Public Key Infrastructure, Digital Certificates, Certificate Revocation and Transparency

29 May 2025
Lecture 9

Some slide Credits: Steve Zdancewic (UPenn)

Topics for Today

- Certificates and PKI
- Certificate Types
- Certificate Validation
  - OCSP
  - Certificate Pinning
  - Certificate Transparency

Trusted Third Parties

- KDC and KTC are *Trusted Third Parties (TTP)*

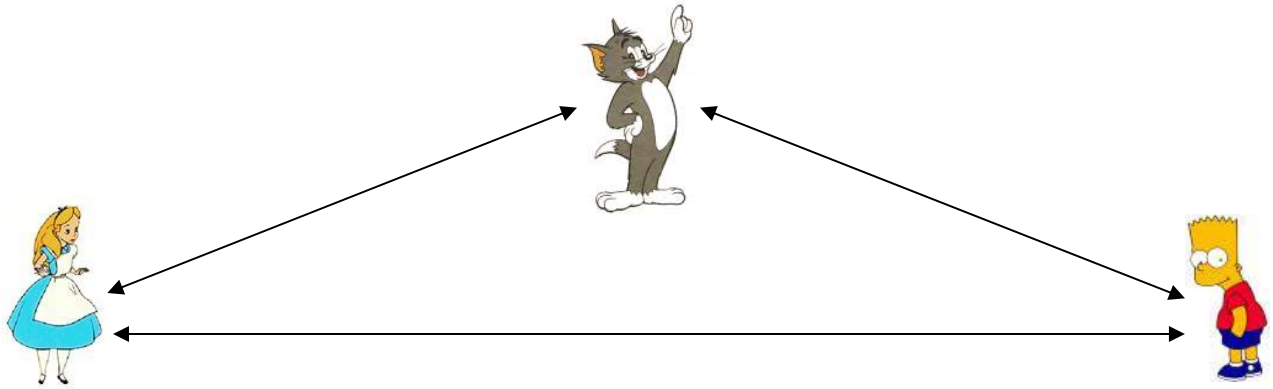
- Generalizing, TTPs can have different kinds of roles in key management and secure communication

In-line	On-Line	Off-Line

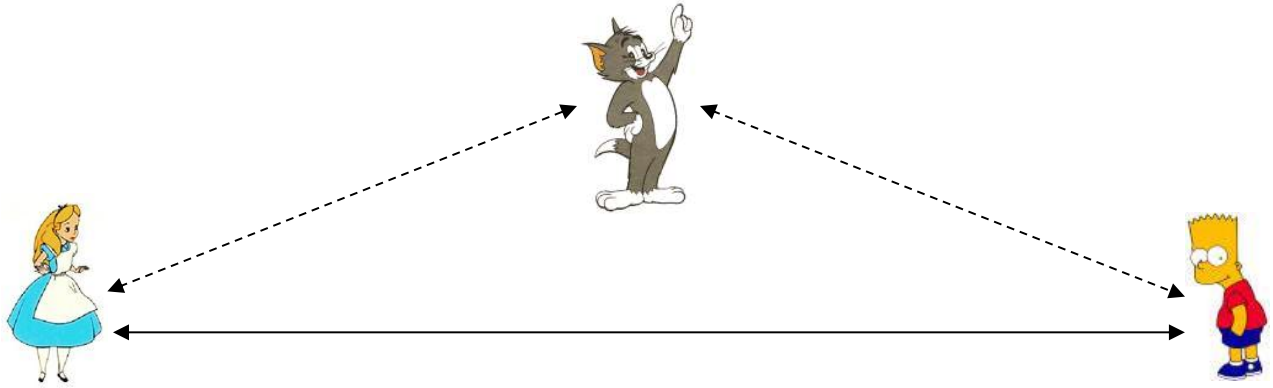
In-line, On-Line, Off-Line

In-Line:

On-Line:



Off-Line:



TTP Examples

In-Line:

- Instant Messaging, SMS

On-Line:

- Needham-Schroeder
- KDC
- KTC
- Kerberos

Offline:

- Public Key Certificate Authorities (CA)

