Port Knocking

Description:

- We are attempting to SSH into a Linux VM with port 22 closed, which will open upon a correct sequence. Evidence will include Wireshark, syslogs from Ubuntu, and a recording of a successful SSH session.

Overview:

- We used a Kali VM and an Ubuntu VM.

Objectives:

- Download and import Ubuntu VM
- Install and configure Wireshark on Ubuntu VM
- Secure SSH on Linux with prespecified closed ports.
- Formulate sequence to open ports.
- Show port knocking implementation logs.
- Test result on wireshark.

Import Ubuntu Virtual Machine

Download Ubuntu using this link:
https://www.linuxvmimages.com/images/ubuntu-1804/#ubuntu-18046

1. Extract the zip file you downloaded to your desktop
2. Open VMWare Workstation
3. Click on Open a Virtual Machine
4. Click on the blue 3D box with the green down arrow to import
5. Log in using ubuntu, ubuntu as the username and password.
6. CHANGE the password (start the terminal and use the passwd command). It is a security issue for there to be a well known username password on running systems.
7. If you have a portable drive you may want to export the VM, or save snapshots to your portable drive.
8. Do the same steps with a Kali VM if you don’t already have one installed.

Launch Terminal

> *sudo dpkg-reconfigure -plow unattended-upgrades*

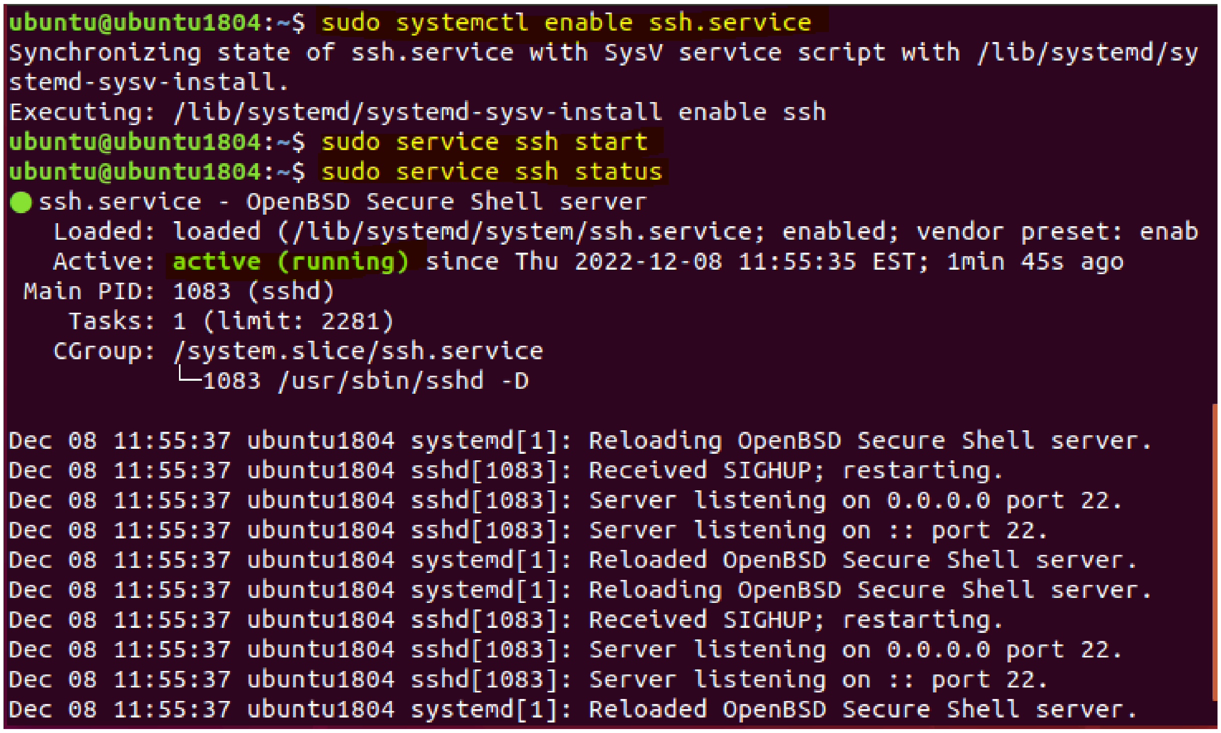
Restart Ubuntu

> *sudo apt-get upgrade*

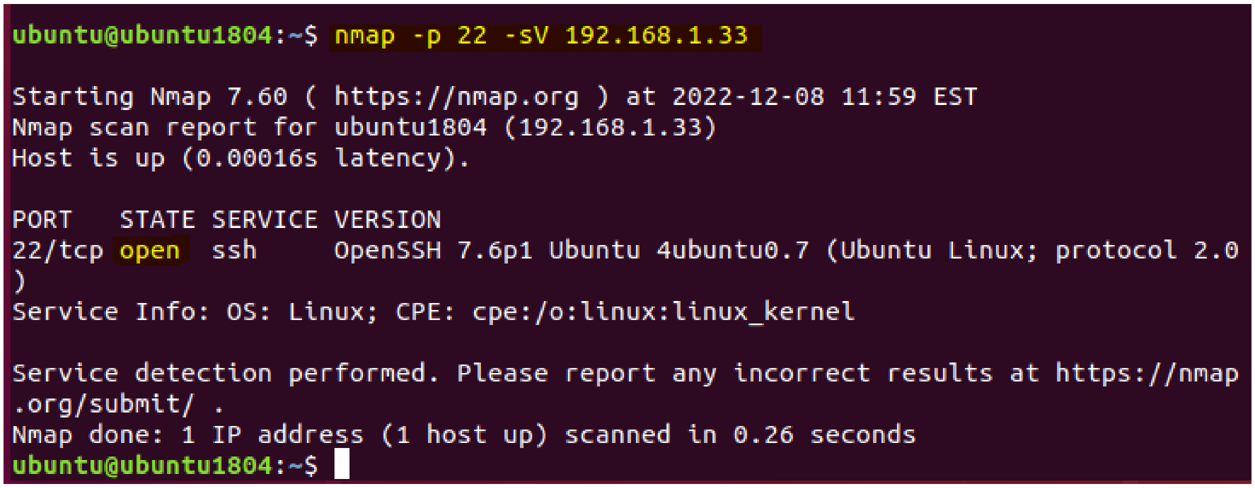
First we have to update the system and apt install and then run the program by using the ‘service’ command.

