

CSF3404 Cyber Security

Chapter 7

Managing Certificates

Lecturer:
Waheed Ghanem
Fakhrul Adli bin Mohd Zaki
Aalim Rozli

**Faculty of Ocean Engineering Technology and Informatics,
Universiti Malaysia Terengganu**

Managing Certificates

- Certificates enable users, servers, clients, and applications to prove their identities and validate their communications across almost any network connection.
- As a security professional, you should be able to manage all the phases of the certificate process:
 - Install a CA Hierarchy
 - Enroll Certificates
 - Secure Network Traffic by Using Certificates
 - Renew Certificates
 - Back Up and Restore Certificates and Private Keys
 - Revoke Certificates

Install a CA Hierarchy

- Certificate-based security is implemented either by obtaining certificates from a public **Certificate Authority** (CA) or by establishing your own CA.
 - If you plan to use your own CA servers to issue certificates on your network, then the **first step** in the process of setting up public key security is installing the **CA servers**.
- CA servers are used to issue certificates on the network.
- CA servers are installed in the CA hierarchy.
- The entire certificate security system will fail if the basic CA hierarchy is not properly established and authorized.
 - A security professional requires you to implement a CA design by installing CAs.

Digital Certificates

- It is an electronic document that associates credentials with a public key.
- Both users and devices can hold certificates.
- The certificate validates the certificate holder's identity and is also a way to distribute the holder's public key.
- A server called a **Certificate Authority** (CA) issues certificates and the associated public/private key pairs.



User with Certificate

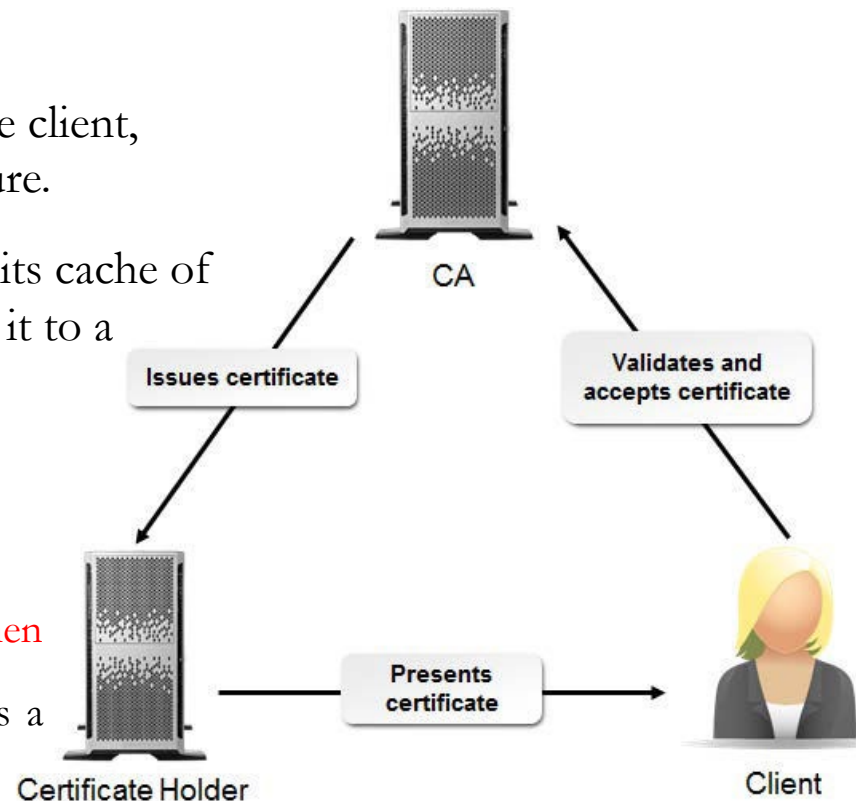


Device with Certificate

Certificate Authentication

- The user presents a digital certificate in place of a user name and password.
 - A user is authenticated if the certificate is validated by a CA.
- Certificate authentication is the process of identifying end-users in a transaction that involves a series of steps to be carried out before the user's identity is confirmed.
- These can include initiating a secure transaction, such as a client requesting access to a secure site;
 - The secure site presents its digital certificate to the client, enclosing its public key and verified digital signature.
 - The client browser validates the signature against its cache of trusted and acknowledged certificates, comparing it to a library of CAs.
- Once the digital signature is accepted,
 - Certificate authentication is successful.

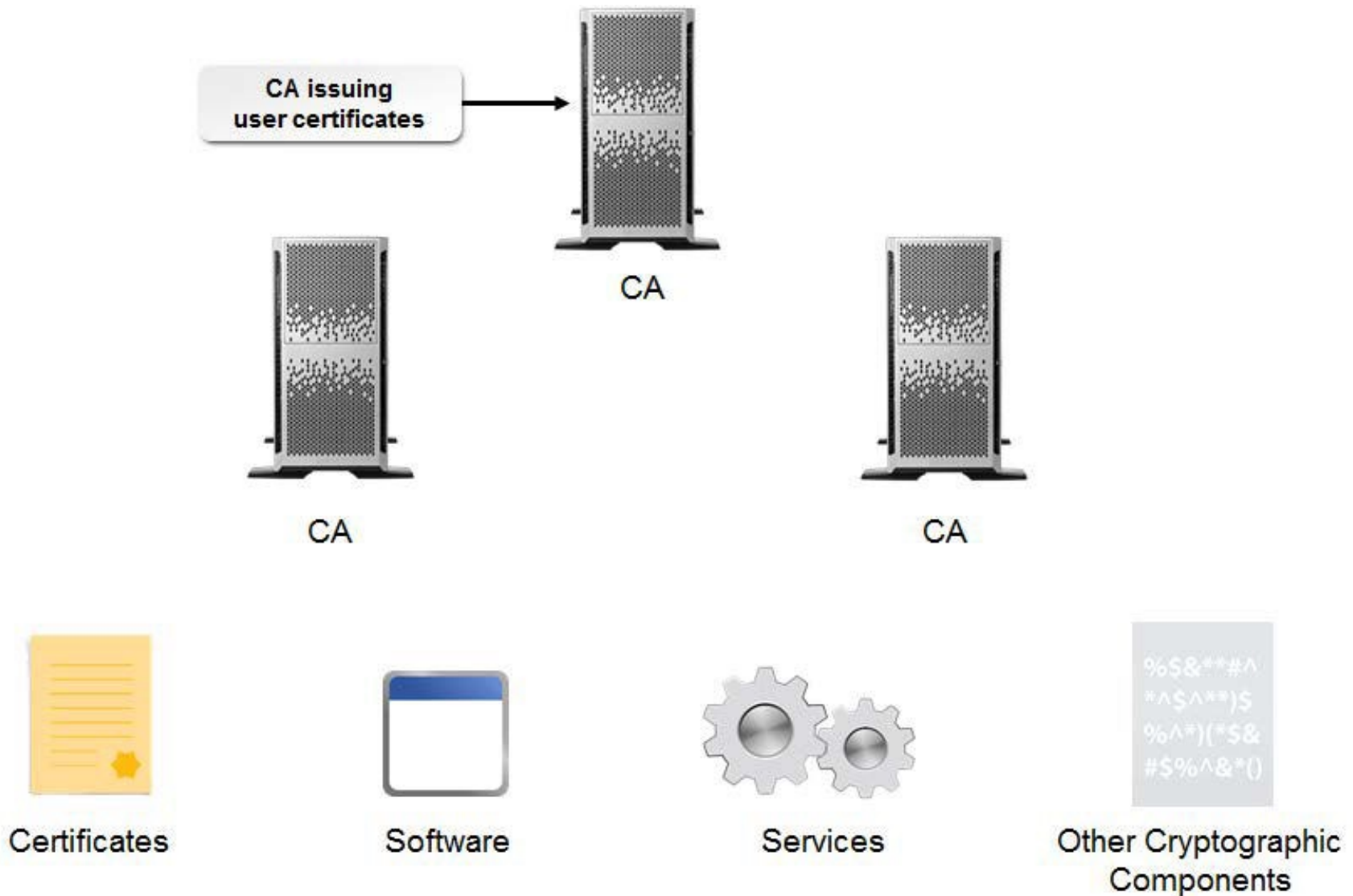
If the issuing CA does not match one in the library of CAs, then certificate authentication is unsuccessful and the user obtains a notification that the digital certificate supplied is invalid.



Public Key Infrastructure

- It is a system that is composed of a CA, certificates, software, services, and other cryptographic components, for the purpose of enabling authenticity and validation of data and entities.
- The PKI can be implemented in various hierarchical structures and can be publicly available or maintained privately by an organization.
- A PKI can be used to secure transactions over the Internet.

Public Key Infrastructure



A PKI enables data and entity authenticity and validation.

PKI Components:

A PKI contains several components:

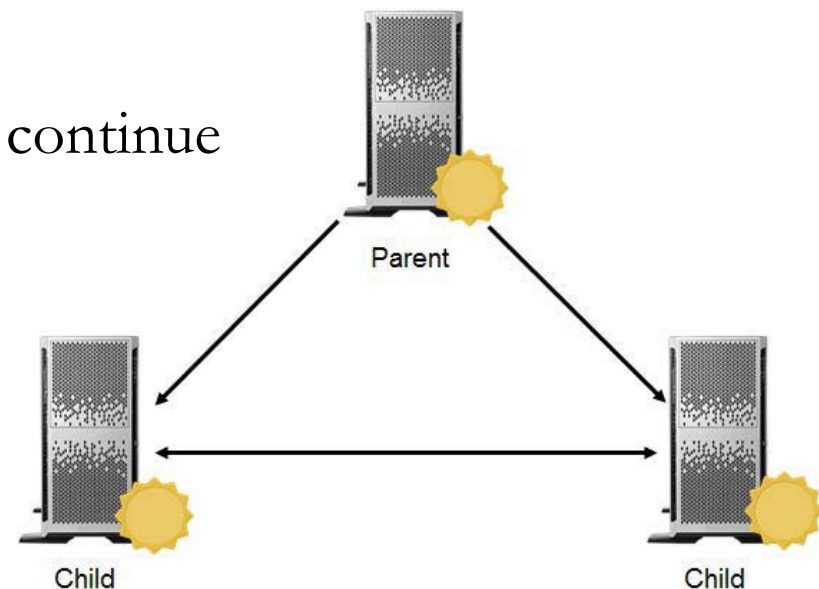
- Digital certificates, to verify the identity of entities.
- One or more **CAs**, to issue digital certificates to computers, users, or applications.
- A Registration Authority (**RA**), is responsible for verifying users' identities and approving or denying requests for digital certificates.
- A certificate repository database, to store the digital certificates.
- A certificate management system, to provide software tools to perform the day-to-day functions of the PKI.
- A certificate signing request (**CSR**), is a message sent to a **CA** in which a resource applies for a certificate.

Public Key Cryptography Standards

- It is the most common **CSR** format, developed by a consortium of vendors to send information over the Internet in a secure manner using a PKI.
- Important **PKCS** standards include:
 - .
 - PKCS #7 → Cryptographic Message Syntax Standard:
 - A PKCS that describes the general syntax used for cryptographic data, such as digital signatures.
 - PKCS #10 → Certification Request Syntax Standard:
 - A PKCS that describes the syntax used to request certification of a public key and other information.

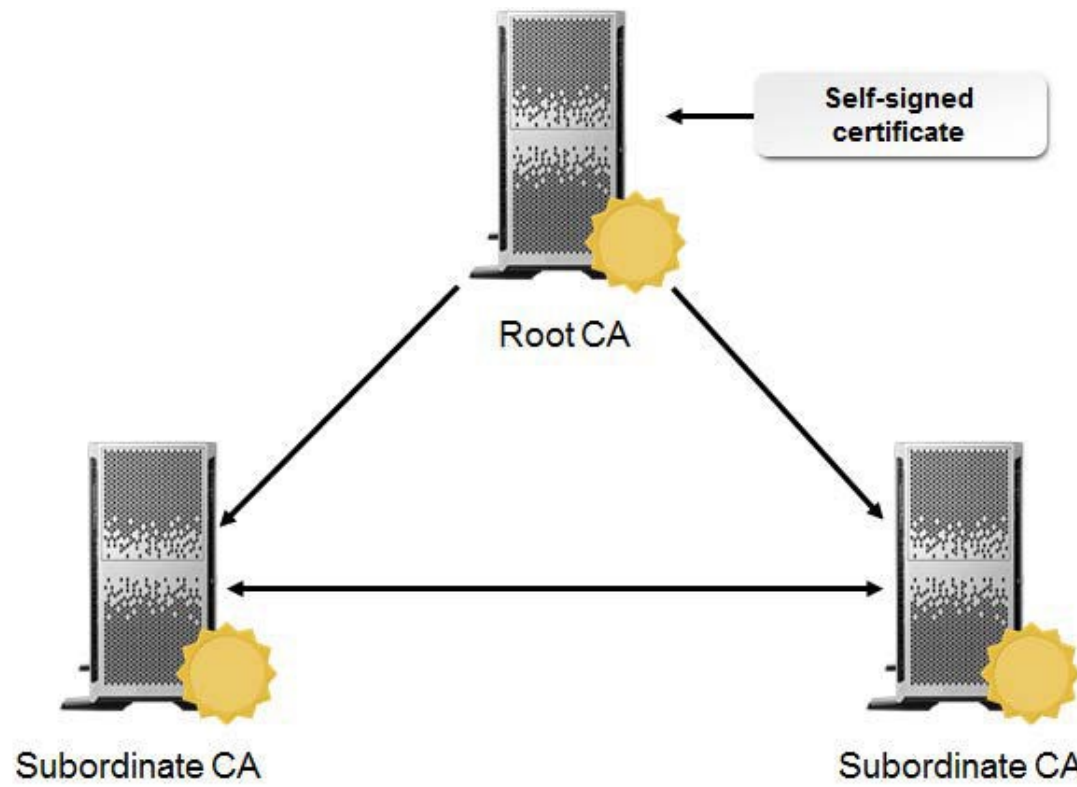
CA Hierarchies

- A CA hierarchy or **trust model** is a single CA or group of CAs that work together to issue digital certificates.
- Each CA in the hierarchy has a parent-child relationship with the CA directly above it.
- A CA hierarchy provides a way for multiple CAs to distribute the certificate workload and provide certificate services more efficiently.
- If a CA is compromised, only those certificates issued by that particular CA and its children are invalid.
- The remaining CAs in the hierarchy will continue to function.



The Root CA

- The root CA is the topmost CA in the hierarchy and, consequently, the most trusted authority.
- The root CA issues and self-signs the first certificate in the hierarchy.
- The root CA must be secured, because if it is compromised, all other certificates become invalid.



Public and Private Roots

- Root CAs can be designated as either public or private:
- A private root CA is created by a company for use primarily within the company itself.
 - The root can be set up and configured in-house or contracted to a third-party vendor.
- A public root CA is created by a third-party or commercial vendor for general access by the public.



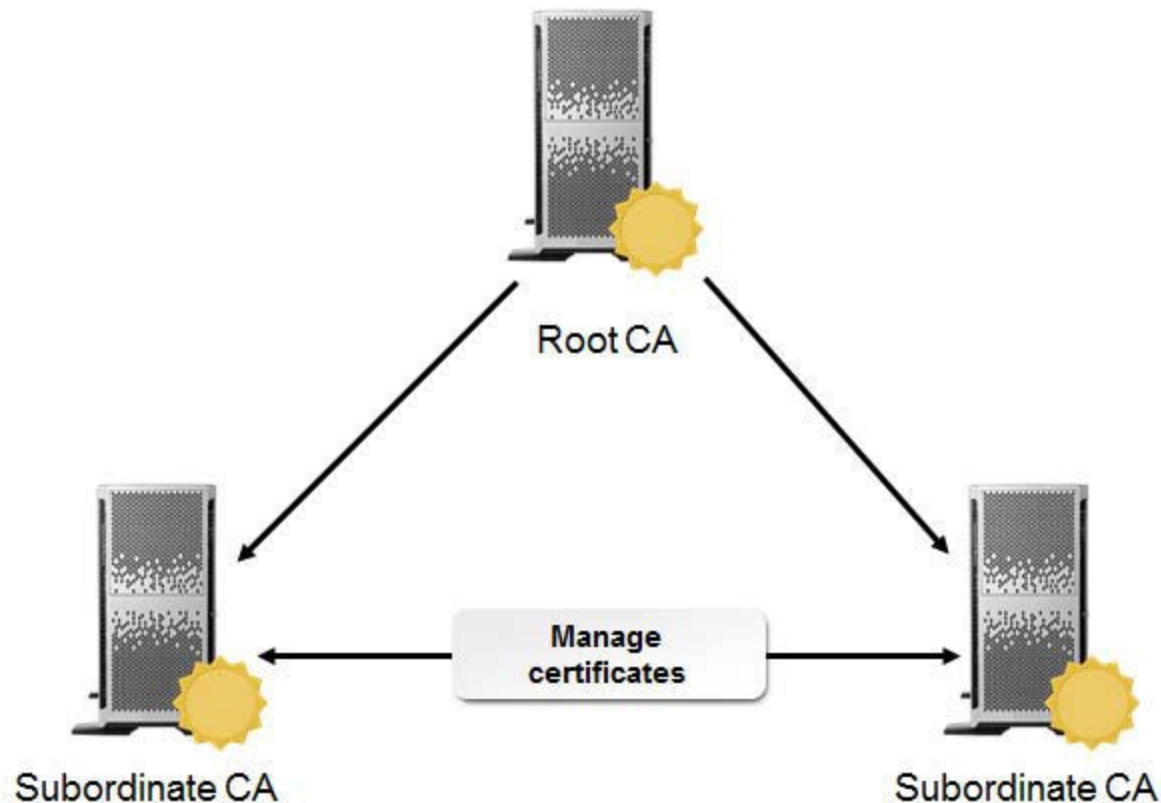
Private Root CA



Public Root CA

Subordinate CAs

- *Subordinate CAs* are any CAs below the root in the hierarchy.
- Subordinate CAs issue certificates and provide day-to-day management of the certificates, including renewal, suspension, and revocation.



Offline Root CAs

- To provide the most secure environment possible for the root CA, companies will often set up the root CA and then take it offline, allowing the subordinate CAs to issue all certificates.
- The root CA remains offline and is not patched again once it is taken offline.
- All updates are installed physically on all subordinate CAs.
- This strategy ensures that the root CA is not accessible by anyone on the network and thus, it is much less likely to be compromised.

CA Hierarchy Design Options

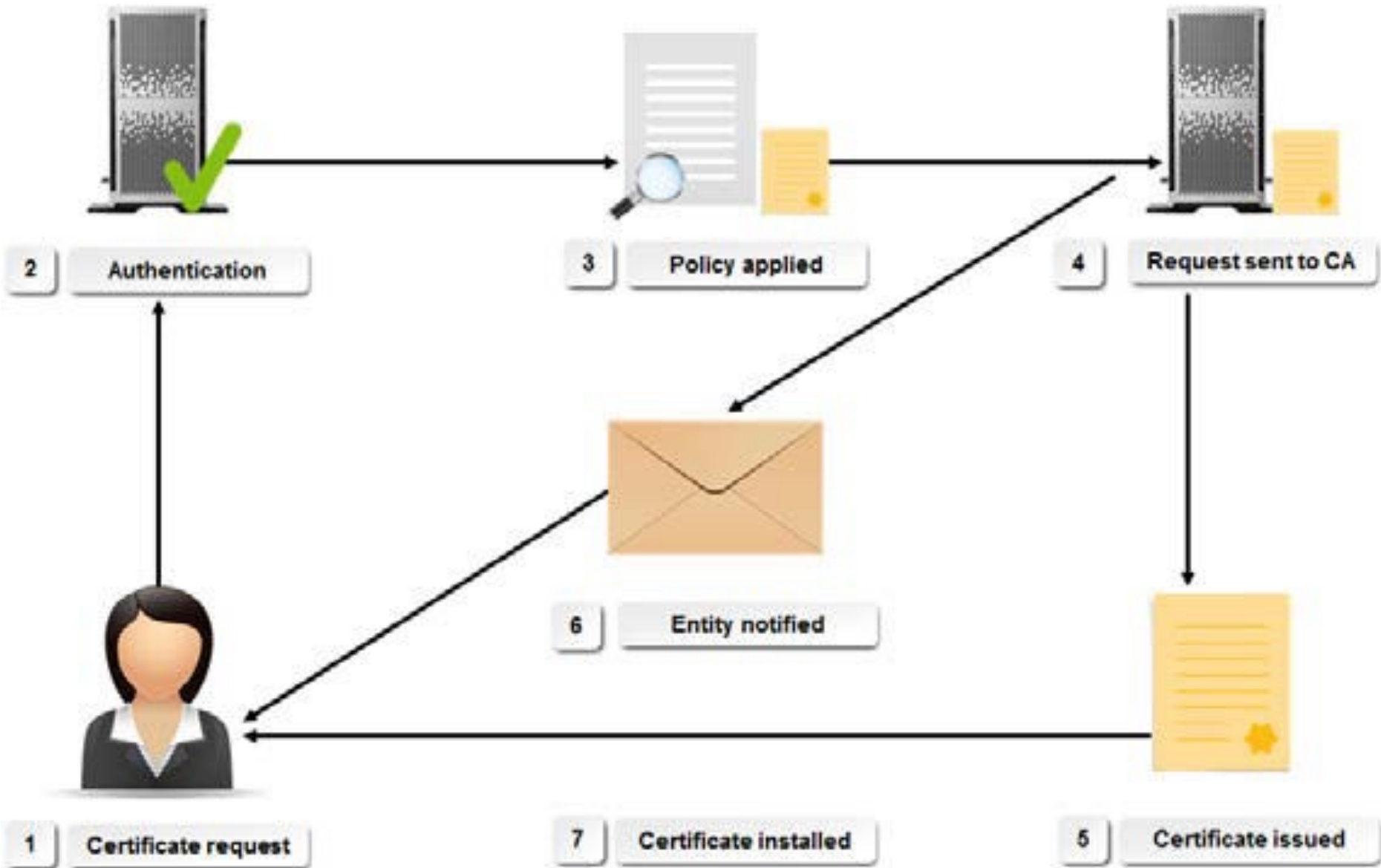
- The design of your CA hierarchy will **depend on** your organization's business and security requirements.
 - The following table describes how CA hierarchies are implemented in different company profiles.

Company Profile	CA Hierarchy Implementation
Thousands of employees worldwide	The subordinate CAs are designated by geographic location to balance the number of issued certificates among the individual CAs.
Individuals need to access specific applications only	The subordinate CAs are designated by function or department so the individual CAs serve groups of people with specific resource needs.
Tight security allows individuals to have differing levels of access to the same resources	The subordinate CAs are designated by the security required to obtain a certificate. Some CAs may be set up to issue a certificate with a network ID and password; other CAs may require a person to present a valid driver's license.

Enroll Certificates

- **Using certificates is a process that has several stages.**
 1. **Entity requests certificate:**
 - An entity follows the procedure (for example, filling out an online form) to obtain a certificate.
 2. **RA authenticates entity:**
 - Authentication is determined by the certificate policy requirements.
 3. **Policy applied to request:**
 - The RA applies the certificate policy that pertains to the particular CA that will issue the certificate.
 4. **Request sent to CA:**
 - If the identity of the entity is authenticated successfully and the policy requirements are met, the RA sends the certificate request on to the CA.
 5. **CA issues certificate:**
 - The CA creates the certificate and puts it in the repository.
 6. **Entity notified:**
 - The CA notifies the entity that the certificate is available, and the certificate is delivered.
 7. **Certificate installed:**
 - Once the certificate is obtained, it can be installed using the appropriate tool.

The certificate enrollment process



The Certificate Life Cycle

There are several main phases in the certificate life cycle:

- **Issuance** The life cycle begins when the root CA has issued its self-signed key pair. The root CA then begins issuing certificates to other CAs and end users.
- **2. Enrollment** Users and other entities obtain certificates from the CA through certificate enrollment.
- **3a. Renewal** Certificates can be renewed more than once depending on the certificate policy parameters.
- **3b. Revocation** Certificates can be revoked before their expiration date, which renders them permanently invalid. Certificates can be revoked for a variety of reasons, including misuse, loss, or compromise.
- **3c. Expiration** Certificates expire after a given length of time, which is established in the certificate policy and configured in the issuing CA. The expiration parameter is part of the certificate data. If the root CA's certificate expires, the entire CA becomes inactive.
- **3d. Suspension** Some CAs support temporary suspension of certificates, in addition to permanent revocation.

The Certificate Life Cycle



There are several main phases in the certificate life cycle.

Certificate Life Cycle Factors:

The following points show the most common factors that affect a certificate's life cycle, although this is not an exhaustive list.

1. Length of the private key.
2. Strength of the cryptography used.
3. Physical security of the CA and private key.
4. Security of issued certificates and their private keys.
5. Risk of attack.
6. User trust.
7. Administrative involvement.

Secure Network Traffic by Using Certificates

- The certificate is used to secure network traffic flowing to and from that entity.
- The certificates are used to secure network communications.
- Unsecured network communication is open to a variety of attacks and eavesdropping.
- Securing data with certificates provides another way to keep attackers out of critical components of your network.

The SSL Enrollment Process

- The certificates are used to implement the **Secure Sockets Layer** (SSL).
- The process has several steps, as explained in the following table:

1. Request	The client requests a session with the server.
2. Response	The server responds by sending its digital certificate and public key to the client.
3. Negotiation	The server and client then negotiate an encryption level.
4. Encryption	Once they agree on an encryption level, the client generates a session key, encrypts it, and sends it with the public key from the server.
5. Communication	The session key then becomes the key used in the conversation

Renew Certificates

- The first concern is to renew existing certificates at appropriate intervals. Because certificates are temporary and can expire.
- **Certificate Renewal:**
 - Certificates expire and need to be renewed.
 - The renewal process upholds security and accessibility.
 - Because if certificates did not expire, an entity on the network could use one indefinitely even if its job role or function had changed.

Back Up and Restore Certificates and Private Keys

- We need to renew the CA server certificate to keep it from expiring.
- Certificates and their associated private keys may be lost or destroyed, so we will need some way to recover them.
- Methods for backing up certificates and keys are important to restore them if they are lost or compromised

Private Key Protection Methods

- Private keys are crucial to the security of a CA hierarchy and must be protected from loss, theft, or compromise.
- **To secure a private key:**
 - Back it up to removable media and store the media securely.
 - Delete it from insecure media.
 - Require a password to restore the private key.
 - Never share a key.
 - Never transmit a key on the network or across the Internet after it is issued.
 - Consider using key escrow to store a private key with trusted third parties.

Key Escrow

- **Key escrow** is an arrangement in which the keys needed to decrypt encrypted data are held in escrow so that, under certain circumstances, an authorized **third party** may gain access to those keys.
- The **third party** is called the **key escrow agent**.
- **For example**, in certain situations, a government agency might require private keys to be placed in escrow with the agency.
- Commercial CAs can also act as escrow agents on a contract basis for organizations that do not want to back up and manage their own private keys.

M of N Control

- There are only a certain number of agents or trustees that have the authority to recover a key.
- To prevent a single authorized agent from recovering a key, the M of N scheme is commonly used.
- The M of N scheme is a **mathematical control** that takes into account the total number of key recovery agents (N) along with the number of agents required to perform a key recovery (M).
- If the number of agents attempting to recover a key does not meet or exceed M , then the key will not be recovered.
- The exact values of M and N will vary with the implementation.

Private Key Restoration Methods

- In the event that a private key is lost or damaged, you must restore the key from a backup or from escrow before you can recover any encrypted data.
- If you are using **key escrow**, the key is divided among escrow agents.
 - The agents can use the parts to reconstruct the lost key or decrypt the information directly.
- If the **key has been backed up** to removable media, it can be restored from the backup location.

The EFS Recovery Agent

- The **Encrypting File System** (EFS) uses Microsoft Windows NTFS-based public key encryption.
 - **Windows Server 2012 R2** automatically creates encryption certificates and public keys based on a user's credentials;
 - or, Windows Server 2012 R2's **Active Directory Certificate Services** (AD CS) to distribute certificates and keys.

Key Archival and Recovery

- The AD CS is used to archive private keys in the protected CA database, which enables the private keys to be recovered.
- Key recovery does not recover encrypted data or messages, but it does enable a user or administrator to recover keys that can subsequently be used for data recovery (or data decryption)

Private Key Replacement

If a private **key is lost**, you might wish to replace the key entirely after you recover any encrypted data:

1. First, recover the private key.
2. Decrypt any encrypted data.
3. Destroy the original private key.
4. Obtain a new key pair.
5. Finally, re-encrypt the data using the new private key.

Revoke Certificates

- Certificate revocation is a (**usually manual**) process in which a certificate is deemed invalid before the end of its lifecycle.
 - It can be due to any number of reasons.
 - it's an important method that lets the RADIUS know to immediately stop authenticating a certificate from then on.

Certificate Revocation

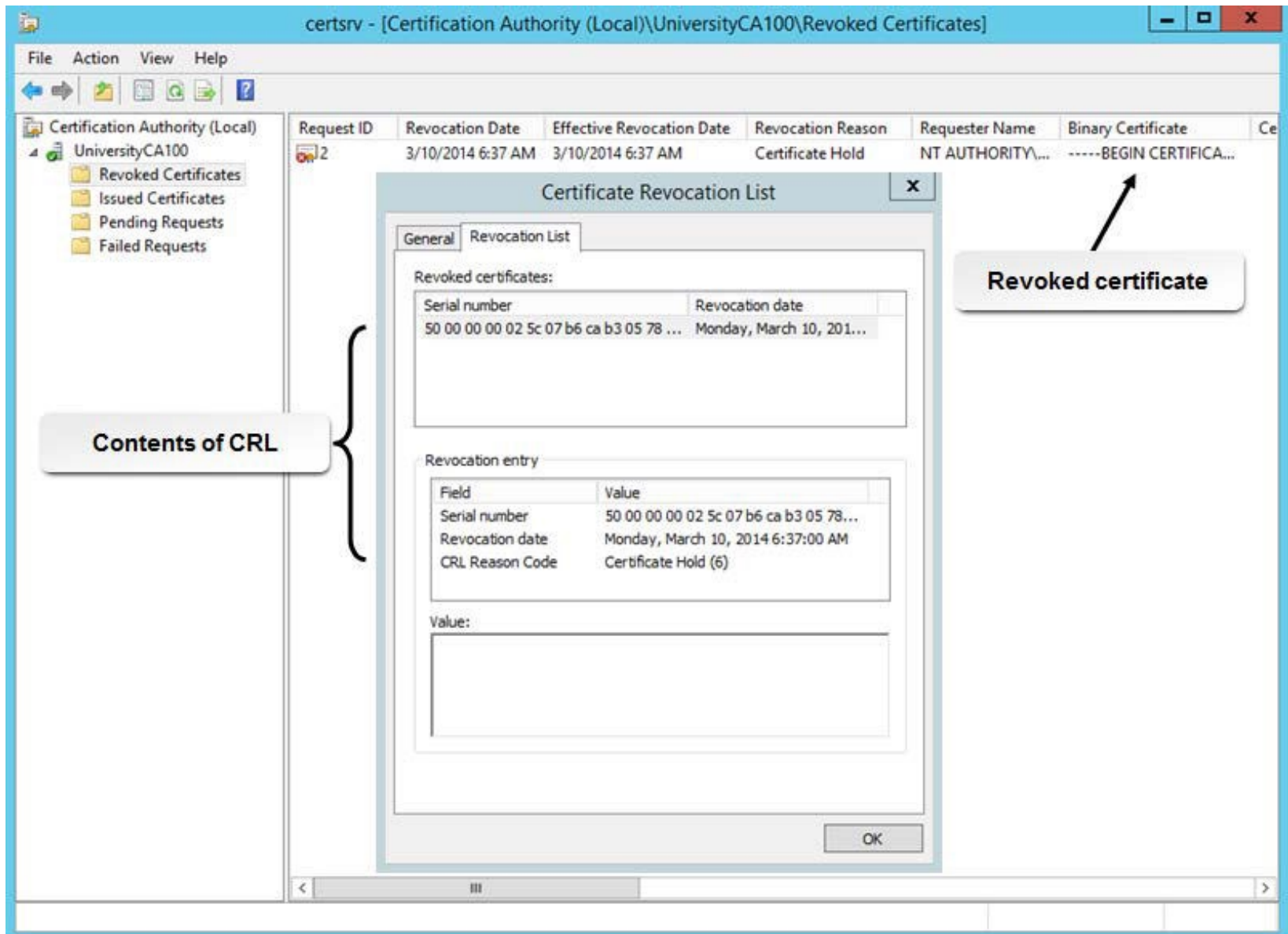
Certificates can be revoked before expiration for one of several reasons:

- The certificate owner's private key has been compromised or lost.
- The certificate was obtained by fraudulent means.
- The certificate holder is no longer trusted.
- This can occur in normal circumstances, such as when an employee leaves a company, or it can be due to a system intrusion, such as when a subordinate CA is attacked.

Certificate Revocation List

- It is a list of certificates that were revoked before the expiration date.
- Each CA has its own **CRL** that can be accessed through the directory services of the network operating system or a website.
- The **CRL** generally contains the requester's name, the request ID number, the reason why the certificate was revoked, and other pertinent information.
- **CRL Checked By Software**
 - Many software programs, such as email applications, will check the CRL for the status of a certificate before accepting it, and will reject revoked certificates.

Certificate Revocation List



The screenshot displays the 'certsrv - [Certification Authority (Local)\UniversityCA100\Revoked Certificates]' window. The left pane shows the 'Revoked Certificates' folder under 'UniversityCA100'. The right pane shows a table of revoked certificates. A callout box labeled 'Contents of CRL' points to the 'Revoked Certificates' folder. Another callout box labeled 'Revoked certificate' points to a specific entry in the CRL list.

Request ID	Revocation Date	Effective Revocation Date	Revocation Reason	Requester Name	Binary Certificate
2	3/10/2014 6:37 AM	3/10/2014 6:37 AM	Certificate Hold	NT AUTHORITY\...	-----BEGIN CERTIFICA...

Certificate Revocation List

General | Revocation List

Revoked certificates:

Serial number	Revocation date
50 00 00 00 02 5c 07 b6 ca b3 05 78 ...	Monday, March 10, 201...

Revocation entry

Field	Value
Serial number	50 00 00 00 02 5c 07 b6 ca b3 05 78...
Revocation date	Monday, March 10, 2014 6:37:00 AM
CRL Reason Code	Certificate Hold (6)

Value:

OK

- It is an HTTP-based alternative to a CRL for checking the status of revoked certificates.
- OCSP servers, also called responders, accept a request to check a specific certificate's status.
- The responder uses the certificate's serial number to search for it in the CA's database.
- The server then sends the certificate's status to the requester.

- The main advantage of using OCSP over a CRL:
 - It is that it lowers overhead.
 - OCSP responses for specific certificate requests contain less data than entire revocation lists, which can benefit both the client and the network.
 - Because OCSP does not by default encrypt these standard HTTP transmissions.
 - So, An attacker may be able to glean that a network resource used a specific certificate at a specific time during this OCSP transaction.

Reflective Questions

- What types of certificate management functions have you performed or do you plan on performing at your job?
- What method of backing up private keys would you prefer to use? Why?

Thank you