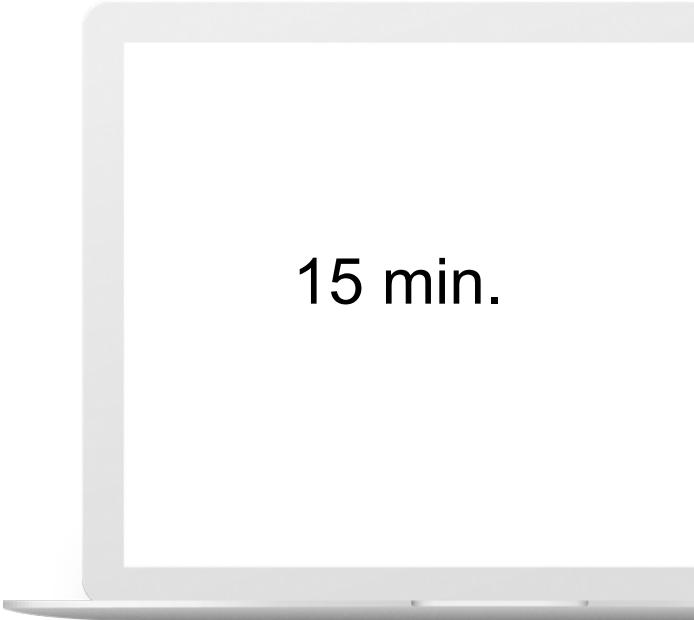
Finding #	Unique #	Patch #	Group	CWE	OWASP	WASC	CVSS V
13957418	377bc1f1-57de-48f8-875f-dae58826ea5f	-	Cross-Site Scripting	<a href="#">CWE-79</a>	<a href="#">A7 Cross-Site Scripting (XSS)</a>	<a href="#">WASC-8 CROSS-SITE SCRIPTING</a>	
6.1	CVSS V3 Temporal 5.8						

Web Application Authentication DFW Bodgeit Store Not Used

First Time Detected 17 Aug 2021 8:54AM GMT-0500 Last Time Detected 17 Aug 2021 8:54AM GMT-0500 Last Time Tested 17 Aug 2021 8:54AM GMT-0500 Times Detected [1 View History...](#) External References - CVSS V3 Attack Vector NETWORK

When viewing Web App reports, click the “CWE,” “OWASP,” or “WASC” links for more details.

# Session Break



15 min.



# Reporting

# WAS Reporting

- Scan Report
- Web Application Report
- Scorecard Report
- Catalog Report

Report Creation Turn help tips: On | Off 

Step 1 of 2 (1) Details ✓ (2) Target

Choose the type of report to create

Select Report Type (\*) REQUIRED FIELDS

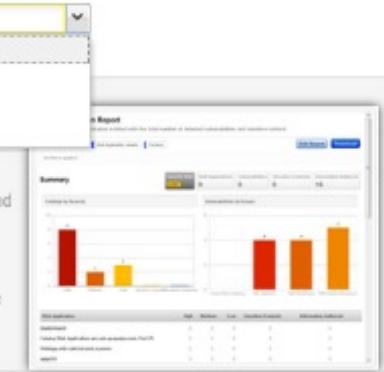
Choose a report type, then click Continue to define the report target.

Report type\* Web Application Report

Web Application Report  
Scan Report  
Scorecard Report  
Catalog Report

For each web application you'll see the total number of detected vulnerabilities and sensitive contents. Report details include detection data and verified solutions for remediation.

From the finished report, you can edit the settings and apply content filters.



Cancel Continue

# Scan Report

The screenshot shows the Qualys Web Application Scanning interface. The top navigation bar includes 'Web Application Scanning' dropdown, 'Dashboard', 'Web Applications', 'Scans' (which is the active tab), 'Burp', 'Reports', 'Configuration', and 'KnowledgeBase'. Below the navigation is a blue header bar with tabs: 'Scan Management' (selected), 'Scan List' (highlighted in white), 'Schedules', and 'Option Profiles'. On the left, there's a 'Search Results' sidebar with a search input and a 'Search' button, followed by 'Filter Results' sections for 'Quick Filters' (My Scans, Multi Scans), 'Type' (Vulnerability Scan, Discovery Scan), and 'Mode' (Scheduled, On Demand). The main content area is titled 'Scan List' and displays a table of scan instances. The table columns are 'Name', 'Actions (0)', 'New Scan', 'Name', 'Severity', and 'Scan Date'. The table lists six scan instances for 'Daily Bank of Qualys Scan' at different dates and times, all marked as 'HIGH' severity. A red callout bubble points to the third scan instance, which has a green checkmark icon next to its name. The callout contains the text: 'Analyze a specific scan instance from your "Scan List."'

