metasploit® pro

Vulnerability Validation

Tutorial

Vulnerability scanning plays a key role in the vulnerability management process. It helps you find potential vulnerabilities so that you can quantify and evaluate the threats that can adversely impact a network. While vulnerability scans can give you visibility into a network, they can yield a copious amount of data that require an extensive amount of time and resources to assess and validate. With so many threats facing an organization, it can be difficult to prioritize and remediate security risks.

Metasploit Pro simplifies and expedites this process. It provides a unified, guided interface, called the Vulnerability Validation Wizard, that walks you through each step of the vulnerability validation process—from importing Nexpose data to auto-exploiting vulnerabilities to sending the validation results back to Nexpose. You can even define exceptions for vulnerabilities that were not successfully exploited and generate a report that details the vulnerability testing results directly from Metasploit Pro.

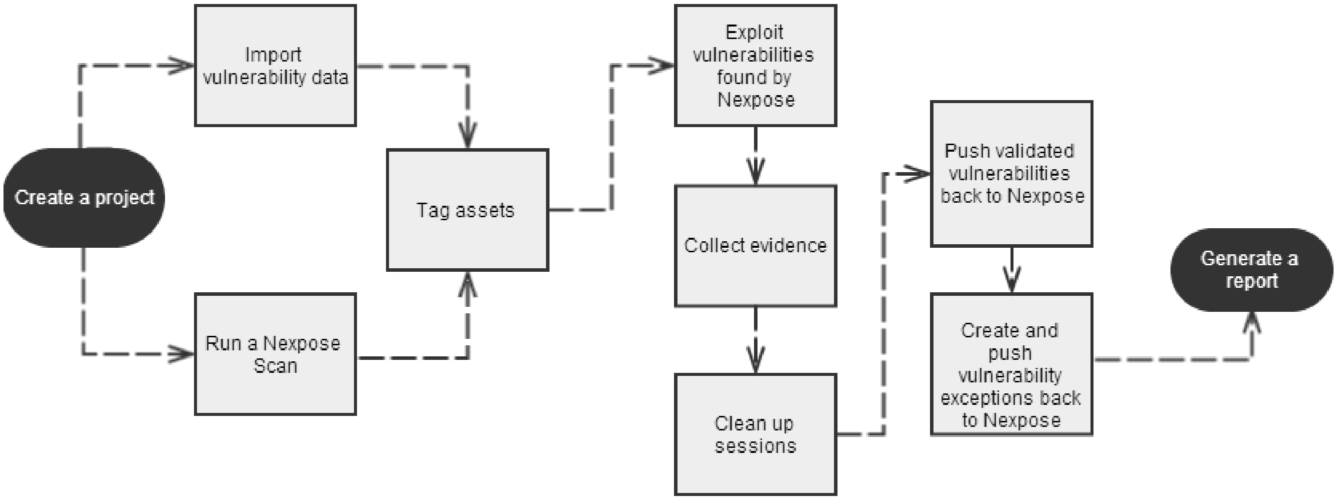
Nexpose and Metasploit Pro seamlessly integrate to streamline the vulnerability validation workflow. It creates a closed-loop security risk assessment solution so that you can find potential vulnerabilities, exploit them, and identify the security flaws that pose a real threat to a network.

Tutorial Objectives

This tutorial will guide you through the Vulnerability Validation Wizard and cover the following tasks:

- ❑ Creating a project to store Nexpose data and test results.
- ❑ Importing sites and vulnerability data from Nexpose.
- ❑ Tagging hosts.
- ❑ Exploiting vulnerabilities that have matching exploits.
- ❑ Pushing validated vulnerabilities back to Nexpose.
- ❑ Creating exceptions for vulnerabilities that Metasploit Pro was unable to exploit.
- ❑ Generating a report that details the vulnerabilities found.

Vulnerability Validation Wizard Workflow


Create a project
Import vulnerability data
Run a Nexpose Scan
Tag assets
Exploit vulnerabilities found by Nexpose
Collect evidence
Clean up sessions
Push validated vulnerabilities back to Nexpose
Create and push vulnerability exceptions back to Nexpose
Generate a report

Vulnerability Validation Terminology

Asset	The Nexpose term for a host or target.
Nexpose Push	The process that sends Metasploit Pro vulnerability validation results to Nexpose.
Site	The Nexpose term for a collection of assets.
Validated Vulnerability	An vulnerability found by Nexpose that Metasploit Pro was able to successfully exploit and obtain a session.
Vulnerability	A security flaw or weakness in an application or system that enables an attacker to compromise the target system.
Vulnerability Exception	A vulnerability found by Nexpose that Metasploit Pro was unable to exploit.
Vulnerability Exception Reason	The reason why a vulnerability exists and why it should be excluded from the vulnerability assessment.
Vulnerability Result Code	The reason why a module did not run successfully.

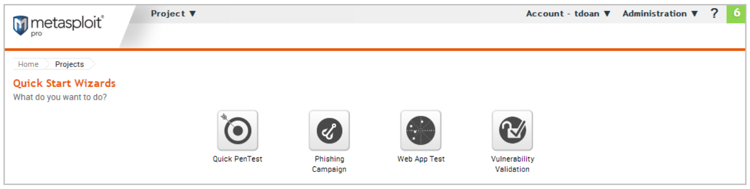
Before You Begin

Configure a Nexpose Console	You can configure a Nexpose console directly from the Vulnerability Validation Wizard. However, to simplify the workflow, it is recommended that you globally add the Nexpose Consoles you intend to use prior to launching the wizard. Note: Only Nexpose Enterprise and Nexpose Consultant can be used to perform vulnerability validation with Metasploit Pro. To configure a Nexpose Console, go to **Administration > Global Settings > Nexpose Consoles**. You must provide the following information: • **Console Address** - The IP address to the server that runs Nexpose. You can also specify the server name. • **Console Port** - The port that runs the Nexpose service. The default port is 3780. • **Console Username** - The Nexpose user name that will be used to log in to the console. • **Console Password** - The Nexpose password that will be used to authenticate the user account.
Clear your browser’s cache	After you install or update Metasploit, you must clear your browser’s cache to ensure that all user interface elements load properly. To learn how to clear the browser cache, visit the documentation for your web browser.
Set up a site in Nexpose	Before you can import Nexpose data, you must have at least one site set up in Nexpose. To learn how to set up a site, please view the *Nexpose Installation and Quick Start Guide*.