> *Sudo apt update && dist-upgrade -y*
> *Sudo apt install ssh*

- Once we update our system, install SSH and enable.

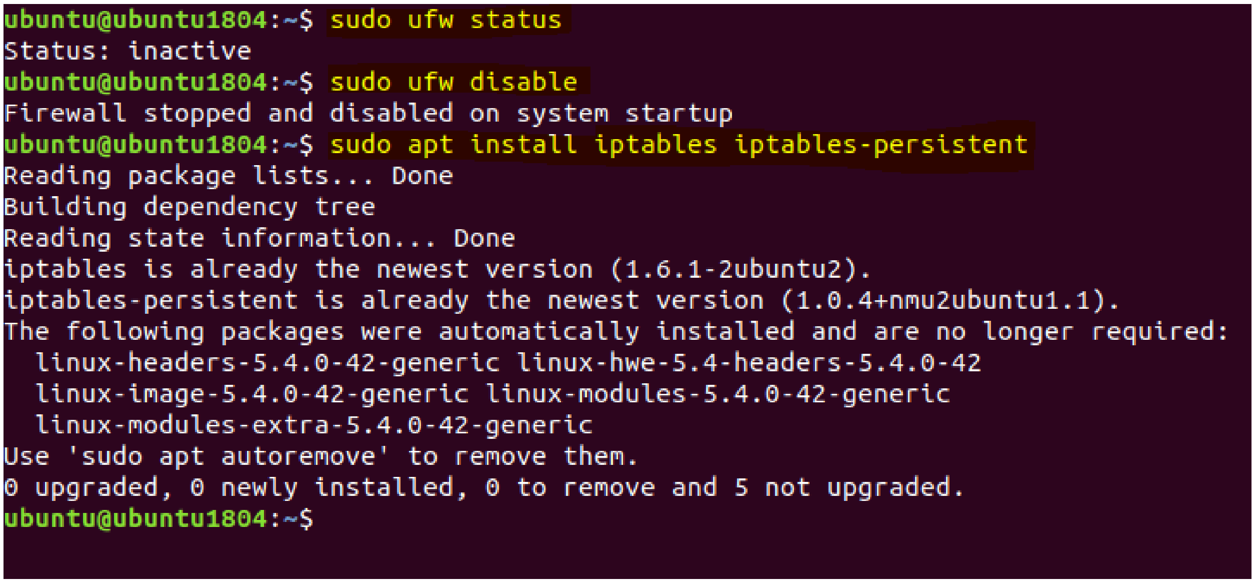


```
ubuntu@ubuntu1804:~$ sudo systemctl enable ssh.service
Synchronizing state of ssh.service with SysV service script with /lib/systemd/sy
stemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable ssh
ubuntu@ubuntu1804:~$ sudo service ssh start
ubuntu@ubuntu1804:~$ sudo service ssh status
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enab
   Active: active (running) since Thu 2022-12-08 11:55:35 EST; 1min 45s ago
 Main PID: 1083 (sshd)
    Tasks: 1 (limit: 2281)
   CGroup: /system.slice/ssh.service
           └─1083 /usr/sbin/sshd -D

Dec 08 11:55:37 ubuntu1804 systemd[1]: Reloading OpenBSD Secure Shell server.
Dec 08 11:55:37 ubuntu1804 sshd[1083]: Received SIGHUP; restarting.
Dec 08 11:55:37 ubuntu1804 sshd[1083]: Server listening on 0.0.0.0 port 22.
Dec 08 11:55:37 ubuntu1804 sshd[1083]: Server listening on :: port 22.
Dec 08 11:55:37 ubuntu1804 systemd[1]: Reloaded OpenBSD Secure Shell server.
Dec 08 11:55:37 ubuntu1804 systemd[1]: Reloading OpenBSD Secure Shell server.
Dec 08 11:55:37 ubuntu1804 sshd[1083]: Received SIGHUP; restarting.
Dec 08 11:55:37 ubuntu1804 sshd[1083]: Server listening on 0.0.0.0 port 22.
Dec 08 11:55:37 ubuntu1804 sshd[1083]: Server listening on :: port 22.
Dec 08 11:55:37 ubuntu1804 systemd[1]: Reloaded OpenBSD Secure Shell server.
```



```
ubuntu@ubuntu1804:~$ nmap -p 22 -sV 192.168.1.33

Starting Nmap 7.60 ( https://nmap.org ) at 2022-12-08 11:59 EST
Nmap scan report for ubuntu1804 (192.168.1.33)
Host is up (0.00016s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.7 (Ubuntu Linux; protocol 2.0
)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap
.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.26 seconds
ubuntu@ubuntu1804:~$ 
```