Name	Severity	Scan Date
Daily Bank of Qualys Scan http://demo6.sea.qualys.com/	HIGH	06 Jul 2015 [31]
Daily Bank of Qualys Scan http://demo6.sea.qualys.com/	HIGH	07 Jun 2015 [31]
Daily Bank of Qualys Scan http://demo6.sea.qualys.com/	HIGH	09 Aug 2015 [31]
<b>Daily Bank of Qualys Scan http://demo6.sea.qualys.com/</b>	Finished	29 [31]
Daily Bank of Qualys Scan http://demo6.sea.qualys.com/	HIGH	06 Aug 2015 [31]
Daily Bank of Qualys Scan http://demo6.sea.qualys.com/	Finished	26 [31]
Daily Bank of Qualys Scan http://demo6.sea.qualys.com/	Finished	30 [31]
Daily Bank of Qualys Scan http://demo6.sea.qualys.com/	HIGH	09 Jul 2015 [31]
Daily Bank of Qualys Scan http://demo6.sea.qualys.com/	HIGH	27 May 2015 [31]

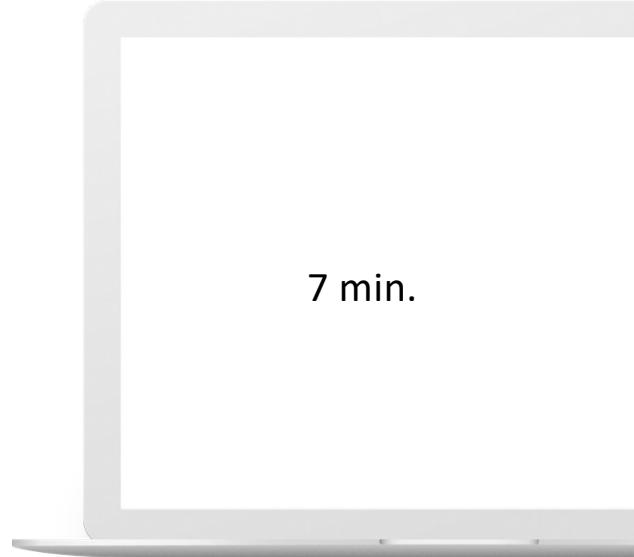
Focus on single scan instance (date + time), therefore NO vulnerability history data.

# Lab Tutorials

Please follow **pages 20 – 21** in the Lab Tutorial Supplement

- Lab 10 – Scan Report, p. 20

7 min.



# Web Application Report

 150013 Browser-Specific Cross-Site Scripting Vulnerabilities

URL: <http://54.84.232.118:8080/bodgeit/search.jsp>

Ignore Active

Finding #	1410533	Web Application	Webex application - 17 march
Group	Cross-Site Scripting	Authentication	Not Used
CWE	CWE-79		
OWASP	A3 Cross-Site Scripting (XSS)	First Time Detected	17 Mar 2015 9:59AM GMT-0500
WASC	WASC-8 Cross-Site Scripting	Last Time Detected	17 Mar 2015 11:47AM GMT-0500
CVSS Base	4.3	CVSS Temporal	4.3
		Last Scan Date	17 Mar 2015 11:47AM GMT-0500
		Times Detected	4

History Back...

Status	Authentication	Date
● Finding has been detected	auth record - 17 march Authentication not used	17 Mar 2015 11:47AM GMT-0500 was/1426610846148.5575190
● Finding has been detected	auth record - 17 march Authentication not used	17 Mar 2015 11:09AM GMT-0500 was/1426608546812.5574749
● Finding has been detected	auth record - 17 march Authentication not used	17 Mar 2015 10:22AM GMT-0500 was/1426605746132.5574247
● Finding has been detected	auth record - 17 march Authentication not used	17 Mar 2015 9:59AM GMT-0500 was/1426604347702.5574020

- Combines all scans performed on a single Web application.
- Vulnerability History and Status included (New, Active, Re-opened, Fixed).

# Lab Tutorials

Please follow **pages 20 - 21** in the Lab Tutorial Supplement

- Lab 11 – Web App Report, p. 21

5 min.



