

Public Key Infrastructure (PKI)



Chapter 6

■ Infrastructure:

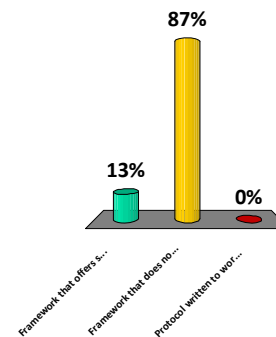
- Provides a sustaining groundwork upon which other things can be built.
- Works at a low level to provide a predictable and uniform environment that allow other high level entities to work together

Background

- Public key infrastructures (PKIs)
 - Central security foundation for organizations.
 - A framework that offers confidentiality, integrity, authentication, and non repudiation.
 - Becoming a central security foundation for managing identity credentials in many companies.
 - Allows for different types of users and entities to be able to communicate securely and in a predictable manner
 - Authenticates users for network participation and resource access.
 - Manages the sharing of trust using a third party to vouch for the trustworthiness of a claim of ownership over a credential document, called a *certificate*

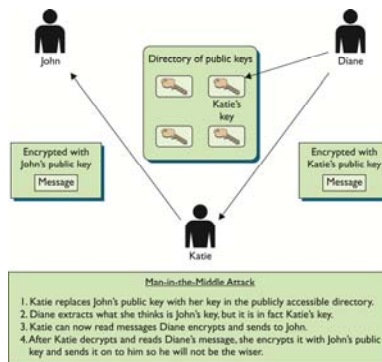
Public Key Infrastructure is a

1. Framework that offers specific technologies and algorithms that must be used
2. Framework that does not specify any technology but provides a foundation for confidentiality, integrity, and availability
3. Protocol written to work with a large subset of algorithms, applications & protocols



Can You Trust That Key?

- John and Diane want to communicate securely.
- John can generate his own public/private key pair and send his public key to Diane or place it in a directory that is available to everyone.
- Katie might have replaced John's key with her's.



Man-in-the-middle attack

Without a PKI, individuals could spoof each other's identities.

Can You Trust That Key?

- If Diane receives John's public key, either from him or from a public directory, how does she know it really came from John?
 - Perhaps an individual is masquerading as John and has replaced John's public key with their own.
- If this took place, Diane would believe that her messages could be read only by John and that the replies were actually from him.
- However, she would really be communicating with Katie.
- **What is needed:**
 - A way to verify an individual's identity, to ensure that the public key is bound to the person's identity, and thus ensure that the previous scenario does not take place.
 - Although many people may not trust John to identify himself truthfully, they could trust a reputable third party



Getting a License

- PKI environments use entities called registration authorities (RAs) and certificate authorities (CAs).
- PKIs work like the DMV.
 - You prove you who you are to the DMV by bringing the information they require.
 - If you have met the requirements, you are issued an Identification card.
 - When people ask you who you are, you show the ID from the DMV.
 - They should now believe you are who you say you are.
- PKI helps prevent a man-in-the-middle attack.



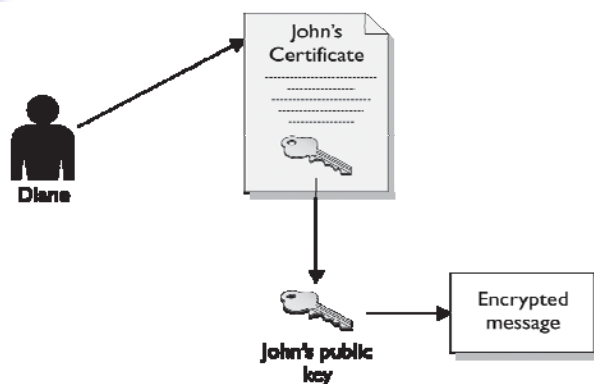
Basics of Public Key Infrastructures

- PKI is composed of several elements:
 - Certificates (including keys)
 - Certificate authorities (CA)
 - Regional authorities (RA)
 - Certificate revocation lists (CRLs)
 - Trust models
- Hardware
 - Applications
 - Policies
 - Services
 - Programming interfaces
 - Cryptographic algorithms
 - Protocols
 - Users
 - Utilities
- These components work together to allow communication using public key cryptography and symmetric keys for digital signatures, data encryption, and integrity.

Third-party trust model


- Registration authorities require proof of identity from the individual requesting a certificate and validate this information.
- Registration authority then advises the certificate authority (CA) to generate a certificate, which is analogous to a driver's license.
- The certificate authority digitally signs the certificate using its private key.
- This is commonly referred to as a **third-party trust model**.

Digital Certificates and Public Keys




1. Diane validates the certificate.
2. Diane extracts John's public key.
3. Diane uses John's public key for encryption purposes.

Public keys are components of digital certificates



Certificate Authorities (CA)

- **CAs**
 - A trusted authority that certifies identities and creates digital certificates.
 - more than just a piece of software.
 - Actually made up of the software, hardware, procedures, policies, and people who are involved in validating individuals' identities and generating the certificates.
 - If one of these components is compromised, it can negatively affect the CA overall and can threaten the integrity of the certificates it produces.
- **Digital certificates**
 - Establish an association between the subject's identity and a public key.
 - The private key is paired with the public key in the certificate and is stored separately.
 - **Free Personal email certificates**
 - www.instantssl.com/ssl-certificate-products/free-email-certificate.html



Certificate Authorities (CA)

- **Certification practices statement (CPS)**
 - [Helps to establish the trust between users and CA](#)
 - Outlines how identities are verified.
 - The steps the CA follows to generate, maintain and transmit certificates
 - How the keys are secured?
 - What data is placed within a digital certificate
 - How revocations will be handles.
 - Why the CA can be trusted to fulfill its responsibilities/
 - If a company decides to use a public CA, the CA's CPS should be reviewed
- **Certificate server**
 - Actual service that issues the certificate based on data provided during registration
 - Constructs the digital certificate and combines the user's public key with the resulting certificate.
 - The certificate is digitally signed with the CA's private key.




Registration Authority (RA)

- The registration authority
 - PKI component that accepts a request for a digital certificate.
 - Verifies the identity of the certificate requestor on behalf of the CA.
 - The CA generates the certificate using information forwarded by the RA.

- Types of certificates:
 - Class 1
 - Class 2
 - Class 3

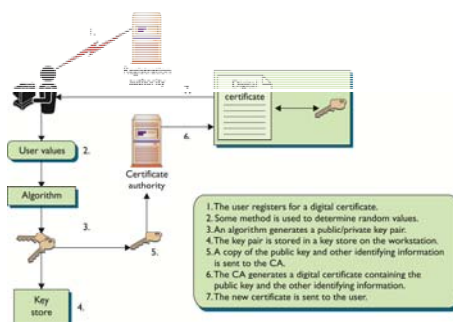
- Each CA outlines the certification classes it provides and the identification requirements that must be met to acquire each type of certificate.
- Each higher class of certificate can perform more powerful and critical tasks than the ones before it.

- 
- Types of certificates:
 - Class 1
 - Used to verify an individual's identity through e-mail.
 - Recipients can use their public/private key pair to digitally sign e-mail and encrypt message contents.

 - Class 2
 - Used for software signing by software vendors
 - Allows the receiver of the software to verify from where the software came.

 - Class 3
 - Used by a company to set up its own certificate authority.
 - The certificate authority allows performing the identification verification and generates certificates internally.

Obtaining a Digital Certificate



Steps for obtaining a digital certificate

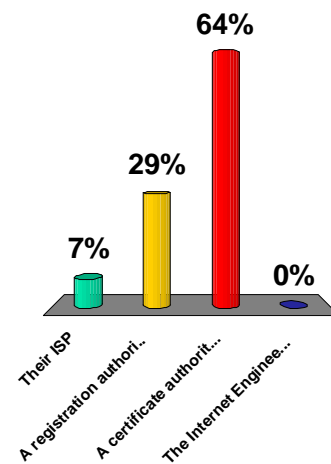
- A 1 : 1 correspondence does not necessarily exist between identities & certificates
- An entity can have multiple key pairs using, separate public keys for separate purposes
- This flexibility allows entities total discretion in how they manage their keys.

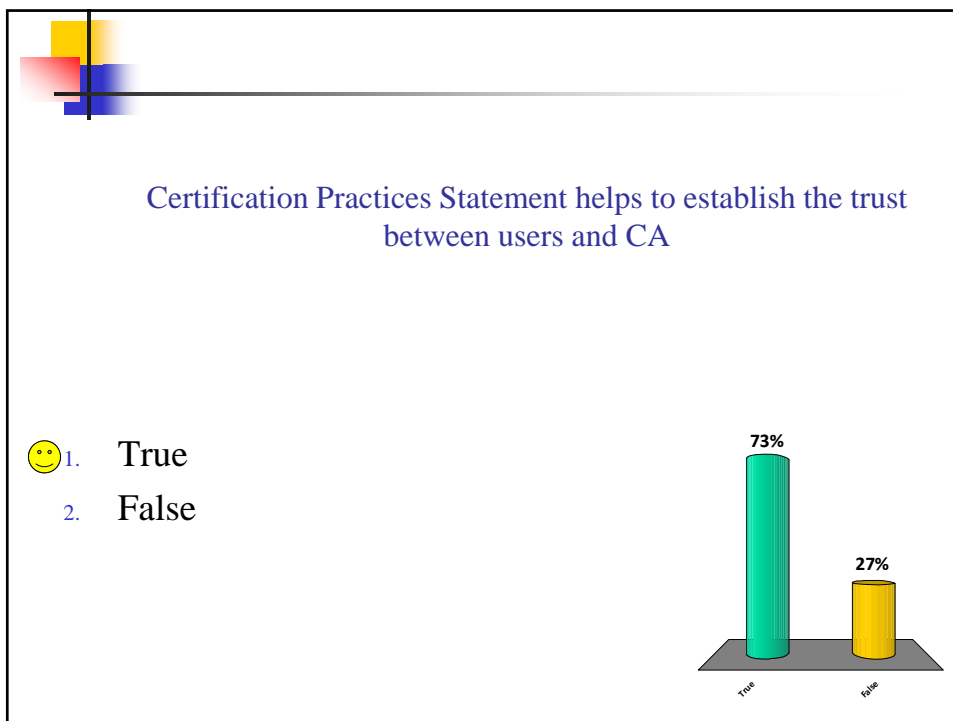
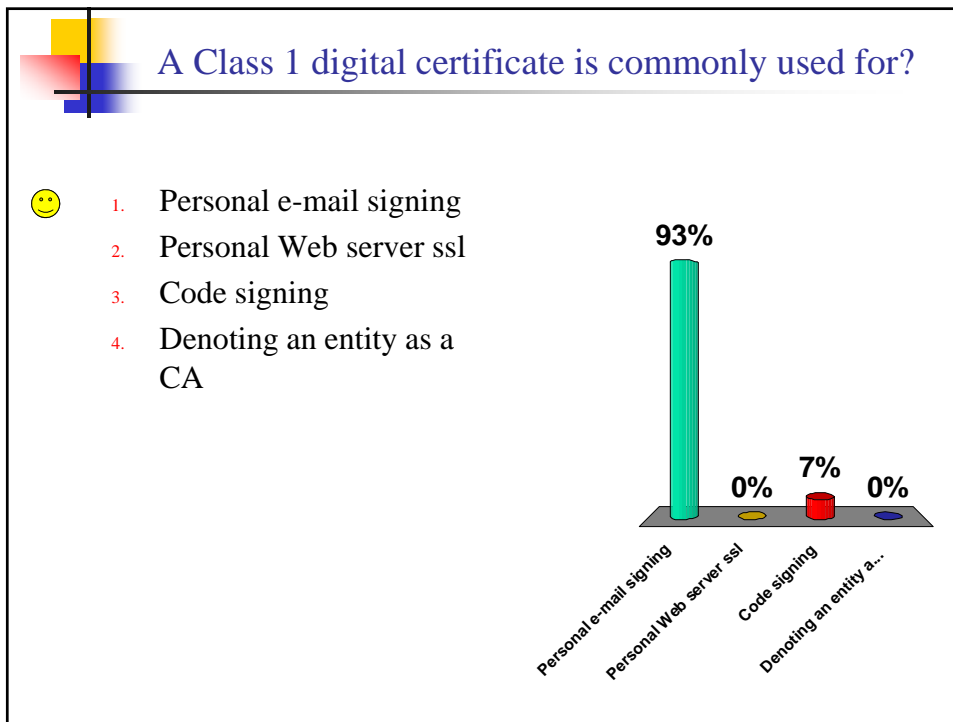
Local Registration Authorities (LRA)

- Performs the same functions as an RA.
- Instead of requiring users to communicate with a central RA, each site can have its own LRA.
 - The LRA performs identification, verification, and registration functions.
 - It then sends the request, along with the user's public key, to a centralized CA so that the certificate can be generated.
- Implemented in companies that have their own internal PKIs and have distributed sites.
- Reduces the traffic created by several users making requests across wide area network (WAN) lines.

To obtain a certificate, a user must contact whom?

1. Their ISP
- 😊 2. A registration authority (RA)
3. A certificate authority (CA)
4. The Internet Engineering Task force, via a X.509 request



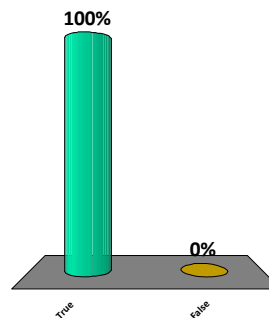


Certificate Repositories

- Certificate Repository
 - A centralized directory that can be accessed by others.
 - Holding place for individuals' certificate and public keys.
 - Stores the public key and their corresponding certificate, once it is registered, identity is proven, and a key is pair generated.
 - They must be available to whoever requires them to communicate within a PKI environment.
 - Can be accessed and searched using the Lightweight Directory Access Protocol (LDAP).
 - Security requirements not as high as needed for actual CA

A certificate repository is a holding place for individual certificates and public keys that are participating in a particular PKI environment

1. True
2. False

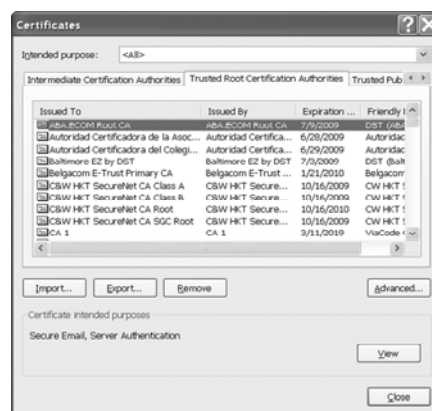
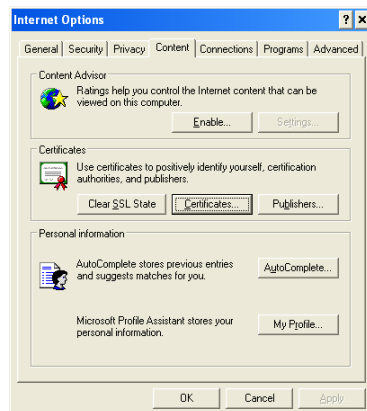



Trust and Certificate Verification

- Use PKI if you do not automatically trust individuals you do not know.
- A third party that is trusted by both the first and second party is needed.
- When a user chooses to trust the CA, the user will download CA's digital certificate and public key., which will then be stored on local computer
- Most browsers have a list of CAs configured to be trusted by default
 - The user can add / remove from this list as needed.
- Certificate authorities you trust can be found in your browser's list.

Trust and Certificate Verification

- The Microsoft CAPI environment can add and remove CAs from this list as needed.
- Tools > Internet Options > Certificates





Validating a Certificate

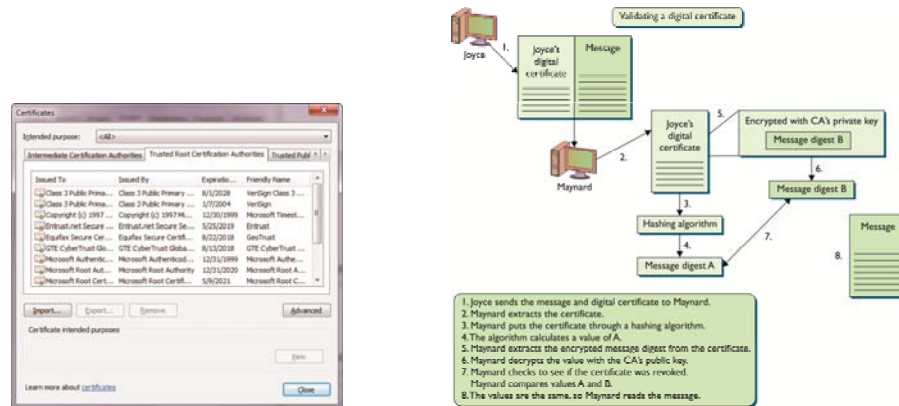
1. Suppose Angelina receives a digitally signed message from Brad who she does not know.
 1. Brad also included her digital certificate with her message which has her public key embedded within it
2. Before Angelina can ensure the authenticity of the message, she has to do some work
3. Angelina compare the CA that digitally signed the certificate to a list of CAs that have already been loaded into the receiver's computer.
 1. If CA is not in the list, she would not accept
4. CA is in the list, so she now needs to verify that the certificate has not been altered.
 1. Using the CA's public key and the digest of the certificate, she verifies the integrity of the certificate
 2. The use of digital signatures allow certificates to be saved in public directories without the concern of being accidently or intentionally altered



Validating a Certificate (Contd.)

5. Review the validity dates.
6. Check a revocation list to see if the certificate has been revoked.
7. Angelina now trusts that this certificate is legitimate and belongs to Brad
8. The certificate holds Brad's public key
9. She calculates a message digest using Brad's public key and matches the value with the value stored in the certificate.
10. It assures that the message has not been altered during transmission. Thus, she has confidence in integrity of the message
11. How does she know that the message is indeed sent by Brad
 1. Because she can decrypt the digital signature using Brad's public key, it indicates that only the associated private key could be used
 2. Can someone create an identical key pair? Yes, but impractical
12. After all this, Angelina reads her message, which says:
Hi Angelina,

Steps for Verifying a Certificate

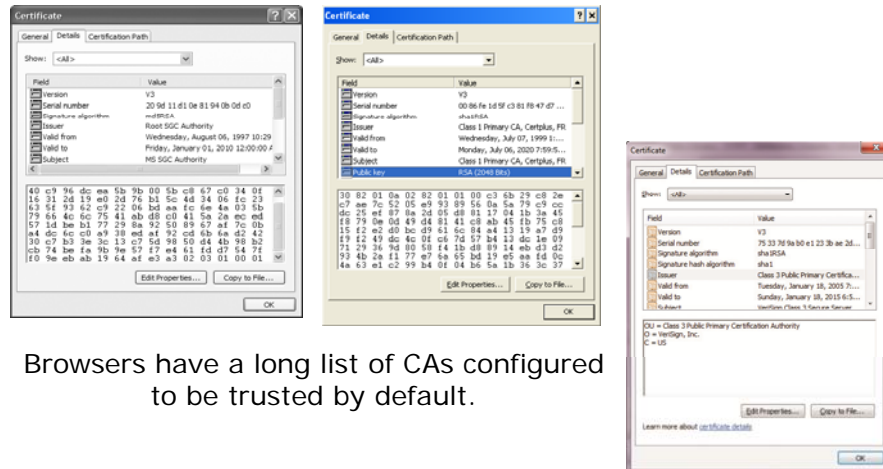


The Browser's List of CAs

Digital Certificates

- A digital certificate **binds an individual's identity to a public key**.
 - Contains all information a receiver needs to be assured of the identity of the public key owner.
- The certificates are created and formatted based on the X.509 standard.
 - International Telecommunication Union (www.itu.int).
 - It outlines the necessary fields and values of a certificate.
 - As of this writing, version 3 is the most current

Samples



Browsers have a long list of CAs configured to be trusted by default.

Fields Within a Digital Certificate

Tools > Internet options > Certificates > Certificate Intended purpose > View

Standard Fields

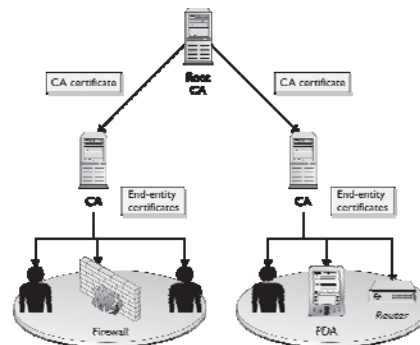
- Version Number
 - Identifies the version of the X.509 standard that was followed to create the certificate.
 - The version number indicates the format and fields that can be used.
- Subject
 - Specifies the owner of the certificate and can be a network device (router, Web server, firewall, and so on), an application, a department, a company, or a person.
- Public key
 - Contains the public key being bound to the certified subject, which also identifies the algorithm that was used to create the private/public key pair.
- Issuer
 - Identifies the CA that generated and digitally signed the certificate.

Standard Fields

- Serial number
 - Contains a unique number identifying a specific certificate issued by a particular CA.
- Validity
 - Specifies the dates through which the certificate is valid for use.
- Certificate usage
 - Specifies the approved use of a certificate, which dictates the use of this public key.
- Signature algorithm
 - Identifies the hashing algorithm and the digital signature algorithm used to digitally sign the certificate.
- Extensions
 - Allow additional data to be encoded into the certificate to expand the functionality of the certificate.

Certificate Attributes

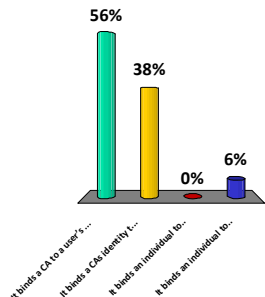
- **The four main types of certificates are:**
 - End-entity certificates
 - CA certificates
 - Cross-certification certificates
 - Policy certificates



End-entity and CA certificates

What is the purpose of a digital certificate

1. It binds a CA to a user's identity
2. It binds a CA's identity to the correct RA
3. It binds an individual to a RA
4. It binds an individual to a public key



Purpose	Percentage
It binds a CA to a user's...	56%
It binds a CA's identity to...	38%
It binds an individual to...	0%
It binds an individual to...	6%

Types of Certificates

- **End-entity certificates**
 - End-entity certificates are issued by a CA to a specific subject such as Joyce, the accounting department, or a firewall.

- **CA certificates**
 - A CA certificate may be self-signed in case of a stand-alone or root CA, or it may be issued by a superior CA within a hierarchical model.
 - The superior CA gives the authority and allows the subordinate CA to accept certificate requests and generate the individual certificates.

Types of Certificates

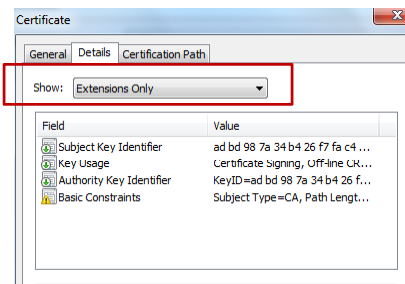
- **Cross-certification certificates**
 - Cross-certificates, or cross-certification certificates, are used when independent CAs establish peer-to-peer trust relationships.
 - They are a mechanism through which one CA can issue a certificate allowing its users to trust another CA.

- **Policy certificates**
 - Within sophisticated CAs used for high-security applications, a mechanism is required to provide centrally controlled policy information to PKI clients. This is often done by placing the policy information in a policy certificate.

Certificate Extensions

- Allow for further information to be inserted within the certificate
 - Can be used to provide more functionality in a PKI implementation
 - Can be standard or private
- Standard certificate extensions are implemented for every PKI implementation.

- Private certificate extensions
 - Defined for specific organizations; allows companies to further define uses for digital certificates to best fit their business needs





Critical and Non-Critical Extensions

- Certificate extensions are considered either critical or non-critical indicated by a specific flag within the certificate.
 - Flag setting- Critical,
 - The extension **must** be understood and processed by the receiver.
 - If the receiver is not configured to understand a particular extension marked as critical, and thus cannot process it properly, the certificate cannot be used for its proposed purpose.
 - Flag setting- Non Critical
 - If the flag does not indicate that the extension is critical, then the certificate can be used for the intended purpose, even if the receiver does not process the appended extension.



Certificate Lifecycles

- Keys and certificates should **have lifetime settings** that force the user to register for a new certificate after a certain amount of time.
- Determining the proper length of these lifetimes:
 - Shorter lifetimes limit the ability of attackers to crack them.
 - Longer lifetimes lower system overhead.
 - More-sophisticated PKI implementations perform automated and transparent key updates to avoid having users register for new certificates when old ones expire.
- Certificate management involves
 - Registration, certificate and key generation, renewal, and revocation. Additional management functions include CRL distribution, certificate suspension, and key destruction.

Setting certificate lifetimes way into the future and using them for long periods of time provides attackers with extended windows to attack the cryptography.



Registration and Generation

- Key pair can be generated
 - Locally, by an application and stored on a local key store on the user's workstation
 - Remotely, by a central key-generation server. Keys would then have to be securely transmitted.

- **Proof of possession**
 - The act of verifying that an individual indeed has the corresponding private key for a given public key
 - If a key pair is used for encryption, the RA can send a challenge value to the individual.
 - That person will use the private key to encrypt that value and return it to the RA.
 - If the RA can successfully decrypt this value with the public key, the keys are registered



Renewal

- The certificate itself has its own lifetime, which can be different from the key pair's lifetime.
 - The certificate's lifetime is specified by the validity dates inserted into the digital certificate.
 - The certificate cannot be used before the start date, nor after the end date.
- **Renewal**
 - If the certificate has not been revoked, the original keys and certificate are used to provide authentication for renewal.
- **New certificate**
 - If the certificate just expired a new certificate can be generated with new validity dates.
 - If the functionality needs to be expanded or restricted, a new certificate is generated.



