Elements of Cryptography and Computer and Networking Security
COMPSCI 134
Fall 2016
Instructor: Karim ElDefrawy

Solution of Homework 1
Released: Friday, 09/30/2016
Due: Friday, 10/14/2016 at 11:55pm PT
Solution Posted: on Wednesday, 10/19/2016

Full Name:
UCI ID Number:
Sources:

Guidelines:

- Use any word processor (or handwrite and scan your answers). Upload your solutions as a **PDF** to the associated EEE dropbox (labeled "CS134: Homework 1"). No late submission will be accepted into the EEE dropbox. The solution to the homework will be posted on Wednesday 10/19/2016, no late submission (even via email) will be accepted after posting of the solution.
- No collaboration is allowed. The only people you may ask for help are the TA and professor of this course.
- Copying or rephrasing answers from the Internet or other sources is not allowed, and to do so would be a violation of academic honesty. You must list any sources you used to arrive at your answers (e.g., reference books, Wikipedia etc).

Warning: any submission not following the above guidelines may receive a score of zero.

1 [5 pts total] Multiple Choices

There is only one correct answer for each of the following question. Answer justification is not required.

1. Which of the following is not considered an active attack?
   A. Denial of Service
   B. Man in the middle attack
   C. Wiretapping
   D. IP address spoofing
   E. All of the above are active attacks

2. Release of messages content is a threat against which security service?
   A. Confidentiality
   B. Integrity
   C. Authentication
   D. Access control
   E. Availability

3. Which of the following ciphertexts is not possible from encrypting a plaintext "CRYPTO" using Affine cipher?
   A. MRCZJQ
   B. XCNKUB
   C. TYJGQX
   D. QVQDNU
   E. All of the above are possible

4. Rotokas language alphabet consists of 12 letters. Suppose you receive a cipher in Rotokas consisting of 20 letters and know that this cipher is produced by One-Time Pad Cipher. How many possible key combinations are there that can produce this cipher?
   A. 12
   B. 12!
   C. 12^{20}
   D. 20^{12}
   E. None of the above

5. What type of attainable security does One-Time Pad provide when used correctly?
   A. Insecure
   B. Information theoretically secure
   C. Ad hoc secure
   D. Provably secure
   E. None of the above

6. Which of the following is known to be true about P, NP, NP-Hard and NP-Complete
   A. P = NP
   B. NP-Hard $\subseteq$ NP
   C. NP-Complete $\subset$ NP

D. NP-Hard $\subset$ NP-Complete

E. None of the above

7. Which of the following component/algorithm is not in Rijndael?

   A. Substitution box

   B. Fiestel function

   C. Key expansion

   D. Column mixing

   E. All of the above

8. What security service cannot be provided by Symmetric Key/Secret Key/Conventional Cryptograph?

   A. Integrity

   B. Confidentiality

   C. Authentication

   D. Non-repudiation

   E. None of the above

9. What method of defense can be used to prevent insider attacks?

   A. Physical controls

   B. Policies

   C. Hardware controls

   D. Software controls

   E. None of the above

10. If Alice wants to increase speed of encrypting her data by utilizing parallel computing in her encryption scheme, which block cipher mode is the most appropriate in her case?

