Establishing Trust With Digital Certificates

Keith Bogart

CCIE #4923

- kbogart@ine.com
- @keithbogart1
- linkedin.com/in/keith-bogart-2a75042

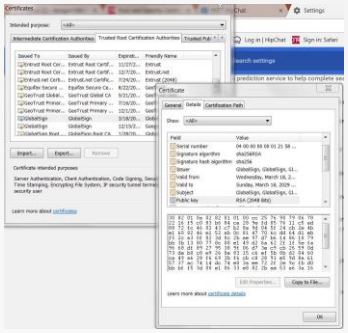
CCIE Routing & Switching

Topic Overview

- What Are Digital Certificates (Review)
- Establishing Trust
- Identity & Root Certificates
- What Is A Certificate Authority
- What Is A Digital Signature

Digital Certificates

- Digital document used for authentication purposes.
- Commonly obtained by, and stored by, web browsers
- Contains the public key of a webserver, VPN endpoint, etc
- Also called by other names:
  - Public Key Certificates
  - RSA Certificates
  - X.509 Certificates



x.500: series of standards describing databases and how they should be structured.
x.509: subset of x.500 that defines how Digital Certificates should be formatted

Establishing Trust

- Identity Certificate indicates that you can trust the remote host/website.
- But how is this “trust” established?
  - By trusting whoever issued the Cert…i.e. a Certificate Authority (CA)
  - Also called a “Root CA” or “Trusted Root”

Establishing Trust

- The Certificate Authority is a well-known company, trusted by most web browsers and operating systems.
- Must pass rigorous verification checks prior to OS installment:
  - Microsoft Root Certificate Program
    https://technet.microsoft.com/en-us/library/cc751157.aspx
  - Apple Root Certificate Program
  - Mozilla Root Certificate Program

The various “Root Certificate Programs” above are protocols created by vendors of Operating Systems as a series of verifications and checks to ensure that a Root Certificate is valid and can be trusted.

-

New Root CA’s are automatically downloaded to your OS as part of your periodic Update process (like MS Updates)

Categories Of Digital Certificates

- Root Certificate = Digital Certificate of the Certificate Authority itself.
  - Includes Public Key of the CA
  - Installed by default into most web browsers
- Identity Certificate = Digital Certificate of a non-CA entity (i.e. webserver, VPN endpoint, etc)

Which Authority

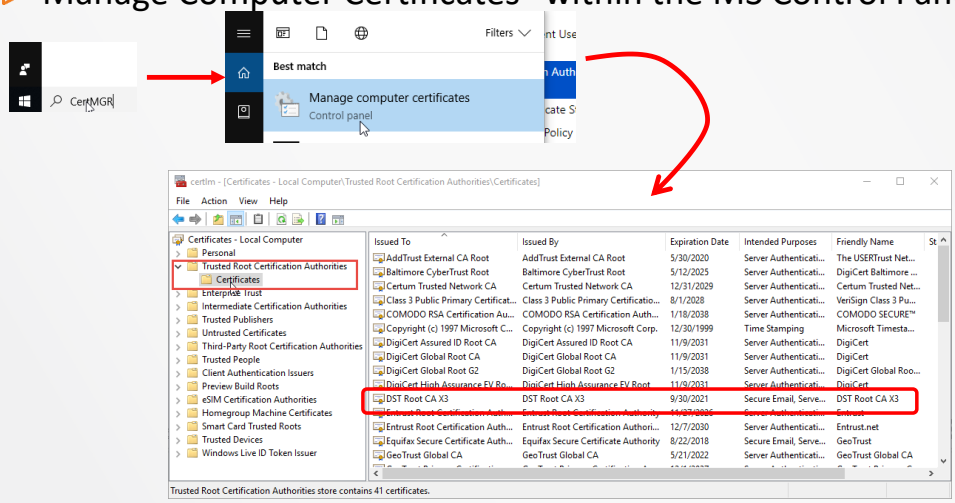
- Many different Certificate Authorities exist.
- Their offerings vary based on things like:
  - Cost (free to $$$)
  - Options available
  - Levels of certificate verification available

