

Cyber@UC Meeting 72

Firewalls/IPTables

If You're New!

- Join our Slack: cyberatuc.slack.com
- Check out our website: cyberatuc.org
- **SIGN IN!** (*Slackbot will post the link in #general every Wed@6:30*)
- Feel free to get involved with one of our committees:
Content Finance Public Affairs Outreach Recruitment
- Ongoing work in our research lab!



Announcements



- **LOGO UPDATE**
 - Isn't she a beauty?
- **NSA Internship Application** window closing Oct 31st
- **NSA Codebreaker Hack-a-thon**
 - Saturday 10/20/18
 - Hosted by Cyber@UC
- Outreach to **Lakota East** yesterday
- **Chipotle fundraiser**
 - Saturday Nov 3rd 4pm–8pm
- **Election Nominations!**

Weekly Content

Google+ shutting down

- Shutting down after a massive security breach leaked data of **>500k** users
 - Allowed 3rd party developers access to usernames, emails, addresses, occupation, date of birth, photos, and gender
- No evidence of leak being used by 438 developers that could have used it
- Vulnerability began in 2015, fixed when found in March 2018
 - Google chose not to disclose because of Facebook/Cambridge Analytica
- Nature of vuln appears similar to Facebook API flaw
- Google+ will be shut down by end of August 2019, but will continue to be offered as an enterprise product
- Added new privacy controls to dev access through **Project Strobe**
 - Permission requests asked individually

Silk Road Admin Pleads Guilty

- Silk Road: dark web marketplace, mainly known for drug trafficking
- Gary Davis, a.k.a. Libertas, was a Silk Road admin
- Plead guilty for drug trafficking
- Silk Road fell after servers were raided in 2013 and founder was arrested
 - Sentenced to life in prison
- Bitcoins currently valued at 33.6 million were also seized
 - Sold in auction, bet they are regretting that right now huh?
- Davis helped the site run smoothly, essentially playing customer service
- Could receive up to 20 years
 - sentencing to occur January 17th 2019



MikroTik Router Vulnerability Resurfaces

- Originally found in April 2018 and patched within a day
- Directory traversal vulnerability: CVE-2018-14847
 - Initially rated as medium, but has been reclassified as critical
- New PoC allows remote code execution and root shell access
- Uses directory traversal to steal admin credentials from user database file, then writes a file on system to gain root shell access remotely
- Could allow malware deployed on routers, or firewalls bypassed
- Follows on a VPNFilter malware and cryptojacker a few months back
- The report also disclosed 4 other new vulns
- While patches are out >70% of MikroTik routers still vulnerable



Recommended Reading

<https://krebsonsecurity.com/2018/10/when-security-researchers-pose-as-cybercrooks-who-can-tell-the-difference/>

<https://www.welivesecurity.com/2018/10/05/virus-bulletin-2018-supply-chain-hacking-grows/>



Services and Security

Quite unlike the birds and the bees



The Topics Today Go Something Exactly Like This

- How Firewalls / Ports work
- Tool Overview
 - Iptables / UFW / GFW
 - Nmap
 - NetCat (NC)
- 127.0.0.1 on the range
 - Making, detecting, and protecting services

What Is a Firewall and What Does It Do?

- Monitors incoming and outgoing network traffic and chooses actions to take on that traffic
 - Allow, block, log, etc.
- Used as a barrier between trusted devices and untrusted devices
- Firewalls can exist as hardware, software, or both
- Firewalls can be used a pretty much any point in a network
 - Between internal and external network, within internal network, and on device



Quick Refresher on Ports

- Every computer has 65535 ports per interface
- Every interface can be independent of one another but we will assume they are all the same here
- “Interfaces” mean “ways of addressing the device” not physical interfaces
- Typically ports are addressed by **<IP or DNS>:Port**
- Example: **www.google.com:80**

Where's the Ethernet?



Table Types

- Mangle Table: Modifying TCP packet quality of service bits before routing, rarely used in a home environment
- NAT Table: Network Address Translation
- Filter Table: responsible for filtering packets, broken down into 3 chains
 - Forward: filters packets to machines behind the firewall
 - Input: filters packets to the firewall
 - Output: filters packets from the firewall



Queue Type	Queue Function	Packet Transformation Chain in Queue	Chain Function
Filter	Packet filtering	FORWARD	Filters packets to servers accessible by another NIC on the firewall.
		INPUT	Filters packets destined to the firewall.
		OUTPUT	Filters packets originating from the firewall
Nat	Network Address Translation	PREROUTING	Address translation occurs before routing. Facilitates the transformation of the destination IP address to be compatible with the firewall's routing table. Used with NAT of the destination IP address, also known as destination NAT or DNAT .
		POSTROUTING	Address translation occurs after routing. This implies that there was no need to modify the destination IP address of the packet as in pre-routing. Used with NAT of the source IP address using either one-to-one or many-to-one NAT. This is known as source NAT , or SNAT .
		OUTPUT	Network address translation for packets generated by the firewall. (Rarely used in SOHO environments)
Mangle	TCP header modification	PREROUTING POSTROUTING OUTPUT INPUT FORWARD	Modification of the TCP packet quality of service bits before routing occurs. (Rarely used in SOHO environments)

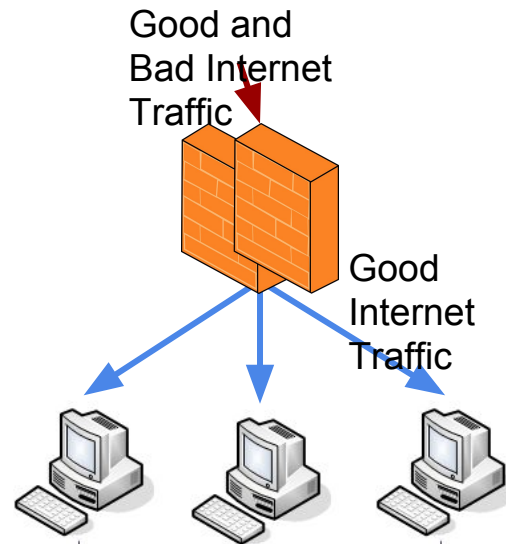


Put on your ~~3D glasses~~ **Linux Distro**
now



What is IPTables? What is UFW?

- UFW: Uncomplicated Firewall
 - Comes by default in ubuntu
 - Essentially just a nicer interface for iptables
- IPTables is a popular firewall/NAT software solution
- Integrates well with Linux Kernel
- Very versatile
- Stateful packet inspection: occasionally views contents of data flows and attempts to predict next action, good for FTP and DNS





Tool Rundown: Nmap



Nmap is the **best tool you will ever use**

Features Include:

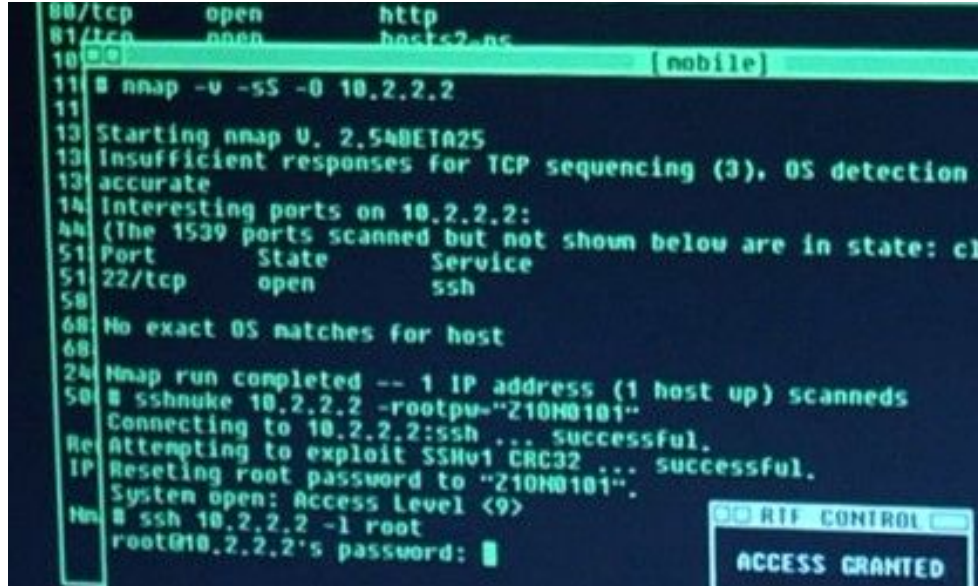
- Host discovery
- Port scanning
- Version detection of services
- OS detection
- Install with **sudo apt install nmap** (or **nmap.org** on windows)



Tool Rundown: Nmap

Nmap was also featured in the Matrix (1999) when they used an actual 0-day

```
80/tcp    open      http
81/tcp    open      hosts2.nc
10 [0] [nobile]
11 # nmap -v -sS -O 10.2.2.2
11
13 Starting nmap V. 2.54BETA25
13 Insufficient responses for TCP sequencing (3), OS detection
13 accurate
14 Interesting ports on 10.2.2.2:
44 (The 1539 ports scanned but not shown below are in state: closed)
51 Port      State      Service
51 22/tcp    open      ssh
58
68 No exact OS matches for host
68
24 Nmap run completed -- 1 IP address (1 host up) scanned
50 # sshnuke 10.2.2.2 -rootpw="210H0101"
Connecting to 10.2.2.2:ssh ... successful.
Re Attempting to exploit SSHv1 CRC32 ... successful.
IP Resetting root password to "210H0101".
System open: Access Level <9>
# ssh 10.2.2.2 -l root
root@10.2.2.2's password: █
```



The image shows a terminal window with a dark background and light green text. The output of an nmap scan is visible, showing open ports 80/tcp (http) and 22/tcp (ssh). Following the scan, the user runs 'sshnuke' to exploit a vulnerability in SSHv1, which is successful. The terminal then shows the user logging in as root and being prompted for a password. In the bottom right corner, there is a small window titled 'RTF CONTROL' with the text 'ACCESS GRANTED'.



Tool Rundown: Netcat



NetCat is a simple utility for opening connections among other things

Features Include:


- Many things
- Install with **sudo apt install netcat** (or **nmap.org** on windows)

Types of Actions

- Accept - stop processing and allow packet through
- Drop - stop processing and block packet
- Log - log packet into and continue processing with next rule
- Reject - like drop but also returns an error message
- DNAT
- SNAT
- Masquerade



iptables Rule Parameters

- -t <table>
 - -j <target/Action>
 - -A <append rule to end of chain>
 - -F <Deletes all rules in selected table>
 - -p <protocol: TCP, UDP, ICMP, etc.>
 - -s <src-ip>
 - -d <dst-ip>
 - -i <interface: eth0>
 - -o <output interface: eth1>
- 

Rule Examples

```
iptables -A INPUT -p tcp --dport ssh -j ACCEPT
```

```
iptables -A OUTPUT -p icmp --icmp-type echo-request -j ACCEPT
```

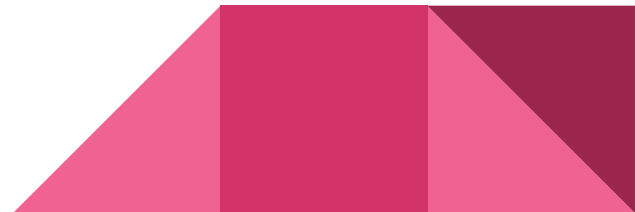
```
iptables -A INPUT -p icmp --icmp-type echo-reply -j ACCEPT
```

```
iptables -A INPUT -p icmp --icmp-type echo-request \  
-m limit --limit 1/s -i eth0 -j ACCEPT
```

```
iptables -A INPUT -p tcp --syn -m limit --limit 5/s -i eth0 -j ACCEPT
```

```
iptables -A INPUT -p tcp --dport 80 -j REJECT
```

```
iptables -A INPUT -p tcp --dport 80 -j DROP
```



More Useful Commands

```
# iptables -n -L -v --line-numbers
```

```
# iptables-save > /etc/iptables/rules.v4
```

```
# iptables-restore -c < /etc/iptables/rules.v4
```

Saving these rules to be persistent would require installing of iptables-persistent



Using nmap

- Keep it easy for now and just run **nmap localhost**
- Teach yourself nmap as well because it's great



Using netcat (nc)

- Keep it simple and just run **nc -l -p (port#)** a few times with different port numbers
- Try to find your ports with nmap then block them with UFW/iptables



Breakout Session

Think of something you would want to accomplish if you were in charge of developing a corporate firewall and try to come up with a rule(s) to handle that

- Open fake services with **nc -l -p (port)**
- Find your fake services with **nmap localhost**
- Protect your local services with **UFW** or **iptables**
- If you have any questions, run **man <command>** to see more info about a command on linux



Some sources

http://www.linuxhomenetworking.com/wiki/index.php/Quick_HOWTO_%3a_Ch14_%3a_Linux_Firewalls_Using_iptables#.W7p362hKiU

<https://help.ubuntu.com/community/IptablesHowTo>

<https://www.cyberciti.biz/tips/linux-iptables-examples.html>

<https://www.digitalocean.com/community/tutorials/how-to-implement-a-basic-firewall-template-with-iptables-on-ubuntu-14-04>

