East West University

Course title: Cyber Security, Law and Ethics
Course code: CSE487
Spring 2023
Section: 1

PROJECT-1: Securing a networked system with Public Key Infrastructure Implementing Transport Layer Security on HTTP for https:// connection

Submitted To:

Rashedul Amin Tuhin (RDA)
Senior Lecturer, Department of CSE
East West University, Dhaka

Submitted By:

Student Name	ID Number
K. M. Safin Kamal	2020-1-60-235
Mysha Maliha Priyanka	2020-1-60-230
Turna Mehzabin	2019-2-60-019

Department of Computer Science and Engineering
East West University, Dhaka

Date of Submission: 06/ 05 / 2023

Hardware:

CPU: Intel i7 13th gen 13700k unlocked

RAM: 32GB DDR5 5200MHz

GPU: RTX3060ti 8GB

Boot Storage: 1TB SSD

(This mini project can be run in lower configure PC also)

Software:

VirtualBox 7.0: Downloads – Oracle VM VirtualBox

Ubuntu 20.04: https://old-releases.ubuntu.com/releases/20.04/

Firefox 75.0: https://ftp.mozilla.org/pub/firefox/releases/75.0/linux-x86_64/en-US/

Xampp 8.0.28: Download XAMPP (apachefriends.org)

Command Lines:

1. Configuration of Certification Authority and Implementation of Transport

1. Preparing the environment

→Moving to the root using

----

sudo -i

→Creating directory:

----

mkdir -p ca/{root-ca,sub-ca,server}/{private,certs,newcerts,crl,csr}

→Changing the root of ca and sub ca private folder

----

chmod -v 700 ca/{root-ca,sub-ca,server}/private

→Creating file index in both root ca and sub ca

----

touch ca/{root-ca,sub-ca}/index

→Generating hexadecimal random number of 16 charecter

----

openssl rand -hex 16

→writing serial number of root ca

---

openssl rand -hex 16 > ca/root-ca/serial

→writing serial number of sub ca

---

openssl rand -hex 16 > ca/sub-ca/serial

→moving to ca directory

---

cd ca

2. Generating private key for root ca, sub ca and server

→Public key for rootCA

---

openssl genrsa -aes256 -out root-ca/private/ca.key 4096

→Public key for subCA

---

openssl genrsa -aes256 -out sub-ca/private/sub-ca.key 4096

→Public key for server

---

openssl genrsa -out server/private/server.key 2048

3. Generating certificates

Root-CA,

→Creating root ca.config

---

gedit root-ca/root-ca.conf

→Moving inside root-ca

---

cd root-ca

→Generating root ca certificate

---

openssl req -config root-ca.conf -key private/ca.key -new -x509 -days 7305 -sha256 -extensions v3_ca -out certs/ca.crt