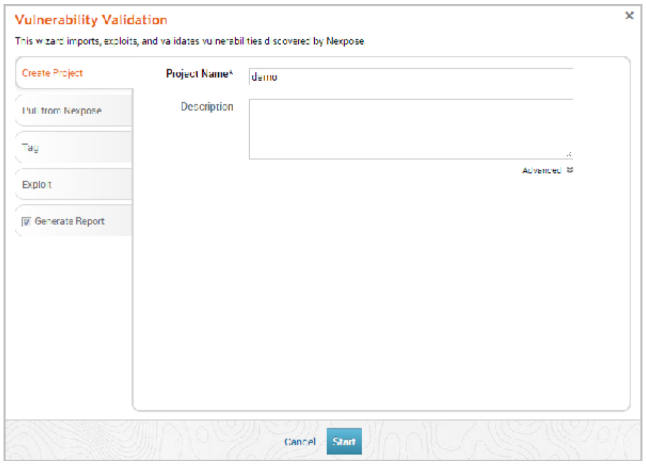
Step 1: Importing and Exploiting Nexpose Vulnerabilities

1. Log in to the Metasploit Pro web interface (https://localhost:3790).

2. When the Projects page appears, find the Quick Start Wizards area and click on the **Vulnerability Validation Wizard**. The Vulnerability Validation Wizard opens and displays the **Create Project** page.



3. In the **Project Name** field, enter a name for the project. The project name can contain any combination of alphanumeric characters, special characters, and spaces.



Vulnerability Validation

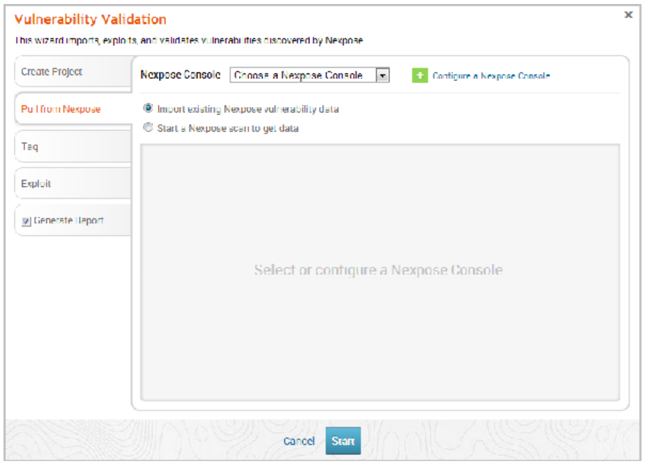
This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose

Project Name* demo

You can also provide a description for the project, which typically explains the purpose and scope of the test. This field is optional.

What is a project? A project contains the workspace for a penetration test and stores the data that you have collected for each host.

4. Click on the **Pull from Nexpose** tab. The Nexpose Consoles page appears.



Vulnerability Validation

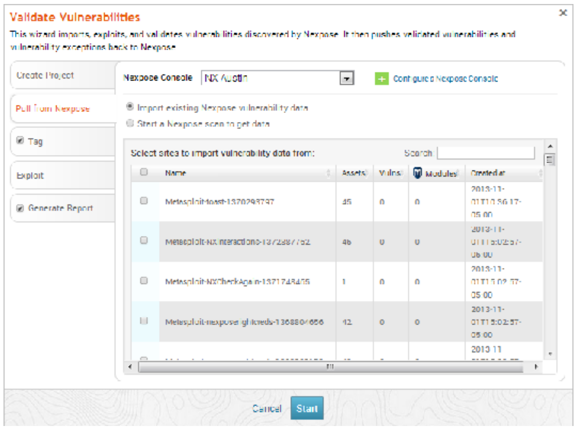
This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose

Nexpose Console: Choose a Nexpose Console

● Import existing Nexpose vulnerability data
○ Start a Nexpose scan to get data

Select or configure a Nexpose Console

5. Click the **Choose a Nexpose Console** dropdown and select the Nexpose Console from which you want to import sites. After you select a console, the wizard displays the list of sites that you can import.



Validate Vulnerabilities

This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose. It then pushes validated vulnerabilities and vulnerability exceptions back to Nexpose.

Nexpose Console: NX Austin

● Import existing Nexpose vulnerability data
○ Start a Nexpose scan to get data

Select sites to import vulnerability data from: Search:

□	Name	Assets	Vulns	Modules	Created at
□	Metasploit-toast-1370293797	45	0	0	2013-11-01T10:56:17-05:00
□	Metasploit-NXInteractions-1372387752	46	0	0	2013-11-01T15:02:57-05:00
□	Metasploit-NXCheckAgain-1371743435	1	0	0	2013-11-01T15:02:57-05:00
□	Metasploit-nexposelightcreds-1368804656	12	0	0	2013-11-01T15:02:57-05:00
□					2013-11

How do I exclude assets from the import?

Metasploit Pro will import all assets from a site unless you explicitly define the assets that you want to exclude. To exclude assets from the import, click the **Excluded Addresses** dropdown and enter the addresses of those assets in the **Excluded Addresses** field.

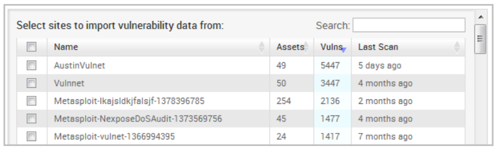
What if I want to run a Nexpose scan instead of importing from an existing site?

Select the **Start a Nexpose Scan to get data** option. When the **Nexpose Scan** page appears, you need to specify the scan template that you want to use and the target that you want to scan. For more information about scan templates, visit the *Nexpose User's Guide*.

When you run a Nexpose scan, a new site is created and listed under the Site Listing in the Nexpose Console. The site name will use the following naming convention: `Metasploit-[project name]-[timestamp]`.

After the scan completes, the Vulnerability Validation Wizard displays the sites under the sites list.

6. From the sites list, select the sites that you want to import into the project. You can use the **Select All** checkbox to choose all of the listed sites, or you can select the sites individually.



Select sites to import vulnerability data from: Search:

□	Name	Assets	Vulns	Last Scan
□	AustinVulnet	49	5447	5 days ago
□	Vulnnet	50	3447	4 months ago
□	Metasploit-lkajsldkjfalsjf-1378396785	254	2136	2 months ago
□	Metasploit-NexposeDoSAudit-1373569756	45	1477	4 months ago
□	Metasploit-vulnet-1366994395	24	1417	7 months ago

Which vulnerabilities are imported?

Metasploit Pro imports only imports vulnerabilities that have open ports.

What information does Metasploit Pro import from each site?

For each asset, Metasploit Pro pulls and displays the IP address, operating system, MAC address, OS flavor, vulnerability name, and vulnerability references.

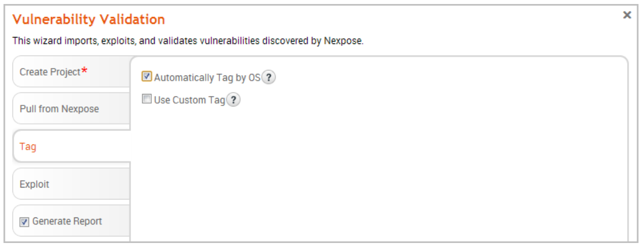
7. After you select the sites you want to import, click the **Tag** tab.

