# INFA723 Cryptography and Network Security

# Lab 9 Use OpenSSL to Generate a Self-Signed Certificate

# 1 Introduction

OpenSSL toolkit provides a command line tool and a crypto library used for a wide range of crypto operations. In this exercise, we are going to learn how to use OpenSSL to generate a self-signed certificate. Note that for all the functions implemented in the lab, you can find a way to implement them using the OpenSSL crypto library.

A lab package has been created for the lab Labs9-10.zip. Go to the class website and download the lab package. Assume your Cygwin installation folder is c:\cygwin. Unzip the package to your Cygwin home folder, e.g., C:\cygwin\home\user name.

The lab demonstrates two options to create certificates. The first option is to create a certificate request and send the certificate request to a third party to get it signed. The second option is to create a self-signed certificate for testing purpose.

The private key and self-signed certificate will also be used in Lab 10 to demonstrate how to use SSL/TLS to secure communication between a client and a server.

# 2 Objectives

* Learn the process how to create a certificate request
* Create a self-signed certificate to test

# 3 Use OpenSSL to Generate a Self-Signed Certificate

1. Go to Lab9 folder.
2. Certificates are related to public key cryptography by containing a public key. With OpenSSL, public keys are easily derived from private keys, so before you create a certificate or a certificate request, you need to create a private key. Create a 1024 bits private key private.key.

**$ openssl genrsa -out private.key 2048**

1. Creating a certificate request. To create a certificate, you need to start with a certificate request (or, as some certificate authorities like to put it, "certificate signing request", since that's exactly what they do, they sign it and give you the result back, thus making it authentic according to their policies). A certificate request can then be sent to a certificate authority to get it signed into a certificate, or if you have your own certificate authority, you may sign it yourself, or if you need a self-signed certificate (because you just want a test certificate or because you are setting up your own CA). The certificate request is created like this:

**$ openssl req -new -key private.key -out cert.csr**

**You are about to be asked to enter information that will be incorporated into your certificate request.**

**What you are about to enter is what is called a Distinguished Name or a DN. There are quite a few fields but you can leave some blank. For some fields there will be a default value, If you enter '.', the field will be left blank.**

**-----**

**Country Name (2 letter code) [AU]:US**

**State or Province Name (full name) [Some-State]:SD**

**Locality Name (eg, city) []:Madison**

**Organization Name (eg, company) [Internet Widgits Pty Ltd]:DSU**

**Organizational Unit Name (eg, section) []:BIS**

**Common Name (e.g. server FQDN or YOUR name) []:INFA723**

**Email Address []:yong.wang@dsu.edu**

**Please enter the following 'extra' attributes**

**to be sent with your certificate request**

**A challenge password []:dakota**

**An optional company name []:**

You can adjust the parameters based on your input. Now, cert.csr can be sent to the certificate authority. You can also use the extra argument '-outform' followed by the keyword for the format to use (see another HOWTO <formats.txt?>).

When the certificate authority has then done the checks the need to do, they will hand over your new certificate to you.

1. Creating a self-signed test certificate. If you don't want to deal with another certificate authority, or just want to create a test certificate for yourself. This is similar to creating a certificate request, but creates a certificate instead of a certificate request. This is NOT the recommended way to create a CA certificate, see ca.txt.

**$openssl req -new -x509 -key private.key -out cacert.cer -days 1095**

**You are about to be asked to enter information that will be incorporated**

**into your certificate request.**

**What you are about to enter is what is called a Distinguished Name or a DN.**

**There are quite a few fields but you can leave some blank**

**For some fields there will be a default value,**

**If you enter '.', the field will be left blank.**

**-----**

**Country Name (2 letter code) [AU]:US**

**State or Province Name (full name) [Some-State]:SD**

**Locality Name (eg, city) []:Madison**

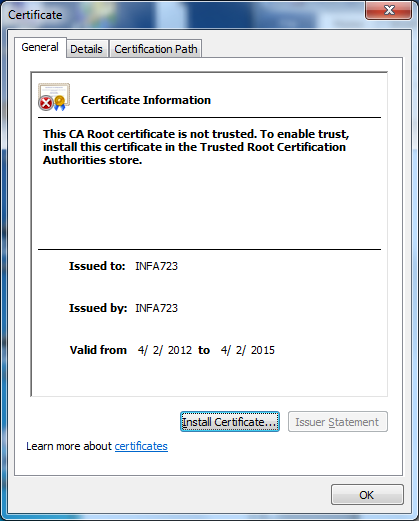
**Organization Name (eg, company) [Internet Widgits Pty Ltd]:DSU**

**Organizational Unit Name (eg, section) []:BIS**

**Common Name (e.g. server FQDN or YOUR name) []:INFA723**

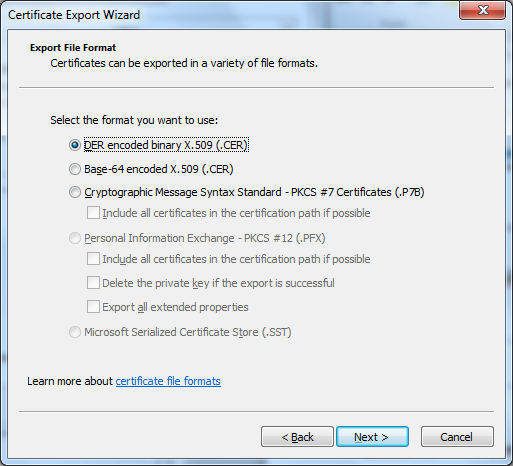
**Email Address []:yong.wang@dsu.edu**

1. In a windows explorer, double click the certificate you just create and a certificate a window will be popped up on windows.