- We must then stop our firewall service from running before installing IPTables

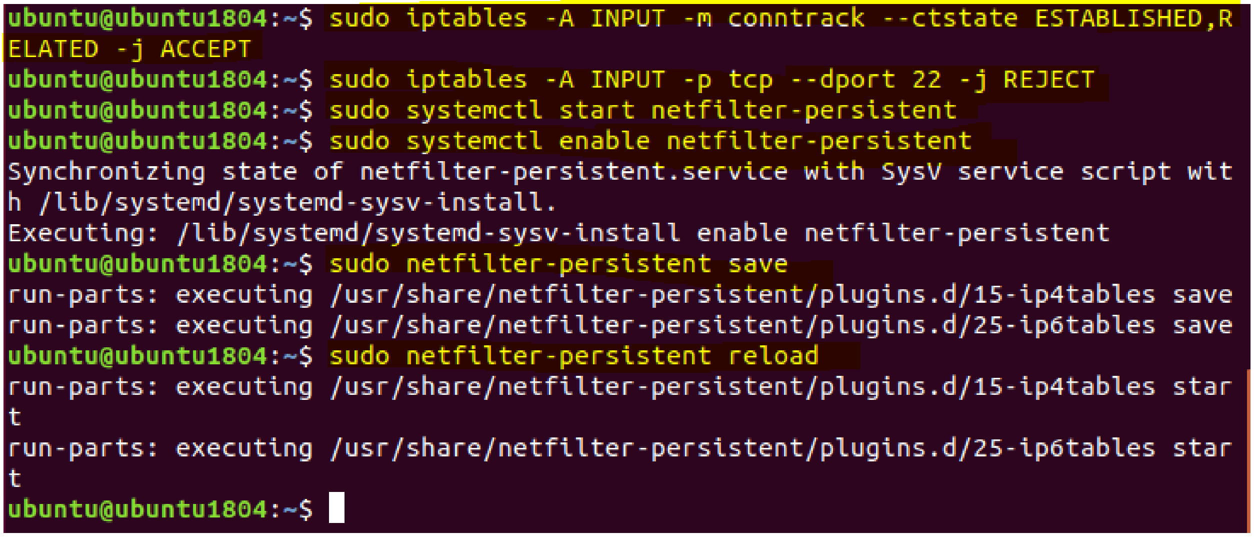


```
ubuntu@ubuntu1804:~$ sudo ufw status
Status: inactive
ubuntu@ubuntu1804:~$ sudo ufw disable
Firewall stopped and disabled on system startup
ubuntu@ubuntu1804:~$ sudo apt install iptables iptables-persistent
Reading package lists... Done
Building dependency tree
Reading state information... Done
iptables is already the newest version (1.6.1-2ubuntu2).
iptables-persistent is already the newest version (1.0.4+nmu2ubuntu1.1).
The following packages were automatically installed and are no longer required:
  linux-headers-5.4.0-42-generic linux-hwe-5.4-headers-5.4.0-42
  linux-image-5.4.0-42-generic linux-modules-5.4.0-42-generic
  linux-modules-extra-5.4.0-42-generic
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.
ubuntu@ubuntu1804:~$
```

- Once IPtables is installed, we have to allow all established connections and on-going sessions through iptables, meaning our current sessions will not be terminated.

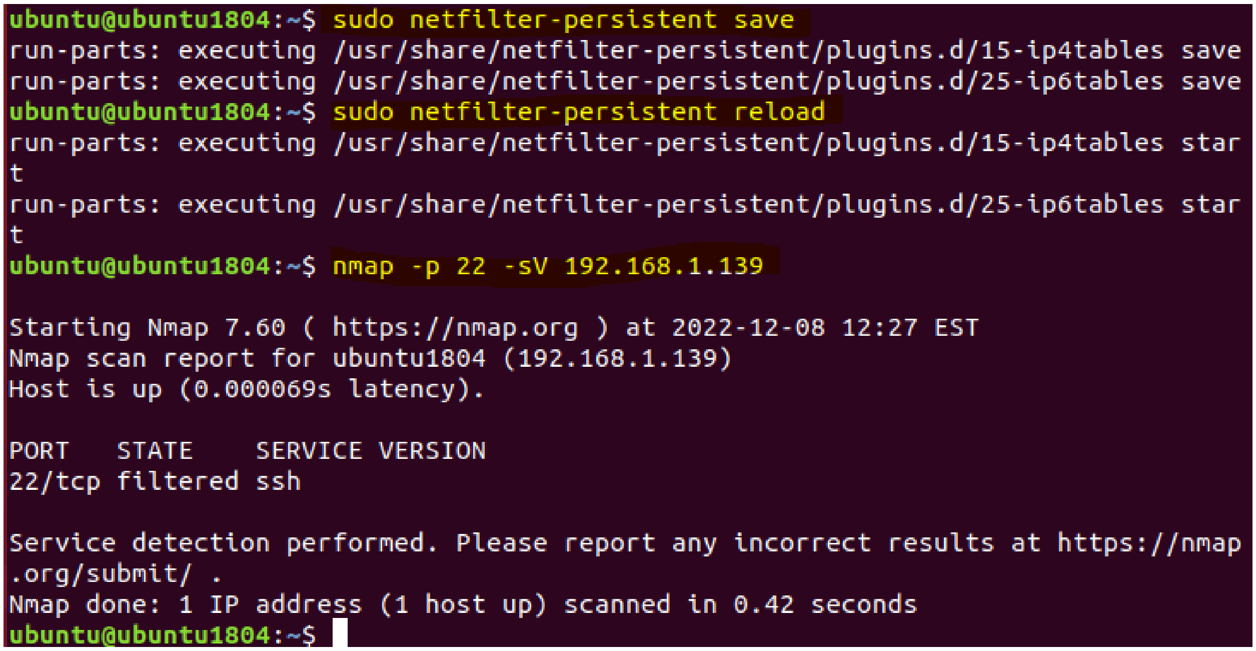
- The next command is: > *sudo iptables -A INPUT -m conntrack –csstate* ESTABLISHED,RELATED -j ACCEPT”
- -A = append chain rule-specification.
- -m commtrack = Works with the network connection tracking capabilities of the kernel
- –csstate = ESTABLISHED,RELATED: specifies the type of connection in which our rule will apply.
- -j = jump target, specifies the target of the rule if the packet matches

