CSCI 530: Security Systems
Assignment 1: Due Sept 18th, 2024

Name: Anne Sai Venkata Naga Saketh
USC Email: annes@usc.edu
USC ID: 3725520208

Question 1:

Why do we use encryption modes of operation to convert block ciphers into stream ciphers?

Usually, block ciphers are used to encrypt blocks of data that are of constant length, but in real-time applications, they often get data in different lengths, and data will be streamed, making the stream ciphers a necessity. We can effectively convert the block ciphers into stream ciphers using different Modes of operations, such as Cipher Block Chaining (CBC), Cipher Feedback Mode (CFB), and Output Feedback Mode (OFB), by enabling counters and feedback mechanisms to encrypt data streams for practical use.

What is important about the initialization vector (IV) in a stream cipher, and what happens if the IV is known by the adversary?

The IV is a very important part of the encryption process as it ensures security. Even if the same plaintext is encrypted with the same key, the resulting cipher text will be different for every operation, making it difficult for intruders to get the actual plain text. The IV is a random number that is generated for every encryption operation and will help prevent attackers from finding patterns and deducing the plain text. If the IV is known to an adversary, it can lead to the compromise of the data, and if the same IV is reused, then it is easy for intruders to deduce the actual text from the cipher text using mechanisms like cryptoanalysis.

What happens if we always use an IV of 0000 or 1111?

The IV is a random number that is generated for every encryption session; if we use a constant IV of 0000 or 1111 for multiple encryption sessions, it allows intruders to deduce the plaintext from the encrypted cipher texts by performing cryptanalysis. Using a simple IV and reusing it can also lead to replay attacks, where the attacker can interpret the original message and relay a different message to the end user.

Can a protocol be designed so it is just as safe to use the same IV to encrypt a message stream as it is to use a different IV for the stream each time? Explain your answer.

Yes, theoretically, an encryption protocol using the same IV can be designed, but the whole purpose of having an IV is its randomness, which can't be achieved any further. By using the constant IV, it makes the encryption scheme vulnerable to various attacks and the whole intent of using an IV is lost.

Question 2:

In RSA, an encryption key of e = 3 can be used so long as (p-1)(q-1) is not divisible by 3. For p = 11 and q = 23, let e = 3. Find d. Show how one enciphers the plaintext m = 3 with e = 3 into a value for c. Show the value of c. Then, show that deciphering c with d yields m again. (note that just showing the equation (exponentiation) and the result is not sufficient; you must show the intermediate calculations involved).

Here are the given values to us: p = 11 and q = 23. We can calculate the following:

Step-by-Step Calculation

1. As a first step, we need to calculate the value of n:
   n = p * q
   - ⇨ 11 * 23 = 253

2. Next, we need to calculate the value of (p-1)*(q-1) i.e., 'e':
   (p-1)*(q-1) = (11-1) * (23-1)
   - ⇨ 10*22
   - ⇨ 220

3. Next we need to determine d such that ed = 1 mod ((p-1)*(q-1))
   3 * d = 1 mod 220
   Up on calculating, we can derive the value as,
   d = 147

4. Now, since we have all the required values, let's encrypt the plain text m:
   $c = (m^e) \bmod n$
   - ⇨ $(3^3) \bmod 253$
   - ⇨ 27 mod 253
   - ⇨ Since 27 is less than 253, the answer will be as follows,
   - ⇨ 27

5. Now, since we have the encrypted text, let's decrypt the ciphertext c:
   $m = (c^d) \bmod n$
   - ⇨ $(27^{147}) \bmod 253$
   - ⇨ Based on the binary representation, we can split $27^{147} \bmod 253$ into the following
   - ⇨ $(27^{128} * 27^{16} * 27^2 * 27^1) \bmod 253$
   - ⇨ $((27^{128} \bmod 253) * (27^{16} \bmod 253) * (27^2 \bmod 253) * (27^1 \bmod 253 *)) \bmod 253$
   - ⇨ (169 * 104 * 223 * 27) mod 253
   - ⇨ (169 * 104) mod 253 = 119
   - ⇨ (223 * 27) mod 253 = 202
   - ⇨ Therefore, (119 * 202) mod 253 = 24038 mod 253
   - ⇨ 3

Now, m = 3, and the original text given to us is also 3

Question 3:

When using XOR with a random key as a method of encryption, why is it important that the key be used only once? What is the method of encryption called?

XOR encryption is a simple symmetric encryption technique where an XOR operation is performed with the encryption key and the plaintext. The XOR encryption keys are suggested to be used only once because they provide a higher level of security and will not allow the cryptanalysts to deduce the encryption key by performing an XOR operation on the plaintext and the encrypted cipher text; this is mainly because the XOR operation can be reversed if the plain text and the cipher text are revealed to the intruder. This method of using a randomly generated XOR key only once is called a one-time pad.

This form of encryption is provably secure with respect to specific security goals, but it is vulnerable with respect to other goals. State the purposes for which it is safe and the purposes for which it is weak and explain the reasons for your assessment.

When the one-time pad is implemented successfully, it ensures that confidentiality is wholly maintained at all times because the key is used only once, randomly generated, and it is as long as the plain text that needs to be encrypted. However, the other security aspects, such as Availability and Integrity, will be compromised to some extent.

Coming to availability, it is compromised because, for every plain text message, a new key that is as long as the plaintext should be randomly generated, and the latest key should be securely communicated with the recipient else he would not be able to decrypt the message, thereby reducing the practicality of using the encryption solutions like one-time-pad.

The requirement of many random keys for every new message sent makes this method of encryption not suitable for frequent message exchanges or for long conversations.

Coming to the integrity aspect of secure communication, the one-time pad does not provide the recipient any way of verifying the message source or the integrity of the message; that is, the signature of the message sender cannot be verified. Due to this issue, the attacker could manipulate the message from the sender before it reaches the recipient and alter its content, and the sender's identity cannot be confirmed or verified. However, to overcome the integrity problem, we can implement additional checks such as Message Authentication Code (MAC).