ITIS 6200/8200 Principles of Information Security and Privacy

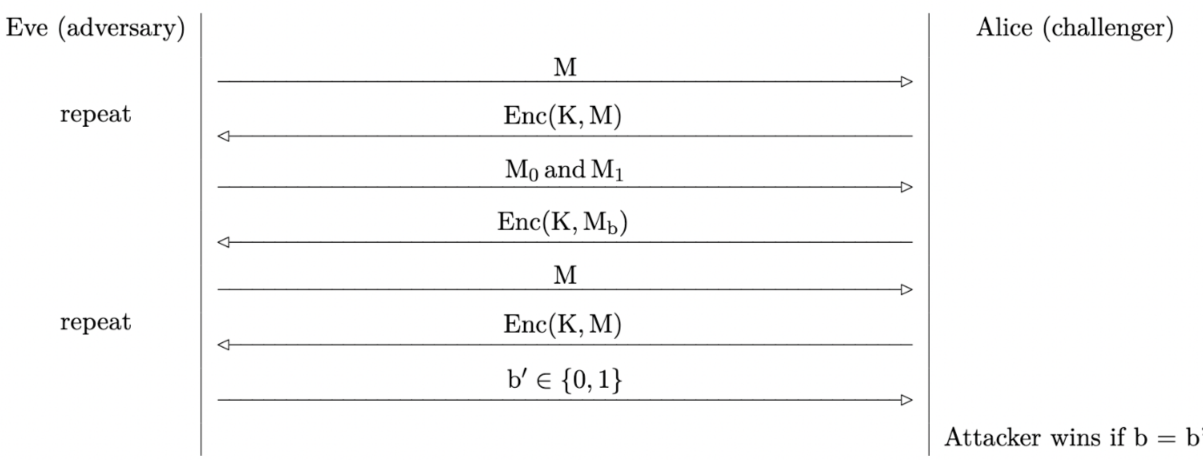
Homework 2

Question 1. Break block cipher DES (10 points)
The DES (Data Encryption Standard) was a symmetric encryption algorithm designed in 1976. It was the government standard until 2001. It has a block size of 64 bits, and key size of 56 bits. If Eve wants to brute-force attack DES, i.e., try all possible keys, how much time does Eve need? Assume that she can try 10^10 keys per second with her personal computer.

Ans: Eve needs to try $2^{56} = 2^{(10\times5.6)}=10^{(3\times5.6)}=10^{(16.8)} = 6.3\times10^{16}$ keys. The needed time is $6.3\times10^{16} / (10^{10})= 6.3\times10^{6}$ seconds, roughly 73 days.

Grading notes: students are allowed to make approximations. Answers that are in a reasonable range of 70 days are acceptable.

Question 2. IND-CPA (20 points)
When formalizing the notion of confidentiality, as provided by a proposed encryption scheme, we introduce the concept of indistinguishability under a chosen plaintext attack, or IND-CPA security. A scheme is considered IND-CPA secure if an attacker cannot gain additional information about a message given its ciphertext. This definition can be defined as an experiment between a challenger and adversary, detailed in the diagram below. Note that the same key K is used for encrypting different messages here.



Q 2.1: Eve sends two messages M_0 and M_1 to Alice. Alice will flip a random bit $b \in \{0,1\}$, encrypt M_b, and send back $C = Enc(k, M_b) = M_b \oplus k$ to Eve. How does Eve determine b with probability > 1/2? (7 points)

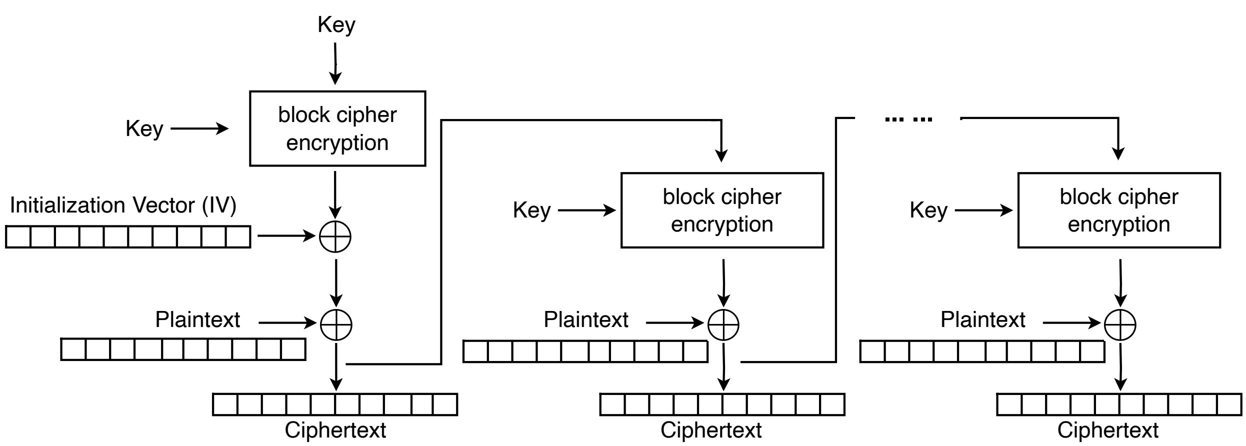
Ans: Eve can trick Alice to encrypt M_0, if the return ciphertext C_0 is the same as C, then the b = 0, otherwise, b = 1.

Q 2.2: Explain how an adversary can always win the IND-CPA game with probability 1 against a deterministic encryption algorithm. Note: Given an identical plaintext, a deterministic encryption algorithm will produce identical ciphertext. (7 points)
Ans: Eve can trick Alice to encrypt M_0, if the return ciphertext C_0 is the same as C, then the b = 0, otherwise, b = 1.

Q 2.3: Explain why reusing keys in one-time pads is dangerous. (6 points)
Ans: Eve can trick Alice to encrypt a message that is all 0, the return ciphertext C_0 is the key used by Alice. Then Eve can decrypt all other ciphertext with the key.

Grading notes: It's OK if the students give similar answers for the three questions, i.e., it is deterministic.

Question 3. Block Cipher Design (25 points)
Alice has developed a new block cipher as below:



The message M is split into j plaintext blocks $M_1 \ldots M_j$ each of size *n*. The encryption mode outputs $(IV, C_1, \ldots, C_j)$ as the overall ciphertext. Assume that IV is randomly generated per encryption.

Q 3.1: Write down the encryption formula. That is, what is the formula for C_1 and C_i ($0 < i <= j$) given (1) plaintext $M_1 \ldots M_j$ (2) encryption algorithm Enc(K, M) which takes a key K and message M as inputs, and (3) a randomly generated IV. (8 points)
Ans: $C_1 = IV \oplus M_1 \oplus Enc(k, k)$

$Ci = Enc(k, C_{i-1}) \oplus M_i$

$C = (IV, C_1, \ldots, C_j)$

Q 3.2: Write the decryption formula for M_i ($0 < i <= j$) using this mode. That is, how to get M_1 and M_i ($0 < i <= j$) given (1) ciphertext ($IV, C_1, \ldots, C_j$) and (2) encryption algorithm Enc(K, M). (8 points)

Ans: $M_1 = IV \oplus C_1 \oplus Enc(k, k)$

$Mi = Enc(k, C_{i-1}) \oplus C_i$

Q 3.3: Is this mode IND-CPA secure? If yes, explain why; if not, describe how an attacker can break IND-CPA. (9 points)

Ans: Not IND-CPA secure. For example, for two messages with the same first block, we can tell if they are the same by XOR out the IV and reveal the value of $Enc(k, k) \oplus M_1$, which is deterministic. The following scheme gives Eve probability of 1 of knowing which message was encrypted by Alice:

1. Eve can send Ma and Mb to Alice for encryption. The two messages have different first block.
2. Alice randomly chooses and encrypts Mx into Cx (x = a or x = b), and sends Cx to Eve.
3. Eve sends Ma to Alice for encryption. Alice sends back Ca, the ciphertext of Ma.
4. Do $Cx \oplus IVx$ and $Ca \oplus IVa$, if the two results have the same value for the first block, then we know x = a, otherwise x = b.