# QID 150021 – Scan Diagnostics

The scan diagnostics data provides technical details about the scanner appliance performance and behavior.

150021 Scan Diagnostics

Finding #	963158* (229849934)	Web Application	My First App
Group	Information Gathered	Authentication	Not Used
CWE	-		
OWASP	-	Detection Date	14 Feb 2017 12:54PM GMT
WASC	-		

**Details** [Show](#)

**Results**

Highlight changes from previous scan

New - this link was not found in the previous scan

Modified - this result was found by the previous scan but its value was different

Removed - this link was not found, but was reported in the previous scan

[Export...](#)

First column indicates HTTP response code,  
Path manipulation: Estimated requests (payloads x links): files with extension:(4 x 27) + files:(15 x 27) +  
directories:(88 x 3) + paths:(15 x 30) = total (1227)  
Batch #0 WS Directory Path manipulation: estimated time < 1 minute (9 tests, 30 inputs)  
WS Directory Path manipulation: 9 vulnsigs tests, completed 27 requests, 1 seconds. Completed 27 requests of 27  
estimated requests (100%). All tests completed.  
Batch #0 WS enumeration: estimated time < 10 minutes (10 tests, 30 inputs)  
WS enumeration: 10 vulnsigs tests, completed 30 requests, 0 seconds. Completed 30 requests of 300 estimated requests  
(10%). All tests completed.  
Batch #1 URI parameter manipulation (no auth): 48 vulnsigs tests, completed 47 requests, 2 seconds. Completed 47  
requests of 48 estimated requests (97.9167%). All tests completed.  
Batch #1 Form parameter manipulation (no auth): 48 vulnsigs tests, completed 658 requests, 24 seconds. Completed 658  
requests of 720 estimated requests (91.3889%). All tests completed.  
Batch #1 URI blind SQL manipulation (no auth): 9 vulnsigs tests, completed 18 requests, 1 seconds. Completed 18

# QID 150100 – Selenium Diagnostics

- Troubleshoot Selenium script
- Search for keyword: “Failure”

Information Gathered Details

150100 Selenium Diagnostics

Finding #	4148039* (412497940) <a href="#">d1df9440-9c7a-45ea-af74-e94773cc143d</a>	Web Application	AWS Bodgeit Store
Unique #			
Group	Scan Diagnostics	Detection Date	18 Aug 2021 3:40PM GMT-0500
CWE	-		
OWASP	-		
WASC	-		

Results

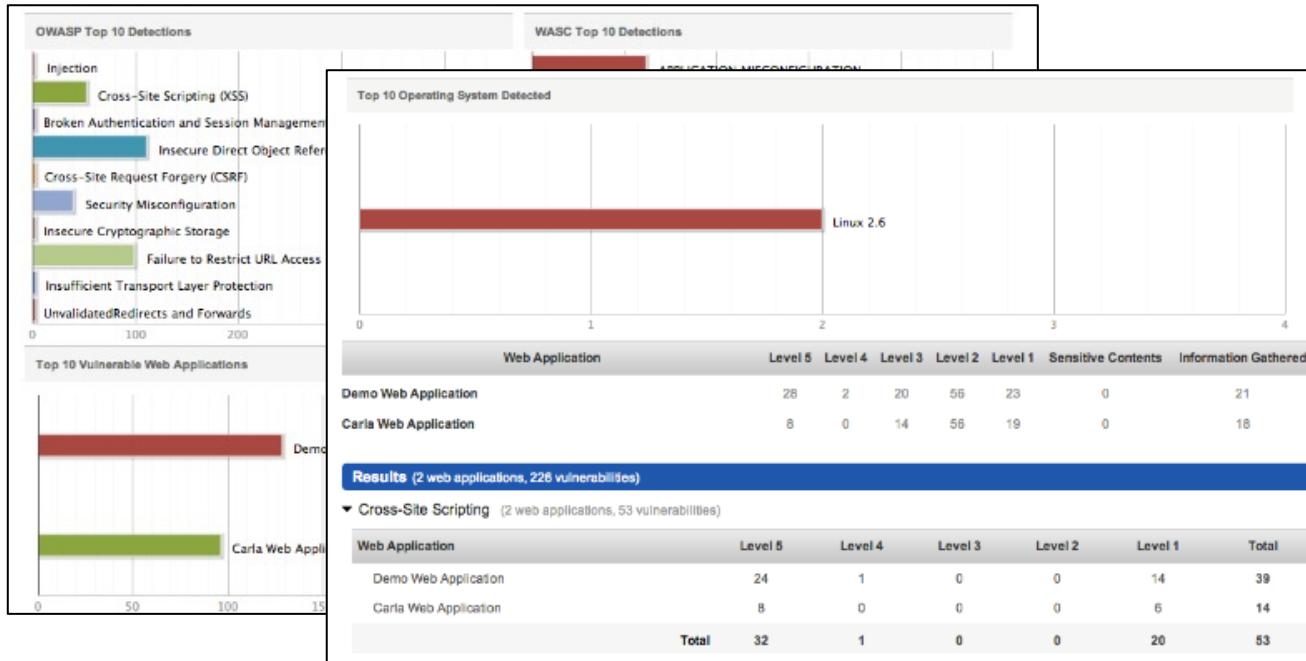
```
Log for Selenium script: authscript
Executing: |open | http://54.173.177.208:8080/bodgeit/login.jsp | |
Executing: |click | id=username | |
Executing: |waitForElementPresent | id=submit | |
Executing: |sendKeys | id=username |
Executing: |click | id=password | |
Executing: |sendKeys | id=password |
Executing: |click | id=submit | |
```

```
Log for Selenium script: crawlscript
Executing: |open | http://54.173.177.208:8080/bodgeit/basket.jsp | |
Executing: |click | link=Widgets | |
Executing: |click | link=Weird Widget | |
Executing: |click | id=submit | |
Executing: |click | id=updat | |
currentTest.recordFailure: Element id=updat not found
```

