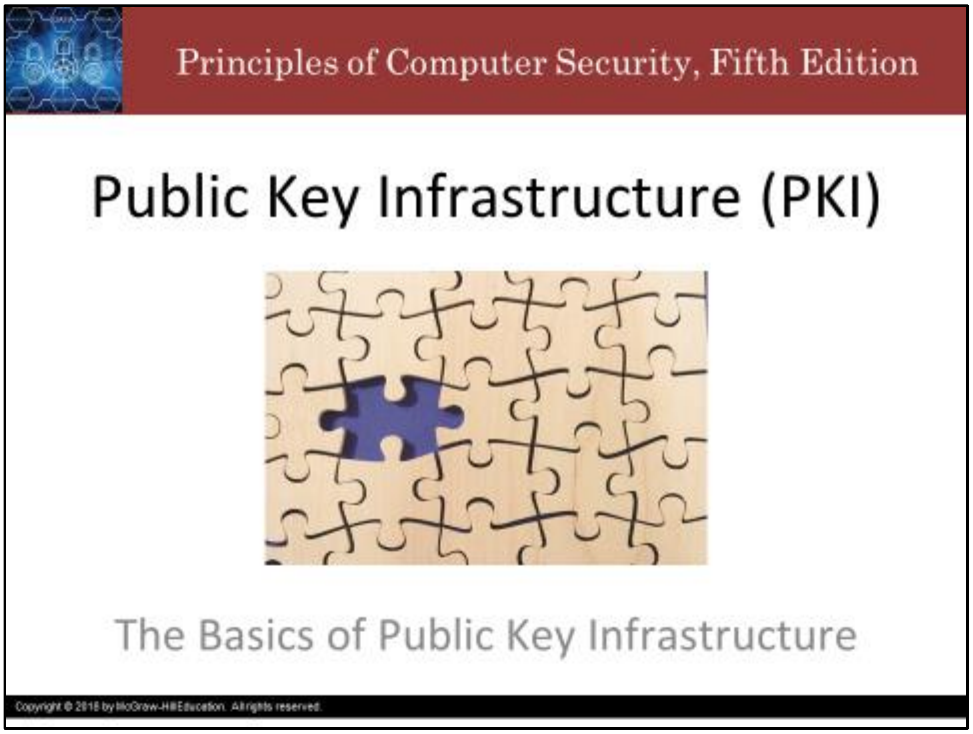
PKI: The Basics of Public Key Infrastructure

Slide 1



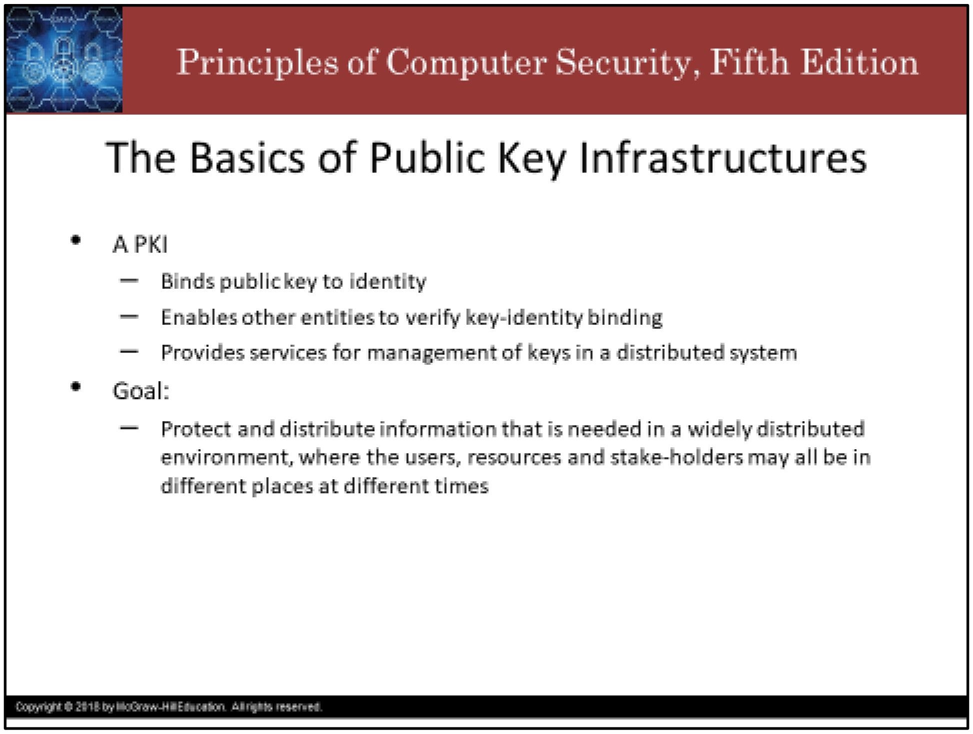
Principles of Computer Security, Fifth Edition

Public Key Infrastructure (PKI)

The Basics of Public Key Infrastructure

Howdy! In this video, we introduce the basics of Public Key Infrastructure.

Slide 2



Principles of Computer Security, Fifth Edition

The Basics of Public Key Infrastructures

- A PKI
  - Binds public key to identity
  - Enables other entities to verify key-identity binding
  - Provides services for management of keys in a distributed system
- Goal:
  - Protect and distribute information that is needed in a widely distributed environment, where the users, resources and stake-holders may all be in different places at different times

With only symmetric keys, the overhead of secure communication is overwhelming. On top of that, there is no identity management since a symmetric key is shared.

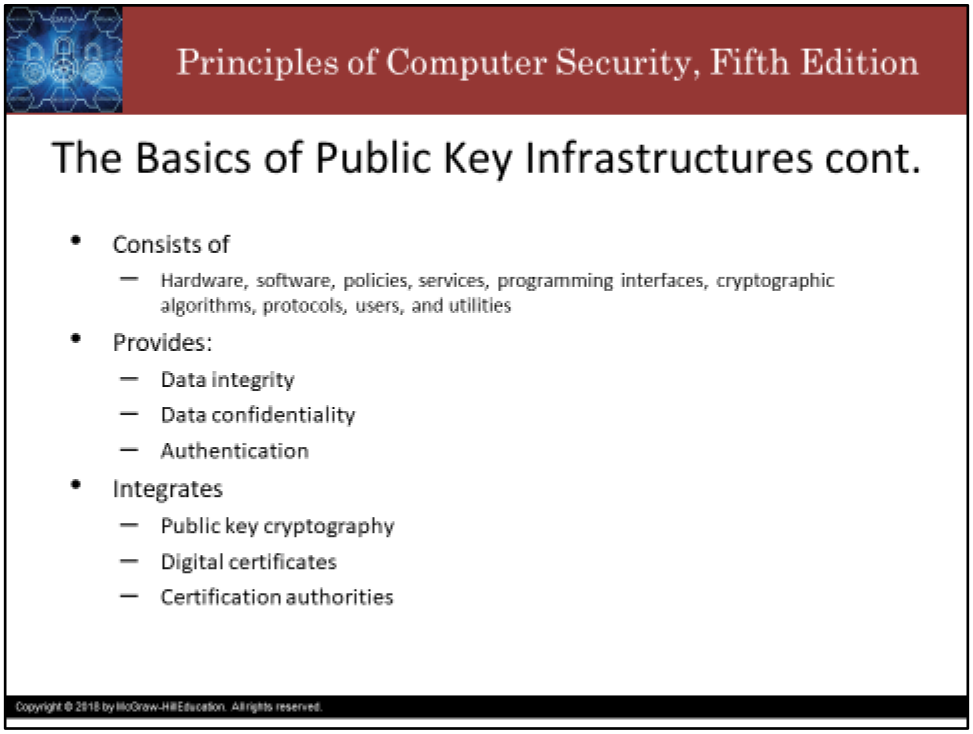
These problems are partially solved by public-key cryptography. But, practical application of public-key cryptography at scale requires more than just algorithms. It requires infrastructure.

A **public key infrastructure (PKI)** provides all the components necessary for different types of users and entities to be able to communicate securely using public key cryptography.

Some of those key functions include the binding of public keys to identities, verification of key-ID bindings, and services for key management.

The goal of a PKI is to protect and distribute information that is needed in a widely distributed environment, where the users, resources and stake-holders may all be in different places at different times.

Slide 3



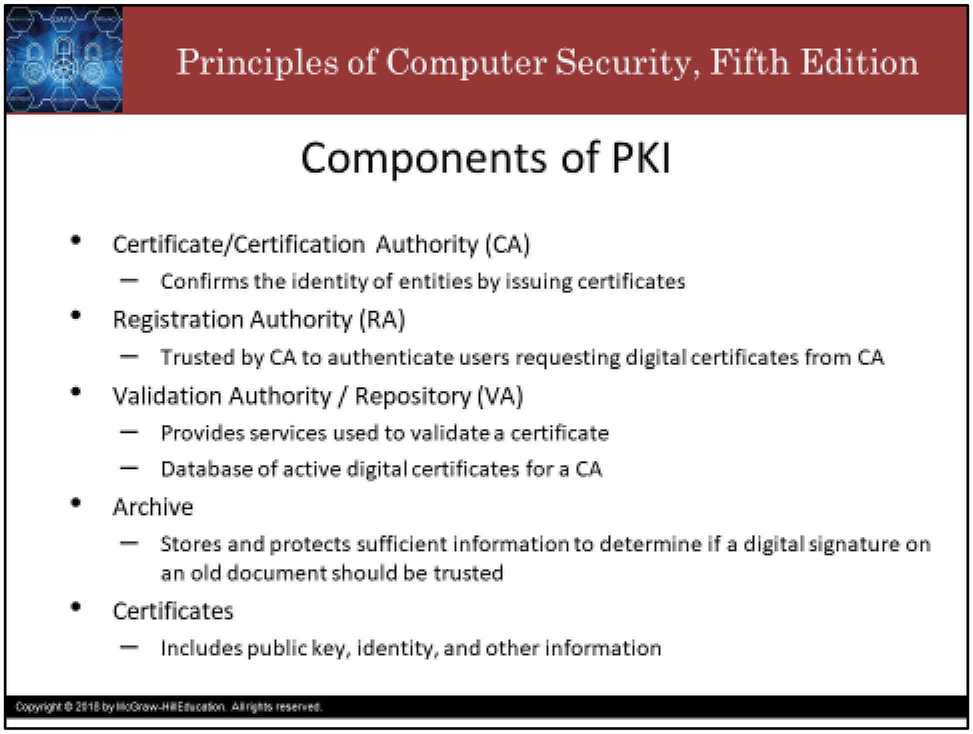
Principles of Computer Security, Fifth Edition

The Basics of Public Key Infrastructures cont.

- Consists of
  - Hardware, software, policies, services, programming interfaces, cryptographic algorithms, protocols, users, and utilities
- Provides:
  - Data integrity
  - Data confidentiality
  - Authentication
- Integrates
  - Public key cryptography
  - Digital certificates
  - Certification authorities

A PKI consists of many components, such as hardware, software, policies, services, programming interfaces, cryptographic algorithms, protocols, users, and utilities. I think that covers everything.
A PKI provides integrity, confidentiality and authentication functions (among other things) by integrating public key cryptography, digital certificates, and certification authorities.

Slide 4



Principles of Computer Security, Fifth Edition

Components of PKI

- Certificate/Certification Authority (CA)
  - Confirms the identity of entities by issuing certificates
- Registration Authority (RA)
  - Trusted by CA to authenticate users requesting digital certificates from CA
- Validation Authority / Repository (VA)
  - Provides services used to validate a certificate
  - Database of active digital certificates for a CA
- Archive
  - Stores and protects sufficient information to determine if a digital signature on an old document should be trusted
- Certificates
  - Includes public key, identity, and other information

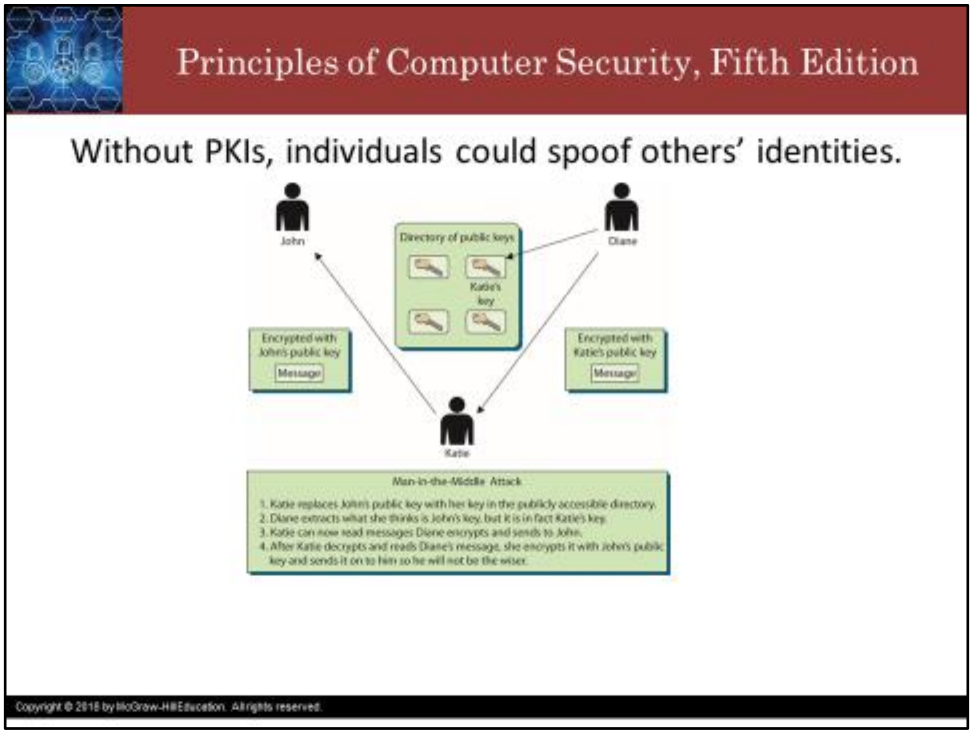
Constructing and implementing a PKI boils down to establishing a level of trust.

In PKI environments, entities like certificate authorities (CAs) and registration authorities (RAs) provide services similar to those of the Department of Motor Vehicles (DMV).

When Alice goes to register for a driver’s license, they have to prove their identity to the DMV by providing their passport, birth certificate, or other identification documentation. If the DMV is satisfied with the proof Alice provides (and Alice passes a driving test), the DMV will create a driver’s license that can then be used by Alice to prove their identity. Whenever Alice needs to identify themself, they can show her driver’s license. Although some people may not trust Alice to identify themself truthfully, they do trust the third party, the DMV. Similarly, the Certificate Authority, as the trusted entity, vouches for the integrity of Alice’s identity.

Other component of a PKI include validation authorities and archives which are used for validation of certificates and signatures.

Slide 5


John
Diane
Katie
Directory of public keys
Katie's key
Encrypted with John's public key
Message
Encrypted with Katie's public key
Message

Principles of Computer Security, Fifth Edition

Without PKIs, individuals could spoof others' identities.

Man-in-the-Middle Attack

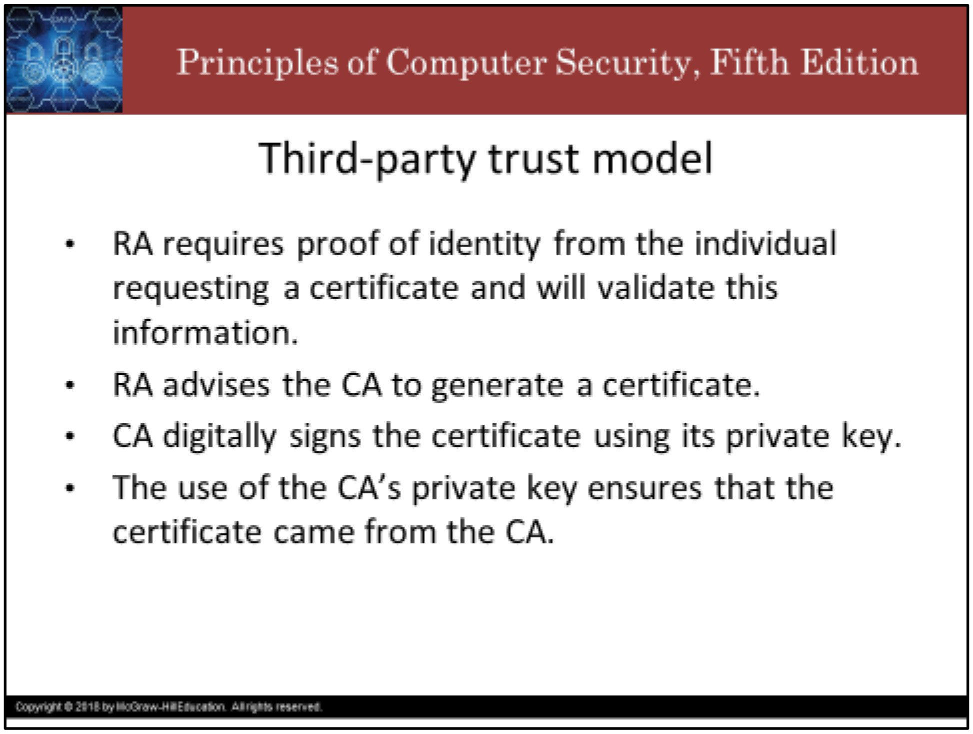
1. Katie replaces John's public key with her key in the publicly accessible directory.
2. Diane extracts what she thinks is John's key, but it is in fact Katie's key.
3. Katie can now read messages Diane encrypts and sends to John.
4. After Katie decrypts and reads Diane's message, she encrypts it with John's public key and sends it on to him so he will not be the wiser.

As I said earlier, the critical step in implementing a PKI is establishing a level of trust.

If Jack and Diane want to communicate securely, Jack can generate their own public/private key pair and send their public key to Diane, or Jack can place their public key in a directory that is available to everyone. When Diane receives Jack's public key, either from Jack or from a public directory, how does Diane know the key really came from Jack? Maybe another individual, Katie, is masquerading as Jack and has replaced Jack's public key with their own, which would be a masquerading attack (half of the man in the middle).

If this took place, Diane would be fooled into believing that their messages could be read only by Jack and that the replies were actually from Jack. However, Diane would actually be communicating with Katie. What is needed is a way to verify an individual's identity, to ensure that a person's public key is bound to their identity and thus ensure that this previous scenario (and others) cannot take place.

Slide 6



Principles of Computer Security, Fifth Edition

Third-party trust model

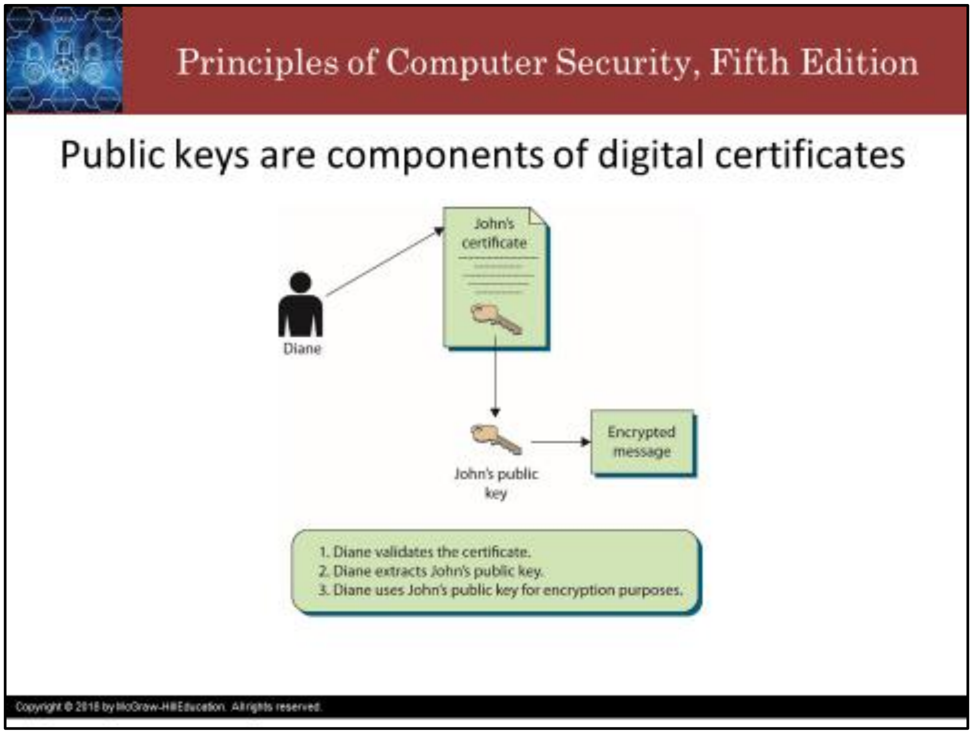
- RA requires proof of identity from the individual requesting a certificate and will validate this information.
- RA advises the CA to generate a certificate.
- CA digitally signs the certificate using its private key.
- The use of the CA's private key ensures that the certificate came from the CA.

Since Diane cannot trust Jack until after Jack has been authenticated, they need some trusted third party that authenticate Jack and provide Diane with assurance that Jack is who they say they is… are. The way this works is that Jack gets a certificate to use when authenticating themself. First, Jack goes to a Registration Authority (usually part of the Certificate Authority), and presents their identifying documents and their public key. The RA verifies and validates that Jack personal identity matches their documented identity (this step obviously depends on some earlier verification and validation done by higher powers, like the DMV, the department of vital records, the department of state, and so on, who have provided Jack with documentation about Jack's identity that only Jack should possess). The RA also confirms that Jack possesses the private key corresponding to the claimed public key. This can be done by asking Jack to correctly decrypt and sign several challenges. Then, the RA forwards Jack's identification information and public key to the CA with their own certification that they have duly authenticated Jack's identity and possession of the private key corresponding to the claimed public key. The CA then creates a certificate that binds Jack's public key with Jack's identity and then the CA signs the certificate with the CA's private key.

When Diane receives Jack's certificate and verifies that it was actually digitally signed by a Certificate Authority that they trust, they will believe that the certificate is actually Jack's - not because Diane trusts Jack, but because Diane trusts the CA that is vouching for Jack's identity.

Public keys are components of digital certificates, so when Diane verifies the CA's digital signature, this verifies that the certificate is truly Jack's and that the public key the certificate contains is also Jack's. This is how Jack's identity is bound to their public key. This process allows Jack to authenticate themself to Diane and others. Using the third-party certificate, Jack can communicate with Diane, using public key encryption, without prior communication or a preexisting relationship.

Slide 7



Principles of Computer Security, Fifth Edition

Public keys are components of digital certificates

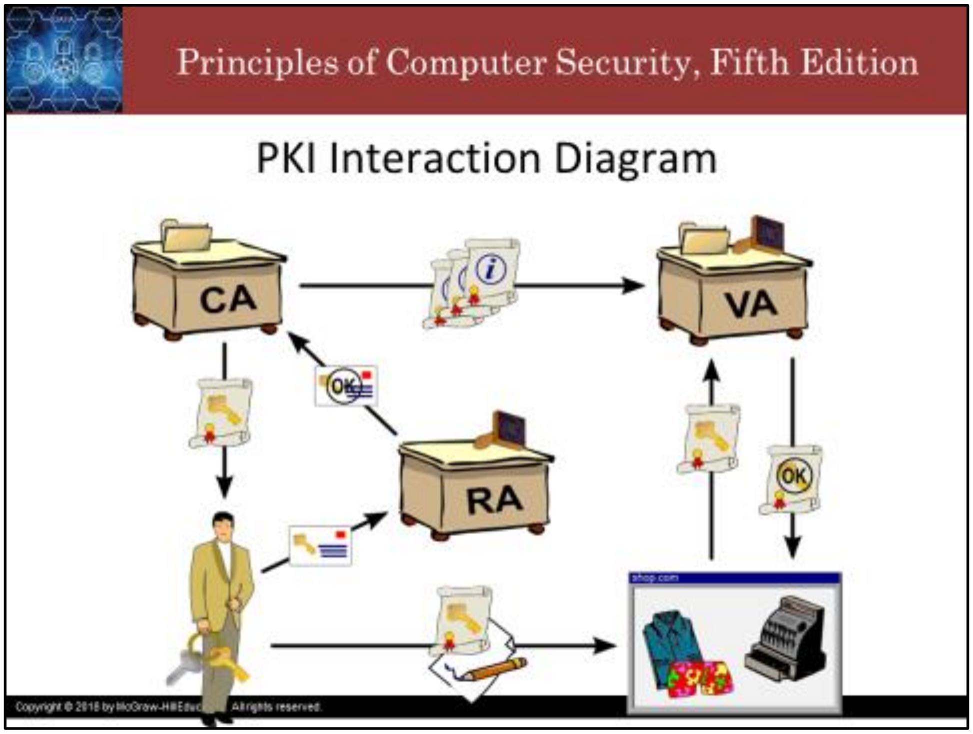
1. Diane validates the certificate.
2. Diane extracts John's public key.
3. Diane uses John's public key for encryption purposes.

Once Diane is convinced of the authenticity of Jack's public key, she can use it to encrypt messages that only Jack can read.

Even if Katie sends Jacks' certificate to Diane, Katie will not be able to read Diane's messages encrypted using Jack's public key.

If, however, Katie manages to convince the RA and the CA that Katie is Jack or steals the CA's private key or Diane simply improperly validates the certificate, the Diane is in trouble since the public key in the certificate that says "Jack" will actually be Katie's and messages that Diane encrypts for Jack will actually be read by Katie and not Jack. In this scenario, Katie succeeds in masquerading as Jack. If Katie can masquerade to Jack as Diane, then Katie pulls of the famed man in the middle attack and now owns Jack and Diane's communication channel. This is why the private keys of certificate authorities must be kept safe from attackers. Being able to create any certificate the attacker wants would be a catastrophe for security on the Internet.

Slide 8

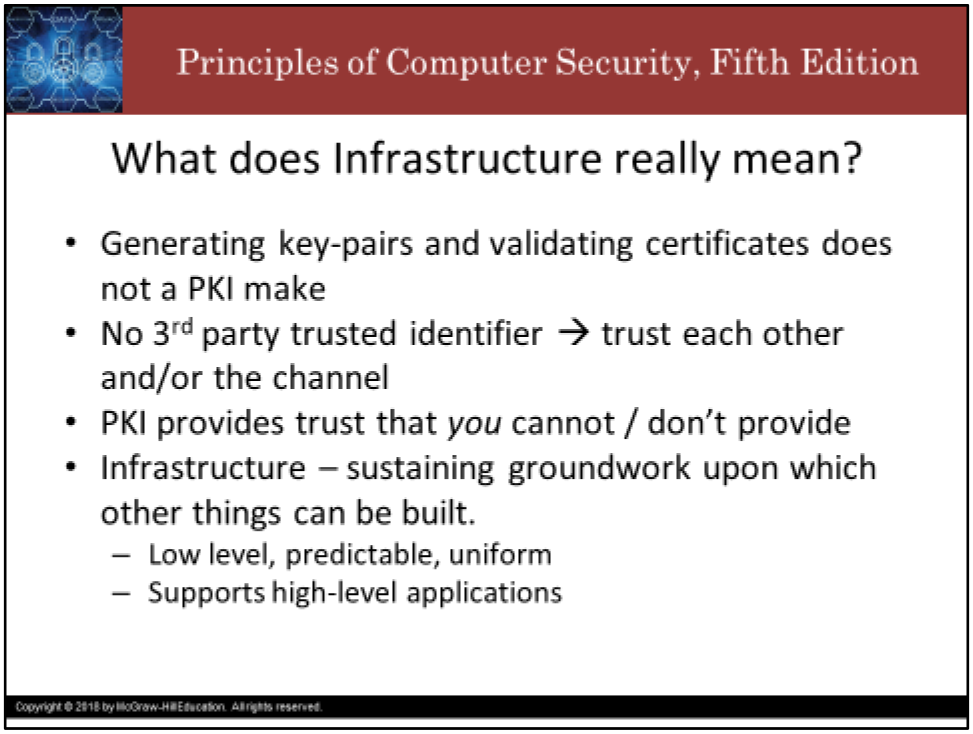


Principles of Computer Security, Fifth Edition

PKI Interaction Diagram

This is a big-picture interaction diagram for how a PKI can operate. Alice/Bob/Jack/whoever is down there in the bottom left. Let's say it's Bob. Bob generates a public-private key pair, gathers their identifying documents and presents themselves to the RA. The RA verifies Bob's identify and possession of the private and produces a proof of authentication for the CA. The CA combines Bob's identity information with Bob's public key and signs it with the CA's private key to produce the certificate, which is sent to Bob. Congratulations, Bob! It's an X.509!!! The CA sends information about the certificates it has signed to the VA for safe keeping. At some point, Bob goes shopping at shop.com, the world's most elite eStore because it requires Bob to authenticate themself (this is an option in TLS, but typically only the server authenticates to the client since how many people do you know who have a public-key certificate signed by a trusted CA? Servers are very chatty, they'll talk to anyone, they don't care who). The store takes Bob's certificate and sends it off the to the VA, who validates the certificates and sends back the OK. Now shop.com has assurance that messages encrypted using the public key in the cert sent by "allegedly Bob" can only be read by "actually Bob". For most intents and purposes, this is sufficient to convince shop.com that "allegedly Bob" is "actually Bob", since anyone who is not "actually Bob" will be unable to read any of the replies (since only "actually Bob" has the private key for decryption).

