CS336 Homework #3

Due: Fri Apr 7, 2023, 5:00pm
Points: 100 pts

1. (20 pts) AES algorithm. Use 4*4 tables to represent a state block. **Explain in detail**.

(a) (AES S-box) Assume you have a State block as defined below, indicate the size of the block and find the new state after applying the AES S-box (can be found in the slides Lec8) transformation. Consider each 4-tuple below a column in the input block.

State: D1 26 B9 3C; 59 C2 AC 42; 15 BC 42 A9; 39 DA D3 26.

(b) (AES Add round key) Assume the round key is as indicated below. Please show the result of the last state of the round assuming that the state before adding the key is the state resulting from 4(a) above. Also, assume that the block key is written in a row format (each 4-tuple is one row of the block)

Round Key: 36 24 A3 82; 00 00 00 00; AA B2 30 57; 11 43 1D C1.

2. (15 pts) Consider a Diffie-Hellman scheme with a common prime q = 11 and a primitive root a = 2. **(Do not use calculator!)**

- If user A has public key Ya = 9, what is A's private key Xa?
- If user B has public key Yb = 3, what is the shared secret key K?

3. (15 pts) In a public-key system using RSA, you intercept the ciphertext C = 10 sent to a user whose public key is e = 5, n = 35. What is the plaintext P? **(Do not use calculator! I WANT step-by-step calculations!)**

4. (10 pts) Suppose that someone suggests the following way to confirm that the two of you are both in possession of the same secret key. You create a random bit string the length of the key, XOR it with the key, and send the result over the channel. Your partner XORs the incoming block with the key (which should be the same as your key) and sends it back. You check, and if what you receive is your original random string, you have verified that your partner has the same secret key, yet neither of you has ever transmitted the key. Is there a flaw in this scheme?

5. (10 pts) Explain why hash collisions occur. That is, why must there always be two different plaintexts that have the same hash value? What property of a hash function means that collisions are not a security problem? That is, why can an attacker not capitalize on collisions and change the underlying plaintext to another from whose value collides with the hash value of the original plaintext?

6. (20 pts) In this problem we will compare the security services that are provided by digital signatures (DS) and message authentication codes (MAC). We assume that Oscar is able to observe all messages sent from Alice to Bob and vice versa. Oscar has no knowledge of any keys but the public one in case of DS. State whether and how (i) DS and (ii) MAC protect against each attack. The value auth(x) is computed with a DS or a MAC algorithm, respectively.

a. (Message integrity) Alice sends a message x = "Transfer $1000 to Mark" in the clear and also sends auth(x) to Bob. Oscar intercepts the message and replaces "Mark" with "Oscar". Will Bob detect this in either case (i) DS and (ii) MAC?

b. (Replay) Alice sends a message x = “Transfer $1000 to Oscar” in the clear and also sends auth(x) to Bob. Oscar observes the message and signature and sends them 100 times to Bob. Will Bob detect this in either case (i) DS and (ii) MAC?

c. (Sender authentication with cheating third party) Oscar claims that he sent some message x with a valid auth(x) to Bob but Alice claims the same. Can Bob clear the question in either case (i) DS and (ii) MAC?

d. (Authentication with Bob cheating) Bob claims that he received a message x with a valid signature auth(x) from Alice (e.g., “Transfer $1000 from Alice to Bob”) but Alice claims she has never sent it. Can Alice clear this question in either case (i) DS and (ii) MAC?

7. (10 pts) Explain what PKI (Public Key Infrastructure) is in your words. Does a PKI perform encryption? Explain your answer. Does a PKI use symmetric or asymmetric encryption? Explain your answer. Why does a PKI need a means to cancel or invalidate certificates? Why is it not sufficient for the PKI to stop distributing a certificate after it becomes invalid?