    A. Cipher block chaining

    B. Cipher feedback

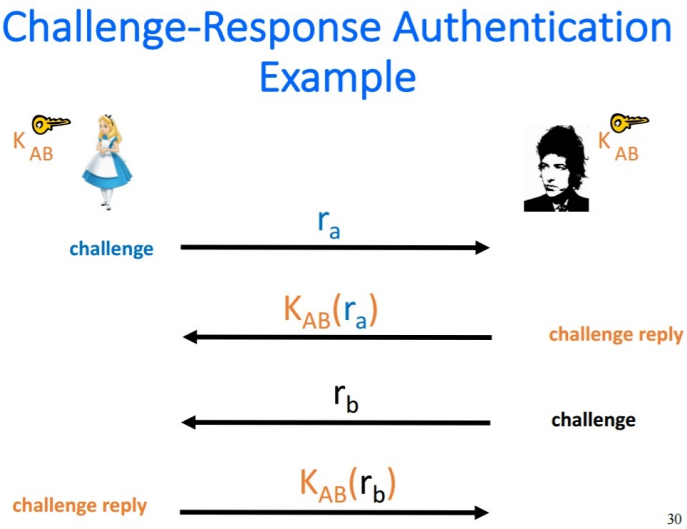
    C. Output feedback

    D. Counter

    E. None of the above

2 [3 pts] Authentication Protocol

Why is the authentication protocol below insecure even when instantiated and used correctly (i.e. r_a and r_b are random numbers or K_{AB} was securely shared in the first place)? Assume that Bob allows, and can accept, more than one communication session at a time and that he does not care about denial-of-service attacks.



Solution:

An adversary can bypass this authentication protocol without knowing K_{AB}. This attack can be carried out as follows:

1. Eve (as an adversary) initiates a connection to Bob by sending r_1 and receiving r_2 and $K_{AB}(r_1)$ back.
2. She opens a second connection to Alice by sending the challenge that she got from Bob in the previous connection, i.e., r_2. As a result, she receives r_3 and $K_{AB}(r_2)$ as a response from Bob.
3. She then can send $K_{AB}(r_2)$ back to Bob on the original connection, leading to the completion of the first authentication protocol with Bob.

3 [5 pts] Security of 3-DES

Consider a variant of 3-DES as follows:

C = $E(K_1, E(K_2, E(K_3, P)))$ where E is the DES encryption function with key size l.

(a) What is the worst-case time complexity (in term of l) of the brute-force attack on this scheme? Briefly justify your answer.

(b) Describe how an adversary can launch the meet-in-the-middle (MITM) attack against this encryption scheme with $O(2^l)$ space complexity. What is the worst-case time complexity (in term of l) of this attack? Briefly justify your answer.

Solution:

[2 pts] (a): To perform brute-force attack on 3-DES, an adversary needs to enumerate all possible keys, which takes $2^l * 2^l * 2^l = 2^{3l}$ iterations. Thus, the worst-case complexity is $O(2^{3l})$.

[3 pts] (b): Given that an adversary knows a [C, P] plain/cipher-text pair, he can pre-compute (by enumerating all values of K_3) and store all possible values of $E(K_3, P)$ into a lookup table. He then can iteratively compute $D(K_2, D(K_1, C))$ and match it with the lookup table; repeat this computation for all possible K_1 and K_2 until a match is found. Note that the matching in the lookup table takes a constant time or $O(1)$. Therefore, the space complexity is just the size of the table or size of all possible values of $E(K_3, P)$, which is $O(2^l)$. The worst-case time complexity in this case is dominated by the time it takes to perform double decryptions: $D(K_2, D(K_1, C))$. Hence, the worst-case time complexity is $O(2^{2l})$.

4 [5 pts] Block Cipher

For each block cipher mode (CBC, OFB, CFB, CTR), explain the precise consequences of a 1-bit error in a single block of ciphertext (the i-the block). Assume that there are n > i plaintext (and ciphertext) blocks.

Solution:

CBC: it will mess up the whole corresponding plaintext block and invert the corresponding bit in the next plaintext block.

OFB: it will invert the corresponding bit in the corresponding plaintext block.

CFB: it will invert the corresponding bit in the same plaintext block and mess up the whole next plaintext block.

CTR: it will invert the corresponding bit in the corresponding plaintext block.

5 [4 pts] Applications of Block Cipher Modes (CBC, OFB, CFB, CTR)

Answer the following question. If more than one answer exists, provide and briefly justify all of them.

(a) A user wants to encrypt an incoming stream of audio data. Which block cipher mode(s) should be used?

(b) A user wants to take advantage of multiprocessing for encrypting her data. Which block cipher mode(s) should be used?

(c) A user wants to construct a message authentication code (MAC) on a message. Which block cipher mode(s) should be used?

(d) A user wants wants to transmit an encrypted message through a noisy communication channel where ciphertext blocks can sometimes be swapped. The user wants to ensure that the decryptor can detect this side-effect when it happens regardless of the value and pattern of plaintext. Which block cipher(s) mode should be used?

Solution:

(a) CFB, OFB and CTR since these modes support encrypting less than a block size.