TTPs in Public-Key Certificates

Name Server

- Resolving and managing names of entities

Registration authority

- Authorizing entities, associating keys with names

Key generator

- May generate the public/private key pair
- May be part of the user's job

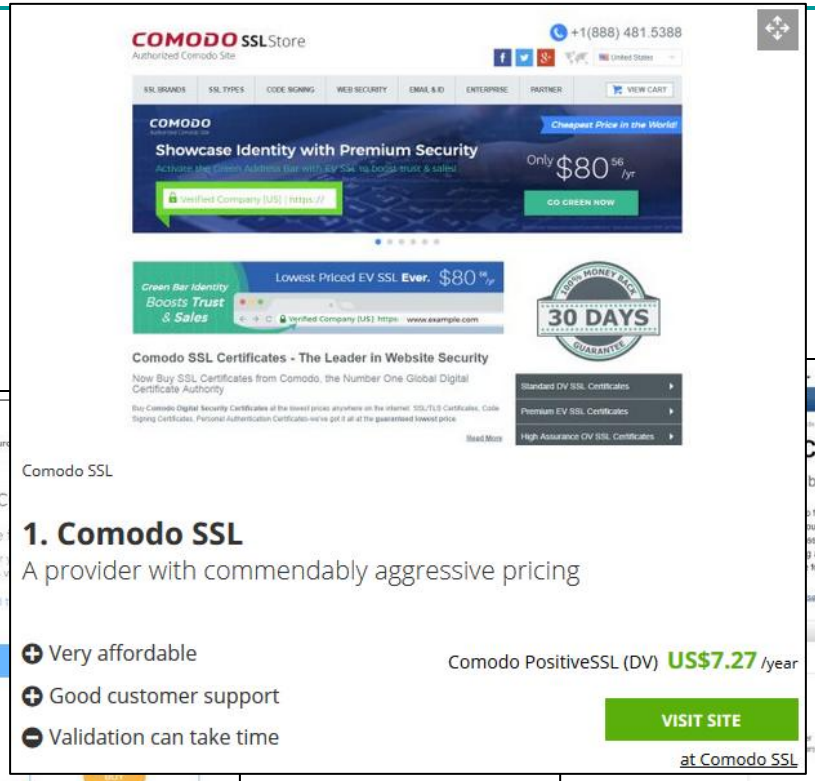
Certificate Directory

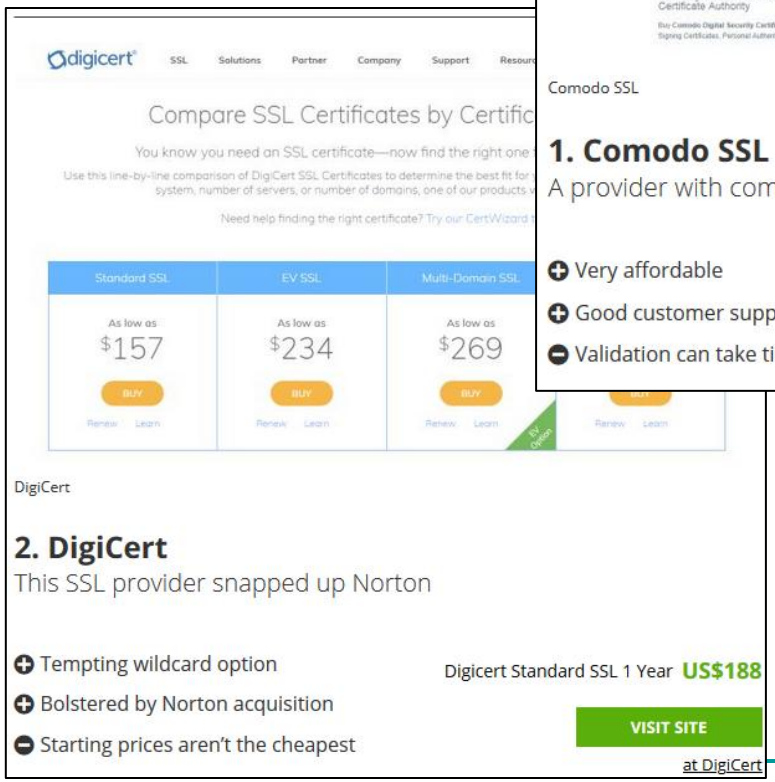
- Store lists of certificates for names
- Readable by anybody

