CHAPTER 17

SSL and TLS

(Solution to Odd-Numbered Problems)

Review Questions

1. Five services are provided by SSL or TLS: *fragmentation*, *compression*, *message integrity*, *confidentiality*, and *framing*.

3. TLS uses the PRF function to create the master secret from the pre-master secret. The first parameter (*secret*) is the pre-master secret; the second parameter (*label*) is the string "master secret"; the third parameter (*seed*) is the concatenation of the client random number and server random

5. TLS uses the PRF function to create key materials from the master secret. The first parameter (*secret*) is the master secret, the second parameter (*label*) is the string "key expansion", and the third parameter (*seed*) is the concatenation of the server random number and the client random number.

7. The following list the four protocols:

a. The Handshake Protocol uses messages to negotiate the cipher suite, to authenticate the server to the client and the client to the server if needed, and to exchange information for building the cryptographic secrets.

b. The ChangeCipherSpec Protocol defines the process of moving cryptographic parameters between the pending and active states.

c. The Alert Protocol is used to report errors and abnormal conditions.

d. The Record Protocol carries messages from the upper layer (Handshake Protocol, ChangeCipherSpec Protocol, Alert Protocol, or application layer).

9. TLS uses the Handshake Protocol defined for SSL with only two small changes in CertificateVerify and Finished messages:

a. In SSL, the hash used in the CertificateVerify message is the two-step hash of the handshake messages plus a pad and the master secret. in TLS the hash is only over the handshake messages.

b. In TLS, a pseudorandom function (PRF) is used to calculate two hashes used for the Finished message.

Exercises

11. The following table shows the size of the key-material in each case

	Client	Server	Client	Server	Client	Server	**Total**
	Auth.	Auth.	Enc.	Enc.	IV	IV	**Size**
a.	**1024**	**1024**	**0**	**0**	**0**	**0**	**2048**
b.	**1024**	**1024**	**0**	**0**	**0**	**0**	**2048**
c.	**1024**	**1024**	**56**	**56**	**64**	**64**	**2288**
d.	**1024**	**1024**	**168**	**168**	**64**	**64**	**2512**
e.	**1024**	**1024**	**56**	**56**	**64**	**64**	**2288**
f.	**1024**	**1024**	**168**	**168**	**64**	**64**	**2512**

We assume that the size of key for RSA authentication is 1024 bits although it can be 512 bits.We also assume that single DES uses a 56-bit bits and triple DES uses a key of 168 bits.

13. At first glance, it looks that TLS uses the premaster secret only once to create the master secret, but if we look more carefully at the data expansion function and the PRF function, we see that this calculation in TLS is more complex than the corresponding calculation in SSL. We believe the calculation in TLS is less efficient than the one in SSL.

15. Although TLS uses only one PRF function, the PRF function is made of two data expansion function and each expansion function is an iteration of two-stage HMAC calculation. Therefore, TLS also uses iteration to create variable-size key materials although it is not as explicit as the SSL in this issue.

17. Authentication keys, encryption keys, and IV's need to be created. The premaster and master secret do not need to be created again.

19. The calculation in TLS is more consistent to other standards for MAC calculation (padding the secret and exclusive-oring them with ipad or opad). We believe the efficiency of both methods is the same.

21. It is difficult to say which one is more efficient, but the one used in TLS looks more secure because it creates two different digests from the handshake message using two different algorithms (MD5 and SHA-1).

23.

a.

Key Material	=	**MD5 (M | SHA-1 ("A" | M | CR | SR)) |**
		MD5 (M | SHA-1 ("BB" | M | CR | SR)) |
		MD5 (M | SHA-1 ("…" | M | CR | SR)) |
		…

b.

**MAC = Hash (WriteSecret | pad-2 | Hash (WriteSecret | Pad-1 |
Sequence number | Compressed type | Compressed length
|Compressed fragment))**

c.

Hash Digest = Hash (M | pad-2 | Hash (Handshake message | M | Pad-1))

d.

Hash Digest = Hash (M | pad-2 | Hash (Handshake message | M | Pad-1))

e.

Expanded Secret = $\text{HMAC}_{\text{Secret}}$ ($\text{HMAC}_{\text{Secret}}$ (Seed) | Seed) |
$\text{HMAC}_{\text{Secret}}$ ($\text{HMAC}_{\text{Secret}}$ (X) | Seed) |
$\text{HMAC}_{\text{Secret}}$ ($\text{HMAC}_{\text{Secret}}$ (Y) | Seed) |
…

Where X = $\text{HMAC}_{\text{Secret}}$ (Seed), **Y** = $\text{HMAC}_{\text{Secret}}$ (X), …

f.

New Secret = MD5 (Label | Seed) ⊕ SHA-1 (Label | Seed)

g.

Master Secret = PRF (PM, "Master Secret", CR | SR)

h.

Key Material = PRF (M, "Key Expansion", SR | CR)

i.

Hash Digest = Hash (Handshake Message)

j.

**Hash Digest = PRF (M | Finished Label |
MD5 (Handshake Message) |
SHA-1 (Handshake Message))**

k.

HMAC = Hash (MAC Secret ⊕ opad |
Hash (MAC Secret ⊕ opad | X | Compressed Fragment)

X = Sequence number | Compressed type | Compressed version | Compress Length

25. The key size in SSL or TLS depends on the algorithm used for encryption. If an encryption algorithm with small key-size (such as single DES) is used, the protocol is less immune to brute-force attack. If an encryption algorithm with a large key size is used (such as 3DES), the protocol is more immune to brute-force attack.

27. The two protocol are equally immune to the man-in-the-middle attack. The immunity depends on the type of algorithm used for key exchange as discussed in Exercise 24.