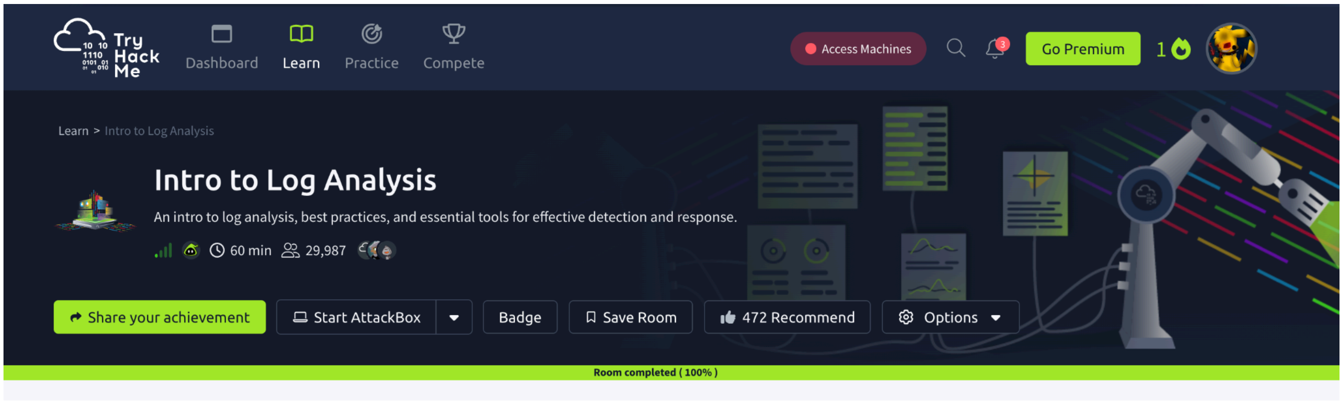
Working with firewalls and logging:

Prep for at home lab:

Before I started the at home lab on my VM, I gained more understanding about firewalls and logging through a TryHackMeLab. The completion of the lab is shown below:



Intro to Log Analysis

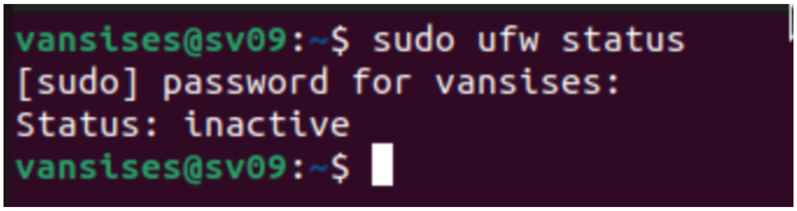
An intro to log analysis, best practices, and essential tools for effective detection and response.

Room completed (100%)

At home lab:

Enabling UFW (Uncomplicated Firewall):

The command I used to check the UFW status was:
 sudo ufw status

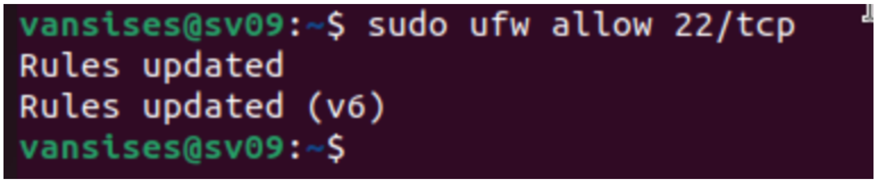


```
vansises@sv09:~$ sudo ufw status
[sudo] password for vansises:
Status: inactive
vansises@sv09:~$
```

The purpose of this command is to:

- Check if the firewall is active or not on my VM
- Confirms if UFW is installed and currently enabled
- Through use on my VM, it was shown the firewall is inactive

The command I used to allow SSH traffic was:
sudo ufw allow 22/tcp

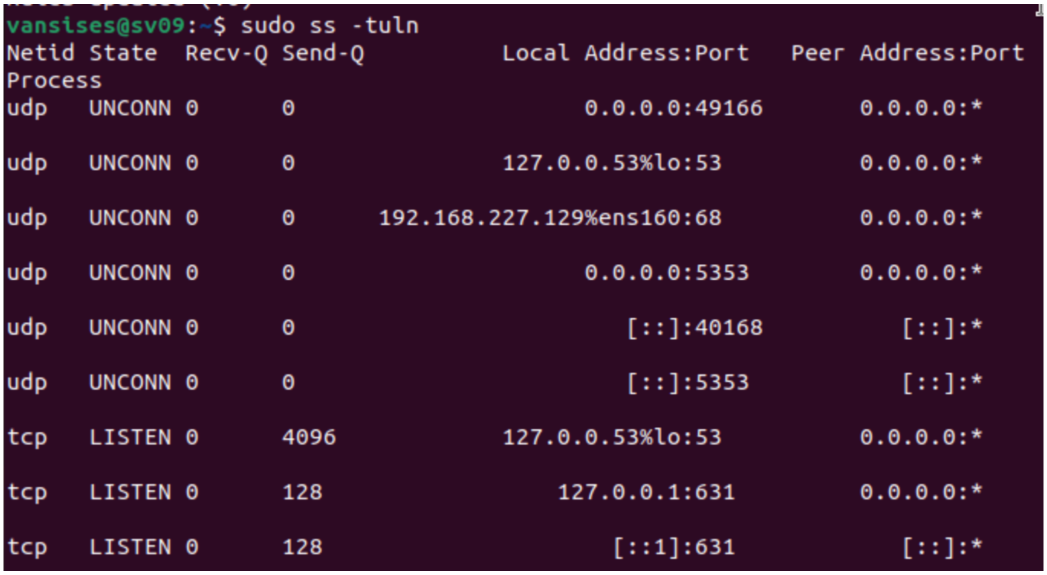


```
vansises@sv09:~$ sudo ufw allow 22/tcp
Rules updated
Rules updated (v6)
vansises@sv09:~$
```

The purpose of this command is to:

- Allows SSH access through port 22 before enabling the firewall
- Prevents the user from locking themself out of a remote session when UFW is activated
- Ensures continued access to your VM over SSH
- Once completed, as shown above the rules are updated

The command I used to view open ports was:
sudo ss -tuln

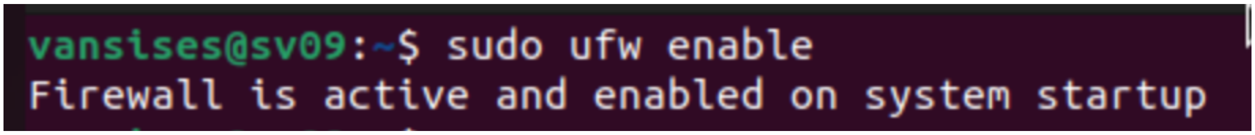


```
vansises@sv09:~$ sudo ss -tuln
Netid State  Recv-Q Send-Q          Local Address:Port   Peer Address:Port
Process
udp   UNCONN 0      0                     0.0.0.0:49166       0.0.0.0:*

udp   UNCONN 0      0               127.0.0.53%lo:53          0.0.0.0:*

udp   UNCONN 0      0       192.168.227.129%ens160:68         0.0.0.0:*

udp   UNCONN 0      0                     0.0.0.0:5353        0.0.0.0:*

udp   UNCONN 0      0                        [::]:40168          [::]:*

udp   UNCONN 0      0                        [::]:5353           [::]:*

tcp   LISTEN 0      4096            127.0.0.53%lo:53          0.0.0.0:*

tcp   LISTEN 0      128                 127.0.0.1:631         0.0.0.0:*

tcp   LISTEN 0      128                     [::1]:631            [::]:*
```

The purpose of this command is to:

- List all TCP and UDP ports that are currently open and listening
- Helps identify which services are exposed to the network
- Shows addresses of ports connected to the network

The command I used to enable UFW was:
sudo ufw enable

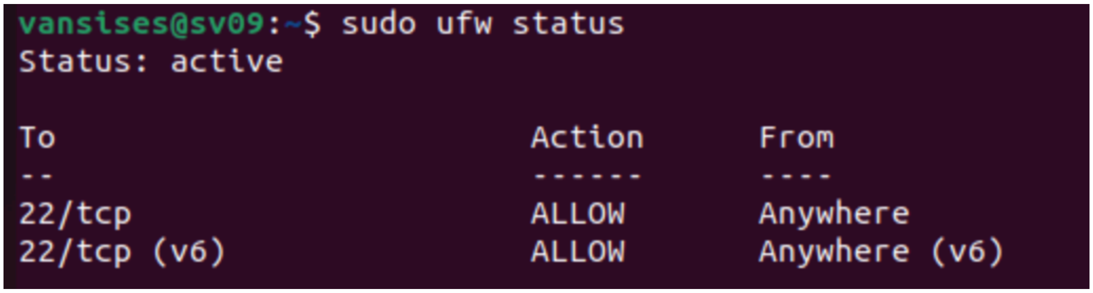


```
vansises@sv09:~$ sudo ufw enable
Firewall is active and enabled on system startup
```

The purpose of this command is to:

- Activate firewall and begin to apply UFW rules
- Once enabled, the default policies are enforced

The command I used to check the firewall status was:
sudo ufw status

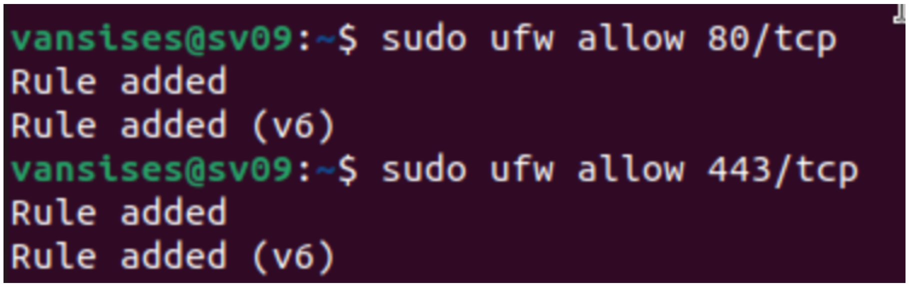


```
vansises@sv09:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
22/tcp                     ALLOW       Anywhere
22/tcp (v6)                ALLOW       Anywhere (v6)
```

The purpose of this command is to:

- Check firewall is now enforced after updating rules
- View the open ports allowed

The command I used to allow web traffic (ports 80 and 443) was:
sudo ufw allow 80/tcp
sudo ufw allow 443/tcp

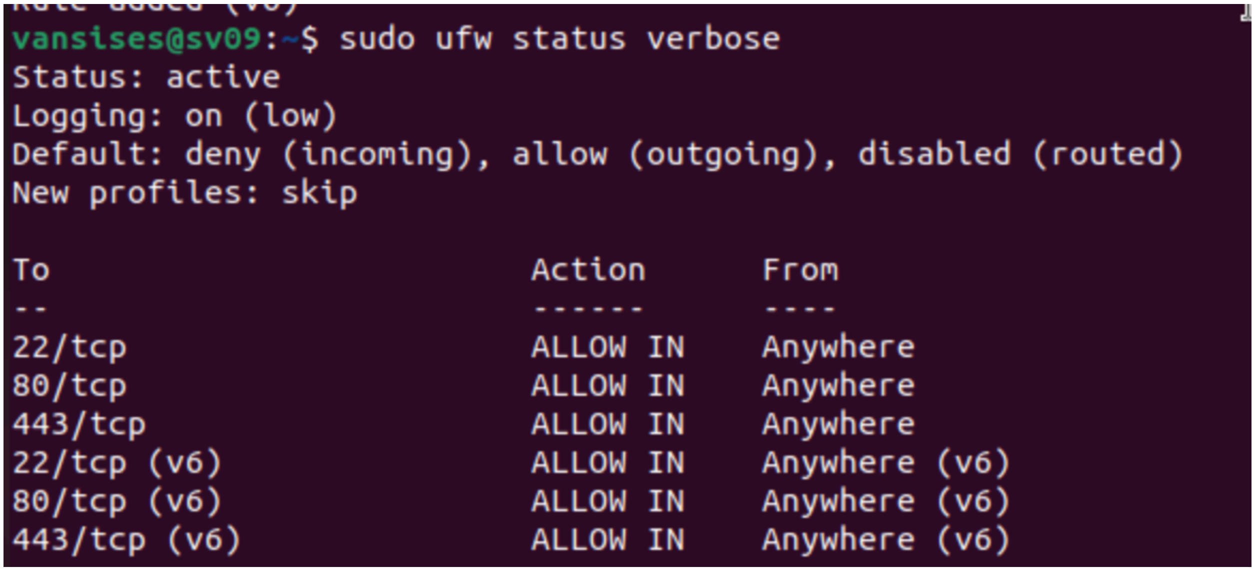


```
vansises@sv09:~$ sudo ufw allow 80/tcp
Rule added
Rule added (v6)
vansises@sv09:~$ sudo ufw allow 443/tcp
Rule added
Rule added (v6)
```

