CS742 MidSem, Sept 14, 2017 (3.00pm to 5.00pm) **[30 Marks]**

Your Name: _____ Your Roll Number: _____

Code of Honour: I hereby declare that I will not cheat (or have not cheated) in this examination. _____ (signature)

Q1	Q2	Q3	Q4	Q5	Q6	Total

General Instructions:

- Handwritten notes (only) are allowed.
- State any assumptions clearly (they should make sense). Draw neatly. Please be crisp and to the point in your answers.
- **No marks for answers without explanation**

Overview: Question 1 [7 Marks]

These are short-answer questions. Please answer in 2-4 lines and not in paragraphs. ***Explain your reasoning behind the answer.*** *Yes/no answers will not fetch you marks.*

1. Pappu copies Lallu's homework. It is a violation of which of confidentiality, integrity, availability, authenticity, or some combination of these?

   Confidentiality: Pappu could copy because Lallu's homework is open (similar to eavesdropping)
   Some mentioned authenticity but to the teacher, Pappu is not pretending to be Lallu.

2. Is it possible for totient function Φ (n) to be bigger than n (in RSA)? Justify.

   No. The largest value it takes is when n is prime. In this case the totient function is n-1.

3. What is the rationale behind that fact that most banks and credit card companies allow their customers to access their accounts from ATM machines using only 4 digits as personal identification?

   The ATM machines locks the account after 3 tries. So, it is pretty secure even though it is only 4 digits of identification.

4. In RSA, what restriction must be placed on the length of the message so that it is unambiguously decrypted?

   That m < n such that m = m mod n

5. In PKCS standard which employs below encoding format, what is the role of the random non-zero bytes?

00	02	Random non-zero bytes (>=8)	00	Plaintext

The above random non-zero bytes are padding. They help with attacks such as small message set; small and constant e.

6. Suppose K is a long-term shared key and R is a nonce sent by a party in one-way authentication. Which among the following is a secure short-term session key? [2 Marks]
    a. K xor R

    Not a secure mechanism. R is sent in the open.
    If the short term session key is compromised, long-term key is also compromised.

    b. $\{K\}_K$ i.e key K encrypted with K

    Not a secure mechanism since the short term session key will not change.

Question 2: Hash Functions [4 Marks]

Message digests are pretty fast, but Mr. Hashwini wants to make them even faster and came up with the following scheme for long messages. Take the message, divide it into 128-bit chunks, and XOR all the chunks together to get a 128-bit result. Do the standard message digest (hash, H) on this result. What do you think of this message digest? Answer with respect to the following properties. Justify your answer.

1. Can be applied to any sized message M.

Yes. XOR can be done repeatedly for any length (with padding when required).

2. Produces fixed-length output h.

    Yes. Since hash done on the result (xors of chunks) is of fixed length.

3. It is easy to compute h=H(M)

    Yes. XoRs is much simpler than taking hash over a long message.

4. Given h it is infeasible to find x such that H(x) = h

    No given that the underlying standard message digest is not invertible.

5. Given x is infeasible to find y such that H(y) = H(x)

    Reorder 128-bit chunks in x to get y. Xors stays the same. X is assumed to be a multiple of 128 bits.

6. Is infeasible to find any x, y where H(y) = H(x)

    Choose some x. Reorder 128-bit chunks in x to get y. Xors stays the same. X is assumed to be a multiple of 128 bits.

Question 3 ***Modern Cryptography*** **[7 Marks]**
Alice wants to communicate with Bob securely achieving both confidentiality and integrity/authentication. They have a long term key K and access to a symmetric key encryption/decryption function (E/D). They also have access to a cryptographic hash function H. However, their computers are not powerful and they can only compute H once and E or D once per message (whether sent or received). Alice has the option of using the following five schemes to send a message M to Bob. Ignore replay attacks. Assume M has high entropy i.e. it is chosen from a very large set of messages.

Part-A: For each scheme, determine whether or not the scheme 1) allows Bob to decrypt messages from Alice. 2) provides confidentiality and 3) provides integrity/authentication? Justify your answers and state any assumptions made clearly. Be sure to specify what Bob has to do in order to decrypt the message and make sure that it came from Alice and has not been tampered with. Remember, Bob can only use E/D once and H once.