Slide 9



Principles of Computer Security, Fifth Edition

What does Infrastructure really mean?

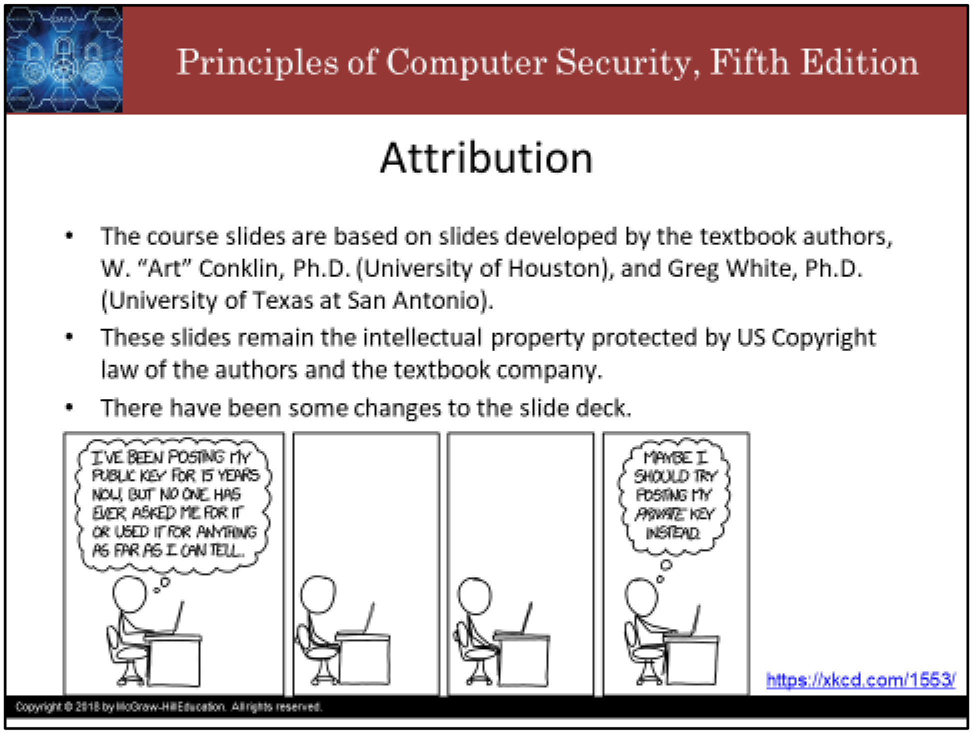
- Generating key-pairs and validating certificates does not a PKI make
- No 3rd party trusted identifier → trust each other and/or the channel
- PKI provides trust that *you* cannot / don't provide
- Infrastructure – sustaining groundwork upon which other things can be built.
  - Low level, predictable, uniform
  - Supports high-level applications

Numerous applications and protocols generate public/private key pairs and provide functionality similar to what a PKI provides.
However, without a trusted third party available to vouch for the other's identity, Alice and Bob must trust each other or the communication channel.
This kind of arrangement can work for small groups, especially ones in which the members have out-of-band methods of communication, including the ability to meet face to face and a social history (i.e. people who know and trust each other already).
Trusting the channel is almost never a good idea.
PKI components provide the necessary level of trust that you (untrusted and anonymous netizen that you are) cannot, or choose not to, provide on your own.
An infrastructure provides a sustaining groundwork upon which other things can be built.
It works at a low level to provide a predictable and uniform environment that allows other, higher-level technologies to work together through uniform access points.

The environment the infrastructure provides allows higher-level applications to communicate with each other and gives them the underlying tools to carry out their tasks.

Infrastructure is important and should be maintained and improved for best operation that works for the benefit of all and not just a well-resourced and privileged few.

Slide 10



Principles of Computer Security, Fifth Edition

Attribution

- The course slides are based on slides developed by the textbook authors, W. "Art" Conklin, Ph.D. (University of Houston), and Greg White, Ph.D. (University of Texas at San Antonio).
- These slides remain the intellectual property protected by US Copyright law of the authors and the textbook company.
- There have been some changes to the slide deck.

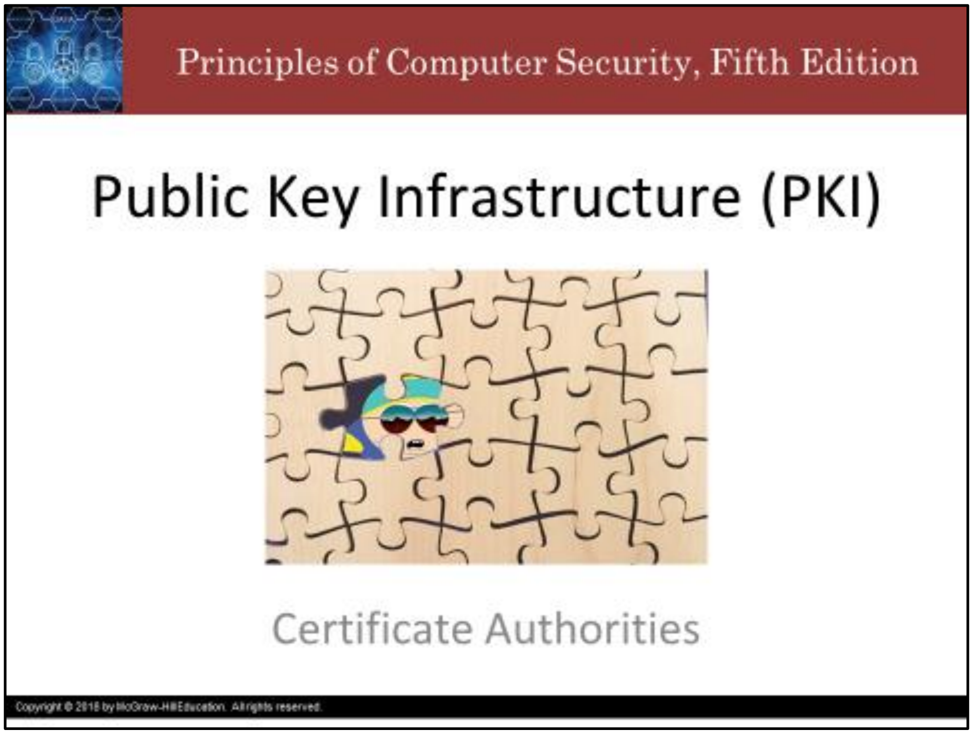
I'VE BEEN POSTING MY PUBLIC KEY FOR 15 YEARS NOW, BUT NO ONE HAS EVER ASKED ME FOR IT OR USED IT FOR ANYTHING AS FAR AS I CAN TELL.
MAYBE I SHOULD TRY POSTING MY PRIVATE KEY INSTEAD.

https://xkcd.com/1553/

Thank you and take care.

PKI: Certificate Authorities

Slide 1



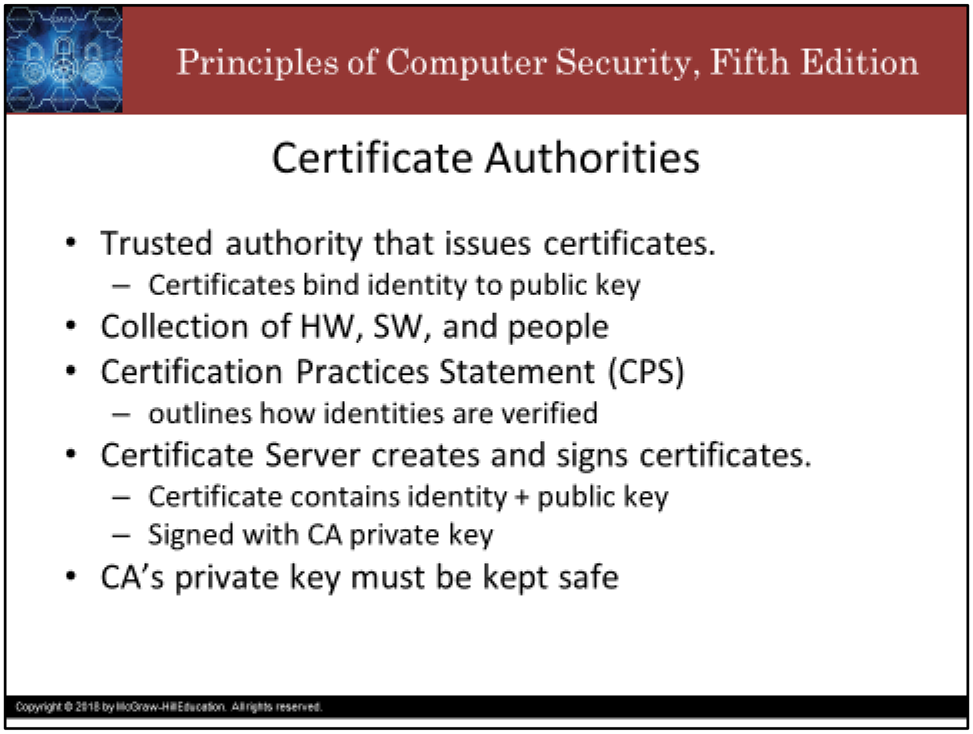
Principles of Computer Security, Fifth Edition

Public Key Infrastructure (PKI)

Certificate Authorities

Howdy! In this video, we discuss Certificate Authorities.

Slide 2



Principles of Computer Security, Fifth Edition

Certificate Authorities

- Trusted authority that issues certificates.
  - Certificates bind identity to public key
- Collection of HW, SW, and people
- Certification Practices Statement (CPS)
  - outlines how identities are verified
- Certificate Server creates and signs certificates.
  - Certificate contains identity + public key
  - Signed with CA private key
- CA's private key must be kept safe

A **certificate authority (CA)** is a basic building block of the PKI. A CA is a trusted authority that certifies individuals' identities and creates electronic documents indicating that individuals are who they say they are.

The electronic document is referred to as a **digital certificate**, and it establishes an association between the subject's identity and a public key.

The private key that is paired with the public key in the certificate is stored separately and kept secret by the subject..
A CA is a collection of hardware, software, and the people who operate it.
If one CA component is compromised, it can negatively affect the overall integrity of the CA.

Every CA should have a **certification practices statement** (**CPS**), which outlines how identities are verified; the steps the CA follows to generate, maintain, and transmit certificates; and why the CA can be trusted to fulfill its responsibilities.

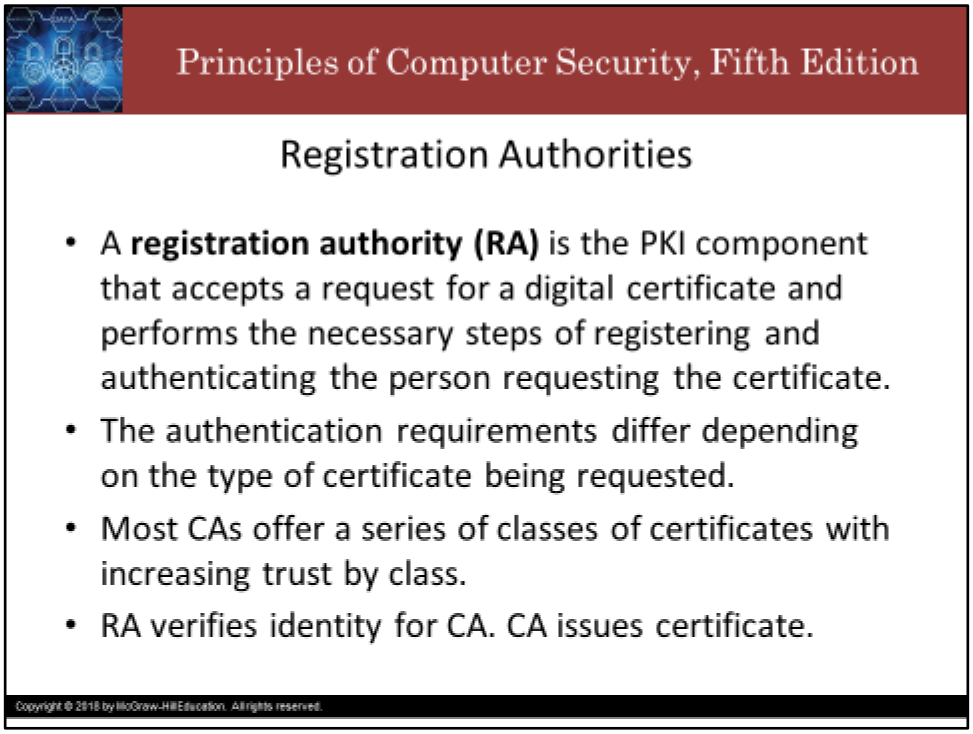
The CPS describes how keys are secured, what data is placed within a digital certificate, and how revocations will be handled. If a company is going to use and depend on a public CA, the company's security officers, administrators, and legal department should review the CA's entire CPS to ensure that it will properly meet the company's needs, and to make sure that the level of security claimed by the CA is high enough for their use and environment. A critical aspect of a PKI is the trust between the users and the CA, so the CPS should be reviewed and understood to ensure that this level of trust is warranted.

A **certificate server** is the actual service that issues certificates based on the data provided during the initial registration process. The server constructs the certificate and signs it with the CA's private key.

By issuing a certificate, CA asserts the subject of the certificate has the private key associated with the public key of the certificate: "yes, this person is who they say they are, and we, the CA, certify that“.

The CA’s private key must be kept extremely safe and secure. If an attacker compromises a CA’s private key, they can use it to sign any certificate they want and do massive amounts of harm to people and systems.

Slide 3



Principles of Computer Security, Fifth Edition

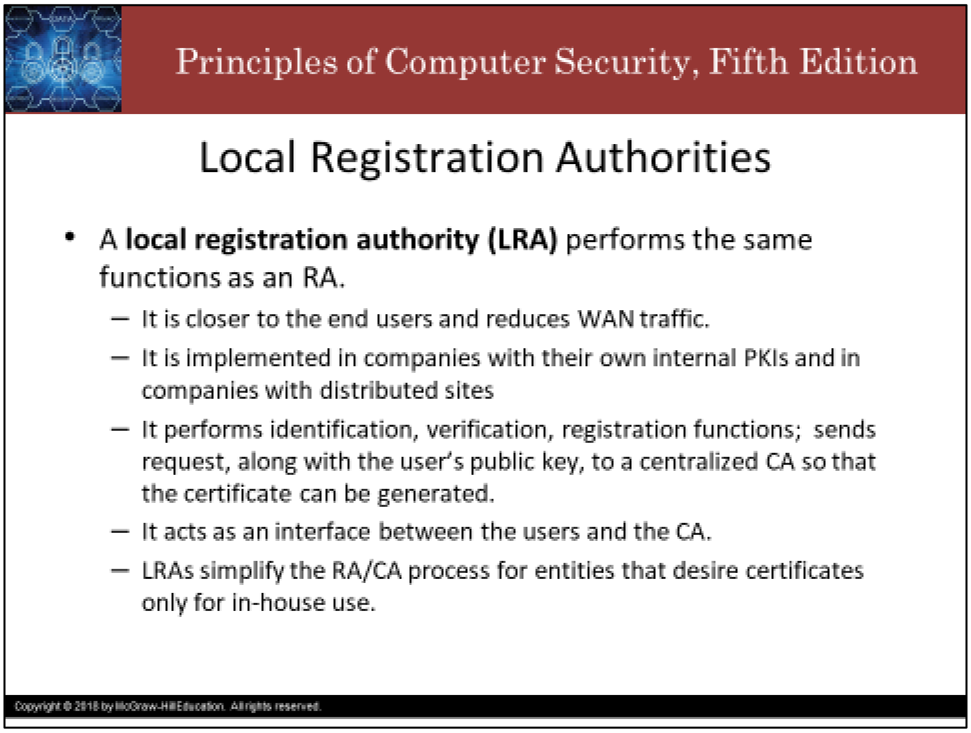
Registration Authorities

- A **registration authority (RA)** is the PKI component that accepts a request for a digital certificate and performs the necessary steps of registering and authenticating the person requesting the certificate.
- The authentication requirements differ depending on the type of certificate being requested.
- Most CAs offer a series of classes of certificates with increasing trust by class.
- RA verifies identity for CA. CA issues certificate.

The Registration Authority (RA) verifies certificate contents for the CA.
The RA collects the identifying information and sends it to the CA.
RAs are usually operated by a single person, whereas a CA is larger and has many people
Each CA maintains a list of trusted and accredited RAs
The RA is known to the CA by a name and public key (i.e. the RA has a certificate of it’s own that identifies it with it’s public key).
That CA can verify the RA's signature, to be sure that the information obtained is reliable.
Like CAs, RAs must provide adequate protection for their private keys.

So, the RA verifies the identity of the certificate requestor on behalf of the CA. Then, the CA generates the certificate using information forwarded by the RA.

Slide 4



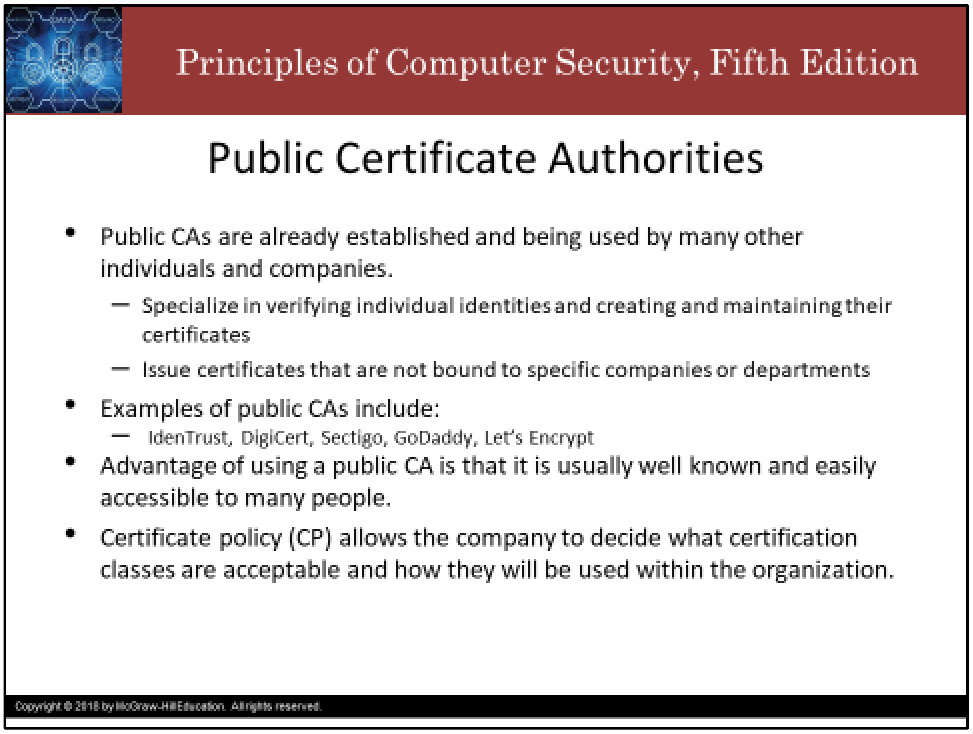
Principles of Computer Security, Fifth Edition

Local Registration Authorities

- A **local registration authority (LRA)** performs the same functions as an RA.
  - It is closer to the end users and reduces WAN traffic.
  - It is implemented in companies with their own internal PKIs and in companies with distributed sites
  - It performs identification, verification, registration functions; sends request, along with the user's public key, to a centralized CA so that the certificate can be generated.
  - It acts as an interface between the users and the CA.
  - LRAs simplify the RA/CA process for entities that desire certificates only for in-house use.

A **local registration authority (LRA)** performs the same functions as an RA, but it is closer to the end users and reduces network traffic.
LRAs are typically implemented in companies with their own internal PKIs and in companies with distributed sites.
The LRA performs identification, verification, and registration functions; it sends the certificate request, which includes the user's public key, to a centralized CA so that the certificate can be generated and issued.
The LRA acts as an interface between the users and the CA.
LRAs simplify the RA/CA process for entities that need certificates for local, in-house use only.

Slide 5



Principles of Computer Security, Fifth Edition

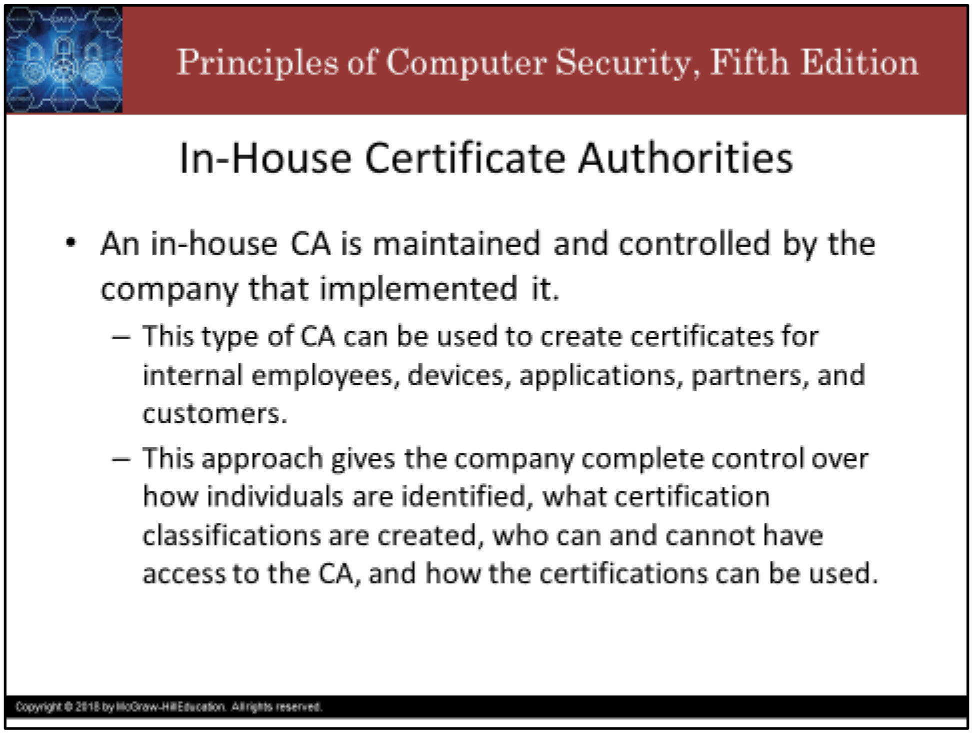
Public Certificate Authorities

- Public CAs are already established and being used by many other individuals and companies.
  - Specialize in verifying individual identities and creating and maintaining their certificates
  - Issue certificates that are not bound to specific companies or departments
- Examples of public CAs include:
  - IdenTrust, DigiCert, Sectigo, GoDaddy, Let's Encrypt
- Advantage of using a public CA is that it is usually well known and easily accessible to many people.
- Certificate policy (CP) allows the company to decide what certification classes are acceptable and how they will be used within the organization.

Public CAs are Cas which have already been established and are used by many other individuals and companies.
They specialize in verifying individual identities and creating and maintaining their certificates.
The certificates issued by public Cas are not bound to specific companies or departments, unlike an in-house CA which is specific to a single company or organization.

Examples of public CAs include: IdenTrust, DigiCert (which also maintains the GeoTrust, Thawte, and RapidSSL brands), Sectigo (the CA formerly knowns as Comodo), GoDaddy, and Let's Encrypt, which together account for over 95% of all certifcates.
The advantage of using a public CA is that it is usually well known, trusted, and easily accessible to many people.
A certificate policy allows a users to decide what certification classes are acceptable and how they will be used within the organization.

Slide 6



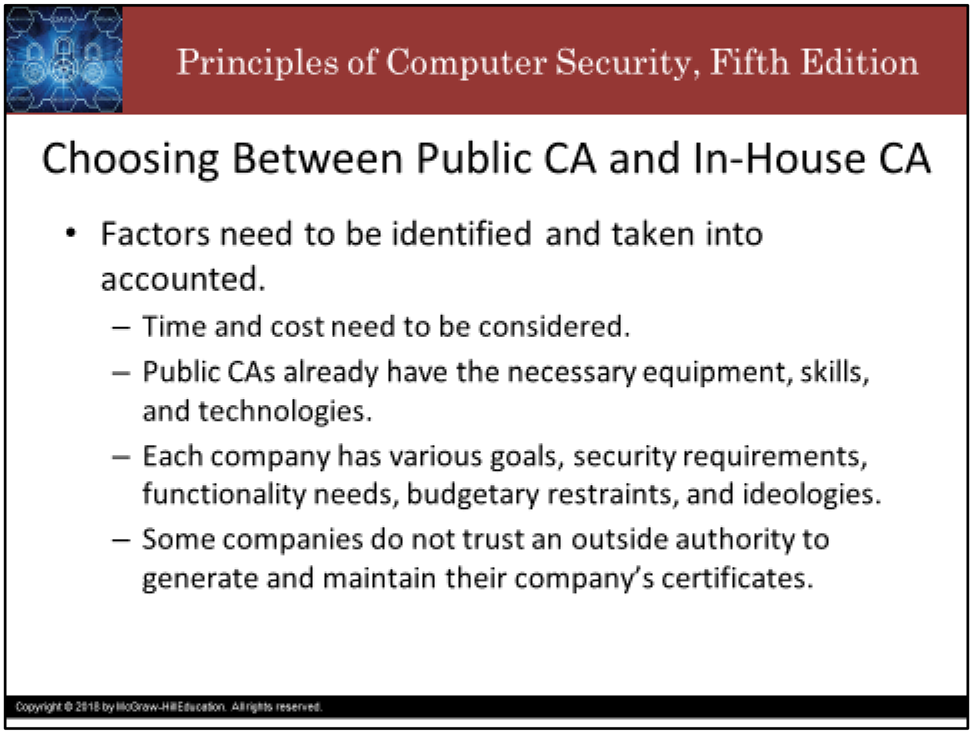
Principles of Computer Security, Fifth Edition

In-House Certificate Authorities

- An in-house CA is maintained and controlled by the company that implemented it.
  - This type of CA can be used to create certificates for internal employees, devices, applications, partners, and customers.
  - This approach gives the company complete control over 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.

An in-house CA is maintained and controlled by the company that implemented it.
This type of CA can be used to create certificates for internal employees, devices, applications, partners, and clients of the organization that owns the CA.
This approach gives the company complete control over 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.

Slide 7



Principles of Computer Security, Fifth Edition

Choosing Between Public CA and In-House CA

- Factors need to be identified and taken into accounted.
  - Time and cost need to be considered.
  - Public CAs already have the necessary equipment, skills, and technologies.
  - Each company has various goals, security requirements, functionality needs, budgetary restraints, and ideologies.
  - Some companies do not trust an outside authority to generate and maintain their company's certificates.

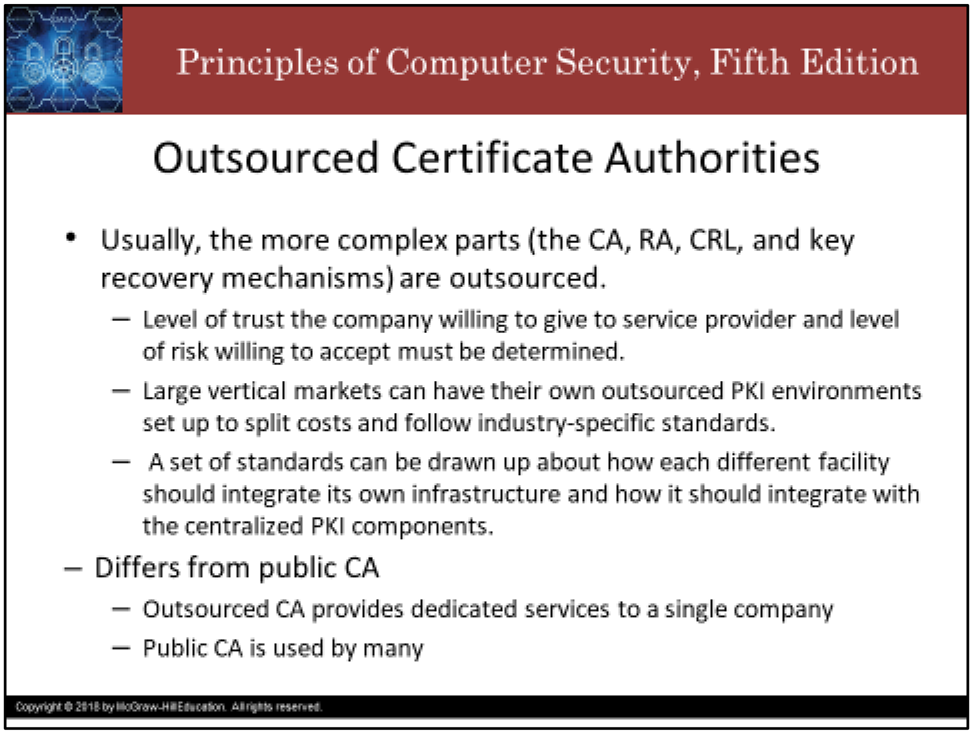
When deciding whether to use a public CA or an in-house CA, there are several factors that need to be identified and taken into account.