   What are tags? A tag is an identifier that you can apply to a host to help you to search for assets, create work queues, and track findings. You can tag assets using the OS name or using a custom tag that you have specified.

   Tags can also be used to create asset groups in Nexpose.

   Do I need to tag hosts? No, tagging is optional. If you do not want to tag assets, go to Step 10.

8. To tag hosts by their operating system, select the **Automatically tag by OS** option. Metasploit Pro tags Window hosts with `os_windows` and Linux host with `os_linux`.



Vulnerability Validation

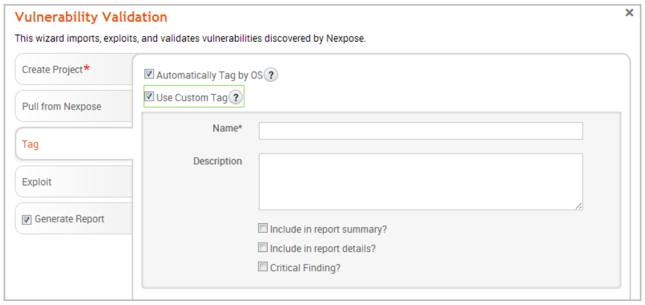
This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose.

Create Project*
Pull from Nexpose
Tag
Exploit
☑ Generate Report

☑ Automatically Tag by OS ?
□ Use Custom Tag ?

9. To apply a custom tag, do the following:

   a. Select the **Use Custom Tag** option.



Vulnerability Validation

This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose.

Create Project*
Pull from Nexpose
Tag
Exploit
☑ Generate Report

☑ Automatically Tag by OS ?
☑ Use Custom Tag ?

Name*
Description
□ Include in report summary?
□ Include in report details?
□ Critical Finding?

   b. Enter a name for the custom tag in the **Name** field.

   c. Enter a description for the custom tag in the **Description** field.

   d. Select any of the following attributes:

   - **Include in report summary** - Provides information about the host in the Executive Summary section of the report.
   - **Include in report details** - Provides information about the host in the Detailed Findings section of the report.
   - **Critical Finding** - Marks the host information as critical.

10. After you configure the tagging options, click on the **Exploit** tab. The Auto-Exploitation page appears.

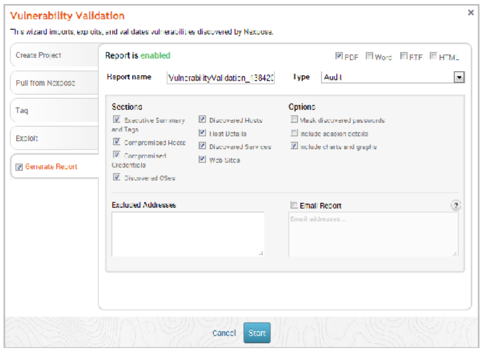
What is auto-exploitation? It's the process that cross-references open services, vulnerability references, and fingerprint data to matching exploits to create an attack plan. Auto-exploitation uses an option called minimum reliability to guarantee the safety of the exploits that Metasploit launches against the target assets. The higher the reliability level, the less likely the exploits used will crash services or negatively impact a target. It is recommended that you use a minimum reliability of **Great** or **Excellent**.

Which vulnerabilities are exploited? Metasploit only exploits vulnerabilities that have a matching remote exploit module that has a ranking of Great or Excellent.

11. Use any of the following options to configure exploitation settings:

Dry Run	Prints a transcript of the exploits in the attack plan without running them.
Collect Evidence	Collects loot, such as screenshots, system files, passwords, and configuration settings from open sessions.
Clean Up Sessions	Closes all sessions after all tasks have run.
Payload Type	Specifies the type of payload that the exploit will deliver to the target. Choose one of the following payload types: • **Command** - A command execution payload that enables you to execute commands on the remote machine. • **Meterpreter** - An advanced payload that provides a command line that enables you to deliver commands and inject extensions on the fly.
Connection Type	Specifies how you want your Metasploit instance to connect to the target. Choose one of the following connection types: • **Auto** - Automatically uses a bind connection when NAT is detected; otherwise, a reverse connection is used. • **Bind** - Uses a bind connection, which is useful when the targets are behind a firewall or a NAT gateway. • **Reverse** - Uses a reverse connection, which is useful if your system is unable to initiate connections to the targets.
Listener Ports	Defines the ports that you want to use for reverse connections.
Listener Host	Defines the IP address you want to connect back to.
Auto Launch Macro	Specifies the macro that you want to run during post-exploitation.
Concurrent Exploits	Specifies the number of exploit attempts you want to launch at one time.
Timeout in Minutes	Defines the number of minutes an exploit waits before it times out.
Transport Evasion	Choose from the following transport evasion levels: • **Low** - Inserts delays between TCP packets. • **Medium** - Sends small TCP packets. • **High** - Sends small TCP packets and inserts delays between them.
Application Evasion	Adjusts application-specific evasion options for exploits involving DCERPC, SMB and HTTP. The higher the application evasion level, the more evasion techniques are applied.
Included Ports	Defines the specific ports you want to target for exploitation.
Excluded Ports	Defines the specific ports you want to exclude from exploitation.

12. Click the **Generate report** tab. Generating a report is optional. If you do not want to generate a report, deselect the **Report** option located directly on the Report tab and go to the last step.



Vulnerability Validation

This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose.

Report is enabled ☑ PDF ☐ Word ☐ RTF ☐ HTML

Report name: VulnerabilityValidation_138420 Type: Audit

Sections
- ☑ Executive Summary and Tags
- ☑ Compromised Hosts
- ☑ Compromised Credentials
- ☑ Discovered OSes
- ☑ Discovered Hosts
- ☑ Host Details
- ☑ Discovered Services
- ☑ Web Sites

Options
- ☐ Mask discovered passwords
- ☐ Include session details
- ☑ Include charts and graphs

Excluded Addresses

☐ Email Report

Email addresses...

What reports can I generate from the Vulnerability Validation Wizard? You can generate either an Audit report or a Compromised and Vulnerable Hosts report.

13. To automatically generate a report after the test completes , do the following:

    a. Enter a name for the report in the **Report Name** field, if you want to use a custom report name. Otherwise, the wizard uses the default report name.

    b. Select whether you want to generate the report in PDF, RTF, or HTML. PDF is the preferred format.

