



Project Report

Course Title: **Cyber Security, Law, and Ethics**

Course Code: **CSE487**

Section: **2**

Group: **12**

Semester: **Spring 2024**

Course Instructor: Dr. Md. Hasanul Ferdaus

Project name:

**Securing a networked system with Public Key
Infrastructure**

Group Information

Name	ID
Fahima Afrin Nidha	2020-2-60-175
Md. Tanvir Islam Mahin	2020-2-60-075
Md. Abu Salha Akram Rayhan	2020-1-60-020
Sifat Khan Shishir	2020-2-60-120

Configuration of Certification Authority AcmeCA with AcmeRootCA as the RootCA:

```
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 character

```
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

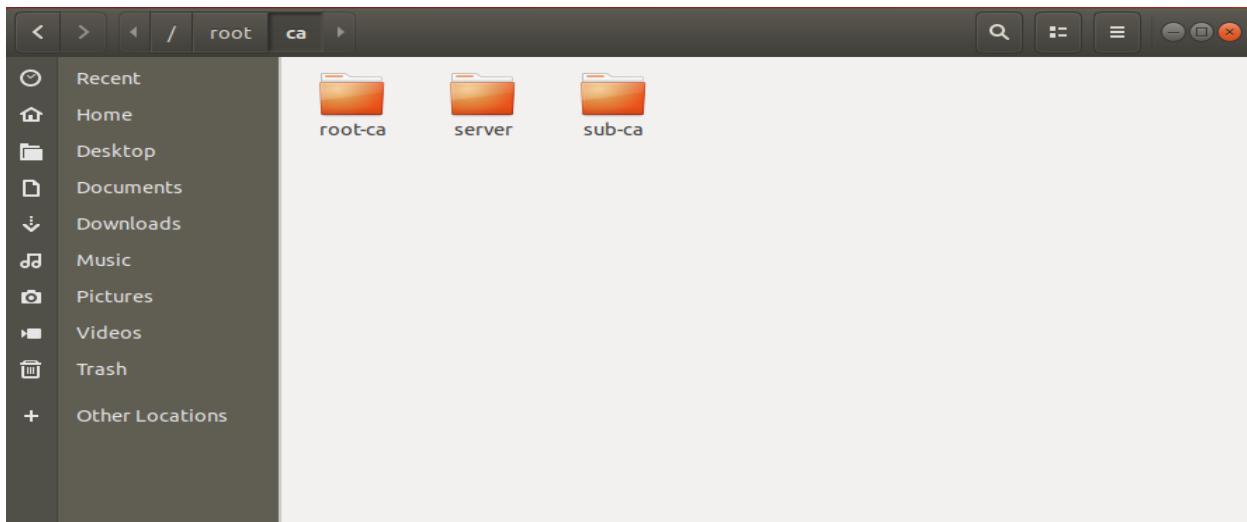
```
openssl genrsa -aes256 -out root-ca/private/ca.key 4096
```

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
```

```
[ca]
```

```
#/root/ca/root-ca/root-ca.conf
```

```
#see man ca
```

```
default_ca = CA_default
```

```
[CA_default]
```

```
dir = /root/ca/root-ca
```

```
certs    = $dir/certs

crl_dir  = $dir/crl

new_certs_dir = $dir/newcerts

database  = $dir/index

serial    = $dir/serial

RANDFILE  = $dir/private/.rand

private_key = $dir/private/ca.key

certificate = $dir/certs/ca.crt

crlnumber = $dir/crlnumber

crl     = $dir/crl/ca.crl

crl_extensions = crl_ext

default_crl_days = 30
```

default_md = sha256

name_opt = ca_default

cert_opt = ca_default

default_days = 365

preserve = no

policy = policy_strict

[policy_strict]

countryName = supplied

stateOrProvinceName = supplied

organizationName = match

organizationalUnitName = optional

commonName = supplied

emailAddress = optional

[policy_loose]

countryName = optional

stateOrProvinceName = optional

localityName = optional

organizationName = optional

organizationalUnitName = optional

commonName = supplied

emailAddress = optional

[req]

Options for the req tool, man req.