For certain, time and cost need to be considered. Implementing and maintaining your own PKI is resources intensive. The costs may outweigh the benefits.

It maybe faster and easier (and possibly even less expensive) to use a public CA.

Each company is unique, with various goals, security requirements, functionality needs, budgetary restraints, and ideologies. The decision of whether to use a private CA or an in-house CA depends on the expansiveness 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, and how it will work with outside entities. This could be quite a large undertaking that ties up staff, resources, and funds, so a lot of strategic planning is required, and what will and won't be gained from a PKI should be fully understood before the first dollar is spent on the implementation.

Slide 8



Principles of Computer Security, Fifth Edition

Outsourced Certificate Authorities

- Usually, the more complex parts (the CA, RA, CRL, and key recovery mechanisms) are outsourced.
  - Level of trust the company willing to give to service provider and level of risk willing to accept must be determined.
  - Large vertical markets can have their own outsourced PKI environments set up to split costs and follow industry-specific standards.
  - A set of standards can be drawn up about how each different facility should integrate its own infrastructure and how it should integrate with the centralized PKI components.
- Differs from public CA
  - Outsourced CA provides dedicated services to a single company
  - Public CA is used by many

In the middle ground between implementing your own PKI from scratch and using a complete solution from a public CA is using outsourced CAs.

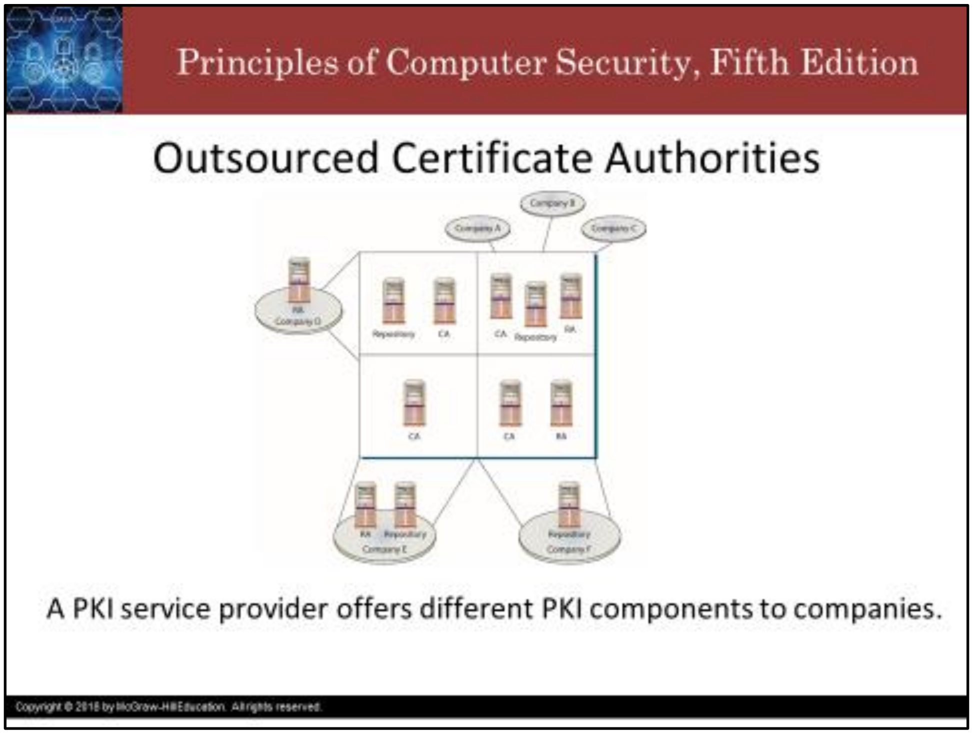
Usually, the more complex parts, like the CA, RA, certificate revocation list, and key recovery mechanisms) are outsourced.

The level of trust that the company is willing to give to the service provider and the level of risk they are willing to accept must be determined.

Some large vertical markets have their own outsourced PKI environments set up because they share similar needs and usually have the same requirements for certification types and uses. This allows several companies within the same market to split the costs of the necessary equipment, and it allows for industry-specific standards to be drawn up and followed.

A set of standards can be drawn up about how each different facility should integrate its own infrastructure and how it should integrate with the centralized PKI components.

Slide 9



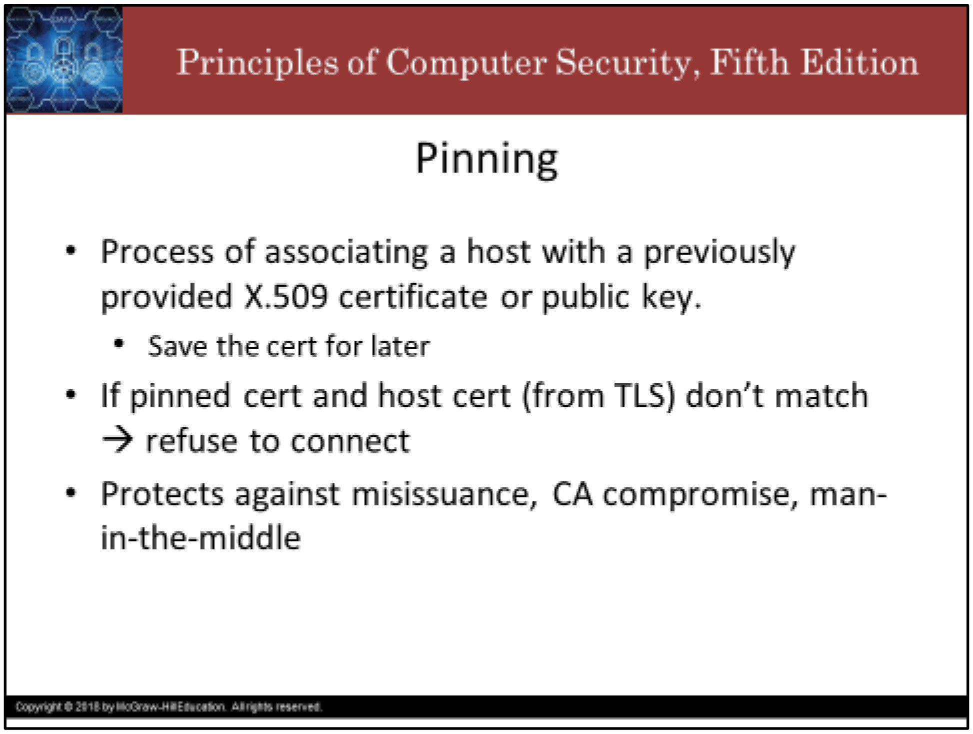
Principles of Computer Security, Fifth Edition

Outsourced Certificate Authorities

A PKI service provider offers different PKI components to companies.

This figure illustrates how one outsourced service provider can offer different PKI components and services to different companies, and how companies within one vertical market can share the same resources.

Slide 10



Principles of Computer Security, Fifth Edition

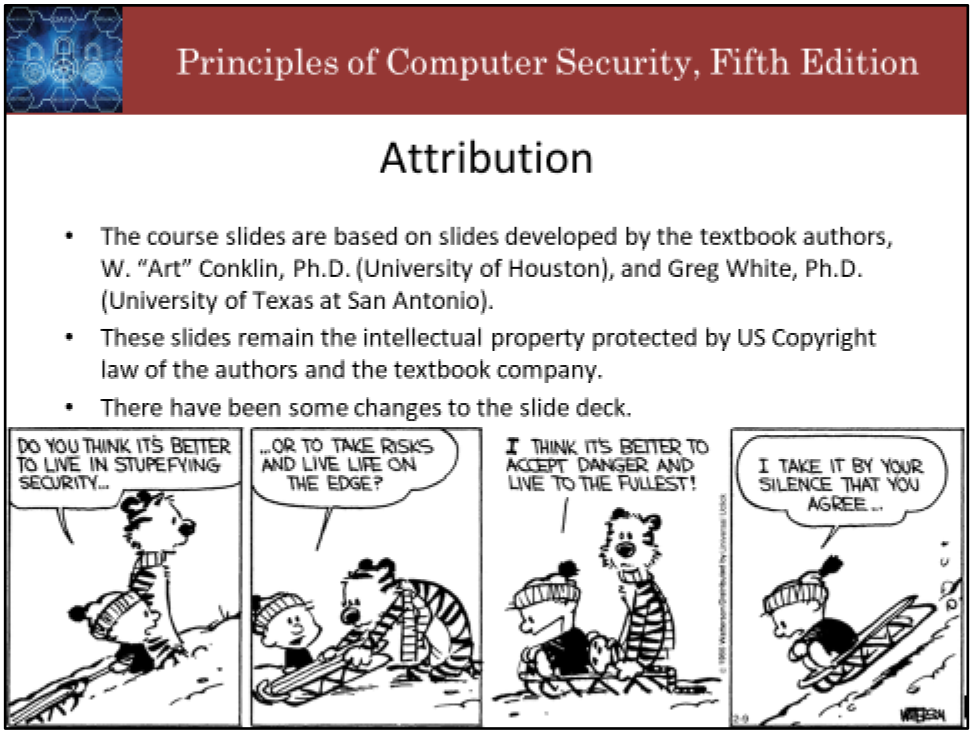
Pinning

- Process of associating a host with a previously provided X.509 certificate or public key.
  - Save the cert for later
- If pinned cert and host cert (from TLS) don’t match → refuse to connect
- Protects against misissuance, CA compromise, man-in-the-middle

When a certificate is presented for a host, a local copy can be saved for use later. This is called pinning.

Pinning protects against hostile networks where the risk of malicious data is high. Assuming the certificate is pinned while the client is on a trusted network, when the client later moves to an untrusted network (such as connecting their phone or laptop to a public WiFi hotspot) and connects to the host, the pinned certificate allows the client to compare the host certificate sent over the untrusted network to the pinned certificate and refuse to connect if the certificates don’t match, which could indicate CA compromise or a man in the middle attack.

Slide 11



Principles of Computer Security, Fifth Edition

Attribution

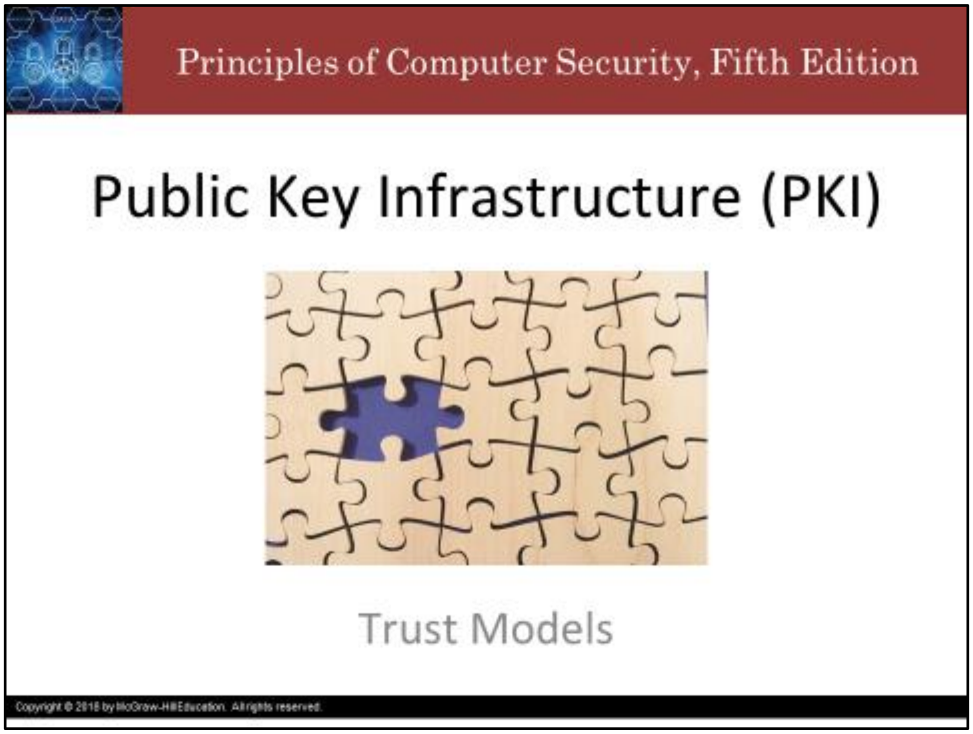
- The course slides are based on slides developed by the textbook authors, W. “Art” Conklin, Ph.D. (University of Houston), and Greg White, Ph.D. (University of Texas at San Antonio).
- These slides remain the intellectual property protected by US Copyright law of the authors and the textbook company.
- There have been some changes to the slide deck.

DO YOU THINK IT'S BETTER TO LIVE IN STUPEFYING SECURITY...
...OR TO TAKE RISKS AND LIVE LIFE ON THE EDGE?
I THINK IT'S BETTER TO ACCEPT DANGER AND LIVE TO THE FULLEST!
I TAKE IT BY YOUR SILENCE THAT YOU AGREE...
WATTERSON

Thank you and take care.

PKI: Trust Models

Slide 1



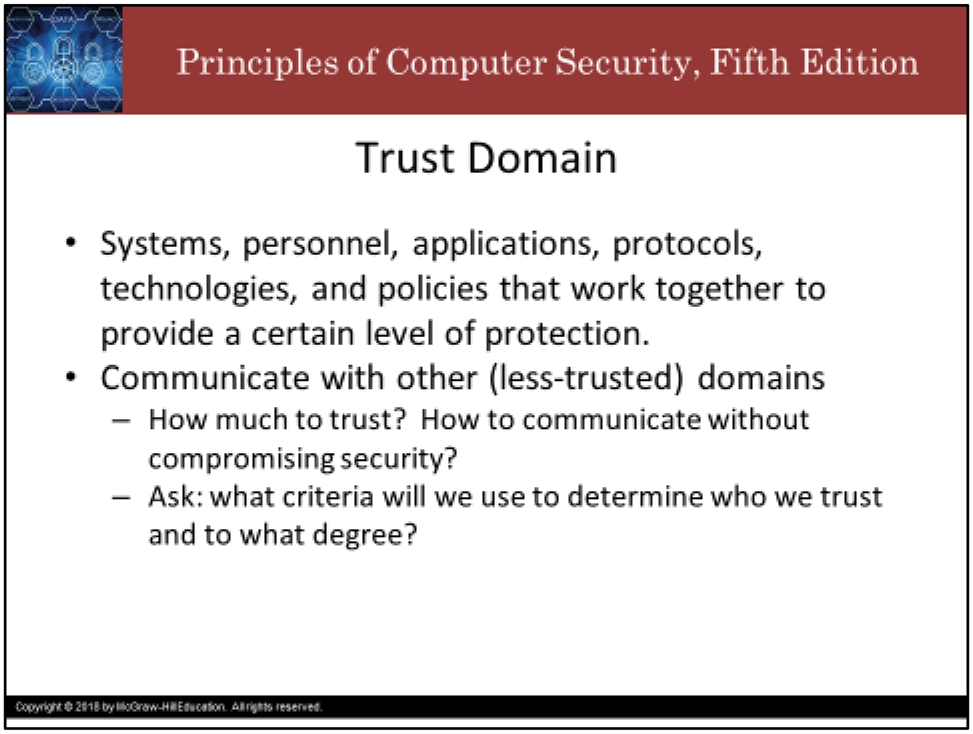
Principles of Computer Security, Fifth Edition

Public Key Infrastructure (PKI)

Trust Models

Howdy! In this video, we discuss trust models.

Slide 2



Principles of Computer Security, Fifth Edition

Trust Domain

- Systems, personnel, applications, protocols, technologies, and policies that work together to provide a certain level of protection.
- Communicate with other (less-trusted) domains
  - How much to trust? How to communicate without compromising security?
  - Ask: what criteria will we use to determine who we trust and to what degree?

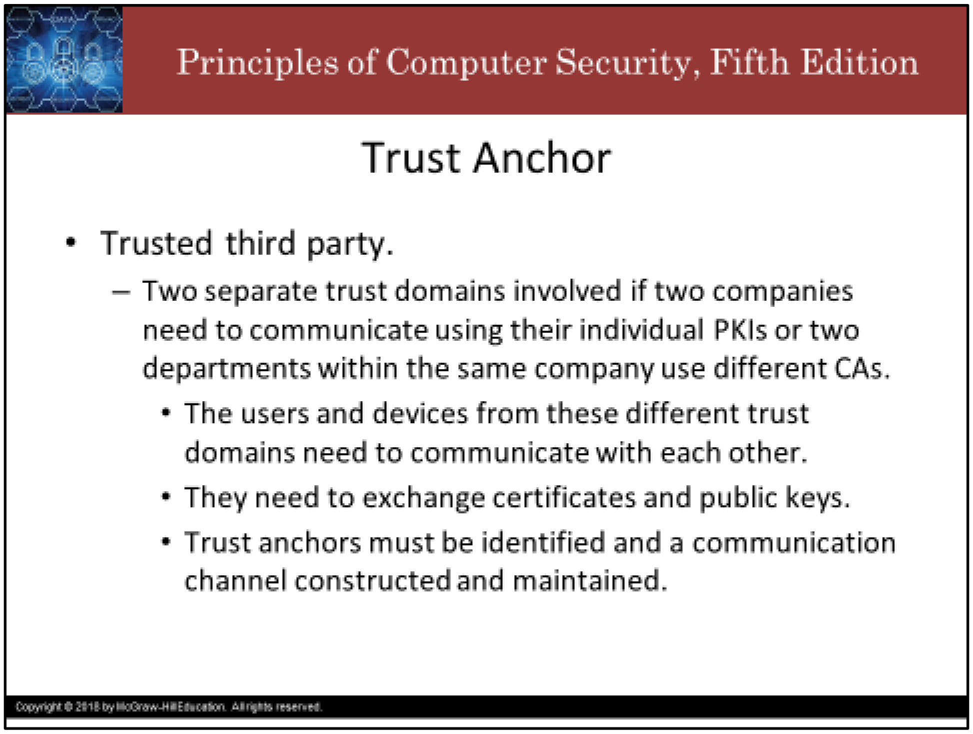
A trust domain is a construct of systems, personnel, applications, protocols, technologies, and policies that work together to provide a certain level of protection.
Most trust domains need to communicate with other, less-trusted domains.
The key problems are how much two different domains should trust each other, and how to implement and configure an infrastructure that would allow these two domains to communicate in a way that will not allow security compromises or breaches.

In the nondigital world, it is difficult to figure out who to trust, how to carry out legitimate business functions, and how to ensure that one is not being taken advantage of or lied to.

Jump into the digital world and add protocols, services, encryption, CAs, RAs, CRLs, and differing technologies and applications, and the business risks can become overwhelming and confusing.

So start with a basic question: What criteria will we use to determine who we trust and to what degree?

Slide 3



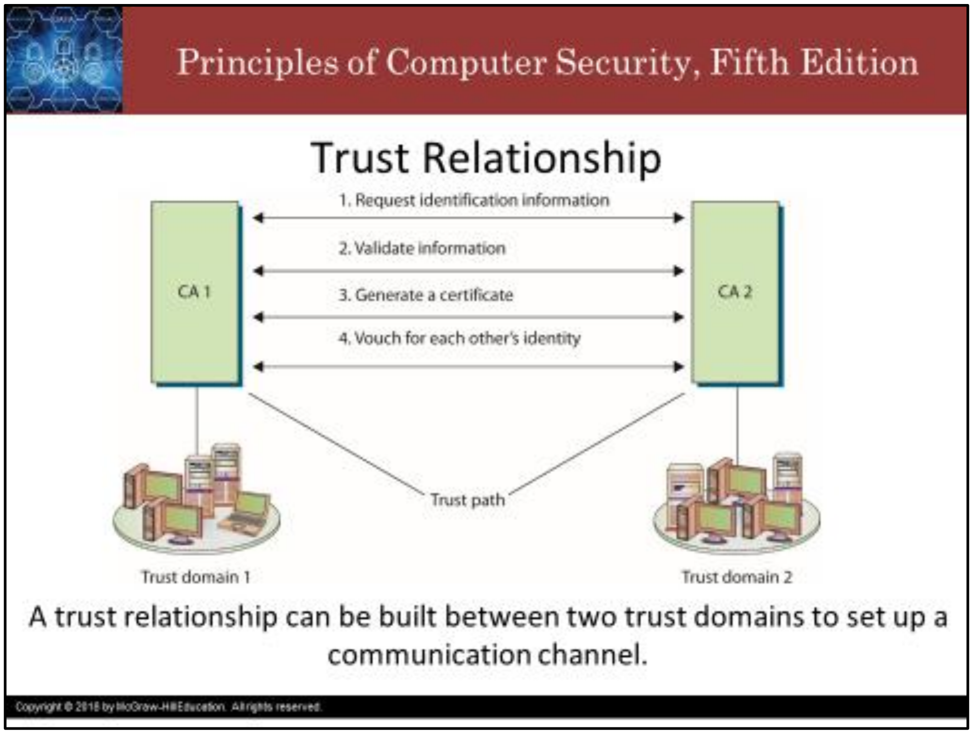
Principles of Computer Security, Fifth Edition

Trust Anchor

- Trusted third party.
  - Two separate trust domains involved if two companies need to communicate using their individual PKIs or two departments within the same company use different CAs.
    - The users and devices from these different trust domains need to communicate with each other.
    - They need to exchange certificates and public keys.
    - Trust anchors must be identified and a communication channel constructed and maintained.

A trust anchor is an agreed-upon trusted third party.
There are two separate trust domains involved when two companies need to communicate using their individual PKIs or two departments within the same company use different CAs.
The users and devices from these different trust domains need to communicate with each other and they need to exchange certificates and public keys.
Trust anchors must be identified and a communication channel constructed and maintained.

Slide 4



Principles of Computer Security, Fifth Edition

Trust Relationship

CA 1

1. Request identification information

2. Validate information

3. Generate a certificate

4. Vouch for each other's identity

CA 2

Trust path

Trust domain 1

Trust domain 2

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

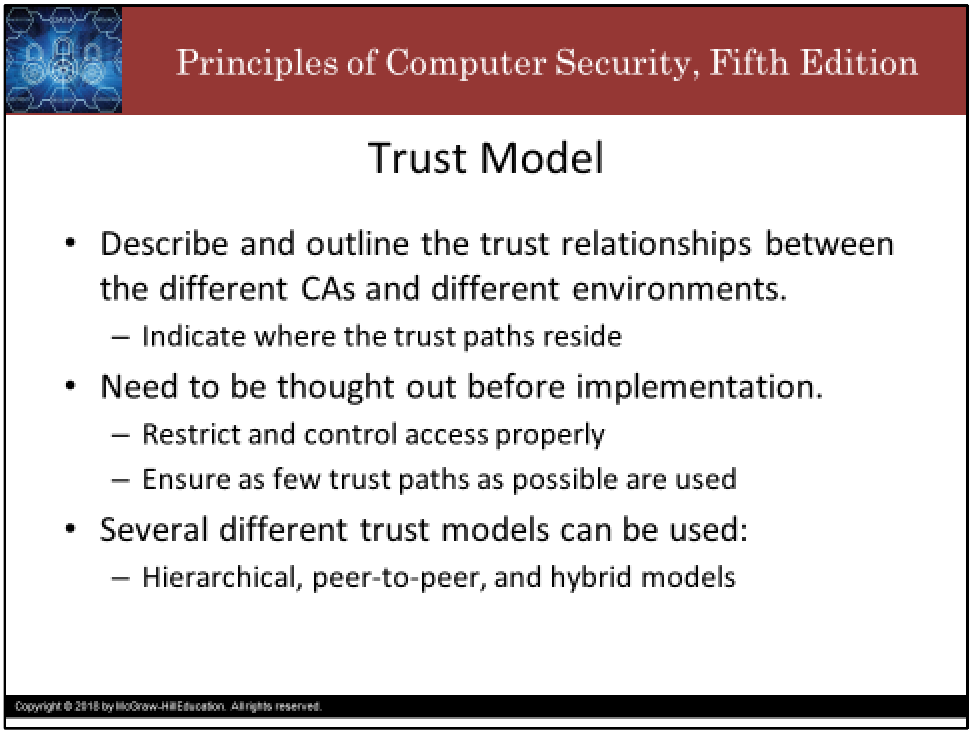
Often, a trust relationship needs to be established between two CAs. To do this, the CAs issue certificates for each other: CA 1 signs CA 2's public key and CA 2 signs CA 1's public key.

This establishes a trust path that can be used by users in one trust domain to verify certificates from the other trust domain.

Trust paths can be bidirectional or unidirectional (one CA trusts the other, but not vice versa).

In the figure, all the users and devices in trust domain 1 trust their own CA, CA 1, which is their trust anchor. All users and devices in trust domain 2 have their own trust anchor, CA 2. The two CAs have exchanged certificates and trust each other, but they do not have a common trust anchor between them.

Slide 5



Principles of Computer Security, Fifth Edition

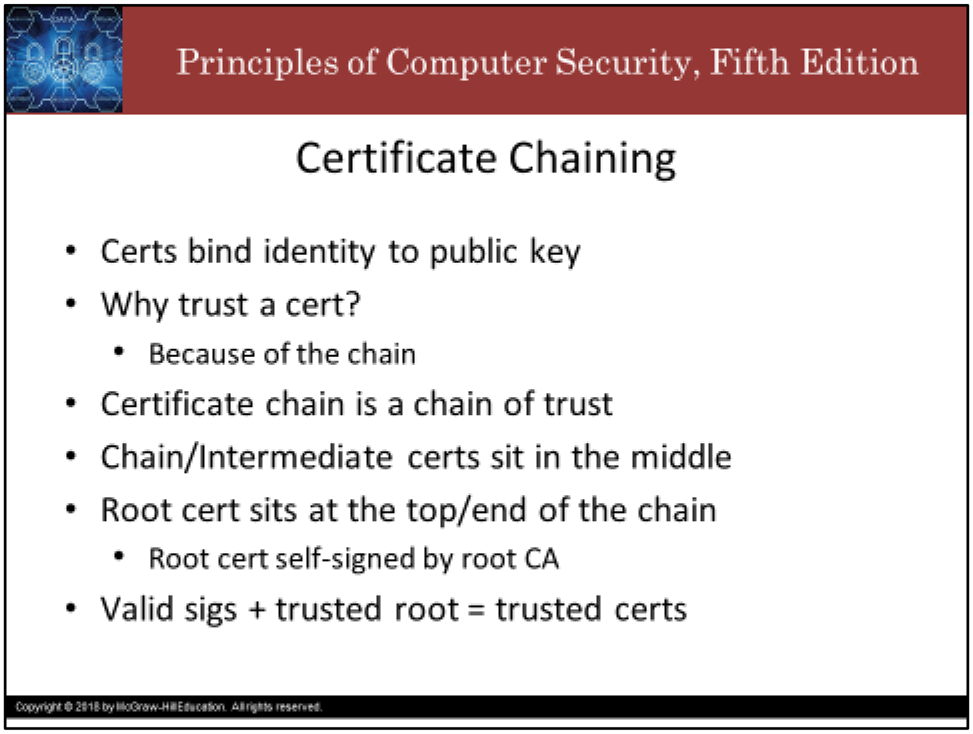
Trust Model

- Describe and outline the trust relationships between the different CAs and different environments.
  - Indicate where the trust paths reside
- Need to be thought out before implementation.
  - Restrict and control access properly
  - Ensure as few trust paths as possible are used
- Several different trust models can be used:
  - Hierarchical, peer-to-peer, and hybrid models

Trust models describe and outline the trust relationships between the different CAs and different environments, which will indicate where the trust paths reside.

The trust models and paths need to be thought out before implementation to restrict and control access properly and to ensure as few trust paths as possible are used
There are several forms of trust models associated with certificates. Hierarchical, peer-to-peer, and hybrid are the primary forms, with the web of trust being a form of hybrid. Each of these models has a useful place in the PKI architecture under different circumstances.

Slide 6



Principles of Computer Security, Fifth Edition

Certificate Chaining

- Certs bind identity to public key
- Why trust a cert?
  - Because of the chain
- Certificate chain is a chain of trust
- Chain/Intermediate certs sit in the middle
- Root cert sits at the top/end of the chain
  - Root cert self-signed by root CA
- Valid sigs + trusted root = trusted certs

Certificates are used to convey identity and public-key pairs to users. But why should we trust the certificate?

