Evan Smith

CSE 644 Internet Security

Lab 6

**Task 1 - Becoming a Certificate Authority (CA):**

I copied the openssl.cnf file into my user location, and then updated the configuration: Text

Description automatically generated

The necessary directories were made, and the setup was created:

Text

Description automatically generated

**Observations:** Here we complete the setup of the VM in order to be a CA for future work. This is analogous to the role that companies like VeriSign that add trust to interactions. We control the seed and password for the RSA private key, which allows us to check authenticity of future clients’ public keys, which we are now able to generate.

**Task 2 - Creating a Certificate for SEEDPKILab2020.com:**

We create the certificate:

Text

Description automatically generated

And can display the key:

Text

Description automatically generated

We then generate a CSR for the fictional SEEDPKILab2020.com:

Text, letter

Description automatically generated

And now generate the certificate using the CA’s signature:

Text, letter

Description automatically generated

**Observations:** After the setup from Task 1, we are able to create a certificate for a client site using openssl. The personal/business information required to make a certificate was interesting to me, in that we often separate websites and communication from their “owners”, but in many places, such as here, that information is integral to trust and security. Technically, there was an additional step here, which was to broaden the matching parameters of the openssl config file such that the CA and client states did not have to match, by using keyword policy\_anything.

**Task 3 – Deploying Certificate in an HTTPS Web Server:**

We edit the etc/hosts file to add out localhost mapped to the fake new site and can launch our webserver using openssl:

Text

Description automatically generated

But when attempting to view it on Firefox we see this error:

Graphical user interface, text, application, email

Description automatically generated

I added the ca.crt used to make my CA into Firefox (company is N/A):

Graphical user interface, text, application, email

Description automatically generated

After making this change, we are able to access the site with no warnings:

Text

Description automatically generated

We now modify a single byte of server.pem and restart the server. Here we change the highlighted bit from 59 to 60:

Graphical user interface, text, application

Description automatically generated

Once the openssl server was restarted, we see the following error:

Graphical user interface, text, application, email

Description automatically generated

With the file restored, the server starts up as before.

If we attempt to connect to the analogous site <https://localhost:4433>, we see this error:

Graphical user interface, text, application, email

Description automatically generated

This shows that the certificate in use is secured by the specific registered site name, and that given certificate is not valid for any other name, including those that map to the same place via DNS.

**Observations:** In this step, we see how CAs like VeriSign work in practice. Their certificates are mainstream enough to be contained within distributions of browsers like Firefox. Once the certificate is stored there, sites that use their public key can be validated. For our case, we needed to add the CA certificate directly, which allowed our localhost-hosted site to be validated.

**Task 4- Deploying Certificate in an Apache-Based HTTPS Website:**

We can alter the default-ssl.conf file to add a test website:

Text

Description automatically generated

We then create the directory referenced at var/www/seedpkilab2021 to place a simple html page:

Graphical user interface, text, application

Description automatically generated

With the files setup, we verify that the configuration is valid and start the Apache server:

Text, letter

Description automatically generated

After the server is launched, we can successfully access the site:

Graphical user interface, text, application, email

Description automatically generated

**Observations:** While the openssl server is primarily for gutchecks and basic testing, this task shows that we can use our custom certificate to secure a more fully fleshed out server with Apache. By linking the same certificate keys that were used for the rest of this lab into the Apache configuration, the rest of the server setup is in line with normal webserver development.