1-Configure SSL (https) site in apache2

* To configure apache for SSL

1-Generate key pairs (public, private)

* + Generate private key

$ openssl genrsa -out mykey.priv 2048

* + Generate public key

$ openssl rsa -in mykey.priv -pubout > mykey.pub

* + Secure private key

$ chmod o-r mykey.priv

2-Generate CSR

$ openssl req -new -key mykey.priv -out mycsr.csr

Text

Description automatically generated

3-Pay for the certificate or use self-signed certificate

$ openssl x509 -req -days 365 -in mycsr.csr -signkey mykey.priv -sha256 -out mycert.crt

4-Configure apache2 for SSL

* + Enable apache for ssl

$ sudo a2enmod ssl

* + Configure the SSL virtualhost /etc/apache2/sites-available/default-ssl.conf

Change certifictes paths SSLCertificateFile,SSLCertificateKeyFile

$ sudo cp mycert.crt /etc/ssl/certs

$ sudo cp mykey.priv /etc/ssl/private

$ sudo nano /etc/apache2/sites-available/default-ssl.conf

Text

Description automatically generated

* + Enable SSL Site

$ sudo a2ensite default-ssl

* + Restart

$ sudo service apache2 restart

A screenshot of a computer

Description automatically generated with medium confidence Graphical user interface, text, application, email

Description automatically generated

2-SQL MAP to apply sql injection

Running an SQL injection attack scan with sqlmap:

$ sqlmap.py -u “<URL>” --batch --banner

A small change in the command will run the same battery of tests but by using a POST as a test method instead of a GET.

Try the following command:

$ sqlmap.py -u “<URL>” --data=“id=1” --banner

Password cracking with sqlmap

$ sqlmap.py -u “<URL>” --batch --password

Get a list of databases on your system and their tables

$ sqlmap.py -u “<URL>” --batch --dbs

References:

<https://www.comparitech.com/net-admin/sqlmap-cheat-sheet/>

<https://cdn.comparitech.com/wp-content/uploads/2021/07/sqlmap-Cheat-Sheet.pdf>

<https://www.geeksforgeeks.org/use-sqlmap-test-website-sql-injection-vulnerability/>