EAST WEST UNIVERSITY

Project Report

MP-1 (Cyber Security)

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

Semester: Summer2022

Course Name: Cybersecurity, Law and Ethics
Course Code: CSE487
Section: 01

Course Instructor: Rashedul Amin Tuhin
Senior Lecturer
Dept. of Computer Science & Engineering

Submitted by:

	Student ID	**Student Name**
Group 105	2019-1-60-034	Sadia Huq
	2018-2-60-071	Fabiha tasneem
	2018-2-60-129	Sanjida Kater

Requirements:

- Configuration of Certification Authority Acme with Acme-RootCA as the RootCA.
- Configuration of the Web Server with Apache2 on a Linux Host.
- DNS configuration for www.verysecureserver.com
- Firewall configuration to allow necessary ports (53, 80, 443) only
- CSR Configuration and Generation for the www.verysecureserver.com
- Transferring the CSR to Acme.
- Certification process (Verification and Certificate Generation from CSR)
- Transferring the certificate from AcmeCA to www.verysecureserver.com
- Installation of the signed the SSL certificate in the server of www.verysecureserver.com
- Making the system trust Acme-RootCA
- Implementation of a simple file uploading page in the server.
- Verifying the security of the connection by inspection (the padlock icon), and with wireshark from another computer.
- Revoke the certificate issued to www.verysecureserver.com from the CA and distribute the first CRL. [bonus]
- Verifying the revocation of previous certificate from the CRL (no padlock icon).
- Configuring IDS [bonus]

❖ For configuring of certification authority and implementing the transport layer security over http, we followed some steps. All the steps are given below:

Step-1: Virtual Machine Installation

✓ We have Installed the Kali Linux (version 2022.3) in the VMware workstation for Host computer (kali 1) where all our website is hosted, and all the certifications are generated.

Step-2: Generating SSL certificate Using openssl

✓ Logged into the virtual machine (kali 1) as a root user and opened a folder in the desktop name ‘ssl’. Now, open the terminal and paste the below code and press enter:

```
echo "\n\n_________________GENERATING ALL DIRECTORIES_________________\n\n"
gr='\033[1;32m'
nc='\033[0m' # No Color
```

```
mkdir -p {root-ca,sub-ca,server}/{private,certs,index,serial,pem,crl,csr}
mkdir generated
touch root-ca/index/index
touch sub-ca/index/index
openssl rand -hex 16 > root-ca/serial/serial
openssl rand -hex 16 > sub-ca/serial/serial
cp root-ca.conf root-ca
cp sub-ca.conf sub-ca
echo "${gr}\n ================ FOLDERS CREATED SUCCESSFULLY ================ \n${nc}"

echo "\n\n_________________GENERATING ALL THE KEYS_________________\n\n"

openssl genrsa -aes256 -out root-ca/private/ca.key 4096
openssl genrsa -aes256 -out sub-ca/private/sub-ca.key 4096
openssl genrsa -out server/private/server.key 2048
echo "${gr}\n ================ KEYS CREATED SUCCESSFULLY ================ \n${nc}"

echo "\n\n_________________GENERATING ROOT CERTIFICATE_________________\n\n"

openssl req -config root-ca/root-ca.conf -key root-ca/private/ca.key -new -x509 -days 7305 -sha256 -extensions v3_ca -out root-ca/certs/ca.crt
echo "${gr}\n ================ ROOT CERTIFICATE CREATED SUCCESSFULLY ================ \n${nc}"

echo "\n\n_________________GENERATING SUB-ROOT REQUEST_________________\n\n"

openssl req -config sub-ca/sub-ca.conf -new -key sub-ca/private/sub-ca.key -sha256 -out sub-ca/csr/sub-ca.csr
echo "${gr}\n ================ SUB-ROOT REQUEST CREATED SUCCESSFULLY ================ \n${nc}"

echo "\n\n_________________GENERATING SUB-ROOT CERTIFICATE_________________\n\n"

openssl ca -config root-ca/root-ca.conf -extensions v3_intermediate_ca -days 3652 -notext -in sub-ca/csr/sub-ca.csr -out sub-ca/certs/sub-ca.crt
echo "${gr}\n ================ SUB-ROOT CERTIFICATE CREATED SUCCESSFULLY ================ \n${nc}"
```

```
echo "\n\n_________________GENERATING SERVER REQUEST_________________\n\n"

openssl req -key server/private/server.key -new -sha256 -out server/csr/server.csr
echo "${gr}\n ================ SERVER REQUEST CREATED SUCCESSFULLY ================ \n${nc}"

echo "\n\n_________________GENERATING SERVER CERTIFICATE_________________\n\n"

openssl ca -config sub-ca/sub-ca.conf -extensions server_cert -days 365 -notext -in server/csr/server.csr -out server/certs/server.crt
openssl pkcs12 -inkey server/private/server.key -in server/certs/server.crt -export -out server/certs/server.pfx
echo "${gr}\n ================ SERVER CERTIFICATE CREATED SUCCESSFULLY ================ \n${nc}"

echo "\n\n_________________GATHERING NECESSARY FILES_________________\n\n"

cp root-ca/certs/ca.crt generated
cp sub-ca/certs/sub-ca.crt generated
cp server/certs/server.crt generated
cp server/private/server.key generated
cp server/certs/server.pfx generated
echo "${gr}\n ================ SUCCESSFULLY GATHERED ================ \n${nc}"

echo "\n\n_________________CREATING HOST ENTRY_________________\n\n"

echo -n "Server CommonName: "
read commonName
echo "127.0.0.1 "$commonName >> /etc/hosts

echo "${gr}\n ================ SUCCESSFULLY APPENDED HOST ================ \n${nc}"
```

