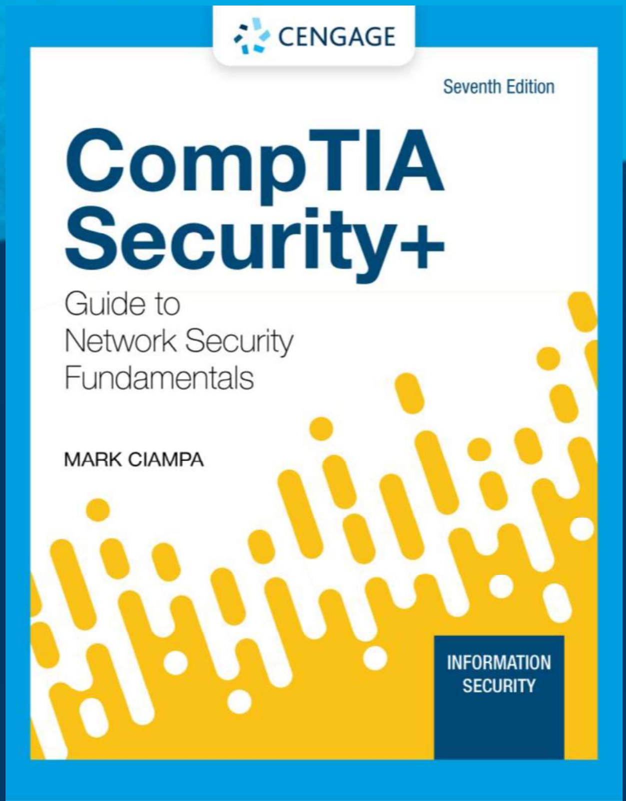

CompTIA Security+ Guide to Network Security Fundamentals, 7th Edition

Module 7: Public Key Infrastructure and Cryptographic Protocols

CENGAGE

Module Objectives

By the end of this module, you should be able to:

1. Define digital certificates
2. Describe the components of Public Key Infrastructure (PKI)
3. Describe the different cryptographic protocols
4. Explain how to implement cryptography

Digital Certificates

- Digital certificates is a common application of cryptography
- Using digital certificates involves
  - Understanding their purpose
  - Knowing how they are managed
  - Determining which type of digital certificate is appropriate for different situations

Defining Digital Certificates (1 of 2)

- A *digital signature* is used to prove a document originated from a valid sender
- Weakness of using digital signatures:
  - It can only prove that the private key of the sender was used to encrypt the digital signature
  - An imposter could post a public key under a sender's name
- *Trusted third party*
  - Used to help solve the problem of verifying identity
  - Verifies the owner and that the public key belongs to that owner
- A **digital certificate** is a technology used to associate a user's identity to a public key that has been "digitally signed" by a trusted third party

Defining Digital Certificates (2 of 2)

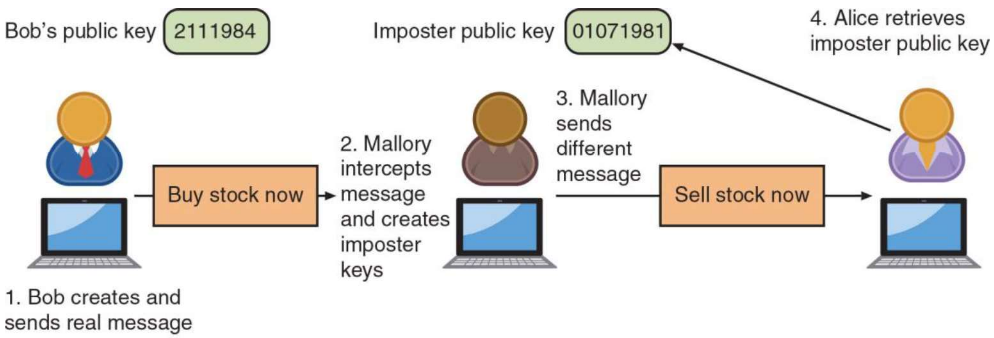

Figure 7-1 Imposter public key

Figure 7-1 Imposter public key

Managing Digital Certificates (1 of 6)

- Several entities and technologies are used to manage digital certificates:
  - **Certificate authorities (CAs)**
  - Tools for managing certificates
- Certificate Authorities
  - If a user wants a digital certificate:
    - After generating a public and private key, the user must complete a request with information such as name, address, email address, known as a **Certificate Signing Request (CSR)**
  - User electronically signs the CSR and sends it to an **intermediate CA**
  - An intermediate CA processes the CSR and verifies the authenticity of the user

Managing Digital Certificates (2 of 6)

- Certificate Authorities (continued)
  - Intermediate CAs are subordinate entities designed to handle specific CA tasks such as:
    - Processing certificate requests
    - Verifying the identity of the individual
  - The person requesting a digital certificate can be authenticated by:
    - Email, documents, in person
  - A common method to ensure security and integrity of a root CA is to keep it in an offline state from the network (**offline CA**)
  - It is only brought online (**online CA**) when needed for specific and infrequent tasks

CENGAGE

Managing Digital Certificates (3 of 6)

- Certificate Management
  - **Certificate Repository (CR)** is a publicly accessible centralized directory of digital certificates
    - It can be used to view certificate status
    - The directory can be managed locally by setting it up as a storage area connected to the CA server
  - **Certificate Revocation**
    - Reasons a certificate would be revoked
      - Certificate is no longer used
      - Details of the certificate have changed, such as user’s address
      - Private key has been lost or exposed (or suspected lost or exposed)
    - A **Certificate Revocation List** (**CRL**) is a list of digital certificates that have been revoked

Managing Digital Certificates (4 of 6)

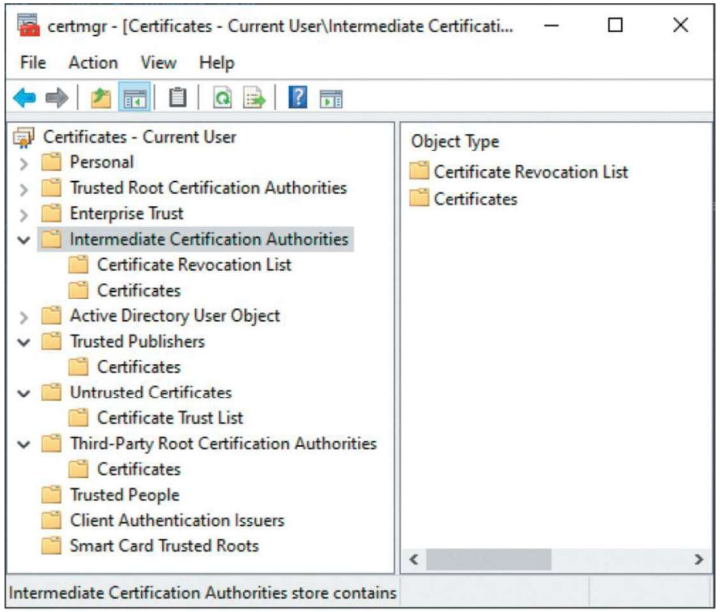

Figure 7-2 Certificate Revocation List (CRL)

Figure 7-2 Certificate Revocation List (CRL)

Managing Digital Certificates (5 of 6)

- Certificate Management (continued)
  - **Online Certificate Status Protocol (OCSP)** performs a real-time lookup of a certificate's status
  - OCSP is called a request-response protocol
  - The browser sends the certificate's information to a trusted entity known as an OCSP Responder
    - The OCSP Responder provides immediate revocation information on that certificate
  - **OCSP stapling**
    - A variation of OCSP where web servers send queries to the OCSP Responder server at regular intervals to receive a signed time-stamped response

Managing Digital Certificates (6 of 6)

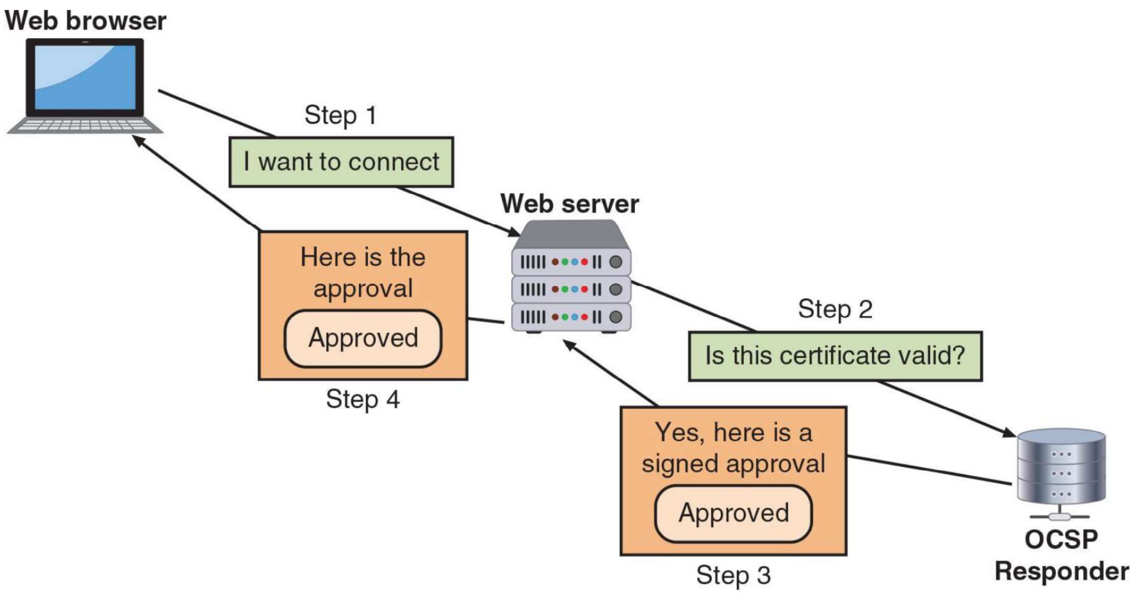

Figure 7-3 OCSP stapling

Figure 7-3 OCSP stapling

Types of Digital Certificates (1 of 6)

- The most common categories of digital certificates are:
  - Root certificates
  - Domain certificates
  - Hardware and software certificates
- Root Digital Certificates
  - The process of verifying a digital certificate is genuine depends upon **certificate chaining**
    - Links several certificates together to establish trust between all the certificates involved
    - The beginning point of the chain is known as a **root digital certificate** and is created and verified by a CA
    - They are **self-signed** and do not depend upon any higher-level authority
    - Endpoint of the chain is the **user digital certificate** itself

Types of Digital Certificates (2 of 6)

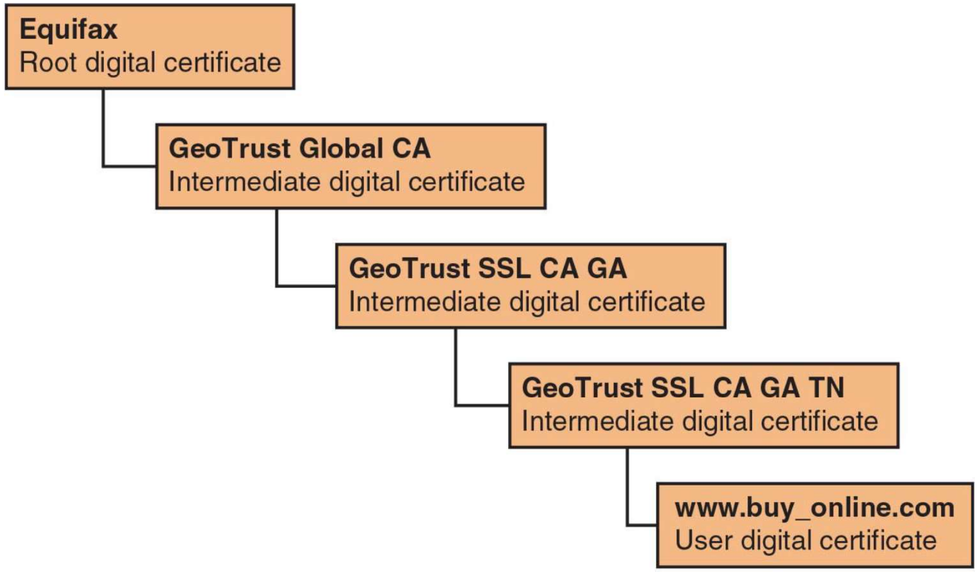

Figure 7-4 Certificate chaining

Figure 7-4 Certificate chaining

Types of Digital Certificates (3 of 6)

