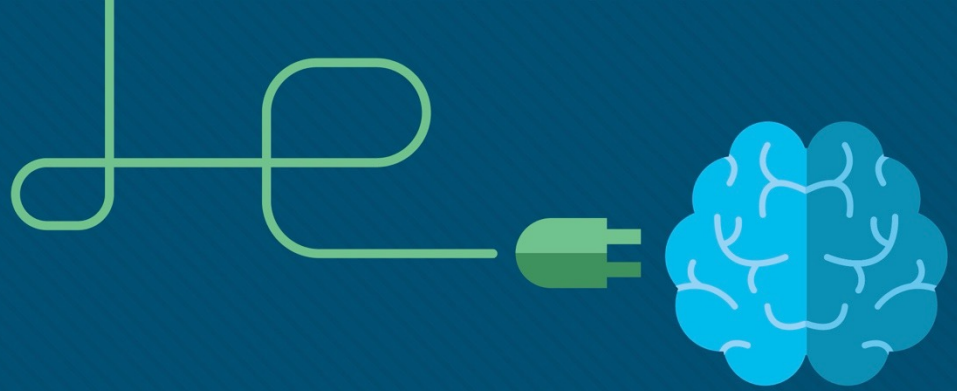


Chapter 7: Cryptography and the Public Key Infrastructure

Information Security

Dr. Ayman Aljarbouh

7.1 Cryptography

Module Objectives

Module Title: Cryptography

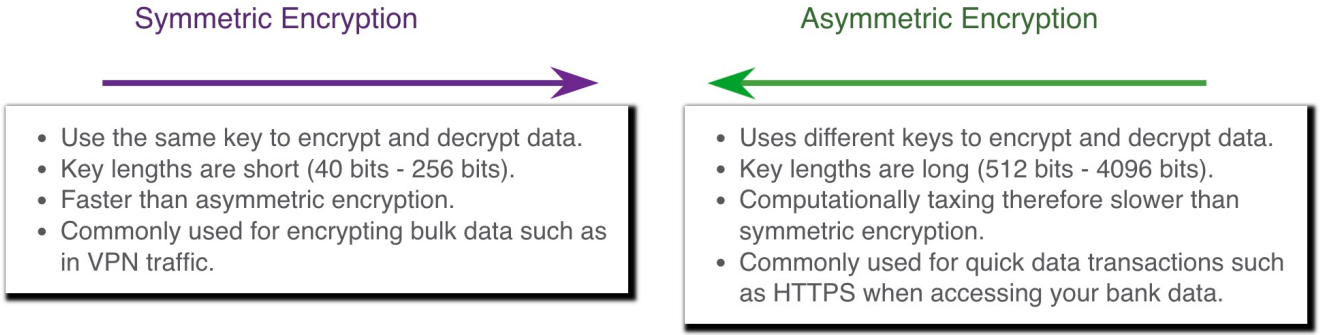
Module Objective: Use tools to encrypt and decrypt data.

Topic Title	Topic Objective
Confidentiality	Explain how cryptographic approaches enhance data confidentiality.

Encryption

These two classes differ in how they use keys:

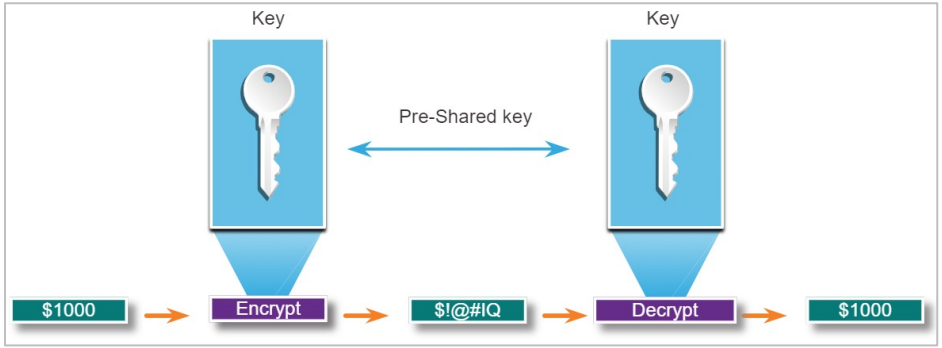
- **Symmetric encryption algorithms** - Encryption algorithms use the same key to encrypt and decrypt data. They are based on the premise that each communicating party knows the pre-shared key.
- **Asymmetric encryption algorithms** - Encryption algorithms use different keys to encrypt and decrypt data. They are based on the assumption that the two communicating parties have not previously shared a secret and must establish a secure method to do so. Asymmetric algorithms are resource intensive and slower to execute.