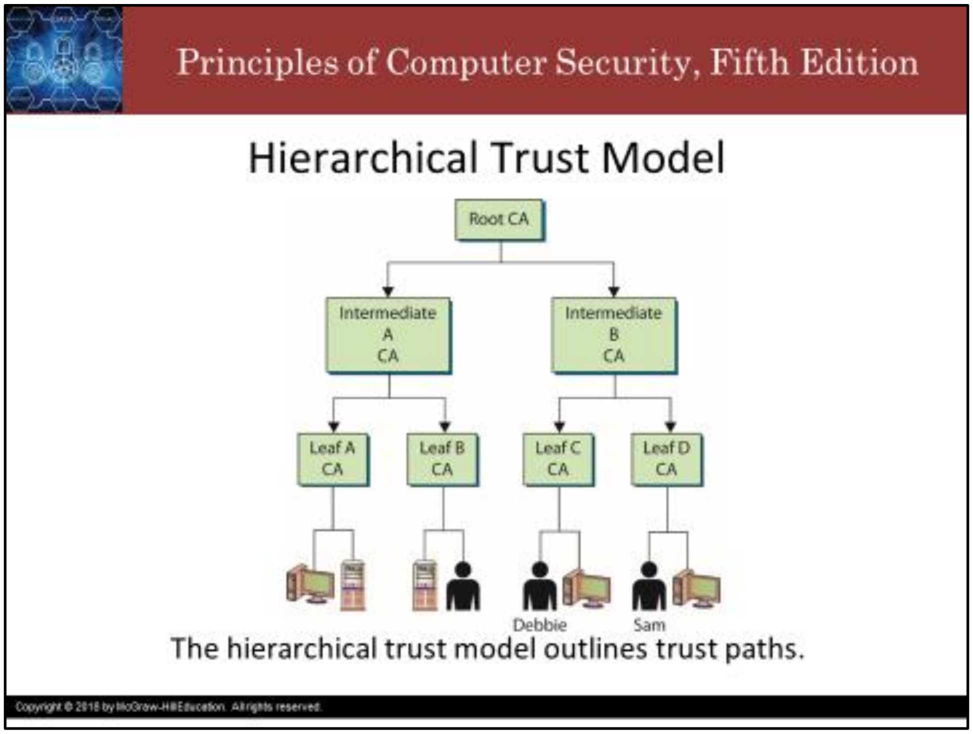
The answer lies in the certificate chain, a chain of trust from one certificate to another until the chain ends with a certificate that we trust.

The transitivity of trust from that certificate all the way down to the one we have in our proverbial hands is why we can trust it.

Certificates that sit between the presented certificate and the root certificate are called chain certificates. The chain CA is the signer of the presented certificate and shows that the chain CA trusts the presenter. The root CA is the signer of the chain certificate and shows that it trusts the chain CA. That relationship applies recursively (if there is a cert above the root, the root is actually a chain, and the old chain becomes the presenter) until it terminates with the final root CA. The root CA is always signed by the CA itself (a self-signed cert is generally not to be trusted unless the signer/self is a root of trust or trust anchor).

If all the signatures a valid and we trust the root CA, then we can trust all the certs all the way down.

Slide 7

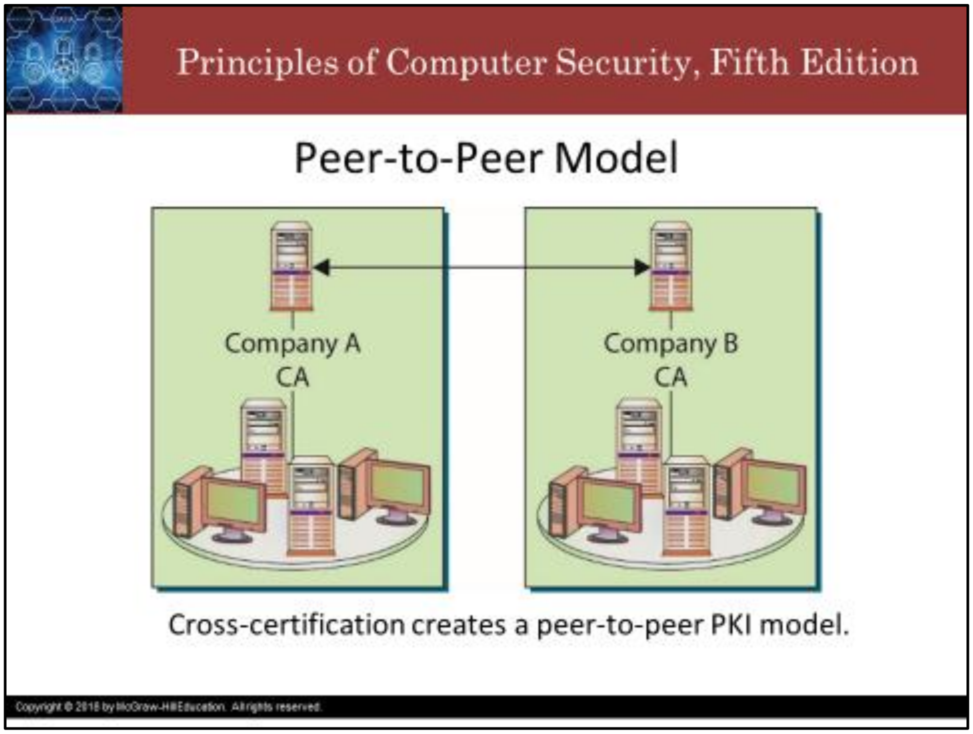

Principles of Computer Security, Fifth Edition

Hierarchical Trust Model

The hierarchical trust model outlines trust paths.

The **hierarchical trust model** is a basic hierarchical structure that contains a root CA, intermediate CAs, leaf CAs, and end-entities.
The structure is a tree.
The root CA is the ultimate trust anchor for all other entities in the tree.
The root CA generates certificates for the intermediate CAs, which in turn generate certificates for the leaf CAs, which generate certificates for the end-point entities.
The intermediate CAs function to transfer trust between different CAs. These CAs are referred to as *subordinate CAs* because they are subordinate to the CA that they reference. The path of trust is walked up from the subordinate CA to the higher-level CA; in essence the subordinate CA is using the higher-level CA as a reference.
In this model, no bidirectional trusts exist—they are all unidirectional trusts, as indicated by the one-way arrows. Since no other entity can certify and generate certificates for the root CA, it creates a self-signed certificate. This means that the certificate’s Issuer and Subject fields hold the same information, both representing the root CA, and the root CA’s public key will be used to verify this certificate when that time comes. This root CA certificate and public key are distributed to all entities.

Slide 8



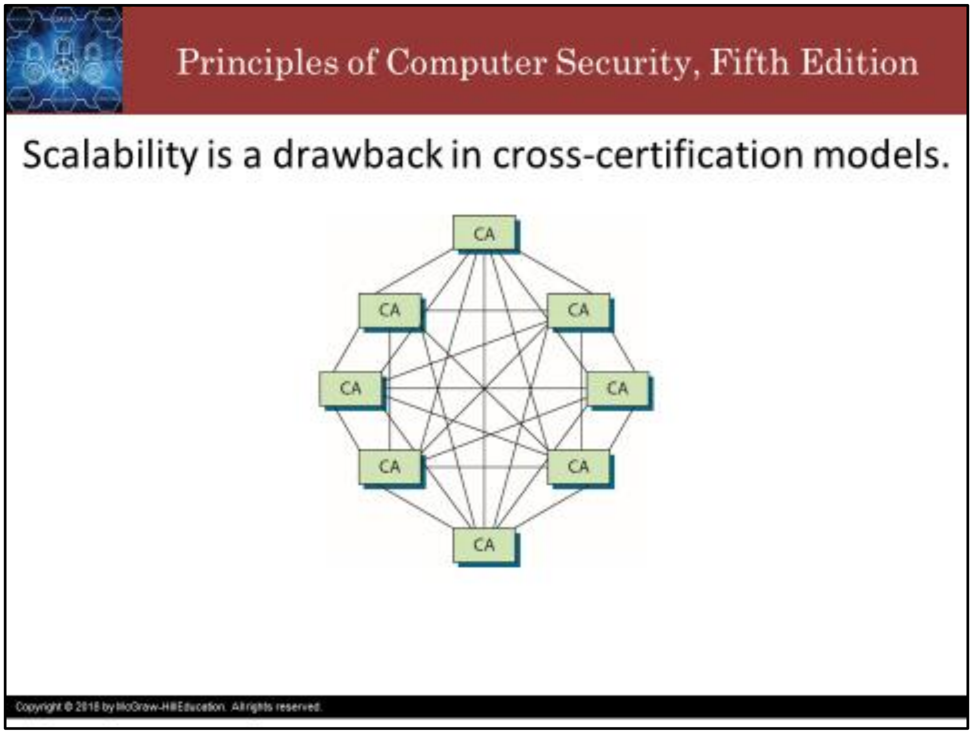
Principles of Computer Security, Fifth Edition

Peer-to-Peer Model

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

In a **peer-to-peer trust model**, one CA is not subordinate to another CA, and no established trusted anchor between the CAs is involved.
The end-entities look to their issuing CA as their trusted anchor, but different CAs will not have a common anchor.
The two different CAs will certify the public key for each other, which creates a bidirectional trust.
This is referred to as *cross-certification*, since the CAs are not receiving their certificates and public keys from a superior CA, but instead are creating them for each other.

Slide 9



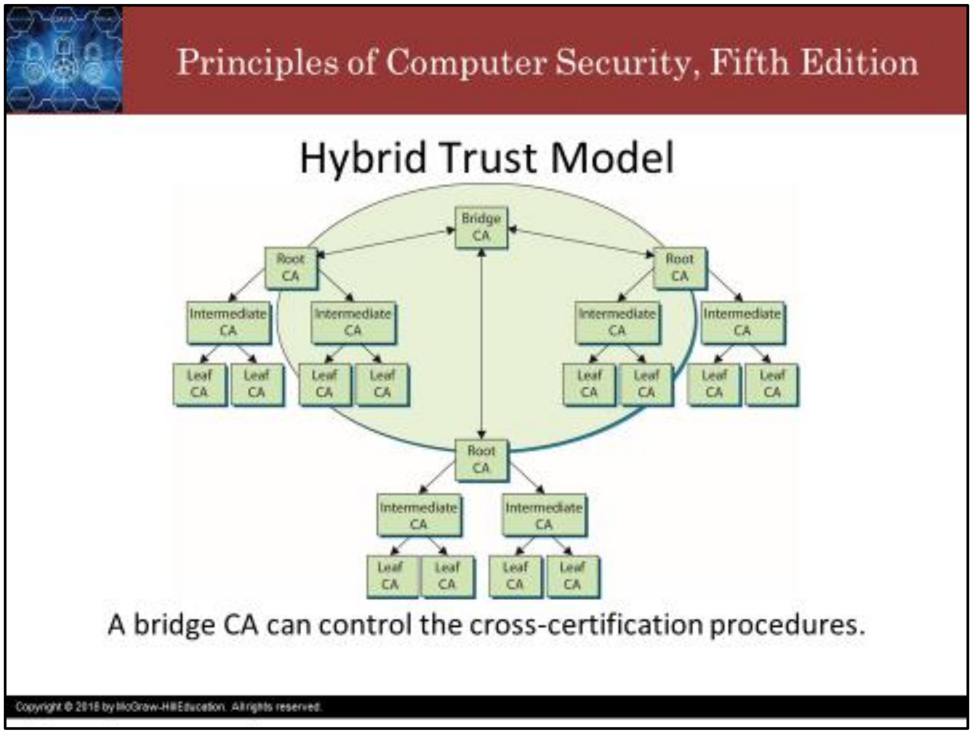
Principles of Computer Security, Fifth Edition

Scalability is a drawback in cross-certification models.

One of the main drawbacks to the peer-to-peer model is scalability.
Each CA must certify every other CA that is participating, and a bidirectional trust path must be implemented.
If one root CA were certifying all the intermediate CAs, scalability would not be as much of an issue.

The figure shows a fully connected *mesh architecture*, meaning that each CA is directly connected to and has a bidirectional trust relationship with every other CA. As you can see, the complexity of this setup can become overwhelming. If there are N CAs (corresponding to N peers), there will be on the order of N squared trust paths, since each CA has N-1 different certs signed by each of the N-1 other CAs.

Slide 10



Principles of Computer Security, Fifth Edition

Hybrid Trust Model

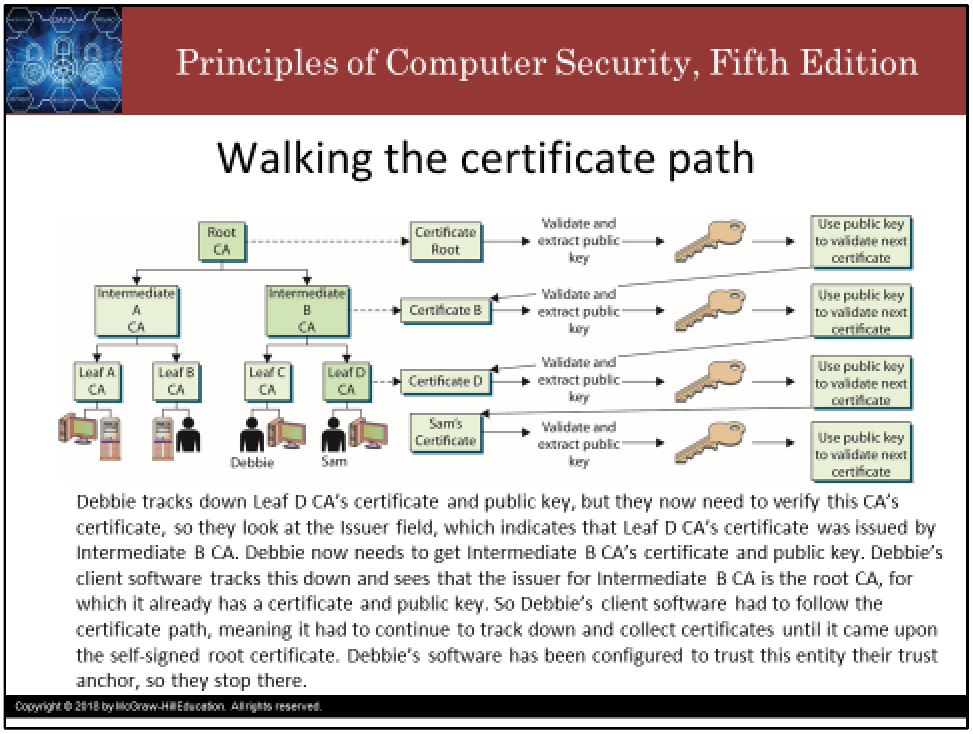
A bridge CA can control the cross-certification procedures.

In a **hybrid trust model**, the two companies have their own internal hierarchical trust models and are connected through a peer-to-peer model using cross-certification.

Another option is to implement a centralized 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.

Slide 11



Principles of Computer Security, Fifth Edition

Walking the certificate path

Root CA
Certificate Root
Validate and extract public key
Use public key to validate next certificate
Intermediate A CA
Intermediate B CA
Certificate B
Validate and extract public key
Use public key to validate next certificate
Leaf A CA
Leaf B CA
Leaf C CA
Leaf D CA
Certificate D
Validate and extract public key
Use public key to validate next certificate
Sam's Certificate
Validate and extract public key
Use public key to validate next certificate
Debbie
Sam

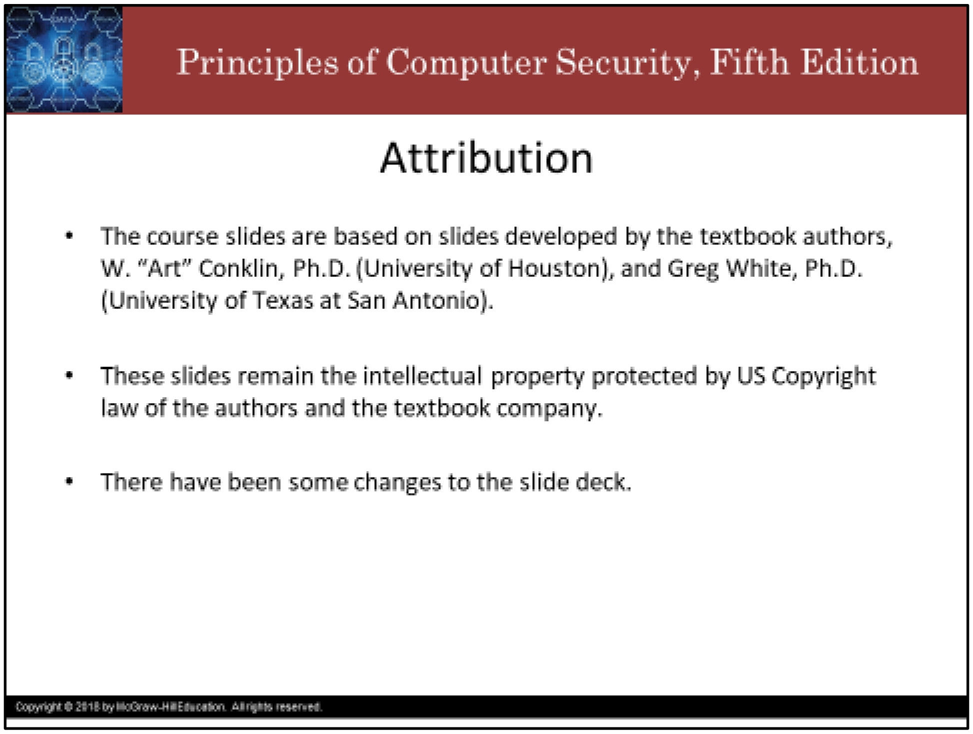
Debbie tracks down Leaf D CA's certificate and public key, but they now need to verify this CA's certificate, so they look at the Issuer field, which indicates that Leaf D CA's certificate was issued by Intermediate B CA. Debbie now needs to get Intermediate B CA's certificate and public key. Debbie's client software tracks this down and sees that the issuer for Intermediate B CA is the root CA, for which it already has a certificate and public key. So Debbie's client software had to follow the certificate path, meaning it had to continue to track down and collect certificates until it came upon the self-signed root certificate. Debbie's software has been configured to trust this entity their trust anchor, so they stop there.

When a user in one trust domain needs to communicate with a user in another trust domain, one user will need to validate the other's certificate. This sounds simple enough, but what it really means is that each certificate for each CA, all the way up to a shared trusted anchor, also must be validated.

Following the **certificate path** means the client software has to continue to track down and collect certificates until it reaches the root certificate, which, by definition, is self-signed. If there is no root, then there is a loop in the chain and it probably shouldn't be trusted.

This type of simplistic trust model works well within an enterprise that easily follows a hierarchical organizational chart, but many companies cannot use this type of trust model because different departments or offices require their own trust anchors. These demands can be derived from direct business needs or from interorganizational politics. This hierarchical model might not be possible when two or more companies need to communicate with each other. Neither company will let the other's CA be the root CA, because each does not necessarily trust the other entity to that degree. In these situations, the CAs will need to work in a peer-to-peer relationship instead of in a hierarchical relationship.

Slide 12



Principles of Computer Security, Fifth Edition

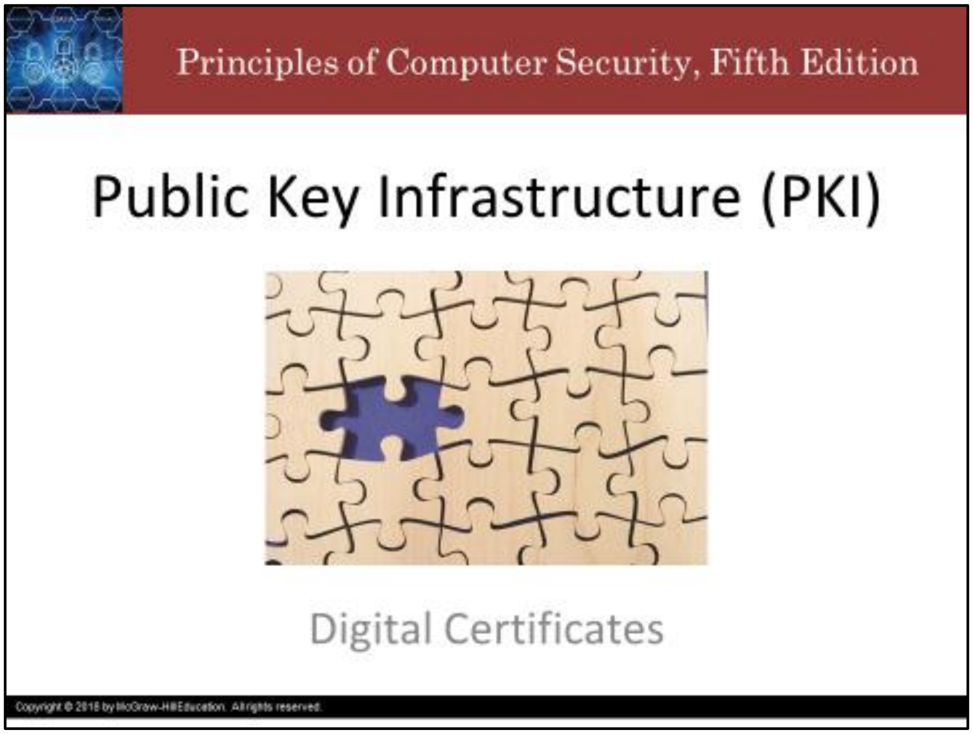
Attribution

- The course slides are based on slides developed by the textbook authors, W. “Art” Conklin, Ph.D. (University of Houston), and Greg White, Ph.D. (University of Texas at San Antonio).
- These slides remain the intellectual property protected by US Copyright law of the authors and the textbook company.
- There have been some changes to the slide deck.

Thank you and take care.

PKI: Digital Certificates

Slide 1



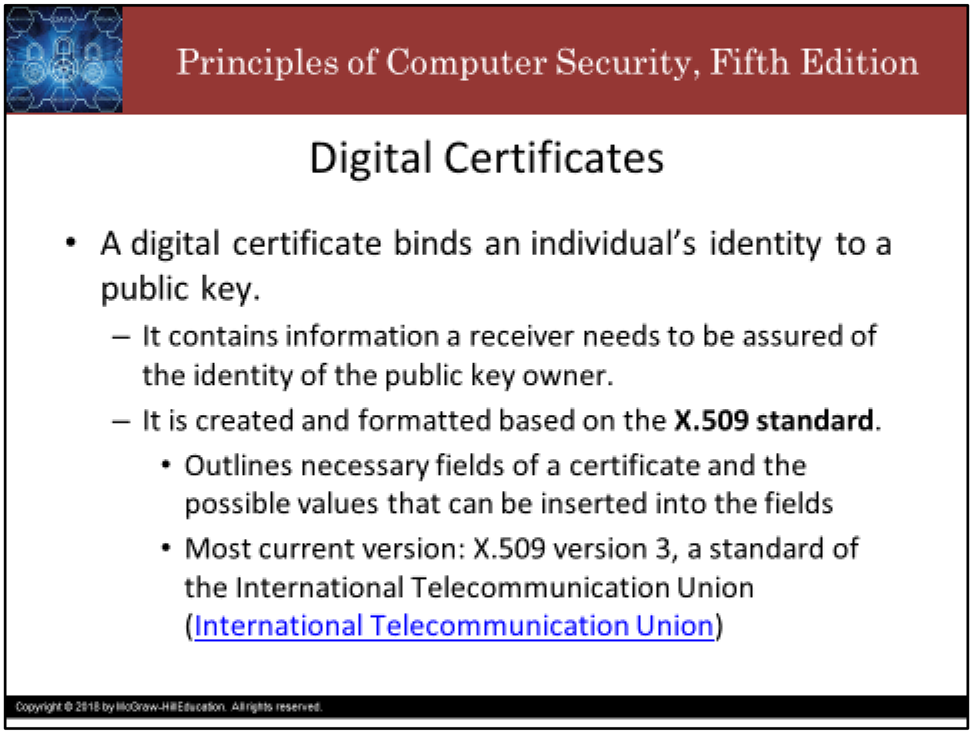
Principles of Computer Security, Fifth Edition

Public Key Infrastructure (PKI)

Digital Certificates

Howdy! In this video, we discuss digital certificates.

Slide 2



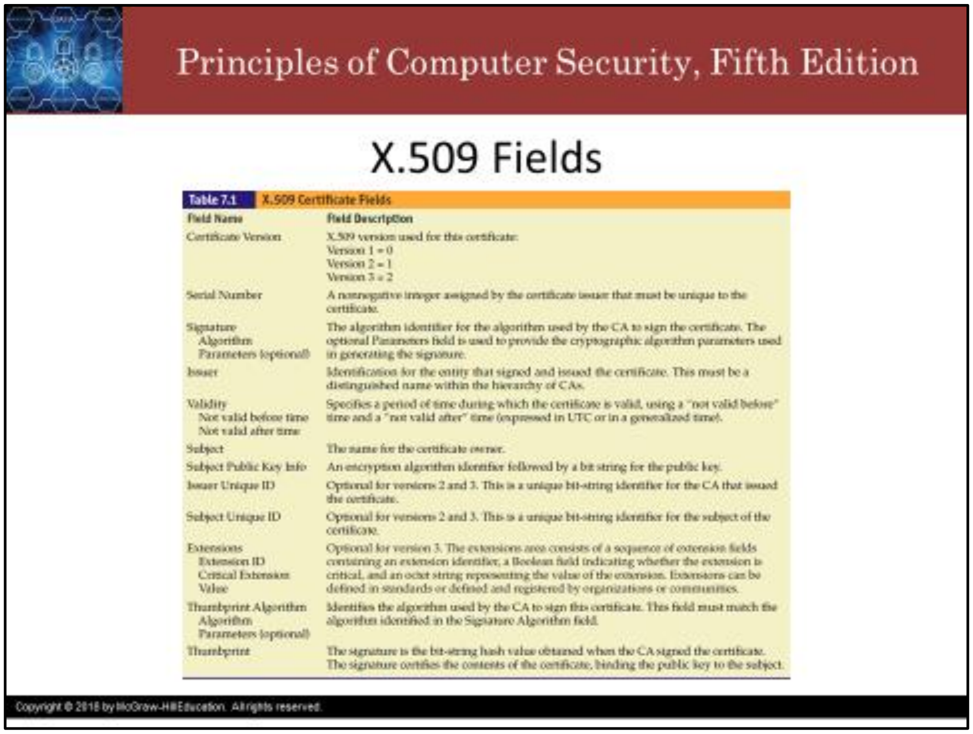
Principles of Computer Security, Fifth Edition

Digital Certificates

- A digital certificate binds an individual's identity to a public key.
  - It contains information a receiver needs to be assured of the identity of the public key owner.
  - It is created and formatted based on the **X.509 standard**.
    - Outlines necessary fields of a certificate and the possible values that can be inserted into the fields
    - Most current version: X.509 version 3, a standard of the International Telecommunication Union (International Telecommunication Union)

A digital certificate binds an individual's identity to a public key.
It contains the information a receiver needs to be assured of the identity of the public key owner.
Digital certificates are created and formatted based on the **X.509 standard** which outlines the necessary fields of a certificate and the possible values that can be inserted into the fields
The most current version is X.509 version 3
The subject of a certificate is commonly a person, but it does not have to be. The subject can also be a network device (router, web server, firewall, and so on), an application, a department, or a company. Each has its own identity that needs to be verified and proven to another entity before secure, trusted communication can be initiated. If a network device is using a certificate for authentication, the certificate may contain the identity of that device. This allows a user of the device to verify its authenticity based on the signed certificate and trust in the signing authority. This trust can be transferred to the identity of the device indicating authenticity.

Slide 3



Principles of Computer Security, Fifth Edition

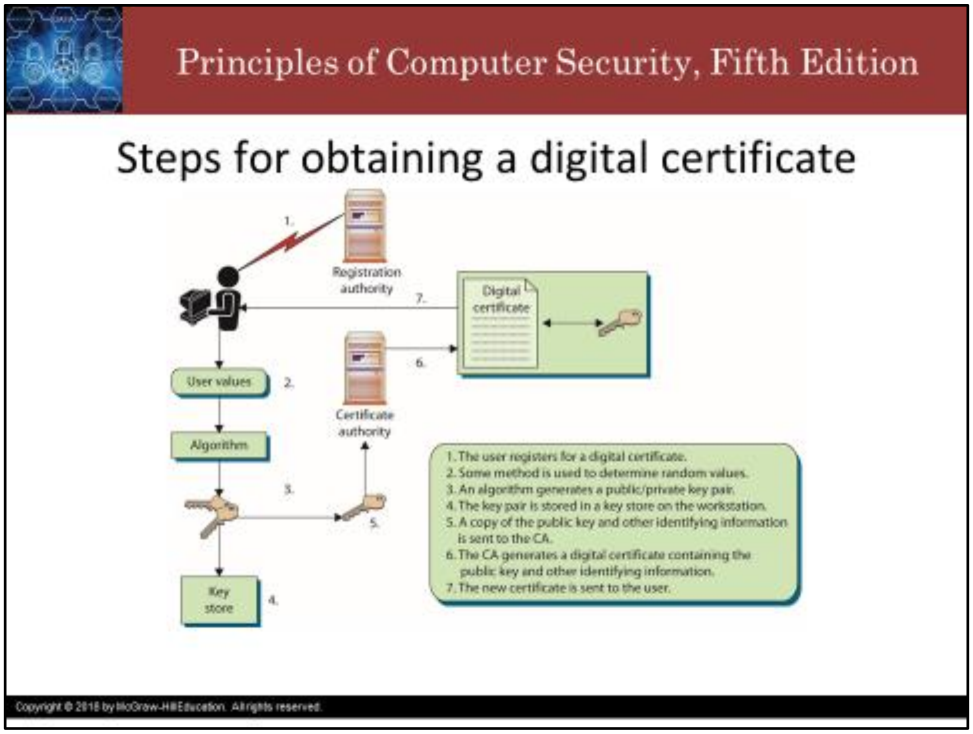
X.509 Fields

Table 7.1 X.509 Certificate Fields

Field Name	Field Description
Certificate Version	X.509 version used for this certificate: Version 1 = 0 Version 2 = 1 Version 3 = 2
Serial Number	A nonnegative integer assigned by the certificate issuer that must be unique to the certificate.
Signature Algorithm Parameters (optional)	The algorithm identifier for the algorithm used by the CA to sign the certificate. The optional Parameters field is used to provide the cryptographic algorithm parameters used in generating the signature.
Issuer	Identification for the entity that signed and issued the certificate. This must be a distinguished name within the hierarchy of CAs.
Validity Not valid before time Not valid after time	Specifies a period of time during which the certificate is valid, using a "not valid before" time and a "not valid after" time (expressed in UTC or in a generalized time).
Subject	The name for the certificate owner.
Subject Public Key Info	An encryption algorithm identifier followed by a bit string for the public key.
Issuer Unique ID	Optional for versions 2 and 3. This is a unique bit-string identifier for the CA that issued the certificate.
Subject Unique ID	Optional for versions 2 and 3. This is a unique bit-string identifier for the subject of the certificate.
Extensions Extension ID Critical Extension Value	Optional for version 3. The extensions area consists of a sequence of extension fields containing an extension identifier, a Boolean field indicating whether the extension is critical, and an octet string representing the value of the extension. Extensions can be defined in standards or defined and registered by organizations or communities.
Thumbprint Algorithm Algorithm Parameters (optional)	Identifies the algorithm used by the CA to sign this certificate. This field must match the algorithm identified in the Signature Algorithm field.
Thumbprint	The signature is the bit-string hash value obtained when the CA signed the certificate. The signature certifies the contents of the certificate, binding the public key to the subject.