- Domain Digital Certificates
  - Most digital certificates are web server digital certificates issued from a web server to a client
  - Web server digital certificates perform two primary functions:
    - Ensure the authenticity of the web server to the client
    - Ensure the authenticity of the cryptographic connection to the web server
  - There are several types of domain digital certificates:
    - **Domain validation digital certificates**
    - **Extended validation (EV) digital certificates**
    - **Wildcard digital certificates**
    - **Subject alternative name (SAN) digital certificates**

Types of Digital Certificates (4 of 6)

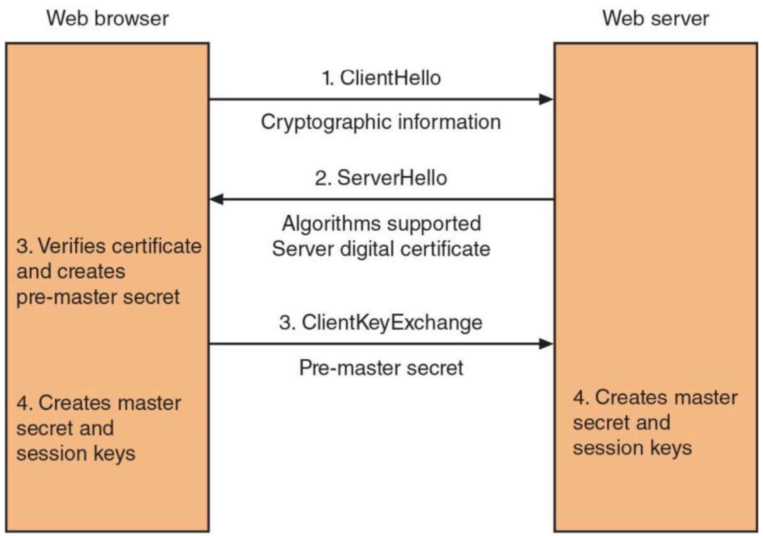

Figure 7-6 Key exchange

Figure 7-6 Key exchange

Types of Digital Certificates (5 of 6)

- Hardware and Software Digital Certificates
  - More specific digital certificates relate to hardware and software:
    - *Machine/computer digital certificate*
    - *Code signing digital certificate*
    - *Email digital certificate*
- Digital Certificate Attributes and Formats
  - The standard format for digital certificates is X.509
  - All x.509 certificates follow the standard ITU-T x.690, which specifies one of three encoding formats:
    - *Basic Encoding Rules (BER)*
    - *Canonical Encoding Rules (CER)*
    - *Distinguished Encoding Rules (DER)*

Types of Digital Certificates (6 of 6)

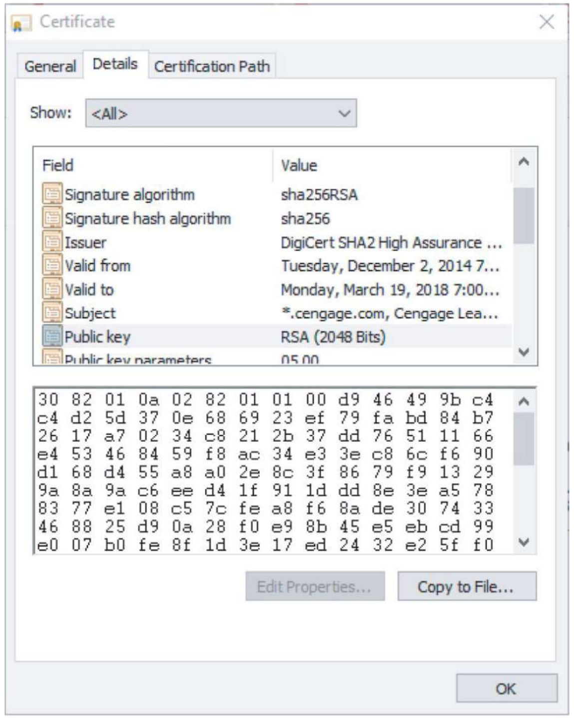

Figure 7-8 Digital certificate attributes

Figure 7-8 Digital certificate attributes

Knowledge Check Activity 1

Which of the following is the beginning point of a certificate chain?

a. User certificate
b. Intermediate certificate
c. Root certificate
d. Top-level certificate

Knowledge Check Activity 1: Answer

Which of the following is the beginning point of a certificate chain?

Answer: c. Root certificate

The beginning point of a certificate chain is the root certificate and they do not depend on a higher-level authority.

Public Key Infrastructure (PKI)

- PKI is one of the most important management tools for the use of:
  - Digital certificates:
  - Asymmetric cryptography
- It is important to understand PKI:
  - Know PKI trust models
  - How it is managed
  - Features of key management

What is Public Key Infrastructure (PKI)?

- There is a need for a consistent means to manage digital certificates
- **Public key infrastructure (PKI)** is a framework for all entities involved in digital certificates
- Certificate management actions facilitated by PKI
  - Create
  - Store
  - Distribute
  - Revoke

Trust Models (1 of 3)

- *Trust* is defined as confidence in or reliance on another person or entity
- A **trust model r**efers to the type of trust relationship that can exist between individuals and entities
- *Direct trust* is a type of trust model where one person knows the other person
- *Third-party trust* refers to a situation where two individuals trust each other because each trusts a third party
- The web of trust model is based on direct trust
  - Each user signs a digital certificate then exchanges certificates with all other users
- Three PKI trust models use a CA:
  - The hierarchical trust model, the distributed trust model, and the bridge trust model

CENGAGE

Trust Models (2 of 3)

- Hierarchical Trust Model
  - The *hierarchical trust model* assigns a single hierarchy with one master CA called *root*
  - The root signs all digital certificate authorities with a single key
  - This model can be used in an organization where one CA is responsible for only that organization’s digital certificates
  - Hierarchical trust model limitations:
    - A single CA private key may be compromised rendering all certificates worthless
    - Having a single CA who must verify and sign all digital certificates may create a significant backlog
- Distributed Trust Model
  - The *distributed trust model* has multiple CAs that sign digital certificates
  - Eliminates limitations of hierarchical trust model

Trust Models (3 of 3)

- Bridge Trust Model
  - The *bridge trust model* is similar to the distributed trust model
  - One CA acts as a *facilitator* to interconnect connect all other CAs
  - Facilitator CA does not issue digital certificates, instead it acts as hub between hierarchical and distributed trust model
  - Allows the different models to be linked

Managing PKI (1 of 2)

- Certificate Policy (CP)
  - A *certificate policy* (*CP*) is a published set of rules that govern operation of a PKI
  - The CP provides recommended baseline security requirements for the use and operation of CA, RA, and other PKI components
- Certificate Practice Statement (CPS)
  - A *certificate practice statement* is a technical document that describes in detail how the CA uses and manages certificates
  - It also covers how to register for a digital certificate, how to issue them, when to revoke them, procedural controls and key pair management

Managing PKI (2 of 2)

- Certificate Life Cycle
  - *Creation*
    - Occurs after user is positively identified
  - *Suspension*
    - May occur when employee on leave of absence
  - *Revocation*
    - Certificate no longer valid
  - *Expiration*
    - Key can no longer be used

Key Management (1 of 2)

- Key Storage
  - Public keys can be stored by embedding them within digital certificates
  - Private keys can be stored on user's local system
  - Software-based storage may expose keys to attackers
  - Alternative: storing keys in hardware
    - Smart-cards
    - Tokens
- Key Usage
  - Multiple pairs of dual keys can be created
    - One pair is used to encrypt information and the public key backed up in another location
    - Second pair would be used only for digital signatures and the public key in that pair would never be backed up

Key Management (2 of 2)

- Key Handling Procedures
  - *Escrow*
  - *Expiration*
  - *Renewal*
  - *Revocation*
  - *Recovery*
  - *Suspension*
  - *Destruction*

Knowledge Check Activity 2

Which of the following is considered a non-secure place where PKI encryption keys may be stored?

a. Smart-card

b. Token

c. In a digital certificate

d. Local system

Knowledge Check Activity 2: Answer

Which of the following is considered a non-secure place where PKI encryption keys may be stored?