- Our Next command will block the desired incoming port, SSH (22)
  > *sudo iptables -A INPUT -p tcp –dport 22 -j REJECT*

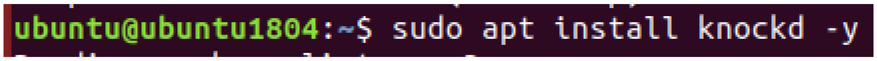


```
ubuntu@ubuntu1804:~$ sudo iptables -A INPUT -m conntrack --ctstate ESTABLISHED,R
ELATED -j ACCEPT
ubuntu@ubuntu1804:~$ sudo iptables -A INPUT -p tcp --dport 22 -j REJECT
ubuntu@ubuntu1804:~$ sudo systemctl start netfilter-persistent
ubuntu@ubuntu1804:~$ sudo systemctl enable netfilter-persistent
Synchronizing state of netfilter-persistent.service with SysV service script wit
h /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable netfilter-persistent
ubuntu@ubuntu1804:~$ sudo netfilter-persistent save
run-parts: executing /usr/share/netfilter-persistent/plugins.d/15-ip4tables save
run-parts: executing /usr/share/netfilter-persistent/plugins.d/25-ip6tables save
ubuntu@ubuntu1804:~$ sudo netfilter-persistent reload
run-parts: executing /usr/share/netfilter-persistent/plugins.d/15-ip4tables star
t
run-parts: executing /usr/share/netfilter-persistent/plugins.d/25-ip6tables star
t
ubuntu@ubuntu1804:~$
```

- Once we input these commands, save and reload the network filter. We can now see port 22 is in a 'filtered' state.

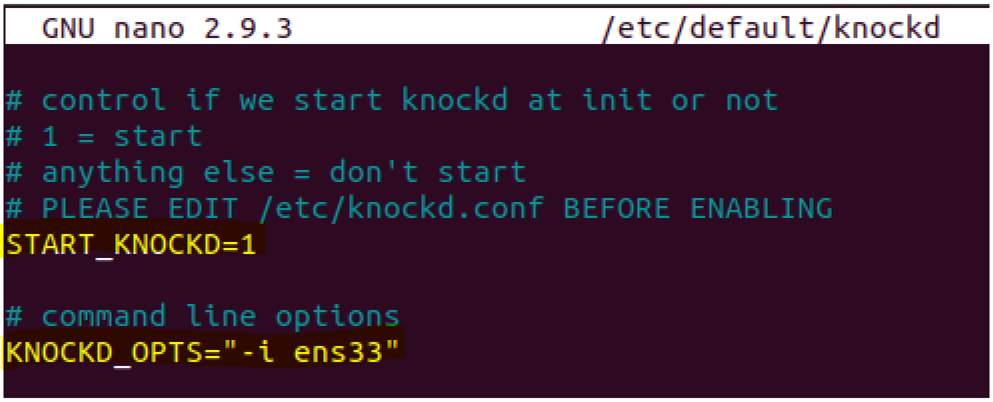


```
ubuntu@ubuntu1804:~$ sudo netfilter-persistent save
run-parts: executing /usr/share/netfilter-persistent/plugins.d/15-ip4tables save
run-parts: executing /usr/share/netfilter-persistent/plugins.d/25-ip6tables save
ubuntu@ubuntu1804:~$ sudo netfilter-persistent reload
run-parts: executing /usr/share/netfilter-persistent/plugins.d/15-ip4tables star
t
run-parts: executing /usr/share/netfilter-persistent/plugins.d/25-ip6tables star
t
ubuntu@ubuntu1804:~$ nmap -p 22 -sV 192.168.1.139

Starting Nmap 7.60 ( https://nmap.org ) at 2022-12-08 12:27 EST
Nmap scan report for ubuntu1804 (192.168.1.139)
Host is up (0.000069s latency).

PORT   STATE    SERVICE VERSION
22/tcp filtered ssh

Service detection performed. Please report any incorrect results at https://nmap
.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.42 seconds
ubuntu@ubuntu1804:~$
```