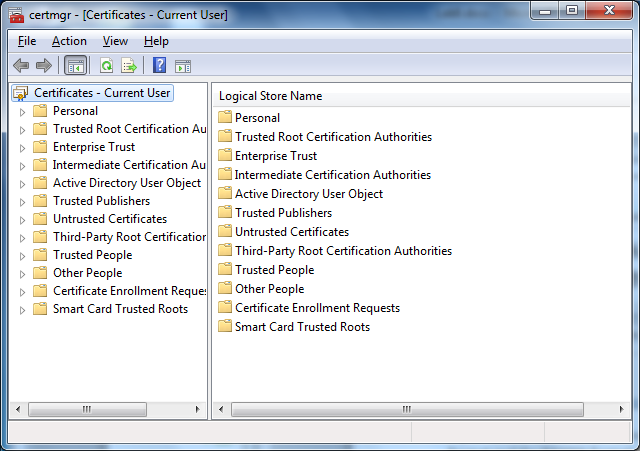
Click each tab to see how the certificate matching with the information just provided.

1. The certificate is saved in BASE64 format. You can convert to DER format.
   1. Click “Details” tab;
   2. Click “Copy to File” button in the window. A certificate Export Wizard will be popped out.
   3. Click “DER encoded binary X.509 (.CER)” option and follow the wizard to save the certificate in DER format.



If you have an editor which can view the file in ascii code, open the certificate in both formats, BASE64 and DER, to see the difference of these two formats.

1. Use windows certificate manager to manage windows certificates. Open Certificate Manager by clicking the **Start** button, typing **certmgr.msc** into the **Search** box, and then pressing enter. Check how windows manage its certificates.



# 4 Questions (20points)

1. (10 points) when you browse a web site using https protocol, warning message as shown in Figure 1 sometimes pops out. Explain why this can happen and what cautions you should use to procced.

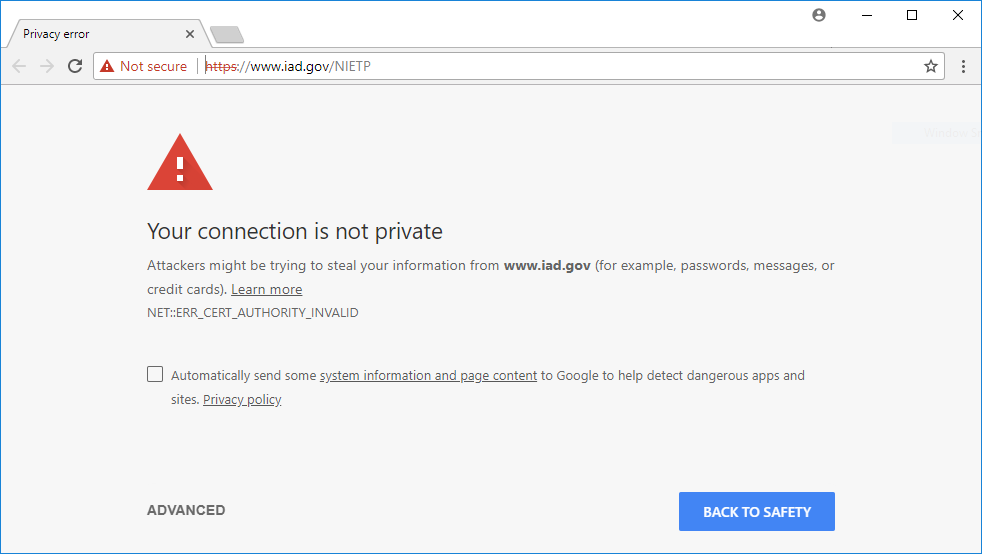


Figure 1 Warning Message from Chrome

2. (10 points) Let’s Encrypt (https://letsencrypt.org/about/) is a free, automated, and open certificate authority. Read “How It Works) (https://letsencrypt.org/how-it-works/) and briefly explain how domain validation (as shown in Figures 2 and 3) works using Lets Encryption CA system.

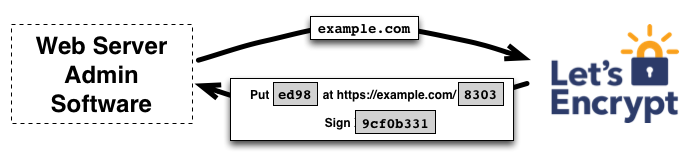


Figure 2 Domain Validation

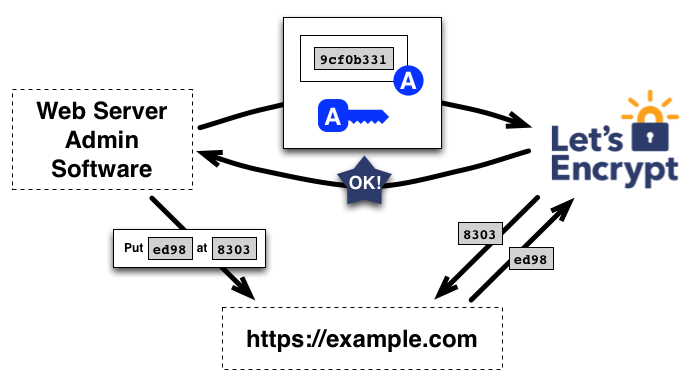


Figure 3 Domain Validation