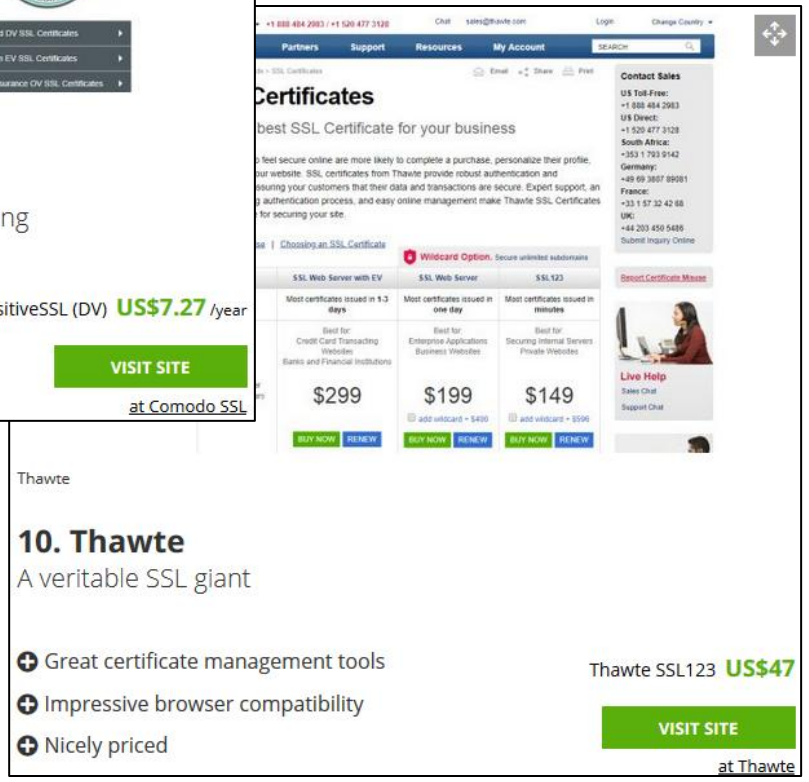
- All this rolls into what's called a *Certification Authority*