Which Authority

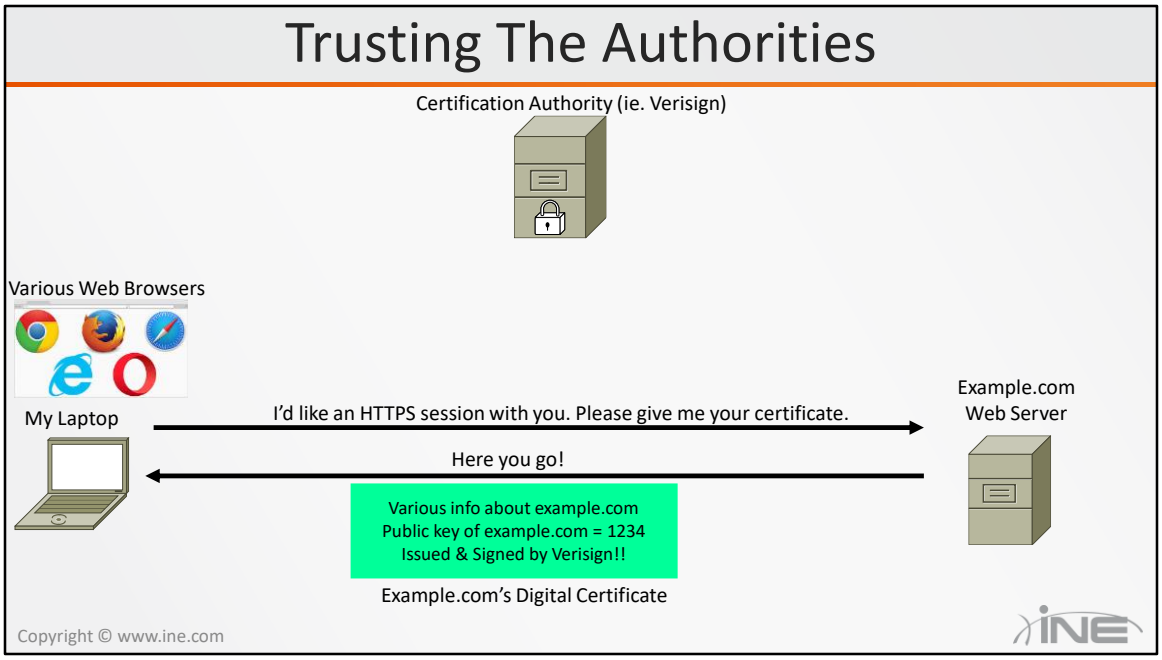
- To view which Root CA signed a particular website:

Viewing Root Certificates In Windows

- "Manage Computer Certificates" within the MS Control Panel

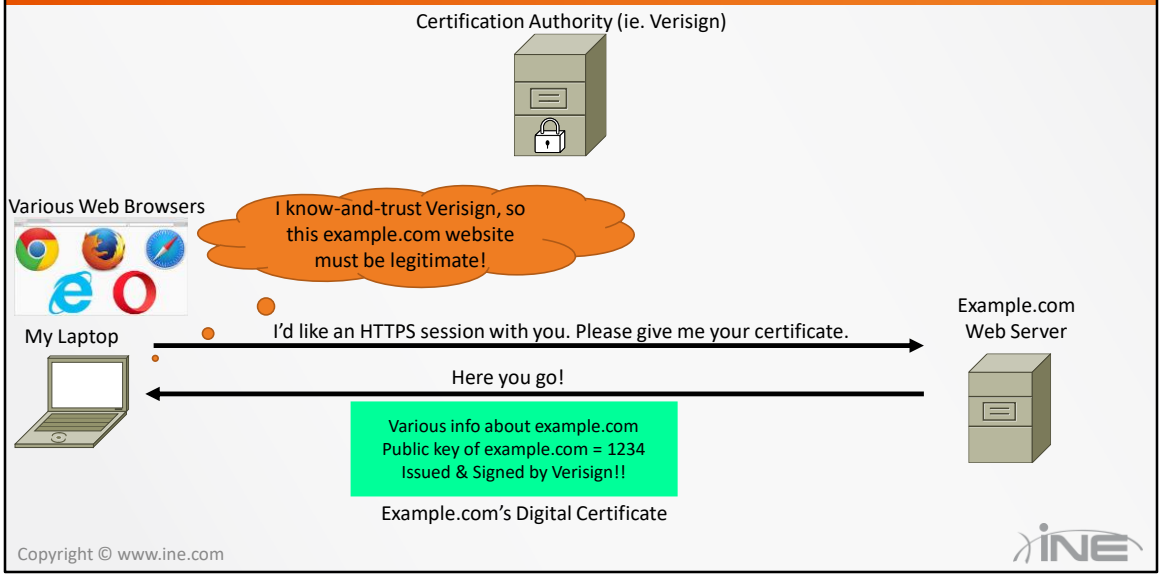


Most browsers today have the built-in certificates and public keys for the mainstream CAs on the Internet today.
-



The term "signed by Verisign" will be explained in a moment.