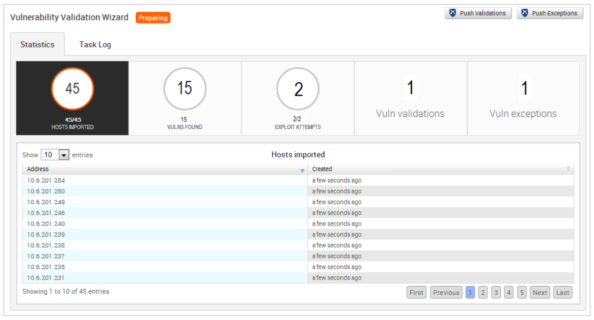
    c. Click the **Type** dropdown and select the report type you want to generate. You can choose the Audit report or the Compromised and Vulnerable Hosts report.

    d. From the **Sections** area, deselect any sections you do not want to include in the report. Skip this step if you want to generate all the report sections.

    e. Select the **Email Report** option if you want to e-mail the report after it generates. If you enable this option, you need to supply a comma separated list of e-mail addresses.

    Note: If you want to e-mail a report, you must set up a local mail server or e-mail relay service for Metasploit Pro to use. To define your mail server settings, select **Administration > Global Settings > SMTP Settings**.

14. Click the **Launch** button. The Findings window appears.



Vulnerability Validation Wizard Preparing

45 — 45/45 HOSTS IMPORTED

15 — 15 VULNS FOUND

2 — 2/2 EXPLOIT ATTEMPTS

1 Vuln validations

1 Vuln exceptions

Show 10 entries

Hosts imported

Address	Created
10.6.201.254	a few seconds ago
10.6.201.250	a few seconds ago
10.6.201.249	a few seconds ago
10.6.201.246	a few seconds ago
10.6.201.240	a few seconds ago
10.6.201.239	a few seconds ago
10.6.201.238	a few seconds ago
10.6.201.237	a few seconds ago
10.6.201.235	a few seconds ago
10.6.201.231	a few seconds ago

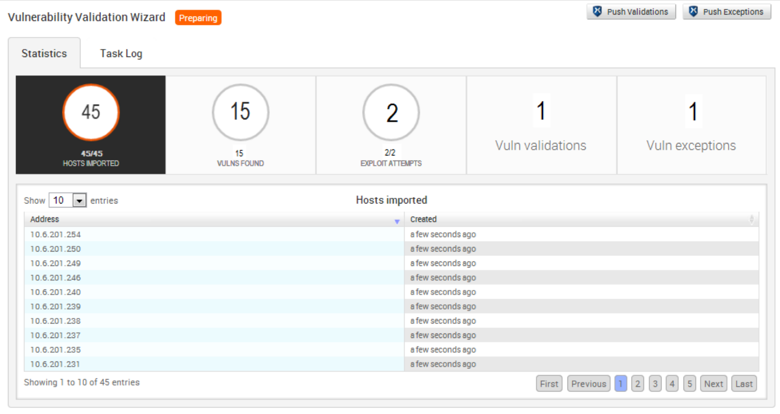
Showing 1 to 10 of 45 entries

Step 2: Tracking Real-Time Statistics and Events

The Findings window displays the real-time statistics for the test and the task log. There are tabs available for each view. You can click on each tab to change the current view.

The Statistics Tab

The **Statistics** tab shows a high-level, numerical overview of hosts, vulnerabilities, and exploits. Each value is displayed in a stat bubble with an orange progress bar. The progress bar wraps around the stat bubble and only displays when there is activity occurring for a particular finding.



Hosts imported

Address	Created
10.6.201.254	a few seconds ago
10.6.201.250	a few seconds ago
10.6.201.249	a few seconds ago
10.6.201.246	a few seconds ago
10.6.201.240	a few seconds ago
10.6.201.239	a few seconds ago
10.6.201.238	a few seconds ago
10.6.201.237	a few seconds ago
10.6.201.235	a few seconds ago
10.6.201.231	a few seconds ago

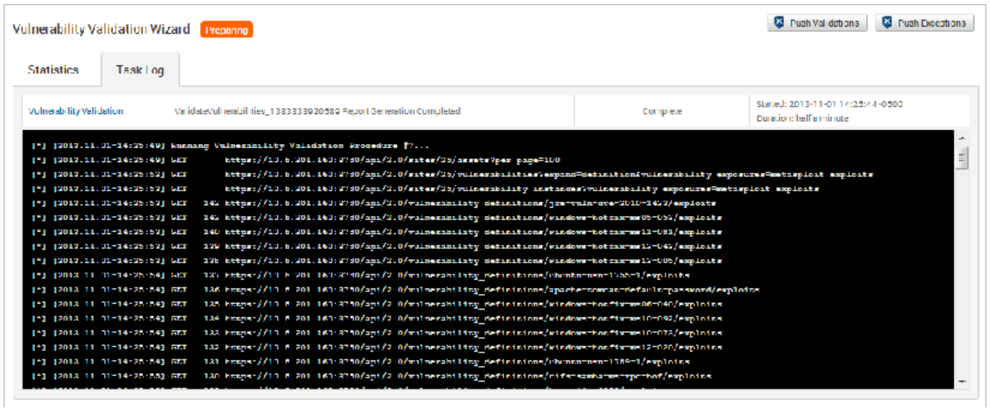
Showing 1 to 10 of 45 entries

From the **Statistics** tab, you can track the following data:

- The total number of hosts that have been scanned.
- The total number of unique vulnerabilities that have been found.
- The total number of exploits attempts that Metasploit Pro has made against a site; this number is the sum of the number of exceptions and validations.
- The total number of vulnerabilities that Metasploit Pro was able to validate.
- The total number of vulnerabilities that Metasploit Pro was unable to validate .

The Tasks Log Tab

The **Tasks Log** tab shows a detailed activity log for the Vulnerability Validation Wizard. Each task that Metasploit Pro performs is documented in the Tasks Log.