Source: https://www.techradar.com/news/best-ssl-certificate-provider

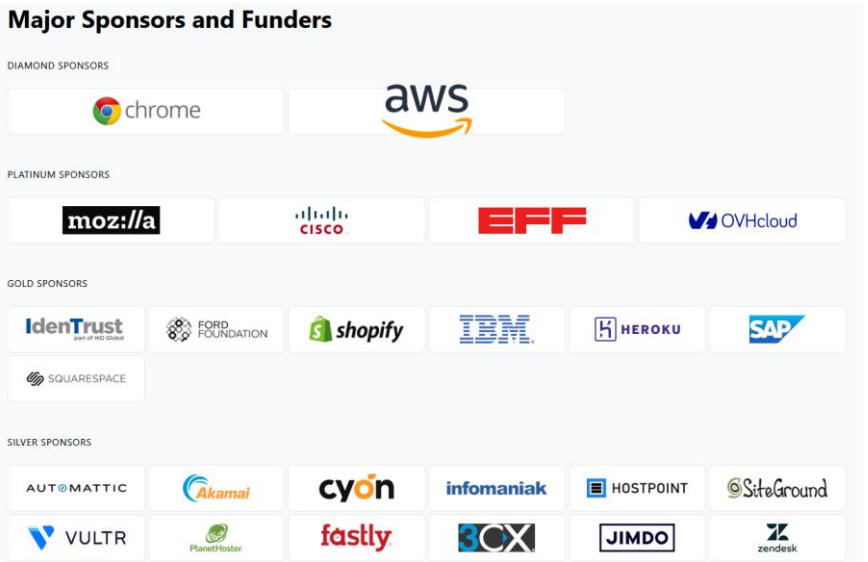
Some players





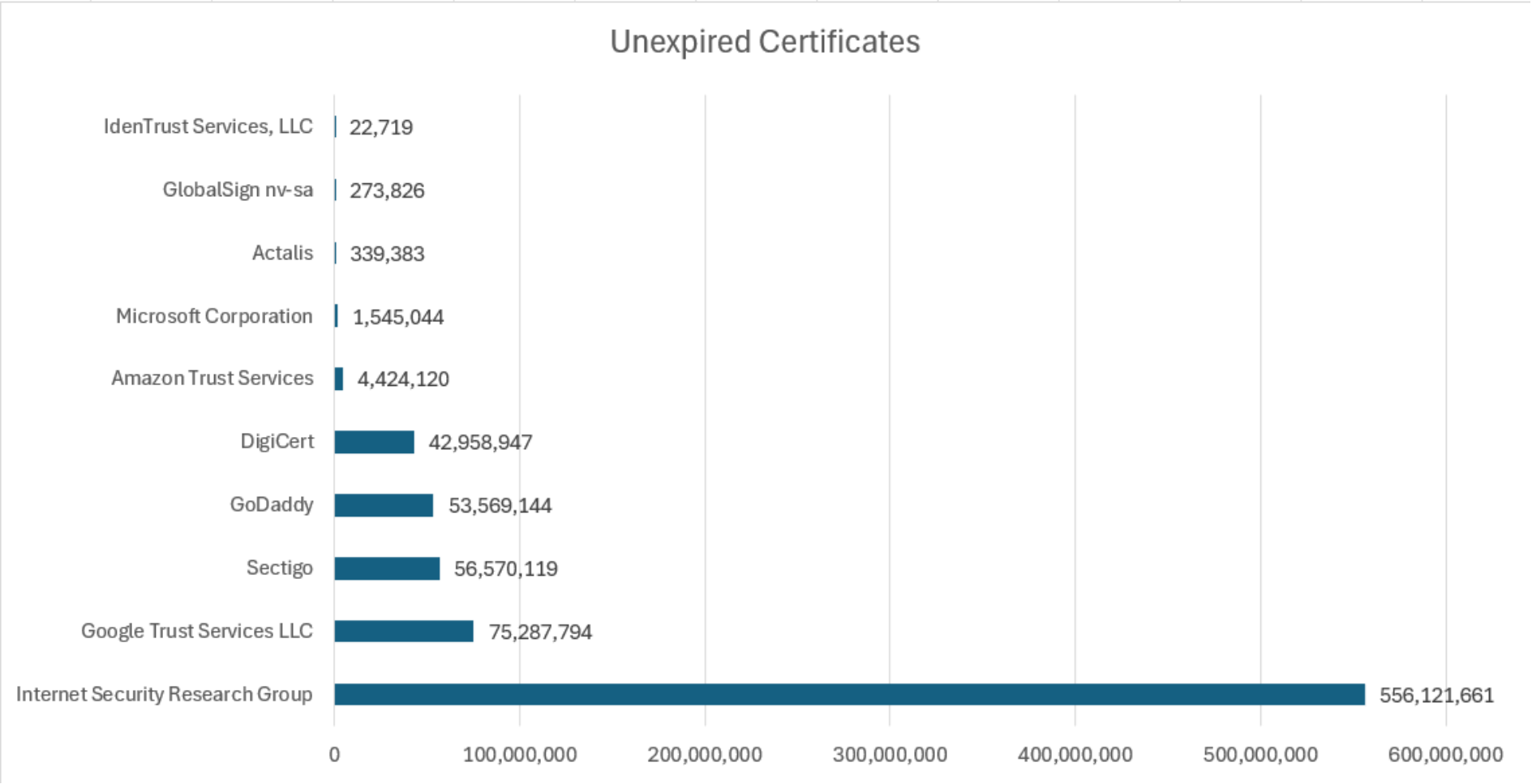


A free one with big backers



Source: https://letsencrypt.org/

Top CAs 2025 (by # issued)



ISRG is the organization behind Let's Encrypt

Basic Idea


I made a public key, sign it for me
Ok, here's your certificate
I'm Alice and this is my public key. Tom signed this certificate

Public Key Infrastructure (PKI)

Public key infrastructure (PKI)

- PKI is the set of services needed to create, manage, store, distribute and revoke digital certificates based on public-key cryptography.

Certification Authorities (CAs)

- A trusted third party that issues certificates and (often) certificate revocation lists.
- Example: GoDaddy

A certificate is (roughly) of the form $M, k_{CA}\{H(M)\}$ where

- $M = Name, K_{Pub_{Name}}, L$
- $Name =$ identifier of a principal (e.g. a URL)
- $K_{Pub_{Name}} =$ the public key of the principal
- $L =$ lifetime of the certificate

X.509 Certificate Standard

- Issued in 1988 by the PKIX working group of the IETF
- Message format that specifies how certificates should be shared:

Certificate

Version, Serial Number, Algorithm ID

Issuer, Validity (Not Before, Not After)

Subject, Subject Public Key Info (Algorithm, Key)

Issuer Unique Identifier (Optional)

Subject Unique Identifier (Optional)

Extensions (Optional)