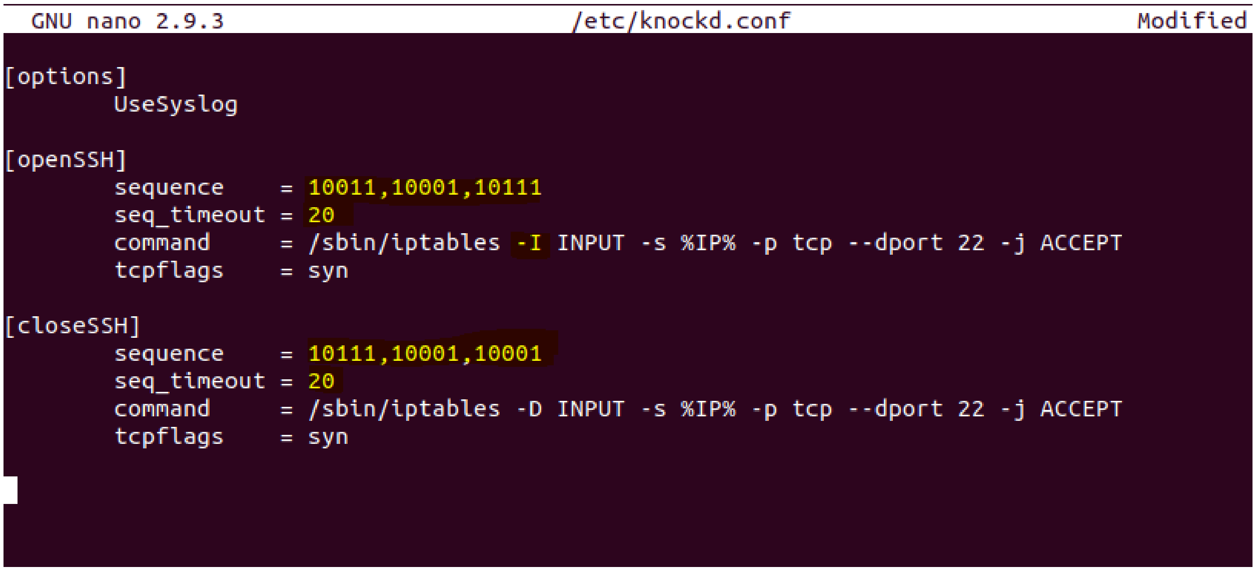
```
ubuntu@ubuntu1804:~$ sudo apt install knockd -y
```

- Install knockd and use either Nano or Vi to edit the /etc/default/knockd file. Change START_KNOCKD= to have a value of '1', and OPTS to -i 'internet adapter'.



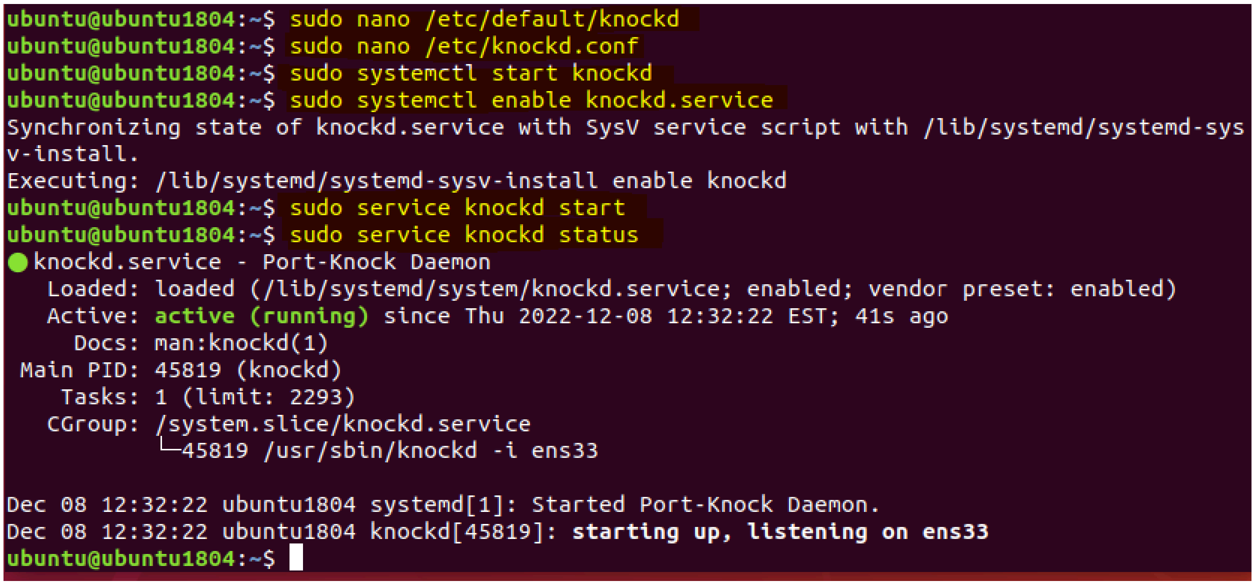
```
  GNU nano 2.9.3                    /etc/default/knockd

# control if we start knockd at init or not
# 1 = start
# anything else = don't start
# PLEASE EDIT /etc/knockd.conf BEFORE ENABLING
START_KNOCKD=1

# command line options
KNOCKD_OPTS="-i ens33"
```

- Save and exit the file, and modify the /etc/knockd.conf file next to replicate this picture:



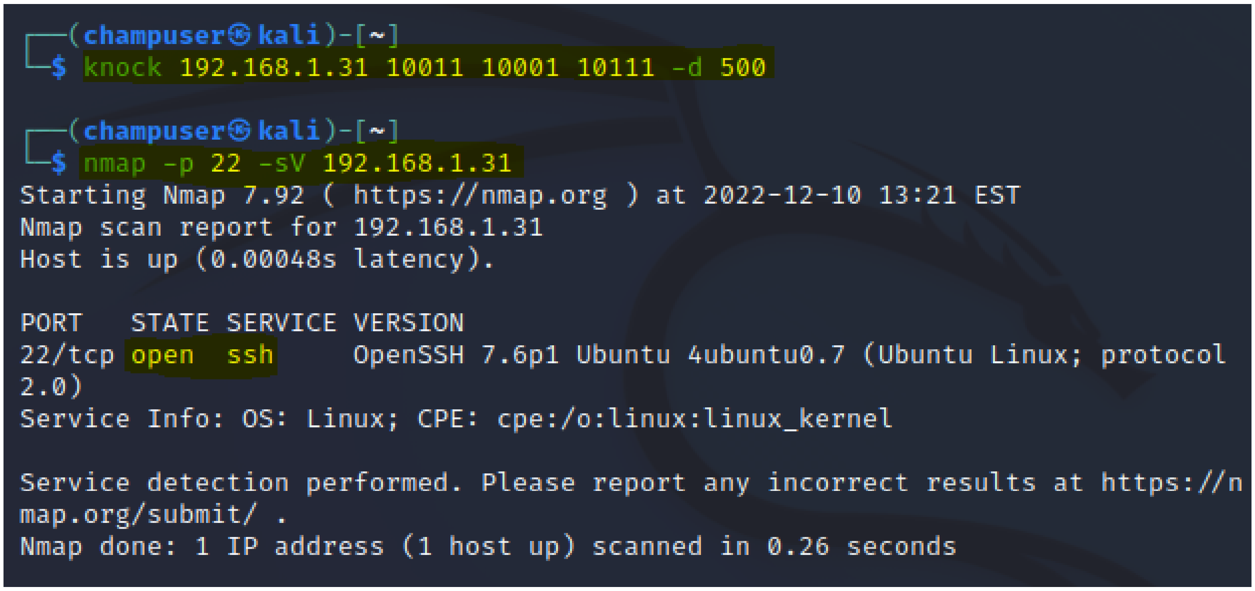
```
  GNU nano 2.9.3                     /etc/knockd.conf                     Modified

[options]
        UseSyslog

[openSSH]
        sequence    = 10011,10001,10111
        seq_timeout = 20
        command     = /sbin/iptables -I INPUT -s %IP% -p tcp --dport 22 -j ACCEPT
        tcpflags    = syn

[closeSSH]
        sequence    = 10111,10001,10001
        seq_timeout = 20
        command     = /sbin/iptables -D INPUT -s %IP% -p tcp --dport 22 -j ACCEPT
        tcpflags    = syn
```

- Once completed, save and exit the file and start the knockd service. We should see it is now listening on ‘ens33’



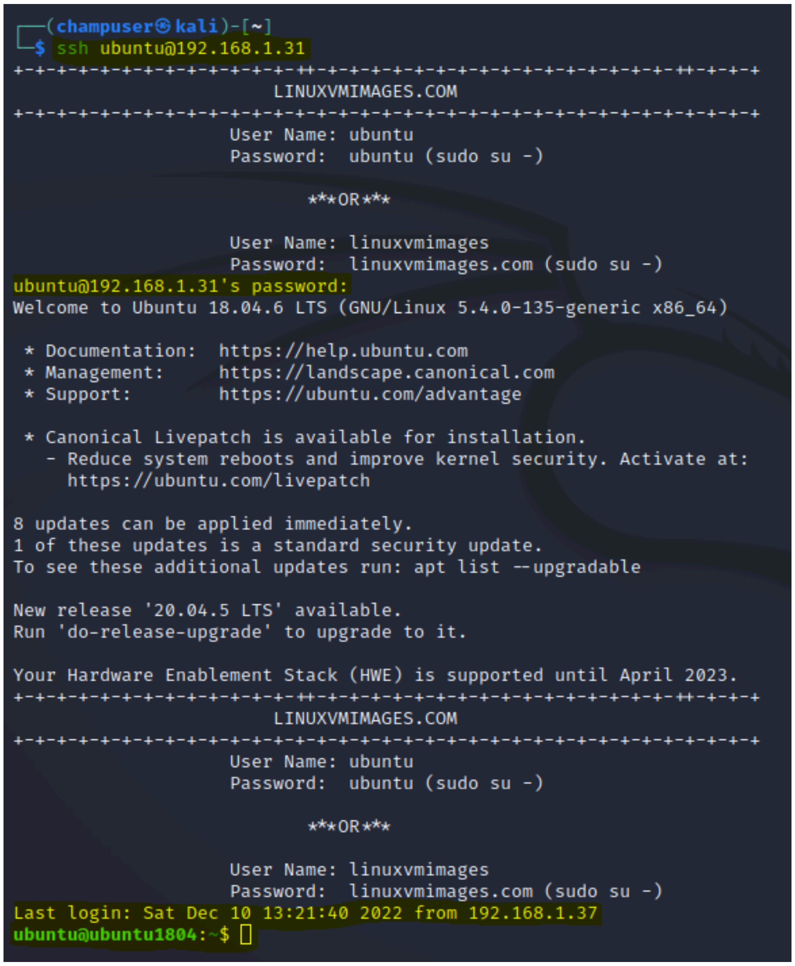
```
ubuntu@ubuntu1804:~$ sudo nano /etc/default/knockd
ubuntu@ubuntu1804:~$ sudo nano /etc/knockd.conf
ubuntu@ubuntu1804:~$ sudo systemctl start knockd
ubuntu@ubuntu1804:~$ sudo systemctl enable knockd.service
Synchronizing state of knockd.service with SysV service script with /lib/systemd/systemd-sys
v-install.
Executing: /lib/systemd/systemd-sysv-install enable knockd
ubuntu@ubuntu1804:~$ sudo service knockd start
ubuntu@ubuntu1804:~$ sudo service knockd status
● knockd.service - Port-Knock Daemon
   Loaded: loaded (/lib/systemd/system/knockd.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2022-12-08 12:32:22 EST; 41s ago
     Docs: man:knockd(1)
 Main PID: 45819 (knockd)
    Tasks: 1 (limit: 2293)
   CGroup: /system.slice/knockd.service
           └─45819 /usr/sbin/knockd -i ens33

Dec 08 12:32:22 ubuntu1804 systemd[1]: Started Port-Knock Daemon.
Dec 08 12:32:22 ubuntu1804 knockd[45819]: starting up, listening on ens33
ubuntu@ubuntu1804:~$
```

- Moving to our Kali machine, we are able to port is closed with that first command.