This table from the textbook shows the different fields in an X.509 certificate. Every certificate has
A Version Number,
A Serial Number,
The Algorithm theca used to sign the cert
The name of the CA that issued the cert
The start and end of the Validity period
The Subject's name and Public Key Info, which includes their Public Key Algorithm and the Public Key
And the CA's signature

Slide 4



Principles of Computer Security, Fifth Edition

Steps for obtaining a digital certificate

1. The user registers for a digital certificate.
2. Some method is used to determine random values.
3. An algorithm generates a public/private key pair.
4. The key pair is stored in a key store on the workstation.
5. A copy of the public key and other identifying information is sent to the CA.
6. The CA generates a digital certificate containing the public key and other identifying information.
7. The new certificate is sent to the user.

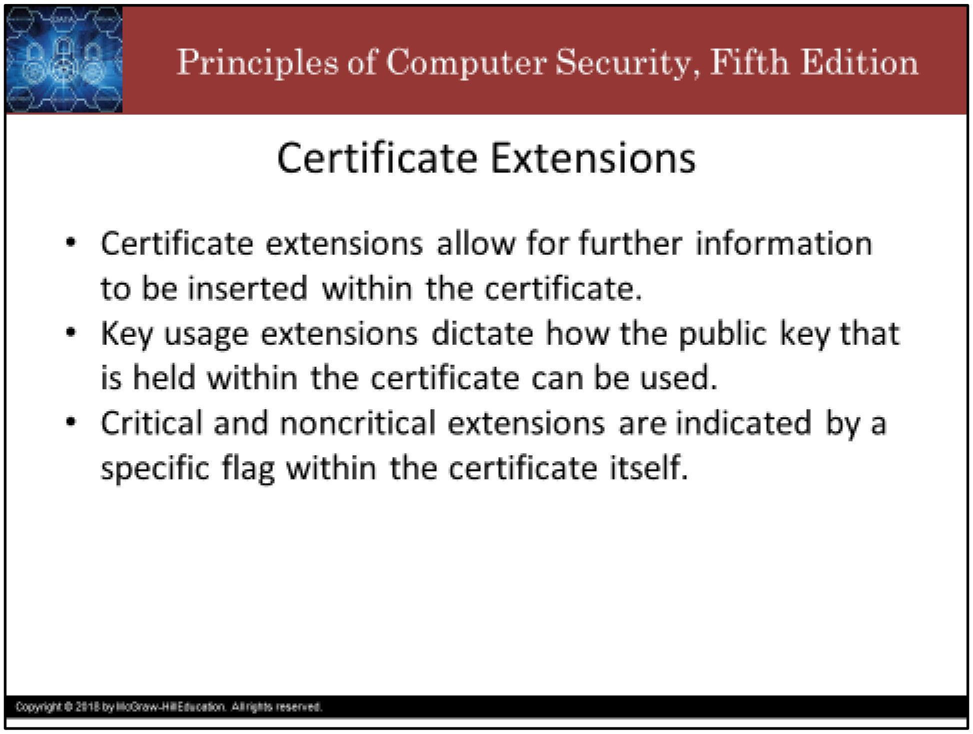
There are different classes of certificates which convey varying levels of trust and power. The higher the class, the more trusted and powerful the cert. A class 1 certificate is the lowest classification and is used for personal mail encryption and signatures.
In most situations, when a user requests a Class 1 certificate, the registration process will require the user to enter specific information into a web-based form. The web page will have a section that accepts the user’s public key, or it will step the user through creating a public/private key pair, which will allow the user to choose the size of the keys to be created.

Once these steps have been completed, the public key is attached to the certificate registration form and both are forwarded to the RA for processing. The RA is responsible only for the registration process and cannot actually generate a certificate.

Once the RA is finished processing the request and verifying the individual’s identity, the RA sends the request to the CA. The CA uses the RA-provided information to generate a digital certificate, integrates the necessary data into the certificate fields (user identification information, public key, validity dates, proper use for the key and certificate, and so on), and sends a copy of the certificate to the user.

The certificate may also be posted to a publicly accessible directory so that others can access it.

Slide 5



Principles of Computer Security, Fifth Edition

Certificate Extensions

- Certificate extensions allow for further information to be inserted within the certificate.
- Key usage extensions dictate how the public key that is held within the certificate can be used.
- Critical and noncritical extensions are indicated by a specific flag within the certificate itself.

Certificate extensions allow for further information to be inserted within the certificate.

Extensions provide additional functionality in a PKI implementation.

Standard certificate extensions are those implemented for every PKI implementation.

Private certificate extensions are defined for specific organizations and they allow companies to further define different, specific uses for digital certificates that best fit their business needs.

Key usage extensions dictate how the public key that is held within the certificate can be used, such as key exchange, data encryption, and digital signatures.

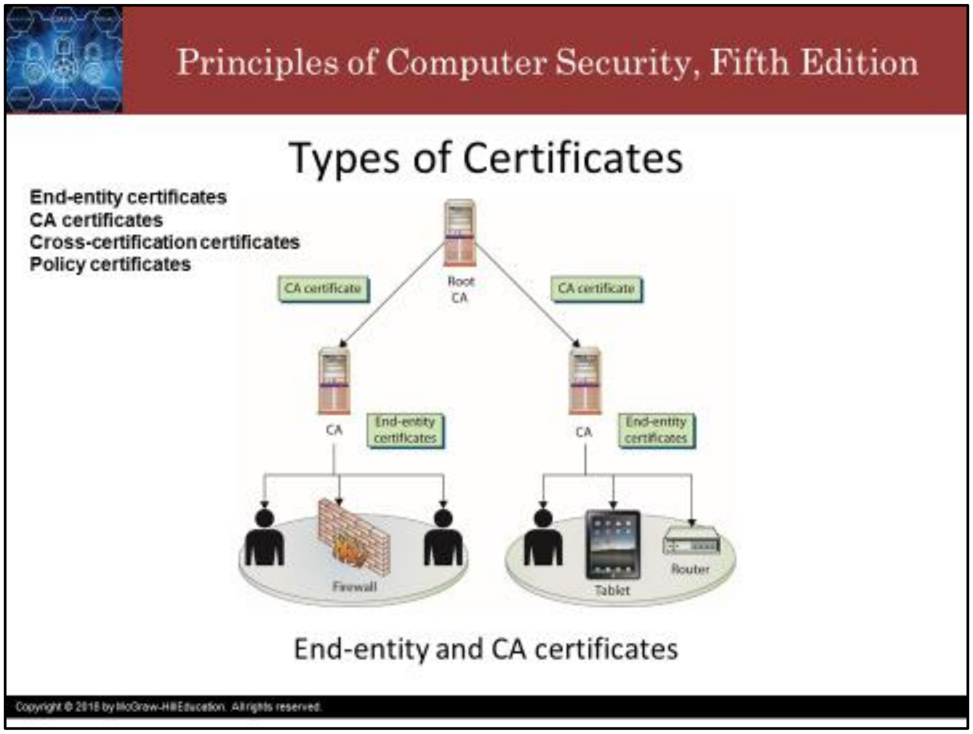
Whether an extension is critical is indicated by a specific flag within the certificate itself.

If the flag set to *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.

If the flag set to *noncritical*, then the certificate can be used for the intended purpose, even if the receiver does not process the appended extension.

Slide 6



Principles of Computer Security, Fifth Edition

Types of Certificates

End-entity certificates
CA certificates
Cross-certification certificates
Policy certificates

End-entity and CA certificates

Four main certificate types of certificates are used:

End-entity certificates

CA certificates

Cross-certification certificates

Policy certificates

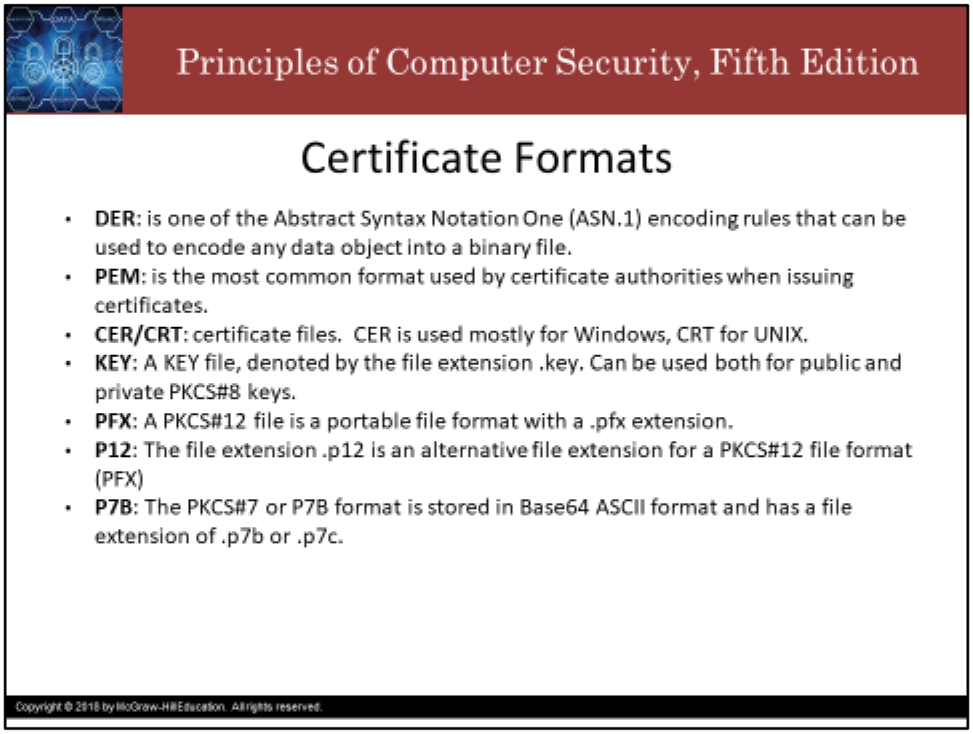
End-entity certificates are issued by a CA to a specific subject, such as Joyce, the Accounting department, or a firewall. An end-entity certificate is the identity document provided by PKI implementations.

A **CA certificate** can be self-signed, in the case of a standalone or root CA, or it can be issued by a superior CA within a hierarchical model. In the Figure, the superior CA gives the authority and allows the subordinate CA to accept certificate requests and generate the individual certificates itself. This may be necessary when a company needs to have multiple internal CAs, and different departments within an organization need to have their own CAs servicing their specific end-entities in their sections. In these situations, a representative from each department requiring a CA registers with the higher trusted CA and requests a Certificate Authority certificate. (Public and private CAs are discussed in the “Public Certificate Authorities” and “In-House Certificate Authorities” sections later in this chapter, as are the different trust models that are available for companies.)

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

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**.

Slide 7



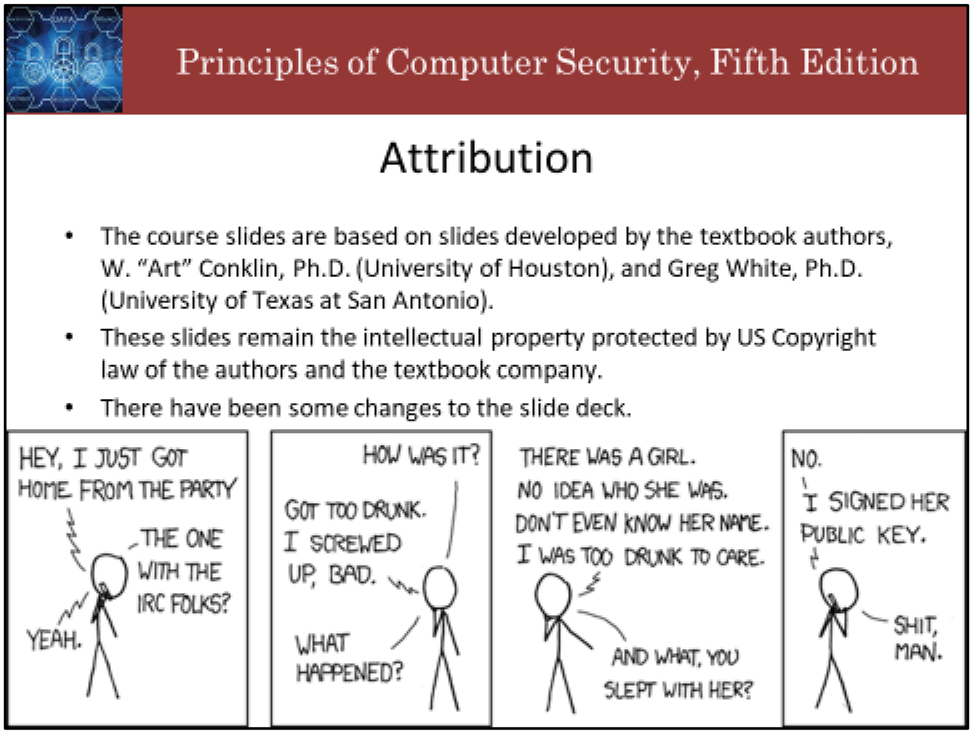
Principles of Computer Security, Fifth Edition

Certificate Formats

- **DER**: is one of the Abstract Syntax Notation One (ASN.1) encoding rules that can be used to encode any data object into a binary file.
- **PEM**: is the most common format used by certificate authorities when issuing certificates.
- **CER/CRT**: certificate files. CER is used mostly for Windows, CRT for UNIX.
- **KEY**: A KEY file, denoted by the file extension .key. Can be used both for public and private PKCS#8 keys.
- **PFX**: A PKCS#12 file is a portable file format with a .pfx extension.
- **P12**: The file extension .p12 is an alternative file extension for a PKCS#12 file format (PFX)
- **P7B**: The PKCS#7 or P7B format is stored in Base64 ASCII format and has a file extension of .p7b or .p7c.

There are many different file formats for X.509 certificates. The most common of these is P E M or “pem”, which used to stand for “privacy enhanced mail”, but now, like the A and M in Texas A&M, it’s just stands for P E M now. The different formats are not interchangeable, but there do exist programs that can convert between certain formats.

Slide 8



Principles of Computer Security, Fifth Edition

Attribution

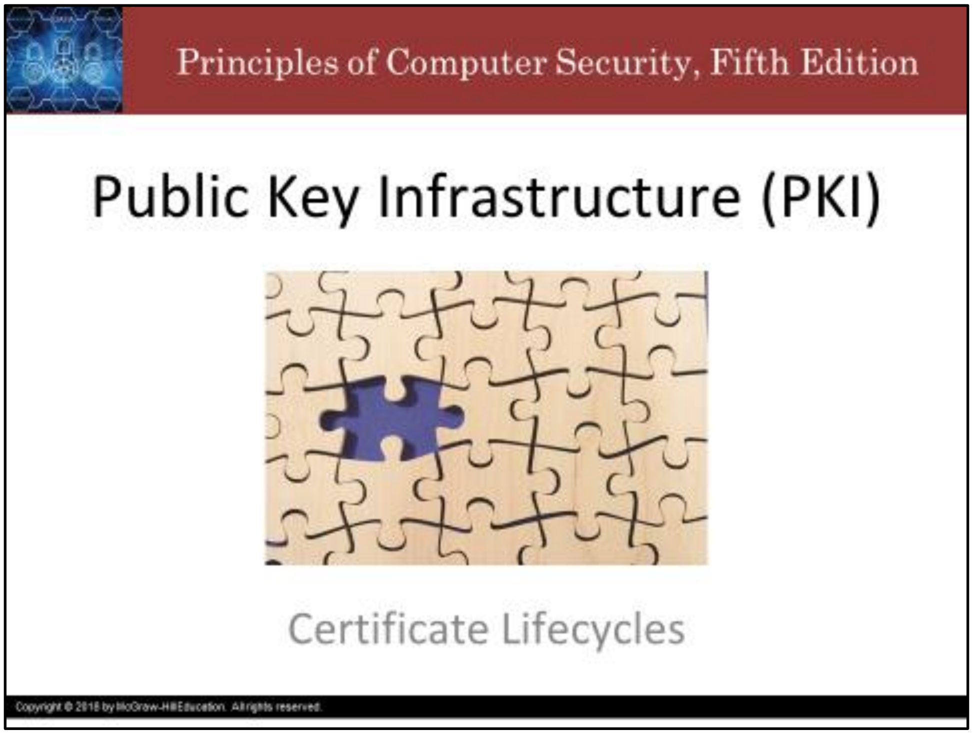
- The course slides are based on slides developed by the textbook authors, W. “Art” Conklin, Ph.D. (University of Houston), and Greg White, Ph.D. (University of Texas at San Antonio).
- These slides remain the intellectual property protected by US Copyright law of the authors and the textbook company.
- There have been some changes to the slide deck.

HEY, I JUST GOT HOME FROM THE PARTY
THE ONE WITH THE IRC FOLKS?
YEAH.
HOW WAS IT?
GOT TOO DRUNK. I SCREWED UP, BAD.
WHAT HAPPENED?
THERE WAS A GIRL. NO IDEA WHO SHE WAS. DON'T EVEN KNOW HER NAME. I WAS TOO DRUNK TO CARE.
AND WHAT, YOU SLEPT WITH HER?
NO.
I SIGNED HER PUBLIC KEY.
SHIT, MAN.

Thank you and take care.

PKI: Certificate Lifecycles

Slide 1



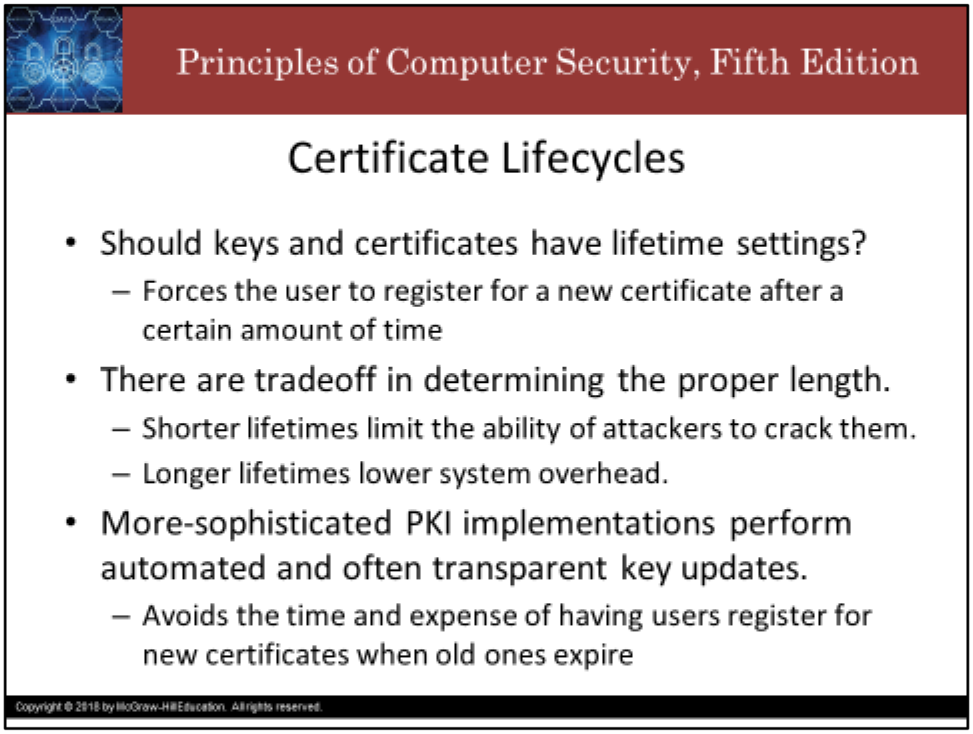
Principles of Computer Security, Fifth Edition

Public Key Infrastructure (PKI)

Certificate Lifecycles

Howdy! In this video, we discuss certificate lifetimes.

Slide 2



Principles of Computer Security, Fifth Edition

Certificate Lifecycles

- Should keys and certificates have lifetime settings?
  - Forces the user to register for a new certificate after a certain amount of time
- There are tradeoff in determining the proper length.
  - Shorter lifetimes limit the ability of attackers to crack them.
  - Longer lifetimes lower system overhead.
- More-sophisticated PKI implementations perform automated and often transparent key updates.
  - Avoids the time and expense of having users register for new certificates when old ones expire

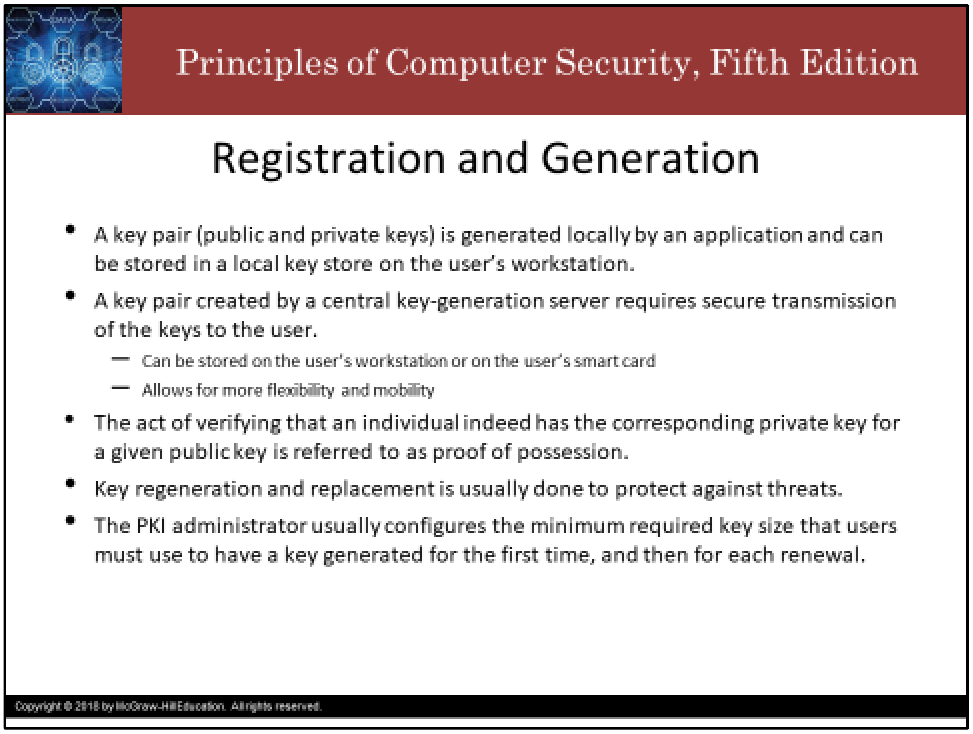
Should keys and certificates have a set lifetime?
This forces the user to register for a new certificate after a certain amount of time.
Companies that collect money for issuing certs want to issue lots of them.

There are tradeoff in determining the proper length of a certificates lifetime.

Shorter lifetimes limit the ability of attackers to crack them.

Longer lifetimes lower system overhead.
More-sophisticated PKI implementations perform automated and often transparent key updates to avoid the time and expense of having users register for new certificates when old ones expire. This means that the certificate and key pair has a lifecycle that must be managed. Certificate management involves administrating and managing each of these phases, including registration, certificate and key generation, renewal, and revocation. Additional management functions include CRL distribution, certificate suspension, and key destruction.

Slide 3



Principles of Computer Security, Fifth Edition

Registration and Generation

- A key pair (public and private keys) is generated locally by an application and can be stored in a local key store on the user's workstation.
- A key pair created by a central key-generation server requires secure transmission of the keys to the user.
  - Can be stored on the user's workstation or on the user's smart card
  - Allows for more flexibility and mobility
- The act of verifying that an individual indeed has the corresponding private key for a given public key is referred to as proof of possession.
- Key regeneration and replacement is usually done to protect against threats.
- The PKI administrator usually configures the minimum required key size that users must use to have a key generated for the first time, and then for each renewal.

A key pair, which consists of public and private keys, is generated locally by an application and can be stored in a local key store on the user's workstation.

A key pair created by a central key-generation server requires secure transmission of the keys to the user and can be stored on the user's workstation or on the user's smart card, which allows for more flexibility and mobility

Not all public/private key pairs can be used for digital signatures, so asking the individual to sign a message and return it to prove that she has the necessary private key will not always work. If a key pair is used for encryption, the RA can send a challenge value to the individual, who, in turn, can use her private key to encrypt that value and return it to the RA. If the RA can successfully decrypt this value with the public key that was provided earlier, the RA can be confident that the individual has the necessary private key and can continue through the rest of the registration phase.

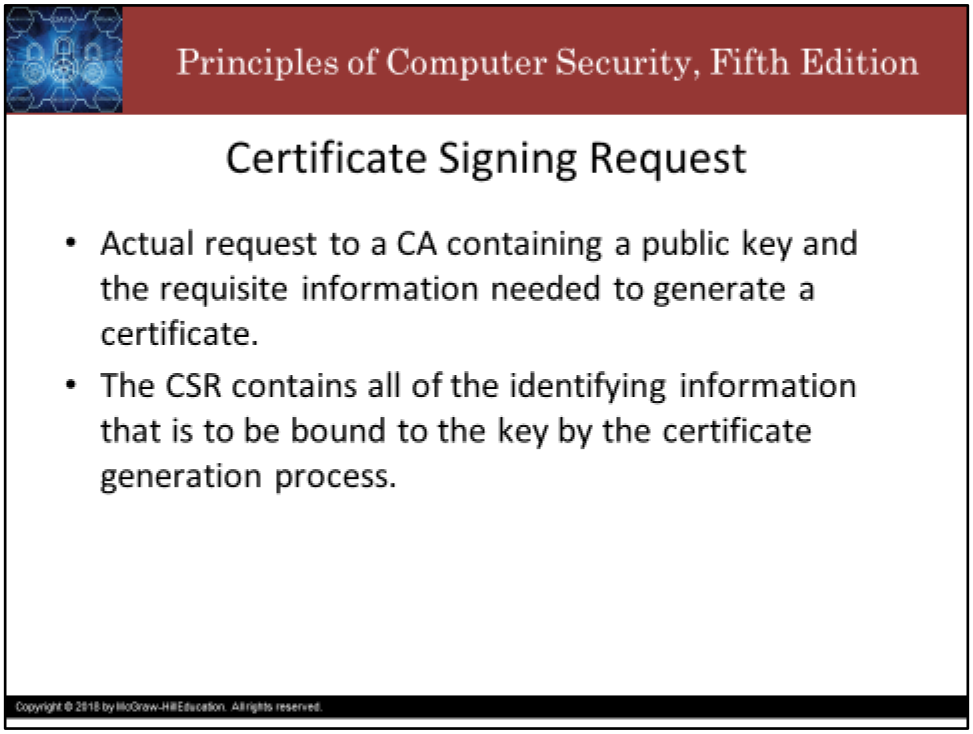
The act of verifying that an individual has the corresponding private key for a given public key is referred to as proof of possession.

Key regeneration and replacement is usually done to protect against threats such as weak or compromised keys, but is also done when the private key is lost, or corrupted.

As the processing power of computers increases and our knowledge of cryptography and new possible cryptanalysis-based attacks expands, key lifetimes may drastically decrease. As with everything within the security field, it is better to be safe now than to be surprised later and sorry.

The PKI administrator usually configures the minimum required key size that users must use to have a key generated for the first time, and then for each renewal. In most applications, there is a drop-down list of possible algorithms to choose from, and possible key sizes. The key size should provide the necessary level of security for the current environment. The lifetime of the key should be long enough that continual renewal will not negatively affect productivity, but short enough to ensure that the key cannot be successfully compromised.