→Ensuring that the certificate has been created properly

---

openssl x509 -noout -in certs/ca.crt -text

```
MIIGBTCCA+2gAwIBAgIQPbT5PREka0H5B4iHVcYnSzANBgkqhkiG9w0BAQsFADB9
MQswCQYDVQQGEwJCRDEOMAwGA1UECAwFRGhha2ExDDAKBgNVBAoMA0VXVTEXMBUG
A1UECwwOQ3liZXJfU2VjdXJpdHkxEjAQBgNVBAMMCUFjbWVTdWJDQTEjMCEGCSqG
SIb3DQEJARYUc2FmaW5AYWNtZXN1Yl9jYS5jb20wHhcNMjMwNTA0MTYwODM3WhcN
MjQwNTAzMTYwODM3WjCBjjELMAkGA1UEBhMCQkQxDjAMBgNVBAgMBURoYWthMREw
DwYDVQQHDAhCYW5hc3JlZTEMMAoGA1UECgwDRVdVMRcwFQYDVQQLDA5DeWJlcl9T
ZWN1cml0eTEUMBIGA1UEAwwLd3d3Lnh5ei5jb20xHzAdBgkqhkiG9w0BCQEWEHNh
ZmluQHNlcnZlci5jb20wggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQDm
kuWj8eufp+g7Ew9W68bw+/rip78jmDacNWmefozvk8R713nuozprWhZm7h7XzpgF
ieHTxvSdLVTLWK6XcOTy4ceCNYsM8bMuY7UE+34Pii2iywq0gKcSJD12j+BsFMKx
jzfVz7Iz4WpC3vFLxEN9tGZvjHxYhrDeUDiOIVmo7rOFcXHoPihF51vzeFs3OjL9
WcKS2ShZ5CBvsuyNNYiIKZk7zvmdieUrlZtQ6fjlrovCQaQI122k++lxJW2HB6tg
4v5K5i3zyldk+tnebCqmhGdMF8dRkpKXf7lB8vZV6co34Nr8jZi2ZTDkgOPTdNMr
wSSnYWq9VXcl6RhukXIlAgMBAAGjggFtMIIBaTAJBgNVHRMEAjAAMBEGCWCGSAGG
+EIBAQQEAwIGQDAzBglghkgBhvhCAQ0EJhYkT3BlblNTTCBHZW5lcmF0ZWQgU2Vy
dmVyIENlcnRpZmljYXRlMB0GA1UdDgQWBBSmpeKhWpVWMDSWbApP43vdX8ewAzCB
zwYDVR0jBIHHMIHEgBT7l3M33nEmzLNLXxEPim69BeJMDqGBmKSBlTCBkjELMAkG
A1UEBhMCQkQxDjAMBgNVBAgMBURoYWthMREwDwYDVQQHDAhCYW5hc3JlZTEMMAoG
A1UECgwDRVdVMRcwFQYDVQQLDA5DeWJlcl9TZWN1cml0eTETMBEGA1UEAwwKQWNt
ZVJvb3RDQTEkMCIGCSqGSIb3DQEJARYVc2FmaW5AYWNtZXJvb3RfY2EuY29tghEA
qoOSC6qtQldu3wWpRzPNizAOBgNVHQ8BAf8EBAMCBaAwEwYDVR0lBAwwCgYIKwYB
BQUHAwEwDQYJKoZIhvcNAQELBQADggIBABuP6EcsWg252Unn87WjZiCy3EuALlqF
gqCWcVmZkv6/UO4bzbQ81uP9MXzHcChVGMmWxBg9koh30vxt5ytA+DryBfH18Myn
pFzSGoVmyyK1zuUj6UkA1Zzvqpoc1GppKpJUc5V/cmOsU2wr2l6Yss6bbYiG5f5S
I8/8zESzAmtxUTZX8SF68NlnSCyZVRaSQaKNTH2ld13Ef1tfertD8eOLHvcq3w0p
Gj+azEi3MuEyzTRXLKg1Iof93JmdVwwUEaKM8XSsCGA/gekP6DgCMJFUBuaS7Dvw
loS+cKdYOSLCJaHJ9Q1A87jllppEFftR4e8+fldpn8I7prBRXRXoPpuKJO2Eh766
8Vf9jcmoqtYDAt1QGhH7XKuGcwQORB+9vaGmje9Pq73AV4JwJYbtxuP8mUcGwyaV
mQ4HeIzCA4aQIzpCIeA5nkwAf2EiHSSKCqPpF9oEgHlDzufqSP7GQqb6M66XDbQR
FLCKELsgiPzE0NaTKUgXFz9leZ+py2EIgPMY69Uv0tSNEbLHKnYbKg6XqzXEVYcT
hzPGTeqb0w5V+pBzMiFcS6T+Xf3DkgnBMLx9O5DLViWuB8xb/Tp4lFsXjcota22T
Z/XQKxU8rzoAmwMIDcdWWwpq0DKBTPWid6sHz37N4BBMsqkG6PJiToURtiKQtp1D
BM2gq0P1TwHe
-----END CERTIFICATE-----
```


→Moving a step back and then to sub-ca

----

cd ../sub-ca

Sub-CA

→Creating sub-ca.config

----

gedit sub-ca.conf

→Requesting for sub ca certificate signing request.

----

openssl req -config sub-ca.conf -new -key private/sub-ca.key -sha256 -out csr/sub-ca.csr

→moving to the previous folder

----

cd -

→Signing the request of sub ca by root ca

----

openssl ca -config root-ca.conf -extensions v3_intermediate_ca -days 3652 -notext -in ../sub-ca/csr/sub-ca.csr -out ../sub-ca/certs/sub-ca.crt

to confirm insert "y"