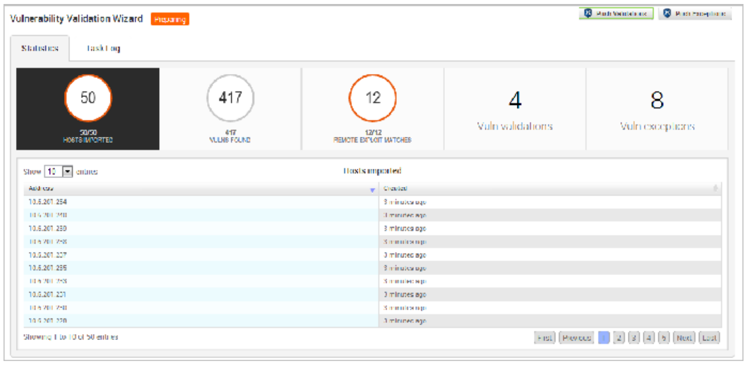
Vulnerability Validation | ValidateVulnerabilities_1383333920589 Report Generation Completed | Complete | Started: 2013-11-01 14:25:44 -0500 Duration: half a minute

```
[*] [2013.11.01-14:25:49] Running Vulnerability Validation Procedure [?]...
[*] [2013.11.01-14:25:49] GET       https://10.6.201.163:3780/api/2.0/sites/25/assets?per_page=100
[*] [2013.11.01-14:25:52] GET       https://10.6.201.163:3780/api/2.0/sites/25/vulnerabilities?expand=definition&vulnerability_exposures=metasploit_exploits
[*] [2013.11.01-14:25:52] GET       https://10.6.201.163:3780/api/2.0/sites/25/vulnerability_instances?vulnerability_exposures=metasploit_exploits
[*] [2013.11.01-14:25:52] GET   142 https://10.6.201.163:3780/api/2.0/vulnerability_definitions/jre-vuln-cve-2010-4452/exploits
[*] [2013.11.01-14:25:52] GET   141 https://10.6.201.163:3780/api/2.0/vulnerability_definitions/windows-hotfix-ms09-050/exploits
[*] [2013.11.01-14:25:52] GET   140 https://10.6.201.163:3780/api/2.0/vulnerability_definitions/windows-hotfix-ms11-083/exploits
[*] [2013.11.01-14:25:52] GET   139 https://10.6.201.163:3780/api/2.0/vulnerability_definitions/windows-hotfix-ms12-045/exploits
[*] [2013.11.01-14:25:52] GET   138 https://10.6.201.163:3780/api/2.0/vulnerability_definitions/windows-hotfix-ms13-005/exploits
[*] [2013.11.01-14:25:54] GET   137 https://10.6.201.163:3780/api/2.0/vulnerability_definitions/ubuntu-usn-1155-1/exploits
[*] [2013.11.01-14:25:54] GET   136 https://10.6.201.163:3780/api/2.0/vulnerability_definitions/apache-tomcat-default-password/exploits
[*] [2013.11.01-14:25:54] GET   135 https://10.6.201.163:3780/api/2.0/vulnerability_definitions/windows-hotfix-ms06-040/exploits
[*] [2013.11.01-14:25:54] GET   134 https://10.6.201.163:3780/api/2.0/vulnerability_definitions/windows-hotfix-ms10-092/exploits
[*] [2013.11.01-14:25:54] GET   133 https://10.6.201.163:3780/api/2.0/vulnerability_definitions/windows-hotfix-ms10-073/exploits
[*] [2013.11.01-14:25:54] GET   132 https://10.6.201.163:3780/api/2.0/vulnerability_definitions/windows-hotfix-ms12-020/exploits
[*] [2013.11.01-14:25:54] GET   131 https://10.6.201.163:3780/api/2.0/vulnerability_definitions/ubuntu-usn-1189-1/exploits
[*] [2013.11.01-14:25:55] GET   130 https://10.6.201.163:3780/api/2.0/vulnerability_definitions/cifs-samba-ms-rpc-bof/exploits
```

Where do I go to create exceptions and push validated vulnerabilities? In addition to tracking the latest test data, there are two key actions you can perform from the Findings Window: you can push validated vulnerabilities back to Nexpose, and you can create exceptions for vulnerabilities that Metasploit Pro was unable to exploit.

Step 3: Pushing Validated Vulnerabilities Back to Nexpose

When validated vulnerabilities are ready to be sent to Nexpose, the **Push Validated Vulnerabilities** button becomes available on the Findings window. If Metasploit was unable to validate any vulnerabilities, the button will not be available.



Show 10 entries

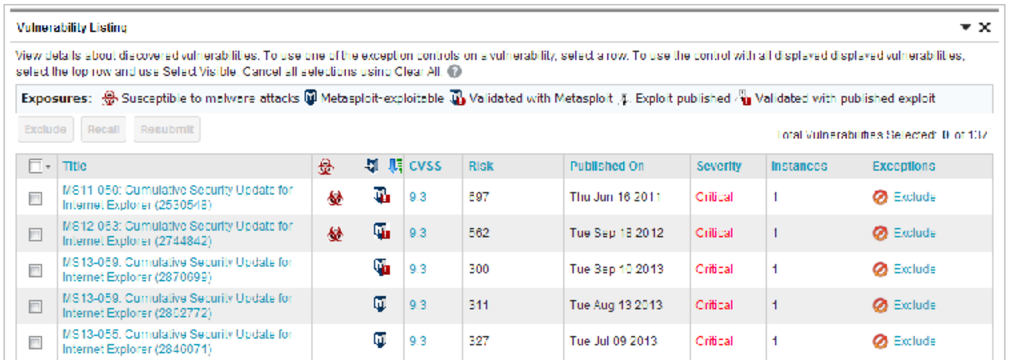
Hosts imported

Address	Created
10.6.201.254	3 minutes ago
[illegible]	3 minutes ago
10.6.201.250	3 minutes ago
[illegible]	3 minutes ago
10.6.201.237	3 minutes ago
10.6.201.235	3 minutes ago
[illegible]	3 minutes ago
10.6.201.231	3 minutes ago
[illegible]	3 minutes ago
10.6.201.220	3 minutes ago

Showing 1 to 10 of 50 entries

To send the validated vulnerabilities to Nexpose, you just need to click the **Push Validated Vulnerabilities** button. All vulnerabilities that have been exploited will be pushed to Nexpose.

In the Nexpose Console, any vulnerability validated by Metasploit Pro will have the **Validated with Metasploit** icon located in the **Exploits** column of the Vulnerability Listing.



View details about discovered vulnerabilities. To use one of the exception controls on a vulnerability, select a row. To use the control with all displayed displayed vulnerabilities, select the top row and use Select Visible. Cancel all selections using Clear All.

Exposures: Susceptible to malware attacks Metasploit-exploitable Validated with Metasploit Exploit published Validated with published exploit

Total Vulnerabilities Selected: 0 of 137

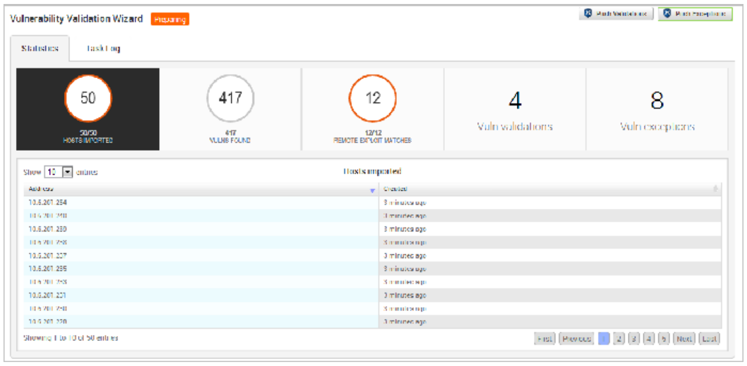
☐	Title			CVSS	Risk	Published On	Severity	Instances	Exceptions
☐	MS11 050: Cumulative Security Update for Internet Explorer (2530548)	☣		9.3	597	Thu Jun 16 2011	Critical	1	Exclude
☐	MS12 063: Cumulative Security Update for Internet Explorer (2744842)	☣		9.3	562	Tue Sep 18 2012	Critical	1	Exclude
☐	MS13-069. Cumulative Security Update for Internet Explorer (2870699)			9.3	300	Tue Sep 10 2013	Critical	1	Exclude
☐	MS13-059. Cumulative Security Update for Internet Explorer (2862772)			9.3	311	Tue Aug 13 2013	Critical	1	Exclude
☐	MS13-055. Cumulative Security Update for Internet Explorer (2846071)			9.3	327	Tue Jul 09 2013	Critical	1	Exclude

