Exercise 7.5: Create and test a self-signed SSL certificate

Use the following information to create a self-signed certificate.

- Private-key pass phrase: `this is a long passphrase`
- Country Name: `US`
- State Name: `Awesome`
- Locality Name: `Awesometown`
- Organization Name: `Example Incorporated`
- Organizational Unit Name: `IT`
- Common Name: `ipvhost.example.com` where X is a unique number to your classroom or lab.
- Email Address: `admin@example.com` where X is a unique number to your classroom or lab.

Solution 7.5

1. Backup the original private key, if one exists.
   - On **CentOS**:
     ```
     # mv /etc/pki/tls/private/localhost.key /etc/pki/tls/private/localhost.key.orig
     ```
   - On **Ubuntu**:
     ```
     # mv /etc/ssl/private/ssl-cert-snakeoil.key \
          /etc/ssl/private/ssl-cert-snakeoil.key.orig
     ```
   - On **OpenSUSE**: There is no key by default so nothing needs to be backed up.
2. Create a new private key
   - On **CentOS**:
     ```
     # /usr/bin/openssl genrsa -aes128 2048 > /etc/pki/tls/private/localhost.key
     ```
   - On **OpenSUSE**:
     ```
     # /usr/bin/openssl genrsa -aes128 2048 > /etc/apache2/ssl.key/server.key
     ```
   - On **Ubuntu**:
     ```
     # /usr/bin/openssl genrsa -aes128 2048 > /etc/ssl/private/server.key
     ```
3. Create a new self-signed SSL certificate
   - On **CentOS**:
     ```
     # /usr/bin/openssl req -utf8 -new -key /etc/pki/tls/private/localhost.key -x509\
                          -days 365 -out /etc/pki/tls/certs/localhost.crt -set_serial 0
     ```
   - On **OpenSUSE**:


```
# /usr/bin/openssl req -utf8 -new -key /etc/apache2/ssl.key/server.key -x509 \
                  -days 365 -out /etc/apache2/ssl.crt/server.crt -set_serial 0
```

- On **Ubuntu**:

```
# /usr/bin/openssl req -utf8 -new -key /etc/ssl/private/server.key -x509 \
                  -days 365 -out /etc/ssl/certs/server.crt -set_serial 0
```

4. Update the **Apache** configuration (if needed)

   - On **Ubuntu**: Enable SSL vhost

     ```
     # ln -s /etc/apache2/sites-available/default-ssl.conf /etc/apache2/sites-enabled/
     ```

     Enable SSL module and configuration

     ```
     # ln -s /etc/apache2/mods-available/ssl.conf /etc/apache2/mods-enabled/
     # ln -s /etc/apache2/mods-available/ssl.load /etc/apache2/mods-enabled/
     ```

     Edit the file `/etc/apache2/sites-enabled/default-ssl.conf` and modify the paths for the key and crt files so they look like this:

     ```
     SSLCertificateFile /etc/ssl/certs/server.crt
     SSLCertificateKeyFile /etc/ssl/private/server.key
     ```

     Note: You may have to comment out the directives **SSLSessionCache** and **SSLSessionCacheTimeout** from the **/etc/apache2/mods-enabled/ssl.conf** file.

   - On **OpenSUSE**: Enable SSL vhost

     ```
     # cp /etc/apache2/vhosts.d/vhost-ssl.template /etc/apache2/vhosts.d/vhost-ssl.conf
     ```

     Enable the SSL server module, edit the file `/etc/sysconfig/apache2` and add the string "SSL" to the variable `APACHE_SERVER_FLAGS` so it looks like this:

     ```
     APACHE_SERVER_FLAGS="SSL"
     ```

   - On **CentOS**:

     There are no configuration changes needed.

5. Restart **Apache** and test your new certificate. You may have to add `ipvhost.example.com` to your `/etc/hosts` file.

   - On **CentOS**:

     ```
     # systemctl restart  httpd
     ```

   - On **Ubuntu** or **OpenSUSE**:

     ```
     # systemctl restart apache2
     ```