1. H(E(M))
   Since M is encrypted it provides confidentiality.
   M however cannot be decrypted since hash is not invertible.
   Integrity/authentication make no sense when the message itself cannot be decrypted.

2. E(M), H(E(M))
   Since M is encrypted it provides confidentiality. M can be decrypted via D(C) (C = E(M)).
   This however does not provide integrity/authentication since if Mallory replaces the message with (x, H(x)) for a random x. Bob wouldn't realize that the message was not from Alice. Bob may decrypt it to some gibberish message D(x), but he will still believe it is what Alice sent.

3. E(H(M))

   Similar to (1), M cannot be decrypted.

4. E(H(M)), H(M)

   Similar to (1), M cannot be decrypted.

5. E(M), H(M)

Since M is encrypted it provides confidentiality. M can be decrypted via D(C) (C = E(M)). It provides integrity also. H(M) is not a MAC but M recovered from E(M) can be checked against it. Mallory does not know the key. if she replaces E(M) with some C, she cannot know D(C), and so she cannot compute H(D(C)).

Part-B: Consider option 5 above. If M has low entropy i.e there are only a few likely possibilities of M, how would your answer to option 5 change?

It does not provide confidentiality. Certainly as before Mallory cannot recover M from E(M). However, Mallory also knows H(M). Since H is a cryptographic hash function, Mallory cannot invert H to get M. However, if M is low-entropy (there are only a few likely possibilities of M) then Eve could just compute H(M') for all of these M' and see which ones match the value of H(M) sent by Alice. Other answers remain the same as above.

Question4: TLS/SSL [5 Marks]

a) In TLS/SSL, what security properties listed below are achieved, and what components of the protocol enables these properties?

Confidentiality: Session keys are derived during the handshake and these are used for later encryption.

Integrity: During handshake, the finish message ensures integrity. During regular communication a MAC is calculated for every record.

One-way Authentication: Server certificate followed by client passing the pre-master key using server's public key followed by hash mac style hash of messages exchanged ensures derived from per-master key and nonces ensure server is authenticated.

b) In practice, TLS as used on the web typically only provides one-way authentication i.e. the client authenticates the server and not vice versa. Why is TLS usually used this way?

One reason is that it is rather inconvenient for the average user to have to set up their own client certificate. Websites would not want to block users who haven't set up a client certificate. Another reason is that in many cases, the server doesn't care who is connecting to it, since it's providing a service over the web that anyone can use. A user certainly cares that they are connecting to the correct server; a server likely expects connections from many clients, so it may not need or want to authenticate them all.

c) How else might a web server authenticate a user/client? (if the user is not authenticated by TLS)

Servers often authenticate the user by requiring them to establish a username and password with the site. That way, once a secure TLS connection has been established, the user passes the password details and server knows which of its users it is communicating with.

Question 5: ***Modes of Encryption*** **[3 Marks]**

Consider the following encryption mode for applying AES-128 with a key K to a message M that consists of l 128-bit blocks, $M_1; \ldots\ldots ; M_l$. The sender first picks a random 128-bit string, C_0, which is the first block of ciphertext. Then for $i > 0$, the i^{th} ciphertext block is given by $C_i = C_{i-1} \oplus \text{AES-128}_K(M_i)$. The ciphertext is the concatenation of these individual blocks: $C = C_0 \mid C_1 \mid C_2 \mid \cdots \mid C_l$.

a) What is the intent behind the random value C_0?

C_0 is an Initialization Vector. The intent behind using it is to ensure that if the same text is encrypted in two distinct messages, the ciphertexts will differ, so an eavesdropper can't infer the relationship between the messages.

b) Is this mode of encryption secure? If so, state what desirable properties it has that make it secure. If not, sketch a weakness.

It is not secure. Since the ciphertext is visible to an eavesdropper, the eavesdropper knows C_i for all values of i. This allows them to directly determine AES-128$_K(M_i)$ for all i due to the inverse nature of exclusive-or, which makes the scheme equivalent to ECB in terms of revealing whenever two message blocks contain the same text.

c) Suppose we replace the computation of C_i with $C_i = \text{AES-128}_K(C_{i-1} \oplus M_i)$. Does this make the mode of encryption more secure, less secure, or unchanged? Briefly explain your answer.