Grading notes: For the first two questions, if the answers have reasonable information, e.g., Enc(k, k), give partial credits.

Question 4. Hash (20 points)

Alice is sending message M to Bob in the following way:

$$\text{Ciphertext } c = c_1 \,||\, c_2 \text{ where } c_1 = Enc(K, m) \text{ and } c_2 = Hash(c_1)$$

Here, Enc(K,m) is the secure encryption scheme AES-CBC, and Hash(m) is the cryptographic hash function SHA-256.

Q 4.1: Does this scheme provide confidentiality? E.g., can an eavesdropper Eve learn about the contents of the message? Why? (5 points)

Ans: It provides confidentiality.

Q 4.2: Does this scheme provide integrity? E.g., can Mallory tamper with message without being detected? Why? (5 points)

Ans: No integrity, since SHA-256 may suffer from length extension attack.

Q 4.3: Can you design an approach for sending the message so it provides both integrity and confidentiality? (10 points)

Ans: Ciphertext $c = c_1 \,||\, c_2$ where $c_1 = Enc(K, m)$ and $c_2 = MAC(K, c_1)$

Or Ciphertext $c = c_1 \| c_2$ where $c_1 = \mathrm{Enc}(K_1, m)$ and $c_2 = \mathrm{MAC}(K_2, c_1)$

Grading notes: Give only partial points if the answer for Q 4.2 doesn't explain why. And there may be other schemes for Q 4.3.

Question 5. PRNGs and Diffie-Hellman Key Exchange (25 points)

Eve is an eavesdropper between Alice and Bob.

1. Alice and Bob each seed a PRNG with different random inputs.
2. Alice uses her PRNG from the previous step to generate a, and Bob uses his PRNG from the previous step to generate b.
3. Alice and Bob perform a Diffie-Hellman key exchange using their generated secrets (a and b). Recall that, in Diffie-Hellman, neither a nor b are directly sent over the channel.
4. Alice and Bob, without reseeding, each use their PRNG to generate some pseudorandom output.
5. Eve learns both Alice's and Bob's pseudorandom outputs.

Assume that Eve can learn the internal state of a PRNG at step 5. And Eve wants to learn the Diffie-Hellman shared secret $g^{ab} \bmod p$.

Q 5.1: If Alice and Bob both use a PRNG that are not rollback-resistant. Can Eve learn about the shared secret $g^{ab} \bmod p$? If yes, how? If no, why? (5 points)
Ans: Yes. Eve may learn about a and b, thus the shared secret $g^{ab} \bmod p$.

Q 5.2: If Alice uses a PRNG that is not rollback-resistant. Bob uses a PRNG that is rollback-resistant. Can Eve learn about the shared secret $g^{ab} \bmod p$? If yes, how? If no, why? (5 points)
Ans: Yes. Eve may learn about b, thus the shared secret $g^{ab} \bmod p$ by $(g^a \bmod p)^b \bmod p$.

Q 5.3: Assume that at step 2, Alice generates a secret value a = 3, and Bob generates a secret value b = 2. For step 3, the values of g and p are 5 and 7 respectively. Then, the shared secret should be $g^{ab} \bmod p = 5^6 \bmod 7 = 1$. However, Diffie-Hellman Key Exchange is vulnerable to Man-in-the Middle attack. Assume that Mallory is successfully launching Man-in-the Middle attack against the key exchange between Alice and Bob. Can you find a positive value m from Mallory, such that the shared secret Alice computes is the same as the shared secret Bob computes? (Hint: consider writing a short loop program in whatever programming languages you prefer to try different m) (10 points)
Ans: Yes. There are many such m values, for example, m = 6*x for x = 1 to 10.

Grading notes: