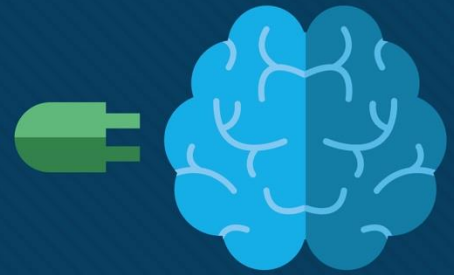


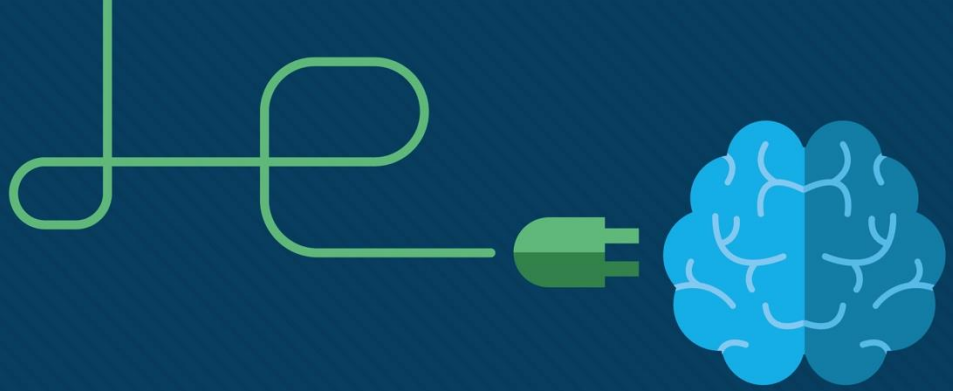
Chapter 9: Cryptography and the Public Key Infrastructure

Instructor Materials

CCNA Cybersecurity Operations v1.1

Chapter 9: Cryptography and the Public Key Infrastructure

CCNA Cybersecurity Operations v1.1
Planning Guide



Chapter 9: Cryptography and the Public Key Infrastructure

CCNA Cybersecurity Operations v1.1

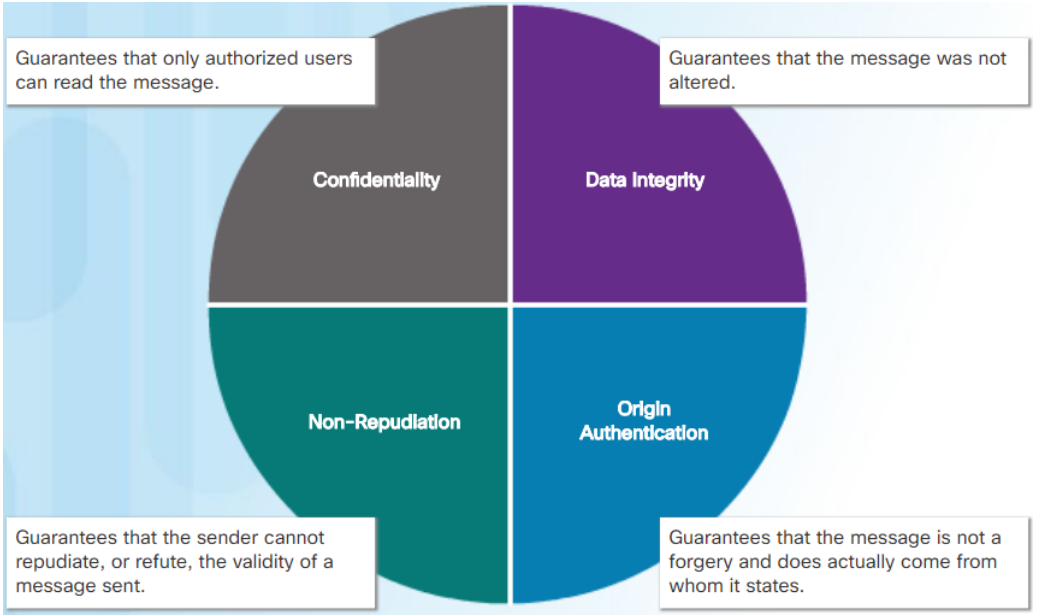
Chapter 9 - Sections & Objectives

- 9.1 Cryptography
  - Use tools to encrypt and decrypt data.
    - Use cryptography to secure communications.
    - Explain the role of cryptography in ensuring the integrity and authenticity of data.
    - Explain how cryptographic approaches enhance data confidentiality.
- 9.2 Public Key Cryptography
  - Explain how the public key infrastructure (PKI) supports network security.
    - Explain public key cryptography.
    - Explain how the public key infrastructure functions.
    - Explain how the use of cryptography affects cybersecurity operations.

9.1 Cryptography

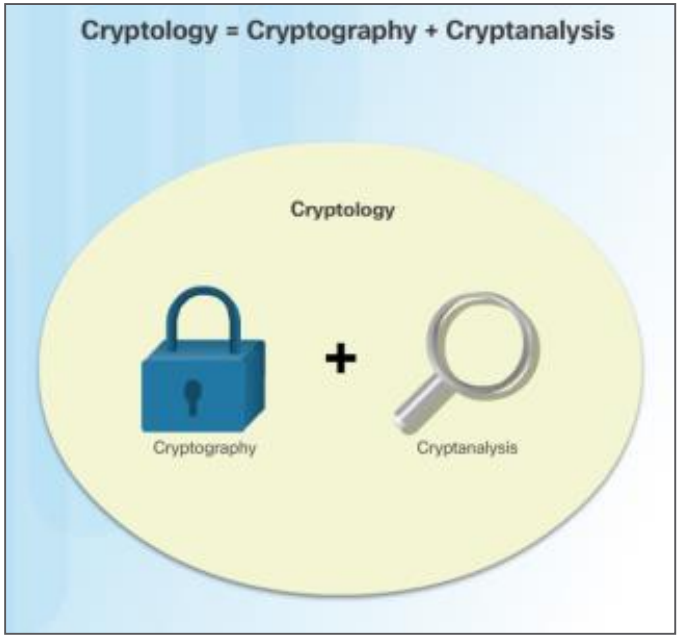
Securing Communications

- Information security concerns protecting network infrastructure devices and securing data as it travels on the network.
- Cryptography helps realize the four objectives of information security:
  - **Data Confidentiality** - only authorized users can read the data.
  - **Data Integrity** - the data has not been altered by unauthorized parties.
  - **Origin authentication** - the data has actually originated at the expected source.
  - **Non-repudiation** – the integrity of the message is irrefutable by the sender.



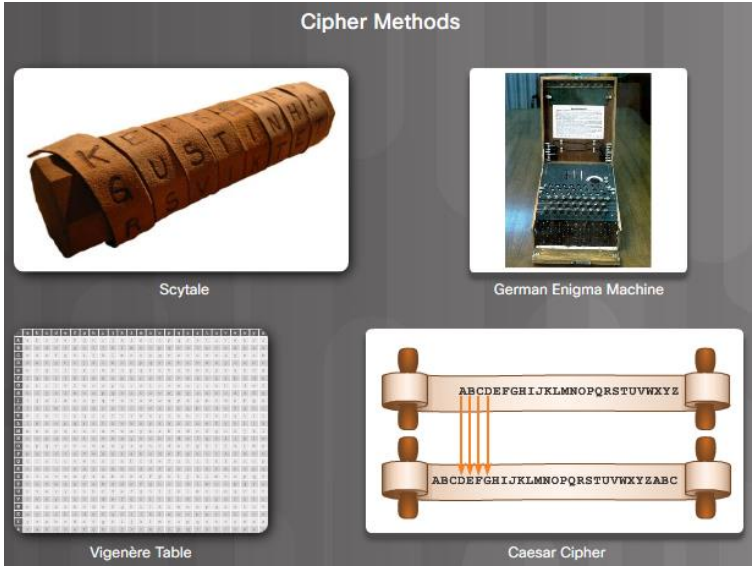
Cryptology

- Cryptology is the science of making and breaking secret codes. There are two disciplines:
  - **Cryptography** –This is the development and use of codes that are used for communicating privately. Specifically, it is the practice and study of techniques to secure communications.
  - **Cryptanalysis** – This is the breaking of those codes. Specifically, it is the practice and study of determining and exploiting weaknesses in cryptographic techniques.