https://www.youtube.com/watch?v=pArLLJmgX10

Symmetric Encryption

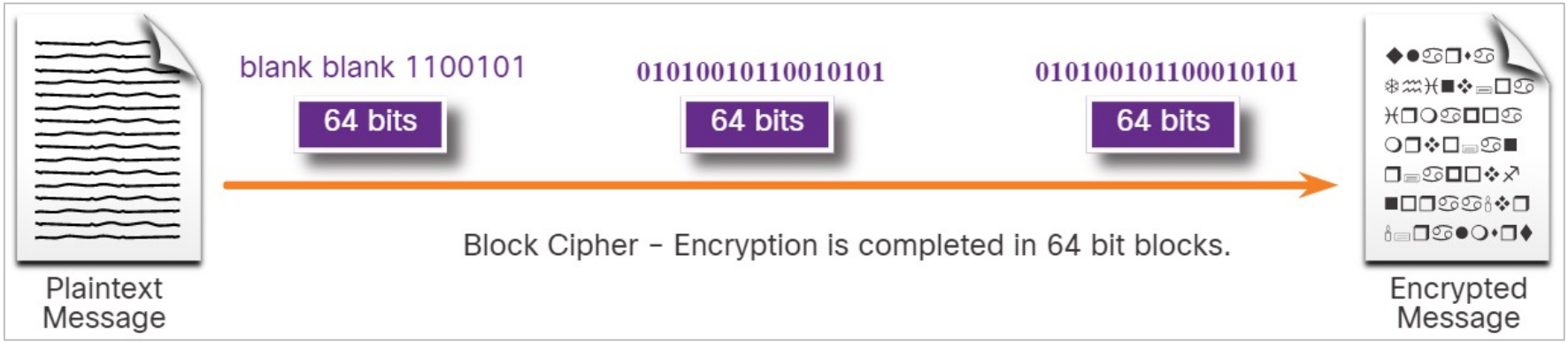
- Symmetric algorithms use the same pre-shared key (secret key) to encrypt and decrypt data.
- Symmetric encryption algorithms are commonly used with VPN traffic because they use less CPU resources than asymmetric encryption algorithms.
- When using these algorithms, the longer the key, the longer it will take for someone to discover the key.
- Most encryption keys are between 112 and 256 bits. Use a longer key for more secure communications.
- Symmetric encryption algorithms are sometimes classified as a block cipher or a stream cipher.



Symmetric Encryption (Contd.)

Block Ciphers

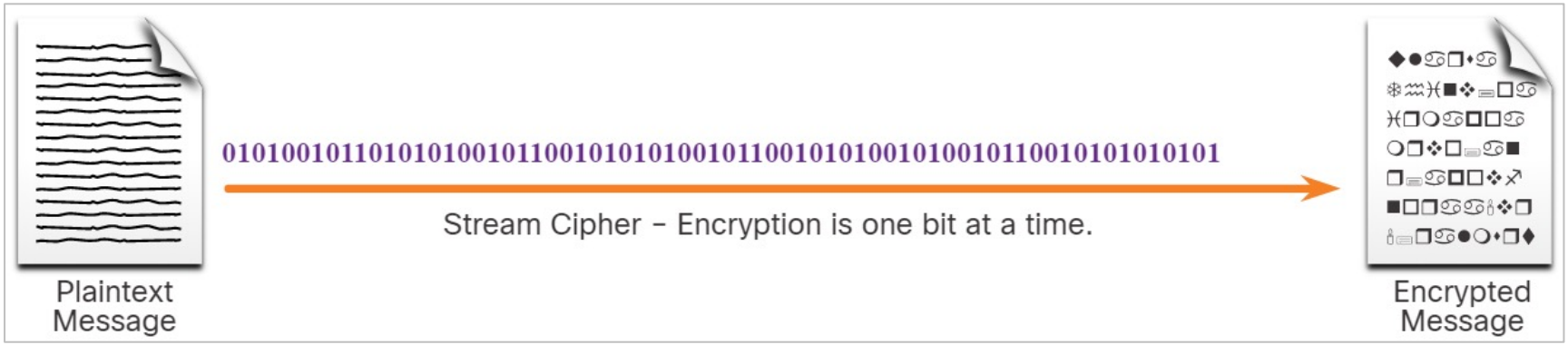
- Block ciphers transform a fixed-length block of plaintext into a common block of ciphertext of 64 or 128 bits.
- Common block ciphers include DES with a 64-bit block size and AES with a 128-bit block size.



