Internet Infrastructure Security (COMP444)
Assignment Three

Each question carries six marks, unless stated otherwise.
** Due on 11 April 2014 **

Rocky K. C. Chang

March 21, 2014

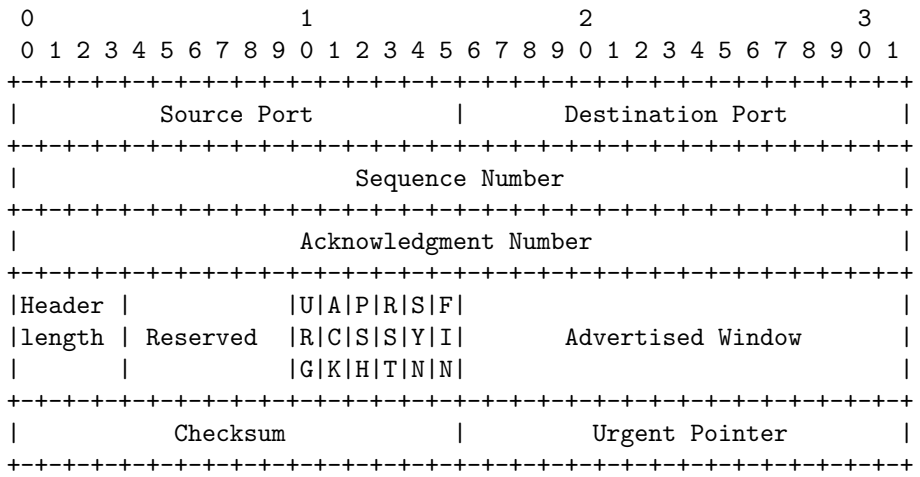
1. `[10 marks]` (TCP SYN in IPSec) Consider two routers that have established an IPSec tunnel SA between them. The SA is ESP with both encryption (AES-CBC-192) and message authentication (HMAC-SHA1-160). A host sends a TCP SYN segment without option which goes through the IPSec tunnel.

   (a) `[8 marks]` What is the length of the ESP packet for this message? Show your calculation clearly.

   (b) `[2 marks]` What is the next legitimate size of the ESP packet?

   You may assume the following:

   - The 20-byte IP header does not have option.
   - The TCP header is given below.
   - An IV is included at the beginning of the ESP payload.
   - A minimal ESP padding.



2. [6 marks] (ESP) Owing to the fact that block encryption and cryptographic hash functions generate discrete output sizes, an IPSec packet can take on only some legitimate sizes. In this question, the block size for encryption is 256 bits (e.g., using AES-256), and the hash's output size for message authentication is 512 bits (e.g., SHA-3). Determine whether the following two packet sizes are legitimate. Note that an IP packet is always on the 4-byte boundary, and the IP header is 20 bytes long.

   (a) [3 marks] The IPSec packet size (including the IP header) is 146 bytes.

   (b) [3 marks] The IPSec packet size (including the IP header) is 440 bytes.

3. [8 marks] (IKE main mode) Recall from slide 11 of Internet Key Exchange that there are six message exchanges between an initiator and a responder in the main mode. Identify the IKE message (1 to 6) that corresponds to each of the partial WireShark captures (excluding Ethernet, IP and other unnecessary information) below. As a result, there may be more than one IKE message that matches the partial WireShark capture. Provide sufficient evidence to support your answers.

   (a) [2 marks] The first capture is given in Figure 1.