Export...



# WAS Scorecard Report



Easily Analyze the status of multiple Web application projects.

# WAS Catalog

### Catalog Report

A report on the entries in the web application catalog.

Status: New, Rogue, Approved, Ignored

Total Entries: 9 | New Entries: 9 | Rogue Entries: - | Approved Entries: - | Ignored Entries: -

#### Summary

Number of Entries: 10

IP Address	Port	FQDN	NetBios	Operating System	Creation Date	Status
10.0.30.18	80	-	-	-	20 Jul 2012	New
10.0.30.20	80	xp-sp2	XP-SP2	-	20 Jul 2012	New
10.0.30.20	443	xp-sp2	XP-SP2	-	20 Jul 2012	New
10.0.30.21	80	2kserver-sp4	2KSERVER-SP4	-	20 Jul 2012	New
10.0.30.24	80	2kserver-sp0.corp.qualys-training.com	2KSERVER-SP0	-	20 Jul 2012	New
10.0.30.26	80	win2klls	WIN2KllS	-	20 Jul 2012	New
64.39.106.243	80	2k-sp4-oe501	2K-SP4-OE501	-	20 Jul 2012	New
64.39.106.247	443	demo6.sea.qualys.com	-	-	20 Jul 2012	New
64.39.106.247	80	demo6.sea.qualys.com	-	-	20 Jul 2012	New

Manage and track Web applications throughout your entire enterprise architecture.



# Tags and Users

# Asset Tags

Asset Tags can be added or removed from most WAS objects:

- Users
- Web Applications
- Templates
- Option Profiles
- Brute Force Lists
- Search Lists
- Scanners
- Parameter Sets
- Authentication Records
- and more...



# User Roles

**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 (4 of 4)
- ▶ WAS Burp Permissions (7 of 7)
- ▶ WAS Remediation Permissions (3 of 3)
- ▶ WAS Authentication Record Permissions (3 of 3)

Set granular permissions to build custom roles.

# Combine Roles & Tags



## Administration

Control user access permissions and activity in your subscription.

- Combine user roles with Asset Tags to provide access privileges.
- Ensure that user role privileges meet or exceed those required for all tags in the user's scope.

User Edit: Tom Smykowski (quays2ws6) Turn help tips: On | Off

**Edit Mode**

- User Details
- Profile Settings
- Roles And Scopes**
- Action Log
- Account Activity

**Edit role(s) and scope**

Allow user full permissions and scope (The user will have full access to everything)

Each role grants you a set of permissions that will apply to the objects you have access to.