Symmetric Encryption (Contd.)

Stream Ciphers

- Stream ciphers encrypt plaintext one byte or one bit at a time.
- Stream ciphers are basically a block cipher with a block size of one byte or bit.
- Stream ciphers are typically faster than block ciphers because data is continuously encrypted.
- Examples include RC4 and A5 which is used to encrypt GSM cell phone communications.



Symmetric Encryption (Contd.)

Well-known symmetric encryption algorithms are described in the table.

Symmetric Encryption Algorithms	Description
Data Encryption Standard (DES)	This is a legacy algorithm. It uses a short key length that makes it insecure.
3DES (Triple DES)	This is the replacement for DES and repeats the DES algorithm three times. It should be avoided as it is scheduled to be retired in 2023. If implemented, use very short key lifetimes.
Advanced Encryption Standard (AES)	It offers combinations of 128-, 192-, or 256-bit keys to encrypt 128, 192, or 256 bit-long data blocks.
Software-Optimized Encryption Algorithm (SEAL)	It is a stream cipher that uses a 160-bit encryption key and has a lower impact on the CPU compared to other software-based algorithms.
Rivest ciphers (RC) series algorithms	RC4 is a stream cipher that was used to secure web traffic. It has been found to have multiple vulnerabilities which have made it insecure. RC4 should not be used.

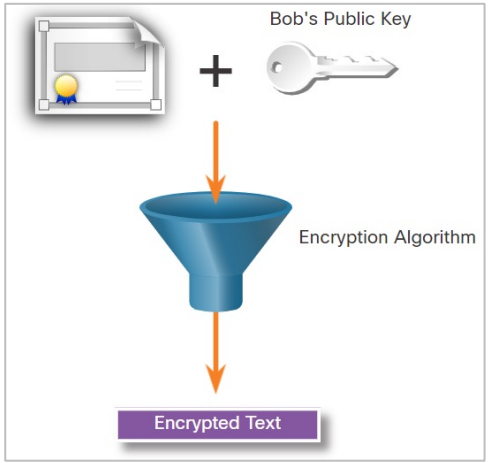
Asymmetric Encryption

- Asymmetric algorithms, also called public-key algorithms, are designed so that the key that is used for encryption is different from the key that is used for decryption.
- The decryption key cannot, in any reasonable amount of time, be calculated from the encryption key and vice versa.
- Asymmetric algorithms use a public key and a private key.
- Both keys are capable of the encryption process, but the complementary paired key is required for decryption.
- The process is also reversible in that data encrypted with the public key requires the private key to decrypt.
- This process enables asymmetric algorithms to achieve confidentiality, authentication, and integrity.

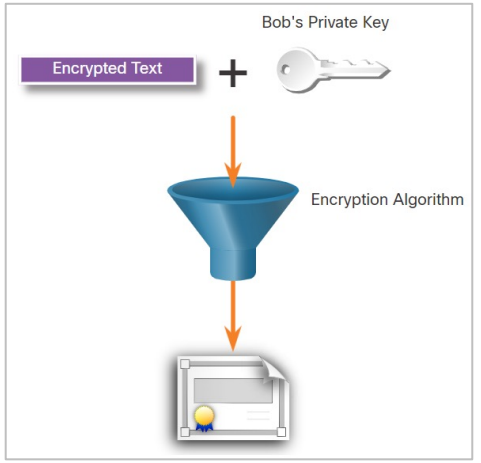
Asymmetric Encryption - Confidentiality

- Asymmetric algorithms are used to provide confidentiality without pre-sharing a password.
- The confidentiality objective of asymmetric algorithms is initiated when the encryption process is started with the public key.
- The process can be summarized using the formula: **Public Key (Encrypt) + Private Key (Decrypt) = Confidentiality**
- When the public key is used to encrypt data, the private key must be used to decrypt data.
- Only one host has the private key; therefore, confidentiality is achieved.

Example: **Data exchange between Bob and Alice**



Alice acquires and uses Bob's public key to encrypt a message and then send it to Bob.



Bob decrypts the message with the private key and as he is the only one with the private key, confidentiality is achieved.

