Name: YIN SOKNARA

ID: e20191298

Group: GIC-C

Question and Answer

1. Why do we need cryptography in our system?
   – Because we need to keep some secrets by doing encryption on data so that it is unreadable and can only be decrypted to understand the data by the one we want to share with
   – It provide confidentiality by ensuring the information is always keeping safe and integrity which verifying the data that it cannot be violent or acquire during transmission
   – It is great for security in login authentication to verify the user that try to log into the specific system

2. What is Symmetric Key Encryption, Asymmetric Key Encryption, and hash function, and how is it used in cryptography? What are some examples that are commonly used?
   - Symmetric Key Encryption is an encryption system which sender and receiver use a single key to encrypt the message and decrypt the message. This encryption is efficient and fast. Example: DES, AES, etc…
   - Asymmetric Key Encryption is also an encryption system which sender and receiver use pair of keys (Public key/Private key) to lock and unlock the message. That means public key will be used to do encryption while private key is apply to decryption and these keys are different. This encryption enables secure key exchange. Example: RSA, ECC, etc…
   - Hash function is an algorithm that take input to produce fix-size string of character which cannot be read. Example of hash function are: MD5, SHA-1, SHA-256, etc…
3. Advantages and disadvantages of Symmetric Key Encryption, Asymmetric Key Encryption, Hashing algorithm
   - Symmetric Key Encryption:
     - Advantages:

        - Symmetric key encryption are faster and efficient compared to asymmetric key encryption
        - Implementation is simple
        - Suitable for encrypting large data
        - Only one secret key is needed to be secure in sharing
    - Disadvantages:
        - Sharing secret key can be challenge
        - No mechanism for exchanging secret key
- Asymmetric Key Encryption:
    - Advantages:
        - Security in key exchange
        - Secure communication in a scene of large amount of people
    - Disadvantages:
        - Complexity in implementing
        - Slower than Symmetric Key Encryption
        - Managing and Storing private keys can be challenge with large scale systems
- Hash algorithm:
    - Advantages:
        - Reliable way to verify integrity of data, even with small change of input data can still provide different hash value
        - Fast and efficient in generating hash value for any size of data
        - Commonly used for security in password storage by hashing the password into unreadable string not the original password from input to be store in storage
    - Disadvantages:
        - Collision vulnerability: different input data can provide the same hash value
        - Lacking of key management: hash function only operate on hashing the input data

4. What is the difference between Symmetric, Asymmetric? And what are some situations where you might use one over the other?
    - Symmetric:
        - Use a single key for encryption and decryption
        - Secret key is shared between sender and receiver
        - I might use this encryption for file transferring, database encryption
    - Asymmetric:

    - Use a pair of keys: public key and private key
    - Public key is used for encryption and private is for decryption
    - Each user has their own key pair
    - I might use this encryption when there is a need to perform key exchange and communication among many user.
5. What is the different between encryption and hashing, and what are some situations where you might use one over the other?
   - Encryption:
     - Transform original text into unreadable text using encryption algorithm and key
     - Ensure confidentiality of information, data storage, privacy
     - I might apply this process when I need to protect large data storage
   - Hashing:
     - Converting input data into fixed-size hash value using hash function
     - Generate unique hash value for each unique input
     - Mostly used in password storage
     - I might use this process when I just need to make the input password more securely
6. What are some common attacks that are used to break symmetric key cryptography, and how can they be mitigated?
   → Some common attacks are:
     - Brute Force Attack:
       - The attacker tries all possible combinations of keys until they found the correct one
       - To mitigate the attack, **Use strong and long keys**, since long key can take more computing effort to get the possible combination.
     - Known Plaintext Attack:
       - The attacker has access to both the plaintext and its corresponding cipher text
       - To mitigate the attack, use encryption algorithm which can resist to known plaintext attack, such as AES. Moreover, updating and strengthening the encryption algorithm is helpful
     - Replay Attack:
       - The attacker intercepts and retransmits encrypted messages to acquire unauthorized access or information

        - To mitigate the attack, incorporate measures such as message authentication codes or timestamps to ensure integrity of the message
    - Side-Channel Attack:
        - Leak information during encryption process, such as timing information, power consumption, electromagnetic radiation
        - To mitigate the attack, try to implement countermeasures at hardware and software level.