To view a list of validated vulnerabilities in the Nexpose Console:

1. Open a site and find the Asset Listing.
2. Click on the asset name or asset address to open the Asset Properties. The Asset Properties will display the Vulnerability Listing.
3. Click on the **Exploited** column to sort by validated vulnerabilities.

Step 4: Creating Exceptions for Non-Exploitable Vulnerabilities

1. Click the **Create Exceptions** button located on the Findings window. The Create Nexpose Exceptions page appears.



Show 10 entries

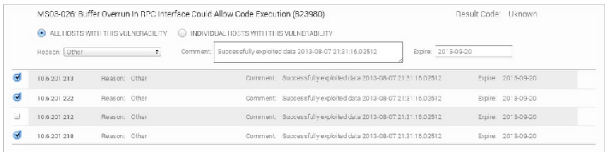
Hosts imported

Address	Created
10.6.201.254	3 minutes ago
10.6.201.240	3 minutes ago
10.6.201.230	3 minutes ago
10.6.201.238	3 minutes ago
10.6.201.227	3 minutes ago
10.6.201.235	3 minutes ago
10.6.201.253	3 minutes ago
10.6.201.231	3 minutes ago
10.6.201.240	3 minutes ago
10.6.201.220	3 minutes ago

Showing 1 to 10 of 50 entries

Why is the Create Exceptions button unavailable? The **Create Exceptions** button becomes available on the Findings window when there are vulnerabilities ready for you to create exceptions for.

2. Select the hosts that you want to create exceptions for. Use the **Select All Hosts** checkbox if you want to create exceptions for all hosts that have a non-exploitable vulnerability.



MS03-026: Buffer Overrun In RPC Interface Could Allow Code Execution (823980) Result Code: Uknown

● ALL HOSTS WITH THIS VULNERABILITY ○ INDIVIDUAL HOSTS WITH THIS VULNERABILITY

Reason: Other Comment: Successfully exploited data 2013-08-07 21:31:16.02512 Expire: 2013-09-20

☑ 10.6.201.213	Reason: Other	Comment: Successfully exploited data 2013-08-07 21:31:16.02512	Expire: 2013-09-20
☑ 10.6.201.222	Reason: Other	Comment: Successfully exploited data 2013-08-07 21:31:16.02512	Expire: 2013-09-20
□ 10.6.201.212	Reason: Other	Comment: Successfully exploited data 2013-08-07 21:31:16.02512	Expire: 2013-09-20
☑ 10.6.201.218	Reason: Other	Comment: Successfully exploited data 2013-08-07 21:31:16.02512	Expire: 2013-09-20

Can I assign vulnerability exceptions per host instead of per vulnerability? Yes. Select the **Individual Hosts with this Vulnerability** option. The Vulnerability Exceptions page will display a **Reason** dropdown for each host.

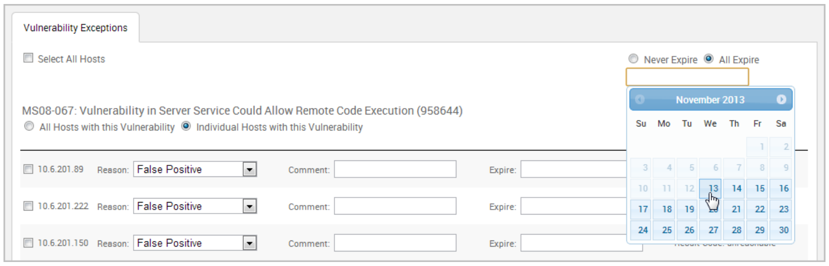
3. For each vulnerability, click the **Reason** dropdown and choose the vulnerability exception reason you want to assign to it. You can also provide additional information for the exception in the **Comment** field.

What are the vulnerability exception reasons I can choose from?

- **False positive** - Use this exception reason for a vulnerability that does not exist.
- **Compensating control** - Use this exception reason to indicate that a vulnerability is a compensating control, or a workaround for a security requirement.
- **Acceptable use** - Use this exception reason for any vulnerability that is used as part of organizational practices.
- **Acceptable risk** - Use this exception reason for any vulnerability that are considered low risk. These vulnerabilities tend to pose minimal security risk and are likely to consume more resources than they are worth.
- **Other** - Use this exception reason to define a custom exception. If you select **Other**, you can provide a custom exception reason in the **Comment** field.

4. Choose the **All Expire** option if you want to set an expiration date for all the vulnerability exceptions. If you do not want to set an expiration date for any vulnerability exceptions, keep the default **Never Expire** option selected and go to Step 6.

   To set the same expiration date for all vulnerability exceptions, select on the **All Expire** option. A calendar appears. Find and select the date that you want to use.



Vulnerability Exceptions

☐ Select All Hosts

○ Never Expire ● All Expire

MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644)

○ All Hosts with this Vulnerability ● Individual Hosts with this Vulnerability

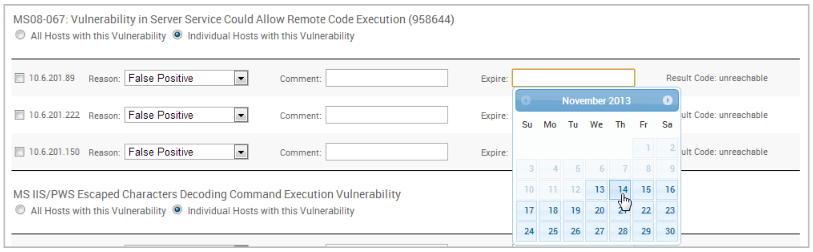
☐ 10.6.201.89 Reason: False Positive Comment: Expire:

☐ 10.6.201.222 Reason: False Positive Comment: Expire:

☐ 10.6.201.150 Reason: False Positive Comment: Expire:

   If you want to set a unique expiration date for each host, skip this step and go to step 5.

5. To set a unique expiration date for each host, select the **Individual hosts with this vulnerability** option. Click on the **Expire** field next to each exception to display the calendar. Find the expiration date that you want to use and select it.



MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644)

○ All Hosts with this Vulnerability ● Individual Hosts with this Vulnerability

☐ 10.6.201.89 Reason: False Positive Comment: Expire: Result Code: unreachable

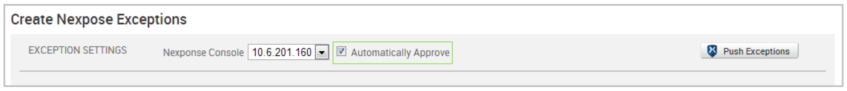
☐ 10.6.201.222 Reason: False Positive Comment: Expire: ult Code: unreachable

☐ 10.6.201.150 Reason: False Positive Comment: Expire: ult Code: unreachable

MS IIS/PWS Escaped Characters Decoding Command Execution Vulnerability

○ All Hosts with this Vulnerability ● Individual Hosts with this Vulnerability

6. Verify that you want to approve all vulnerability exception requests from Metasploit Pro. If the **Automatically Approve** option is selected, Nexpose will automatically approve vulnerability exception requests imported from Metasploit Pro.



Create Nexpose Exceptions

EXCEPTION SETTINGS Nexponse Console 10.6.201.160 ☑ Automatically Approve

What if I want to manually submit exception requests through the Nexpose Console? You can deselect the **Automatically Approve** option. Metasploit will still push the vulnerability exceptions back to Nexpose, which will allow you to manage exception requests directly from the Nexpose Console.

How do I manually submit an exception request in Nexpose? To manually submit exception requests, you will need to go to the Vulnerability Exception Listing (**Administration > Exceptions and Overrides > Manage**) and look at the **Exception** column. If an exception request has not been previously submitted for that vulnerability, the **Exception** column displays an **Exclude** icon. You will need to click on the icon and follow the instructions in the *Nexpose User's Guide* for submitting an exception request.

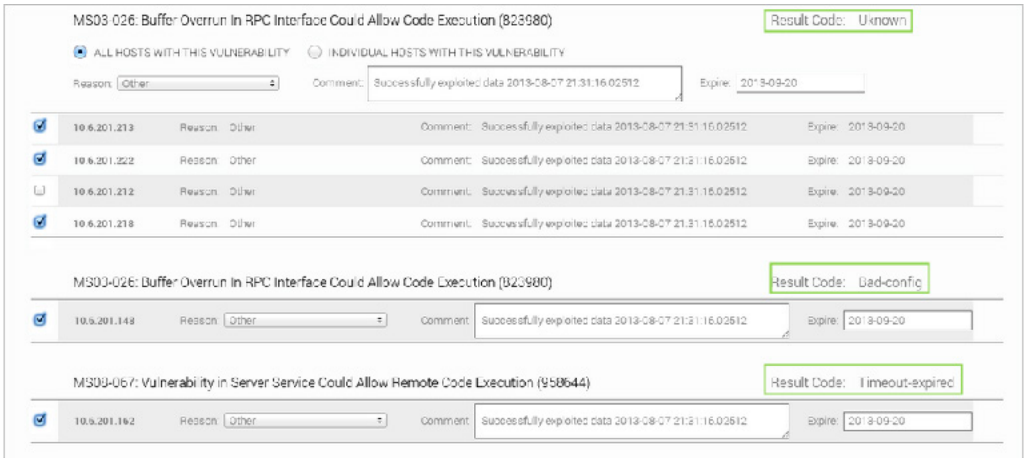
7. When you are ready to push the exceptions, click the **Push Exceptions** button.

   How can I view the vulnerability exceptions in Nexpose? From within the Nexpose Console, you can go to the Vulnerability Exception Listing (**Administration** > **Exceptions and Overrides** > **Manage**) to view the vulnerability name, exception reason, the person who reported the vulnerability exception, and the person who approved the vulnerability exception.

Frequently Asked Questions

What are result codes?

A result code is the reason an exploit failed. You can view the result code for a vulnerability on the Vulnerability Exceptions page.



MS03-026: Buffer Overrun In RPC Interface Could Allow Code Execution (823980) — Result Code: Uknown

● ALL HOSTS WITH THIS VULNERABILITY ○ INDIVIDUAL HOSTS WITH THIS VULNERABILITY

Reason: Other Comment: Successfully exploited data 2013-08-07 21:31:16.02512 Expire: 2013-09-20

☑ 10.6.201.213	Reason: Other	Comment: Successfully exploited data 2013-08-07 21:31:16.02512	Expire: 2013-09-20
☑ 10.6.201.222	Reason: Other	Comment: Successfully exploited data 2013-08-07 21:31:16.02512	Expire: 2013-09-20
☐ 10.6.201.212	Reason: Other	Comment: Successfully exploited data 2013-08-07 21:31:16.02512	Expire: 2013-09-20
☑ 10.6.201.218	Reason: Other	Comment: Successfully exploited data 2013-08-07 21:31:16.02512	Expire: 2013-09-20

MS03-026: Buffer Overrun In RPC Interface Could Allow Code Execution (823980) — Result Code: Bad-config

☑ 10.6.201.148	Reason: Other	Comment: Successfully exploited data 2013-08-07 21:31:16.02512	Expire: 2013-09-20

MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644) — Result Code: Timeout-expired

☑ 10.6.201.162	Reason: Other	Comment: Successfully exploited data 2013-08-07 21:31:16.02512	Expire: 2013-09-20

The following result codes are available:

- **None** - Indicates that Metasploit Pro could not determine if the module ran successfully or failed.
- **Unknown** - Indicates that Metasploit Pro could not determine if the module ran successfully or failed.
- **Unreachable** - Indicates that Metasploit Pro could not reach the network service.
- **Bad-config** - Indicates that the exploit settings were configured incorrectly.
- **Disconnected** - Indicates that the network service disconnected during a module run.
- **Not-found** - Indicates that Metasploit Pro could not find the application or service.
- **Unexpected-reply** - Indicates that Metasploit Pro did not receive the expected response from the application.
- **Timeout-expired** - Indicates that a timeout occurred.
- **User-interrupt** - Indicates that the user stopped the module run.
- **No-access** - Indicates that Metasploit Pro could not access the application.
- **No-target** - Indicates that the module configuration was not compatible with the target.
- **Not-vulnerable** - Indicates that the application was not vulnerable.
- **Payload-failed** - Indicates that Metasploit Pro delivered a payload, but was unable to open a session.

Can vulnerabilities only be validated through the Vulnerability Validation Wizard?

No. You can manually perform each step of the vulnerability validation process.