Asymmetric Encryption - Authentication

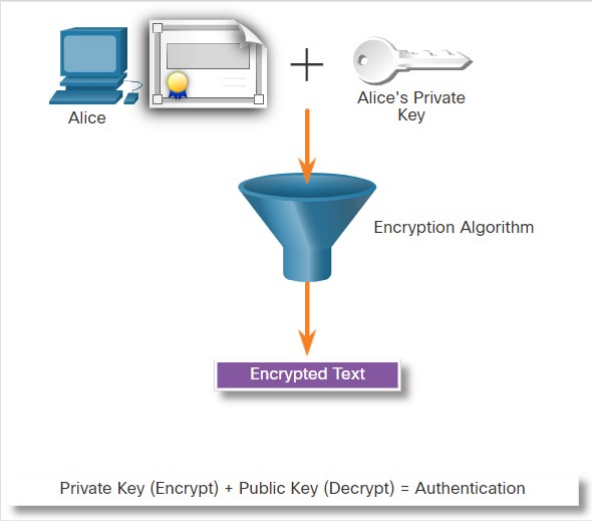
- The authentication objective of asymmetric algorithms is initiated with the private key encryption process.
- The process can be summarized using the formula: **Private Key (Encrypt) + Public Key (Decrypt) = Authentication**
- When the private key is used to encrypt the data, the corresponding public key must be used to decrypt the data.
- Because only one host has the private key, only that host could have encrypted the message, providing authentication of the sender.
- When a host successfully decrypts a message using a public key, it is trusted that the private key encrypted the message, which verifies who the sender is. This is a form of authentication.

Asymmetric Encryption - Authentication (Contd.)

- Let's see how the private and public keys can be used to provide authentication to the data exchange between Bob and Alice.

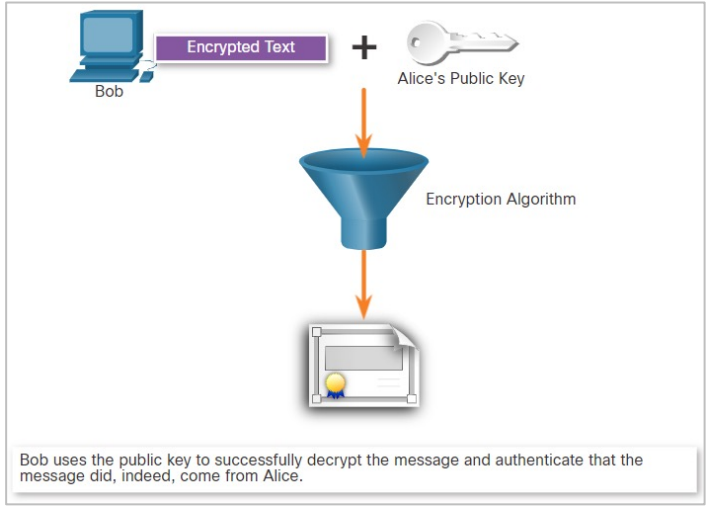
Alice uses her private key

Alice encrypts a message using her private key and sends it to Bob.



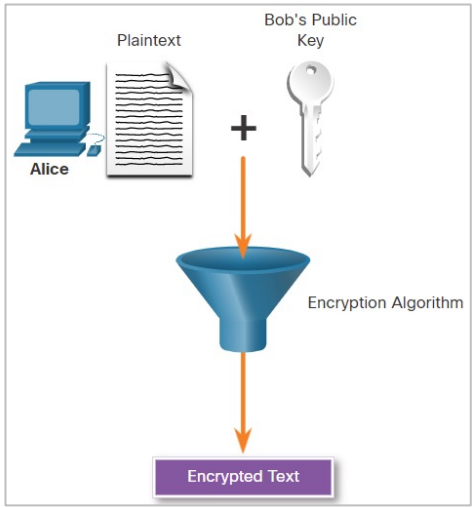
Bob decrypts using the public key

After Bob obtains Alice's public key, he uses it to decrypt the message and to authenticate that the message has been received from Alice.