Cryptography – Ciphers

- A cipher is an algorithm that consists of a series of well-defined steps that can be followed as a procedure when encrypting and decrypting messages.
- The following are types of ciphers that have been used over the years:
  - **Substitution cipher** – Substitution ciphers retain the letter frequency of the original message.
  - **Transposition cipher** - In transposition ciphers, no letters are replaced; they are simply rearranged.
  - **Polyalphabetic ciphers** - Polyalphabetic ciphers are based on substitution, using multiple substitution alphabets.



Cryptanalysis – Code Breaking

- A number of code breaking (cryptanalysis) methods exist, such as brute-force, ciphertext, and known-plaintext, among others.

- Several methods are used in cryptanalysis:
  - **Brute-force** - The cryptanalyst tries every possible key knowing that eventually one of them will work.
  - **Ciphertext** - The cryptanalyst has the ciphertext of several encrypted messages but no knowledge of the underlying plaintext.
  - **Known-Plaintext** - The cryptanalyst has access to the ciphertext of several messages and knows something about the plaintext underlying that ciphertext.
  - **Chosen-Plaintext** - The cryptanalyst chooses which data the encryption device encrypts and observes the ciphertext output.
  - **Chosen-Ciphertext** - The cryptanalyst can choose different ciphertext to be decrypted and has access to the decrypted plaintext.
  - **Meet-in-the-Middle** - The cryptanalyst knows a portion of the plaintext and the corresponding ciphertext.

Keys

- With modern technology, security of encryption lies in the secrecy of the keys, not the algorithm.

Two terms that are used to describe keys are:

- **Key length** - Also called the key size, this is measured in bits. In this course, we will use the term key length.
- **Keyspace** - This is the number of possibilities that can be generated by a specific key length.

- As key length increases, the keyspace increases exponentially.

DES Key	Keyspace	# of Possible Keys
56-bit	2^{56} 11111111 11111111 11111111 11111111 11111111 11111111 11111111	72,000,000,000,000,000
57-bit	2^{57} 11111111 11111111 11111111 11111111 11111111 11111111 11111111 1	144,000,000,000,000,000
58-bit	2^{58} 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11	288,000,000,000,000,000
59-bit	2^{59} 11111111 11111111 11111111 11111111 11111111 11111111 11111111 111	576,000,000,000,000,000
60-bit	2^{60} 11111111 11111111 11111111 11111111 11111111 11111111 11111111 1111	1,152,000,000,000,000,000

Lab – Encrypting and Decrypting Data Using OpenSSL

Networking Academy

Lab - Encrypting and Decrypting Data Using OpenSSL

Objectives

Part 1: Encrypting Messages with OpenSSL

Part 2: Decrypting Messages with OpenSSL

Background / Scenario

OpenSSL is an open source project that provides a robust, commercial-grade, and full-featured toolkit for the Transport Layer Security (TLS) and Secure Sockets Layer (SSL) protocols. It is also a general-purpose cryptography library. In this lab, you will use OpenSSL to encrypt and decrypt text messages.

Note: While OpenSSL is the de facto cryptography library today, the use presented in this lab is NOT recommended for robust protection. Below are two security problems with this lab:

1) The method described in this lab uses a weak key derivation function. The ONLY security is introduced by a very strong password.
2) The method described in this lab does not guarantee the integrity of the text file.

This lab should be used for instructional purposes only. The methods presented here should NOT be used to secure truly sensitive data.

Required Resources

- CyberOps Workstation Virtual Machine
- Internet access

Lab – Encrypting and Decrypting Data Using a Hacker Tool

Networking Academy

Lab - Encrypting and Decrypting Data using a Hacker Tool

Objectives

Part 1: Create and Encrypt Files

Part 2: Recover Encrypted Zip File Passwords

Background / Scenario

What if you work for a large corporation that had a corporate policy regarding removable media? Specifically, it states that only encrypted zipped documents can be copied to portable USB flash drives.

In this scenario, the Chief Financial Officer (CFO) is out-of-town on business and has contacted you in a panic with an emergency request for help. While out-of-town on business, he attempted to unzip important documents from an encrypted zip file on a USB drive. However, the password provided to open the zip file is invalid. The CFO contacted you to see if there was anything you could to do.

Note: The provided scenario is simple and only serves as an example.

There may some tools available to recover lost passwords. This is especially true in situations such as this where the cybersecurity analyst could acquire pertinent information from the CFO, such as the length of the password, and an idea of what it could be. Knowing pertinent information dramatically helps when attempting to recover passwords.

Examples of password recovery utilities and programs include hashcat, John the Ripper, Lophtcrack, and others. In our scenario, we will use **fcrackzip** which is a simple Linux utility to recover the passwords of encrypted zip files.

Lab – Examining Telnet and SSH in Wireshark

Networking Academy

Lab - Examining Telnet and SSH in Wireshark

Objectives

Part 1: Examine a Telnet Session with Wireshark

Part 2: Examine an SSH Session with Wireshark

Background / Scenario

In this lab, you will configure a router to accept SSH connectivity and use Wireshark to capture and view Telnet and SSH sessions. This will demonstrate the importance of encryption with SSH.

Required Resources

- CyberOps Workstation VM

Part 1: Examining a Telnet Session with Wireshark

You will use Wireshark to capture and view the transmitted data of a Telnet session.

Step 1: Capture data.

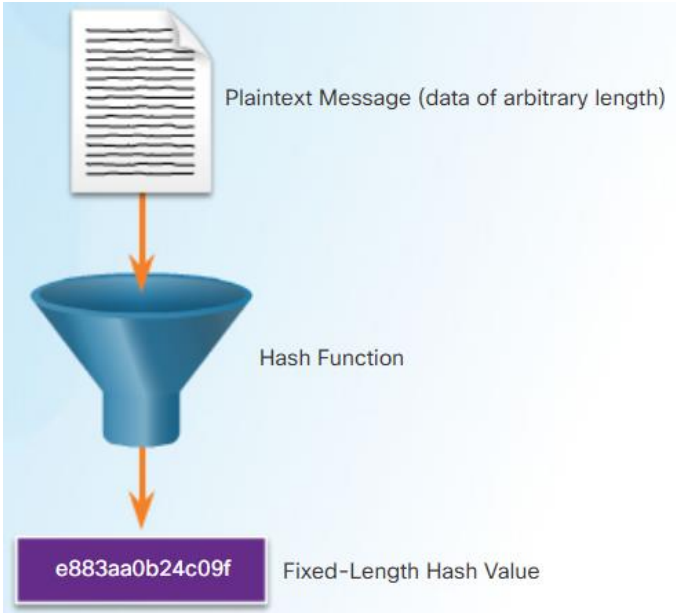
a. Start the CyberOps Workstation VM and log in with username **analyst** and password **cyberops**.

b. Open a terminal window and start Wireshark. Press **OK** to continue after reading the warning message.

```
[analyst@secOps analyst]$ sudo wireshark-gtk
[sudo] password for analyst: cyberops
```

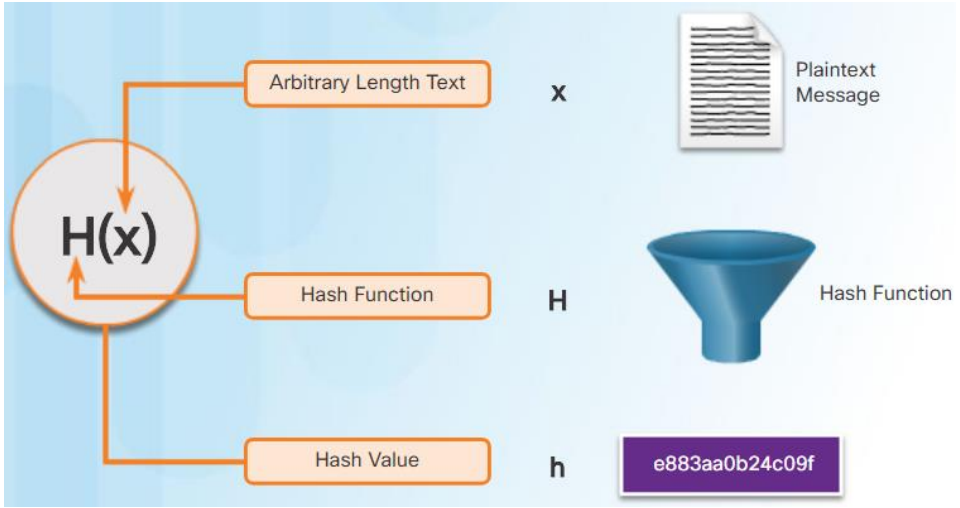
Cryptographic Hash Functions

- Cryptographic hashes are used to verify and ensure data integrity.
- Hashing is based on a one-way mathematical function that is relatively easy to compute, but significantly harder to reverse.
- The cryptographic hashing function can also be used to verify authentication.
- A hash function takes a variable block of binary data, called the message, and produces a fixed-length, condensed representation, called the hash.
- The resulting hash is also sometimes called the message digest, digest, or digital fingerprint.
- With hash functions, it is computationally infeasible for two different sets of data to come up with the same hash output.
- Every time the data is changed or altered, the hash value also changes.



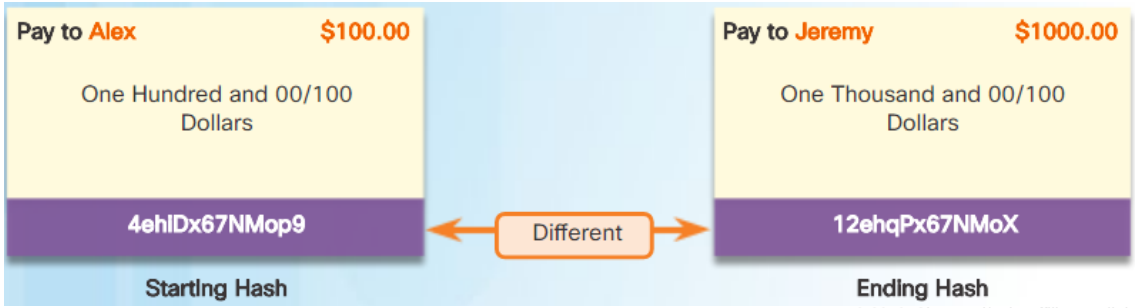
Cryptographic Hash Operation

- Mathematically, the equation ***h= H(x)*** is used to explain how a hash algorithm operates.
- A cryptographic hash function should have the following properties:
  - The input can be any length.
  - The output has a fixed length.
  - H(x) is relatively easy to compute for any given x.
  - H(x) is one way and not reversible.
  - H(x) is collision free, meaning that two different input values will result in different hash values.



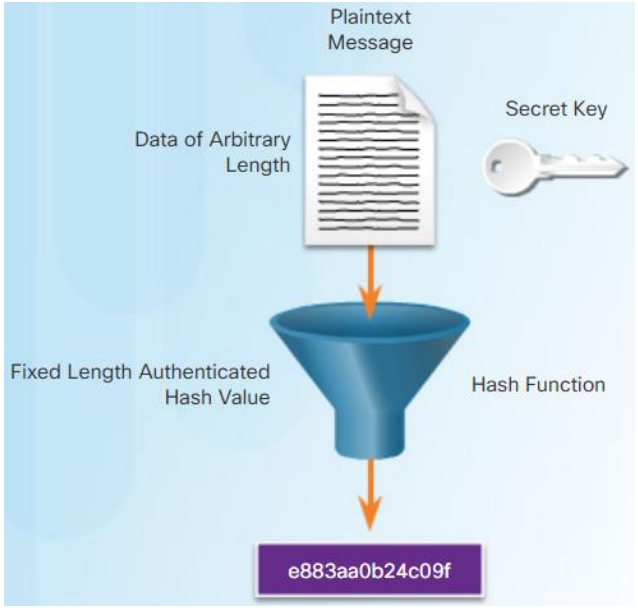
MD5 and SHA

- Hash functions are used to ensure the integrity of a message. They ensure data has not changed accidentally or intentionally.
- Three well-known hashing algorithms are 128-bit MD5, SHA-1, and SHA-2.
  - **MD5 with 128-bit digest** - A one-way function that produces a 128-bit hashed message. MD5 is considered to be a legacy algorithm. It is recommended that SHA-2 be used instead.
  - **SHA-1** – Very similar to the MD5 hash functions. Several versions exist. SHA-1 creates a 160 bit hashed message and is slightly slower than MD5. SHA-1 has known flaws and is a legacy algorithm.
  - **SHA-2** –Next-generation algorithm and should be used whenever possible.
- While hashing can be used to detect accidental changes, it cannot be used to guard against deliberate changes. There is no unique identifying information from the sender in the hashing procedure.



Hash Message Authentication Code

- To add authentication to integrity assurance, a keyed-hash message authentication code (HMAC) is used.
- To add authentication, HMAC uses an additional secret key as input to the hash function.
- Only the sender and the receiver know the secret key, and the output of the hash function now depends on the input data and the secret key.
- Only parties who have access to that secret key can compute the digest of an HMAC function.
- If the digest that is calculated by the receiving device is equal to the digest that was sent, the message has not been altered.



Lab – Hashing Things Out

Networking Academy

Lab – Hashing Things Out

Objectives

Part 1: Creating Hashes with OpenSSL

Part 2: Verifying Hashes

Background / Scenario

Hash functions are mathematical algorithms designed to take data as input and generate a fixed-size, unique string of characters, also known as the hash. Designed to be fast, hash functions are very hard to reverse; it is very hard to recover the data that created any given hash, based on the hash alone. Another important property of hash functions is that even the smallest change done to the input data yields a completely different hash.

While OpenSSL can be used to generate and compare hashes, other tools are available. Some of these tools are also included in this lab.

Required Resources

- CyberOps Workstation VM
- Internet access

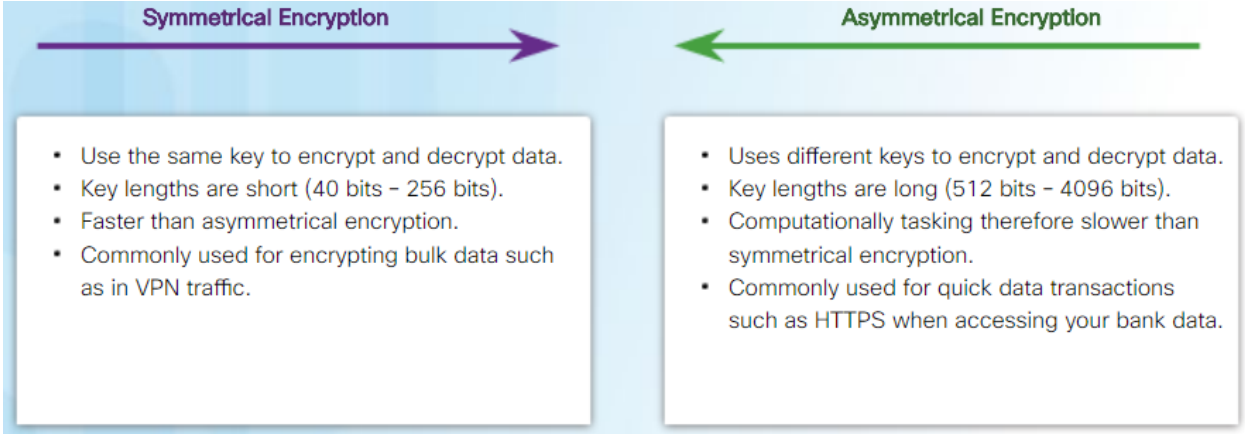
Part 1: Creating Hashes with OpenSSL

OpenSSL can be used as a standalone tool for hashing. To create a hash of a text file, follow the steps below:

Encryption

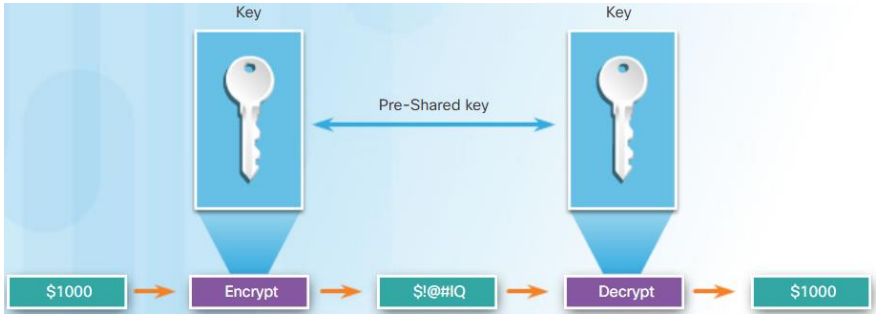
These two classes differ in how they use keys:

- **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.



Symmetric Encryption

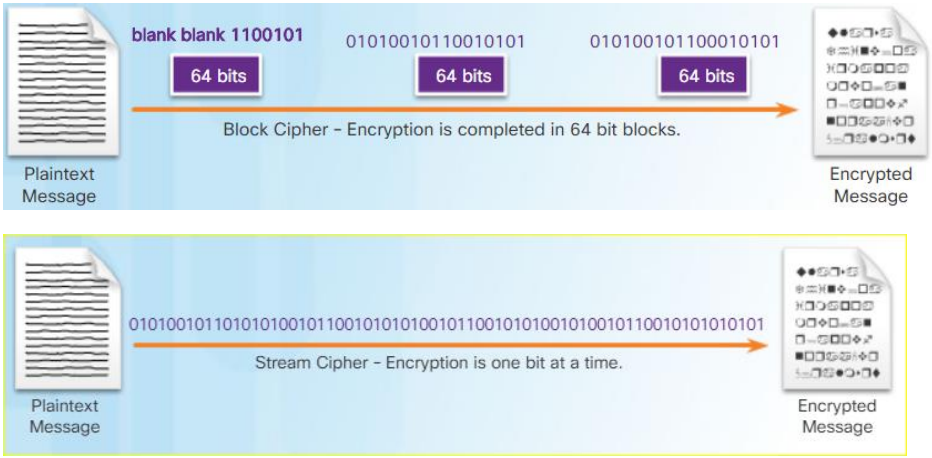
- Symmetric algorithms use the same pre-shared key to encrypt and decrypt data.
- Today, symmetric encryption algorithms are commonly used with VPN traffic. This is because symmetric algorithms use less CPU than asymmetric encryption algorithms.
- When using symmetric encryption algorithms, like any other type of encryption, 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

Encryption algorithms are often classified as either:

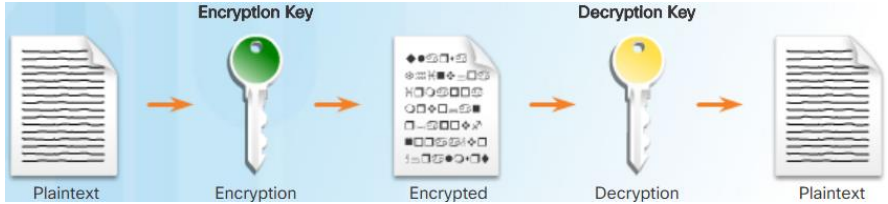
- **Block ciphers** - Block ciphers transform a fixed-length block of plaintext into a common block of ciphertext of 64 or 128 bits.
- **Stream Ciphers** - Stream ciphers encrypt plaintext one byte or one bit at a time.



Well-known symmetric encryption algorithms include: **Data Encryption Standard (DES), 3DES (Triple DES), Advanced Encryption Standard, (AES) Software-Optimized Encryption Algorithm (SEAL), Rivest ciphers (RC)**

Asymmetric Encryption Algorithms

- 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.

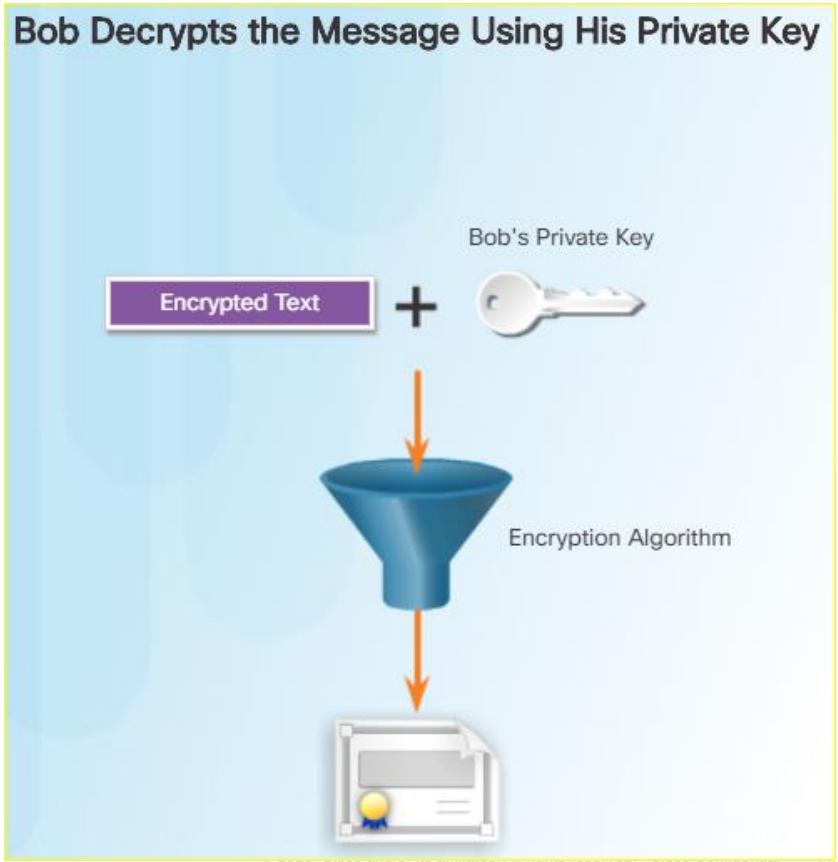


Asymmetric Encryption – Confidentiality

- 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 the data, the private key must be used to decrypt the data.
- Only one host has the private key.

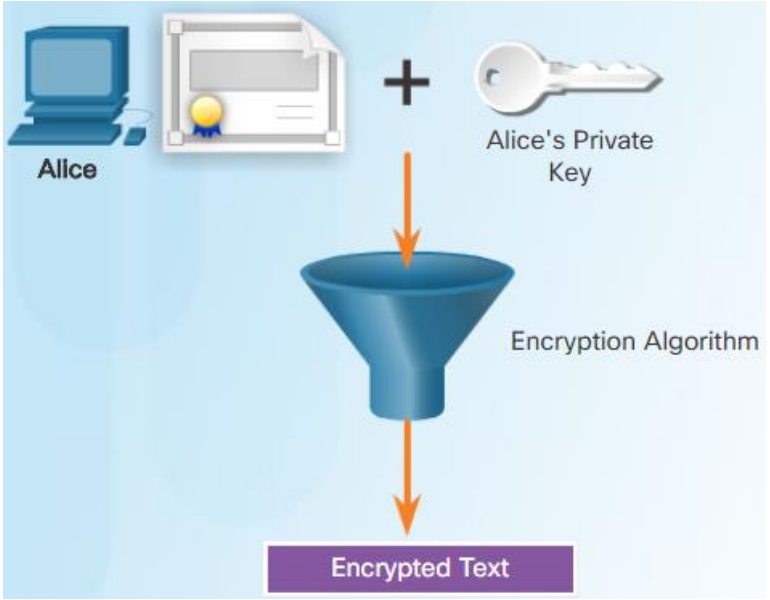


Asymmetric Encryption – Authentication

- The authentication objective of asymmetric algorithms is initiated with the private key encryption process.

