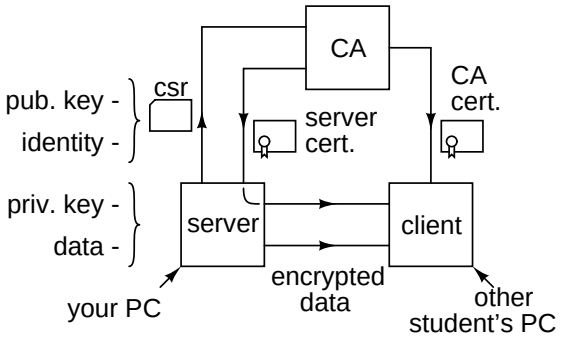
Public Key Infrastructure

Introduction

In this lab you will use public/private key pairs and signed certificates to configure a web server and web browser to enable authenticated and encrypted communication between them. You will use Wireshark to compare client-server exchanges using secure and unsecured HTTP connections.

The following diagram summarizes the operations involved:



The sequence of steps is:

- add a certificate authority's (CA) certificate to your web browser so your browser will trust certificates signed by this CA,
- install a web server on your PC and use Wireshark to view the contents of a page served over an unsecured connection,
- create a public/private key pair,
- create a Certificate Signing Request (CSR) containing the public key and your identity and have the CA sign the CSR to create the server's certificate,
- install the signed certificate and private key and configure the web server to use the server certificate and secret key when serving web pages, and
- use a web browser to access a web page over a secure ("HTTPS") connection and use wireshark to compare with the unsecured connection.

For this lab the instructor will act as a certificate authority (CA). The instructor will provide you with a certificate to temporarily install in your browser and will sign your server certificate. For a public web server these functions would be performed by a widely-trusted CA such as `http://letsencrypt.org` whose (free!) certificates are trusted by most browsers.

Each student will configure their own web server and browser. You will need to work in pairs to test both the browser and the server because you won't be able to test your own server[1]. Although you will work in pairs, you will submit your own lab report.

Procedure

Install the CA's Root Certificate

The instructor will act as the Certificate Authority (CA) for this lab. Download a certificate containing the CA's public key (ca.crt) from the course web site. This certificate will be signed by the CA itself (a "self-signed" certificate[2] rather than by a CA whose certificate is already installed in your browser. Thus you will need to install this CA's certificate in your browser so that certificates signed by this CA will be trusted.

To keep things simple we will use the Firefox browser as the client for this lab. To install the CA's certificate, select Settings (☰) / Options / Advanced / Certificates / View Certificates... Select the Authorities tab and click Import.... Select the CA's certificate (the `.crt`) file and click Open then select "Trust to identify websites" and click OK.

Disable Services on Port 443

VMWare's Remote Access service on the lab computers listens on the TCP port normally used for secure

[1] The packet capture installed on the lab PCs cannot capture from the loopback adapter.

[2] Becoming a widely-trusted CA is impractical unless you want to do it as a business

HTTP (443). You must disable this service to allow a secure HTTP server to run. Run the "Services" application (or run the `services.msc` command), find the VMWare Workstation Server service and double-click it to see its properties. Change it's Startup Type to "Manual" and click "Stop" to terminate it.

You will have to do this each time the PC reboots because the service is configured to start on boot.

Find Your PC's IP Address

Run Control Panel / Network and Sharing Center and click on the Ethernet interface connected to the BCIT network. Click on Details... to find the IPv4 Address of your computer. Record the address. It will probably be of the form 142.232.250.NNN (where NNN is a number specific to your PC). Double-check the IP address and give it to the student you're working with so they will be able to test your server.

Install Nginx

Download the stable Windows version of the nginx web server from the course web site. Extract the contents of the zip file to a convenient temporary directory such as a directory on the D: drive. Note that if you work off a flash drive you will not be able to remove it unless you shut down and restart nginx.

You must extract the files from the zip folder, you will not be able to edit the configuration file or add an HTML file if you run nginx from the zip file.

Examine the Insecure Connection

Use a text editor such as Notepad++ to create a file containing a friendly greeting and your full name and save it as the file `pkilab.html` in the `html` subdirectory of the directory were `nginx.exe` is located.

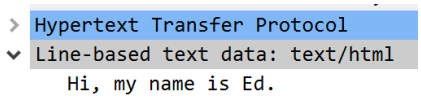
Open a command or powershell window in the directory were `nginx.exe` is located (from Windows Explorer use shift-right-click and select "Open PowerShell window here"). Run nginx by typing `.\nginx`.

Run Wireshark and start capturing.

Have the other student access `http://<yourIPaddress>/pkilab.html`. They should see a web page containing your greeting.

If at any point it is necessary to reload a page, press Control-F5 so the browser does not use a cached version of the page.

In the wireshark window filter on the `http` protocol. You should see a `GET /pkilab.html` request and a corresponding response with the `200 OK` response code. Expand the "Line-based text data" section. You should see the HTML corresponding the web page; something like:



Have the other student take a screen capture showing the two IP addresses and the text/html content. They will need this for their report.

Create Your Server's Private/Public Key Pair

Open a cygwin terminal ("cyglaunch") and run the following openssl command to generate a 2048-bit key pair:

```
openssl genrsa -out serverNNN.key 2048
```

where `NNN` is the last 3 digits of your PC's IP address as found above.

Create a CSR

Run the following command (all on one line) to generate a certificate signing request (CSR) called `serverNNN.csr`:

```
openssl req -new -key serverNNN.key
                 -out serverNNN.csr
```

The program will prompt you for identity information that will be included in the certificate. Use CA for country, BC for state, your full name for Organization, your server's full IP address for the Common Name, and leave the other fields blank.

Normally the identity information in the server CSR would include unambiguous identification of the server's owner (e.g. Microsoft Corporation) and the server's domain name (e.g. support.microsoft.com).

On the lab PCs your cygwin home directory will probably be under: `C:\cygwin64\home\A00`... and this is where you will probably find the `.key` and `.csr` files.

Have CA Sign Your CSR

Copy the `serverNNN.csr` file to a USB drive and give it to the lab instructor. This CSR contains only the public key and the identification information.

The instructor will examine the CSR to make sure the identification information is correct and will then sign it with the CA's private key to show others that the CA believes the public key in the certificate belongs to the individual named in the certificate. The instructor will put the signed certificate back on your USB drive in a file called `serverNNN.crt`.

Install Server Certificate and Private Key

Use a text editor to edit the `nginx.conf` file in the `conf` sub-directory of the directory were `nginx.exe` is located. Un-comment the `https` section at the bottom of the configuration file, starting at the `server` line.

Instead of `cert.pem` and `cert.key`, use the name of your signed server certificate (`serverNNN.crt`) and private key (`serverNNN.key`).

Copy these two files onto the `conf` sub-directory of the directory were `nginx.exe` is located.

Signal the server to reload its configuration file by opening another shell (command prompt) window in the directory were `nginx.exe` is located and running the command `.\nginx -s reload`.

Monitor the Secure Connection

In general, we cannot be certain that a connection is secure. However, we can check for compliance with security protocols. In this lab we will limit ourselves to verifying that a TLS connection is negotiated and to checking the identity and trust chain displayed by the browser.

Ask the other student to access the URL `https://<yourIPaddress>/pkilab.html`. As before, press Control-F5 if you need to reload the page.

If they get a security warning, determine if the problem is on the client side (e.g. the CA that signed the certificate is not trusted) or the server side (e.g. the certificate is not valid for that domain or IP address) and resolve the problem.

Once they are able to load the web page without errors, have them take one screen capture showing the URL and the page contents and a second screen capture showing the certificate details (click on the padlock / > / More Information / View Certificate).

Use the `tcp.port == 443` Wireshark display filter to view only secure HTTP connections. Take a screen capture showing the secure exchange between the client and server, starting with the initial SYN packet. Include only the summary information section, not the packet contents.

Report

Your report should include, in addition to the usual identification information, the names of the two students, the IP addresses of their lab PCs and the following screen captures:

- the Wireshark capture of the non-secured http response showing the two IP addresses and the text/html content
- the Wireshark capture of the TLS dialog for the secure connection (you won't be able to see the actual HTTP protocol exchange – it's secure!)
- a browser screen capture showing the URL and the web page contents (the other student's name) for the secure web page.
- a screenshot of the browser's certificate details page showing the other student's IP address in the Common Name field and their full name in the Organization field.

PKI Good Practices

In this lab we are not following good security practices that would help ensure that secret keys are kept secret. In particular, good practice requires using a hierarchy of certificates with different security levels and short durations so that when keys are compromised they can be revoked or allowed to expire.

High-value keys are typically generated, stored and used only in a dedicated "Hardware Security Module" (HSM) – a device designed to prevent disclosure of the keys, typically by detecting tampering and zeroing the keys.

To allow for disaster recovery, keys can be backed up using secret sharing – encryption that allows decryption by sets of people (e.g. any 2 of 3 trusted staff).