Answer: d. Local system

Private keys can be stored on a user's local system but this can leave keys open to attacks due to possible vulnerabilities in the OS. Storing keys in hardware such as tokens and smart-cards is usually a more secure alternative.

Cryptographic Protocols

- The most common cryptographic transport algorithms include the following:
  - Secure Sockets Layer
  - Transport Layer Security
  - Secure Shell
  - Hypertext Transport Protocol Secure
  - S/MIME
  - Secure Real-time Transport Protocol
  - IP Security

Secure Sockets Layer (SSL)

- **Secure Sockets Layer (SSL)** is one of the most common cryptographic protocols
  - Developed by Netscape in 1994
  - The design goal was to create an encrypted data path between a client and a server
  - SSL uses the Advanced Encryption Standard (AES)
  - SSL version 3.0 is the current version

Transport Layer Security (TLS)

- **Transport Layer Security (TLS)** is a replacement for SSL
  - Versions starting with v1.1 are significantly more secure than SSL v3.0
  - Current version is TLS v1.2
  - A *cipher suite* is a named combination of the encryption, authentication, and message authentication code (MAC) algorithms that are used with SSL and TLS

CENGAGE

Secure Shell (SSH)

- **Secure Shell** (**SSH**) is an encrypted alternative to the Telnet protocol used to access remote computers
- It is a Linux/UNIX-based command interface and protocol
- SSH is a suite of three utilities: *slogin*, *ssh*, and *scp*
- Client and server ends of the connection are authenticated using a digital certificate and passwords are encrypted
- SSH can be used as a tool for secure network backups

Hypertext Transport Protocol Secure (HTTPS)

- A common use of TLS and SSL is to secure **Hypertext Transport Protocol (HTTP)** communications between a browser and Web server
- The secure version is actually "*plain*" HTTP sent over SSL or TLS and is called Hypertext Transport Protocol Secure (HTTPS)
- HTTPS uses port 443 instead of HTTP's port 80
- Users must enter URLs with https://

Secure/Multipurpose Internet Mail Extensions (S/MIME)

- **Secure/Multipurpose Internet Mail Extensions (S/MIME)** is a protocol for securing email messages
- MIME is a standard for how an electronic message will be organized, so S/MIME describes how encryption information and a digital certificate can be included as part of the message body
- S/MIME allows users to send encrypted messages that are also digitally signed

Secure Real-time Transport Protocol (SRTP)

- **Secure Real-time Transport Protocol (SRTP)** is a secure extension protecting transmission using the Real-time Transport Protocol (RTP)
- SRTP provides protection for Voice over IP (VoIP) communications
- Adds security features such as message authentication and confidentiality for VoIP Communications

IP Security (IPsec)

- **IPsec** is a protocol suite for securing Internet Protocol (IP) communications
- IPsec is considered to be a *transparent* security protocol
  - Transparent to applications, users, and software
- IPsec provides three areas of protection that correspond to three IPsec protocols:
  - *Authentication*
  - *Confidentiality*
  - *Key management*
- IPsec supports two encryption modes:
  - **Transport mode** encrypts only the data portion of each packet and leaves the header unencrypted
  - **Tunnel mode** encrypts both the header and the data portion

Weaknesses of Cryptographic Protocols

- Due to the complexity of networking, cryptographic protocols are notoriously difficult to design
- While the mathematics and related security of basic cryptographic algorithms have been extensively studied, the same cannot be said of cryptographic protocols
- Older cryptographic protocols were designed by networking experts and not by cryptographic protocol experts
- The associated security proofs to guarantee the correctness of cryptographic protocols are much more complicated than those for cryptographic algorithms

Knowledge Check Activity 3

Which encryption protocol is used for securing email messages?

a. S/MIME

b. SRTP

c. HTTPS

d. TLS

Knowledge Check Activity 3: Answer

Which encryption protocol is used for securing email messages?

Answer: a. S/MIME

Secure/Multipurpose Internet Mail Extensions (S/MIME) is used to secure email messages. SRTP provides VOIP protection, HTTPS is used, along with TLS, to secure communication between a Web browser and Web server.