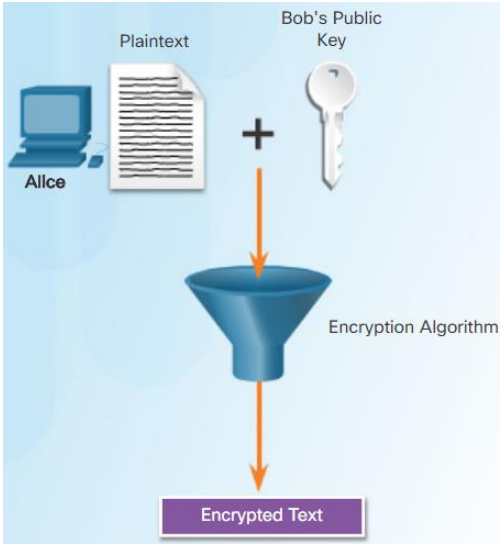
The process can be summarized using the formula

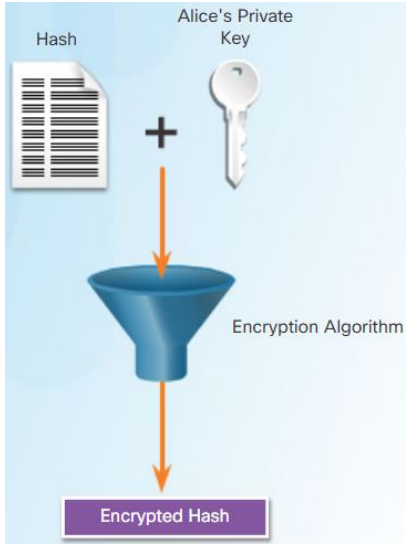
- **Private Key (Encrypt) + Public Key (Decrypt) = Authentication**
- When the private key is used to encrypt the data, the corresponding public key must be used to decrypt the data.
- Because only one host has the private key, only that host could have encrypted the message, providing authentication 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 the sender.

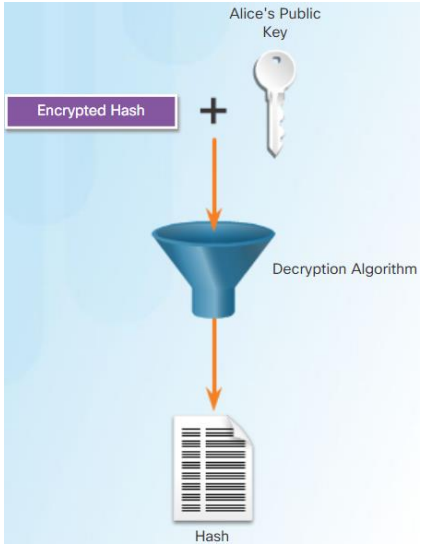


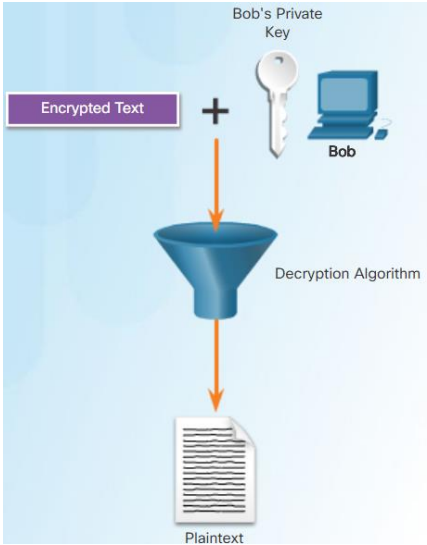
Asymmetric Encryption – Integrity

- Combining the two asymmetric encryption processes provides message confidentiality, authentication, and integrity.



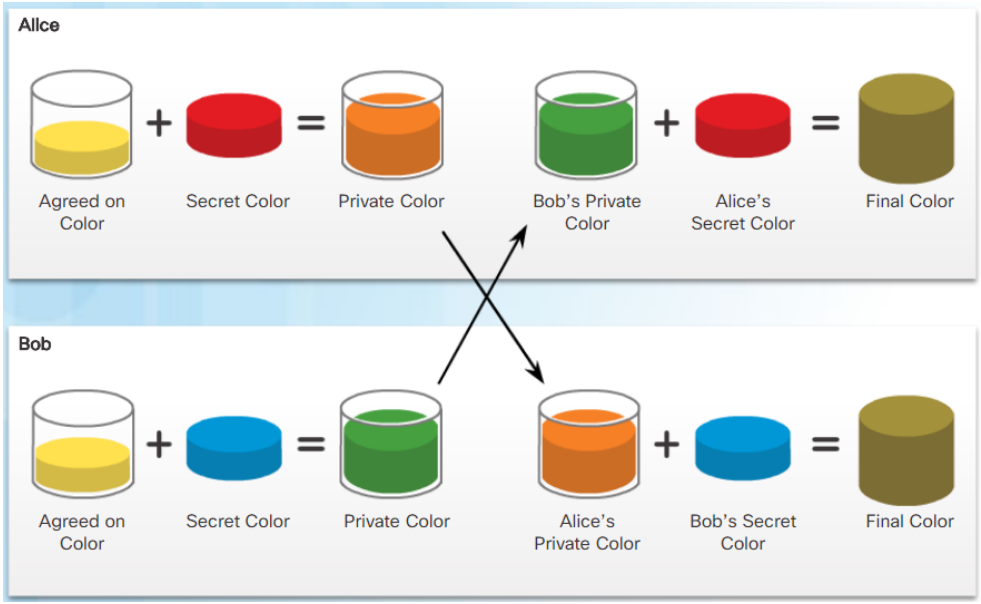






Diffie-Hellman

- 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.
- However, because both parties know it, the key can be used by an encryption algorithm to encrypt traffic between the two systems.
- The security of DH is based on the fact that it uses unbelievably large numbers in its calculations.
- Unfortunately, asymmetric key systems are extremely slow for any sort of bulk encryption. This is why it is common to encrypt the bulk of the traffic using a symmetric algorithm.



9.2 Public Key Infrastructure

Using Digital Signatures

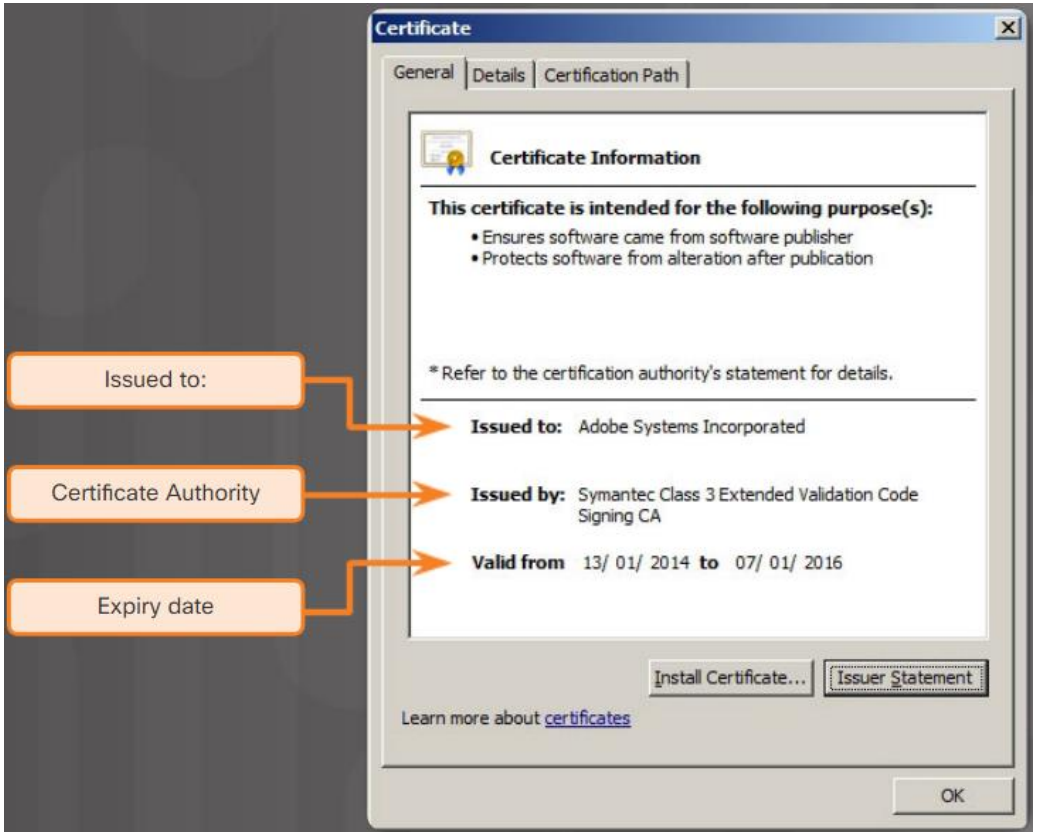
- Digital signatures are a mathematical technique used to provide authenticity, integrity, and nonrepudiation in the form of code signing and digital certificates.
- Digital signatures are commonly used in the following two situations:
  - **Code signing** –Code signing is used to verify the integrity of executable files downloaded from a vendor website.
  - **Digital certificates** – These are used to authenticate the identity of a system and exchange confidential data.
- There are three Digital Signature Standard (DSS) algorithms used for generating and verifying digital signatures:
  - **Digital Signature Algorithm (DSA)**
  - **Rivest-Shamir Adelman Algorithm (RSA)**
  - **Elliptic Curve Digital Signature Algorithm (ECDSA)**

