WebApps RPT-SQLi Quick Guide

FORTRA

Thank you for using Core Impact™! This Quick Guide will step you through a basic WebApps Rapid Penetration Test (RPT) with focus on SQL injection (SQLi) vulnerabilities. To perform these steps, launch Core Impact and open a valid workspace.

Before continuing, make sure you are authorized to perform penetration tests on your perspective target applications and systems.

This quick guide is meant to serve as an introduction to the penetration testing capabilities of Core Impact. There are many variations of Core Impact's functionality that are not represented here. For more details, use the Help link within Core Impact.

WebApps RPT-SQLi Summary

The WebApps RPT for SQLi vulnerabilities is comprised of three steps which, when run in sequence, provide a comprehensive test of your web applications.

WebApps RPTs organize your web applications and test settings into Scenarios. You can set up multiple Scenarios to test a single web application in different ways and easily track your results.

WebApps Information Gathering - This step scans the location of a Web Application and attempts to identify web pages that may be vulnerable to malicious attacks. If pages are located, they are shown on the Web tab of Core Impact's Entity Database under the scenario name.

WebApps Attack and Penetration - Using the details from the previous step, Core Impact attempts to exploit known vulnerabilities on the hosts with the goal of launching an Agent on each one. Core Impact's Agents can then serve as gateways to compromised hosts, allowing you to further penetrate the machine itself or its surrounding network.

WebApps Report Generation - As with all Core Impact RPTs, the Report Generation step provides detailed reports on information gathered during the previous steps. These reports are essential in compiling vulnerability data and developing remediation strategies.

Run a WebApps RPT for SQL Injection

TIP:
Before beginning the test, you will need the address of the web application that you wish to test.

WebApps Information Gathering

1. Click the **Web Applications RPT** button .

2. Click the **WebApps Information Gathering** step, then click **Next** to get started.

   This will launch the Information Gathering wizard.
3. Enter the name of your test **Scenario**.

   You can alternatively select an existing Scenario.
4. Select **Crawl a known web application**.

   Core Impact can also evaluate known hosts that might be running HTTP servers.
5. Use **Automatic Web Crawling** so the RPT scans the web app pages for you and enter the **URL** for your target Web Application.

   Interactive Web Crawling allows you to navigate within your web application while Core Impact records each page that you visit.
6. Set your Internet connection method and click **Next**.
7. On the **Automatic Crawling Options** page, use the default settings and click **Next**.

   On this step, you can specify:
   - The browser the RPT should impersonate when the test runs
   - The Max. number of pages the crawler should process
   - The Max. depth level to crawl
   - If the test can crawl to other domains
   - If the test should try to detect web server and application framework details
8. On the **Automatic Crawling Options (contd)** page, use the default settings and click **Next**.

   On this step, you can specify:
   - If the test should evaluate JavaScript for vulnerabilities
   - If the test should try to detect a robots.txt file and follow the links it contains
   - If the test should try to submit any forms it finds within the web application
   - A custom module you create to parse dynamic links within the web application
   - Whether Session management is to be used to log in to the web application
9. On the **Web Services Discovery Options** page, use the default settings and click **Finish**.

   Core Impact will also attempt to identify any SOAP-based web services that may be running within the target Web Application.

The Information Gathering step will launch the necessary modules - you will see these modules and can monitor their progress in the Executed Modules pane. When the modules are completed, look for new pages in the Web tab of the Entity Database under the Scenario name. At this stage Core Impact has located pages associated with a web application. In the next step, it will test them for vulnerabilities. Move to the Attack and Penetration step of the RPT.

WebApps Attack and Penetration

1. Click the **WebApps Attack and Penetration** step to launch the Attack and Penetration wizard.
2. Click the **Next** button to get started.
3. Click the ellipsis button ... to see a list target pages and/or scenarios. Place a check next to each scenario or page you wish to target, then click **OK**. You will return to the wizard.
4. Click the **Next** button to proceed with the test.
5. Un-check all options except **A3 - Injection**. Then click **Next**.

   Core Impact can also test for Cross-site Scripting and Remote File Inclusion vulnerabilities.
6. Use the default settings for **SQL Injection test configuration** and click **Next**.

   These options allow advanced customization of the RPT.
7. Use the default settings for **Session Management** and click **Finish**.

   These options can prevent the test from self-terminating.

The Attack and Penetration step will launch the necessary modules - monitor the Executed Modules pane to track them. When the modules are completed, expand the Vulnerable Pages folder in the Entity Database and click on the SQL Injection folder. Any pages in this folder have been identified by Core Impact as being vulnerable to SQLi attacks. Each page will also have a SQL Agent under it. These SQLi Agents are not deployed on the web application but instead represent the capability of using SQL Injection to access visiting users' browsers.

Once a SQL Agent exists, you can use Core Impact modules to exploit the SQLi vulnerability and further test the overall security of your web application. For example, there are a number of modules in the **Information gathering -> Local** folder (using the Modules tab) that can be used to gather information from a web application that is vulnerable to SQLi attacks.

WebApps Report Generation

1. Click the **WebApps Report Generation** step to launch the WebApps Report Generation wizard.
2. Click the **Next** button to get started.
3. Click the ellipsis button ... to view and select a Scenario for which you want a report. Click **OK** to return to the wizard, then click **Next**.
4. Select the report you wish to run, then click **Next**.

   See below for report types and descriptions.
5. Each report will have one or more additional options - use the default settings for now and click the **Finish** button to generate the report.

   Additional options vary depending on the report type.

Available WebApps RPT Reports

- **WebApps Executive Report**: A high-level report showing summary data of the RPT results.
- **WebApps Vulnerability Report**: A detailed report showing all vulnerabilities that were found by the RPT.
- **WebApps Delta Report**: Provides a comparison between two workspaces.