Exercise 5: Security analysis on TLS:

(c) Assume the key establishment algorithm is RSA, and the client authentication is not conducted, that is, message CertificateVerify is not sent. Try to identify an attack which hijacks the session by sending an attacker-generated "pre-master secret" to the server, where the messages Finished can carry along without being detected by either the client or the server.

An attacker can use a tool such as Wireshark to monitor network traffic between the client and server. The following is a sample scenario that entails its usage:

I. Client sends plaintext message ClientHello to the server. The message is not encrypted – thus the attacker knows client random and has knowledge of all the cipher suites requested.

II. Server sends plaintext message ServerHello to the client. The message is not encrypted – thus the attacker knows server random and has knowledge of the cipher suites that the server has chosen. The server will also send an SSL certificate as part of ServerHello, which the attacker can then use to contact the CA that issued the certificate.
   a. There is no ServerKeyExchange from the server under RSA
   b. There is no CertificateRequest given that client verification is not performed in this scenario

III. Attacker can actively intercept messages exchanged between the client and server.
   a. Under RSA, the attacker establishes the pre-secret master key α and encrypts it using the server's public key pk_S as follows: $E = (pk_S, \alpha)$ – this is generated as part of the ClientKeyExchange. The attacker is now capable of generating the Finished message on its side.

IV. The server and attacker have established a pre-message secret – the attacker also has knowledge of the client random and server random from part II, along with the cipher suite that was chosen by the server.
   a. Therefore, the attacker can generate the master secret and any other keying material necessary such as the client write key which is used to send messages to the server.

V. The master key is necessary to generate the Finished message – it will be used to generate the MAC key and the encryption key. The attacker will collect the ClientHello from the victim client, the server hello, and the ClientKeyExchange into a single message. This message is used with the MAC key to generate a tag. The message and tag are then combined to generate a payload which is encrypted using the encryption key, creating the Finished message.
   a. The attacker will send the ClientKeyExchange and Finished messages to the server – the server will respond with its Finished message directly to the attacker and the client will not have any knowledge of this message. Consequently, the attacker will have successfully hijacked the session and the Finished messages are generated without the server and client being aware of the attacker's presence.

(d) Explain why the attack identified in a) will not gain access to the server, if the client must enter a password before any further application data will be exchanged.

Clients are required to enter their password before application data can be transmitted by the server in TLS. This step is independent of the key establishment and authentication steps. Given that the client is the only party aware of the password, the attacker is unable to gain access to the data on the server without knowledge of that password. TLS was designed to establish a secure tunnel through which sensitive user data can be transmitted. Otherwise, sending the user's application data without requiring the user's password enables attackers to hijack a session with relative ease.

(e) Try to explain why key establishment algorithms RSA and DH cannot provide perfect forward secrecy.

Perfect forward secret can only be achieved if disclosure of long-term secret keying material does not compromise the secrecy of the exchanged keys from earlier sessions.

While using RSA for key transport under TLS, the pre-secret master key α is chosen by the client and encrypted using the server's public key pk_S such that only the server is able to decrypt the pre-secret master key. Though the pre-secret master key is newly chosen each session, the encryption is done using the same long-term key pk_S. Therefore, if the long-term key is compromised, attackers can decrypt any of the session keys that were exchanged in the current session, as well as in any previous session provided the encrypted key exchanges are stored. Consequently, RSA is unable to provide perfect forward secrecy.

Using DH, the client chooses a new key c for each session. The client computes $Y_c = g^c$ and sends Y_c to the server. The server does not choose a new key however, and instead the long-term key s is used to compute the pre-master secret key α at the client and server as $\alpha = g^{sc}$. Consequently, α is directly related to the long-term key s in each session and if this key is compromised, the attackers can decrypt any past or present session key as long as the ClientKeyExchange Y_cs are stored in advance. As a result, DH is also unable to provide perfect forward secrecy.