→.pem file has been generated, See the signing

---

cat index

→Root ca signed sub ca, Seeing detail

---

openssl x509 -noout -text -in ../sub-ca/certs/sub-ca.crt


```
-----BEGIN CERTIFICATE-----
MIIGATCCA+mgAwIBAgIRAKqDkguqrUJXbt8FqUczzYswDQYJKoZIhvcNAQELBQAw
gZIxCzAJBgNVBAYTAkJEMQ4wDAYDVQQIDAVEaGFrYTERMA8GA1UEBwwIQmFuYXNy
ZWUxDDAKBgNVBAoMA0VXVTEXMBUGA1UECwwOQ3liZXJfU2VjdXJpdHkxEzARBgNV
BAMMCkFjbWVSb290Q0ExJDAiBgkqhkiG9w0BCQEWFXNhZmluQGFjbWVyb290X2Nh
LmNvbTAeFw0yMzAzMTgxMzM1MTdaFw0zMzAzMTcxMzM1MTdaMH0xCzAJBgNVBAYT
AkJEMQ4wDAYDVQQIDAVEaGFrYTEMMAoGA1UECgwDRVdVMRcwFQYDVQQLDA5DeWJl
cl9TZWN1cml0eTESMBAGA1UEAwwJQWNtZVN1YkNBMSMwIQYJKoZIhvcNAQkBFhRz
YWZpbkBhY21lc3ViX2NhLmNvbTCCAiIwDQYJKoZIhvcNAQEBBQADggIPADCCAgoC
ggIBAKtrCjbiph5GEf6CN9I9wdkf6+E4k9fD4vkQfY1jA4OTdH4jiuCuGjml1aEY
95hSjz5cqn2L+G+kaaJxzUZ4e40UYEBjLQ3akETlNst00N1rvQO7DgDhojbaLMr6
HcfMkPYXighQ50e5mPwsXMqW4Tx2AXzkyGJg5QINWn8xuKnl7Cpu6hwIFr5tCOMW
juFUnekHPEPnslo5O5U1rO+4wpET1pgoKbb8CYXbaw/ivjKmo9bVSXWmL1gbe4iH
H+FxH2ppYG8nD1z5PegqAm4swuNWg16luvLsa7/7G9fXurqal+enqCxzG1oRmraV
hOHdTOD7EnXtqahIv3OJwII3vQBtd9y35fp/bOYaobSXYT2hgrTnRHlHlxqFtr9h
1wso9oqsJjH+XJYnpbgqlO6ayc8A375J/PVxx2nidL7ksBAQGEuHFOH/r9W267KU
L9iHb5wsdacb+Pf1nZdAwqW6nax39BHtrAKVM7uV9z0jnxN7ZNobMgLWx10WrZFb
nL8t/6wUM5JEmu8+1bc5M/Fmlla0pnzF1oCszVFibSXHimXll8N9Eg2GaFsOJ2S2
yuI0rd5nDZbJr+xj5ZoAbdH0c5f80n8BWb8wYjtBDAbq2tuLYN4w9DYyYi1ne4XA
7QV4FkBPJ9CUdZS3FF9ivqg4ubRnbIyok7sW4lePF5+f2Og9AgMBAAGjZjBkMB0G
A1UdDgQWBBT7l3M33nEmzLNLXxEPim69BeJMDjAfBgNVHSMEGDAWgBTpvHZk39qx
FcmtCstSZ/gpl+eEdTASBgNVHRMBAf8ECDAGAQH/AgEAMA4GA1UdDwEB/wQEAwIB
hjANBgkqhkiG9w0BAQsFAAOCAgEASehpgYX4f6ExsFS/O1TEmlLedKWDjno9d8up
sOjye/HozW0oheEnzJoKEr+fsWj08bmCM1wX387eTtrh5lXeiirtvjjH4xZifMSI
uSjU2GzWiTqoUU5jjuKR5mWivixb5C+eI+C2kineHglFGmwlLghOou7lcalPOQ9n
eT1QUdlVq1vsDInDWjQAJJauH/REUT5/EH+8aV27UT5MkhKdZJBGtwq/Do0Nqssb
y55dcNlwzVuYZJ/NS6PNPQRNITae26aj8XK2lHai++THcfcDwaJQSIKGsxSbjDym
0nGYWwb7E94u8ocfvzzPg9PIJLBoyRJna5LkoZ0j8/N1yzFZbsJvltGWqoZLNegc
UtVilBjEHhks0JKeSn5FRHzB3W8qgmELfWECcJ857ZFvf08agBykNU1E9c0/BSJD
6fVY7L896pI8bmX2VvveAMVA1C73xADHX65HN8nytyICGjKMAFiB3tGgsXCVnF4O
Wn5wCuAxHq9zzfmV/ja1CvhM02hz3YWABuhi/aIs9jzl6UjNvHXi5fJnFJ+LkVJ7
Q0sOEx72+j350OeX5A9QXL/eI7NIbLZrHR632uWkcfCWR5fJGpTEaNYEp93J9ooD
3Te5s25ubLT8RxdE/lVa6mtWXHz/QuD0/QmZ29vC5wiU/SLKjlwsiw4H2DL3N7ga
luEKcxI=
-----END CERTIFICATE-----
```


