CompTIA

Security+

SY0 - 601

05- Summarizing Basic Cryptographic Concepts

Ahmed Sultan
Senior Technical Instructor
ahmedsultan.me/about

CRYPTOGRAPHIC CONCEPTS

- **Cryptography**
  - ✓ (literally meaning "secret writing") has been around for thousands of years.
  - ✓ It is the art of making information secure by encoding it.
  - ✓ Security through obsecurity means keeping something a secret by hiding it.
  - ✓ This is generally acknowledged to be impossible (or at least, high risk) on any sort of computer network.
  - ✓ With cryptography, it does not matter if third parties know of the existence of the secret, because they can never know what it is without obtaining an appropriate credential.

CRYPTOGRAPHIC CONCEPTS (cont.)

- The following terminology is used to discuss cryptography:
  - ➢ Plaintext (or cleartext)—an unencrypted message.
  - ➢ Ciphertext—an encrypted message.
  - ➢ Cipher—the process (or algorithm) used to encrypt and decrypt a message.
  - ➢ Cryptanalysis—the art of cracking cryptographic systems.

- In discussing cryptography and attacks against encryption systems, it is customary to use a cast of characters to describe different actors involved in the process of an attack, The main characters are:
  - ➢ Alice—the sender of a genuine message.
  - ➢ Bob—the intended recipient of the message.
  - ➢ Mallory—a malicious attacker attempting to subvert the message in some way.