- ✓ Now, in the terminal it asked to enter the PEM pass phrase and to verify it.

We wrote the PEM pass phrase same as our VM machine password. Thus, All the Keys got generated.

- ✓ Enter pass phrase for root-ca/private/ca.key: (entered the PEM pass phrase)

Now to generate the Root certificate the below information has been given:
Country Name (2 letter code) [BD]:
State or Province Name [Dhaka]:
Locality Name [Rampura]:
Organization Name:
Organizational Unit Name [Acme]:
Common Name: Acme-RootCA
Email Address:
The Root certificate is generated.

- ✓ Enter pass phrase for sub-ca/private/sub-ca.key: (entered the PEM pass phrase)

Now to generate the Sub-Root certificate the below information has been given:
Country Name (2 letter code) [BD]:
State or Province Name [Dhaka]:
Locality Name [Rampura]:
Organization Name:
Organizational Unit Name [Acme]:
Common Name: Acme
Email Address:
The Sub-Root certificate is generated. Now Enter pass phrase again and press y for certificate credentials and commit y to certificate request.

- ✓ Now to generate the Server Request the below information has been given:

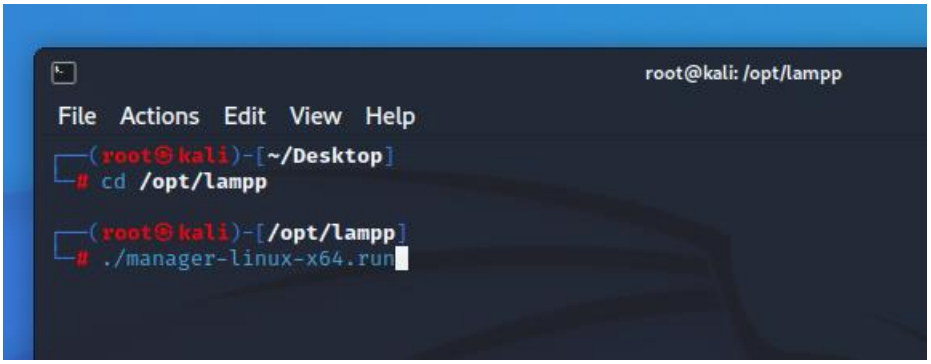
Country Name (2 letter code) [AU]: BD
State or Province Name [Full name]: Dhaka
Locality Name [eg. city]:
Organization Name [eg. company]: Acme
Organizational Unit Name [eg. section]:
Common Name: verysecureserver.com
Email Address:
The Server request created successfully.

- ✓ Generate server certificate
- ✓ Enter pass phrase for sub-ca/private/sub-ca.key:

Now press y for certificate credentials and commit y to certificate request. And again, enter server common name.

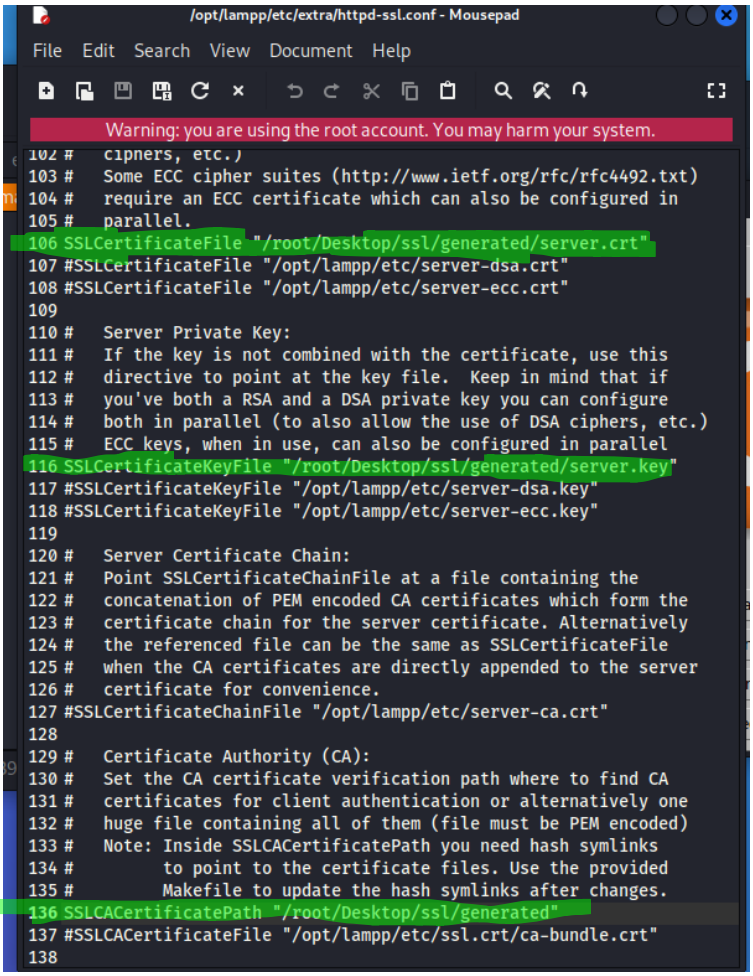
Step-3: Install Openssl Generated certificates in Xampp

- Install Xampp and open terminal and write below command:



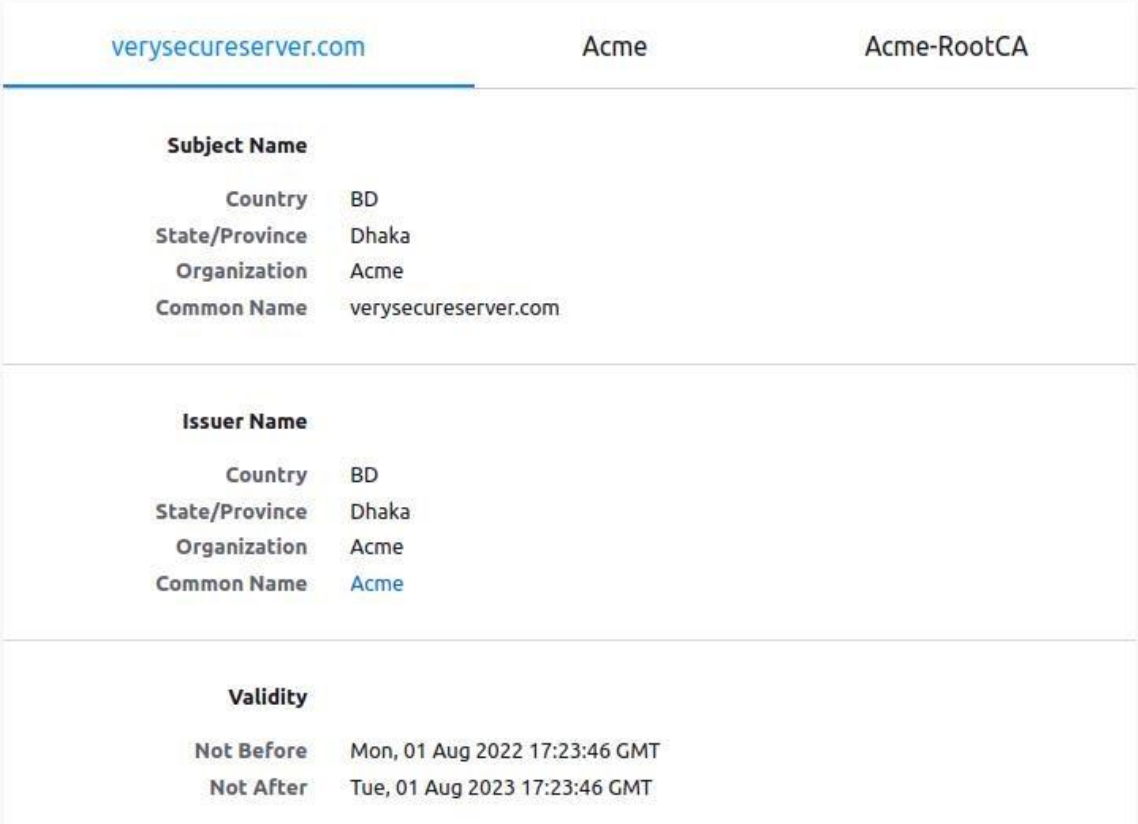
- Then xampp control panel opens up and from there go to Application folder >opt>lamp>etc>extra> httpd-ssl.conf

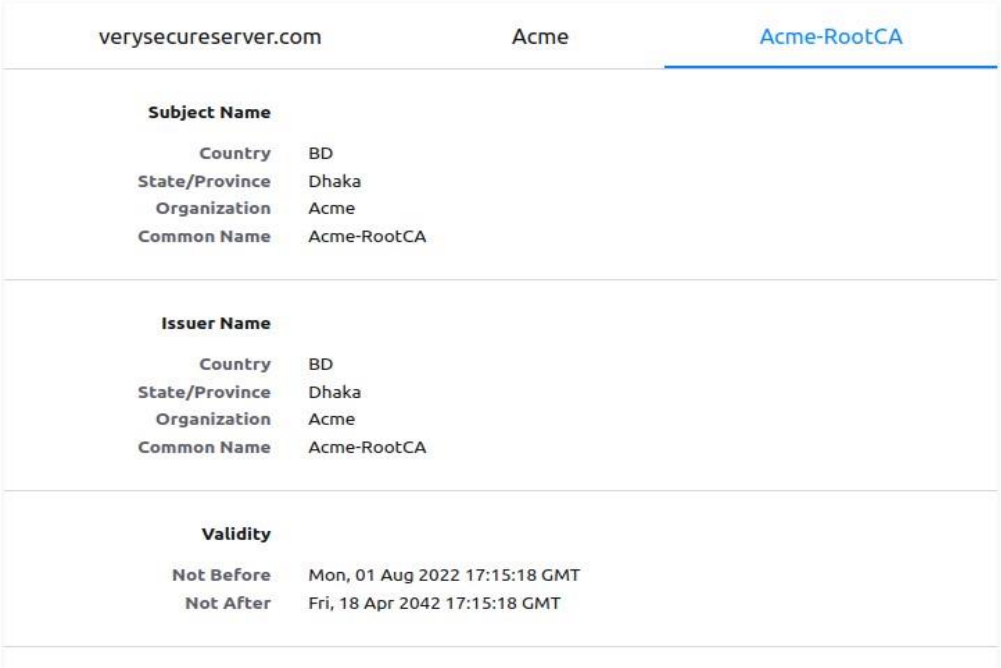
And changed the following lines to these directories.



- Now write server url in browser and it will show unsecured. So, go to browser setting and import certificate files (ca.crt, sub-ca.crt) in authorities and trust them. And import server.pfx in your certificates. And now the padlock sign is shown in the browser as we enter the url again.

✓ The chained certificate:





verysecureserver.com | Acme | Acme-RootCA

Subject Name

Country	BD
State/Province	Dhaka
Organization	Acme
Common Name	Acme-RootCA

Issuer Name

Country	BD
State/Province	Dhaka
Organization	Acme
Common Name	Acme-RootCA

Validity

Not Before	Mon, 01 Aug 2022 17:15:18 GMT
Not After	Fri, 18 Apr 2042 17:15:18 GMT

- ✓ For Revocation write following command on Terminal:
  - cd sub-ca
  - openssl ca -config sub-ca.conf -revoke ../server/certs/server.crt
  - cd sub-ca
  - nano crlnumber
  - #type: 1002
  - openssl ca -config sub-ca.conf -gencrl -out crl/rev.crl

Step-4: Install DNS server in Linux using Webmin and bind9 installation

- For client machine again open Virtual machine kali Linux as Kali 2.

Open machine and Run the following command as kali user;

- sudo apt update
- sudo apt-get install open-vm-tools-desktop
- sudo apt install kali-root-login
- sudo passwd
- reboot

----------#switch user

- Log in as root user

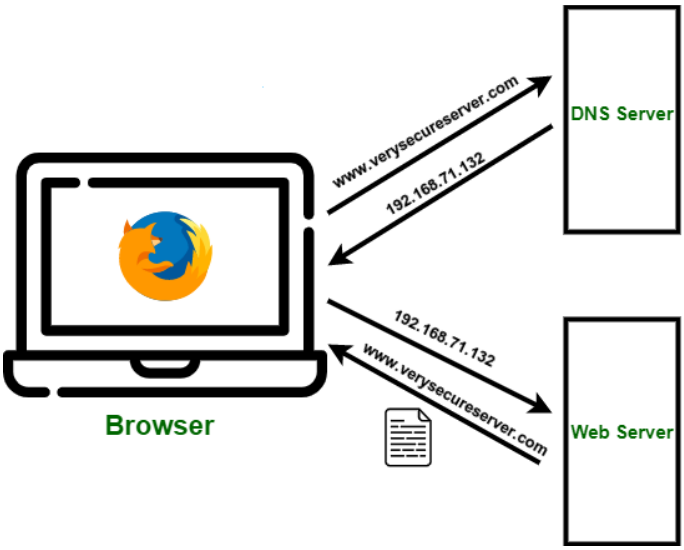
----------#webmin installation

- sudo nano /etc/apt/sources.list
- ----------#add this line
- deb http://download.webmin.com/download/repository sarge contrib

- wget -q -O- http://www.webmin.com/jcameron-key.asc | sudo apt-key add
- sudo apt update
- sudo apt install webmin
- sudo ufw allow 10000
- cd /usr/lib/systemd/system
- cp named.service bind9.service
- systemctl webmin status

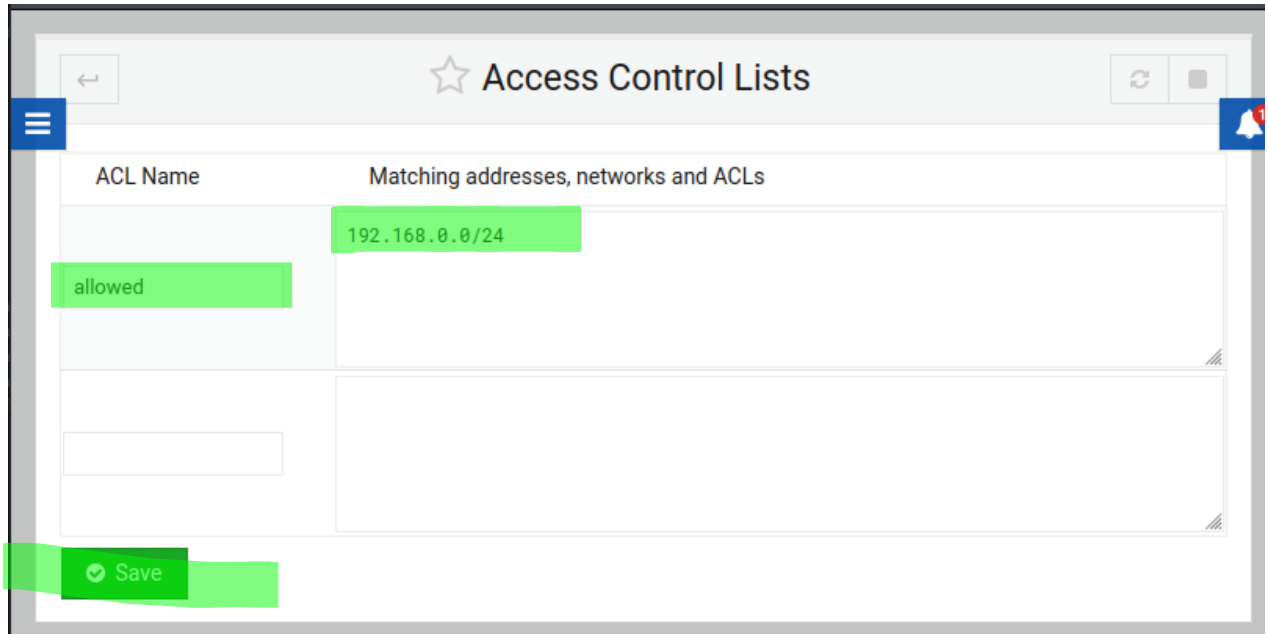
- Now go to browser and write localhost:10000 and it enters the webmin. Now install bind9 packages from this interface.

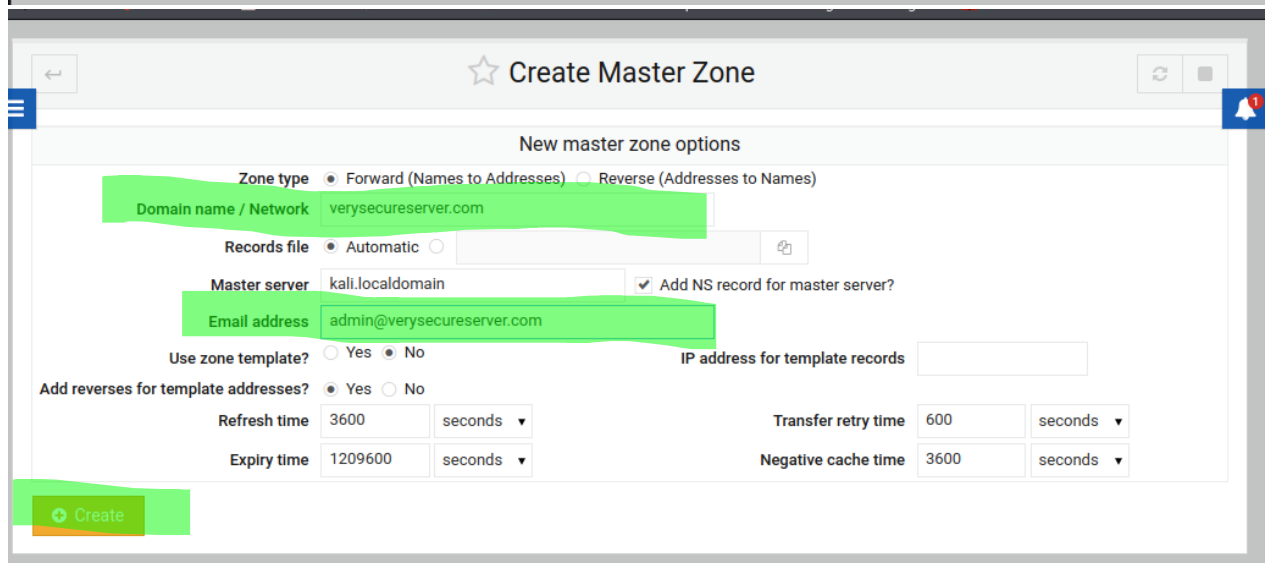
Step-5: Configure DNS server in Linux

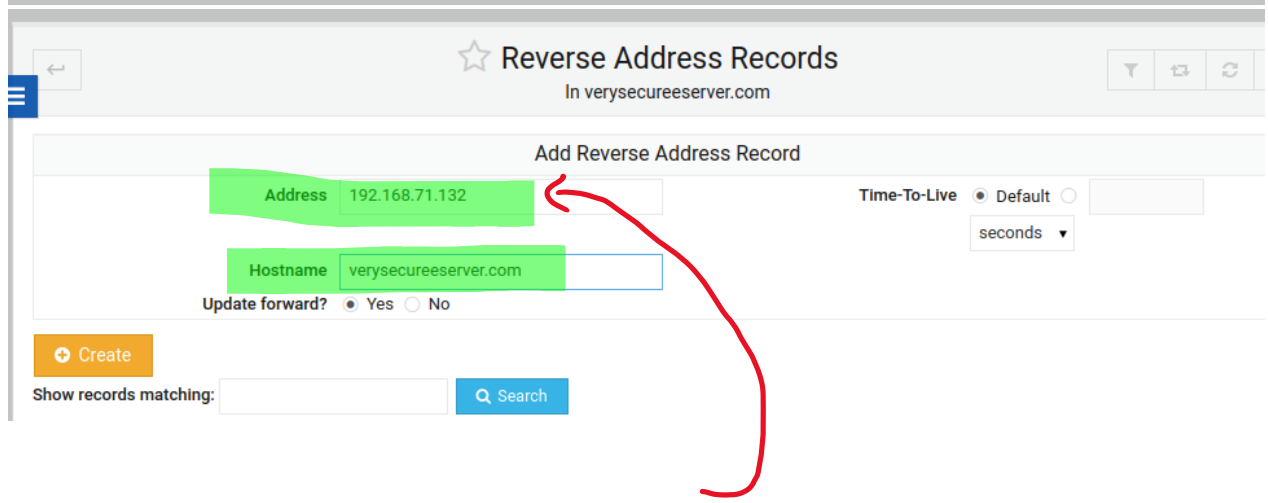


In Kali 2 machine Log into the Webmin and refresh the page. Now, we find BIND DNS Server in the servers.

- Start the BIND DNS Server and Do the following:

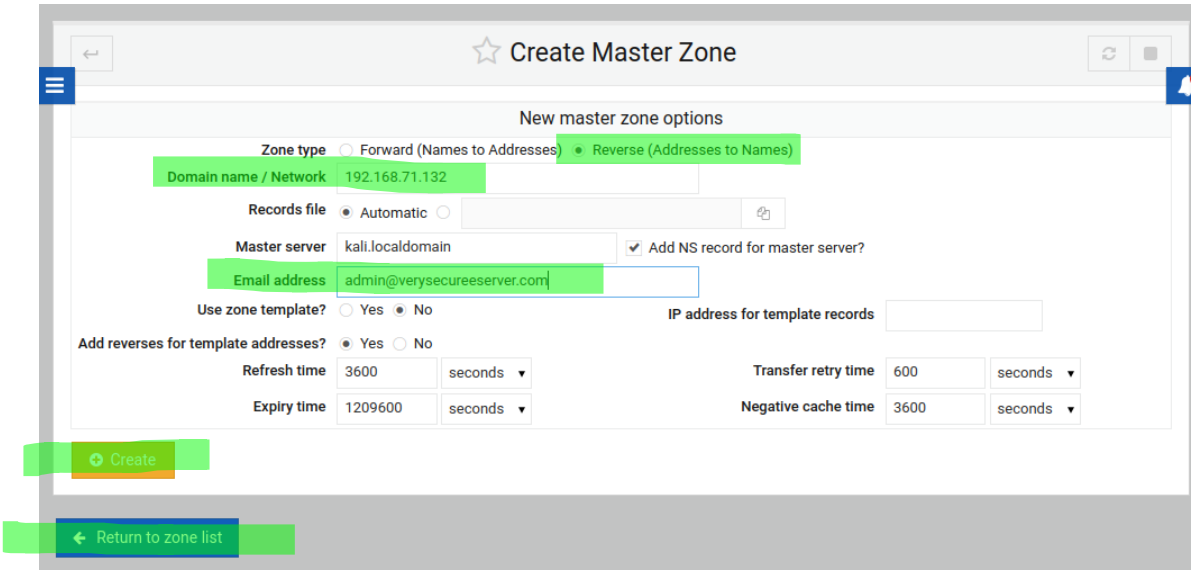






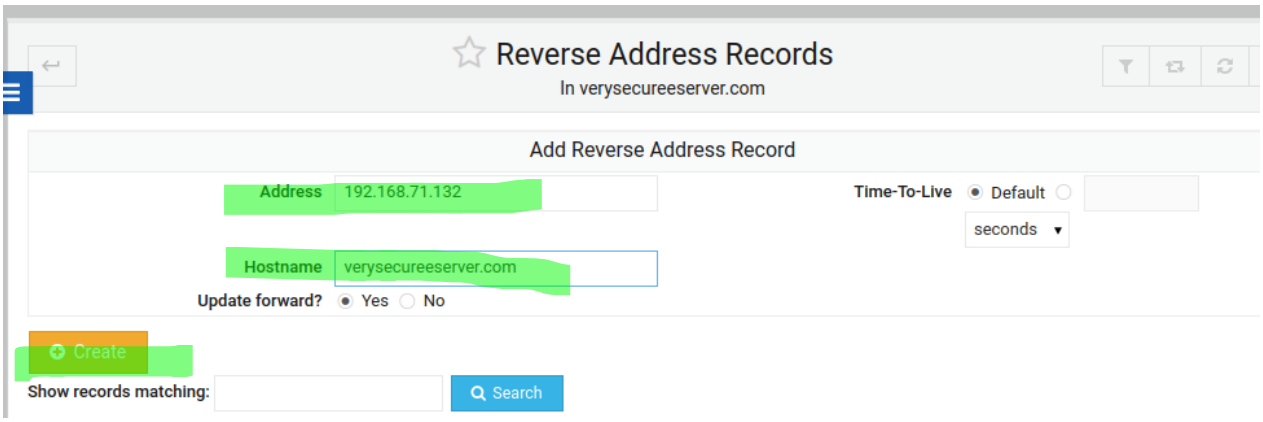
Put the Kali 2 machine's Ip address in the address bar.

- Again, go to create master zone and put info for reverse address.

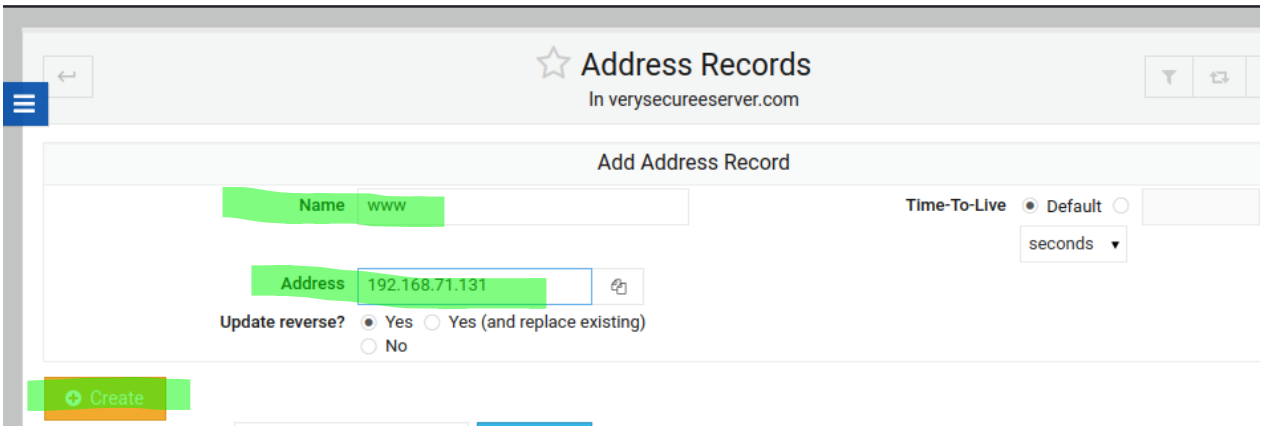


- Every time we create and save this steps we return to zone list afterwards.

Now edit option arrives for reverse address.



- Now to add Xampp address (kali 1 Ip address) we go to Master server and then click on Address and put the Ip of Kali 1 in the address and www in Name section.



- ✓ Write the following command on the Kali 2 machine Terminal (the address of client machine and the name of the server) it shows the Ip address and Domain name of the Server as it finds the information from the DNS Server.

  - nslookup 192.168.71.132
  - nslookup verysecureserver.com

References:

- https://www.youtube.com/watch?v=FynQxz8eysY
- https://www.youtube.com/watch?v=iUp3SChgZTo&list=LL&index=2&t=1238s
- https://www.youtube.com/playlist?list=PLtBXzZRmf04JnfeQO4fStxDSW4NlX5nxu