Digital Signatures for Code Signing

- Digital signatures are commonly used to provide assurance of the authenticity and integrity of software code.
- Executable files are wrapped in a digitally signed envelope, which allows the end user to verify the signature before installing the software.
- Digitally signing code provides several assurances about the code:
  - The code is authentic and is actually sourced by the publisher.
  - The code has not been modified since it left the software publisher.
  - The publisher undeniably published the code. This provides nonrepudiation of the act of publishing.

Digital Signatures for Digital Certificates

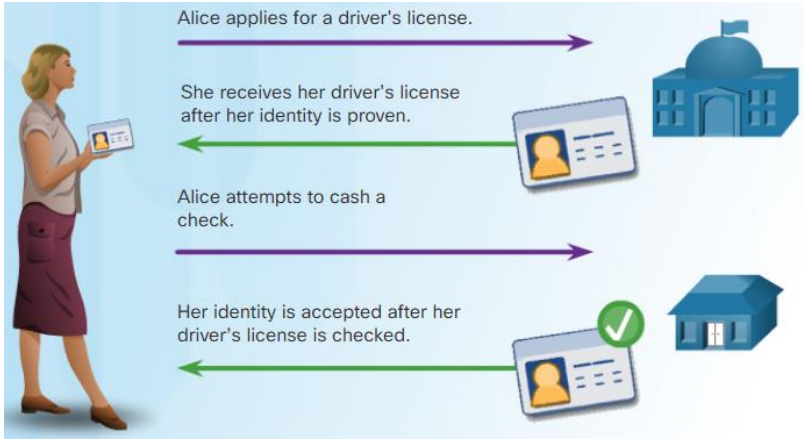
- A digital certificate enables users, hosts, and organizations to securely exchange information over the Internet.
- Specifically, a digital certificate is used to authenticate and verify that users sending a message are who they claim to be.
- Digital certificates can also be used to provide confidentiality for the receiver with the means to encrypt a reply.



Public Key Management

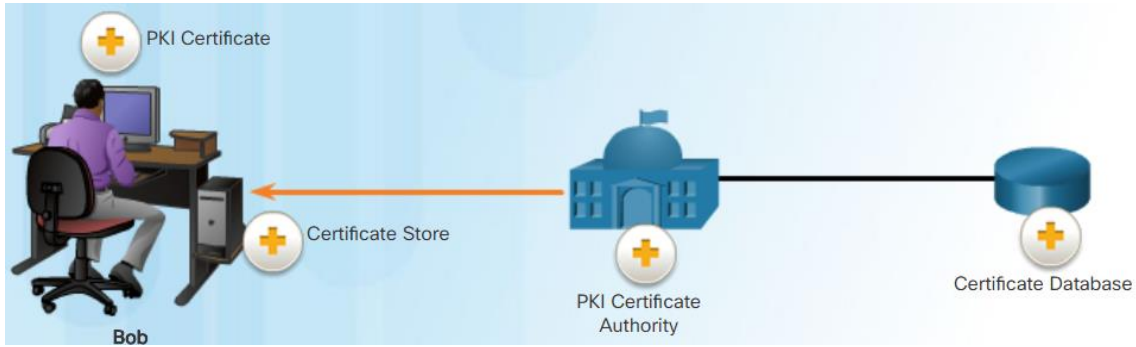
- When establishing an asymmetric connection between two hosts, the hosts will exchange their public key information.
- Trusted third parties on the Internet validate the authenticity of these public keys using digital certificates. The third party issues credentials that are difficult to forge.
- From that point forward, all individuals who trust the third party simply accept the credentials that the third party issues.

- The Public Key Infrastructure (PKI) is an example of a trusted third-party system referred to as certificate authority (CA).
- The CA issues digital certificates that authenticate the identity of organizations and users.
- These certificates are also used to sign messages to ensure that the messages have not been tampered with.



The Public Key Infrastructure

- PKI is needed to support large-scale distribution and identification of public encryption keys.
- The PKI framework facilitates a highly scalable trust relationship.
- It consists of the hardware, software, people, policies, and procedures needed to create, manage, store, distribute, and revoke digital certificates.
- Not all PKI certificates are directly received from a CA. A registration authority (RA) is a subordinate CA and is certified by a root CA to issue certificates for specific uses.



The PKI Authorities System

- Many vendors provide CA servers as a managed service or as an end-user product.
- Organizations may also implement private PKIs using Microsoft Server or Open SSL.
- CAs issue certificates based on classes which determine how trusted a certificate is.
- The class number is determined by how rigorous the procedure was that verified the identity of the holder when the certificate was issued.
- The higher the class number, the more trusted the certificate.
- Some CA public keys are preloaded, such as those listed in web browsers.
- An enterprise can also implement PKI for internal use.

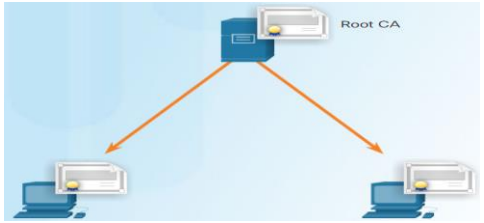
Class	Description
0	Used for testing purposes in which no checks have been performed.
1	Used for individuals with a focus on verification of email.
2	Used for organizations for which proof of identity is required.
3	Used for servers and software signing for which independent verification and checking of identity and authority is done by the issuing certificate authority.
4	Used for online business transactions between companies.
5	Used for private organizations or governmental security.

The PKI Trust System

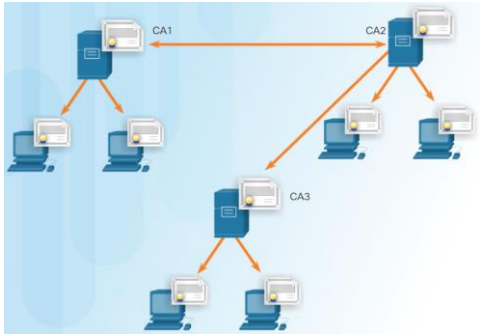
- PKIs can form different topologies of trust. The simplest is the single-root PKI topology.

On larger networks, PKI CAs may be linked using two basic architectures:

- **Cross-certified CA topologies** - This a peer-to-peer model in which individual CAs establish trust relationships with other CAs by cross-certifying CA certificates.
- **Hierarchical CA topologies** - The highest level CA is called the root CA. It can issue certificates to end users and to a subordinate CA.