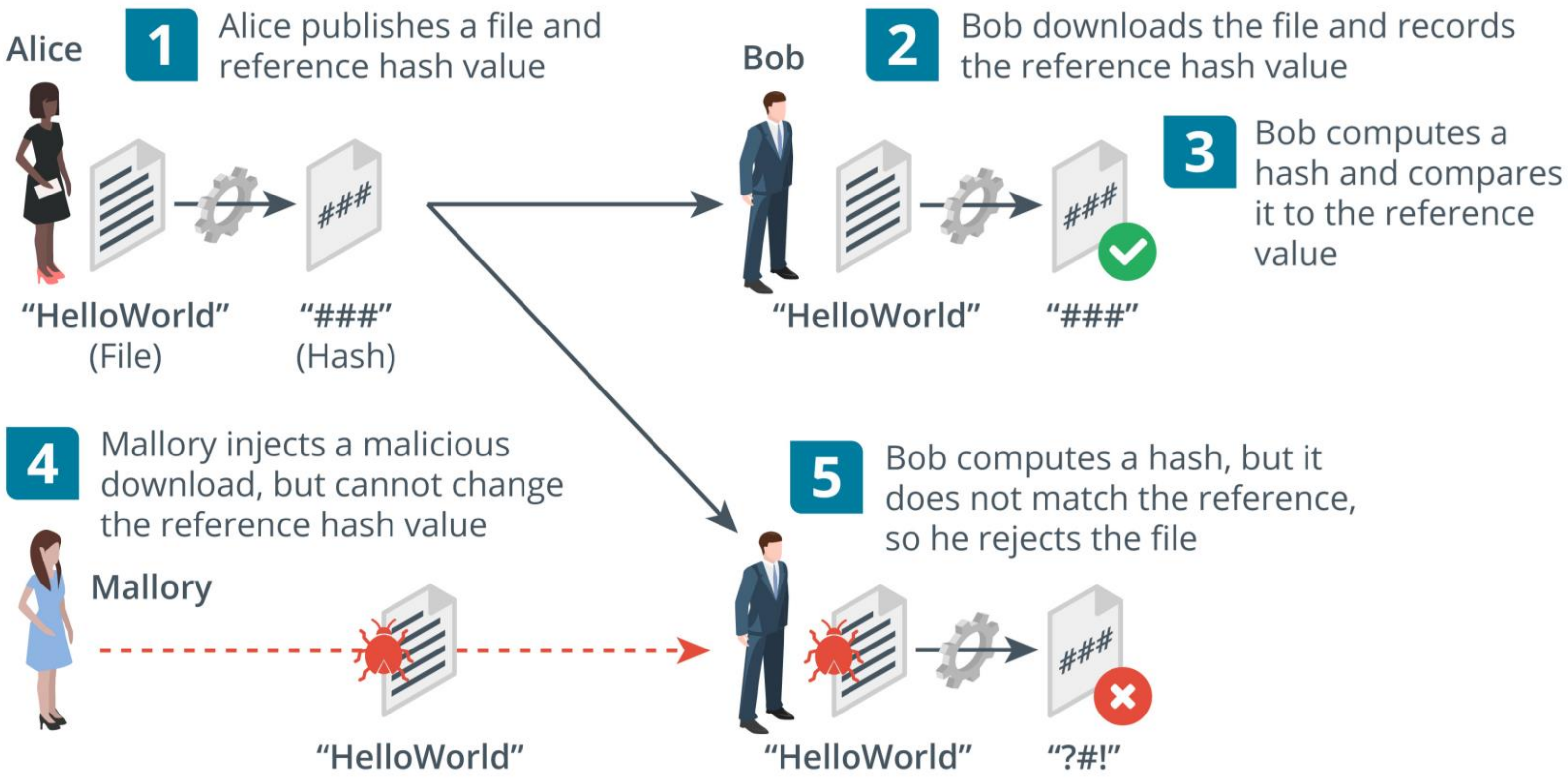
HASHING ALGORITHMS

- Hashing is the simplest type of cryptographic operation.
- A cryptographic hashing algorithm produces a **fixed length** string from an input **plaintext** that can be of any length.
- The output can be referred to as a **checksum**, **message digest**, or **hash**.
- The function is designed so that it is impossible to recover the plaintext data from the digest (one-way) and so that different inputs are unlikely to produce the same output (a collision).

HASHING ALGORITHMS (cont.)

- A hashing algorithm is used to prove integrity.
- **For Example:** Bob and Alice can compare the values used for a password in the following way:
  - ✓ Bob already has a digest calculated from Alice's plaintext password.
  - ✓ Bob cannot recover the plaintext password value from the **hash**.
  - ✓ When Alice needs to authenticate to Bob, she types her password, converts it to a **hash**, and sends the digest to Bob.
  - ✓ Bob compares Alice's digest to the **hash** value he has on file.
  - ✓ If they match, he can be sure that Alice typed the same password.

HASHING ALGORITHMS (cont.)



1 Alice publishes a file and reference hash value

2 Bob downloads the file and records the reference hash value

3 Bob computes a hash and compares it to the reference value

4 Mallory injects a malicious download, but cannot change the reference hash value

5 Bob computes a hash, but it does not match the reference, so he rejects the file

HASHING ALGORITHMS (cont.)

- As well as comparing password values, a hash of a file can be used to verify the integrity of that file after transfer.
  - ✓ Alice runs a hash function on the **setup.exe** file for her product.
  - ✓ She publishes the digest to her website with a download link for the file.
  - ✓ Bob downloads the **setup.exe** file and makes a copy of the digest.
  - ✓ Bob runs the same hash function on the downloaded **setup.exe** file and compares it to the reference value published by Alice.
  - ✓ If it matches the value published on the website, the integrity of the file can be assumed.
  - ✓ Consider that Mallory might be able to substitute the download file for a malicious file.
  - ✓ Mallory cannot change the reference hash, however.
  - ✓ This time, Bob computes a hash but it does not match, leading him to suspect that the file has been tampered with.

HASHING ALGORITHMS (cont.)

- There are two popular implementations hash algorithms:
  - ✓ Secure Hash Algorithm (SHA)
    - ▪ Considered the strongest algorithm.
    - ▪ There are variants that produce different-sized outputs, with longer digests considered more secure.
    - ▪ The most popular variant is SHA-256, which produces a **256-bit** digest.
  - ✓ Message Digest Algorithm #5 (MD5)
    - ▪ Produces a **128-bit** digest.
    - ▪ MD5 is not considered to be quite as safe for use as SHA-256, but it might be required for compatibility between security products.

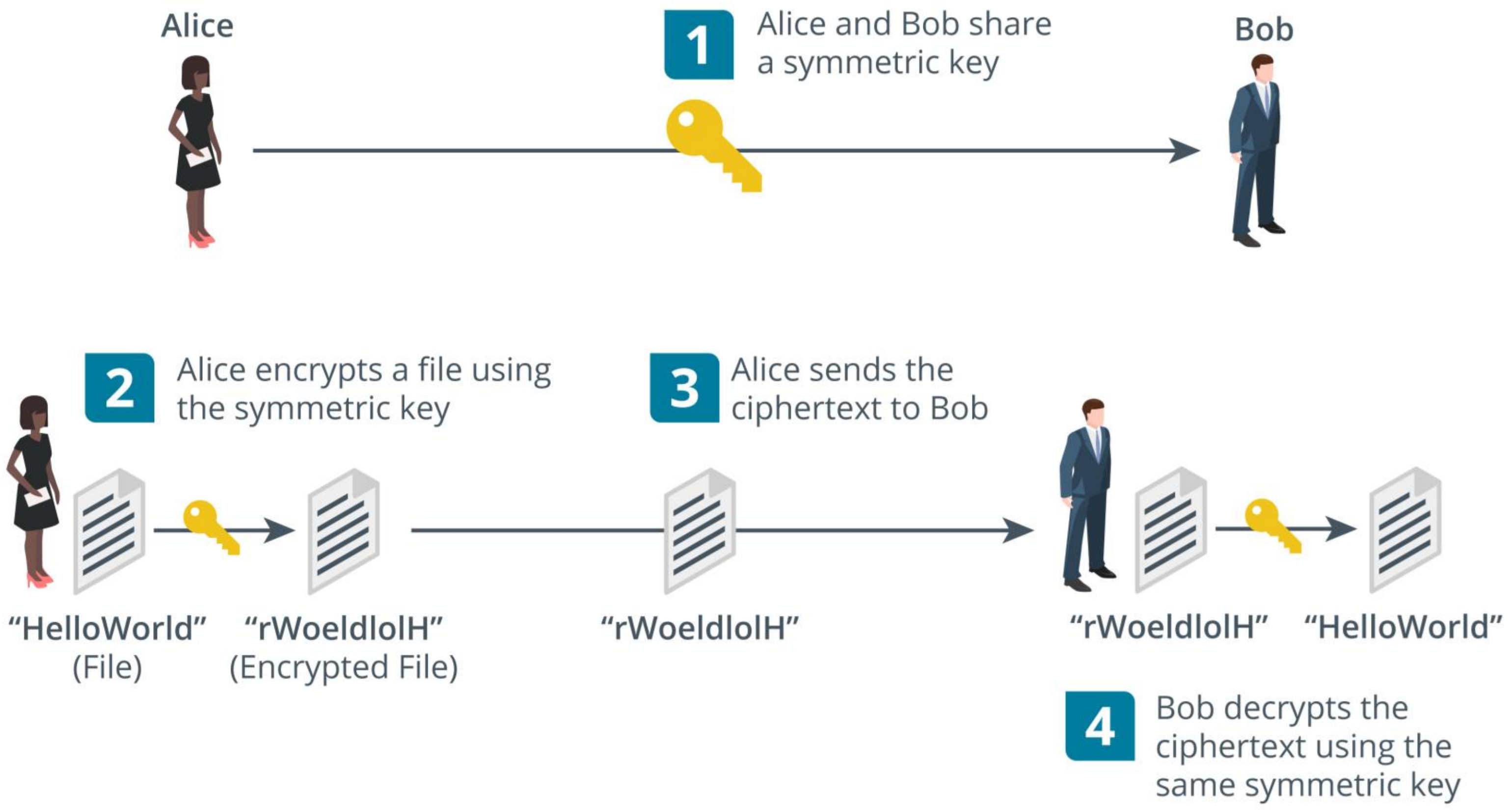
ENCRYPTION CIPHERS AND KEYS

- While a hash function can be used to prove the integrity of data, it cannot be used to store or transmit data.
- The plaintext cannot be recovered from the digest.
- An encryption algorithm is a type of cryptographic process that encodes data so that it can be recovered, or decrypted.
- The use of a key with the encryption cipher ensures that decryption can only be performed by authorized persons.