Slide 4



Principles of Computer Security, Fifth Edition

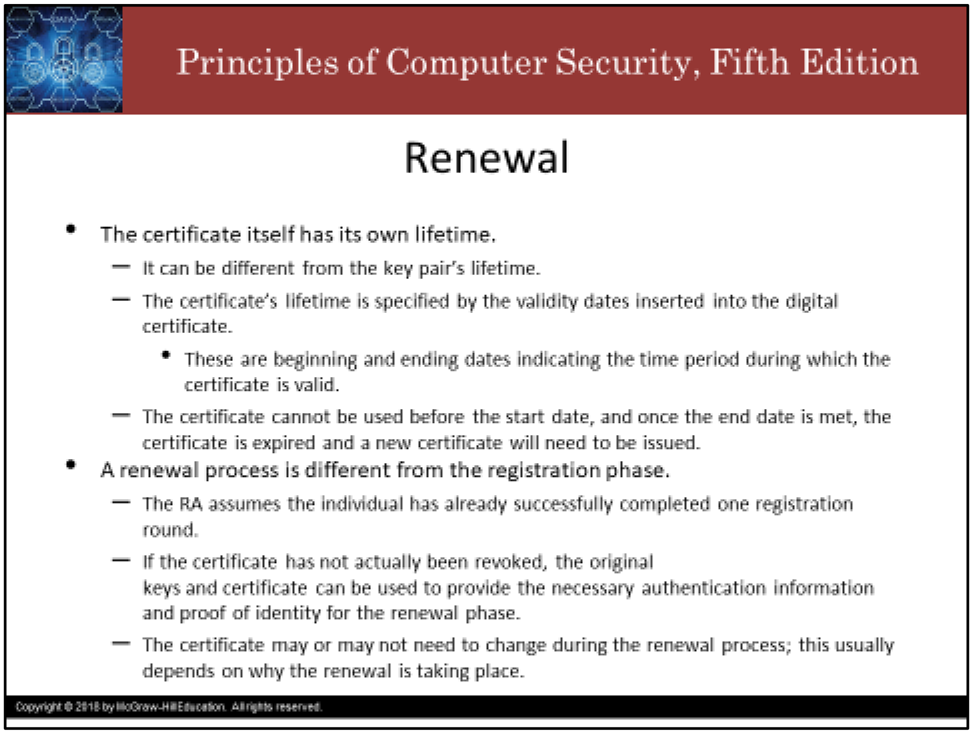
Certificate Signing Request

- Actual request to a CA containing a public key and the requisite information needed to generate a certificate.
- The CSR contains all of the identifying information that is to be bound to the key by the certificate generation process.

A **certificate signing request (CSR)** is the actual request to a CA containing a public key and the requisite information needed to generate a certificate.

The CSR contains all of the identifying information that is to be bound to the key by the certificate generation process.

Slide 5



Principles of Computer Security, Fifth Edition

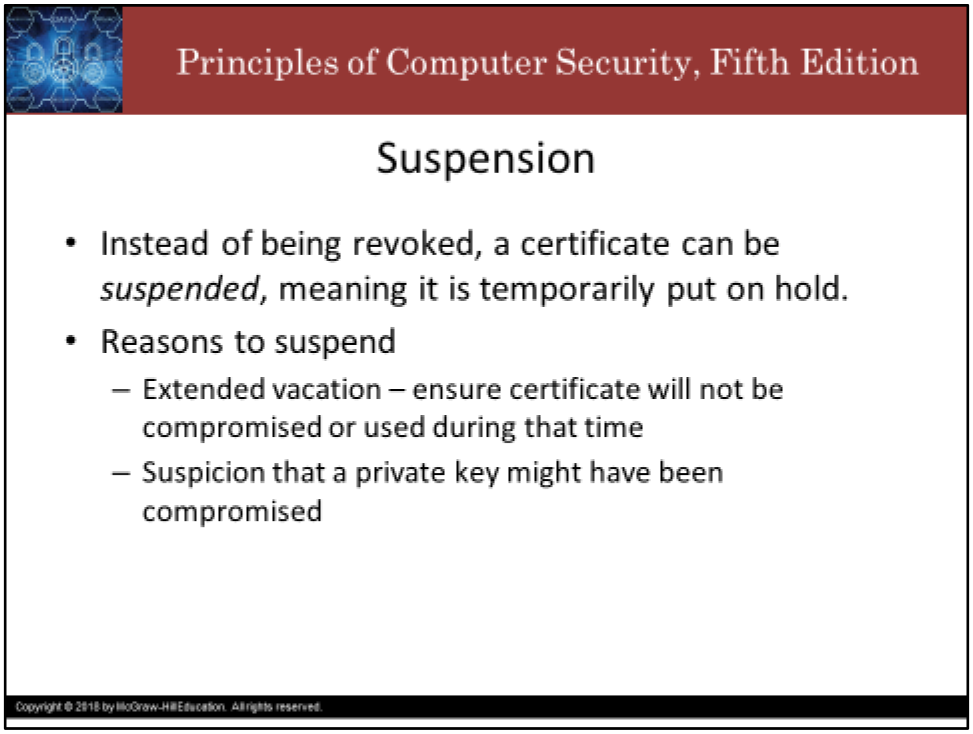
Renewal

- The certificate itself has its own lifetime.
  - It can be different from the key pair's lifetime.
  - The certificate's lifetime is specified by the validity dates inserted into the digital certificate.
    - These are beginning and ending dates indicating the time period during which the certificate is valid.
  - The certificate cannot be used before the start date, and once the end date is met, the certificate is expired and a new certificate will need to be issued.
- A renewal process is different from the registration phase.
  - The RA assumes the individual has already successfully completed one registration round.
  - If the certificate has not actually been revoked, the original keys and certificate can be used to provide the necessary authentication information and proof of identity for the renewal phase.
  - The certificate may or may not need to change during the renewal process; this usually depends on why the renewal is taking place.

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 date fields on the certificate.
These are the beginning and ending dates indicating the time period during which the certificate is valid.
The certificate cannot be used before the start date, and once the end date is met, the certificate is expired and a new certificate will need to be issued.
A renewal process is different from the registration phase.
The RA assumes the individual has already successfully completed one registration round.
If the certificate has not been revoked, then the original keys and certificate can be used to provide the necessary authentication information and proof of identity for the renewal phase.
The rest of the certificate, besides the validity dates, may not need to change during the renewal process; this usually depends on the reason that the renewal is taking place.
If the certificate merely expired and the keys will still be used for the same purpose, a new certificate can be generated with new validity dates.
If, however, the key pair functionality needs to be expanded or restricted, new attributes and extensions may need to be integrated into the new certificate. These new functionalities may require more information to be gathered from the individual renewing the certificate, especially if the class changes or the new key uses or allows for more powerful abilities.

This renewal process is required when the certificate has fulfilled its lifetime and its end validity date has been met. Unless, of course, the user does not want to renew, in which case, they can just leave the certificate expired. Although in that case, the right thing to do once you decide not to renew is to revoke the certificate to be extra safe.

Slide 6



Principles of Computer Security, Fifth Edition

Suspension

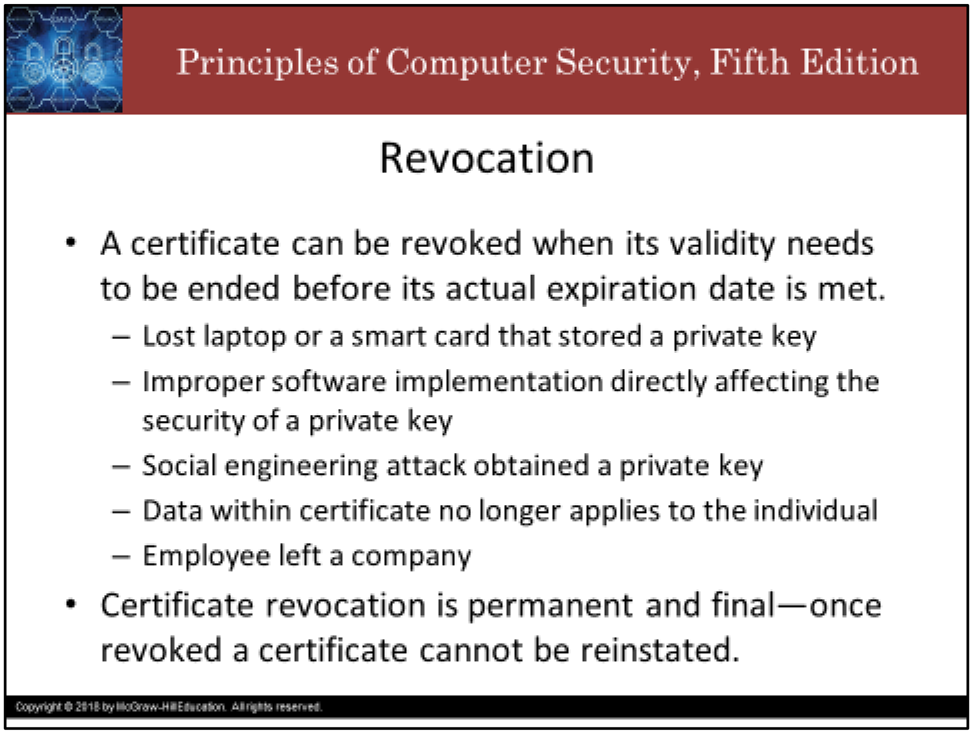
- Instead of being revoked, a certificate can be *suspended*, meaning it is temporarily put on hold.
- Reasons to suspend
  - Extended vacation – ensure certificate will not be compromised or used during that time
  - Suspicion that a private key might have been compromised

In between the options of renewing a certificate or revoking it (or just letting it expire) is the option to *suspend* the certificate.

One reason to suspend would be a user taking an extended vacation, where the suspension ensures that the certificate will not be compromised or used during that time.

Another reason to suspend would be suspicion that the private key might have been compromised, where suspension prevents the certificate from being used, but does not incur the full cost of revocation and renewal.
A certificate suspension can be a useful tool while investigating whether or not a certificate should be considered to be valid.

Slide 7



Principles of Computer Security, Fifth Edition

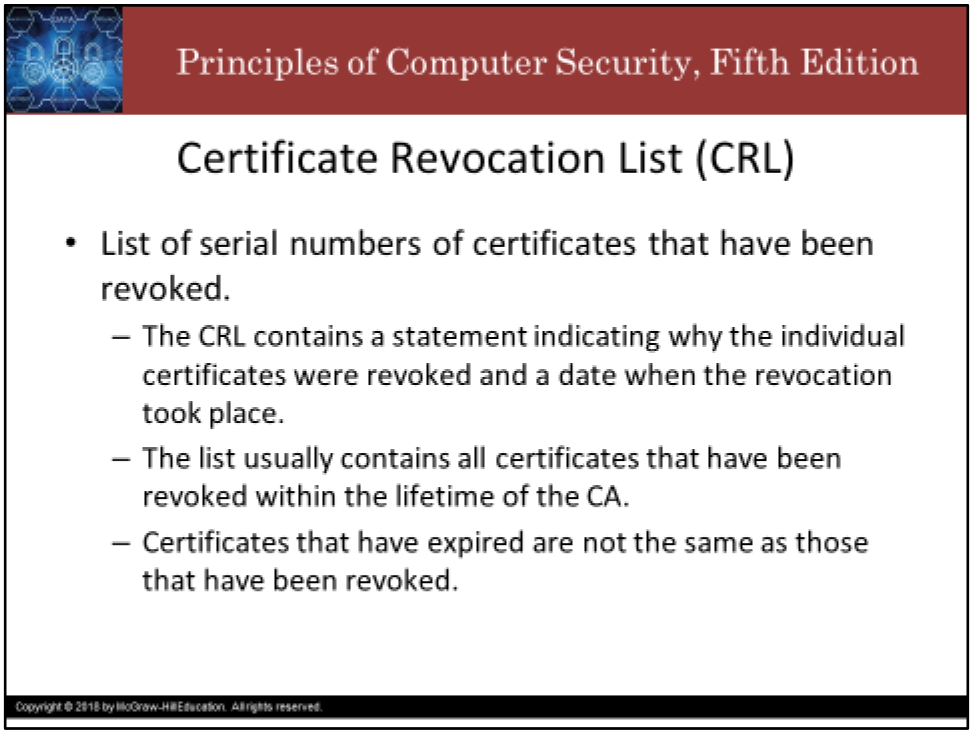
Revocation

- A certificate can be revoked when its validity needs to be ended before its actual expiration date is met.
  - Lost laptop or a smart card that stored a private key
  - Improper software implementation directly affecting the security of a private key
  - Social engineering attack obtained a private key
  - Data within certificate no longer applies to the individual
  - Employee left a company
- Certificate revocation is permanent and final—once revoked a certificate cannot be reinstated.

A certificate can be revoked when its validity needs to be ended before its actual expiration date is met. This can happen for a number of reasons, such as a lost or stolen private key, or someone leaving the organization for any reason.

Once revoked, a certificate cannot be reinstated. This is to prevent an unauthorized reinstatement by someone who has unauthorized access to the key(s). A key pair can be reinstated for use by issuing a new certificate if at a later time the keys are found to be secure. The old certificate would still be void, but the new one would be valid.

Slide 8



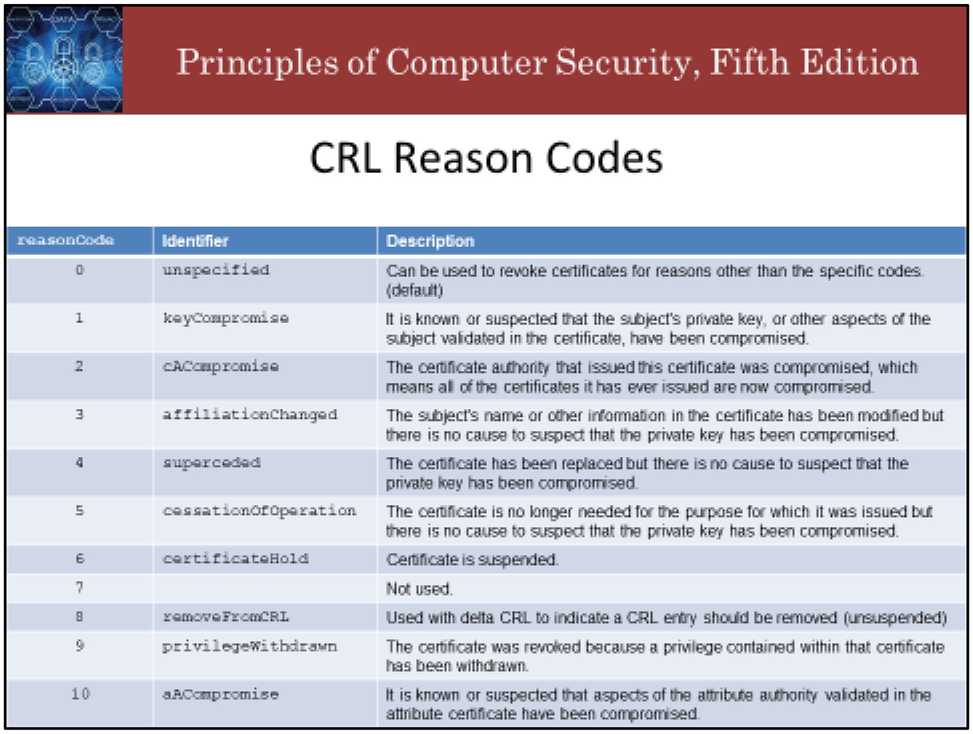
Principles of Computer Security, Fifth Edition

Certificate Revocation List (CRL)

- List of serial numbers of certificates that have been revoked.
  - The CRL contains a statement indicating why the individual certificates were revoked and a date when the revocation took place.
  - The list usually contains all certificates that have been revoked within the lifetime of the CA.
  - Certificates that have expired are not the same as those that have been revoked.

A **certificate revocation list (CRL)** is a list of serial numbers of certificates that have been revoked.
Each element of the CRL contains a statement indicating why the certificate was revoked and the date the revocation took place.
The list usually contains all certificates that have been revoked within the lifetime of the CA.
Certificates that have expired are not the same as those that have been revoked. An expired certificate means that its end validity date was reached.

Slide 9



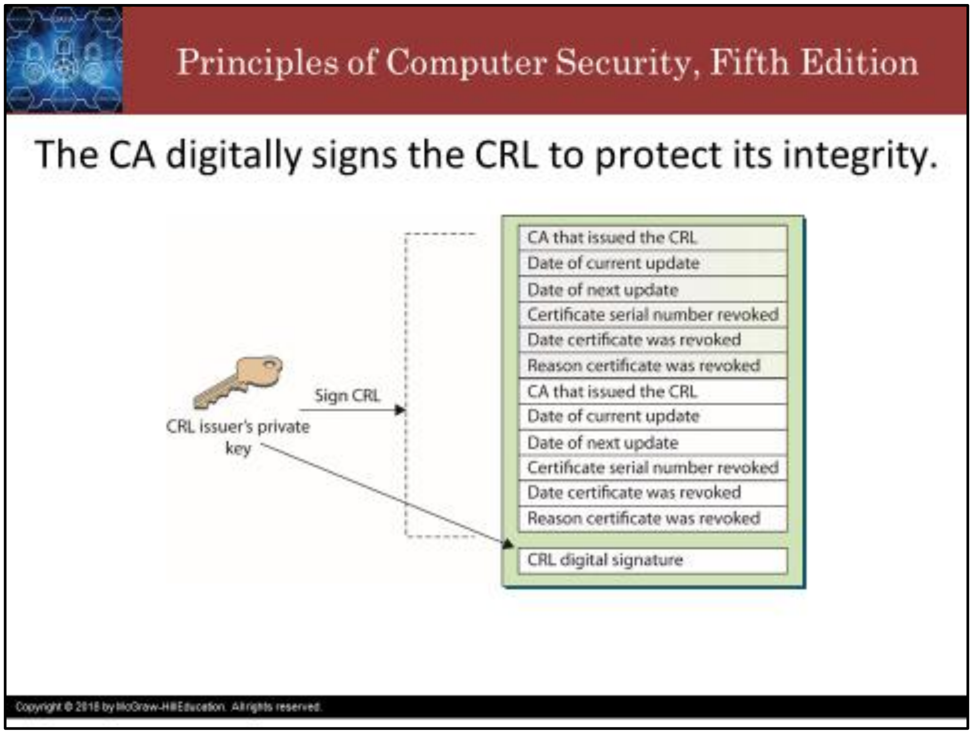
Principles of Computer Security, Fifth Edition

CRL Reason Codes

reasonCode	Identifier	Description
0	unspecified	Can be used to revoke certificates for reasons other than the specific codes. (default)
1	keyCompromise	It is known or suspected that the subject's private key, or other aspects of the subject validated in the certificate, have been compromised.
2	cACompromise	The certificate authority that issued this certificate was compromised, which means all of the certificates it has ever issued are now compromised.
3	affiliationChanged	The subject's name or other information in the certificate has been modified but there is no cause to suspect that the private key has been compromised.
4	superceded	The certificate has been replaced but there is no cause to suspect that the private key has been compromised.
5	cessationOfOperation	The certificate is no longer needed for the purpose for which it was issued but there is no cause to suspect that the private key has been compromised.
6	certificateHold	Certificate is suspended.
7		Not used.
8	removeFromCRL	Used with delta CRL to indicate a CRL entry should be removed (unsuspended)
9	privilegeWithdrawn	The certificate was revoked because a privilege contained within that certificate has been withdrawn.
10	aACompromise	It is known or suspected that aspects of the attribute authority validated in the attribute certificate have been compromised.

The reasonCode is a non-critical CRL entry extension that identifies the reason for the certificate revocation. CRL issuers are strongly encouraged to include meaningful reason codes in CRL entries.

Slide 10



Principles of Computer Security, Fifth Edition

The CA digitally signs the CRL to protect its integrity.

CRL issuer's private key

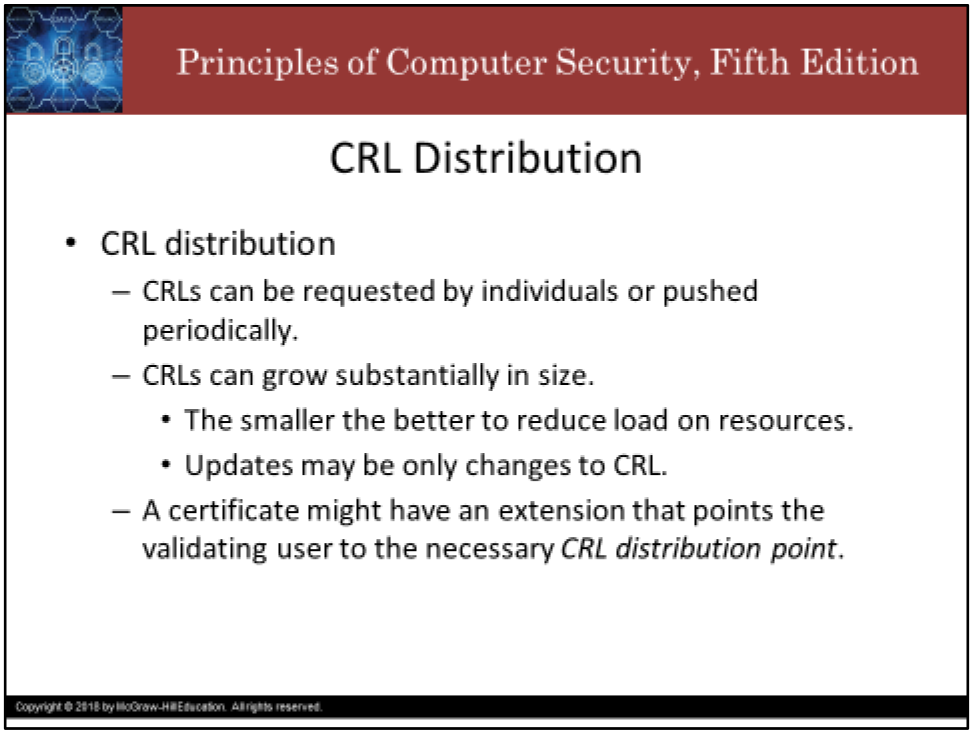
Sign CRL

CA that issued the CRL
Date of current update
Date of next update
Certificate serial number revoked
Date certificate was revoked
Reason certificate was revoked
CA that issued the CRL
Date of current update
Date of next update
Certificate serial number revoked
Date certificate was revoked
Reason certificate was revoked

CRL digital signature

The format of the CRL message is defined by X.509. The list is signed to prevent tampering and contains information on certificates that have been revoked and the reasons for their revocation. These lists can grow quite long and so there are provisions for timestamping the list and for issuing delta lists, which show changes since the last list was issued.

Slide 11



Principles of Computer Security, Fifth Edition

CRL Distribution

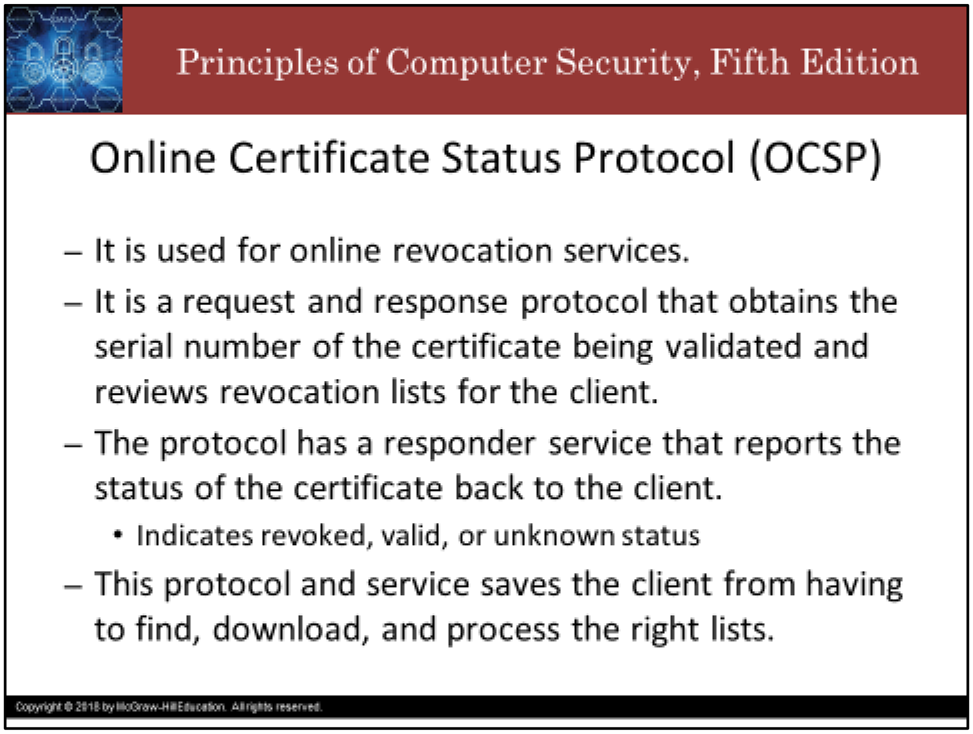
- CRL distribution
  - CRLs can be requested by individuals or pushed periodically.
  - CRLs can grow substantially in size.
    - The smaller the better to reduce load on resources.
    - Updates may be only changes to CRL.
  - A certificate might have an extension that points the validating user to the necessary *CRL distribution point*.

CRL files can be requested by individuals who need to verify and validate a newly received certificate, or the files can be periodically pushed to all users participating within a specific PKI. This means the CRL can be pulled by individual users when needed or pushed to all users within the PKI on a timed interval.

It is also possible to first push the full CRL and subsequently push only *delta* CRLs, which contain only the changes to the original or base CRL. This can greatly reduce the amount of bandwidth consumed when updating CRLs.

In implementations where the CRLs are not pushed 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. The certificate might have an extension that points the validating user to the necessary *CRL distribution point.* The network administrator sets up the distribution points, and one or more points can exist for a particular PKI. The distribution point holds one or more lists containing the serial numbers of revoked certificates, and the user's PKI software scans the lists for the serial number of the certificate the user is attempting to validate. If the serial number is not present, the user is assured that it has not been revoked. This approach helps point users to the right resource and also reduces the amount of information that needs to be scanned when checking that a certificate has not been revoked.

Slide 12



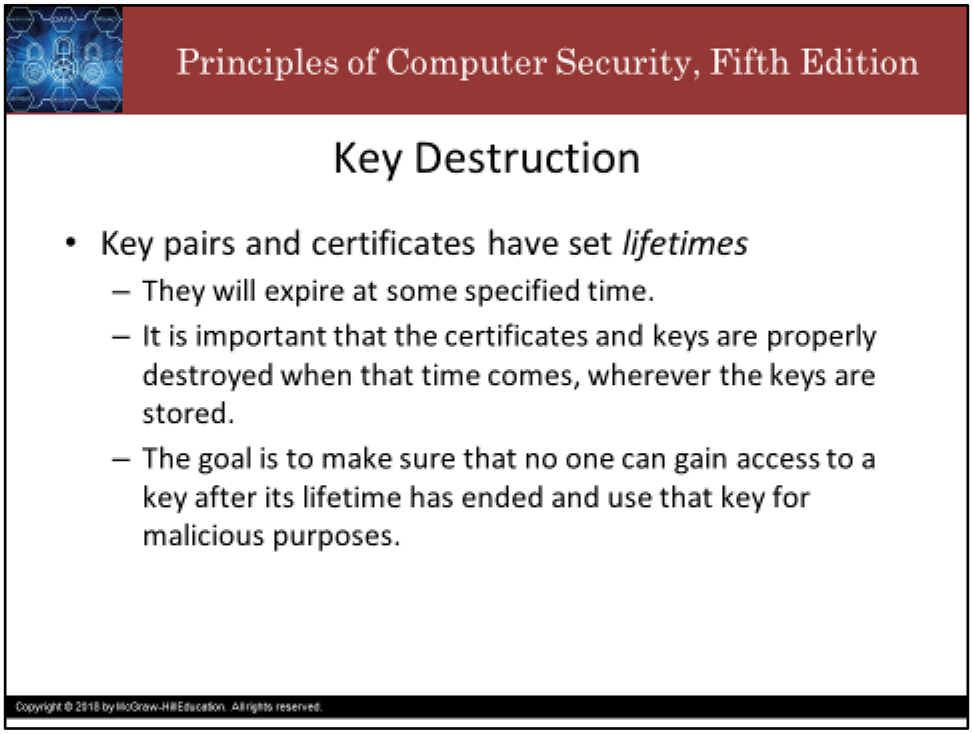
Principles of Computer Security, Fifth Edition

Online Certificate Status Protocol (OCSP)

- It is used for online revocation services.
- It is a request and response protocol that obtains the serial number of the certificate being validated and reviews revocation lists for the client.
- The protocol has a responder service that reports the status of the certificate back to the client.
  - Indicates revoked, valid, or unknown status
- This protocol and service saves the client from having to find, download, and process the right lists.

One last option for checking distributed CRLs is an online service. When a client user needs to validate a certificate and ensure that it has not been revoked, they can communicate with an online service that will query the necessary CRLs available within the environment. This service can query the lists for the client instead of pushing or pulling the full CRL to each and every system. So if Alice receives a certificate from Bob, Alice can contact an online service and send to it the serial number listed in the certificate Bob sent. The online service would query the necessary CRLs and respond to Alice, indicating whether or not that serial number was listed as being revoked.

