East West University

Mini Project-1 (Cybersecurity)

Course Title: Cyber Security, Law and Ethics
Course Code: CSE487
Section: 01
Semester: Summer 2022

Submitted To

Rashedul Amin Tuhin
Senior Lecturer
Department of Computer Science and Engineering

Submitted By

Group no.: 01
Group members:
Rafiur Rahman Rafit
2018-3-60-111
Jaser Maharus
2018-3-60-070

Submission Date

August 25,2022

Title: Securing a Network System with Public Key Infrastructure (Implementing Transport Layer Security on HTTP for https://connection)

For Linux:

Here we are using kali Linux as an operating system

Firstly, create a localhost in Linux and configure the web server with Apache2 on that Linux Host. For this the instructions are:

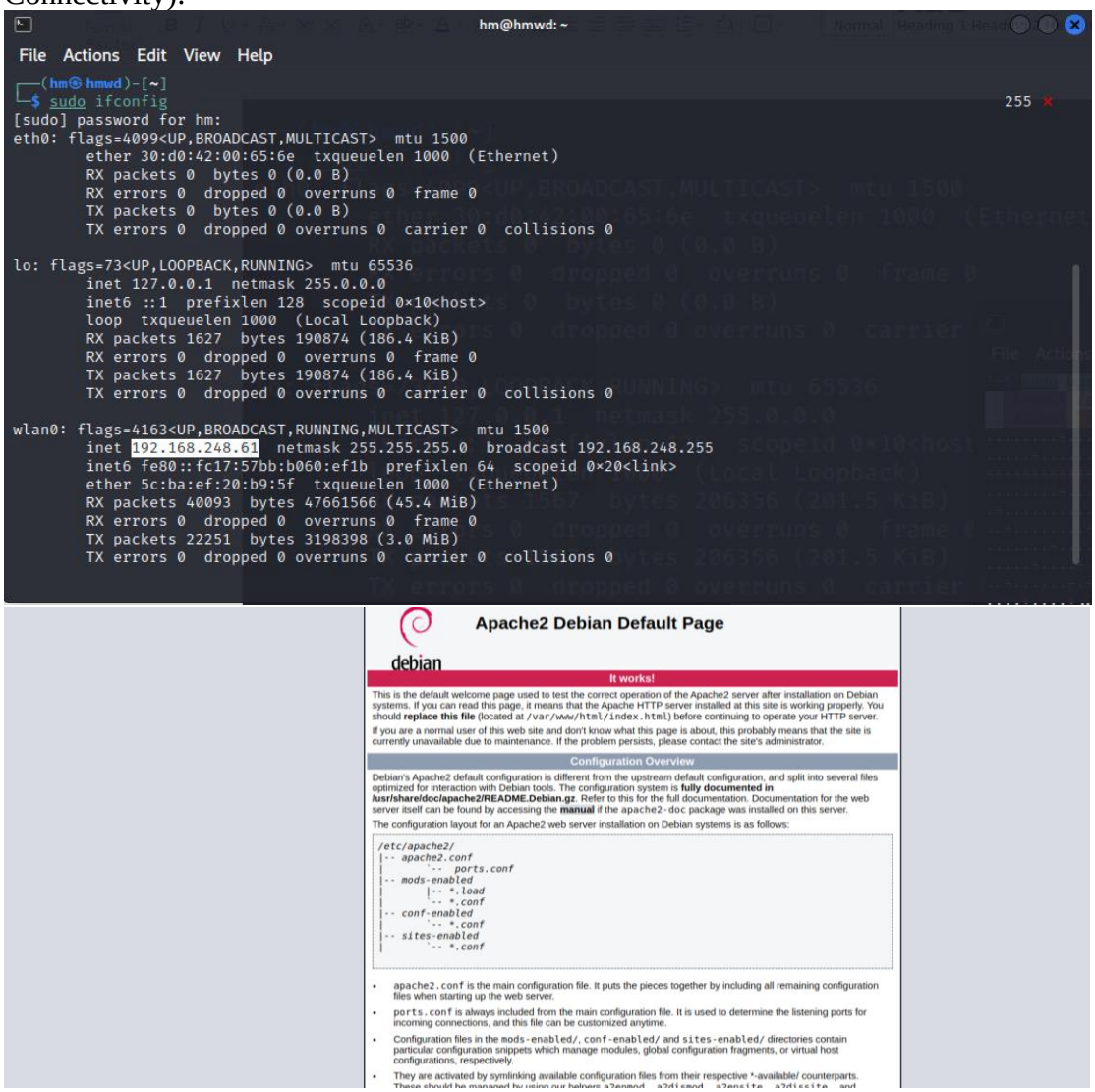
1. Install Apache2 server (sudo apt install apache2)

```
┌──(hm㉿hmwd)-[~]
└─$ sudo apt install apache2
[sudo] password for hm:
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
apache2 is already the newest version (2.4.54-2).
apache2 set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 3 not upgraded.
```

2. Start Apache2 server (sudo service apache2 start)

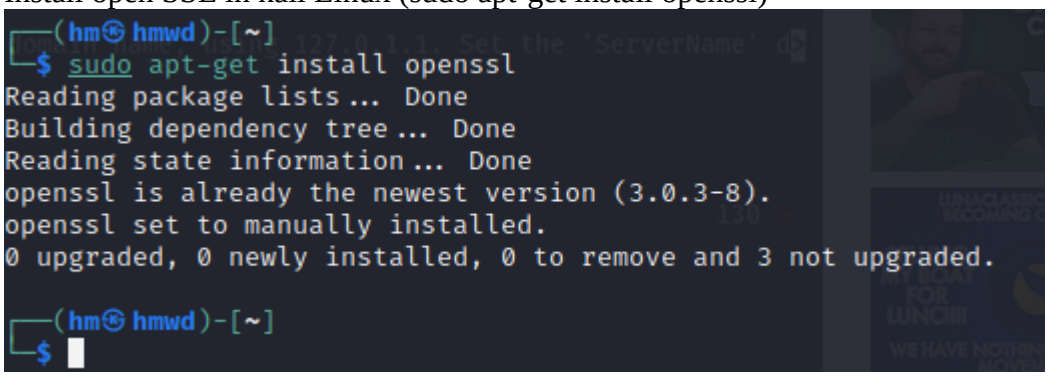
```
┌──(hm㉿hmwd)-[~]
└─$ sudo service apache2 start

┌──(hm㉿hmwd)-[~]
└─$ sudo service apache2 status
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; disabled; vendor preset: disabled)
     Active: active (running) since Fri 2022-07-22 11:42:41 +06; 17s ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 2868 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 2885 (apache2)
      Tasks: 6 (limit: 9178)
     Memory: 19.2M
        CPU: 52ms
     CGroup: /system.slice/apache2.service
             ├─2885 /usr/sbin/apache2 -k start
             ├─2887 /usr/sbin/apache2 -k start
             ├─2888 /usr/sbin/apache2 -k start
```