SYMMETRIC ENCRYPTION

- A symmetric cipher is one in which encryption and decryption are both performed by the **same secret key**.
- The secret key is so-called because it must be kept secret.
- If the key is lost or stolen, the security is breached.
- Symmetric encryption is used for confidentiality.
- For example, Alice and Bob can share a confidential file in the following way:
  - ✓ Alice and Bob meet to agree which cipher to use and a **secret key value**.
  - ✓ They both record the value of the secret key, making sure that no one else can discover it.
  - ✓ Alice encrypts a file using the cipher and key.
  - ✓ She sends only the ciphertext to Bob over the network.
  - ✓ Bob receives the ciphertext and is able to decrypt it by applying the same cipher with his copy of the secret key.

SYMMETRIC ENCRYPTION (cont.)



Alice

1 Alice and Bob share a symmetric key

Bob

2 Alice encrypts a file using the symmetric key

3 Alice sends the ciphertext to Bob

"HelloWorld" (File)

"rWoeldlolH" (Encrypted File)

"rWoeldlolH"

"rWoeldlolH"

"HelloWorld"

4 Bob decrypts the ciphertext using the same symmetric key

SYMMETRIC ENCRYPTION (cont.)

- Symmetric encryption is also referred to as **single key** or **private key** or **shared secret**.
- Symmetric encryption is very fast.
- It is used for bulk encryption of large amounts of data.
- The main problem is secure distribution and storage of the key, or the exact means by which Alice and Bob "meet" to agree on the key.
- If Mallory intercepts the key and obtains the ciphertext, the security is broken.
- Note that symmetric encryption cannot be used for **authentication** or **integrity**, because Alice and Bob are able to create exactly the same secrets, because they both know the same key.