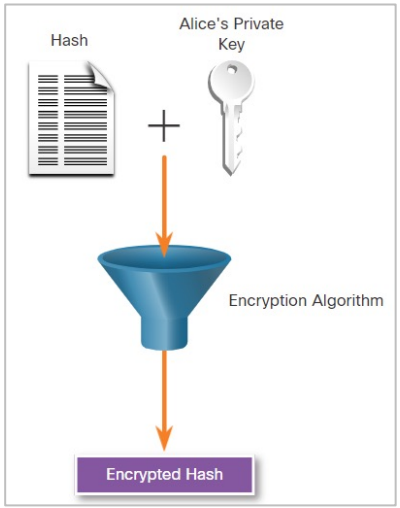


Asymmetric Encryption - Integrity

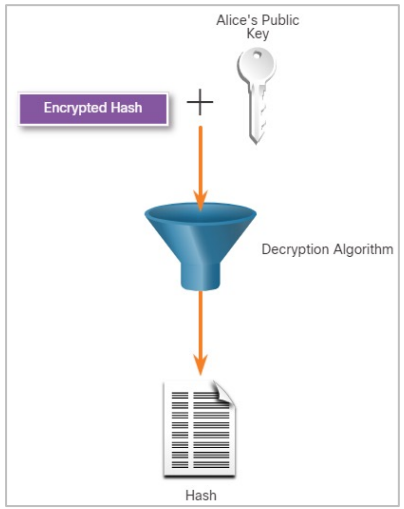
- Combining the two asymmetric encryption processes provides message confidentiality, authentication, and integrity. In this example, a message will be ciphered using Bob's public key and a ciphered hash will be encrypted using Alice's private key.



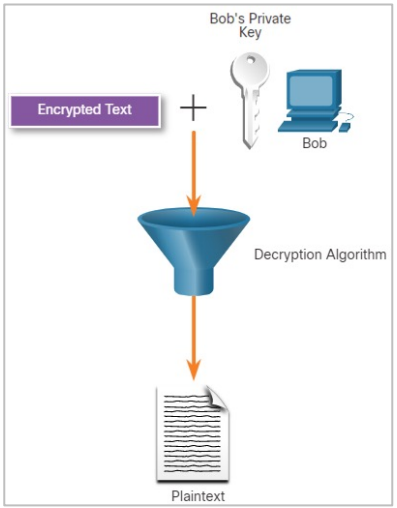
Alice uses Bob's Public Key



Alice encrypts a hash using her private key



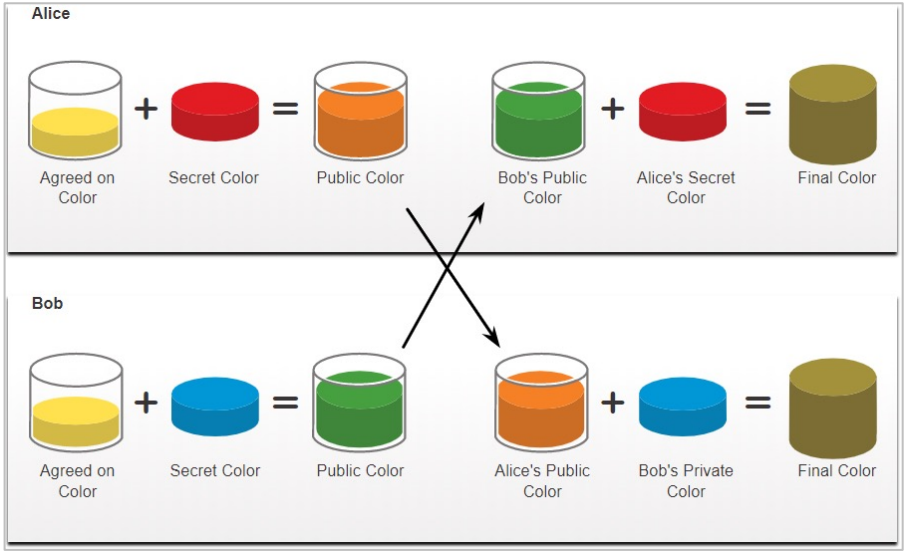
Bob uses Alice's public key to decrypt the hash



Bob uses his private key to decrypt the message

Diffie-Hellman

- Diffie-Hellman (DH) is an asymmetric mathematical algorithm that allows two computers to generate an identical shared secret without having communicated before.
  - The new shared key is never actually exchanged between the sender and receiver.
  - The key can be used by an encryption algorithm to encrypt traffic between the two systems as both parties know it.
  - Following are two examples of instances when DH is commonly used:
    - Data is exchanged using an IPsec VPN
    - SSH data is exchanged
  - The security of DH is based on the fact that it uses very large numbers in its calculations.



DH operation

https://www.youtube.com/watch?v=KXq065YrpiU

New Terms and Commands

• Diffie-Hellman (DH) • Advanced Encryption Standard (AES) • Data Encryption Standard (DES) • Software-Optimized Encryption Algorithm (SEAL)	• asymmetric encryption • symmetric encryption • block ciphers • stream ciphers

Lab 27 - Encrypting and Decrypting Data Using OpenSSL

In this lab, you will complete the following objectives:

- Encrypting Messages with OpenSSL
- Decrypting Messages with OpenSSL

Lab 28 - Encrypting and Decrypting Data Using a Hacker Tool

In this lab, you will complete the following objectives:

- Setup Scenario
- Create and Encrypt Files
- Recover Encrypted Zip File Passwords