Transport Layer Security

C5.1. When we use SSL, why can't SSL automatically verify the common name for us? Why do we have to specifically tell SSL to do so?

C5.2. In the `tls_client.py` program shown in the book, if the common name check fails, which line will fail?

C5.3. What are the major differences between a TLS client and TLS server? Please only list the differences related to TLS.

C5.4. In the TLS server program (Python), which line does the following?

(a) sending out the server's certificate,

(b) asking the server operator to type the password that is used to protect the server's private key,

(c) decrypting the data from the client,

(d) encrypting the data sent to the client.

C5.5. If a company `example.com` wants to use the same certificate for several of its severs, such as `www.example.com`, `mail.example.com`, `vpn.example.com`, etc., how can they do that?

C5.6. A client program usually uses the server's certificate to authenticate the server. How do servers authenticate clients typically?

C5.7. Are HTTPS and HTTP two different protocols? What are their differences and what do they have in common?

C5.8. When we type `https://www.example.com/getinfo.php` in our browser, the `getinfo.php` program on the server will be executed. Since we are using HTTPS, our browser will establish an SSL connection with the server. On the server side, who is responsible for establishing the SSL connection, conducting the SSL handshake protocol with the browser, and providing the server certificate? Is it the `getinfo.php` program?

C5.9. We run our TLS client program to communicate with `https://www.facebook.com`. We want to set up an MITM proxy between our TLS client program and the web server, so the proxy can intercept, view, or even modify the traffic between our TLS client program and Facebook. We do this for debugging purposes. Please describe what needs to be done on the client program to make this work (we are not allowed to change the TLS client program).