4. Configuring server

→Moving to server

---

cd ../server

→Generating certificate signing request from server

---

openssl req -key private/server.key -new -sha256 -out csr/server.csr

→moving to sub ca to sign the server's certificate

---

cd ../sub-ca

→Sub ca signing certificate request of server

---

openssl ca -config sub-ca.conf -extensions server_cert -days 365 -notext -in ../server/csr/server.csr -out ../server/certs/server.crt

```
root@server1:~/ca/server# openssl req -key private/server.key -new -sha256 -out
csr/server.csr
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]:BD
State or Province Name (full name) [Some-State]:Dhaka
Locality Name (eg, city) []:Banasree
Organization Name (eg, company) [Internet Widgits Pty Ltd]:EWU
Organizational Unit Name (eg, section) []:Cyber_Security
Common Name (e.g. server FQDN or YOUR name) []:www.safemyturn.com
```

→seeing detail

----

cat index

→moving to certs folder to see certificate of server

----

cd ../server/certs/

→See the directory by using command:

----

ls

→Now, concating sub-ca.crt, ca.crt and server.crt and naming the new file chained.crt

----

cat server.crt ../../sub-ca/certs/sub-ca.crt ../../root-ca/certs/ca.crt > chained.crt

→moving back to server directory

----

cd ..

echo "127.0.0.2 www. safemyturn.com" >> /etc/hosts

ping www. safemyturn.com

```
server1@server1: ~
server1@server1:~$ ping www.xyz.com
PING www.xyz.com (192.168.0.104) 56(84) bytes of data.
64 bytes from server1.xyz.com (192.168.0.104): icmp_seq=1 ttl=64 time=0.008 ms
64 bytes from server1.xyz.com (192.168.0.104): icmp_seq=2 ttl=64 time=0.018 ms
64 bytes from server1.xyz.com (192.168.0.104): icmp_seq=3 ttl=64 time=0.020 ms
64 bytes from server1.xyz.com (192.168.0.104): icmp_seq=4 ttl=64 time=0.019 ms
64 bytes from server1.xyz.com (192.168.0.104): icmp_seq=5 ttl=64 time=0.028 ms
64 bytes from server1.xyz.com (192.168.0.104): icmp_seq=6 ttl=64 time=0.022 ms
64 bytes from server1.xyz.com (192.168.0.104): icmp_seq=7 ttl=64 time=0.023 ms
64 bytes from server1.xyz.com (192.168.0.104): icmp_seq=8 ttl=64 time=0.020 ms
64 bytes from server1.xyz.com (192.168.0.104): icmp_seq=9 ttl=64 time=0.019 ms
S64 bytes from server1.xyz.com (192.168.0.104): icmp_seq=10 ttl=64 time=0.023 ms
64 bytes from server1.xyz.com (192.168.0.104): icmp_seq=11 ttl=64 time=0.029 ms
64 bytes from server1.xyz.com (192.168.0.104): icmp_seq=12 ttl=64 time=0.025 ms
64 bytes from server1.xyz.com (192.168.0.104): icmp_seq=13 ttl=64 time=0.020 ms
```

→Turning on the ssl port
----
openssl s_server -accept 443 -www -key private/server.key -cert certs/server.crt -CAfile ../sub-ca/certs/sub-ca.crt

→Opening new terminal
----
sudo -i

→See the port number used by different Ip addresses
----
ss -ntl
sudo apt update
sudo apt install curl

→copying the certificate to ca certificate folder
----
cp ca/root-ca/certs/ca.crt /usr/local/share/ca-certificates/

→Updating ca certificate folder
----
update-ca-certificates -v

→Open new terminal
----
sudo -i
tree ca

→Copy .pem files to home for future use
----
cp /root/ca/root-ca/newcerts/21DE5190AF587104493F1750892E9B86.pem ~server/
cp /root/ca/sub-ca/newcerts/ACB9E41C001BD6E31714199EF459CA4C.pem ~server/

→Copy .crt files to certificate folder in home for future use
----
cp /root/ca/root-ca/certs/ca.crt /home/server/certificate
cp /root/ca/sub-ca/certs/sub-ca.crt /home/server/certificate/
cp /root/ca/server/certs/chained.crt /home/server/certificate/
cp /root/ca/server/certs/server.crt /home/server/certificate/
cp /root/ca/server/private/server.key /home/server/certificate/

→Next go to this location using new terminal
----
sudo -i
cd /opt/lampp/etc/extra
chmod 777 httpd-ssl.conf
gedit httpd-ssl.conf

→In line 106 replace this line

SSLCertificateFile "/home/server/certificate/server.crt"

→In line 116 replace this line

SSLCertificateKeyFile "/home/server/certificate/server.key"

→In line 136 replace this line

SSLCACertificatePath "/home/server/certificate"

→For auto redirect to https place this after line 98

<VirtualHost _default_:80>
ServerName www.example.com:80
ServerAdmin you@example.com
Redirect permanent / https://www.safemyturn.com
</VirtualHost>

→Remove all file from htdocs

----

sudo -i
cd /opt/lampp/htdocs
ls
rm -r dashboard img webalizer
rm applications.html bitnami.css index.php

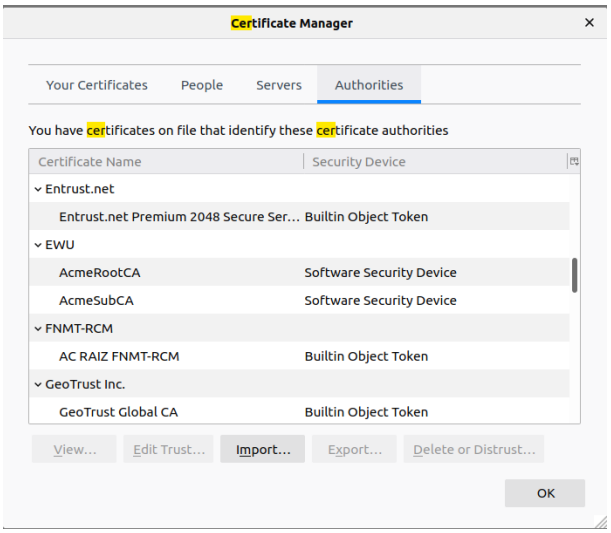
→Now make a html file and write some html code

----

touch index.html
gedit index.html

save and exit

Now on the browser
Settings → privacy and security → view certificate → authorities → import → select the file →open → select purpose → {view: to see the certificate} → OK