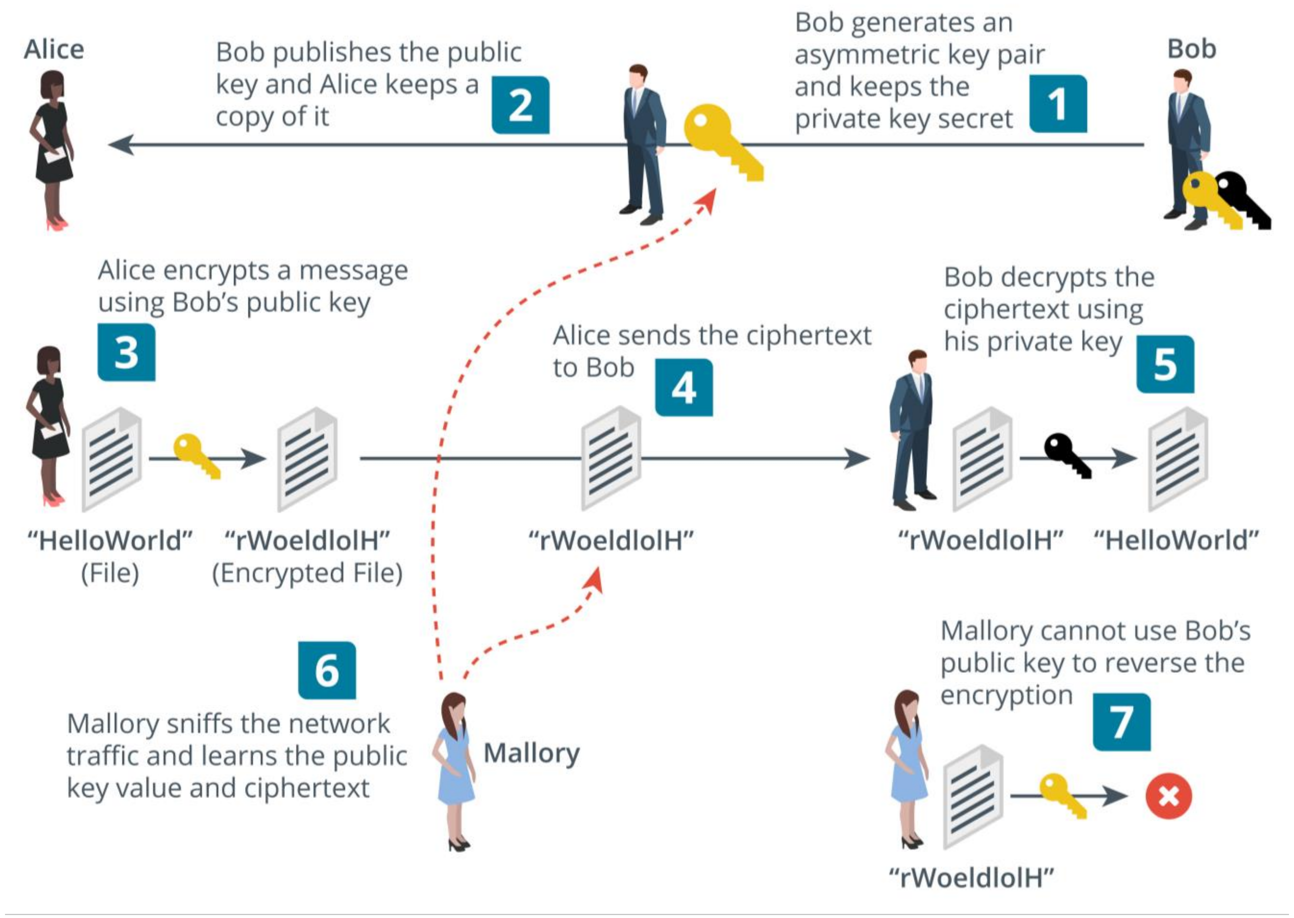
ASYMMETRIC ENCRYPTION

- In a symmetric encryption cipher, the same secret key is used to perform both encryption and decryption operations.
- With an asymmetric cipher, operations are performed by **two different** but **related** public and private keys in a key pair.
- Each key is capable of reversing the operation of its pair.
- **For Example:** if the public key is used to encrypt a message, only the paired private key can decrypt the ciphertext produced.
- The public key cannot be used to decrypt the ciphertext, even though it was used to encrypt it.

ASYMMETRIC ENCRYPTION (cont.)

- The keys are linked in such a way as to make it impossible to derive one from the other.
- This means that the key holder can distribute the public key to anyone he or she wants to receive secure messages from.
- No one else can use the public key to decrypt the messages; only the linked private key can do that.
- Asymmetric encryption can be used to prove identity.
- The holder of a private key cannot be impersonated by anyone else.
- The drawback of asymmetric encryption is that it involves substantial computing overhead compared to symmetric encryption.

ASYMMETRIC ENCRYPTION (cont.)



Alice
Bob publishes the public key and Alice keeps a copy of it
2
Bob generates an asymmetric key pair and keeps the private key secret
1
Bob
Alice encrypts a message using Bob's public key
3
Alice sends the ciphertext to Bob
4
Bob decrypts the ciphertext using his private key
5
"HelloWorld" (File)
"rWoeldlolH" (Encrypted File)
"rWoeldlolH"
"rWoeldlolH"
"HelloWorld"
6
Mallory sniffs the network traffic and learns the public key value and ciphertext
Mallory
Mallory cannot use Bob's public key to reverse the encryption
7
"rWoeldlolH"

ASYMMETRIC ENCRYPTION (cont.)

1. Bob generates a key pair and keeps the private key secret.
2. Bob publishes the public key.
3. Alice wants to send Bob a confidential message, so she takes a copy of Bob's public key.
4. Alice uses Bob's public key to encrypt the message.
5. Alice sends the ciphertext to Bob.
6. Bob receives the message and is able to decrypt it using his private key.
7. If Mallory has been snooping, he can intercept both the message and the public key.
8. However, Mallory cannot use the public key to decrypt the message, so the system remains secure.

PUBLIC KEY CRYPTOGRAPHY ALGORITHMS

- Asymmetric encryption is often referred to as **public key** cryptography.
- Many public key cryptography products are based on the **RSA** algorithm.
- The RSA algorithm provides the mathematical properties for deriving key pairs and performing the encryption and decryption operations.
- This type of algorithm is called a trapdoor function, because it is easy to perform using the public key, but difficult to reverse without knowing the private key.
- **Elliptic curve cryptography (ECC)** is another type of trapdoor function that can be used in public key cryptography ciphers.
- **ECC** used with a key size of **256 bits** is very approximately comparable to **RSA** with a key size of **2048 bits**.