Revocation

- A certificate can be revoked when its validity needs to be ended before its actual expiration date is met.
- Done when the private key has been compromised or the holder of the certificate is no longer with the organization.
- Once revoked, a certificate cannot be reinstated.
- The CA provides this type of protection by maintaining a certificate revocation list (CRL):
 - A list of serial numbers of certificates that have been revoked
 - Also contains a statement indicating why the individual certificates were revoked and a date when the revocation took place.



CA and the Certificate revocation list

- The CA:
 - Is responsible for the status of the certificates it generates.
 - Must be informed of a revocation.
 - Must provide this information to others.
 - Is responsible for maintaining the revocation list and posting it in a publicly available directory.



Acting on a Revocation

- When a revocation request is submitted, the individual submitting the request must be authenticated.
 - The authentication can involve a password that was agreed upon and was created during the registration process.

- Authentication should not be based on the individual proving to have the corresponding private key - it may have been stolen.
 - The CA would be authenticating an imposter.
 - Therefore, involve an agreed upon password created during the registration process.



Protect the certificate revocation list

- The CRL's integrity needs to be protected to ensure that attackers cannot modify data pertaining to a revoked certification from the list.
 - The integrity of the list also needs to be protected to ensure that bogus data is not added to it.
 - The only entity that should be able to modify any information on the CRL is the CA.

- The mechanism used to protect the integrity of a CRL is a digital signature.
 - The CA's revocation service creates a digital signature for the CRL.



CRL Distribution

- CRL files may be requested by individuals who want to verify and validate a newly received certificate.
 - The files can be periodically pushed down to all users participating within a specific PKI.
 - It is also possible to push down the full CRL first.
 - The following CRLs pushed down to the users are delta CRLs, meaning that they only contain the changes to the original or base CRL.
- In implementations where the CRLs are not pushed down to individual systems,
 - The users' PKI software needs to know where to look for the posted CRL that relates to the certificate it is trying to validate.



Certificate Validation

- Certificate Validation:
 - Users refer to the directory where the CRL is posted.
 - Download the list, and verify the CA's digital signature to ensure that the proper authority has signed the list and that the list was not modified in an unauthorized manner.
 - Looks through the list to see if the serial number of the certificate they are trying to validate is listed.
 - If the serial number is on the list, the private key should no longer be trusted, and the public key should no longer be used.



Suspension

- Instead of being revoked, a certificate is sometimes suspended, meaning it is temporarily put on hold.
 - If, for example, an employee is taking an extended vacation and wants to ensure that the certificate will not be compromised or used during that time, a suspension request can be made to the CA.
 - The CRL would list this certificate and its serial number, and in the field that describes why a certificate was revoked, it would instead indicate a hold state.
 - Once the employee returns to work, a request can be made to the CA to remove the certificate from the list.



Key Destruction

- Key pairs and certificates have set *lifetimes*.
- It is important that the certificates and keys are properly destroyed wherever the keys are stored (on users' workstations, centralized key servers, USB token devices, smart cards, and so on).
- Prevents potential malicious activity:
 - An attacker might use the key to digitally sign or encrypt a message with the hopes of tricking someone else about his identity.
 - Might try to brute force attack the cryptosystem.



Centralized or Decentralized Infrastructures

- Keys generation in companies - centralized v/s decentralized manner.

- **Centralized infrastructure** - Keys are generated and stored on a central server, and keys are transmitted to individual systems as needed.
 - Workstations may not have processing power to produce keys
 - Easier backups and recovery procedures

- **Decentralized infrastructure** - Software on individual computers generates and stores cryptographic keys.
 - Avoids the difficulty of secure key distribution
 - Avoids single point of failure
 - Better to generate end-user keys on a local machine to eliminate doubt about who did the work and “owns” the keys



Centralized Drawbacks

- Keys generated on a server must be securely transmitted to the clients.
- The server that stores the keys needs to be available and provide single point of failure.
- It must have fault tolerance or redundancy mechanism.
- All keys are in one place, which is a prime target for an attacker.
- If the central key server is compromised, the whole environment is compromised.
- Some applications create their own public/private key pairs and do not allow other keys to be imported and used.

- **If a public/private key pair is being generated for digital signatures, and if the company wants to ensure that it can be used to provide true authenticity and nonrepudiation, the keys should not be generated at a centralized server.**



Private Key Protection

- A crucial component of any PKI implementation.
- Private key needs to stay private
- Backbone of authenticity and nonrepudiation claims / proof



Private Key Protection

- The key size should provide the necessary level of protection for the environment.
- The lifetime of the key should correspond with how often it is used and sensitivity level of data.
 - The key should be changed at end of lifetime.
- Key should be properly destroyed at end of lifetime.
- The key should never be exposed in clear text.
- No copies of the private key should be made.
- The key should not be shared.
- The key should be stored securely.
- Authentication should be required before the key can be used.
- The key should be transported securely.
 - Software implementations that store and use the key should be evaluated to ensure they provide the necessary level of protection.



Key Recovery

- **Key archiving:**
 - A way of backing up keys and securely storing them in a repository.
- **Key recovery**
 - The process of restoring lost keys to the users or the company.
- Dual control is when two people have to be present to carry out a specific task.
 - Can be used as part of a system to back up and archive data encryption keys
 - Can be configured to require multiple individuals to be involved in any key recovery process



Multiple Personnel to Recover

- PKI systems can be configured to allow multiple individuals to be involved in any key recovery process.
- When a key recovery is required, at least two people are required to authenticate by the key recovery software before the recovery procedure is performed.
 - Enforces separation of duties.
 - **The process that requires two individuals to recover a lost key is called dual control.**
- All key recovery procedures should be audited.
 - The audit logs should capture at least what keys were recovered, who was involved in the process, and the time and date.



M of *N* Authentication

- n number of people can be involved in the key recovery process, but at least m (which is a smaller number than n) *must* be involved before the task can be completed.
- The goal is to minimize fraudulent or improper use of access and permissions.
- This form of secret splitting can increase security by requiring multiple people to perform a specific function.



Key Escrow

- Key escrow
 - Key recovery process that allows recovering lost keys.
 - A process of giving keys to a third party so that they can decrypt and read sensitive information when required.
 - Almost always pertains to handing over encryption keys to the government so that they can use them to collect evidence during investigations.



Public Certificate Authorities

- Public Certificate Authorities
 - Specialize in verifying individual identities and creating and maintaining their certificates
 - Issue certificates that are not bound to specific companies or departments
 - An individual or company may decide to rely on a CA that is already established and being used by many other individuals and companies. This would be a public CA.
- Advantage of using a public
 - They are easily accessible to many people since most Web browsers have a list of public CAs installed and configured by default, along with their corresponding root certificates.
- Examples: VeriSign, Entrust, and Go Daddy.



Private CA

- An organization can act as a private CA (in-house) when it decides to establish a CA for internal use.
 - This type of CA can be used to create certificates for internal employees, devices, applications, partners, and customers.
 - Gives more control over the certificate registration and generation process and allows them to configure items specifically for their own needs.
 - Allows the company to control how individuals are identified, what certification classifications are created, who can and cannot have access to the CA, and how the certifications can be used.
 - If the CA will be used over an extended period, this can be a cost-effective method of generating and using certificates than having to purchase them through a public CA.
 - In some situations, it is better for a company to use a public CA, since public CAs already have the necessary equipment, skills, and technologies.
 - While in other situations, companies may feel it is a better business decision to take on these efforts themselves.

Choosing Between a Public CA and an In-House CA

- Each company is unique, and many factors must be taken into consideration.
 - It is not just a financial decision.
 - The scale of the PKI within the organization.
 - How integrated it will be with different business needs and goals.
 - Its interoperability with a company's current technologies.
 - The number of individuals who will be participating in the decision-making process.
 - How it works with outside entities.
- Using public CAs
 - Public CAs already have the necessary equipment, skills, and technologies.
- Using in-house CAs
 - Some companies do not trust an outside authority to generate and maintain their company's certificates.

Outsourced Certificate Authorities

- An outsourced CA is different from a public CA.
 - It provides dedicated services, and possibly equipment, to an individual company, whereas a public CA can be used by hundreds or thousands of companies.



A PKI service provider (represented by the four boxes) can offer different PKI components to companies.



Outsourced Certificate Authorities

- Outsource different parts of it to a specific service provider.
 - The more complex parts are outsourced, such as the CA, RA, CRL, and key recovery mechanisms.

- It is used when the company does not have the necessary skills to implement and carry out a full PKI environment.

- You must determine the level of trust the company is willing to give the service provider and what level of risk it is willing to accept.



Tying Different PKIs Together

- More than one CA may be needed for a specific PKI to work properly.
 - Different PKIs may need to intercommunicate.
 - Examples:
 - Communication with various suppliers, customers and business partners via PKI
 - Varying security requirements between departments
 - Such situations can add much more complexity to the overall infrastructure, intercommunication capabilities, and procedures for certificate generation and validation.
 - To control this complexity from the beginning, these requirements need to be understood, addressed, and planned.
 - **The necessary trust model needs to be chosen and molded for the company to give the company a solid foundation from the beginning.**



Trust Models

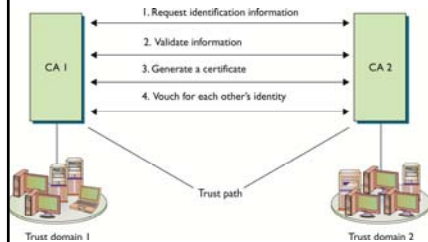
- A trust anchor is the agreed-upon trusted third party.
 - Different trust domains trust the common anchor, and therefore can trust each other
- Trust domain is a construct of systems, personnel, applications, protocols, technologies, and policies that work together to provide a certain level of protection.
- All the components can work together seamlessly within the same trust domain since they are known to the other components within the domain and are trusted to some degree.
- The different trust domains:
 - Are managed by different administrators.
 - Have different security policies.
 - Restrict outsiders from privileged access.
- Most trust domains and need to communicate with other, less-trusted domains



Unidirectional and Bidirectional

- The trust models describe and outline the trust relationships between the different CAs and different environments, which indicate where the trust paths reside.
- The trust path can be unidirectional or bidirectional, so either the two CAs trust each other (bidirectional) or only one trusts the other (unidirectional).
- The trust models and paths need to be thought out before implementation to restrict and control access and to ensure that there are as few trust paths as possible.
- **There are several forms of trust models.**
 - **Hierarchical, peer-to-peer, and hybrid**

Establishing Trust

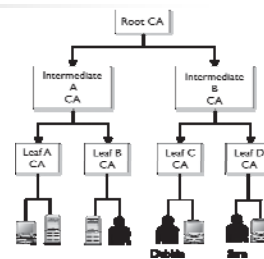


- All the users and devices in trust domain 1 trust their own CA 1, which is their trust anchor and all users and devices in trust domain 2 have their own trust anchor, CA 2.
- If the two CAs have exchanged certificates and trust each other, they do not have (or need) a common trust anchor between them.

A trust relationship can be built between two trust domains to set up a communication channel.

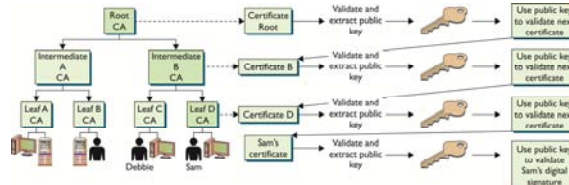
Hierarchical Trust Model

- The first type of trust model is a basic hierarchical structure that contains a root CA, intermediate CAs, leaf CAs, and end-entities.
- The root CA is the ultimate trust anchor for all other entities in this infrastructure.
 - It generates certificates for the intermediate CAs,
 - which in turn generate certificates for the leaf CAs,
 - and the leaf CAs generate certificates for the end-entities (users, network devices, and applications).
 - There are no bidirectional trusts. They are all unidirectional trusts.
- Since no other entity can certify and generate certificates for the root CA, it creates a self-signed certificate.



Verifying Certificates in a Certificate Path

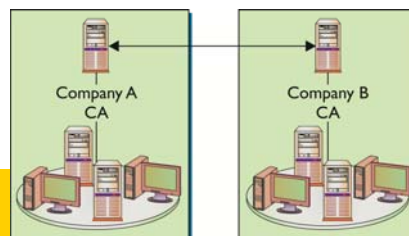
- When a user in one trust domain needs to communicate with another user in another trust domain, one user will need to validate the other's certificate.
 - Thus, each certificate for each CA, up to a shared trusted anchor, needs to be validated.
- Certificate path, is the path needed to be taken to continue to track down and collect certificates until it came upon a self-signed certificate
- Following the certificate path refers to when software has traversed the hierarchy to track and collect certificates until it comes upon a self-signed certificate.
 - A self-signed certificate indicates it was signed by a root CA.



Peer-to-Peer Model

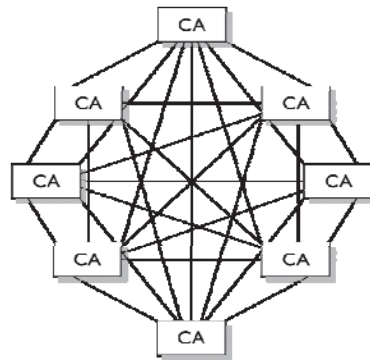
- In peer-to-peer trust models, one CA is not subordinate to another, and there is no established trusted anchor between the CAs.
 - the two CAs will certify the public key for each other, which creates a bidirectional trust.
 - **Also known as cross certification model**, since the CAs do not receive their certificates and public keys from a superior CA, but instead they create them for each other.

Cross certification creates a peer-to-peer PKI model.



Peer-to-Peer Model- Complexity

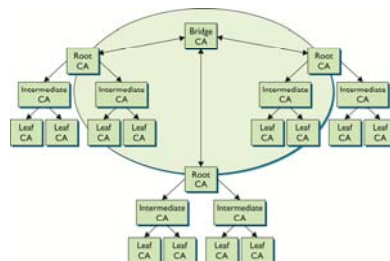
- One of the main drawbacks of this model is scalability.
 - Each CA must certify every other CA that is participating, and a bidirectional trust path must be implemented.
- The diagram represents fully connected mesh architecture, meaning each CA is directly connected to and has a bidirectional trust relationship with every other CA.




Scalability is a drawback in cross-certification models.

Hybrid Trust Model

- In a hybrid trust model, two companies have their own internal hierarchical models and are connected through a peer-to-peer model using cross-certification.
- **Another hybrid configuration is to implement a bridge CA.**
 - **Responsible for issuing cross-certificates for all connected CAs and trust domains**
 - The bridge is not considered a root or trust anchor, but merely the entity that generates and maintains the cross-certification for the connected environments.

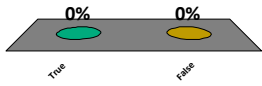


A bridge CA control the cross-certification procedures.




A Cross-certification model is used when independent CAs establish peer-to-peer trust relationship

1. True
2. False




Option	Percentage
True	0%
False	0%



Certificate-Based Threats

- Certificates bring much capability to security through practical management of trust however they also can present threats.
- Much of the actual work is done without direct user involvement.
- Can create a false sense of security.
- If an HTTPS connection is compromised...
 - Spoofing, phishing, pharming, and a wide range of sophisticated attacks are possible

- 
- If a hacker wishes to have something recognized as legitimate, he may have to obtain a certificate that proves this point to the end-user machine.
 - **Alt 1: Forge a false certificate**
 - But this is challenging because of the public key signing of certificates by CAs.
 - **Alt 2: hacker install a false, self-signed root certificate on the end-user PC.**
 - This false key can then be used to validate malicious software as coming from a trusted source.
 - This attack preys on the fact that end users do not know the contents of their root certificate store, nor do they have a means to validate changes.
 - **In an enterprise environment, this attack can be thwarted by locking down the certificate store and validating changes against a white list.**