Trusting The Authorities



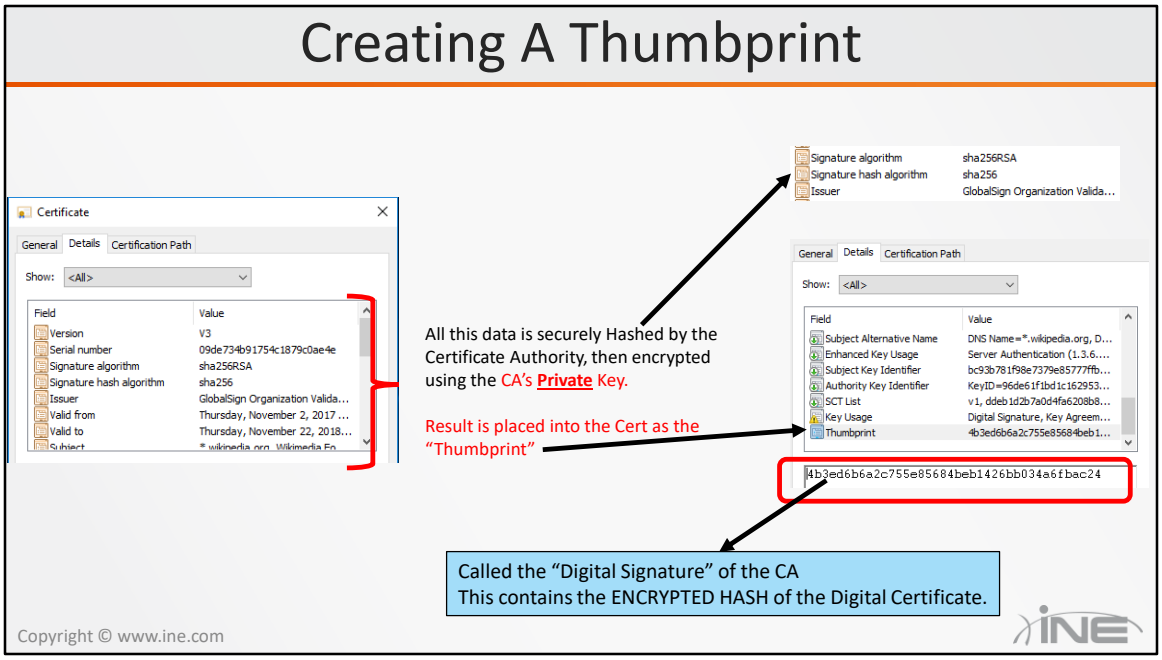
Creating Digital Signatures

- How do I know that the Identity Certificate I received was REALLY generated by a trusted Certificate Authority?
- Answer: By verifying the "Digital Signature" in the Certificate.
- Digital Signature:
  - CA Hashes all contents of the cert with a well-known Hash Function (currently SHA-2 using 256-bit Digest)
  - Resulting digest is then encrypted, using CA's **Private** Encryption Key
  - Encrypted Digest appended to Certificate as a "thumbprint".

Why not skip the Hashing step and just encrypt the entire cert with the CA's Private Key?

-

Answer: Asymmetric encryption is computationally slow and CPU-expensive. It is easier to encrypt something smaller-in-size (like a 256-bit Hash Digest) than something larger in size (like the entire contents of the Certificate).



All of the info you provided to Cert Authority is hashed using a well-known Hashing algorithm (SHA256 above).

-

This hash is then ENCRYPTED by the CA using THEIR "private key".

-

When you give your cert to someone else, they have the CA's "public key" that is used to decrypt, and reveal the hash.

Trusting Digital Signatures

- Once your laptop receives the Digital Certificate with thumbprint (aka Digital Signature) it;
  - Locates the Public Key associated with the CA listed on the Cert
  - Decrypts the Thumbprint, revealing the SHA-2 Hash Digest
  - Your local system performs the same SHA-2 Hash of the Certificate contents and derives its own Hash Digest
- If locally-derived Digest = decrypted Digest then you know this Certificate MUST have been encrypted by the Certificate Authority (whom you trust).

Remember that you already have Public Key of the Cert Authority stored as part of your Operating System in your Root Store.