3. Check IP or type local host in browser by following these steps (here IP Change due to Network Connectivity):



Our server is running but it is not secure, so we must use SSL Certificate and make browser trust that our certificate is real. For doing this the instructions are:

1. Install open SSL in kali Linux (sudo apt-get install openssl)



Create SSL certificate (“s sudo openssl req -x509 -newkey rsa:4096 -sha256 -days 3650 -nodes \ -keyout /etc/ssl/private/server.key -out /etc/ssl/certs/server.crt -subj "/CN=www.acmesecureserver.com" \ -addext"subjectAltName=DNS:www.acmesecureserver.com,DNS:*.webview.acmesecureserver.com,IP:192.168.60.61")

```
┌──(hm㉿hmwd)-[~]
└─$ sudo openssl req -x509 -newkey rsa:4096 -sha256 -days 3650 -nodes \
  -keyout /etc/ssl/private/server.key -out /etc/ssl/certs/server.crt -subj "/CN=www.acmesecureserver.com" \
  -addext "subjectAltName=DNS:www.acmesecureserver.com,DNS:*.webview.acmesecureserver.com,IP:192.168.60.61"
```

2.

```
rganization Name (eg, company) [Internet Widgits Pty Ltd]:hakuna matata
rganizational Unit Name (eg, section) []:hm
ommon Name (e.g. server FQDN or YOUR name) []:rafit
mail Address []:rafiurrahmanrafit
```

3. sudo a2enmod ssl

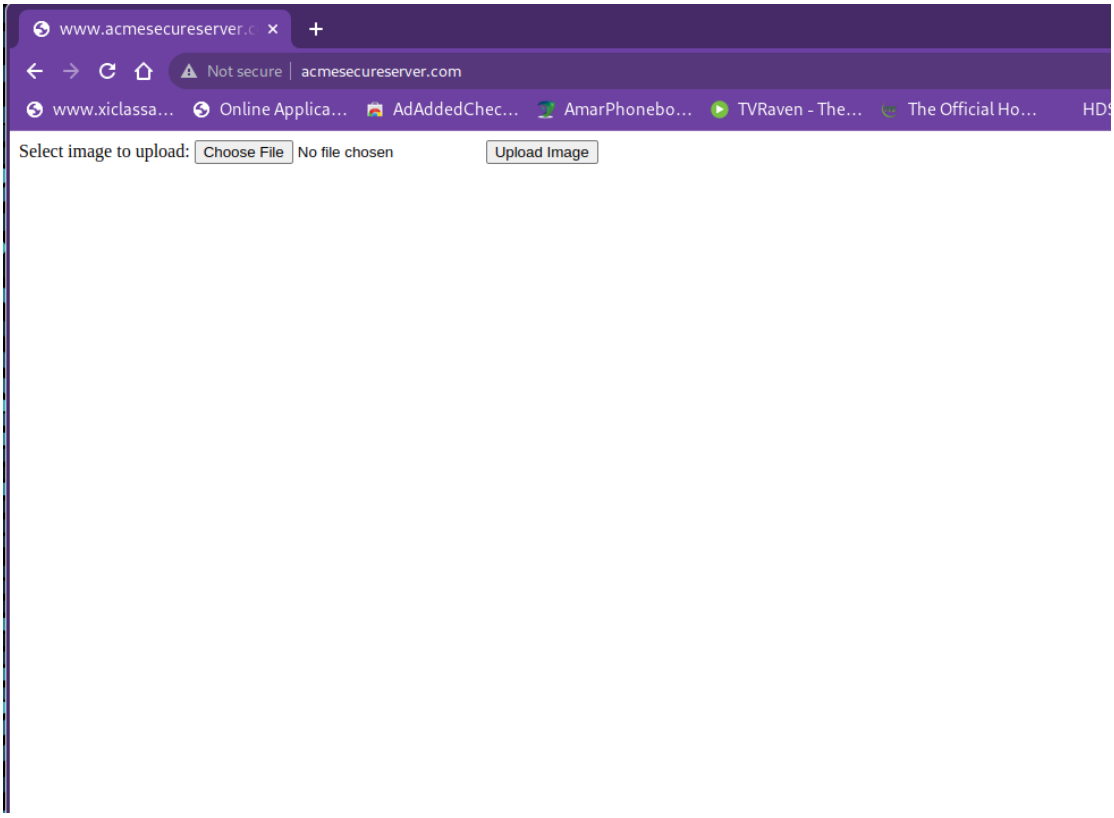
4. restart server(systemctl restart apache2)

5. sudo ln -s /etc/apache2/sites-available/default-ssl.conf /etc/apache2/sites-available/000-default-ssl.conf

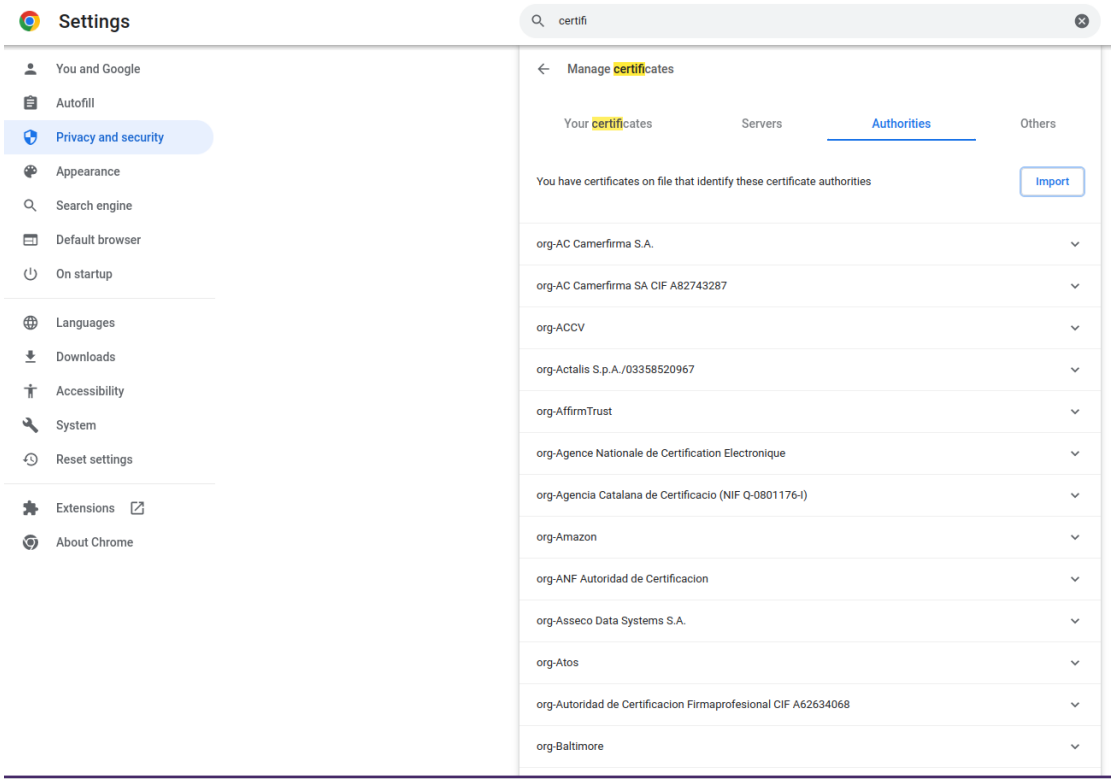
6. edit file (/etc/apache2/sites-enabled/000-default-ssl.conf)

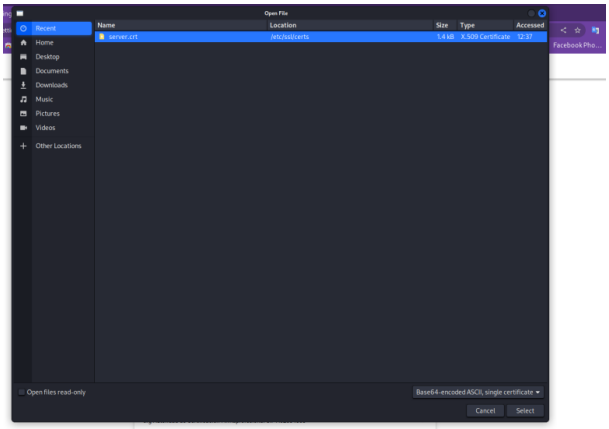
```
CustomLog ${APACHE_LOG_DIR}/access.log combined

# For most configuration files from conf-available/, which are
# enabled or disabled at a global level, it is possible to
# include a line for only one particular virtual host. For example the
# following line enables the CGI configuration for this host only
# after it has been globally disabled with "a2disconf".
#Include conf-available/serve-cgi-bin.conf

#   SSL Engine Switch:
#   Enable/Disable SSL for this virtual host.
SSLEngine on

#   A self-signed (snakeoil) certificate can be created by installing
#   the ssl-cert package. See
#   /usr/share/doc/apache2/README.Debian.gz for more info.
#   If both key and certificate are stored in the same file, only the
#   SSLCertificateFile directive is needed.
SSLCertificateFile  /etc/ssl/certs/server.crt
SSLCertificateKeyFile /etc/ssl/private/server.key

#   Server Certificate Chain:
#   Point SSLCertificateChainFile at a file containing the
#   concatenation of PEM encoded CA certificates which form the
#   certificate chain for the server certificate. Alternatively
#   the referenced file can be the same as SSLCertificateFile
#   when the CA certificates are directly appended to the server
#   certificate for convinience.
#SSLCertificateChainFile /etc/apache2/ssl.crt/server-ca.crt

#   Certificate Authority (CA):
#   Set the CA certificate verification path where to find CA
```

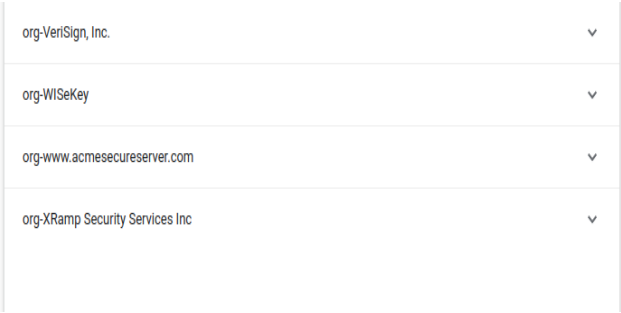
7. reboot server



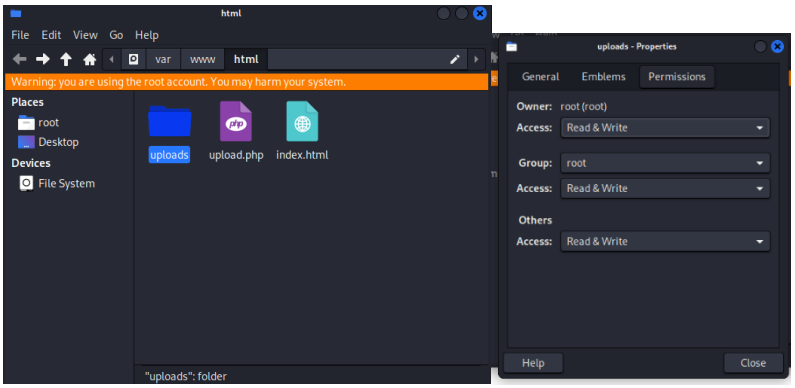
8. Now in chrome we must add certificate. For doing this the instructions are:



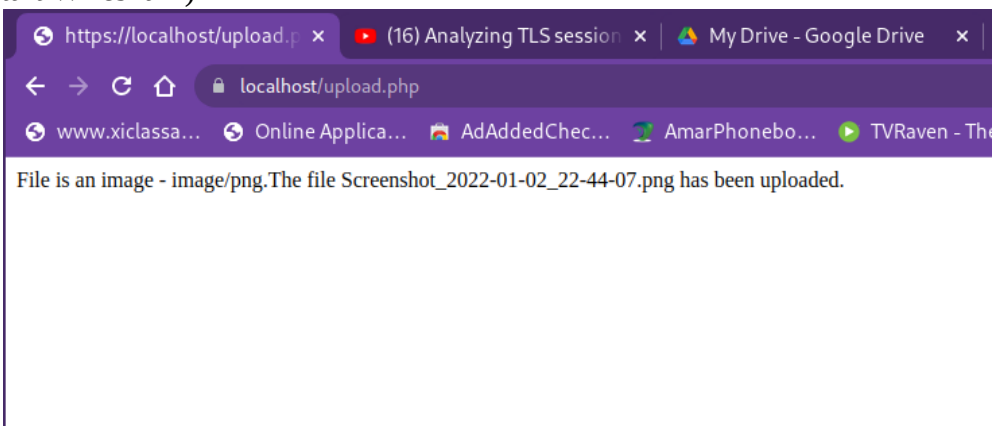


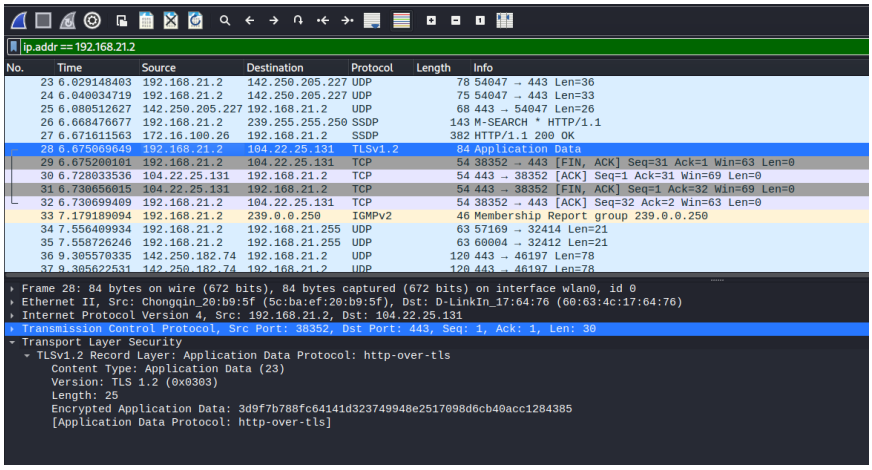


9. Now Create simple upload page, simply create an upload html page with PHP language for uploading a file. and give read and write permission to upload folders also. The instructions for doing this are:



10. Test security in Wireshark (Open wire Wireshark app, filter it by giving the IP address of the web server and start Wireshark)

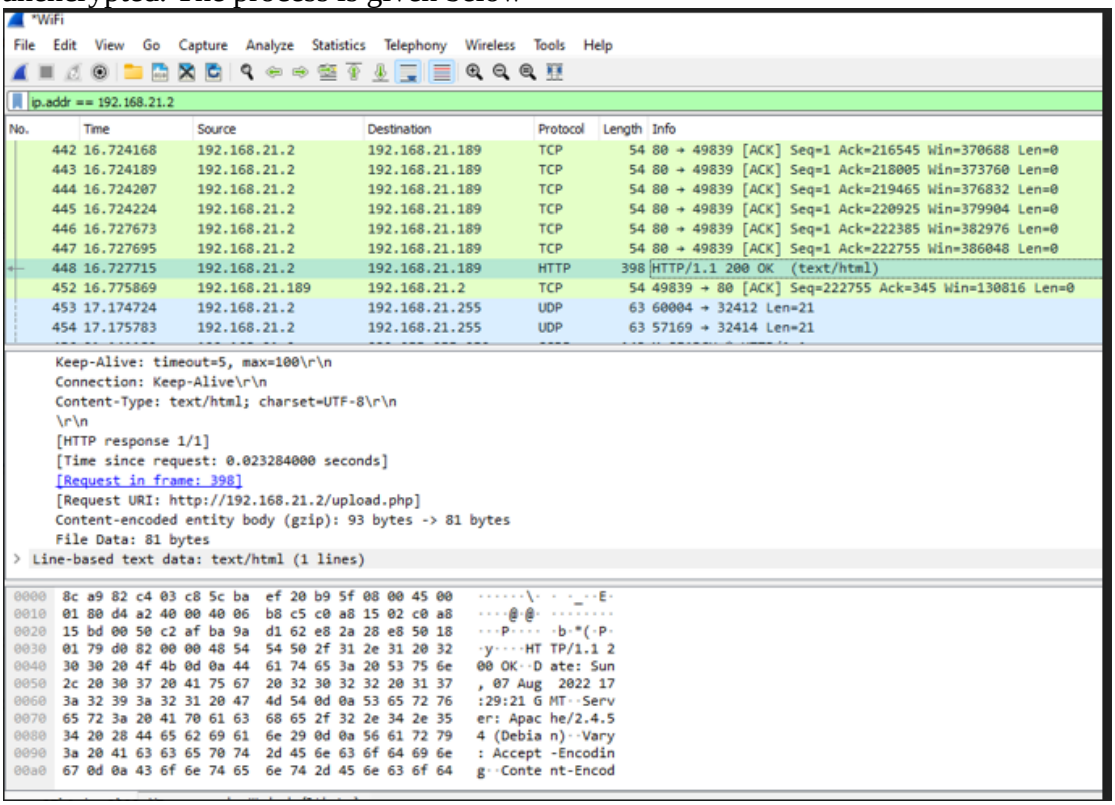




TLS 1.3/1.2 is the latest version of the internet's most deployed security protocol, which encrypts data to provide a secure communication channel between two endpoints.

Here wire shark file link has been given
https://drive.google.com/file/d/1p2smKi6Q_lSCHFRFV2k-KA4CWSHN0mXm/view?usp=sharing

Now from client pc where certificate is not installed in the browser, when we run Wireshark on that we see our data is unencrypted. The process is given below-



https://drive.google.com/file/d/1vOfPDZMaIhRj8hH8QNZ999k0Ioe9E7f3/view?usp=sharing

Now to remove Certificate we need to type command in Linux. The instructions are:

- a2dissite default-ssl.conf
- a2enmod ssl

Now For IDS System, we install SNORT, it is supported for Ubuntu system. To install SNORT, we need to add Ubuntu Repository in Kali Linux

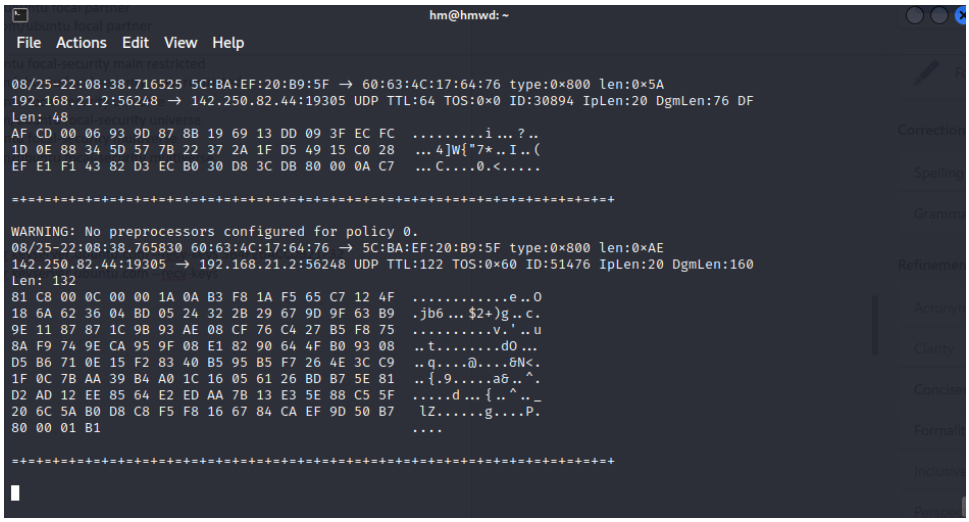
```
# deb cdrom:[Ubuntu 20.04 LTS _Focal Fossa_ - Release amd64 (20200423)]/ focal main restricted

# See http://help.ubuntu.com/community/UpgradeNotes for how to upgrade to
# newer versions of the distribution.
deb http://archive.ubuntu.com/ubuntu focal main restricted
# deb-src http://in.archive.ubuntu.com/ubuntu/ focal main restricted


deb http://archive.ubuntu.com/ubuntu focal-updates main restricted
# deb-src http://in.archive.ubuntu.com/ubuntu/ focal-updates main restricted


deb http://archive.ubuntu.com/ubuntu focal universe
# deb-src http://in.archive.ubuntu.com/ubuntu/ focal universe
deb http://archive.ubuntu.com/ubuntu focal-updates universe
# deb-src http://in.archive.ubuntu.com/ubuntu/ focal-updates universe


deb http://archive.ubuntu.com/ubuntu focal multiverse
# deb-src http://in.archive.ubuntu.com/ubuntu/ focal multiverse
deb http://archive.ubuntu.com/ubuntu focal-updates multiverse
# deb-src http://in.archive.ubuntu.com/ubuntu/ focal-updates multiverse


deb http://archive.ubuntu.com/ubuntu focal-backports main restricted universe multiverse
# deb-src http://in.archive.ubuntu.com/ubuntu/ focal-backports main restricted universe multiverse


# deb http://archive.canonical.com/ubuntu focal partner
# deb-src http://archive.canonical.com/ubuntu focal partner

deb http://archive.ubuntu.com/ubuntu focal-security main restricted
# deb-src http://security.ubuntu.com/ubuntu focal-security main restricted
deb http://archive.ubuntu.com/ubuntu focal-security universe
# deb-src http://security.ubuntu.com/ubuntu focal-security universe
deb http://archive.ubuntu.com/ubuntu focal-security multiverse
# deb-src http://security.ubuntu.com/ubuntu focal-security multiverse
```

Then type these commands -

- sudo apt update
- sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys 3B4FE6ACC0B21F32
- sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys 871920D1991BC93C
- sudo apt-get -y install snort

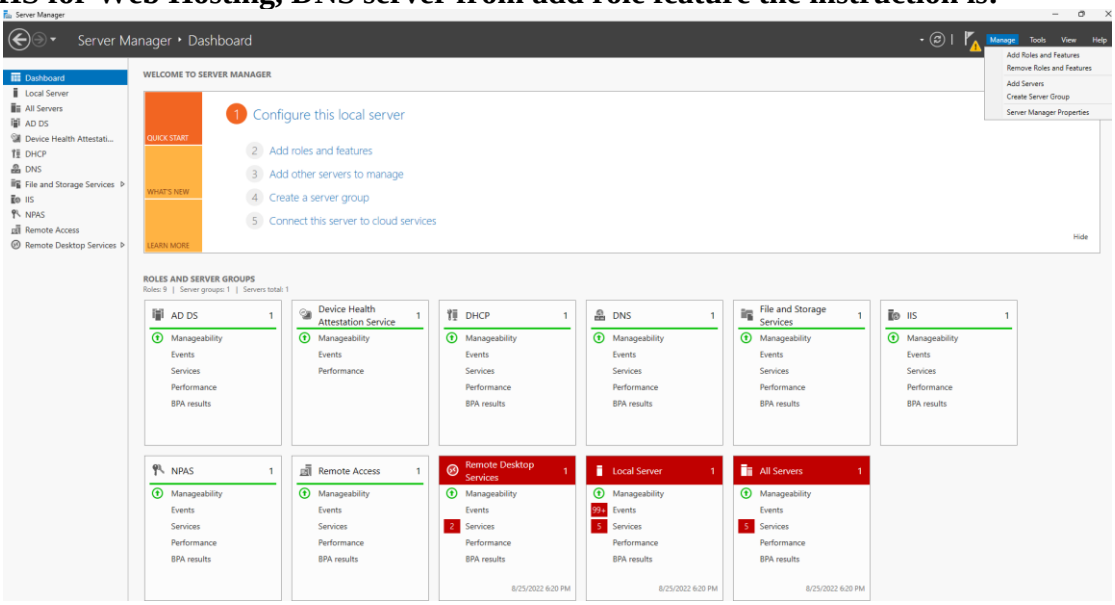
Then start SNORT

The instruction for this is:
sudo snort -vde

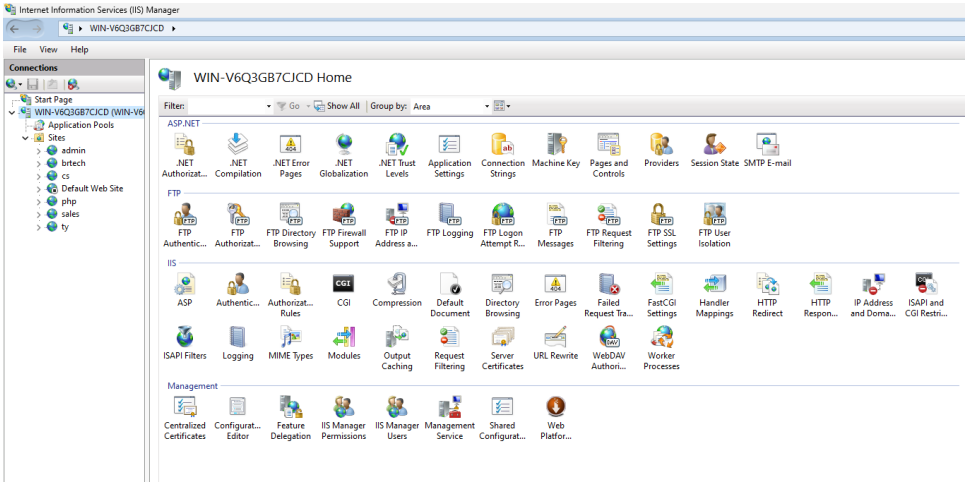


In this Project We are also making website using Windows Server 2022 and securing that site using SSL (Lets Encrypt)

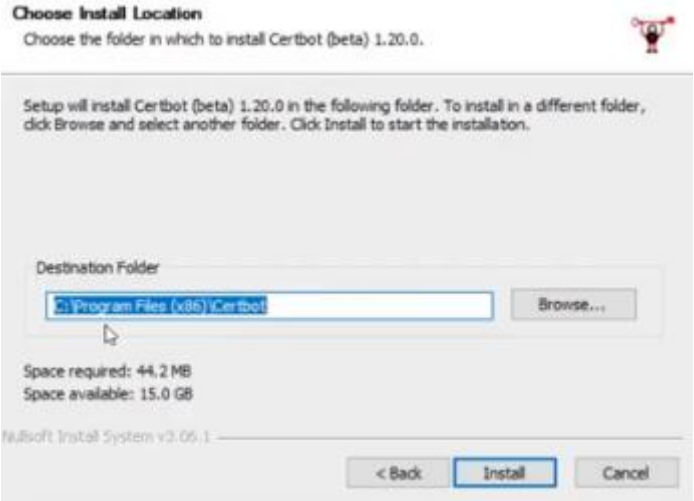
To install IIS for Web Hosting, DNS server from add role feature the instruction is:



1. Create a Website/Web APP in IIS



2. Encrypt the Web using Let's Encrypt using Certbot Acme Client



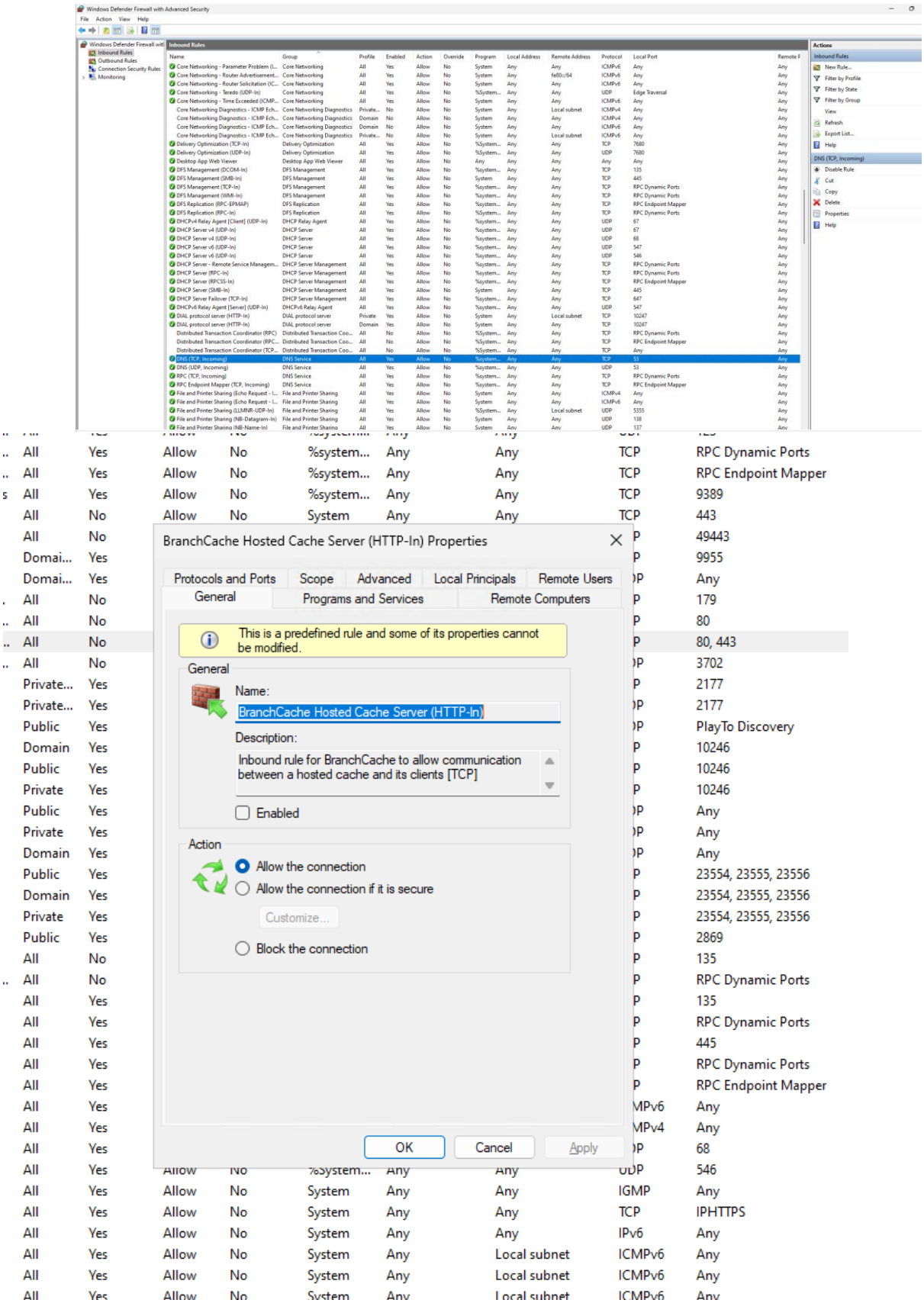
3. Run Certbot in CMD

```
C:\Users\Administrator>certbot
Saving debug log to C:\Certbot\log\letsencrypt.log
Certbot doesn't know how to automatically configure the web server on this system. However, it can still get a certifica
te for you. Please run "certbot certonly" to do so. You'll need to manually configure your web server to use the resulti
ng certificate.

C:\Users\Administrator>_
```

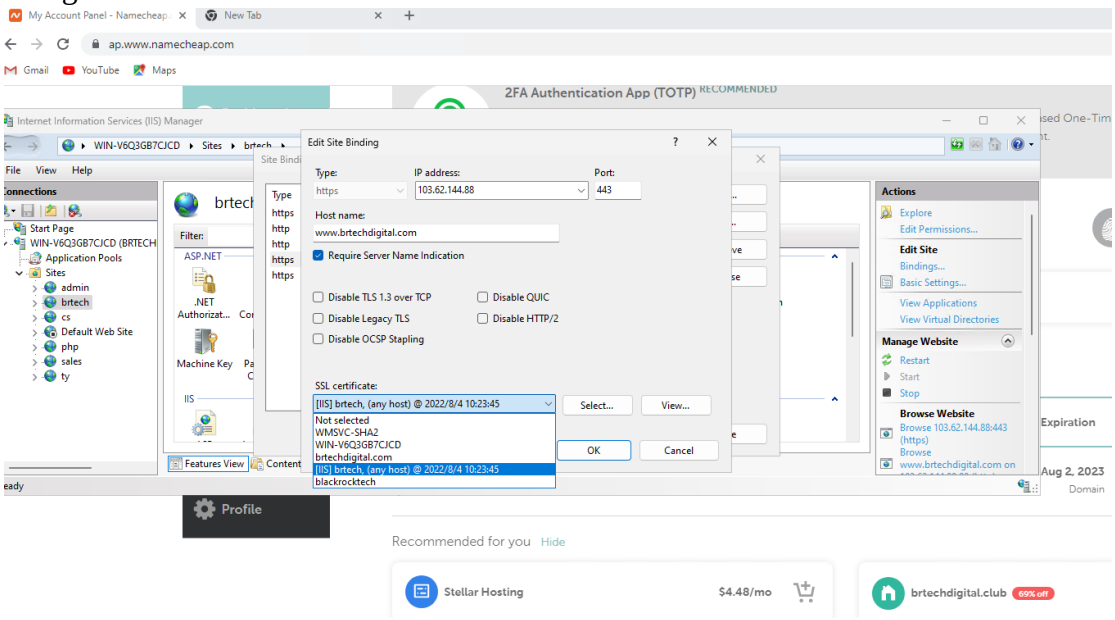
4. Create SSL Certificate by giving host name, Domain Name, Email Address. For this the instruction is:

```
C:\Users\Administrator>certbot certonly --standalone
Saving debug log to C:\Certbot\log\letsencrypt.log
Please enter the domain name(s) you would like on your certificate (comma and/or
space separated) (Enter 'c' to cancel): blackrocktech.xyz
Requesting a certificate for blackrocktech.xyz
```

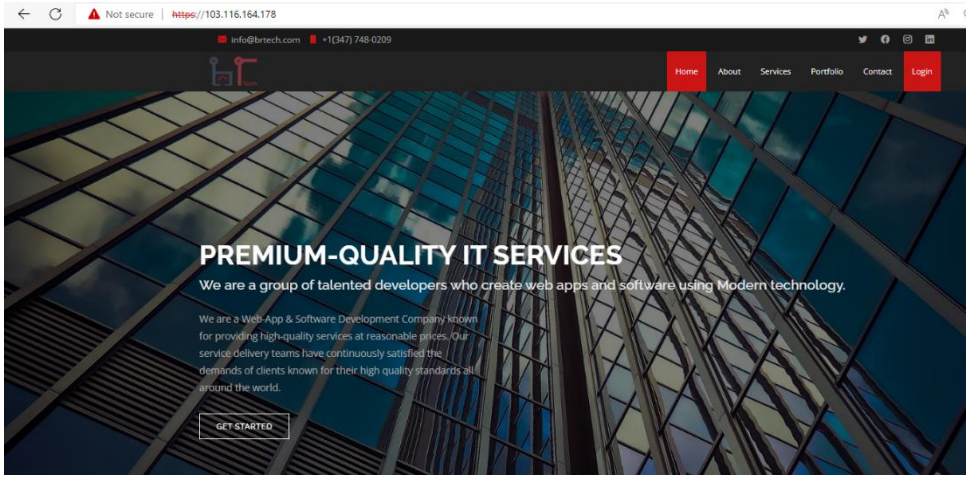
5. Allowing Particular Port in Firewall Settings and by default 80 and 443 port is opened



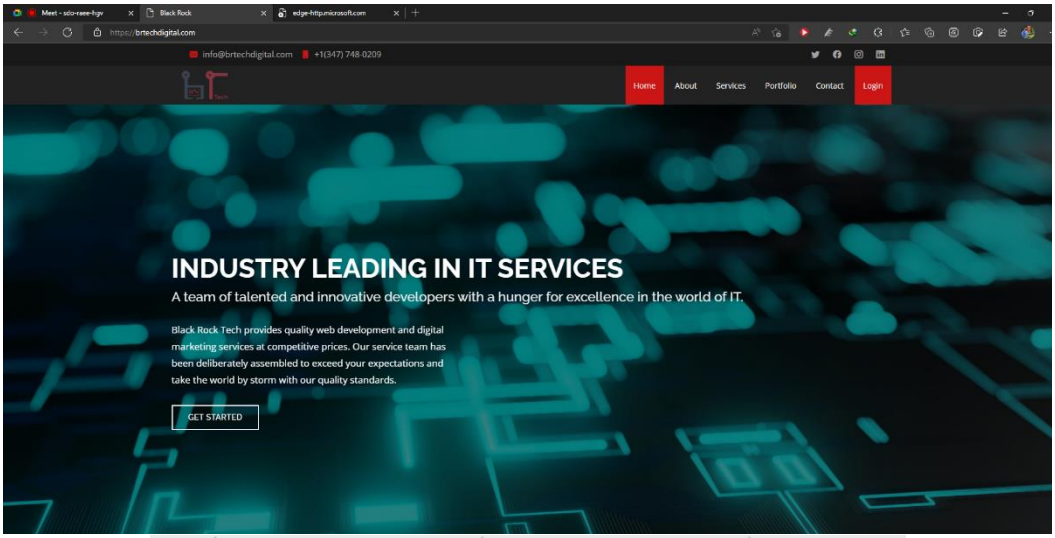
6. Declare server IP address, here we are using public IP and for domain name we are using Namecheap. Our Domain cost is 12 dollars (private-company domain). Add the SSL certificate that was generated.

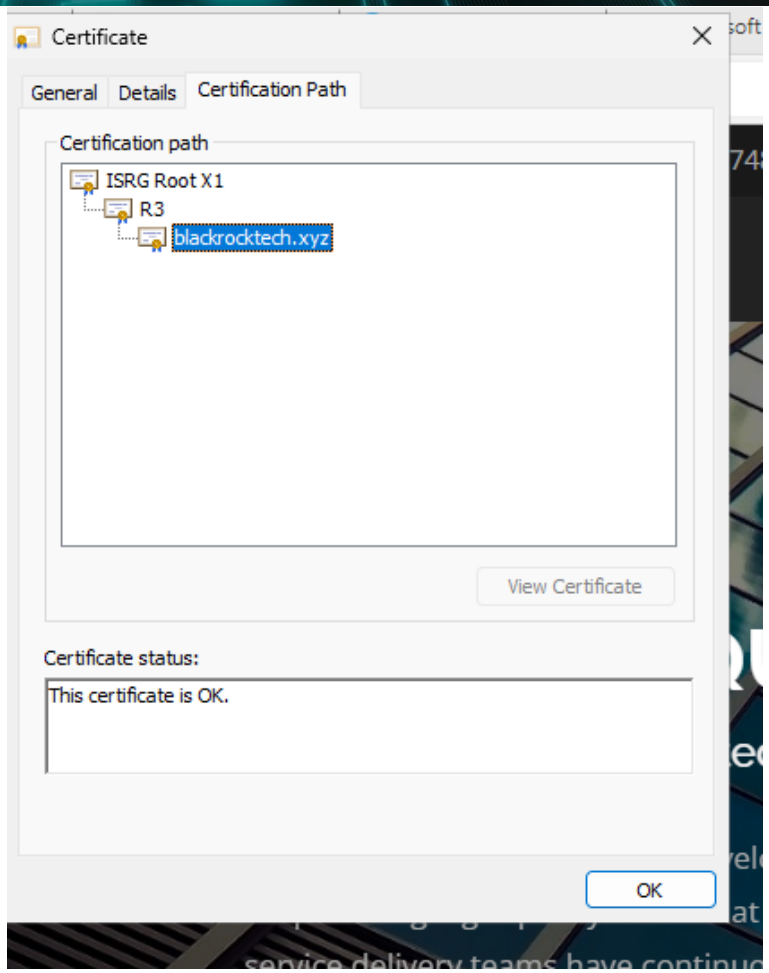


7. Domain Name must be same as SSL certificate domain, or its certificate will not be accepted

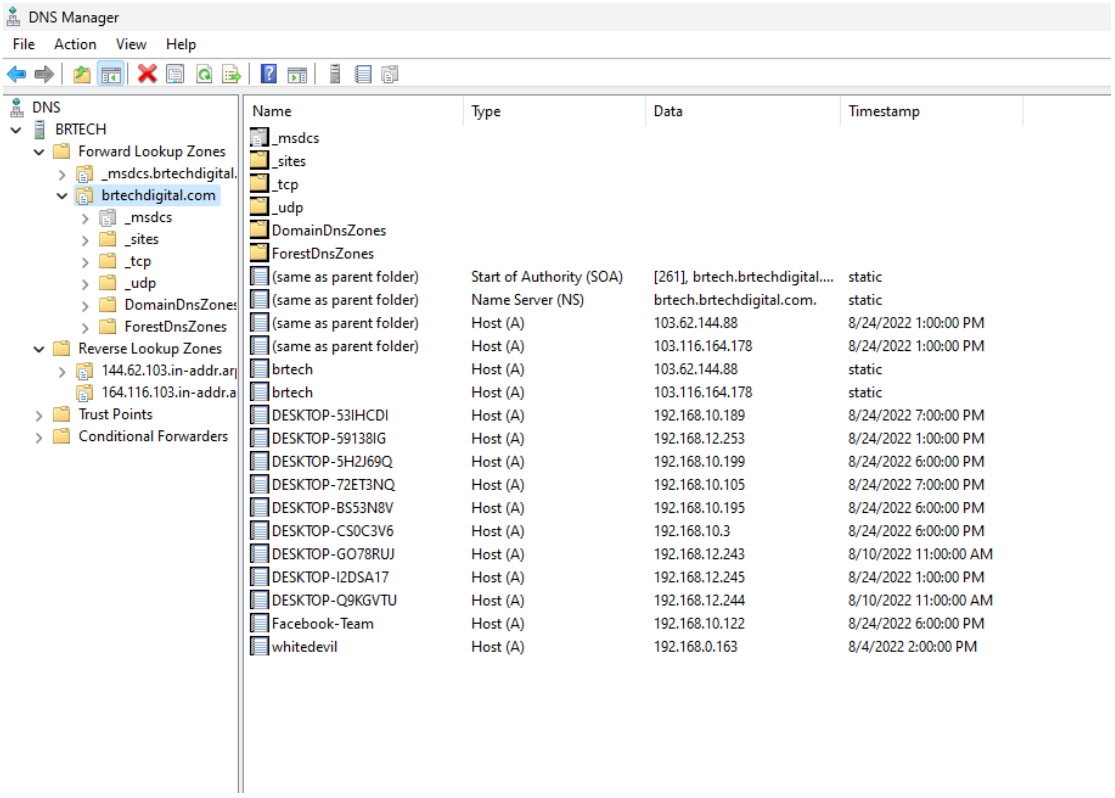


8. Finally, our website is protected

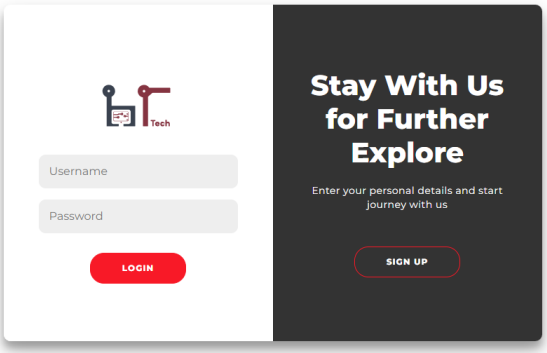


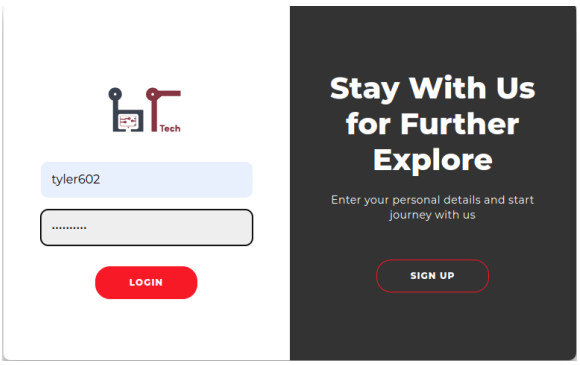


9. Now From Any Device our device will show secure SSL certificate
10. Now we add DNS Configuration also

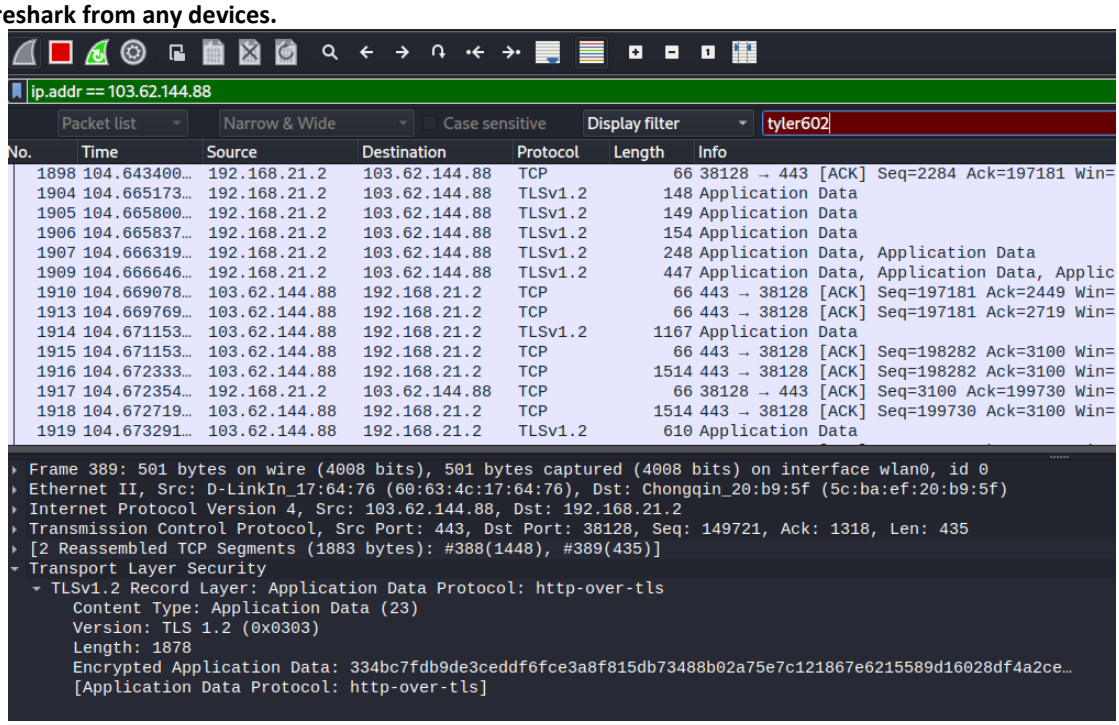


Now in windows server, our website brtechdigital.com is encrypted with let's encrypt. now we test our website in login panel





Running Wireshark from any devices.



Here we see our data has been encrypted and file is in the link
https://drive.google.com/file/d/1GDykaAFo5m3vaQBU-1NxHFgSpiqXTbjY/view?usp=sharing