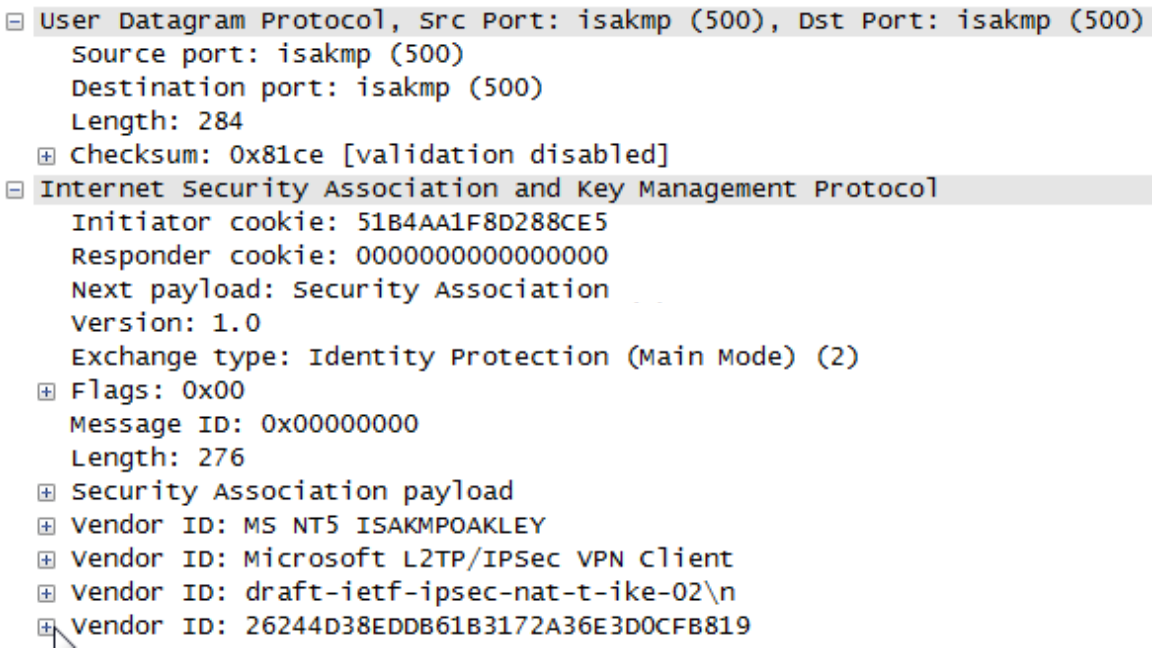


Figure 1: The first WireShark capture.

   (b) [4 marks] The second capture is given in Figure 2.

   (c) [2 marks] The third capture is given in Figure 3.

4. [6 marks] (IKEv1 aggressive mode) Recall from the IKE slides that the messages cannot be encrypted in the IKE aggressive mode. Therefore, the endpoint identities cannot be concealed.

```
⊟ User Datagram Protocol, Src Port: isakmp (500), Dst Port: isakmp (500)
    Source port: isakmp (500)
    Destination port: isakmp (500)
    Length: 240
  ⊞ Checksum: 0xc90e [validation disabled]
⊟ Internet Security Association and Key Management Protocol
    Initiator cookie: 51B4AA1F8D288CE5
    Responder cookie: C29F2926ADCFCE81
    Next payload: Key Exchange
    Version: 1.0
    Exchange type: Identity Protection (Main Mode) (2)
  ⊞ Flags: 0x00
    Message ID: 0x00000000
    Length: 232
  ⊟ Key Exchange payload
      Next payload: Nonce (10)
      Payload length: 132
      Key Exchange Data (128 bytes / 1024 bits)
  ⊞ Nonce payload
  ⊞ NAT-D (draft-ietf-ipsec-nat-t-ike-01 to 03) payload
  ⊞ NAT-D (draft-ietf-ipsec-nat-t-ike-01 to 03) payload
```

Figure 2: The second WireShark capture.

```
⊟ User Datagram Protocol, Src Port: isakmp (500), Dst Port: isakmp (500)
    Source port: isakmp (500)
    Destination port: isakmp (500)
    Length: 76
  ⊞ Checksum: 0xd0e7 [validation disabled]
⊟ Internet Security Association and Key Management Protocol
    Initiator cookie: 51B4AA1F8D288CE5
    Responder cookie: C29F2926ADCFCE81
    Next payload: Identification
    Version: 1.0
    Exchange type: Identity Protection (Main Mode) (2)
  ⊞ Flags: 0x01
    Message ID: 0x00000000
    Length: 68
    Encrypted payload (40 bytes)
```

Figure 3: The third WireShark capture.

However, Figure 4 shows another version of the IKE aggressive mode which can encrypt the endpoint identities for the digital signature authentication.

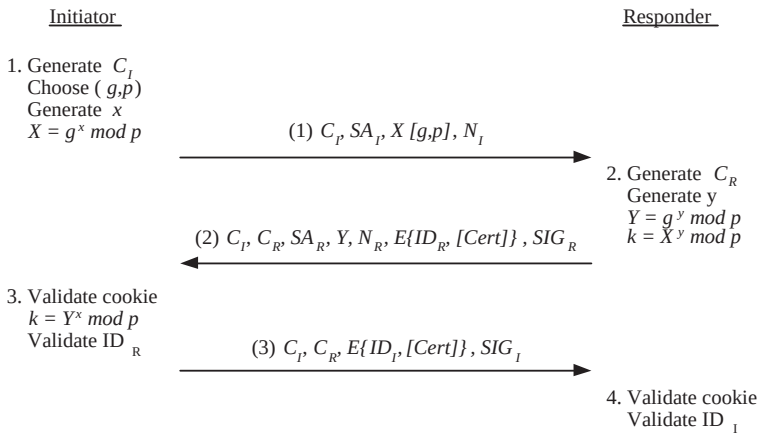


Figure 4: Modified public-key digital signature authentication in the IKE aggressive mode.

(a) `[3 marks]` Explain why the new version can allow both sides to conceal their identities.

(b) `[3 marks]` However, similar changes may not be applicable to other authentication methods. Explain why this cannot be done for the pre-shared key operating in the aggressive mode.

5. `[8 marks]` (TLS/SSL) Please answer the following questions concerning TLS/SSL.

(a) `[2 marks]` If the ServerHello message is lost in the network but the later retransmission is successful, how does this message loss affect the TLS session?

(b) `[2 marks]` Suppose that the ServerHello is sent in two IP/TCP packets, but these two packets arrive at the client side in a different order (the later packet arrives first). How does this packet reorder affects the TLS session?

(c) `[3 marks]` If an attacker captures a copy of an old TLS record and sends it out later. How does this old record affect the TLS session.

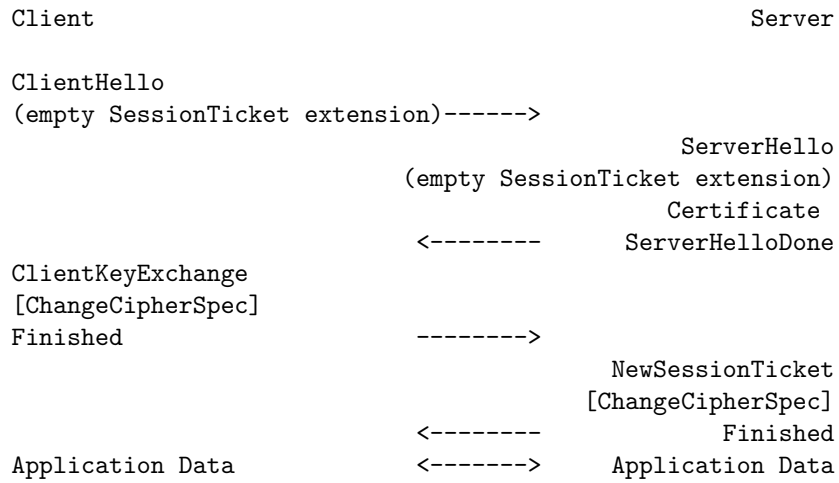
(d) `[3 marks]` If an attacker sends out a TLS record for which the TCP sequence number is considered new to the TCP receiver (at the TLS end host), how will this "new" TLS record affect the TLS session? You may assume that this new TCP segment can be buffered at the receiver.

6. `[6 marks]` (Attacking TLS) Consider the following two attacks on TLS.

(a) `[3 marks]` If an attacker could decode the secret keys used in a TLS connection between a client and a server. Could the secret keys help the attacker decode the secret keys in another TLS connection between them?

(b) `[3 marks]` If an attacker could decode the secret keys used in a TLS connection between a client and a server. Could the secret keys help the attacker decode the master secret in their TLS session?

7. (TLS session reuse) We re-visit the TLS session reuse question in the assignment. Recall that:



(a) [3 marks] The protocol specifies that "in the case of a full handshake, the server MUST verify the client's Finished message before sending the ticket." What is the main reason for this requirement?

(b) [3 marks] In the assignment, we said that "the server must encrypt the ticket and add an MAC to protect its integrity. Both keys must be secured by the server." Could an attacker establish a TLS session with the server by replaying a copy the `NewSessionTicket` message captured from an earlier TLS session, assuming that the servers' keys for protecting the ticket have not been compromised? Explain your answer.