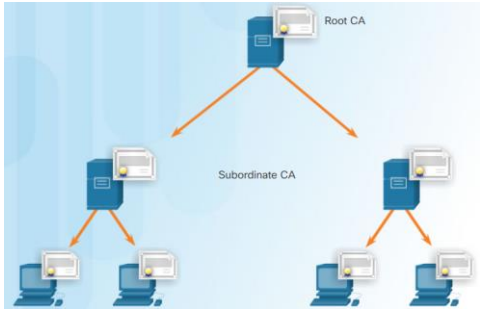


Single-Root PKI



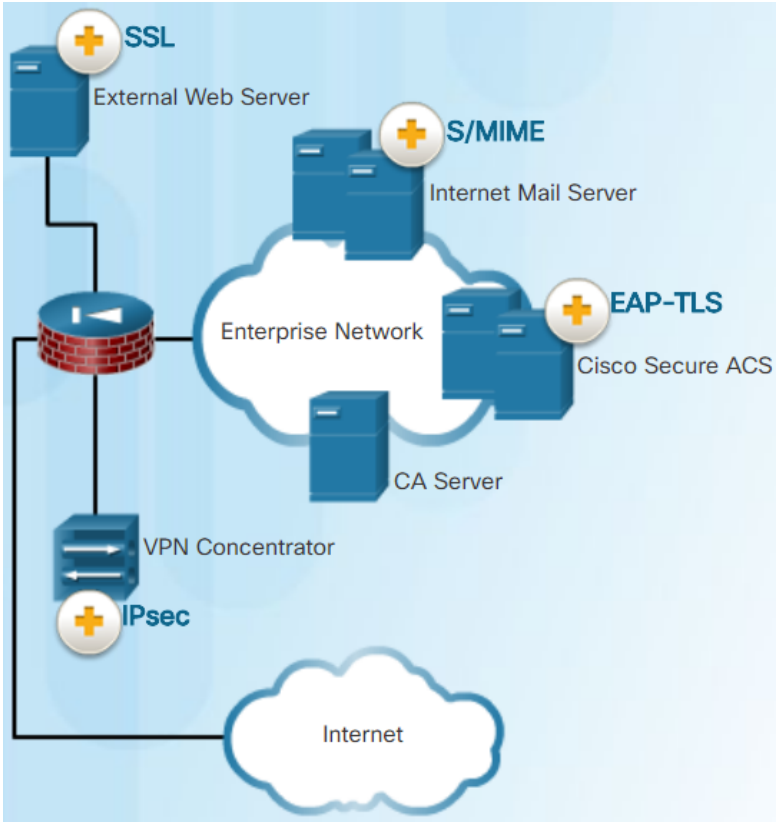
Cross-certified CA

Hierarchical CA



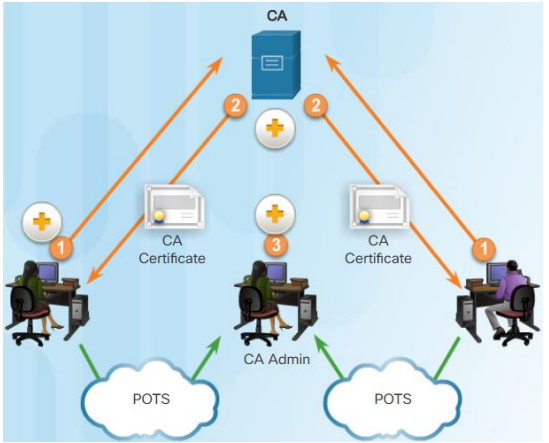
Interoperability of Different PKI Vendors

- Interoperability between a PKI and its supporting services is a concern because many CA vendors have proposed and implemented proprietary solutions instead of waiting for standards to develop.
- To address this interoperability concern, the IETF published the Internet X.509 Public Key Infrastructure Certificate Policy and Certification Practices Framework (RFC 2527).
- The X.509 version 3 (X.509v3) standard defines the format of a digital certificate.



Certificate Enrollment, Authentication, and Revocation

- All systems that leverage the PKI must have the CA's public key, called the self-signed certificate.
- The CA public key verifies all the certificates issued by the CA and is vital for the proper operation of the PKI.
- The certificate enrollment process begins when CA certificates are retrieved in-band over a network, and the authentication is done out-of-band (OOB) using the telephone.
- The system enrolling with the PKI contacts a CA to request and obtain a digital identity certificate for itself and to get the CA's self-signed certificate.
- The final stage verifies that the CA certificate was authentic and is performed using an OOB method such as the Plain Old Telephone System (POTS) to obtain the fingerprint of the valid CA identity certificate.
- A digital certificate can be revoked if key is compromised or if it is no longer needed.



Lab – Certificate Authority Stores

Networking Academy

Lab – Certificate Authority Stores

Objectives

Part 1: Certificates Trusted by Your Browser

Part 2: Checking for Man-In-Middle

Background / Scenario

As the web evolved, so did the need for security. HTTPS (where the 'S' stands for security) along with the concept of a Certificate Authority was introduced by Netscape back in 1994 and is still used today. In this lab, you will:

- List all the certificates trusted by your browser (completed on your computer)
- Use hashes to detect if your Internet connection is being intercepted (completed in the CyberOps VM)

Required Resources

- CyberOps Workstation VM
- Internet access

Part 1: Certificates Trusted by Your Browser

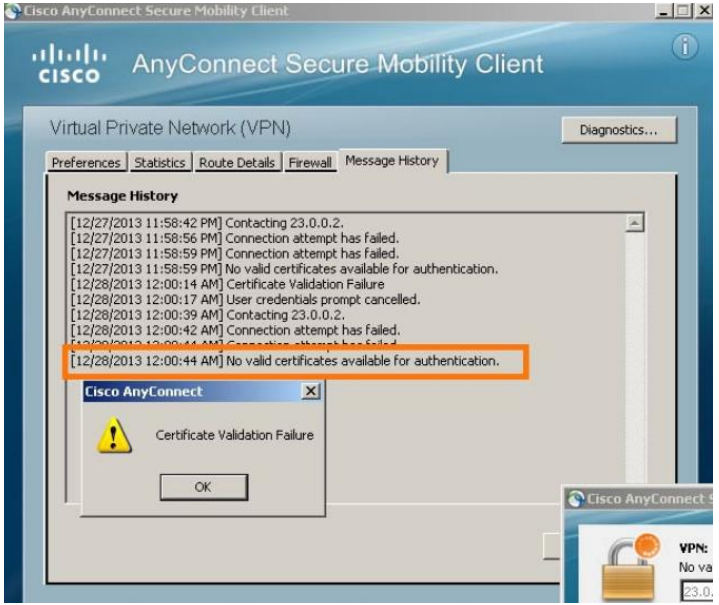
HTTPS relies on a third-party entity for validation. Known as Certification Authority (CA), this third-party entity verifies if a domain name really belongs to the organization claiming its ownership. If the verification checks, the CA creates a digitally signed certificate containing an information about the organization, including its public key.

PKI Applications

- Some of the many applications of PKIs are:
  - SSL/TLS certificate-based peer authentication
  - Secure network traffic using IPsec VPNs
  - HTTPS Web traffic
  - Control access to the network using 802.1x authentication
  - Secure email using the S/MIME protocol
  - Secure instant messaging
  - Approve and authorize applications with Code Signing
  - Protect user data with the Encryption File System (EFS)
  - Implement two-factor authentication with smart cards
  - Securing USB storage devices

Encrypting Network Transactions

- Threat actors can use SSL/TLS to introduce regulatory compliance violations, viruses, malware, data loss, and intrusion attempts in a network.
- Other SSL/TLS-related issues may be associated with validating the certificate of a web server. When this occurs, web browsers will display a security warning. PKI-related issues that are associated with security warnings include:
  - **Validity date range** - The X.509v3 certificates specify “not before” and “not after” dates. If the current date is outside the range, the web browser displays a message.
  - **Signature validation error** - If a browser cannot validate the signature on the certificate, there is no assurance that the public key in the certificate is authentic.



Encryption and Security Monitoring

- Network monitoring becomes more challenging when packets are encrypted.
- Because HTTPS introduces end-to-end encrypted HTTP traffic (via TLS/SSL), it is not as easy to peek into user traffic.
- Here is a list of some of the things that a security analyst could do:
  - Configure rules to distinguish between SSL and non-SSL traffic, HTTPS and non-HTTPS SSL traffic.
  - Enhance security through server certificate validation using CRLs and OCSP.
  - Implement antimalware protection and URL filtering of HTTPS content.
  - Deploy a Cisco SSL Appliance to decrypt SSL traffic and send it to intrusion prevention system (IPS) appliances to identify risks normally hidden by SSL.