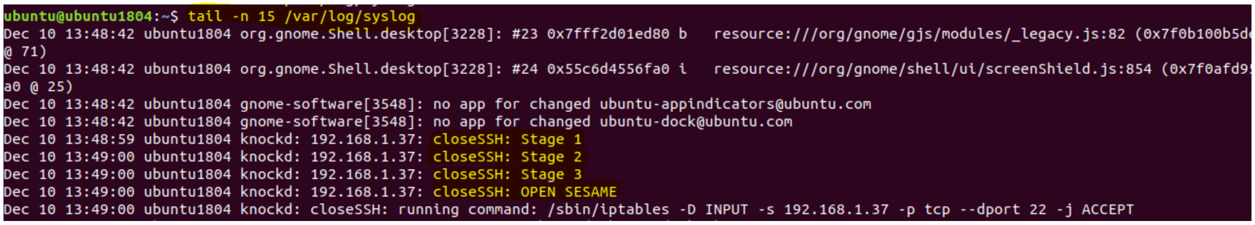
```
┌──(champuser㉿kali)-[~]
└─$ knock 192.168.1.31 10011 10001 10111 -d 500

┌──(champuser㉿kali)-[~]
└─$ nmap -p 22 -sV 192.168.1.31
Starting Nmap 7.92 ( https://nmap.org ) at 2022-12-10 13:21 EST
Nmap scan report for 192.168.1.31
Host is up (0.00048s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.7 (Ubuntu Linux; protocol
2.0)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://n
map.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.26 seconds
```

- We are not able to SSH into Ubuntu while the port is closed. Once we open the port, we are now able to SSH into port 22.

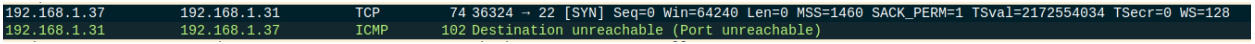


```
┌──(champuser㉿kali)-[~]
└─$ ssh ubuntu@192.168.1.31
+-+-+-+-+-+-+-+-+-+-+-+-+-++-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-++-+-+-+
                        LINUXVMIMAGES.COM
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                    User Name: ubuntu
                    Password:  ubuntu (sudo su -)

                           ***OR***

                    User Name: linuxvmimages
                    Password:  linuxvmimages.com (sudo su -)
ubuntu@192.168.1.31's password:
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 5.4.0-135-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

8 updates can be applied immediately.
1 of these updates is a standard security update.
To see these additional updates run: apt list --upgradable

New release '20.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Your Hardware Enablement Stack (HWE) is supported until April 2023.
+-+-+-+-+-+-+-+-+-+-+-+-+-++-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-++-+-+-+
                        LINUXVMIMAGES.COM
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                    User Name: ubuntu
                    Password:  ubuntu (sudo su -)

                           ***OR***

                    User Name: linuxvmimages
                    Password:  linuxvmimages.com (sudo su -)
Last login: Sat Dec 10 13:21:40 2022 from 192.168.1.37
ubuntu@ubuntu1804:~$ 
```

- Within our Ubuntu logs, we can see commands opening port 22, with the second picture of a failed attempt while the port is closed.



```
ubuntu@ubuntu1804:~$ tail -n 15 /var/log/syslog
Dec 10 13:48:42 ubuntu1804 org.gnome.Shell.desktop[3228]: #23 0x7fff2d01ed80 b   resource:///org/gnome/gjs/modules/_legacy.js:82 (0x7f0b100b5d
@ 71)
Dec 10 13:48:42 ubuntu1804 org.gnome.Shell.desktop[3228]: #24 0x55c6d4556fa0 i   resource:///org/gnome/shell/ui/screenShield.js:854 (0x7f0afd9
a0 @ 25)
Dec 10 13:48:42 ubuntu1804 gnome-software[3548]: no app for changed ubuntu-appindicators@ubuntu.com
Dec 10 13:48:42 ubuntu1804 gnome-software[3548]: no app for changed ubuntu-dock@ubuntu.com
Dec 10 13:48:59 ubuntu1804 knockd: 192.168.1.37: closeSSH: Stage 1
Dec 10 13:49:00 ubuntu1804 knockd: 192.168.1.37: closeSSH: Stage 2
Dec 10 13:49:00 ubuntu1804 knockd: 192.168.1.37: closeSSH: Stage 3
Dec 10 13:49:00 ubuntu1804 knockd: 192.168.1.37: closeSSH: OPEN SESAME
Dec 10 13:49:00 ubuntu1804 knockd: closeSSH: running command: /sbin/iptables -D INPUT -s 192.168.1.37 -p tcp --dport 22 -j ACCEPT
```



```
192.168.1.37      192.168.1.31      TCP       74 36324 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=2172554034 TSecr=0 WS=128
192.168.1.31      192.168.1.37      ICMP     102 Destination unreachable (Port unreachable)
```

- The ICMP packet shows that there was an error trying to ssh into the box when the port was closed

References

You will be using outside resources than the labs since most of the topics expand on what we have done thus far. Please cite your references here.

https://www.youtube.com/watch?v=Gi2l4tPRP7o
https://www.youtube.com/watch?v=4xvLIAxm-gc
https://www.reelix.za.net/2020/10/wireshark-filtering-for-port-knocking.html
https://www.tecmint.com/port-knocking-to-secure-ssh/
https://vk9-sec.com/secure-ssh-server-using-port-knocking-knockd-on-linux/