New role Search unassigned roles

Assigned roles	Remove all
WAS SCANNER	Remove

Unassigned roles	Add all
SCANNER	Add
UNIT MANAGER	Add
WAS MANAGER	Add
WAS USER	Add
WebEx	Add

**Edit Scope**

Allow user view access to all objects (Other permissions are granted by the user's roles)

Define what assets the user can access by tag

Unassigned Business Unit Bank of Qualys App  BoQ Auth  BodgeIT Store App  BodgeIT Auth

Internal vSphere Demo Web App

Cancel Save

A red arrow points from the "WAS SCANNER" role in the Assigned roles section to the asset tags listed below. Another red arrow points from the "Internal vSphere" tag in the asset tags section to the "Add" button in the Unassigned roles section.

# Lab Tutorials

Please follow **pages 22 – 25** in the Lab Tutorial Supplement

- Lab 12 – Tagging, p. 22
- Lab 13 – User Management, p. 24

10 min.

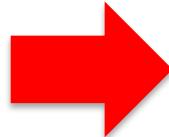




# WAS Integrations

# Burp Suite Professional

- Import Burp findings into Qualys WAS.
- View detections lists.
- Create Web Application reports to display both Burp and WAS findings.



Web Application Scanning

# Burp Suite Professional Integration

The screenshot shows the Qualys Detection Management interface. At the top, there are tabs for 'Detection Management' (selected), 'Detection List' (active), 'Burp', and 'Bugcrowd'. Below the tabs is a search bar and a 'Search' button. To the right of the search bar are buttons for 'Actions (0)', page navigation (21 - 40 of 66), and settings.

The main area is titled 'Search Results' and contains a table of findings. The columns are: Status, QID, Name, Age, Patch, and Severity. A red box highlights the first finding in the list:

Status	QID	Name	Age	Patch	Severity
Active	-	>Password field with autocomplete enabled http://34.201.91.241:8080/bodgeit/contact.jsp	1635	[Red]	Medium
-	-	HTML does not specify charset http://34.201.91.241:8080/bodgeit/register.jsp	1635	[Blue]	Low
-	-	HTML does not specify charset http://34.201.91.241:8080/bodgeit/search.jsp	1635	[Blue]	Low
New	-	Cleartext submission of password http://54.243.54.81:8080/bodgeit/register.jsp	1635	[Red]	High
New	-	Cookie without HttpOnly flag set http://54.243.54.81:8080/bodgeit/home.jsp	1635	[Red]	High
-	-	Frameable response (potential Clickjacking) http://54.243.54.81:8080/bodgeit/admin.jsp	1635	[Blue]	Low
-	-	HTML does not specify charset http://54.243.54.81:8080/bodgeit/basket.jsp	1635	[Blue]	Low
-	-	HTML does not specify charset http://54.243.54.81:8080/bodgeit/login.jsp	1635	[Blue]	Low

On the left side, there are filters for 'Target', 'Web Application', 'Tags', 'Last Scan Date', and 'Finding Type'. Under 'Finding Type', the 'Burp' checkbox is checked, indicated by a red arrow pointing to it. The 'Qualys' and 'Bugcrowd' checkboxes are also present but not checked.

# Lab Tutorial

Please follow **page 26** from the Lab Tutorial Supplement

- Lab 14 – Burp Integration, p. 26

3 min.



# Bugcrowd Integration

- Import approved Bugcrowd submissions into Qualys WAS.
- View and report on vulnerabilities identified by WAS, as well as those found via Bugcrowd bug bounty programs.

The screenshot shows the Qualys Enterprise web application scanning interface. The top navigation bar includes links for Dashboard, Web Applications, Scans, Detections, Reports, Configuration, and KnowledgeBase. The main content area is titled 'Detection Management' and features tabs for 'Detection List', 'Burp', and 'Bugcrowd'. The 'Bugcrowd' tab is selected, displaying a table of detected vulnerabilities. The table columns include Status, QID, Name, Group, Last Detected, Age, Patch, and Severity. Several rows are listed, with one row highlighted in yellow. A sidebar on the left contains filters for Tags, Last Scan Date, Finding Type (Qualys, Burp, Bugcrowd), Confirmed Vulnerability Level (1-5), Potential Vulnerability Level (1-5), and Sensitive Content Level. A preview panel at the bottom shows details for a specific vulnerability, including its reference number, state (CLOSED), sub-state (RE SOLVED), bug URL (template.com/test/), priority (1), first detected date (21 Apr 2017), last detected date (21 Apr 2017), and times detected (1).



# Qualys.<sup>®</sup>

Thank You

[training@qualys.com](mailto:training@qualys.com)