ITIS 6200/8200 Principles of Information Security and Privacy

Midterm

**Note: cite any online resources you have used. Chat GPT is not allowed.**

**Question 1. Security Principle (10 points)**

We have learned 11 secure principles in the class. Name two of them that you like/familiar with, and give real-life examples. Don’t use examples from slides and assignment #2.

**Question 2. Symmetric encryption algorithms (10 points)**

Name at least two advantages and two disadvantages of symmetric encryption algorithms, and provide examples to explain them.

**Question 3. Asymmetric encryption algorithms (10 points)**

Name at least two advantages and two disadvantages of asymmetric encryption algorithms, and provide examples to explain them.

**Question 2. Block Cipher Design (25 points)**

**Question 2**. **Block Cipher Design (25 points)**

One student in our class suggests the following design of block cipher operating mode. The same IV is used for every block. The message M is split into j plaintext blocks M1 … Mj each of size *n*. The encryption mode outputs (IV, C1, …, Cj) 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 C1 and Ci (0 < i <= j) given (1) plaintext M1 … Mj (2) encryption algorithm Enc(K, M) which takes a key K and message M as inputs, and (3) a randomly generated IV.

**Ans**: C1= IV M1 Enc(k, k)

Ci = Enc(k, Ci-1) Mi

C = (IV, C1, …, Cj)

Q 3.2: Write the decryption formula for Mi (0 < i <= j) using this mode. That is, how to get M1 and Mi (0 < i <= j) given (1) ciphertext (IV, C1, …, Cj) and (2) encryption algorithm Enc(K, M).

**Ans**: M1= IV C1 Enc(k, k)

Mi = Enc(k, Ci-1) Ci

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

Ans: Not IND-CPA secure. For example, for two messages with the same contents, we can tell if they are the same by XOR out the IV and reveal the value of Enc(k, k) ⊕ M1, which is deterministic.

* Bob has a public-private key pair (pub\_Bob, pri\_Bob). Alice needs to send some information to Bob. She wants to make sure that when Bob opens the message, he can verify that this is from Alice but not anyone else. So she sends out the message as: [ Alice, Epub\_Bob(message) ] to Bob. Basically, she first sends out her name in clear text, then encrypts the message with Bob’s public key. Please discuss, can an attacker M impersonate Alice and send out a packet in Alice’s name? How can he do it? Here we assume that M also has the public key of Bob. For the same question, if Alice sends out [ Epub\_Bob(Alice, message) ], can M still impersonate Alice? (Here Alice puts her name in the encryption.)
* Alice’s computer stores the files in the following way: for every file F, the computer will calculate the hash value of the file hash(F) and store it after the file. Every time when Alice login, the machine will automatically hash all the files and compare the results to the stored hash values. In this way, if by accident the hard drive is mis-functioning and flips a few bits in a file, Alice can immediately detect it since the hash value will be different. Now an attacker hacks into Alice’s machine and he tries to change several files. The attacker also knows the hash function that the computer uses. Please describe what the attacker needs to do so that the next time Alice login, the machine will not detect the changes. **Also, please discuss how we should improve the mechanism to detect such changes.**

**Question 3**. Length extension attack on some MAC() design