The purpose of this command is to:

- Allow HTTP and HTTPS traffic through the firewall
- Needed for VM to run a web server
- Rule added confirmed that each command worked

The command I used to check the firewall status in more detail was:
sudo ufw status verbose

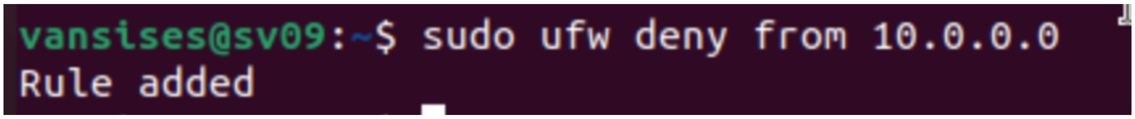


```
Rule added (v6)
vansises@sv09:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip

To                           Action      From
--                           ------      ----
22/tcp                       ALLOW IN    Anywhere
80/tcp                       ALLOW IN    Anywhere
443/tcp                      ALLOW IN    Anywhere
22/tcp (v6)                  ALLOW IN    Anywhere (v6)
80/tcp (v6)                  ALLOW IN    Anywhere (v6)
443/tcp (v6)                 ALLOW IN    Anywhere (v6)
```

The purpose of this command is to:

- For viewing detailed information your about current firewall rules
- Allows troubleshooting or verifying rule behavior

The command I would use to block a specific IP (Ex: 10.0.0.0) is:
 sudo ufw deny from 10.0.0.0

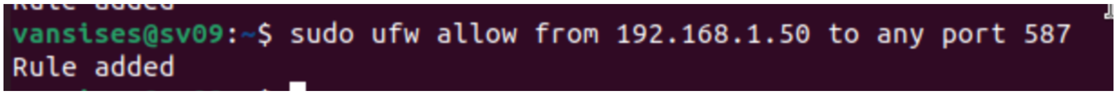


```
vansises@sv09:~$ sudo ufw deny from 10.0.0.0
Rule added
```

The purpose of this command is to:

- Prevents all traffic from a specific given IP address
- Useful for blocking known malicious and unwanted IPs
- Rule added confirms that the command worked

The command I would use to allow a specific IP access to port 587 is:
 sudo ufw allow from 192.168.1.50 to any port 587

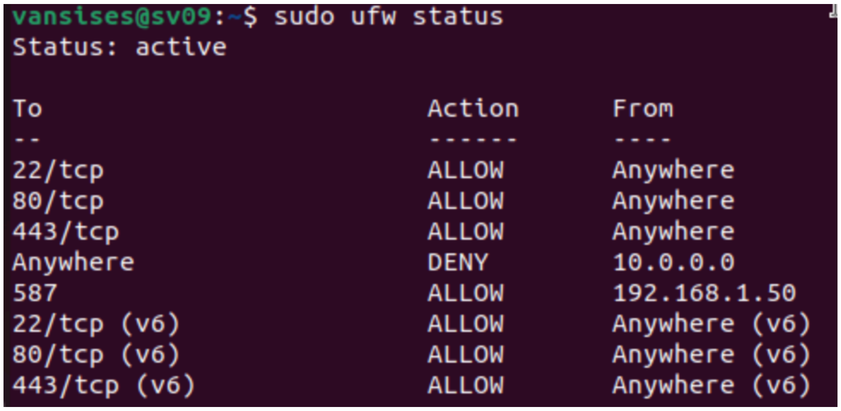


```
Rule added
vansises@sv09:~$ sudo ufw allow from 192.168.1.50 to any port 587
Rule added
```

The purpose of this command is to:

- Grants 192.168.1.50 access to port 587 on my VM
- Port 587 is used for email submission via SMTP

The command I used to confirm my current rules again was:
 sudo ufw status

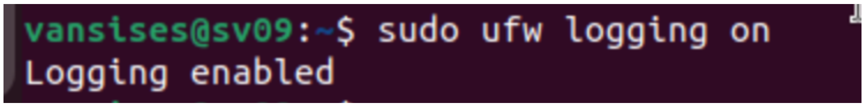


```
vansises@sv09:~$ sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
22/tcp                     ALLOW       Anywhere
80/tcp                     ALLOW       Anywhere
443/tcp                    ALLOW       Anywhere
Anywhere                   DENY        10.0.0.0
587                        ALLOW       192.168.1.50
22/tcp (v6)                ALLOW       Anywhere (v6)
80/tcp (v6)                ALLOW       Anywhere (v6)
443/tcp (v6)               ALLOW       Anywhere (v6)
```

As shown above, we can see all the new rules were added.

Enabling and Configuring UFW Logging:

The command I used to turn on UFW logging was:
sudo ufw logging on

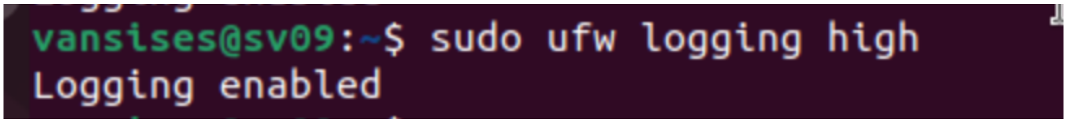


```
vansises@sv09:~$ sudo ufw logging on
Logging enabled
```

The purpose of this command is to:

- Begin logging UFW events
- Important for monitoring network activity
- “Logging enabled” confirms the command went through

The command I used to increase the logging level to high was:
sudo ufw logging high



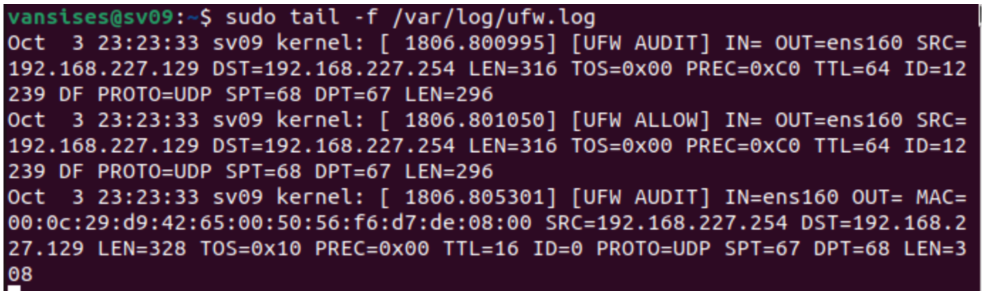
```
vansises@sv09:~$ sudo ufw logging high
Logging enabled
```

The purpose of this command is to:

- Set the log detail to "high", which includes all blocked packets and any connection attempts
- Used to analyze security threats as well as unauthorized access attempts

Logging is very useful because it reveals information that would otherwise be unknown to the user about any blocked packets or connection attempts going through your system. The log will supply information from the sources IP, destination IP, MAC address, source port, destination port and more. It also indicates which packets are blocked by the firewall settings. With this information the user can better gauge what is going on with their network.

The command I used to monitor my VM log in real time was:
sudo tail -f /var/log/ufw.log

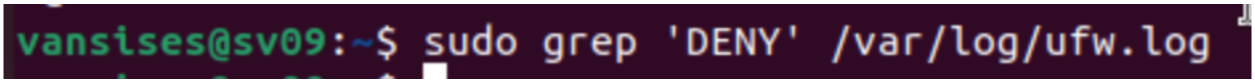


```
vansises@sv09:~$ sudo tail -f /var/log/ufw.log
Oct  3 23:23:33 sv09 kernel: [ 1806.800995] [UFW AUDIT] IN= OUT=ens160 SRC=
192.168.227.129 DST=192.168.227.254 LEN=316 TOS=0x00 PREC=0xC0 TTL=64 ID=12
239 DF PROTO=UDP SPT=68 DPT=67 LEN=296
Oct  3 23:23:33 sv09 kernel: [ 1806.801050] [UFW ALLOW] IN= OUT=ens160 SRC=
192.168.227.129 DST=192.168.227.254 LEN=316 TOS=0x00 PREC=0xC0 TTL=64 ID=12
239 DF PROTO=UDP SPT=68 DPT=67 LEN=296
Oct  3 23:23:33 sv09 kernel: [ 1806.805301] [UFW AUDIT] IN=ens160 OUT= MAC=
00:0c:29:d9:42:65:00:50:56:f6:d7:de:08:00 SRC=192.168.227.254 DST=192.168.2
27.129 LEN=328 TOS=0x10 PREC=0x00 TTL=16 ID=0 PROTO=UDP SPT=67 DPT=68 LEN=3
08
```