Alternate correct answers:

OFB and CTR are also accepted if the justification mentions that error will be minimized or a key-stream can be pre-computed .

(b) CTR since its encryption algorithm can be carried out in parallel.

Alternate correct answers:

OFB is also accepted if the justification mentions that the key-stream can be pre-computed in advance, which makes only the final operation ($\oplus$) parallelizable.

(c) CBC and CFB. Recall that a MAC is used for providing integrity of a message and is generally sent along with that message. To construct MAC mode from other modes, we can just encrypt a message normally using that block cipher mode and take only the last ciphertext block and use it as MAC.

CTR and OFB should not be used as MAC because they cannot guarantee the integrity of a message since each ciphertext block is independent of other ciphertext blocks. This will allow an adversary to add, delete or swap any plaintext blocks before the last block, which still results in the same MAC as the original plaintext. This is, however, not true for CBC and CFB since their ciphertext blocks depend on previous blocks.

MAC mode is also accepted as a correct answer.

(d) CBC, CFB, OFB and CTR. All of them support detection of block rearrangement.

6 [5 pts] Reusing Key and IV in CBC

Why is it a bad idea to reuse both the key and IV in CBC mode of operation? (HINT: given two CBC ciphertexts produced by the same key and IV, what can an adversary learn about the corresponding plaintexts relative to each other?)

Solution:

In the CBC mode, if we use the same key and IV to encrypt a plaintext twice, it would obviously result in the same ciphertext for both encryptions. We will use this fact to answer this question.

Suppose we have two plain-/cipher-text pairs, $[P, C]$ and $[P', C']$, where C and C' are produced by the CBC mode of a block cipher with the same key and IV.

By comparing C with C', an adversary can tell whether $P = P'$.

Specifically, if $C = C'$, an adversary learns for sure that $P = P'$. Otherwise, he also learns that $P \neq P'$.

To be more precise, an adversary can tell if the first i^{th} blocks of the corresponding plaintexts are equal to each other or not, using the same argument.

7 [5 pts] More on Block Ciphertext

Suppose you receive a ciphertext (with m blocks) produced by AES-CBC encryption algorithm (AES cipher in CBC mode) and finds out that all ciphertext blocks have the same value, i.e., $C_i = C_j$ where $1 \leq \text{i} < \text{j} \leq \text{m}$.

(a) What can you learn about the plaintext blocks and IV (if any)?

(b) What if the same ciphertext was produced by OFB, can you still learn anything about the plaintext message and IV? Why or why not?

Solution:

(a) Recall an encryption equation in CBC: $C_i = E_K(P_i \oplus C_{i-1})$ and $C_0 = \text{IV}$.

It is given from the question that $C_i = C_j$ for all $1 \leq \text{i} < \text{j} \leq \text{m}$.

We then can substitute C_i and C_j with the expression above. Then, we get: $E_K(P_i \oplus C_{i-1}) = E_K(P_j \oplus C_{j-1})$.

At this point, we can remove $E_K()$ from both sides since the same key is used for both encryptions.

As a result, we get: $P_i \oplus C_{i-1} = P_j \oplus C_{j-1}$.

Now consider two cases:

Case 1 - $\text{i} \geq 2$: Then, we know that $C_{i-1} = C_{j-1}$, which leads to $P_i = P_j$. **This means all plaintext blocks, starting from 2nd block, have the same value.**

Case 2 - $\text{i} = 1$: Then, $C_{i-1} = C_0 = IV$. The equation now becomes $P_1 \oplus IV = P_j \oplus C_{j-1}$. Here, we cannot cancel out IV and C_{j-1} since IV is not known in the question. Hence, **the adversary cannot tell anything about the first block of the plaintext and IV** except their XOR result being equal to the XOR of another plaintext block and ciphertext block. Note that if your justification assumes that IV is known to the public, then it is OK to ignore this case.

(b) We cannot do the same trick in OFB by removing $E_K()$ since the encryption/decryption does not cover the whole expression.

The encryption function in OFB: $C_j = P_j \oplus E_K(IV_j)$, where $IV_j = E_K(IV_{j-1})$ and $IV_0 = IV$.

Thus, the adversary cannot gain any knowledge of plaintext and IV.