```
default_bits = 2048
```

```
distinguished_name = req_distinguished_name
```

```
string_mask = utf8only
```

```
default_md = sha256
```

```
# Extension to add when the -x509 option is used.
```

```
x509_extensions = v3_ca
```

```
[ req_distinguished_name ]
```

```
countryName = Country Name (2 letter code)
```

```
stateOrProvinceName = State or Province Name
```

```
localityName = Locality Name
```

```
0.organizationName = Organization Name
```

```
organizationalUnitName = Organizational Unit Name
```

```
commonName = Common Name
```

emailAddress = Email Address

countryName_default = BD

stateOrProvinceName_default = Dhaka

localityName_default = Aftabnagar

0.organizationName_default = EWU

organizationalUnitName_default = Cyber_Security

commonName_default = AcmeRootCA

emailAddress_default = cyber@securityroot_ca.com

[v3_ca]

Extensions to apply when creating root ca

Extensions for a typical CA, see x509v3_config

subjectKeyIdentifier = hash

authorityKeyIdentifier = keyid(always,issuer)

```
basicConstraints = critical, CA:true

keyUsage = critical, digitalSignature, cRLSign, keyCertSign

[v3_intermediate_ca]

# Extensions to apply when creating intermediate or sub-ca

# Extensions for a typical intermediate CA, same man as above

subjectKeyIdentifier = hash

authorityKeyIdentifier = keyid:always,issuer

#pathlen:0 ensures no more sub-ca can be created below an intermediate

basicConstraints = critical, CA:true, pathlen:0

keyUsage = critical, digitalSignature, cRLSign, keyCertSign

[ server_cert ]

# Extensions for server certificates
```

```
basicConstraints = CA:FALSE
```

```
nsCertType = server
```

```
nsComment = "OpenSSL Generated Server Certificate"
```

```
subjectKeyIdentifier = hash
```

```
authorityKeyIdentifier = keyid,issuer:always
```

```
keyUsage = critical, digitalSignature, keyEncipherment
```

```
extendedKeyUsage = serverAuth
```

```
organizationalUnitName      = Organizational Unit Name
commonName                  = Common Name
emailAddress                = Email Address
countryName_default = BD
stateOrProvinceName_default = Dhaka
localityName_default = Aftabnagar
o.organizationName_default = EWU
organizationalUnitName_default = Cyber_Security
commonName_default = AcmeRootCA
emailAddress_default = cyber@securityroot_ca.com
```

```
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
```

```
cd ..sub-ca
```

Sub-CA

Creating sub-ca.config

```
gedit sub-ca.conf
```

[ca]

```
#/root/ca/sub-ca/sub-ca.conf
```

#see man ca

```
default_ca = CA_default
```

[CA_default]

```
dir = /root/ca/sub-ca
```

```
certs = $dir/certs
```

crl_dir = \$dir/crl

new_certs_dir = \$dir/newcerts

database = \$dir/index

serial = \$dir/serial

RANDFILE = \$dir/private/.rand

private_key = \$dir/private/sub-ca.key

certificate = \$dir/certs/sub-ca.crt

crlnumber = \$dir/crlnumber

crl = \$dir/crl/ca.crl

crl_extensions = crl_ext

default_crl_days = 30

default_md = sha256

name_opt = ca_default

cert_opt = ca_default

default_days = 365

preserve = no

policy = policy_loose

[policy_strict]

countryName = supplied

stateOrProvinceName = supplied

organizationName = match

organizationalUnitName = optional

commonName = supplied

emailAddress = optional

[policy_loose]

countryName = optional

stateOrProvinceName = optional

localityName = optional

organizationName = optional

organizationalUnitName = optional

commonName = supplied

emailAddress = optional

[req]

Options for the req tool, man req.

default_bits = 2048

```
distinguished_name = req_distinguished_name
```

```
string_mask = utf8only
```

```
default_md = sha256
```

```
# Extension to add when the -x509 option is used.
```

```
x509_extensions = v3_ca
```

```
[ req_distinguished_name ]
```

```
countryName = Country Name (2 letter code)
```

```
stateOrProvinceName = State or Province Name
```

```
localityName = Locality Name
```

```
0.organizationName = Organization Name
```

```
organizationalUnitName = Organizational Unit Name
```

```
commonName = Common Name
```

```
emailAddress = Email Address
```

countryName_default = BD

stateOrProvinceName_default = Dhaka

localityName_default = Aftabnagar

0.organizationName_default = EWU

organizationalUnitName_default = Cyber_Security

commonName_default = AcmeCA

emailAddress_default = cyber@securitysub_ca.com

[v3_ca]

Extensions to apply when creating root ca

Extensions for a typical CA, man x509v3_config

subjectKeyIdentifier = hash

authorityKeyIdentifier = keyid:always,issuer

basicConstraints = critical, CA:true

keyUsage = critical, digitalSignature, cRLSign, keyCertSign

```
[ v3_intermediate_ca ]  
  
# Extensions to apply when creating intermediate or sub-ca  
  
# Extensions for a typical intermediate CA, same man as above  
  
subjectKeyIdentifier = hash  
  
authorityKeyIdentifier = keyid:always,issuer  
  
#pathlen:0 ensures no more sub-ca can be created below an intermediate  
  
basicConstraints = critical, CA:true, pathlen:0  
  
keyUsage = critical, digitalSignature, cRLSign, keyCertSign  
  
[ server_cert ]  
  
# Extensions for server certificates  
  
basicConstraints = CA:FALSE  
  
nsCertType = server
```

```
nsComment = "OpenSSL Generated Server Certificate"
```

```
subjectKeyIdentifier = hash
```

```
authorityKeyIdentifier = keyid,issuer:always
```

```
keyUsage = critical, digitalSignature, keyEncipherment
```

```
extendedKeyUsage = serverAuth
```

```
countryName_default = BD
stateOrProvinceName_default = Dhaka
localityName_default = Aftabnagar
0.organizationName_default = EWU
organizationalUnitName_default = Cyber_Security
commonName_default = AcmeCA
emailAddress_default = cyber@securitysub_ca.com
```

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
```

Signing the request of sub ca by root ca

```
cd ..
```

```
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
```

Proof of certificate and pem file creation

```

root@naf-VirtualBox:~/ca/sub-ca# ls
certs  csr      index.attr  newcerts  serial      sub-ca.conf
crl    index  index.old  private   serial.old
root@naf-VirtualBox:~/ca/sub-ca# cd certs
root@naf-VirtualBox:~/ca/sub-ca/certs# ls
sub-ca.crt
root@naf-VirtualBox:~/ca/sub-ca/certs# cd ..
root@naf-VirtualBox:~/ca/sub-ca# cd newcerts
root@naf-VirtualBox:~/ca/sub-ca/newcerts# ls
8149874F6A19B1E4811F63FE9BF4DE71.pem

```

- Transferring the certificate from CA to

www.verysecureserver.com:

- Installation of the signed SSL certificate in the server of www.cybersecurity.com:

- Making the system trust RootCA:

after installation

cd /opt/lampp/etc/extra

```

104 #      require an ECC certificate which can also be configured in
105 #      parallel.
106 SSLCertificateFile "/home/naf/certificate/server.crt"
107 #SSLCertificateFile "/opt/lampp/etc/server-dsa.crt"
108 #SSLCertificateFile "/opt/lampp/etc/server-ecc.crt"
109
110 #      Server Private Key:
111 #      If the key is not combined with the certificate, use this
112 #      directive to point at the key file.  Keep in mind that if
113 #      you've both a RSA and a DSA private key you can configure
114 #      both in parallel (to also allow the use of DSA ciphers, etc.)
115 #      ECC keys, when in use, can also be configured in parallel
116 SSLCertificateKeyFile "/home/naf/certificate/server.key"
117 #SSLCertificateKeyFile "/opt/lampp/etc/server-dsa.key"
118 #SSLCertificateKeyFile "/opt/lampp/etc/server-ecc.key"
119
120 #      Server Certificate Chain:
121 #      Point SSLCertificateChainFile at a file containing the
122 #      concatenation of PEM encoded CA certificates which form the
123 #      certificate chain for the server certificate. Alternatively
124 #      the referenced file can be the same as SSLCertificateFile
125 #      when the CA certificates are directly appended to the server
126 #      certificate for convenience.
127 #SSLCertificateChainFile "/opt/lampp/etc/server-ca.crt"
128
129 #      Certificate Authority (CA):
130 #      Set the CA certificate verification path where to find CA
131 #      certificates for client authentication or alternatively one
132 #      huge file containing all of them (file must be PEM encoded)
133 #      Note: Inside SSLCACertificatePath you need hash symlinks
134 #            to point to the certificate files. Use the provided
135 #            Makefile to update the hash symlinks after changes.
136 SSLCACertificatePath "/home/naf/certificate"
137 #SSLCACertificateFile "/opt/lampp/etc/ssl.crt/ca-bundle.crt"

```

sudo gedit httpd-ssl.conf

/home/naf/certificate/server.crt

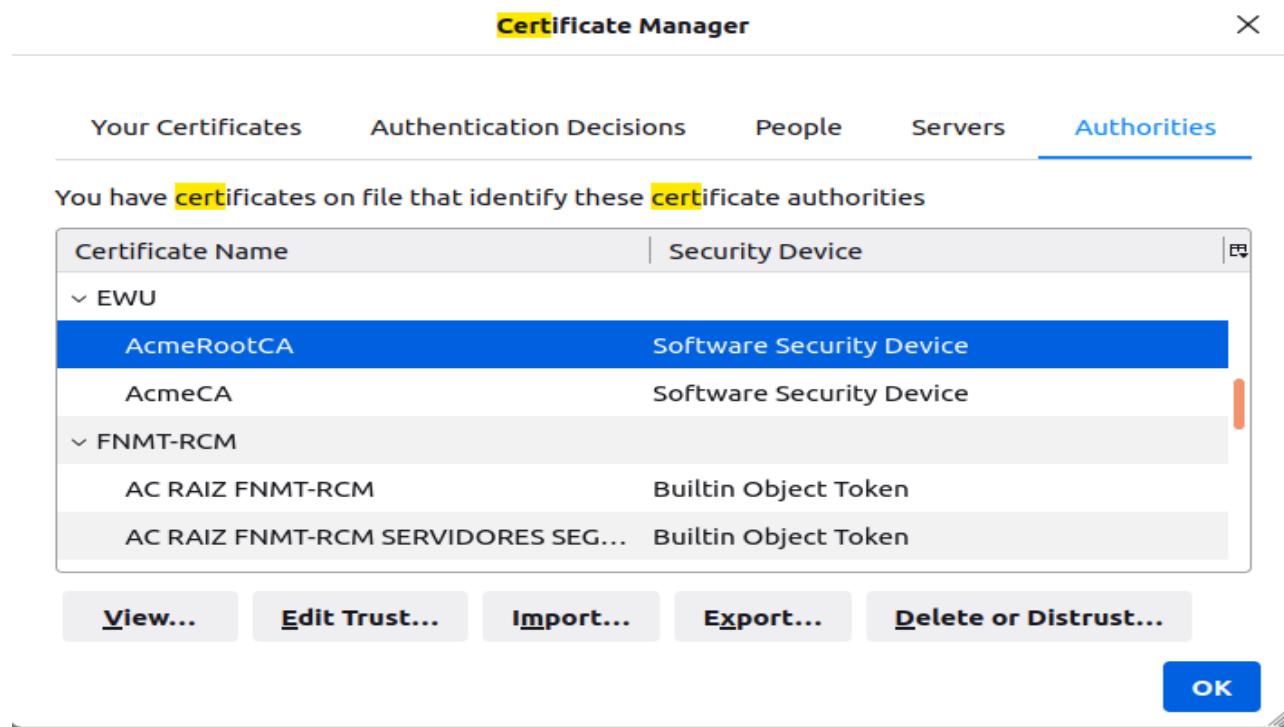
```
cd /etc
```

```
sudo gedit hosts
```

```
add website name
```

www.cybersecurity.com

Importing the certificates into mozilla firefox



The screenshot shows a Firefox browser window with the title "Document" and the tab "Certificate for www.cybersecurity.com". The address bar displays "about:certificate?cert=MIIGJDCCB...". The page content is titled "Certificate" and contains the following information:

www.cybersecurity.com		AcmeCA
Subject Name		
Country	BD	
State/Province	Dhaka	
Locality	Aftabnagar	
Organization	EWU	
Organizational Unit	Cyber_Security	
Common Name	www.cybersecurity.com	
Email Address	www.nafizkhan.com@gmail.com	
Issuer Name		
Country	BD	
State/Province	Dhaka	
Organization	EWU	
Organizational Unit	Cyber_Security	
Common Name	AcmeCA	
Email Address	cyber@securitysub_ca.com	
Validity		
Not Before	Tue, 28 Mar 2024 05:16:00 GMT	
Not After	Wed, 26 Mar 2025 05:16:00 GMT	
Public Key Info		
Algorithm	RSA	
Key Size	2048	
Exponent	65537	
Modulus	BF3B:84:C5:A4:1D:27:E6:C3:0B:76:96:E9:E6:3F:3C:87:31:C9:99:56:87:B4:8F:...	
Miscellaneous		
Serial Number	00:B1:49:87:4F:6A:19:B1:E4:81:1F:d3:FE:9BF4:DE:71	
Signature Algorithm	SHA-256 with RSA Encryption	

On the right side of the page, there is a promotional message: "Activate Windows Go to Settings to activate Windows".

Running the website:

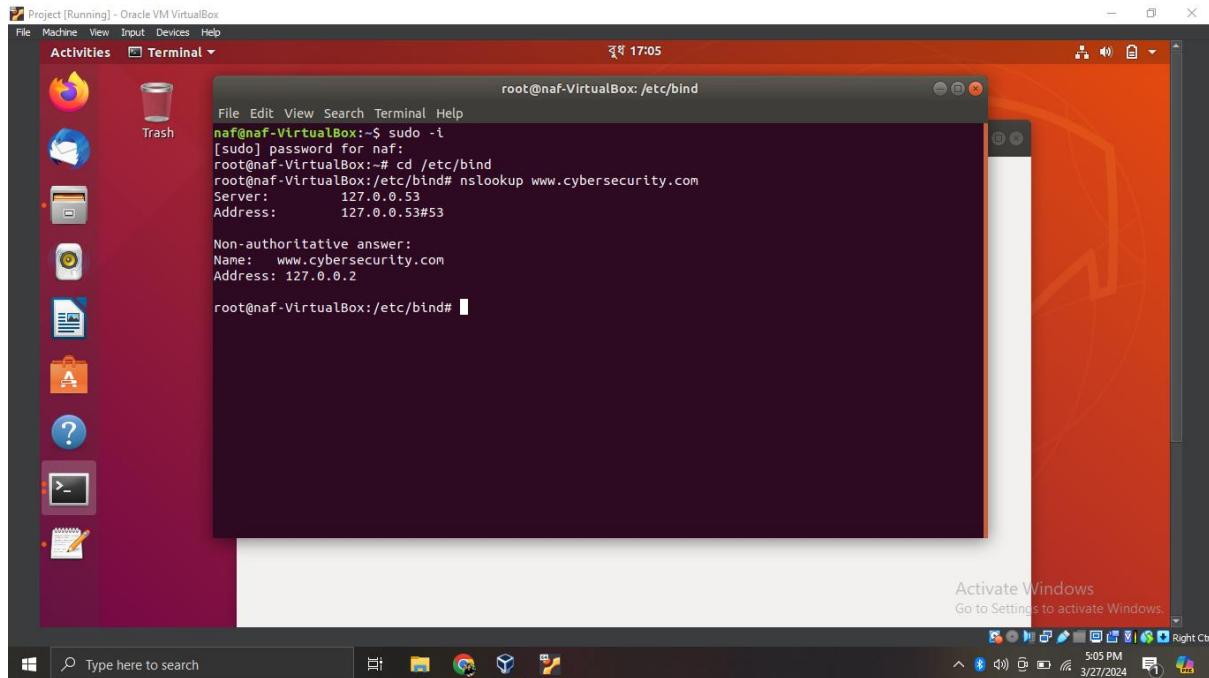
The screenshot shows a Firefox browser window with the address bar displaying "https://www.cybersecurity.com". A security status bar is visible, containing the following information:

- Connection security for www.cybersecurity.com
- You are securely connected to this site.
- Verified by: EWU
- Mozilla does not recognize this certificate issuer. It may have been added from your operating system or by an administrator. [Learn More](#)

Below the status bar, there is a link labeled "More Information".

Configured DNS Server:

nslookup www.cybersecurity.com



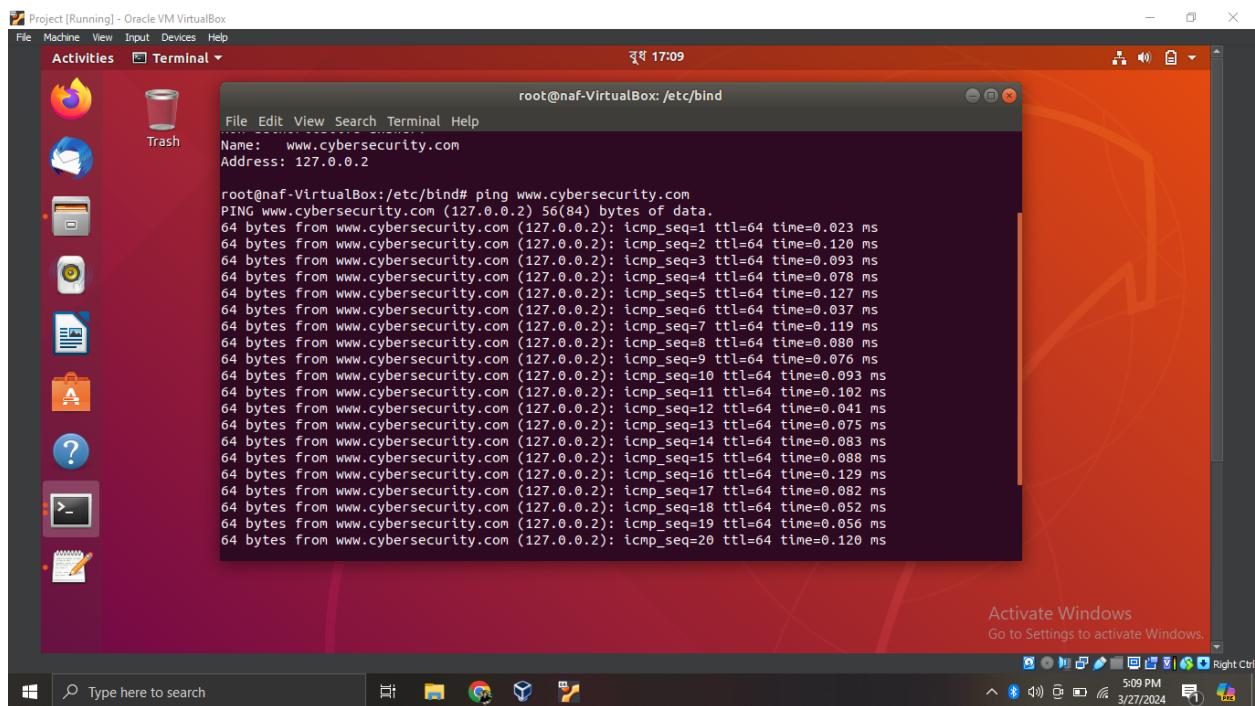
A screenshot of a Linux desktop environment running on a VirtualBox host. The desktop has a dark theme with orange icons. A terminal window titled 'root@naf-VirtualBox:/etc/bind' is open, showing the command 'nslookup www.cybersecurity.com'. The output shows a non-authoritative answer with the name and address of the local DNS server (127.0.0.53). The terminal window is positioned over a Windows taskbar at the bottom.

```
root@naf-VirtualBox:~$ sudo -i
[sudo] password for naf:
root@naf-VirtualBox:~# cd /etc/bind
root@naf-VirtualBox:/etc/bind# nslookup www.cybersecurity.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name: www.cybersecurity.com
Address: 127.0.0.2

root@naf-VirtualBox:/etc/bind#
```

ping www.cybersecurity.com



A screenshot of a Linux desktop environment running on a VirtualBox host. The desktop has a dark theme with orange icons. A terminal window titled 'root@naf-VirtualBox:/etc/bind' is open, showing the command 'ping www.cybersecurity.com'. The output shows a series of ICMP echo requests being sent to the target host, with round-trip times ranging from approximately 0.023 ms to 0.129 ms. The terminal window is positioned over a Windows taskbar at the bottom.

```
root@naf-VirtualBox:/etc/bind# ping www.cybersecurity.com
PING www.cybersecurity.com (127.0.0.2) 56(84) bytes of data.
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=1 ttl=64 time=0.023 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=2 ttl=64 time=0.120 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=3 ttl=64 time=0.093 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=4 ttl=64 time=0.078 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=5 ttl=64 time=0.127 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=6 ttl=64 time=0.037 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=7 ttl=64 time=0.119 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=8 ttl=64 time=0.080 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=9 ttl=64 time=0.076 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=10 ttl=64 time=0.093 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=11 ttl=64 time=0.102 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=12 ttl=64 time=0.041 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=13 ttl=64 time=0.075 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=14 ttl=64 time=0.083 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=15 ttl=64 time=0.088 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=16 ttl=64 time=0.129 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=17 ttl=64 time=0.082 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=18 ttl=64 time=0.052 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=19 ttl=64 time=0.056 ms
64 bytes from www.cybersecurity.com (127.0.0.2): icmp_seq=20 ttl=64 time=0.120 ms
```

Conclusion:

In this project the PKI was implemented using OpenSSL and the DNS server was also configured. PKI helps the users to safely browse the internet and exchange data in encrypted format.