The purpose of this command is to:

- Watch live firewall activity as it happens (-f flag is for real time monitoring)
- Can detect patterns in attacks or network behavior quickly

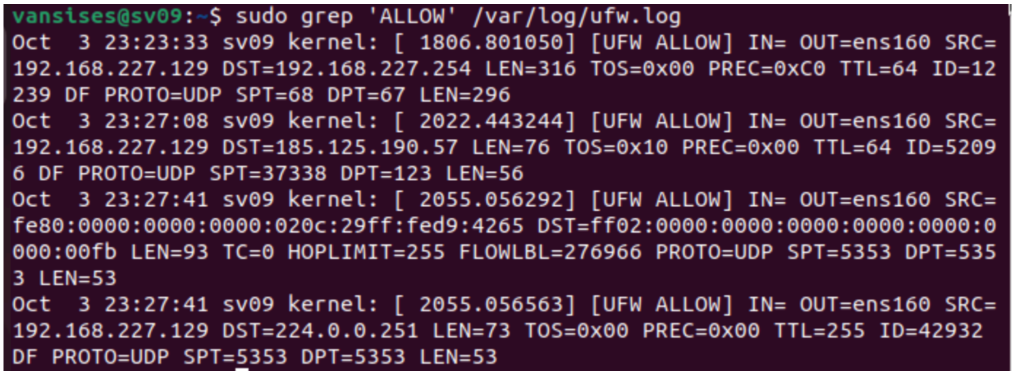
The command I used to filter denied traffic was:
sudo grep 'DENY' /var/log/ufw.log



```
vansises@sv09:~$ sudo grep 'DENY' /var/log/ufw.log
```

Nothing appeared as output because no traffic has yet been denied on my VM.

The command I used to filter allowed traffic was:
sudo grep 'ALLOW' /var/log/ufw.log



```
vansises@sv09:~$ sudo grep 'ALLOW' /var/log/ufw.log
Oct  3 23:23:33 sv09 kernel: [ 1806.801050] [UFW ALLOW] IN= OUT=ens160 SRC=
192.168.227.129 DST=192.168.227.254 LEN=316 TOS=0x00 PREC=0xC0 TTL=64 ID=12
239 DF PROTO=UDP SPT=68 DPT=67 LEN=296
Oct  3 23:27:08 sv09 kernel: [ 2022.443244] [UFW ALLOW] IN= OUT=ens160 SRC=
192.168.227.129 DST=185.125.190.57 LEN=76 TOS=0x10 PREC=0x00 TTL=64 ID=5209
6 DF PROTO=UDP SPT=37338 DPT=123 LEN=56
Oct  3 23:27:41 sv09 kernel: [ 2055.056292] [UFW ALLOW] IN= OUT=ens160 SRC=
fe80:0000:0000:0000:020c:29ff:fed9:4265 DST=ff02:0000:0000:0000:0000:0000:0
000:00fb LEN=93 TC=0 HOPLIMIT=255 FLOWLBL=276966 PROTO=UDP SPT=5353 DPT=535
3 LEN=53
Oct  3 23:27:41 sv09 kernel: [ 2055.056563] [UFW ALLOW] IN= OUT=ens160 SRC=
192.168.227.129 DST=224.0.0.251 LEN=73 TOS=0x00 PREC=0x00 TTL=255 ID=42932
DF PROTO=UDP SPT=5353 DPT=5353 LEN=53
```

Unlike the ‘DENY’ the ‘ALLOW’ has traffic flowing through as shown above.