Slide 13



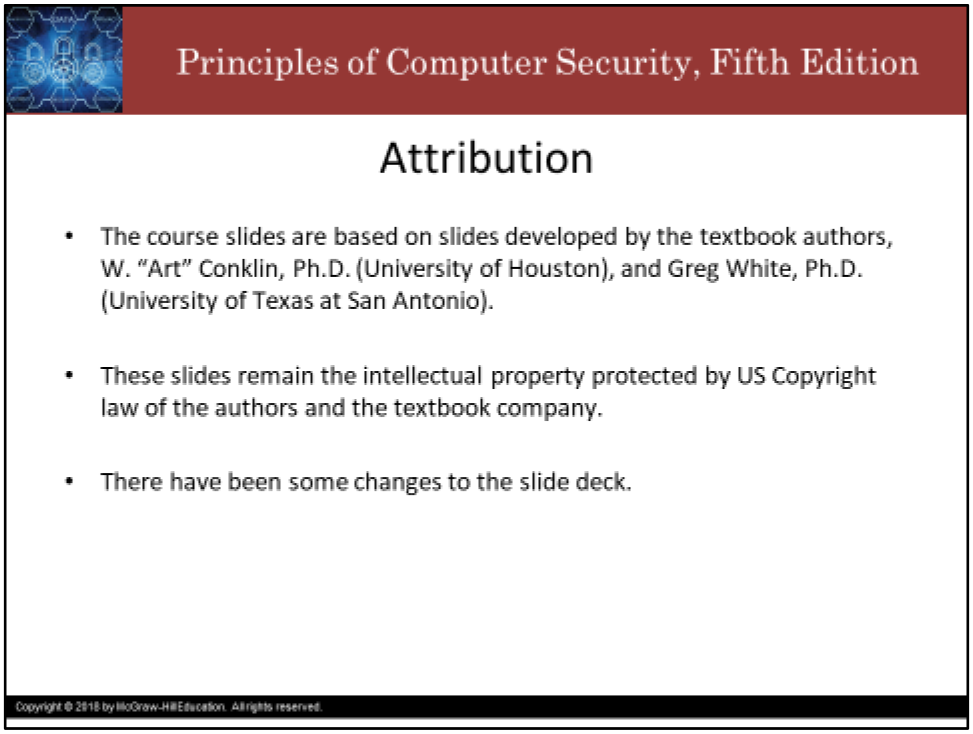
Principles of Computer Security, Fifth Edition

Key Destruction

- Key pairs and certificates have set *lifetimes*
  - They will expire at some specified time.
  - It is important that the certificates and keys are properly destroyed when that time comes, wherever the keys are stored.
  - The goal is to make sure that no one can gain access to a key after its lifetime has ended and use that key for malicious purposes.

Key pairs and certificates will expire at some specified time.
It is important that the certificates and keys are properly destroyed when that time comes.
The goal is to make sure that no one can gain access to a key after its lifetime has ended and use that key for malicious purposes, such as decrypting old traffic or forging signatures.

Slide 14



Principles of Computer Security, Fifth Edition

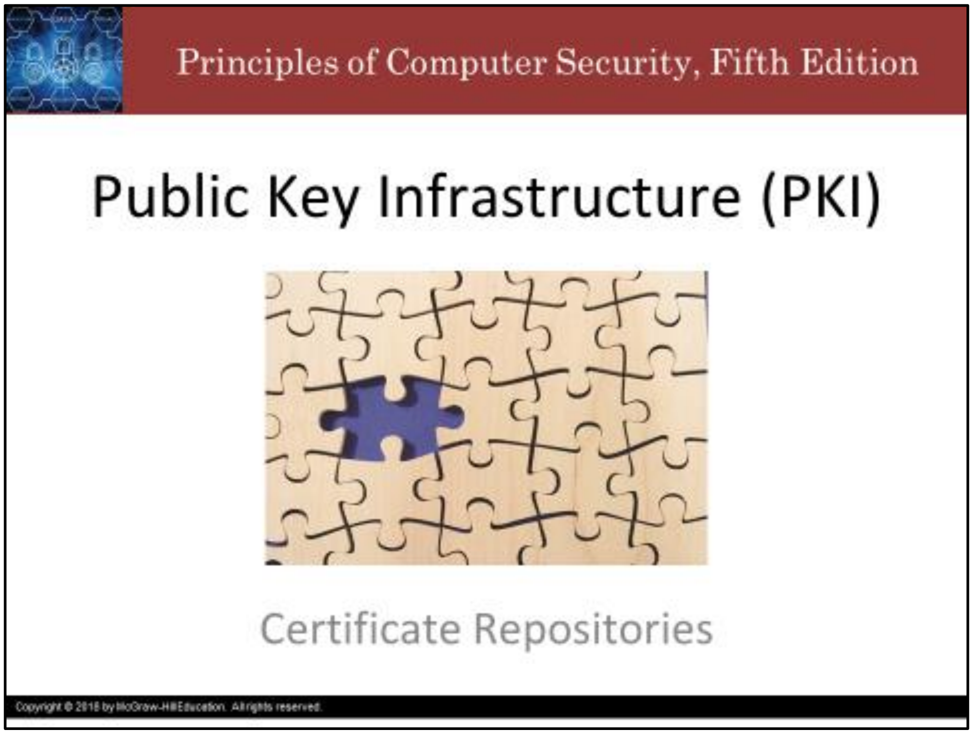
Attribution

- The course slides are based on slides developed by the textbook authors, W. “Art” Conklin, Ph.D. (University of Houston), and Greg White, Ph.D. (University of Texas at San Antonio).
- These slides remain the intellectual property protected by US Copyright law of the authors and the textbook company.
- There have been some changes to the slide deck.

Thank you and take care.

PKI: Certificate Repositories

Slide 1



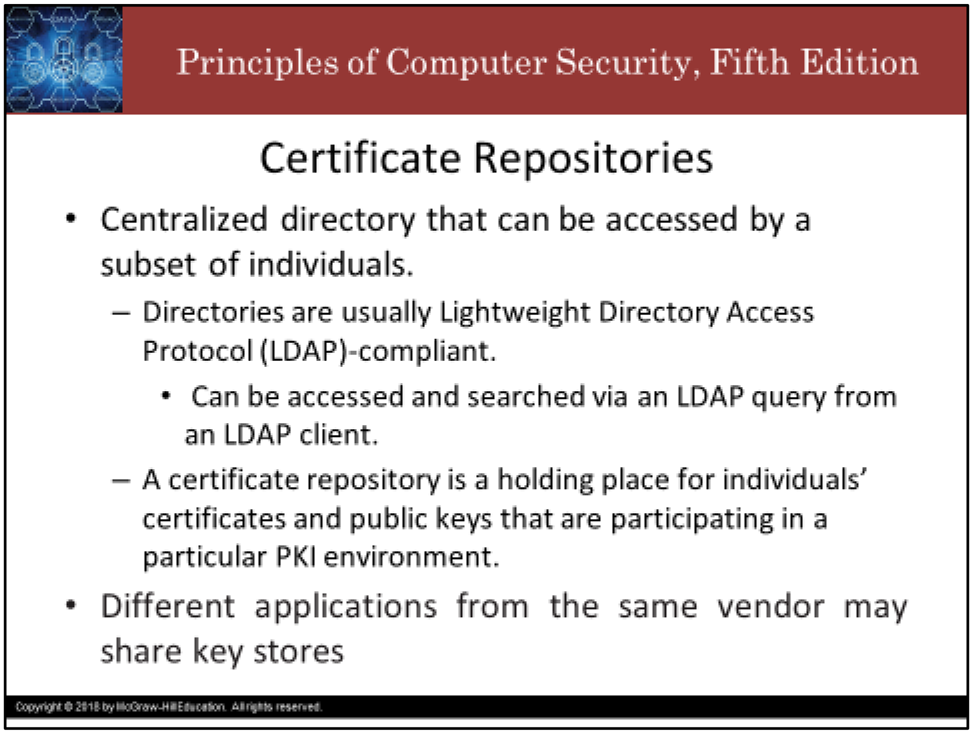
Principles of Computer Security, Fifth Edition

Public Key Infrastructure (PKI)

Certificate Repositories

Howdy! In this video, we discuss certificate repositories.

Slide 2



Principles of Computer Security, Fifth Edition

Certificate Repositories

- Centralized directory that can be accessed by a subset of individuals.
  - Directories are usually Lightweight Directory Access Protocol (LDAP)-compliant.
    - Can be accessed and searched via an LDAP query from an LDAP client.
  - A certificate repository is a holding place for individuals' certificates and public keys that are participating in a particular PKI environment.
- Different applications from the same vendor may share key stores

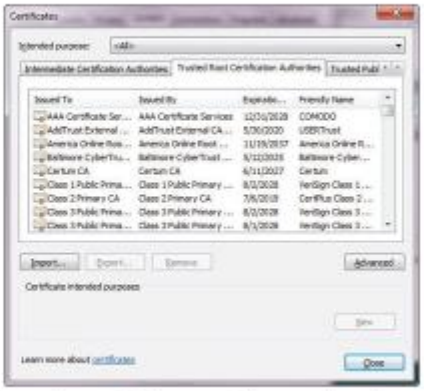
A **certificate repository** is a centralized directory that can be accessed by a subset of individuals. Directories are usually LDAP-compliant and so can be accessed and searched via a query from an LDAP client.
A certificate repository is a holding place for certificates and public keys that are participating in a particular PKI environment.

Different applications from the same vendor may share key stores to cut down on data duplication. The security requirements for repositories themselves are not as high as those needed for actual CAs and for the equipment and software used to carry out CA functions. Since each certificate is digitally signed by the CA, if a certificate stored in the certificate repository is modified, the recipient will be able to detect this change and know not to accept the certificate as valid.

Slide 3

Principles of Computer Security, Fifth Edition

Trust and Certificate Verification



Certificates

Intended purpose: <All>

Intermediate Certification Authorities | Trusted Root Certification Authorities | Trusted Publ

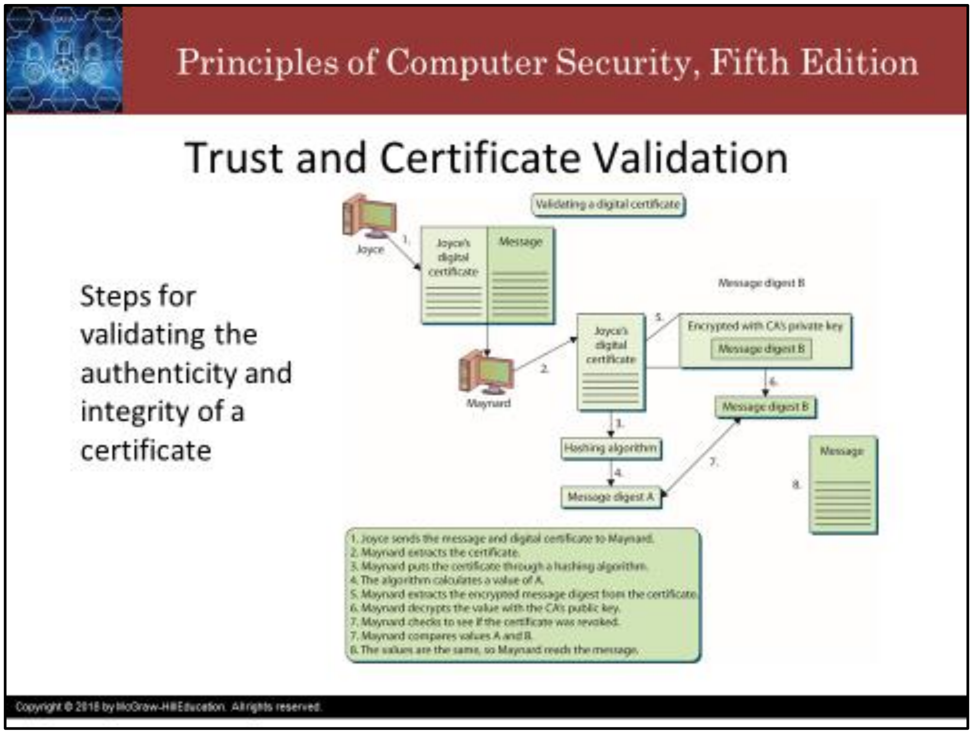
Issued To	Issued By	Expiratio...	Friendly Name
AAA Certificate Ser...	AAA Certificate Services	12/31/2028	COMODO
AddTrust External ...	AddTrust External CA ...	5/30/2020	USERTrust
America Online Roo...	America Online Root ...	11/19/2037	America Online R...
Baltimore CyberTru...	Baltimore CyberTrust ...	5/12/2025	Baltimore Cyber...
Certum CA	Certum CA	6/11/2027	Certum
Class 1 Public Prima...	Class 1 Public Primary ...	8/1/2028	VeriSign Class 1 ...
Class 2 Primary CA	Class 2 Primary CA	7/6/2019	CertPlus Class 2 ...
Class 3 Public Prima...	Class 3 Public Primary ...	8/1/2028	VeriSign Class 3 ...
Class 3 Public Prima...	Class 3 Public Primary ...	8/1/2028	VeriSign Class 3 ...

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

We need to use a PKI if we do not automatically trust individuals we do not know. Security is about being suspicious and being safe, so we need a third party that we do trust to vouch for the other individual before confidence can be instilled and sensitive communication can take place.
When a user chooses to trust a CA, she will download that CA's digital certificate and public key, which will be stored on her local computer. Most browsers have a list of CAs configured to be trusted by default, so when a user installs a new web browser, several of the most well-known and most trusted CAs will be trusted without any change of settings.

In some environments, the user can add and remove CAs from this list as needed. In environments that require a high degree of protection, this list will be pruned, and possibly the only CAs listed will be the company's internal CAs. This ensures that digitally signed software will be automatically installed only if it was signed by the company's CA. Some applications use centrally controlled policies to determine which CAs are to be trusted, instead of expecting the user to make these critical decisions.

Slide 4



Principles of Computer Security, Fifth Edition

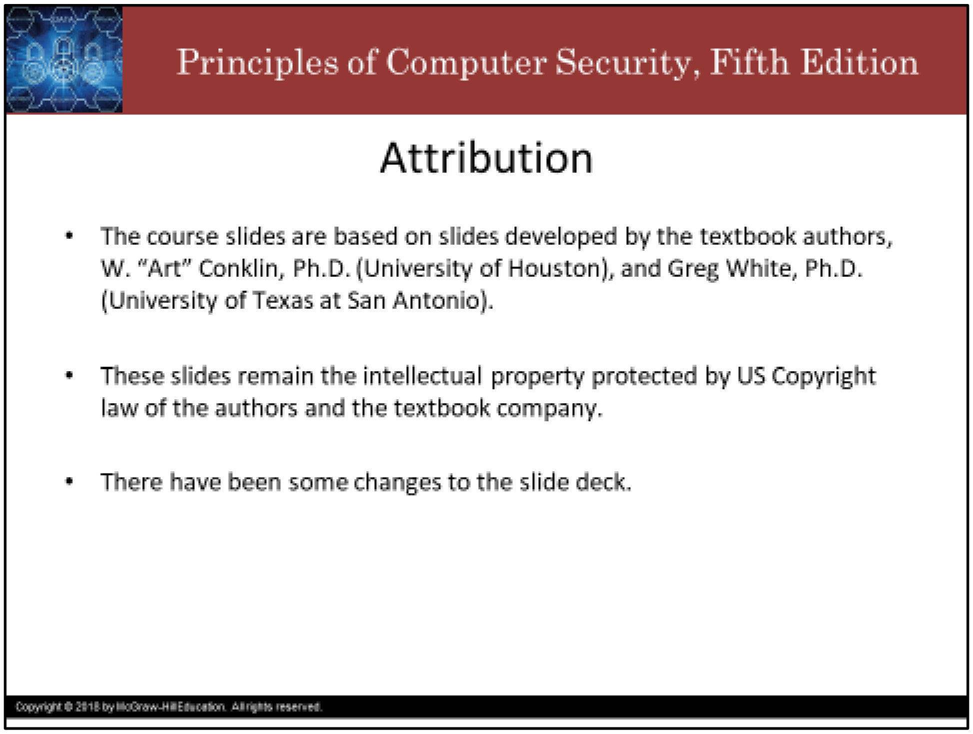
Trust and Certificate Validation

Steps for validating the authenticity and integrity of a certificate

1. Joyce sends the message and digital certificate to Maynard.
2. Maynard extracts the certificate.
3. Maynard puts the certificate through a hashing algorithm.
4. The algorithm calculates a value of A.
5. Maynard extracts the encrypted message digest from the certificate.
6. Maynard decrypts the value with the CA's public key.
7. Maynard checks to see if the certificate was revoked.
7. Maynard compares values A and B.
8. The values are the same, so Maynard reads the message.

A number of steps are involved in checking the validity of a message. Suppose, for example, that Maynard receives a digitally signed message from Joyce, who they do not know or trust. Joyce has also included their digital certificate with their message, which has their public key embedded within it. Before Maynard can be sure of the authenticity of this message, they have some work to do in order to validate the certificate. After the certificate is validated, Maynard uses Joyce's public key to validate the signature on the message. Only after both the certificate and the signature are validated can Maynard trust that the message really is from Joyce. Whether Maynard can trust the information in the message from Joyce is a matter outside the scope of the PKI.

Slide 5



Principles of Computer Security, Fifth Edition

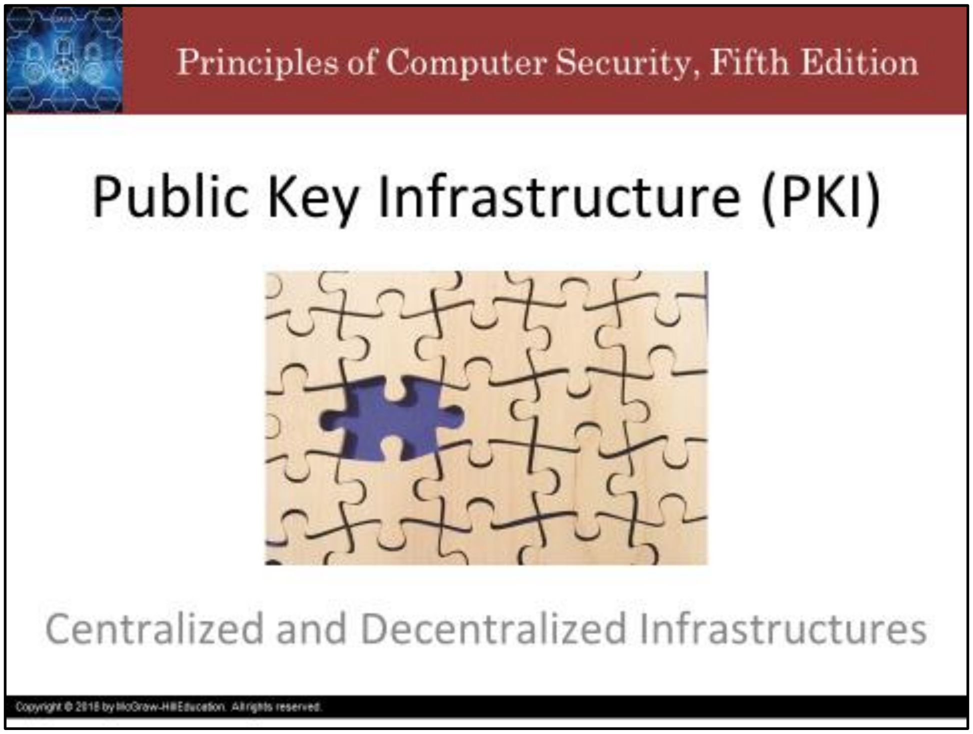
Attribution

- The course slides are based on slides developed by the textbook authors, W. “Art” Conklin, Ph.D. (University of Houston), and Greg White, Ph.D. (University of Texas at San Antonio).
- These slides remain the intellectual property protected by US Copyright law of the authors and the textbook company.
- There have been some changes to the slide deck.

Thank you and take care.

PKI: Centralized and Decentralized Infrastructures

Slide 1



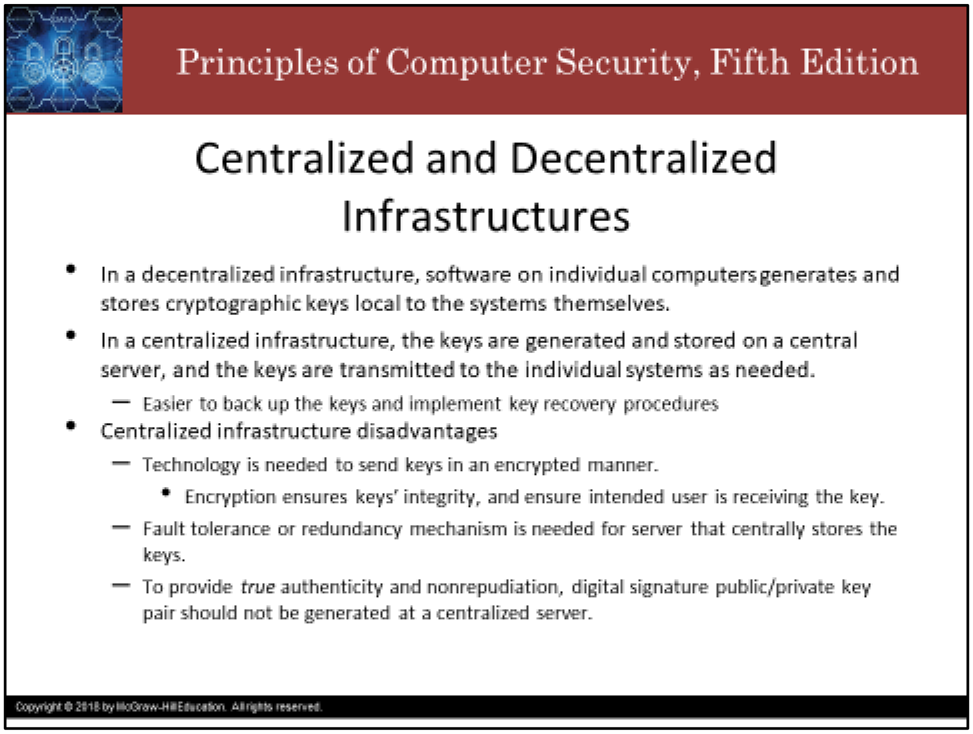
Principles of Computer Security, Fifth Edition

Public Key Infrastructure (PKI)

Centralized and Decentralized Infrastructures

Howdy! In this video, we discuss centralized and decentralized infrastructures.

Slide 2



Principles of Computer Security, Fifth Edition

Centralized and Decentralized Infrastructures

- In a decentralized infrastructure, software on individual computers generates and stores cryptographic keys local to the systems themselves.
- In a centralized infrastructure, the keys are generated and stored on a central server, and the keys are transmitted to the individual systems as needed.
  - Easier to back up the keys and implement key recovery procedures
- Centralized infrastructure disadvantages
  - Technology is needed to send keys in an encrypted manner.
    - Encryption ensures keys' integrity, and ensure intended user is receiving the key.
  - Fault tolerance or redundancy mechanism is needed for server that centrally stores the keys.
  - To provide *true* authenticity and nonrepudiation, digital signature public/private key pair should not be generated at a centralized server.

Keys used for authentication and encryption within a PKI environment can be generated in a centralized or decentralized manner.

In a decentralized infrastructure, software on individual computers generates and stores cryptographic keys local to the systems themselves.

In a centralized infrastructure, the keys are generated and stored on a central server, and the keys are transmitted to the individual systems as needed.

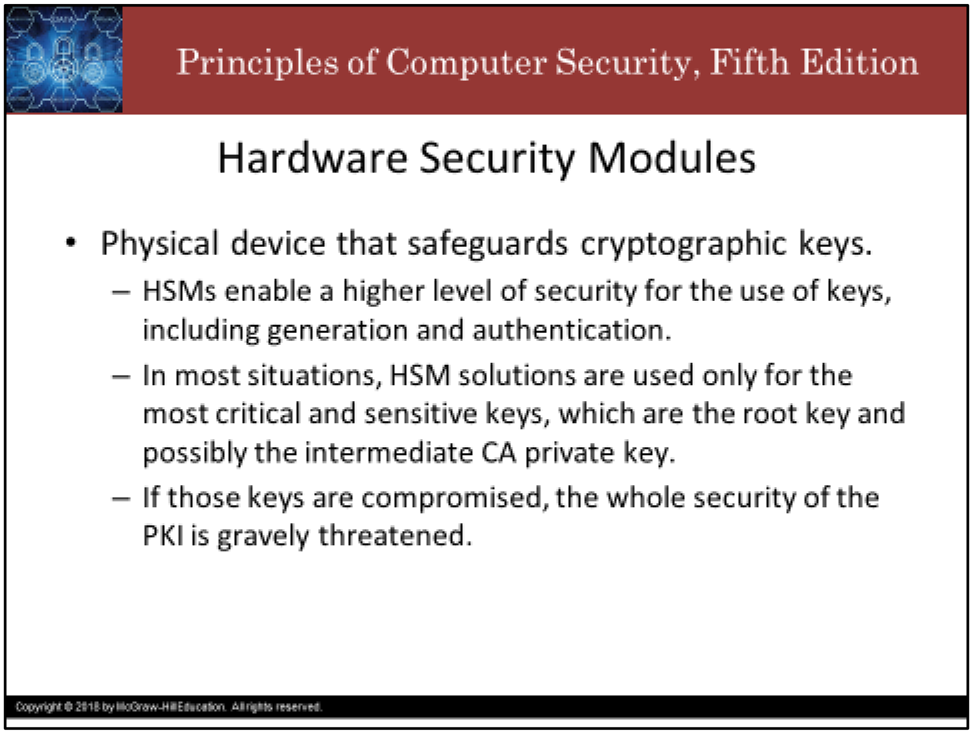
Which one you use depends on how you intend to use it.

If a company uses an asymmetric algorithm that is resource-intensive to generate the public/private key pair, and if large and resource-intensive key sizes are needed, then the individual computers may not have the necessary processing power to produce the keys in an acceptable fashion. In this situation, the company can choose a centralized approach in which a very high-end server with powerful processing abilities is used, probably along with a hardware-based random number generator.

One other issue pertains to how the keys will actually be 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. This would introduce doubt that only the one person had access to a specific private key. It is better to generate end-user keys on a local machine to eliminate doubt about who did the work and “owns” the keys.

If a company uses smart cards to hold users’ private keys, each private key often has to be generated on the card itself and cannot be copied for archiving purposes. This is a disadvantage of the centralized approach. In addition, some types of applications have been developed to create their own public/private key pairs and do not allow other keys to be imported and used. This means the keys would have to be created locally by these applications, and keys from a central server could not be used. These are just some of the considerations that need to be evaluated before any decision is made and implementation begins.

Slide 3



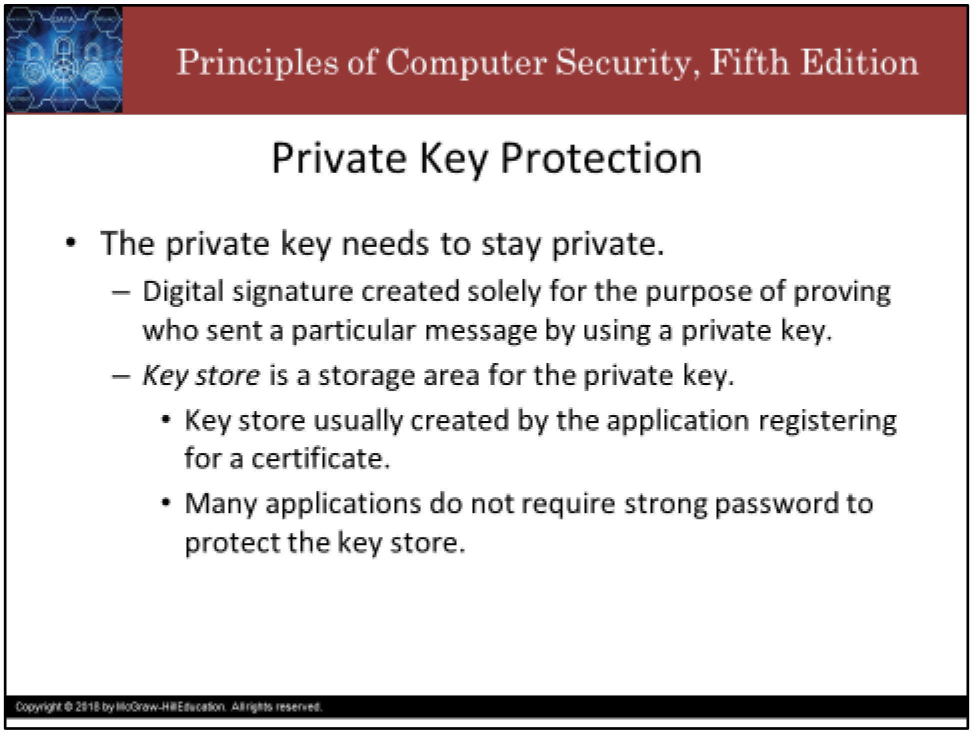
Principles of Computer Security, Fifth Edition

Hardware Security Modules

- Physical device that safeguards cryptographic keys.
  - HSMs enable a higher level of security for the use of keys, including generation and authentication.
  - In most situations, HSM solutions are used only for the most critical and sensitive keys, which are the root key and possibly the intermediate CA private key.
  - If those keys are compromised, the whole security of the PKI is gravely threatened.