Here is the general workflow for manually validating vulnerabilities:

1. Create a project or open an existing project.
2. Run a Nexpose scan or import Nexpose scan data.
3. Run a Discovery scan to enumerate additional ports and services and to find additional vulnerabilities.
4. Run auto-exploitation.
5. Go to **Analysis > Hosts > Vulnerabilities** after auto-exploitation completes.
6. Click the **Push Exploited Vulnerabilities** button to send all validated vulnerabilities back to Nexpose.
7. Click the **Create Exceptions** dropdown and choose whether you want to create exceptions for all non-exploitable vulnerabilities or specific non-exploitable vulnerabilities. You will need to define the exception reason codes for each vulnerability.
8. Push the vulnerability exceptions back to Nexpose.
9. Clean up any open sessions.
10. Generate the Compromised and Vulnerable Hosts report.

How do I view a list of exploits that Metasploit Pro will run without actually running the exploits?

You can perform a dry run of the exploitation phase. A dry run matches vulnerabilities to exploits and generates an attack plan that lists the exploits that Metasploit Pro will launch during the actual test.

When you perform a dry run through the Vulnerability Validation Wizard, Metasploit Pro creates the project and imports assets and vulnerabilities from Nexpose. It then generates an attack plan based on the vulnerability information it has and the minimum reliability that is selected. It does not launch the exploits. However, the project will be created and the assets will be viewable from the project. You will be able to view the attack plan through the task log.

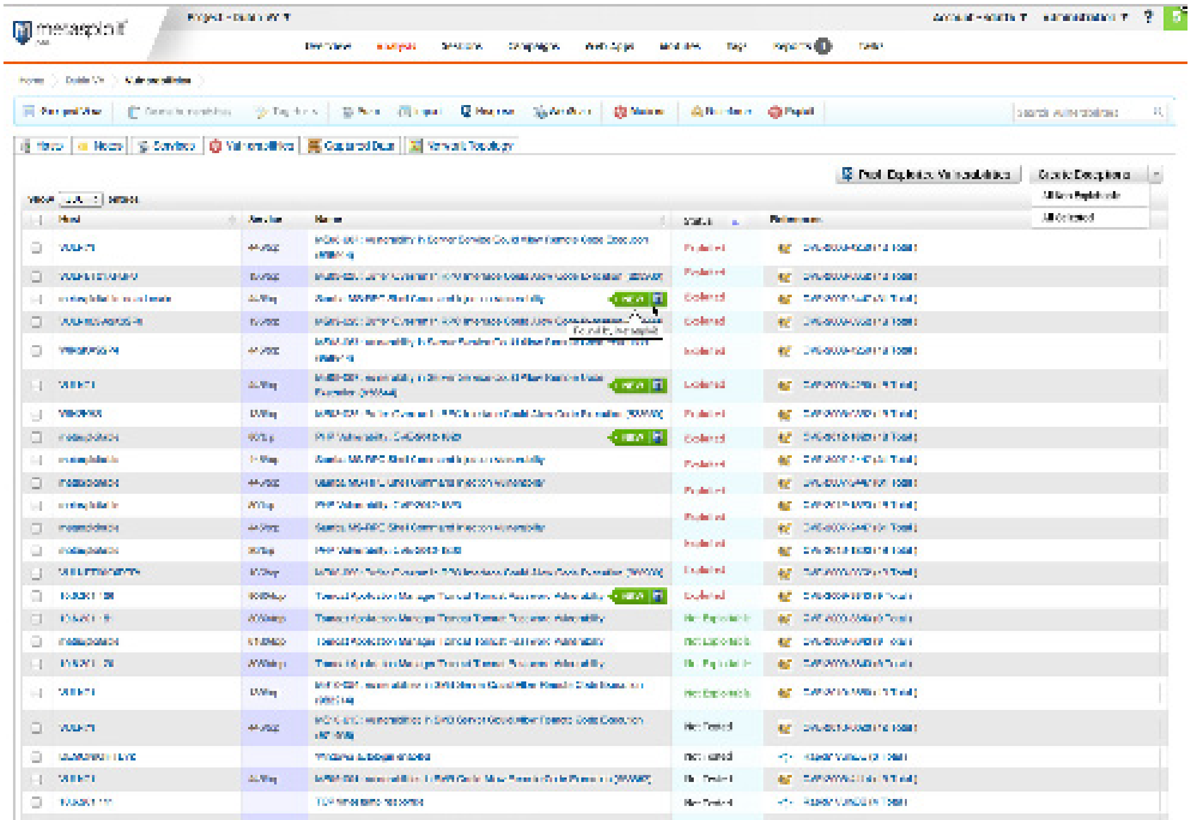
The following steps provide an overview of how you can perform a dry run:

1. Launch the Vulnerability Validation Wizard.
2. Fill out the information on the Create Project and Pull from Nexpose tabs.
3. Click on the Exploit tab.
4. Select the Dry Run option from the Payload area.
5. Select the Minimum Reliability for the exploits you want to use. You should select Great or Excellent.
6. Start the Vulnerability Validation test.

When the Findings window appears, click on the Task Log tab. The task log displays the attack plan for the test.

Where can I view the exploit status for a vulnerability?

When you perform vulnerability validation, Metasploit Pro tracks whether or not a vulnerability was successfully exploited. You can view the results from the Vulnerabilities tab (go to **Analysis > Hosts > Vulnerabilities**). Each vulnerability will have a status of exploited, not exploited, or no exploits available.



What are the exploit statuses?

A vulnerability can have one of the following exploit statuses:

- **Exploited** - Metasploit Pro was able to exploit the vulnerability to obtain a session on the target.
- **Not Exploitable** - Metasploit Pro was unable to exploit the vulnerability.
- **Not Tested** - Metasploit Pro was unable to find a matching exploit for the vulnerability.

Which versions of Nexpose do I need to perform vulnerability validation?

Nexpose Consultant and Nexpose Enterprise 5.7.16+ can be used with Metasploit Pro for vulnerability validation.

Why did Metasploit Pro not import all vulnerabilities from my Nexpose site?

Metasploit Pro only imports vulnerabilities for which it has correlating exploit modules. If Metasploit Pro does not have a matching exploit in its database, it will not import the vulnerability from a site.

I was able to find additional vulnerabilities with Metasploit Pro. Can I push the results for those vulnerabilities back to Nexpose?

Yes, you will be only able to send the vulnerability back to Nexpose if the Nexpose vulnerability database has a matching definition for the vulnerability that Metasploit Pro found.

After I push validated vulnerabilities back to Nexpose, how can I search for them in Nexpose?

There are a few ways you can view Metasploit-validated vulnerabilities in Nexpose:

- View the Vulnerability Listing for each asset.
- Perform a filtered asset search for validated vulnerabilities.
- Create a dynamic asset group for validated vulnerabilities.
- Create a dynamic asset group for validated vulnerabilities and generate a report whose scope is limited to that asset group.

For more information on any of these viewing methods, please view the *Nexpose User's Guide*.