Certificate Signature Algorithm

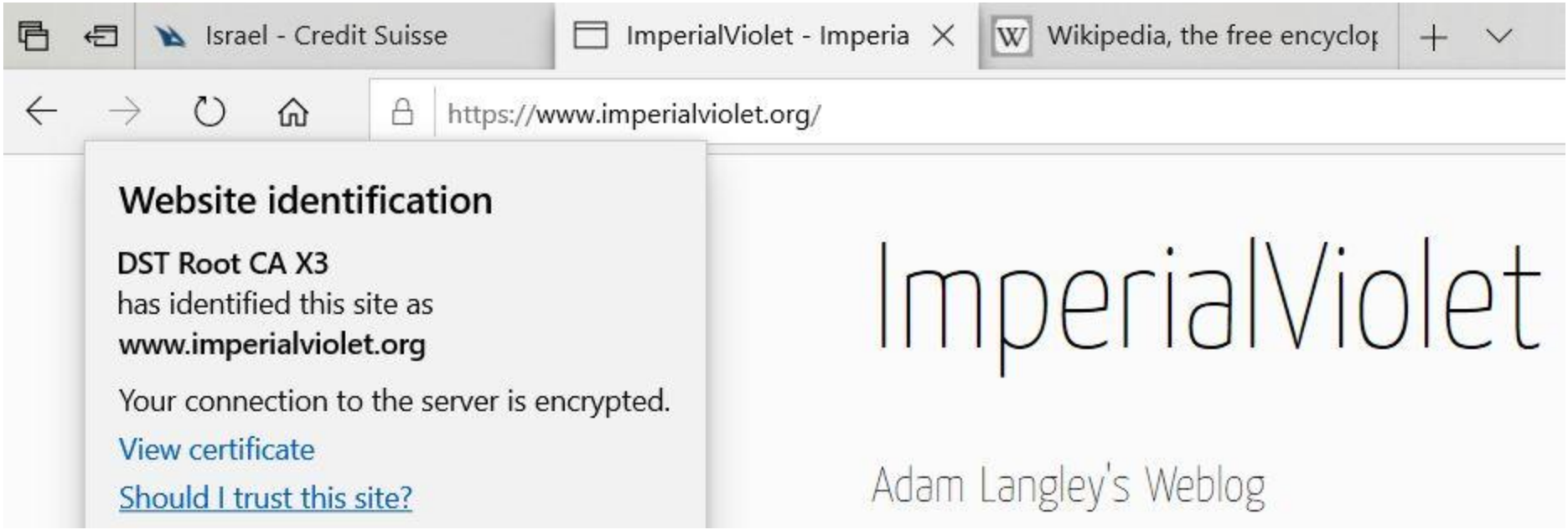
Certificate Signature

Example X.509 certificate

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 03:1b:1f:06:12:df:b5:a0:53:a7:e6:f5:1c:63:52:38:8e:84
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C = US, O = Let's Encrypt, CN = R3
        Validity
            Not Before: Jun  5 01:08:38 2025 GMT