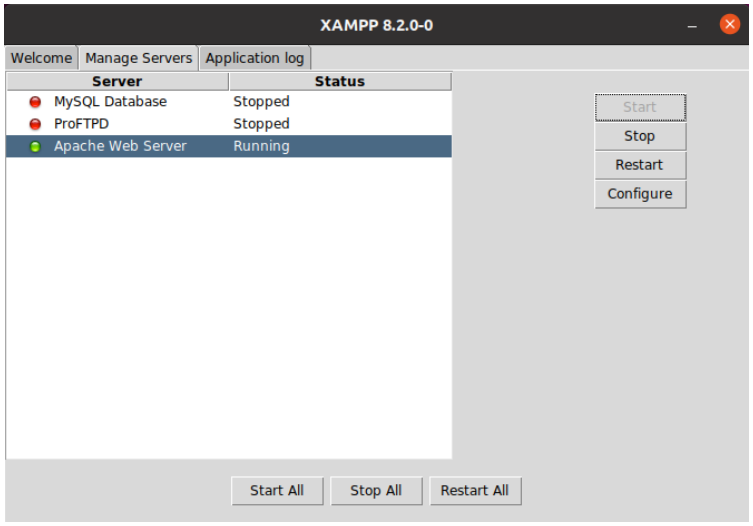


Figure: website with lock

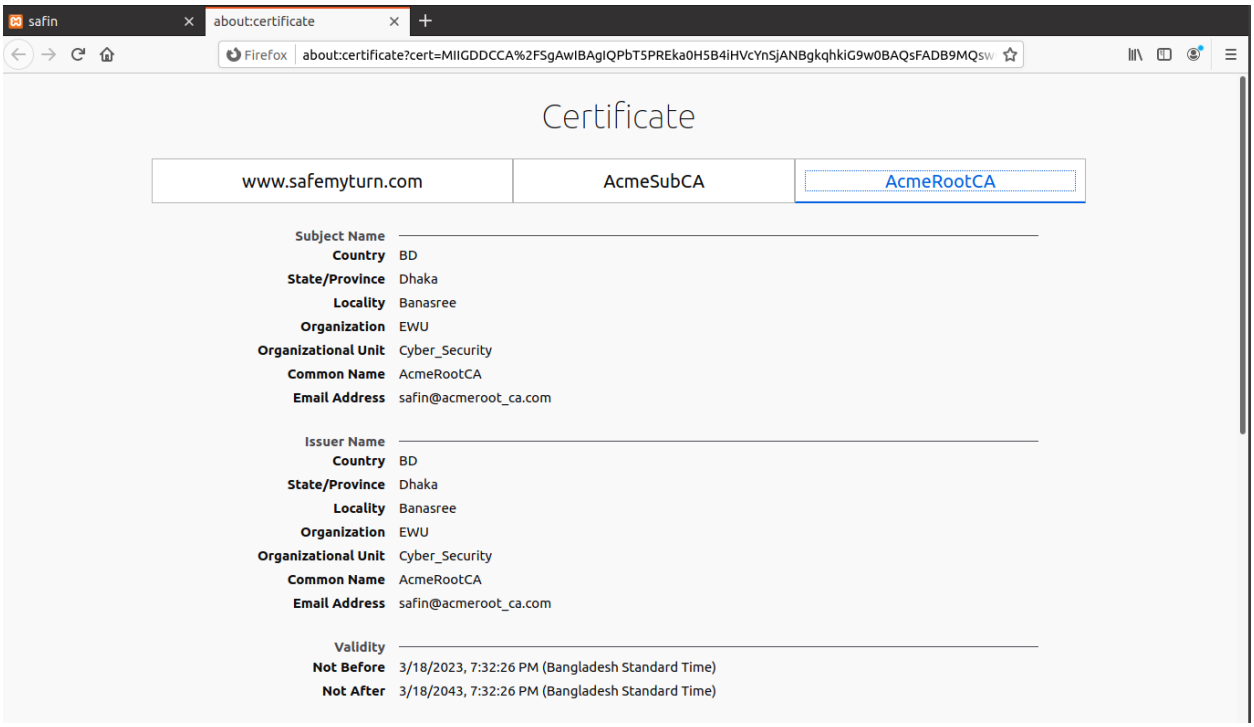


Figure: certificates

2. DNS configuration

ip addr **//To get device ip**
ip route **//To get default gateway ip**
sudo -i
sudo apt install bind9
named -v

cd /etc/bind
ls
hostnamectl status
gedit /etc/hosts

[After the command edit next]

*192.168.0.104 server.safemyturn.com server***//this is your ip which you get from ip addr command**

[save and exit]

---

hostname
dnsdomainname
hostname --fqdn

cp named.conf.options named.conf.options.orig
gedit named.conf.options

[After the command edit next]

//==========

```
        dnssec-validation auto;
        listen-on-v6 { any; };
        recursion yes;
        listen-on{192.168.0.104;};
        allow-transfer {none;};

        forwarders {
        192.168.0.1;

        };
```

[save and exit]

---

cp named.conf.local named.conf.local.orig
gedit named.conf.local

[After the command edit next]

```
//forward lookup zone
zone "safemyturn.com" IN{
        type master;
        file "/etc/bind/db.safemyturn.com";
};

//reverse lookup zone
zone "0.168.192.in-addr.arpa" IN {
        type master;
        file "/etc/bind/db.0.168.192";
};
```

[save and exit]

---

named-checkconf
ls
cat named.conf.local
cp db.local db.genibarta.com
gedit db.genibarta.com

[Replace full file with that text]

```
;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     ns1.safemyturn.com. root.safemyturn.com. (
                              2            ; Serial
                         604800            ; Refresh
                          86400            ; Retry
                        2419200            ; Expire
                         604800 ) ; Negative Cache TTL
;
@       IN      NS      ns1.safemyturn.com.
ns1     IN      A       192.168.0.104
www     IN      A       192.168.0.104
ftp     IN      A       192.168.0.104
@       IN      MX      10      mail
mail    IN      A       192.168.0.104
@       IN      AAAA    ::1
```

[Save and exit]

```
named-checkzone genibarta.com db.genibarta.com
cp db.127 db.0.168.192
gedit db.0.168.192
```

[Replace full file with that text]

