Setting up an SSL Certificate on a local machine

*Note: this setup is for testing a certificate and proxy. The certificate is imported to the firefox browser on the same machine that is running the OWASP ZAP client.

1. Downloaded OWASP ZAP on Windows or zaproxy on ubuntu
2. Generated a Root CA Certificate through ZAP (uses OpenSSL), downloaded file
3. In Firefox, imported CA certificate (settings > certificates > authorities > import root CA)
4. In Firefox, set up a manual proxy configuration. Manual Proxy configuration, use proxy for HTTPS, HTTP proxy: 127.0.0.1:8080.
5. In ZAP, set up a local proxy. IP: localhost, port 8080.
6. In ZAP, traffic starts showing up from traffic from firefox client

Non-gui Version - Generating OpenSSL rootCA Certificate

Generating an OpenSSL certificate for our non-gui operating system
Have to install openSSL if you don't already it
1. Generate a private key
openssl genpkey -algorithm RSA -out rootCA.key
2. Generate the root CA certificate
openssl req -key rootCA.key -new -x509 -out rootCA.crt
This command will prompt you to enter information about your root CA, such as the common name (CN), the country (C), and the organization (O). Be sure to provide accurate and relevant information.
3. verify this root CA certificate: openssl x509 -noout -text -in rootCA.crt
4. the rootCA.crt file is the root CA certificate

Using the rootCA.crt, we can now put that into the proxy. To put the certificate into bettercap

1. Generate certificates like above.
2. openssl pkcs12 -export -in myCA.crt -inkey rootCA.key -out server.p12
    a. It will ask for a password, just [Enter] [Enter] for an empty password
3. Transfer that .p12 over to the target PC
4. Install that into the firefox browser. Go to the settings, certificate, import.
    a. Enter the password from before here, in my case it was empty

5. Run the above commands on bettercap to install the cert, key, and turn on the proxy
6. Set up the proxy on the firefox browser. Settings, proxy, IP address to the bettercap instance, and port is 8080

*Note: Apache2 likes to install its index.html page into /var/www/html as index.html. Nginx probably does some sort of wildcard, so it continues to use that apache2 html, instead of its own. So if you remove apache2, manually remove that html too.

Generating rootCA certificate and importing into NGINX

https://www.digitalocean.com/community/tutorials/how-to-create-a-self-signed-ssl-certificate-for-nginx-in-ubuntu-16-04

Creating and installing rootCA certificate for NGINX

1. Sudo apt install nginx
2. sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/nginx-selfsigned.key -out /etc/ssl/certs/nginx-selfsigned.crt
3. Cd into /etc/ssl to see the selfsigned certs
4. sudo nano /etc/nginx/snippets/self-signed.conf
5. Put both of the following lines into that file, save, close
    a. ssl_certificate /etc/ssl/certs/nginx-selfsigned.crt;
       ssl_certificate_key /etc/ssl/private/nginx-selfsigned.key;
6. sudo nano /etc/nginx/snippets/ssl-params.conf
7. Insert the following into this file

```
# from https://cipherli.st/
# and
https://raymii.org/s/tutorials/Strong_SSL_Security_On_nginx.html

ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
ssl_prefer_server_ciphers on;
ssl_ciphers "EECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH";
ssl_ecdh_curve secp384r1;
ssl_session_cache shared:SSL:10m;
ssl_session_tickets off;
ssl_stapling on;
ssl_stapling_verify on;
resolver 8.8.8.8 8.8.4.4 valid=300s;
resolver_timeout 5s;
# Disable preloading HSTS for now.  You can use the commented out
header line that includes
# the "preload" directive if you understand the implications.
```

```
#add_header Strict-Transport-Security "max-age=63072000;
includeSubdomains; preload";
add_header Strict-Transport-Security "max-age=63072000;
includeSubdomains";
add_header X-Frame-Options DENY;
add_header X-Content-Type-Options nosniff;

ssl_dhparam /etc/ssl/certs/dhparam.pem;
```

8. Backup the current version of the site: sudo cp /etc/nginx/sites-available/default /etc/nginx/sites-available/default.bak
9. sudo nano /etc/nginx/sites-available/default