9.3 Chapter Summary

Chapter Summary

Chapter 9: Cryptography and the Public Key Infrastructure

- Securing communications with cryptography consists of four elements:
  - Data confidentiality to guarantee that only authorized users can read the message.
  - Data integrity to guarantee that the message was not altered.
  - Origin authentication guarantees that the message is not a forgery and does actually come from whom it states.
  - Data non-repudiation to guarantee that the sender cannot repudiate, or refute, the validity of a message sent.
- Cryptology is the science of making and breaking secret codes. There are two disciplines: **Cryptography** and **Cryptanalysis.**
- A cipher is an algorithm that consists of a series of well-defined steps that can be followed as a procedure when encrypting and decrypting messages.
- A number of code breaking (cryptanalysis) methods exist, such as brute-force, ciphertext, and known-plaintext, among others.
- With modern technology, security of encryption lies in the secrecy of the keys, not the algorithm. Specifically the key length and the keyspace.

Chapter 9: Cryptography and the Public Key Infrastructure (Cont.)

- Cryptographic hashes are used to verify and ensure data integrity.
- Hash functions make it computationally infeasible for two different sets of data to come up with the same hash output.
- Mathematically, the equation ***h= H(x)*** is used to explain how a hash algorithm operates.
- Three well-known hash functions include:
  - MD5 with a 128-bit digest
  - SHA-1
  - SHA-2
- To include authentication along with message integrity, an HMAC is added to as an input to a hash function. If two parties share a secret key and use HMAC functions for authentication, a properly constructed HMAC digest of a message that a party has received indicates that the other party was the originator of the message.
- Confidentiality of the data is ensured through one of two types of encryption: symmetric and asymmetric.

Chapter 9: Cryptography and the Public Key Infrastructure (Cont.)

- Confidentiality of the data is ensured through one of two types of encryption: symmetric and asymmetric.
- Symmetric algorithms use the same pre-shared key to encrypt and decrypt data.
- Symmetric encryption algorithms are often classified as either: **Block ciphers** or **Stream Ciphers**.
- 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.
- 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 authentication objective of asymmetric algorithms is initiated with the private key encryption process. Use the formula: **Private Key (Encrypt) + Public Key (Decrypt) = Authentication.**
- Combining the two asymmetric encryption processes provides message confidentiality, authentication, and integrity.
- Diffie-Hellman (DH) is an asymmetric mathematical algorithm that allows two computers to generate an identical shared secret without having communicated before.

Chapter 9: Cryptography and the Public Key Infrastructure (Cont.)

- Digital signatures are a mathematical technique used to provide authenticity, integrity, and nonrepudiation in the form of code signing and digital certificates.
- Digital signatures are commonly used to provide assurance of the authenticity and integrity of software code.
- A digital certificate enables users, hosts, and organizations to securely exchange information over the Internet.
- The Public Key Infrastructure (PKI) is an example of a trusted third-party system referred to as certificate authority (CA).
- PKI is needed to support large-scale distribution and identification of public encryption keys.
- Many vendors provide CA servers as a managed service or as an end-user product. Organizations may also implement private PKIs using Microsoft Server or Open SSL. CAs issue certificates based on classes which determine how trusted a certificate is.
- PKIs can form different topologies of trust. The simplest is the single-root PKI topology. On larger networks, PKI CAs may be linked using two basic architectures: Cross-certified CA topologies and Hierarchical CA topologies.

Chapter 9: Cryptography and the Public Key Infrastructure (Cont.)

- Interoperability between a PKI and its supporting services is a concern because many CA vendors have proposed and implemented proprietary solutions instead of waiting for standards to develop. To address this interoperability concern, the IETF published the Internet X.509 Public Key Infrastructure Certificate Policy and Certification Practices Framework (RFC 2527). The X.509 version 3 (X.509v3) standard defines the format of a digital certificate.
- All systems that leverage the PKI must have the CA's public key, called the self-signed certificate. The CA public key verifies all the certificates issued by the CA and is vital for the proper operation of the PKI.
- There are many applications of PKIs.
- Threat actors can use SSL/TLS to introduce regulatory compliance violations, viruses, malware, data loss, and intrusion attempts in a network.
- Network monitoring becomes more challenging when packets are encrypted. Because HTTPS introduces end-to-end encrypted HTTP traffic (via TLS/SSL), it is not as easy to peek into user traffic. Here is a list of some of the things that a security analyst could do:
  - Configure rules to distinguish between SSL and non-SSL traffic, HTTPS and non-HTTPS SSL traffic.
  - Enhance security through server certificate validation using CRLs and OCSP.
  - Implement antimalware protection and URL filtering of HTTPS content.
  - Deploy a Cisco SSL Appliance to decrypt SSL traffic and send it to intrusion prevention system (IPS) appliances to identify risks normally hidden by SSL.

Chapter 9

New Terms and Commands

- 3DES (Triple DES)
- Advanced Encryption Standard (AES)
- asymmetric encryption
- block ciphers
- cipher
- cryptanalysis
- cryptography
- cryptology
- Data Encryption Standard (DES)
- Diffie-Hellman (DH)
- Digital Signature Algorithm (DSA)
- Digital Signature Standard (DSS)
- EIGamal
- Elliptical curve
- hash
- Hash Message Authentication Code (HMAC)
- Message Digest 5 (MD5)
- Public Key Infrastructure (PKI)
- Rivest ciphers (RC)
- RSA
- Secure Hash Algorithm 1 (SHA-1)
- Secure Hash Algorithm 2 (SHA-2)
- Software-Optimized Encryption Algorithm (SEAL)
- stream ciphers
- symmetric encryption

Cybersecurity Operations Certification

This chapter covers the following areas in the Cybersecurity Operations Certification:

From 210-250 SECFND - Understanding Cisco Cybersecurity Fundamentals:

- **Domain 3: Cryptography**
  - 3.1 Describe the uses of a hash algorithm
  - 3.2 Describe the uses of encryption algorithms
  - 3.3 Compare and contrast symmetric and asymmetric encryption algorithms
  - 3.4 Describe the processes of digital signature creation and verification
  - 3.5 Describe the operation of a PKI

Cybersecurity Operations Certification (Cont.)

This chapter covers the following areas in the Cybersecurity Operations Certification:

From 210-250 SECFND - Understanding Cisco Cybersecurity Fundamentals:

- **Domain 3: Cryptography**
  - 3.6 Describe the security impact of the following commonly used hash algorithms:
    - MD5
    - SHA-1
    - SHA-256
    - RSA4096
    - SHA-512

Cybersecurity Operations Certification (Cont.)

This chapter covers the following areas in the Cybersecurity Operations Certification:

From 210-250 SECFND - Understanding Cisco Cybersecurity Fundamentals:

- **Domain 3: Cryptography**
  - 3.7 Describe the security impact of the following commonly used encryption algorithms and secure communications protocols:
    - DES
    - 3DES
    - AES
    - AES256-CTR
    - RSA
    - DSA
    - SSH
    - SSL/TLS

Cybersecurity Operations Certification (Cont.)

This chapter covers the following areas in the Cybersecurity Operations Certification:

From 210-250 SECFND - Understanding Cisco Cybersecurity Fundamentals:

- **Domain 3: Cryptography**
  - 3.8 Describe how the success or failure of a cryptographic exchange impacts security investigation
  - 3.9 Describe the following in regards to SSL/TLS:
    - Cipher-suite
    - X.509 Certificates
    - Key exchange
    - Protocol version
    - PKCS

Cybersecurity Operations Certification (Cont.)

This chapter covers the following areas in the Cybersecurity Operations Certification:

From 210-250 SECFND - Understanding Cisco Cybersecurity Fundamentals:

- **Domain 5: Security Monitoring**
  - 5.3 Describe the following concepts as they relate to security monitoring:
    - Access Control List
    - NAT/PAT
    - Tunneling
    - TOR
    - Encryption
    - P2P
    - Encapsulation
    - Load Balancing


CISCO