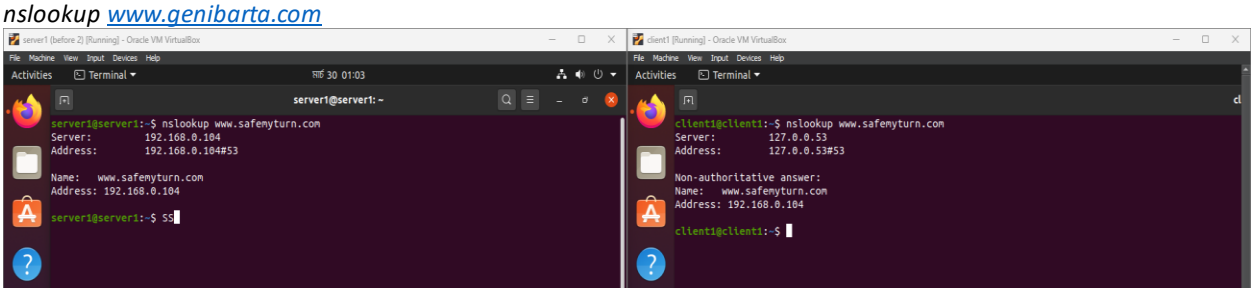
```
;
; BIND reverse data file for local loopback interface
;
$TTL    604800
@       IN      SOA     ns1. safemyturn.com. root. safemyturn.com. (
                              1            ; Serial
                         604800            ; Refresh
                          86400            ; Retry
                        2419200            ; Expire
                         604800 ) ; Negative Cache TTL
;
@       IN      NS      ns1. safemyturn.com.
24      IN      PTR     ns1. safemyturn.com.
24      IN      PTR     www. safemyturn.com.
24      IN      PTR     ftp. safemyturn.com.
24      IN      PTR     mail. safemyturn.com.
```

[Save and exit]

```
named-checkzone 0.168.192.in-addr.arpa db.0.168.192
named-checkconf

service bind9 restart
service bind9 status
```

nslookup www.genibarta.com



If wrong IP is showing

----

cat /etc/resolv.conf
rm /etc/resolv.conf
ln -sf /run/systemd/resolve/resolv.conf /etc/resolv.conf
gedit /etc/resolv.conf

[Replace last line with that text]

nameserver 192.168.0.20
nameserver 192.168.0.1
search localdomain

[Save and exit]

Write the IP of the server in the client PC DNS meanu.

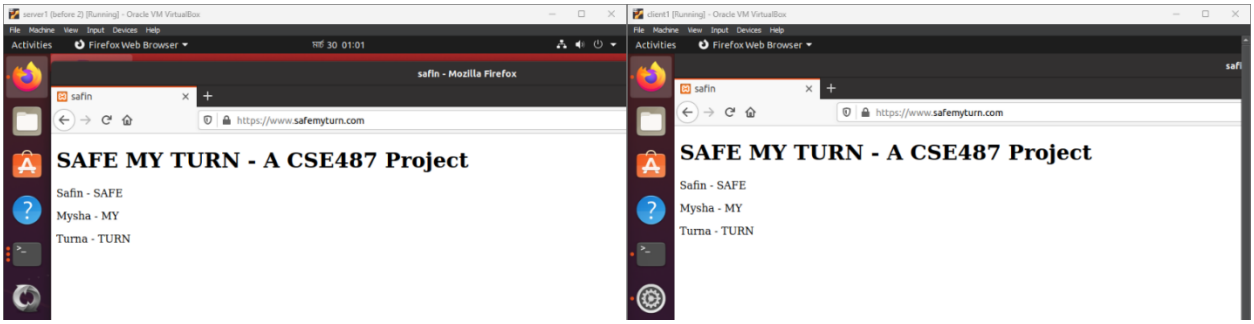


Figure: From client PC

Firewall configuration

→Install ufw package

----

sudo apt install ufw

→Set default rules for ufw firewall

----

ufw default allow outgoing
ufw default deny incoming

→Enable ufw

----

ufw enable

→Allow port 80 (http), 443(https), and 53(DNS)

----

ufw allow 80
ufw allow 443
ufw allow 53

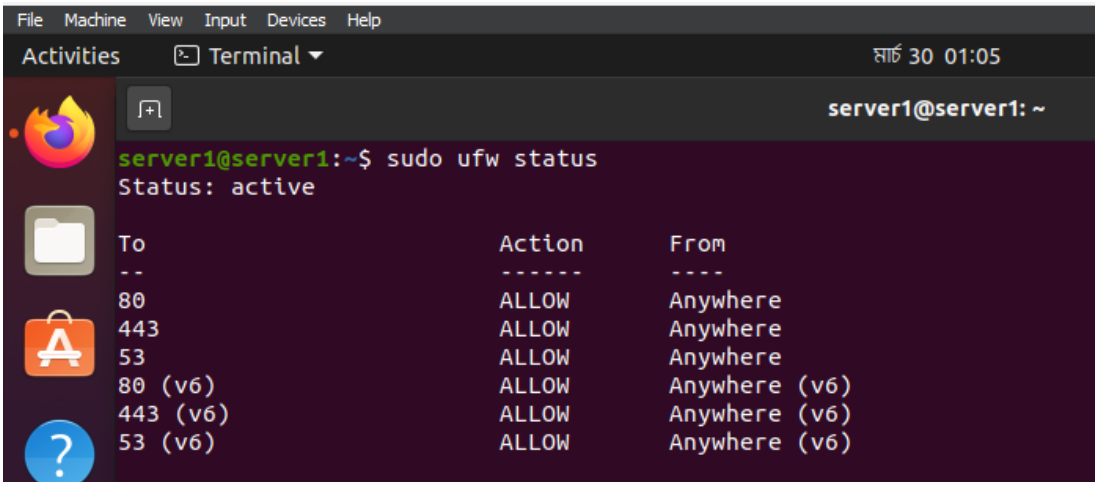


Figure: only 80,443,53 ports are allowed

DOS ATTACK:

In server

→To install snort tool

Sudo apt-get install snort -y

→To start snort

Sudo snort -A console -c /etc/snort/snort.conf

→IDS configure

----

cd /etc/snort
cp snort.conf test_snort.conf
sudo gedit test_snort.conf

→Then go to line 51 and under "ipvar HOME_NET any" write ip var HOME_NET your host ip ipvar HOME_NET 192.168.0.104/24

cd /etc/snort/rules
sudo gedit local.rules

→Write the following line and save exit

----

alert tcp any any -> $HOME_NET any (flags:S; msg:"DoS attack happening"; flow:stateless; detection_filter:track by_dst,count 70,Seconds 10; sid:1000001; rev:1;)

→Validate the conf file

----

sudo snort -T -c /etc/snort/test_snort.conf -i enp0s3

→*S*tart snort

----

sudo snort -A console -q -i enp0s3 -c /etc/snort/test_snort.conf

In attack client

→To install hping3 tool

Apt install hping3 -y

→To start attack

Sudo hping3 -S –flood -V -p 443 192.168.0.104

NOW START ATTACK FROM CLIENT TO SERVER

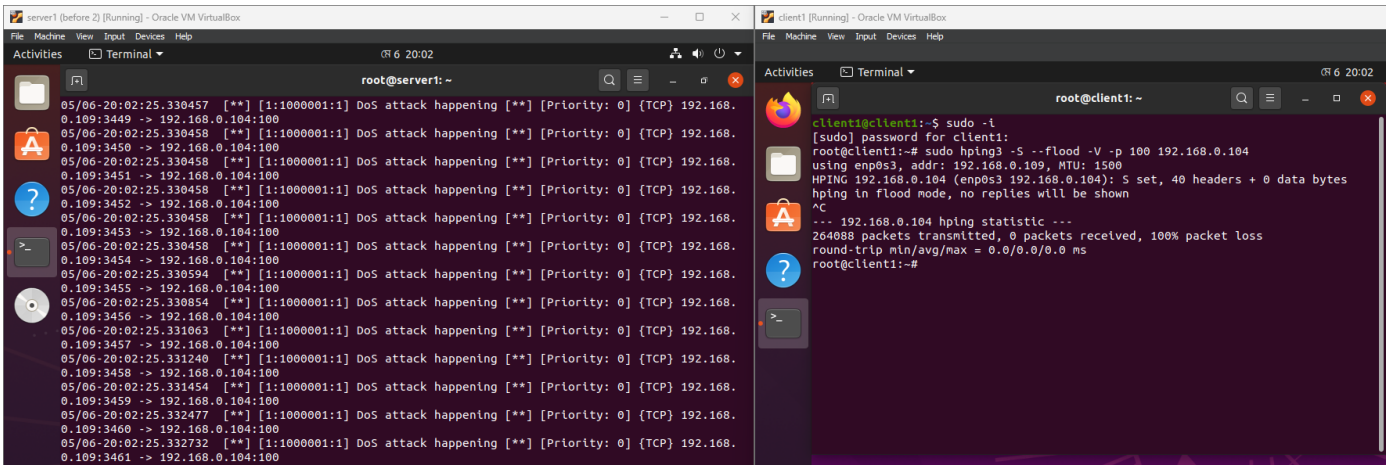


Figure: attacking from client to server