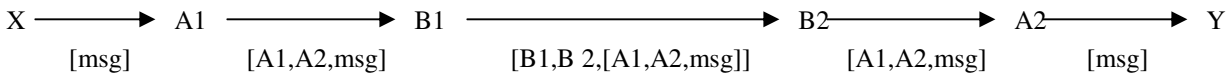
Total points: 55. Total time: 75 minutes. 6 problems over 6 pages. No book, notes, or calculator

1. [10 points]
Are n=221 and d=35 valid numbers for RSA. Explain. If you answer yes, obtain the corresponding e.

2. [6 points]



Every day X talks to Y via nodes A1, A2, B2, B1, as shown above: X sends a msg of 56 octets; A1 attaches a header of "A1,A2"; B1 puts the entire packet in another packet with header "B1,B2"; B2 undoes B1's wrapping; A2 undoes A1's wrapping. Addresses A1, A2, B1, B2 are each 32 bits.

One day, X and Y decide to *encrypt* their communication with a secret key J (i.e., X and Y share J), and B1 and B2 decide to *integrity-protect* their communication with a secret key K (i.e., B1 and B2 share K). Both pairs use DES in CBC mode. Give the size of A1-B1 packet and the size of the B1-B2 packet. Explain your answers briefly.

3. [14 points]
An organization has a PKI (public-key infrastructure) for its employees consisting of a single CA (certification authority) and a single directory server (DS). Answer the following questions. Be brief and precise.

a. Describe the steps taken by a new employee A upon joining the organization.
b. Describe the steps employee A takes to email a message confidentially to an employee B (who may not be online).
c. Describe the steps employee A takes to send a message confidentially to an employee B (who may not be online) such that B can be assured from the contents of the message that it was sent by A (without doing any further interactions).

4. [10 points]

client A (has J)	**server B** (has J)
generate random C_A $N_A \leftarrow$ encrypt C_A with key J send [A, B, conn, N_A] // msg 1	
	receive [A, B, conn, N_A] $R_A \leftarrow$ decrypt N_A with key J $S_A \leftarrow$ encrypt (R_A+1) with key J generate random C_B $N_B \leftarrow$ encrypt C_B with key J send [B, A, S_A, N_B] // msg 2
receive [B, A, S_A, N_B] $T_A \leftarrow$ decrypt S_A with key J if $T_A = C_A$+1 then B is authenticated else abort $R_B \leftarrow$ decrypt N_B with key J $S_B \leftarrow$ encrypt (R_B+1) with key J send [A, B, S_B] // msg 3	
	receive [A, B, S_B] $T_B \leftarrow$ decrypt S_B with key J if $T_B = C_B$+1 then A is authenticated else abort

Client A and server B use the above authentication protocol. J is a key obtained from a password. B handles at most one client at a time. Answer the following; each part below is independent.

a. Consider an attacker that can **only eavesdrop** (i.e., hear messages in transit but cannot intercept messages or send messages with somebody else's sender id). Can this attacker obtain J by off-line password guessing. If you answer no, explain briefly. If you answer yes, describe the attack.

b. Consider an attacker that can **only spoof A** (i.e., send messages with sender id A and receive messages with destination id A, but not eavesdrop or intercept messages). Can this attacker obtain J by off-line password guessing. If you answer no, explain briefly. If you answer yes, describe the attack.

5. [5 points]

The same protocol as in problem 4 except that J is now a high-quality key, B can handle muliple clients at a time, and the different instances of B do not communicate with each other.

client A (has J)	**server B** (has J)
generate random C_A $N_A \leftarrow$ encrypt C_A with key J send [A, B, conn, N_A] // msg 1	
	receive [A, B, conn, N_A] $R_A \leftarrow$ decrypt N_A with key J $S_A \leftarrow$ encrypt (R_A+1) with key J generate random C_B $N_B \leftarrow$ encrypt C_B with key J send [B, A, S_A, N_B] // msg 2
receive [B, A, S_A, N_B] $T_A \leftarrow$ decrypt S_A with key J if $T_A = C_A$+1 then B is authenticated else abort $R_B \leftarrow$ decrypt N_B with key J $S_B \leftarrow$ encrypt (R_B+1) with key J send [A, B, S_B] // msg 3	
	receive [A, B, S_B] $T_B \leftarrow$ decrypt S_B with key J if $T_B = C_B$+1 then A is authenticated else abort

Consider an attacker who can only **spoof A**. Can this attacker impersonate A to B. If you answer no, explain briefly. If you answer yes, describe the attack.

6. [10 points]
Server B, which supports many clients, is attached to the Internet at a well-known (not secret) <TCP port, IP addr> y.
Each client shares a password-dervied key with B. So B has for, each client, an entry consisting of the client id and key.
The clients and server also share Diffie-Hellman parameters g and p (not secret).
B has so many clients that it can decrypt ciphertext encrypted with a client key only if it already knows the client id;
i.e., it is not feasible for B to try all the client keys until it finds one that results in sensible plaintext.

Write down an authentication protocol so that a client A attached at an Internet <TCP port, IP addr> x can connect to B without disclosing its id (i.e., "A") to an attacker that can **only eavesdrop** (i.e., hear messages in transit but cannot intercept messages or send messages with somebody else's sender id). Cliearly identify the operations done at each side and the messages exchanged.

A at x (has g, p and secret key K)	**B at y** (has g, p and a [client id, key] entry for each client)