Build Documentation

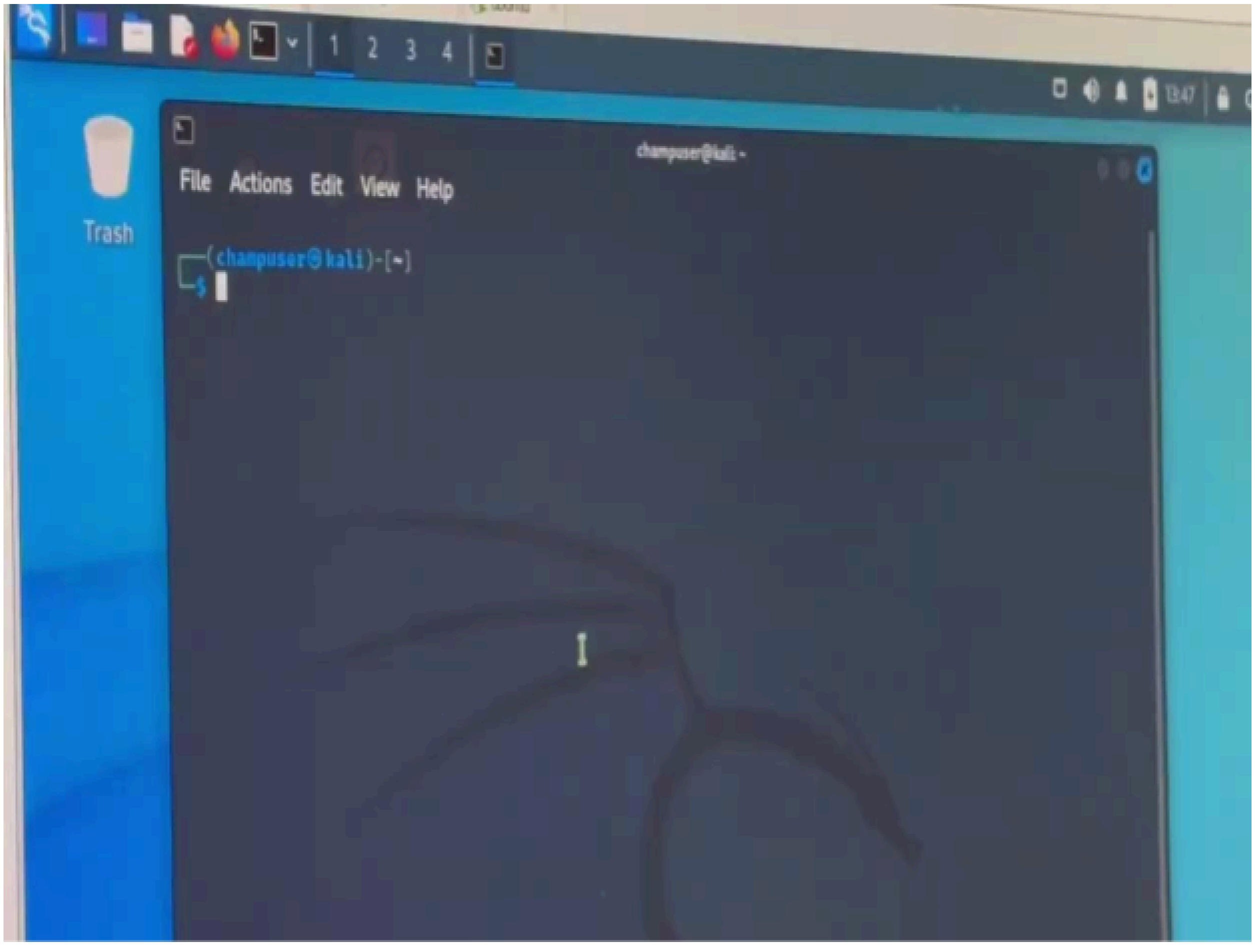
The body of the project. How are things set up? How did you get from point A to point B (the project description). This includes descriptions of what was done and specifics including but not limited to commands entered, files used, system networking settings, etc.

Our lab is set up similarly to the labs we've done all semester. We first started off by looking up to see what port knocking really was and how it's implemented. We saw that Ubuntu had more sources to help us succeed, so we went with Ubuntu. The IP was already preset on both the Kali and Ubuntu box. All we had to do was a trial and error of commands and see what worked and what didn't work. 'Point A' for us was not known much until 'Point B' successfully opened and closed port 22.

Completion Test

Show via descriptions, screenshots, and/or video, your project working. It's important here to display system names or uniquely identifiable info from VMware Workstation or vSphere so the instructor knows what system is what.

- **Completion Test:** To view the recording below, double click on the thumbnail.



```
┌──(champuser㉿kali)-[~]
└─$ 
```

- Hit the arrow top right corner to open!

Discussion

Descriptions of difficulties you faced, how you troubleshooted them, and what the outcome was.

We encountered quite a load of trouble. The internet was very helpful with the troubleshooting process. Using prior knowledge of importing a VM to VMWare, when we booted up the Ubuntu box, we found that we couldn't just 'download' Wireshark. There were lots of issues we ran into trying to install Wireshark. The next issue was finding sources that can get us from nowhere to somewhere to success. This was definitely the hardest part since no one source could do that. We compiled multiple commands from multiple different sources to create our final result.

During our troubleshooting of the Ubuntu box we had no progress, the next thought was to try connecting our Kali boxes on VMware Workstation. By executing the previous command of "nmap -p 22 -sV 192.168.1.31" on our Kali box the port was now closed which is what we needed. We moved back to the next steps of the lab and opened the port from our Kali box with

“Knock 192.168.1.31 10011 10001 10111 -d 500” which opened the port. The next and final step was to SSH into the Ubuntu box, and it finally worked.