A **hardware security module (HSM)** is a physical device that safeguards cryptographic keys.
HSMs enable a higher level of security for the use of keys, including generation and authentication.
In most situations, HSM solutions are used only for the most critical and sensitive keys, which are the root key and possibly intermediate CA private keys.
If those keys are compromised, the whole security of the PKI is gravely threatened.
If a person obtained a root CA private key, they could digitally sign any certificate, and that certificate would be accepted by all entities within the environment. Such an attacker might be able to create a certificate that has extremely high privileges, perhaps allowing them to modify bank account information in a financial institution, or install malware in critical infrastructure, and no alerts or warnings would be initiated because the ultimate CA, the root CA, signed it.

Slide 4



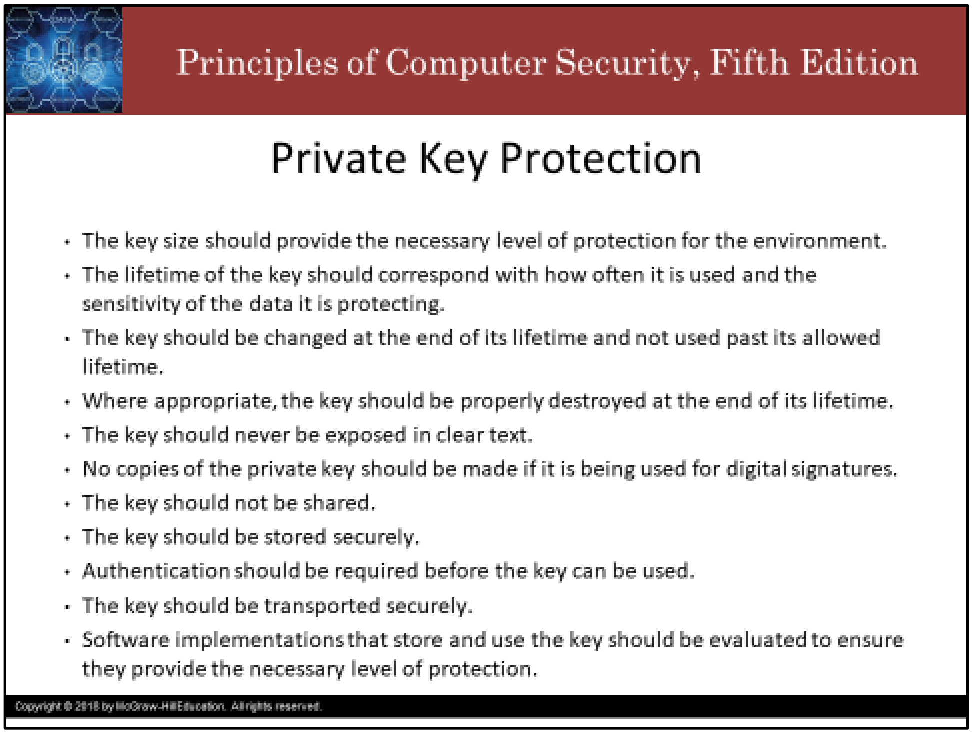
Principles of Computer Security, Fifth Edition

Private Key Protection

- The private key needs to stay private.
  - Digital signature created solely for the purpose of proving who sent a particular message by using a private key.
  - *Key store* is a storage area for the private key.
    - Key store usually created by the application registering for a certificate.
    - Many applications do not require strong password to protect the key store.

A critical concept common to all PKIs is that the private key needs to stay private.
The purpose of a digital signature is to prove who sent a particular message by using a private key.
Anyone who has Alice's private key is effectively Alice as far as the PKI is concerned.
A *key store* is a storage area for private keys, where they are usually kept encrypted by a symmetric secret key which is derived from a passphrase.
Key stores are usually created by the application registering for a certificate.
Unfortunately, many applications do not require strong password to protect the key store.

Slide 5



Principles of Computer Security, Fifth Edition

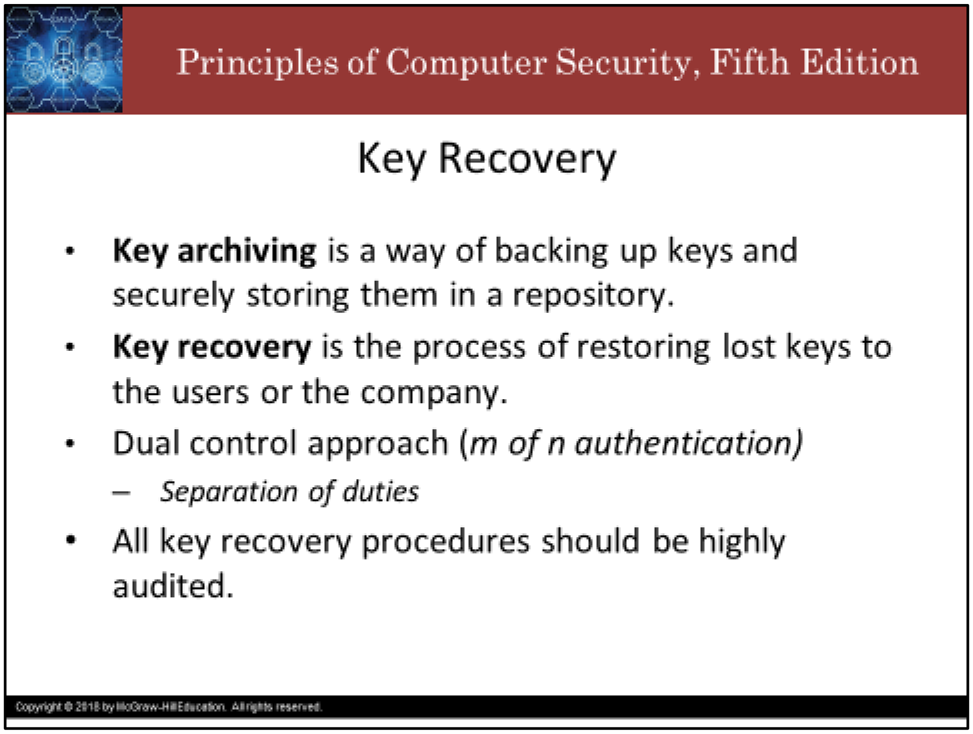
Private Key Protection

- 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 the sensitivity of the data it is protecting.
- The key should be changed at the end of its lifetime and not used past its allowed lifetime.
- Where appropriate, the key should be properly destroyed at the end of its lifetime.
- The key should never be exposed in clear text.
- No copies of the private key should be made if it is being used for digital signatures.
- 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.

Because a private key is a crucial component of any PKI implementation, the key itself should be protected at each stage of its life. This list summarizes the characteristics and requirements of proper private key use and storage:

If digital signatures will be used for legal purposes, these points and others need to be audited to ensure that true authenticity and nonrepudiation are provided.

Slide 6



Principles of Computer Security, Fifth Edition

Key Recovery

- **Key archiving** is a way of backing up keys and securely storing them in a repository.
- **Key recovery** is the process of restoring lost keys to the users or the company.
- Dual control approach (*m of n authentication)*
  - *Separation of duties*
- All key recovery procedures should be highly audited.

If you get hit by a bus, what happens to the data which is secured using your private keys? If you were the only one who knew the password to unlock the keystore or decrypt the harddrive, when you're gone, that data may be lost forever. Generally speaking, this scenario is unacceptable for many companies. So, they will have a key recovery policy and mechanisms to support it.

If a company is going to perform key recovery and maintain a key recovery system, it will generally back up only the key pair used to encrypt data, not the key pairs that are used to generate digital signatures. The reason that a company archives keys is to ensure that if a person leaves the company, falls off a cliff, or for some reason is unavailable to decrypt important company information, the company can still get to its company-owned data. This is just a matter of the organization protecting itself. A company would not need to be able to recover a key pair that is used for digital signatures, since those keys are to be used only to prove the authenticity of the individual who sent a message. A company would not benefit from having access to those keys and really should not have access to them, since they are tied to one individual for a specific purpose.

There are two important systems for backing up and restoring cryptographic keys: **Key archiving** and **Key recovery.**

Key archiving is the process of storing a set of keys to be used as a backup should something happen to the original set.

Key recovery is the process of using the backup keys.

If keys are backed up and stored in a centralized computer, this system must be tightly controlled.

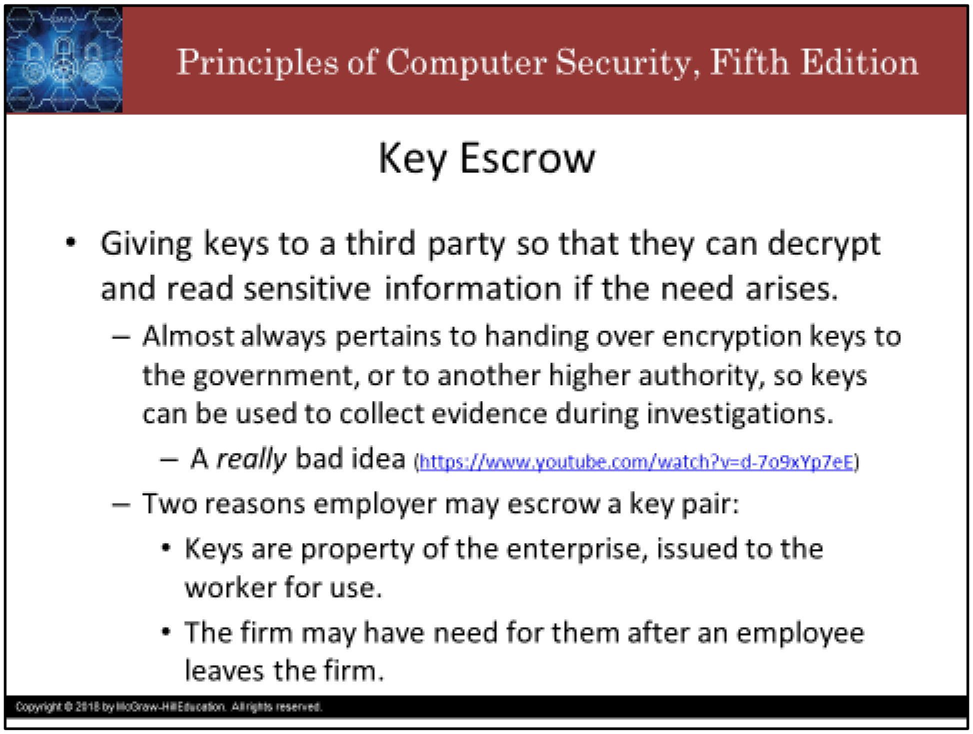
It is usually unwise to authorize a single person to be able to recover all the keys within the environment.

Requiring two individuals to recover a lost key together is called **dual control** and is an example of the principle of separation of duties. Generalizing to more than two people, we get m of n authentication,

where n is the number of people eligible to participate in the process, at least *m* of which must be involved in order to complete the task.
As m increase, so do issues associated with availability (more people, more scheduling conflicts, more risk of not having enough people to do the thing);
As m decreases, the risk of a small number of people colluding in an attack increases.
PKI systems can be configured to require multiple individuals in any key recovery process and so dual control can be used as part of a system to back up and archive data encryption keys.
The goal of dual control is to minimize fraudulent or improper use of access and permissions.

All key recovery procedures should be highly audited. The audit logs should capture at least what keys were recovered, who was involved in the process, and the time and date. Keys are an integral piece of any encryption cryptosystem and are critical to a PKI environment, so you need to track who does what with them.

Slide 7



Principles of Computer Security, Fifth Edition

Key Escrow

- Giving keys to a third party so that they can decrypt and read sensitive information if the need arises.
  - Almost always pertains to handing over encryption keys to the government, or to another higher authority, so keys can be used to collect evidence during investigations.
    - A *really* bad idea (https://www.youtube.com/watch?v=d-7o9xYp7eE)
  - Two reasons employer may escrow a key pair:
    - Keys are property of the enterprise, issued to the worker for use.
    - The firm may have need for them after an employee leaves the firm.

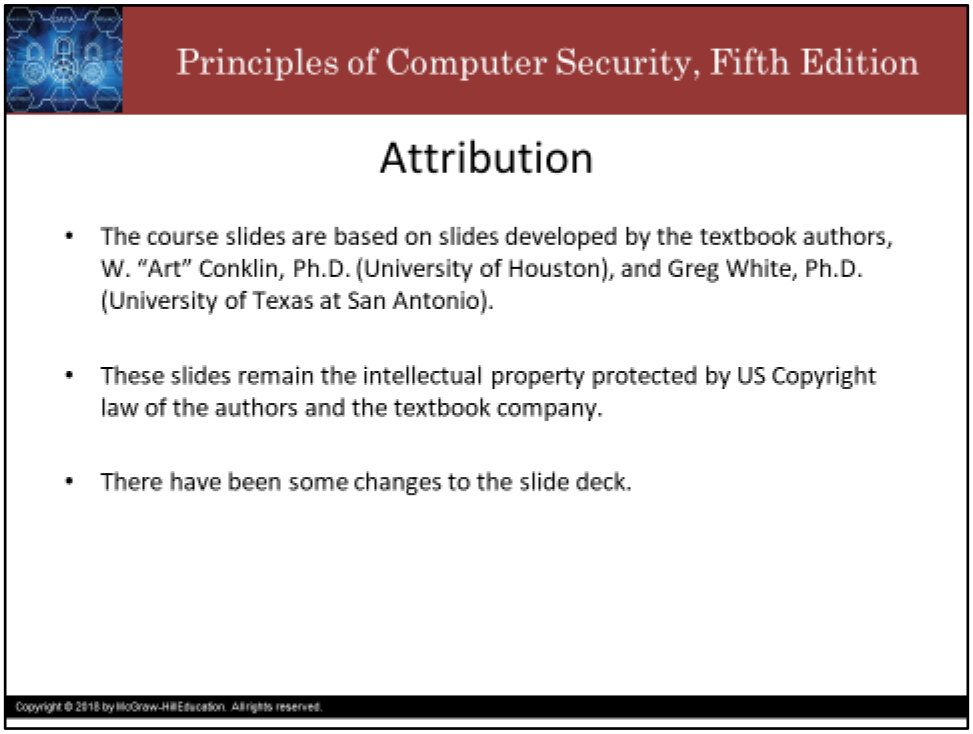
Key escrow is the process of giving keys to a third party so that they can decrypt and read sensitive information if the need arises.
Key escrow has long been a controversial topic. This essential business process provides continuity should the authorized key-holding party leave an organization without disclosing keys. The security of the escrowed key is a concern, and it needs to be managed at the same security level as for the original key.

Several movements, supported by parts of the U.S. government, would require all or many people residing in the United States to hand over copies of the keys they use to encrypt communication channels. The movement in the late 1990s behind the Clipper chip is the most well-known effort to implement this requirement and procedure. It was suggested that all American-made communication devices should have a hardware encryption chip within them. The chip could be used to encrypt data going back and forth between two individuals, but if a government agency decided that it should be able to eavesdrop on this dialog, it would just need to obtain a court order. If the court order was approved, a law enforcement agent would take the order to two escrow agencies, each of which would have a piece of the key that was necessary to decrypt this communication information. The agent would obtain both pieces of the key and combine them, which would allow the agent to listen in on the encrypted communication outlined in the court order.

The idea was that the encryption keys would be escrowed to two agencies, meaning that each agency would hold one piece of the key. One agency could not hold the whole key, because it could then use this key to wiretap people's conversations illegally. Splitting up the key is another example of separation of duties, put into place to try and prevent fraudulent activities. Thankfully, the Clipper chip standard never saw the light of day because it seemed too much like 1984's "Big Brother" to many American citizens who called and wrote to tell their representatives about their concerns about their privacy and security.

See also: https://youtu.be/HeOVbeh2yr0?t=294

Slide 8



Principles of Computer Security, Fifth Edition

Attribution

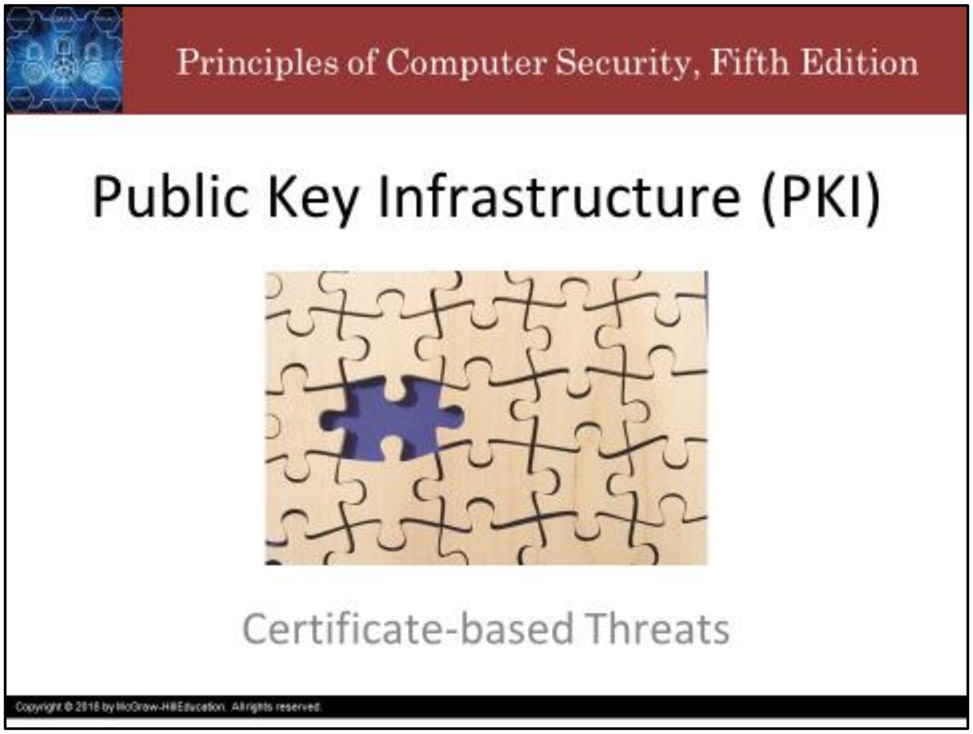
- The course slides are based on slides developed by the textbook authors, W. "Art" Conklin, Ph.D. (University of Houston), and Greg White, Ph.D. (University of Texas at San Antonio).
- These slides remain the intellectual property protected by US Copyright law of the authors and the textbook company.
- There have been some changes to the slide deck.

Copyright © 2018 by McGraw-Hill Education. All rights reserved.

Thank you and take care.

PKI: Certificate-based Threats

Slide 1



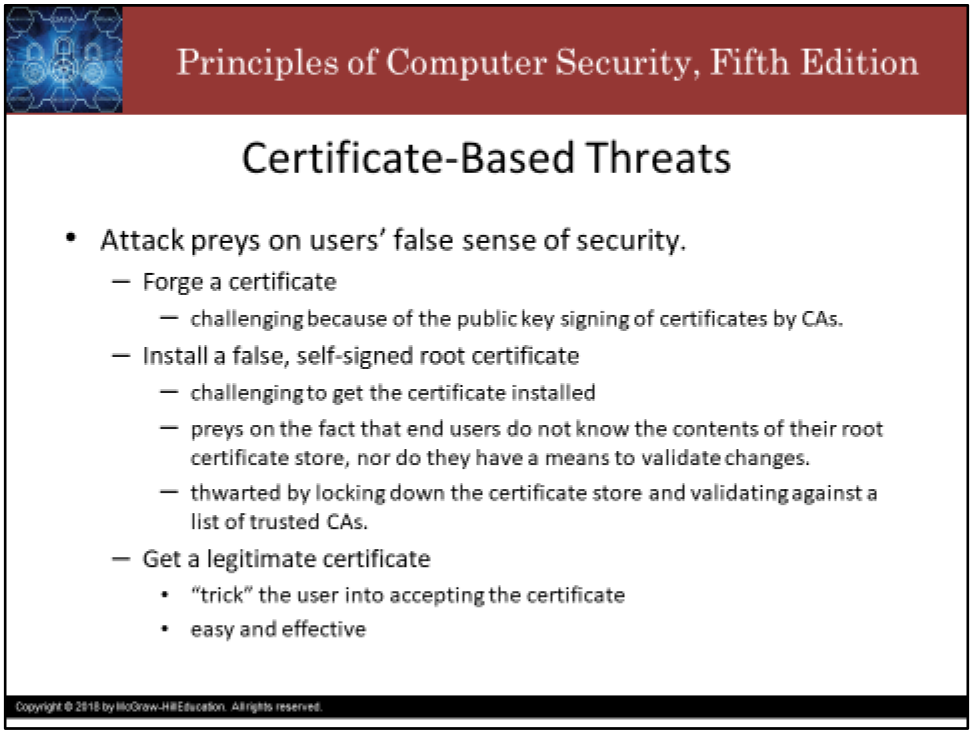
Principles of Computer Security, Fifth Edition

Public Key Infrastructure (PKI)

Certificate-based Threats

Howdy! In this video, we discuss certificate-based threats.

Slide 2



Principles of Computer Security, Fifth Edition

Certificate-Based Threats

- Attack preys on users' false sense of security.
  - Forge a certificate
    - challenging because of the public key signing of certificates by CAs.
  - Install a false, self-signed root certificate
    - challenging to get the certificate installed
    - 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.
    - thwarted by locking down the certificate store and validating against a list of trusted CAs.
  - Get a legitimate certificate
    - "trick" the user into accepting the certificate
    - easy and effective

Although certificates bring much capability to security through practical management of trust, they also can present threats.

Because much of the actual work is done behind the scenes, without direct user involvement, a false sense of security might ensue.

End users might assume that if an HTTPS connection was made with a server, they are securely connected to the proper server.

Spoofing, phishing, pharming, and a wide range of sophisticated attacks prey on this assumption.

If an attacker wants to have something recognized as legitimate, they have to obtain a certificate that proves this to the end-user's machine.

One way would be to forge a false certificate, this is too challenging because of the public key signing of certificates by CAs.

Another way would be for the attacker to install a false, self-signed root certificate on the victim's machine.

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.

It is a bit challenging, too, since it requires the attacker to have physical or remote access to the machine and access with sufficient privilege that allows them to install new root CAs.

It can be thwarted by locking down the certificate store and validating against a list of trusted CAs.

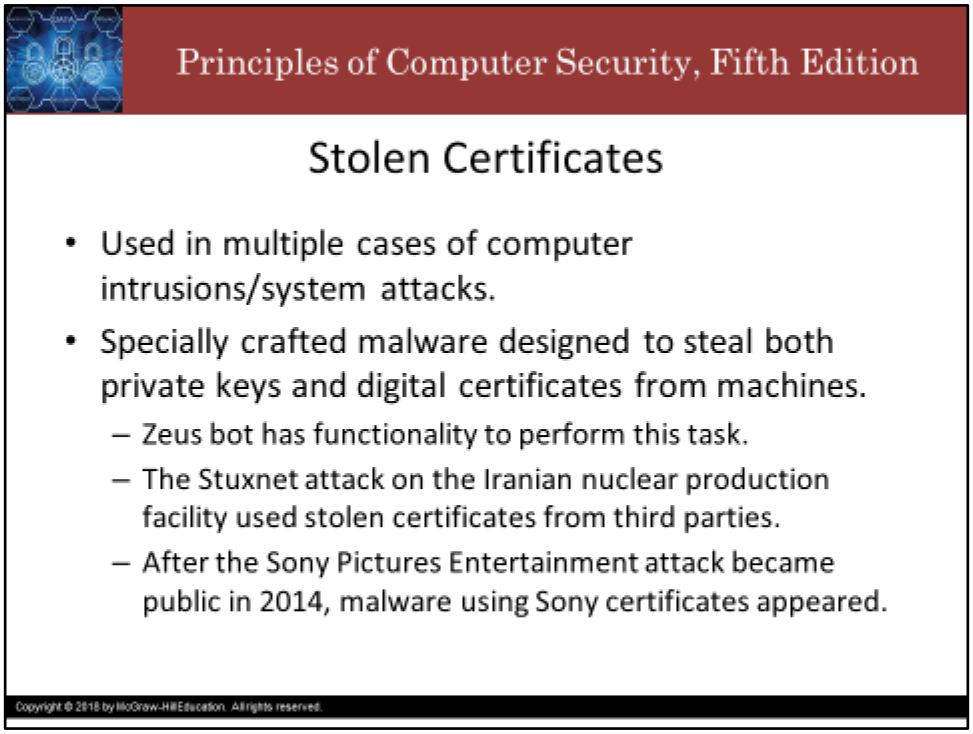
A more effective method is for the attacker to get a legitimate certificate for their false identity either from a CA that issues certs with minimal verification or by signing it themselves.

When the user's agent checks the certificate, it may give a warning that the certificate doesn't match the data (e.g. the website the user is accessing and the subject of the cert are different, but possibly in a subtle way) or that the root CA is untrusted.

But. the user, in a fit of ignorance, may nonetheless choose to continue despite the invalid certificate.

Or, the subject in the cert and the source of the data the victim wants to access do match, and so the computer continues and the user has a nice and secure connection to the attacker.

Slide 3



Principles of Computer Security, Fifth Edition

Stolen Certificates

- Used in multiple cases of computer intrusions/system attacks.
- Specially crafted malware designed to steal both private keys and digital certificates from machines.
  - Zeus bot has functionality to perform this task.
  - The Stuxnet attack on the Iranian nuclear production facility used stolen certificates from third parties.
  - After the Sony Pictures Entertainment attack became public in 2014, malware using Sony certificates appeared.

Certificates and private keys can be stolen, as well. If an attacker manages to steal the private key data for a certificate's public key, they can masquerade as the subject of the certificate.
Stolen certificates have been used in multiple cases of cyberattacks.

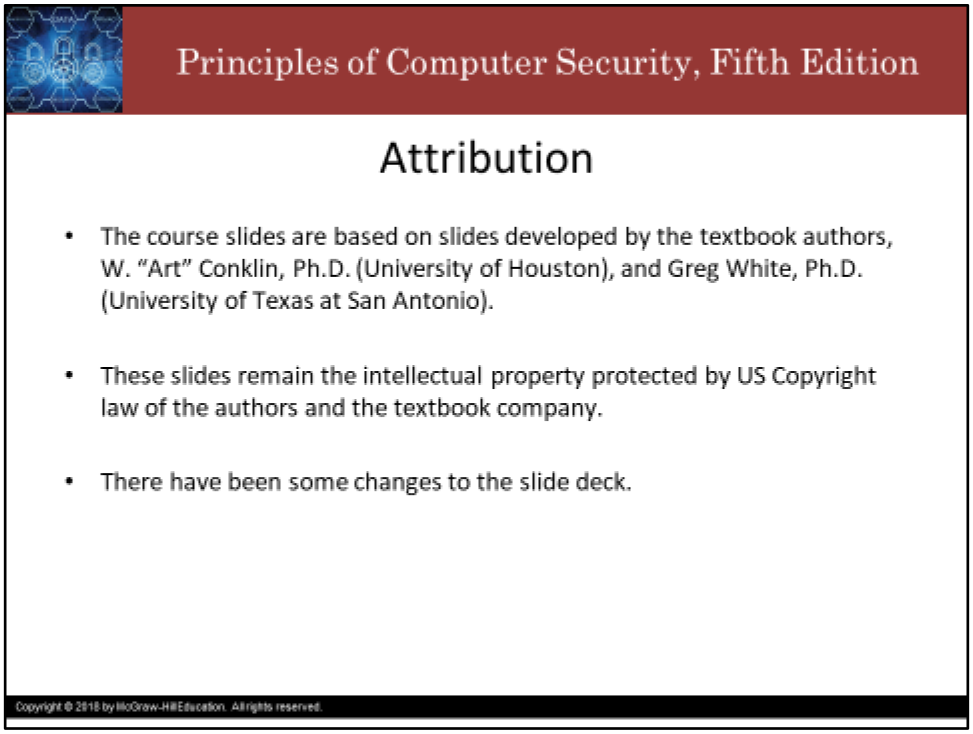
Some specially crafted malware are designed to steal both private keys and digital certificates from machines.

Zeus bot has functionality to perform this task. Zeus, also known as Zbot or WSNPoem, is famous for stealing banking information by using man in the browser keystroke logging and form grabbing. It is also used to install the CryptoLocker ransomeware.

The Stuxnet attack on the Iranian nuclear production facility used certificates stolen from third parties.

After the Sony Pictures Entertainment attack became public in 2014, malware using stolen Sony certificates appeared.

Slide 4



Principles of Computer Security, Fifth Edition

Attribution

- The course slides are based on slides developed by the textbook authors, W. “Art” Conklin, Ph.D. (University of Houston), and Greg White, Ph.D. (University of Texas at San Antonio).
- These slides remain the intellectual property protected by US Copyright law of the authors and the textbook company.
- There have been some changes to the slide deck.

Thank you and take care.