            Not After : Sep  3 01:08:37 2025 GMT
        Subject: CN = kinneret.ac.il
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (4096 bit)
                Modulus: 00:b5:22:1e:77:90:53:65:40:b2:29:3a:82:44:c1: [...]
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Key Usage: critical
                Digital Signature, Key Encipherment […]
            X509v3 Basic Constraints: critical
                CA:FALSE
    [...]
    Signature Algorithm: sha256WithRSAEncryption
    Signature Value: 72:cf:e8:37:e4:0a:a3:10:93:4d:27:d0:ce:22:fa:f5:6c:9b: [...]
```

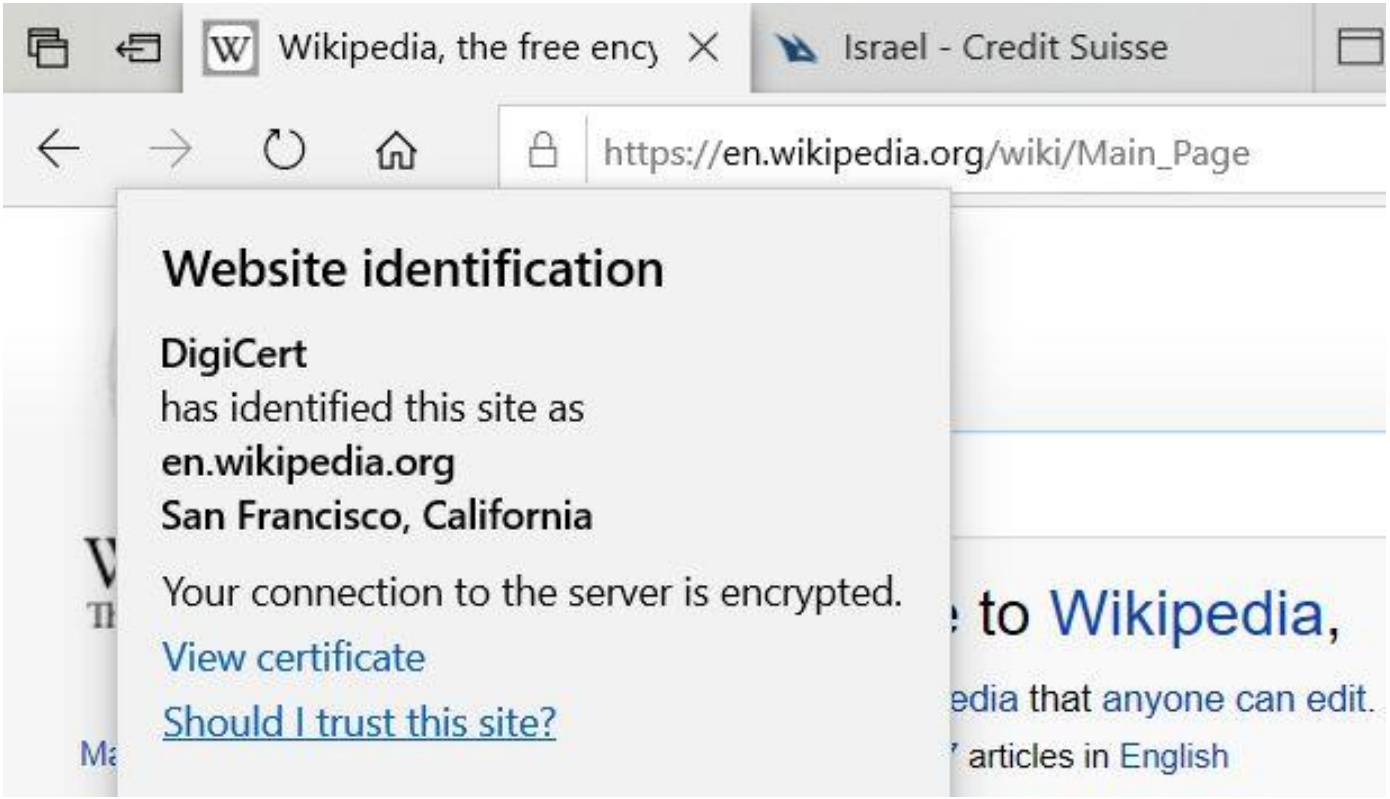
Different Levels of trust

- Chrome (Google) and Edge (MS) show different levels:
- Domain validated (the email in the certificate works):