This mode is more secure. This alternate form is exactly the definition of CBC mode which is better than ECB.

Question 6: Mutual Authentication [4 Marks]

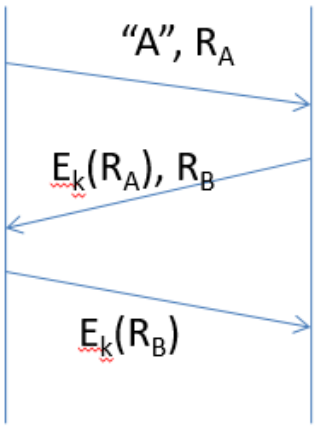
When secret key is derived deterministically from human passwords, one can launch an off-line password guessing attack (based on dictionary of popular passwords) and derive the secret key if the attacker has access to the pair <R, E_k{R}>. This is often easy to obtain during authentication as seen in some of the protocols.

a) Explain how this attack works?

When the attacker has access to the pair <R, E_k{R}>, he can use the dictionary of passwords; for each password in the dictionary, he can derive the secret key (deterministic process) and encrypt the nonce R using the key and see if matches the given Ek{R). If it did, attacker cracked the secret key.

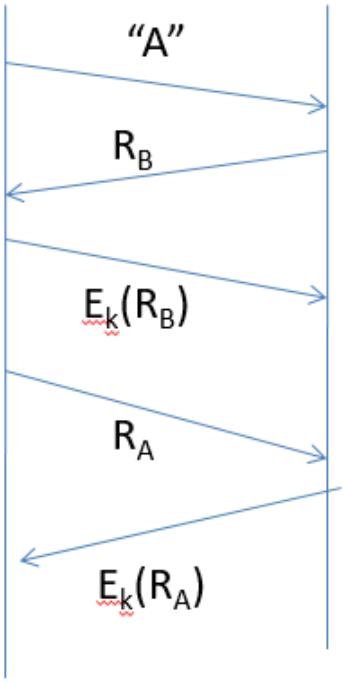
b) Suppose the below authentication protocol is being used (illustrated between two entities A and B) and the attacker **cannot** eavesdrop on the conversations between A

and B, can the offline-password guessing attack still be launched by the attacker? Explain why or why not.



Yes. The attacker M can initiate a conversation with B pretending to be A and pass on A, R_M. B will pass on $E_k\{R_M\}$. From this pair, like explained above, M can use the dictionary of passwords and crack the secret key.

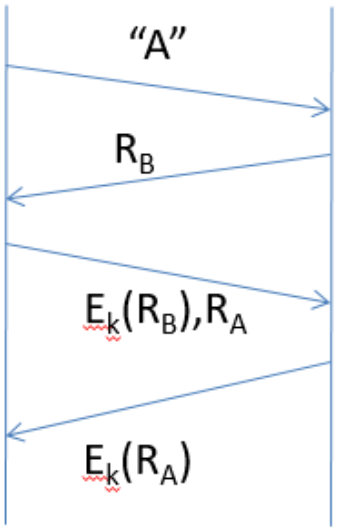
c) Suppose the below authentication protocol is being used (illustrated between two entities A and B) and the attacker **cannot** eavesdrop on the conversations between A and B, can the offline-password guessing attack still be launched by the attacker? Explain why or why not.



No. The attack is not possible anymore. Unless M proves it knows the key (message 2 and 3), the other side will not provide an encrypted version of the nonce M provides (message 4 and 5).

d) Suppose the below authentication protocol is being used (illustrated between two entities A and B) and the attacker **cannot** eavesdrop on the conversations between A

and B, can the offline-password guessing attack still be launched by the attacker? Explain why or why not.



This is identical to case (c) Except that one message has been eliminated by combining 3 and 4 in (c)

Time Pass: What is your passion, something for which you are happy to devote lot of time and energy?

Technology that touches people's lives and caring for the two little brats at home ☺

Rough Work: