1. [10 points]

Solution
See exam 1.

2. [20 points]

Solution
See exam 1.
3. [10 points] Company xLtd has principals \( X, A_1, A_2, \ldots \), where \( X \) issues certificates for the \( A_i \)'s, and is their trust anchor. Company yLtd has principals \( Y, B_1, B_2, \ldots \), where \( Y \) issues certificates for the \( B_i \)'s, and is their trust anchor. One day, xLtd acquires yLtd. You are to obtain a new PKI for the new xLtd. Parts a and b are independent.

**Part a**
Modify the old PKIs to obtain a new PKI in which \( X \) is the sole trust anchor for all \( A_i \)'s and \( B_i \)'s; minimize the number of new certificates.

Give the certificate chain that \( A_1 \) needs to get the public key of \( B_1 \) in the new PKI.

Give the certificate chain that \( B_1 \) needs to get the public key of \( A_1 \) in the new PKI.

**Part b**
Modify the old PKIs to obtain a new PKI in which \( X \) is the sole trust anchor for all \( A_i \)'s, and \( Y \) is the sole trust anchor for all \( B_i \)'s; minimize the number of new certificates.

Give the certificate chain that \( A_1 \) needs to get the public key of \( B_1 \) in the new PKI.

Give the certificate chain that \( B_1 \) needs to get the public key of \( A_1 \) in the new PKI.

**Solution for part a**
Modifications to old PKIs:

- At every \( B_i \), add \( X \)'s public key and remove \( Y \)'s public key. [2 points]
- \( X \) issues a certificate for \( Y \) [1 point]

(Max 1 point if \( Y \) issues certificates for all \( A_i \)'s. −1 point if \( Y \) is online.)

Certificate chain for \( A_1 \) to \( B_1 \): \([X, Y, B_1]\) [1 point]

Certificate chain for \( B_1 \) to \( A_1 \): \([X, A_1]\) [1 point]

**Solution for part b**
Modifications to old PKIs:

- \( X \) issues a certificate for \( Y \) [1 point]
- \( Y \) issues a certificate for \( X \) [2 points]

Certificate chain for \( A_1 \) to \( B_1 \): \([X, Y, B_1]\) [1 point]

Certificate chain for \( B_1 \) to \( A_1 \): \([Y, X, A_1]\) [1 point]
4. [10 points]

Part a

Client $A$ and server $B$ interact over TCP/IP. The client is at TCP port $p$ and IP address $F$. The server listens at TCP port $q$ and IP address $G$.

There are two nodes, $J$ and $K$, on the IP path between $F$ and $G$. An IP packet from $F$ to $G$ goes first to $J$ then to $K$. In the other direction, an IP packet from $G$ to $F$ goes first to $K$ then to $J$.

Give the structure of an IP packet from $A$ to $B$ (i.e., containing payload of $A$) at the following points:

a1. between $F$ and $J$;
a2. between $J$ and $K$;
a3. between $K$ and $G$.

Solution for part a1

IP header
  sndr addr $F$; rcvr addr $G$; next protocol TCP; hop count; chksum;
TCP header
  sndr port $p$; rcvr port $q$; ...
Data of $A$

Solution for parts a2 and a3

Same as for part a1.
Part b

The configuration in part a is changed as follows.

First, $A$ and $B$ use SSL (over TCP).

Second, $F$ and $G$ operate IPsec-ESP in transport mode providing both encryption and authentication. SPI of 11 is used for both directions.

Third, $J$ and $K$ operate IPsec-ESP in tunnel mode providing both encryption and authentication. SPI of 22 is used for both directions.

Give the structure of an IP packet from $A$ to $B$ (i.e., containing payload of $A$) at the following points:

b1. between $F$ and $J$;

b2. between $J$ and $K$;

b3. between $K$ and $G$.

Solution for part b1

IP header
- sndr addr: $F$; rcvr addr: $G$; next protocol id: ESP; hop count; chksum;
- ESP header
  - SPI 11; sequence number; IV;
- ESP data
  - TCP header
    - sndr port: $p$; rcvr port: $q$; ...
  - TCP data
    - SSL-structured data of $A$
- ESP trailer
  - next header: TCP; MAC

Solution for part b2

IP header
- sndr addr: $J$; rcvr addr: $K$; next protocol id: ESP; hop count; chksum;
- ESP header
  - SPI 22; sequence number; IV;
- ESP data
  - IP packet of part b1 ESP trailer
  - next header: TCP; MAC

Solution for part b2

Same as in part b1.
5. **[10 points]** User $A$ logs in for a session in a Kerberos realm. The user shares a password-derived key, $K$, with the KDC. After login, the user has a session key $S$ and a ticket-granting ticket $TGT$.

Now an application $B$ (at say another node) wants to talk (as a client) to the user shell (as a server).

Give the messages exchanged between $A$, $B$ and the KDC in order for $B$ to talk to $A$.

**Solution**

This is the double TGT authentication (section 14.12).

- $B$ sends ["Want your TGT"] to $A$
- $A$ sends [$A$’s TGT] to $B$.
- $B$ sends ["Want tkt for $A$", $A$’s TGT, authenticator] to KDC.
- KDC sends [enc($K_B$, [enc($K_{AB}$, tkt)])] to $B$, where
  - $K_B$ is $B$’s master key shared with KDC,
  - $K_{AB}$ is session key generated by KDC,
  - and tkt is enc($S$, [enc($K_{AB}$, $B$)])
- $B$ sends [tkt, enc($K_{AB}$, nonce)] to $A$. 