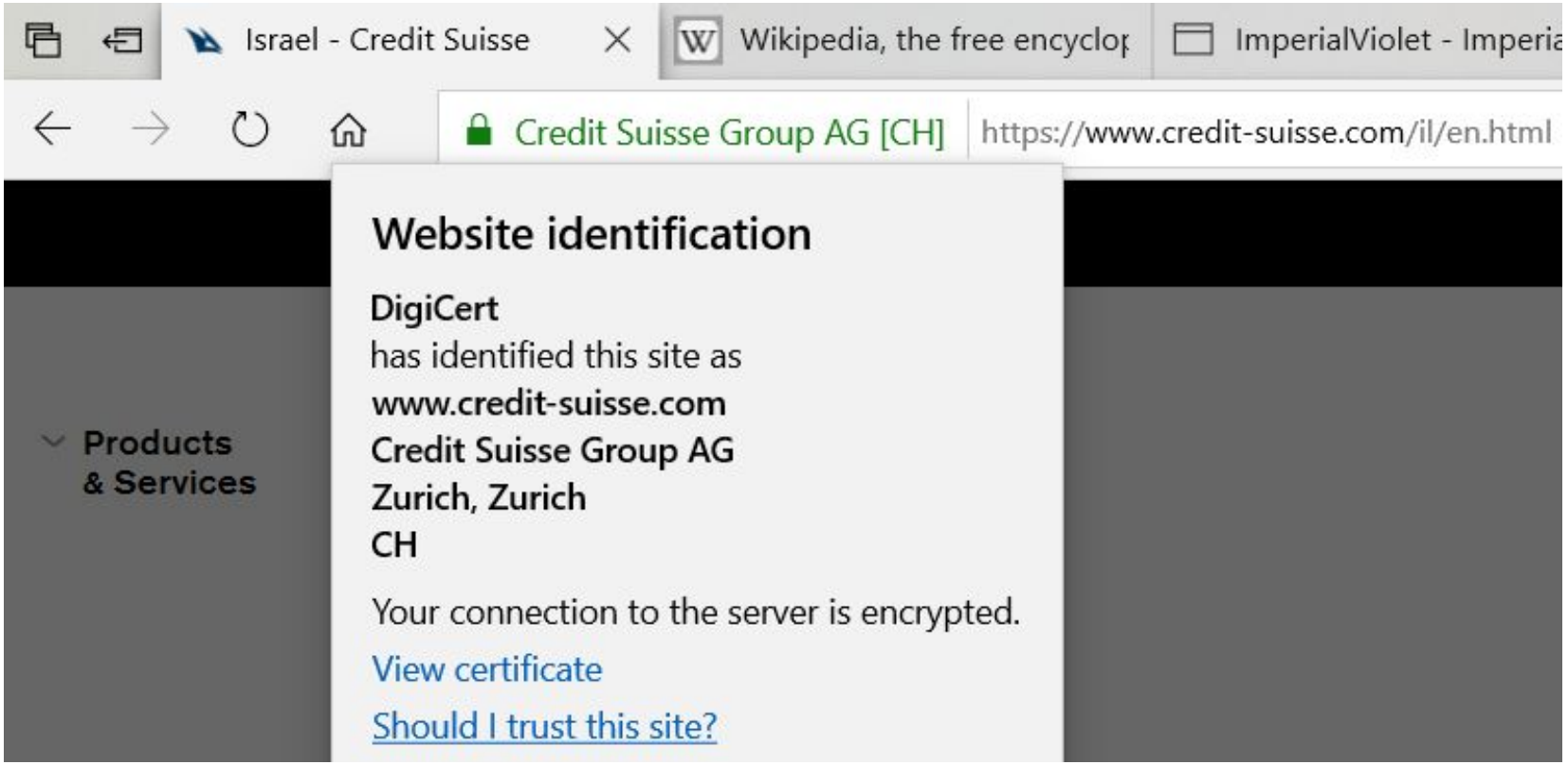
Different Levels of trust

- Premium, High Assurance, Organization Validated (the CA checked out the organization):

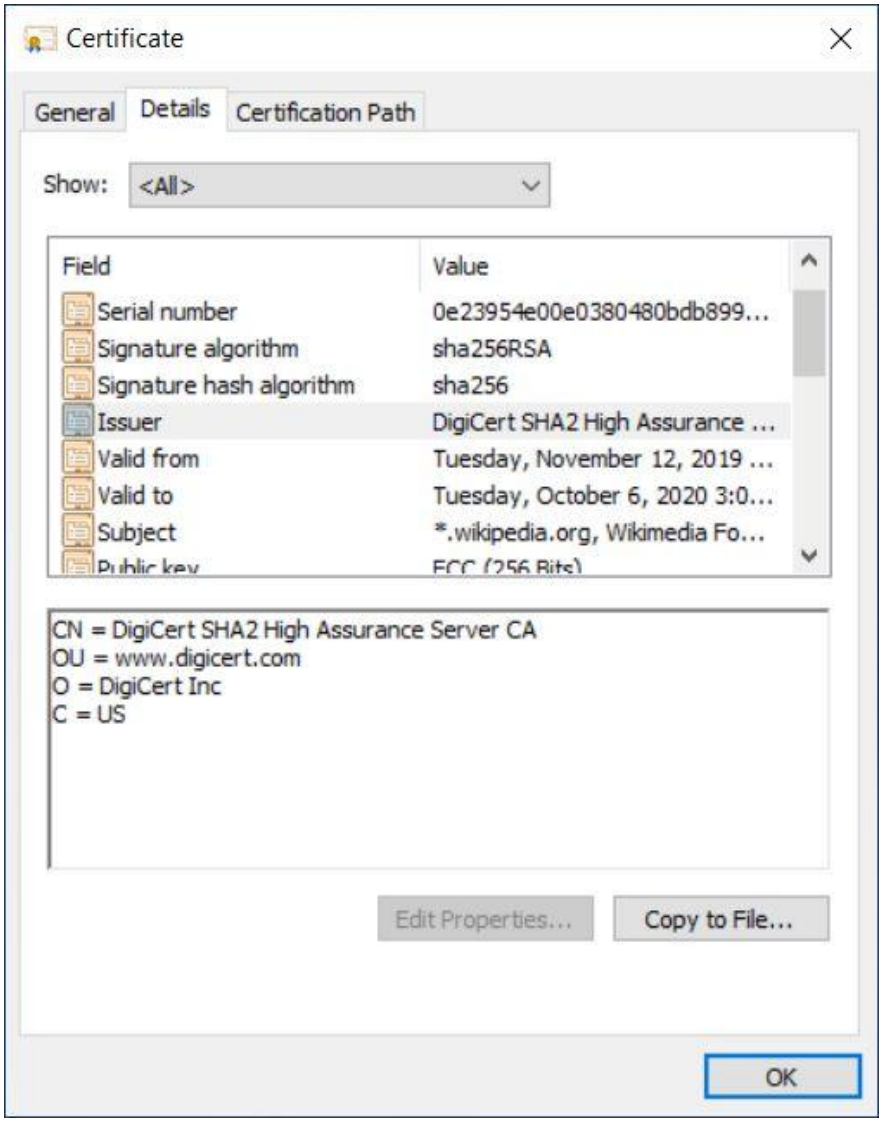


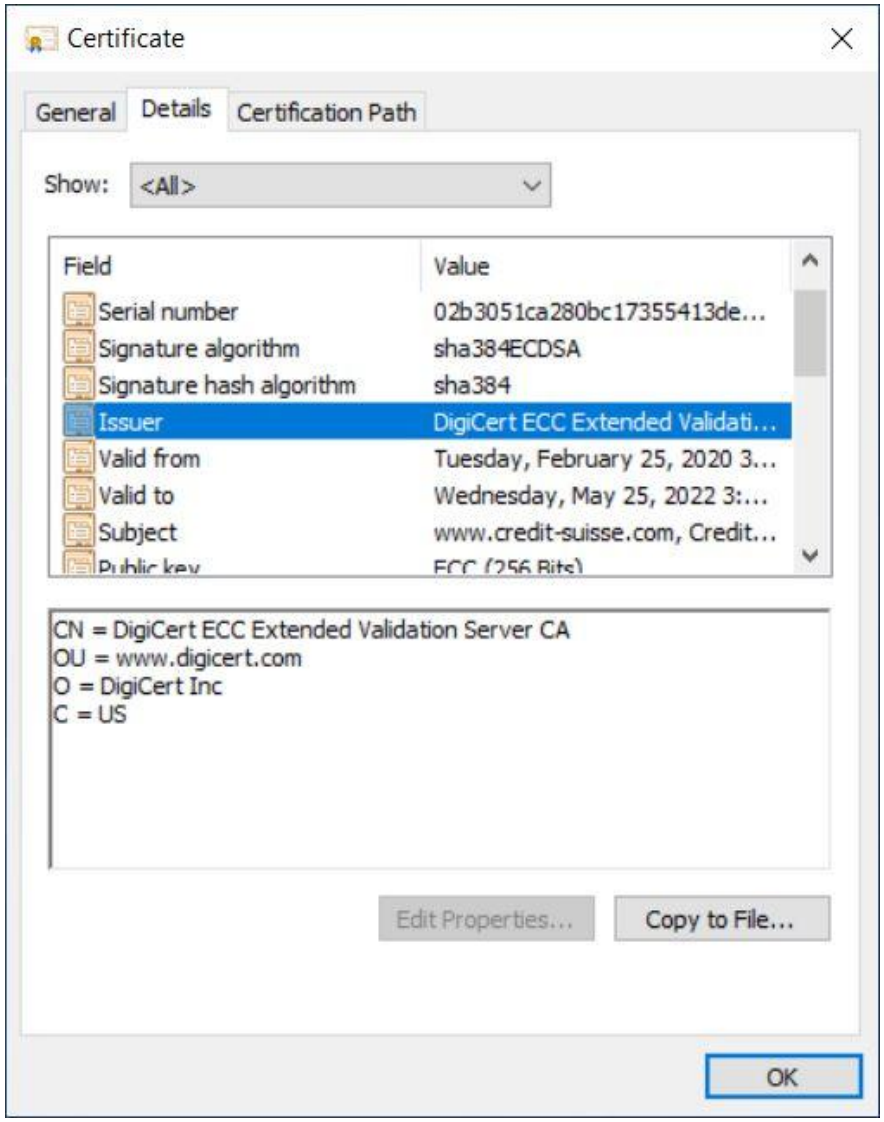
Different Levels of trust

- Extended Validation (in depth investigation):
  - https://cabforum.org/extended-validation/



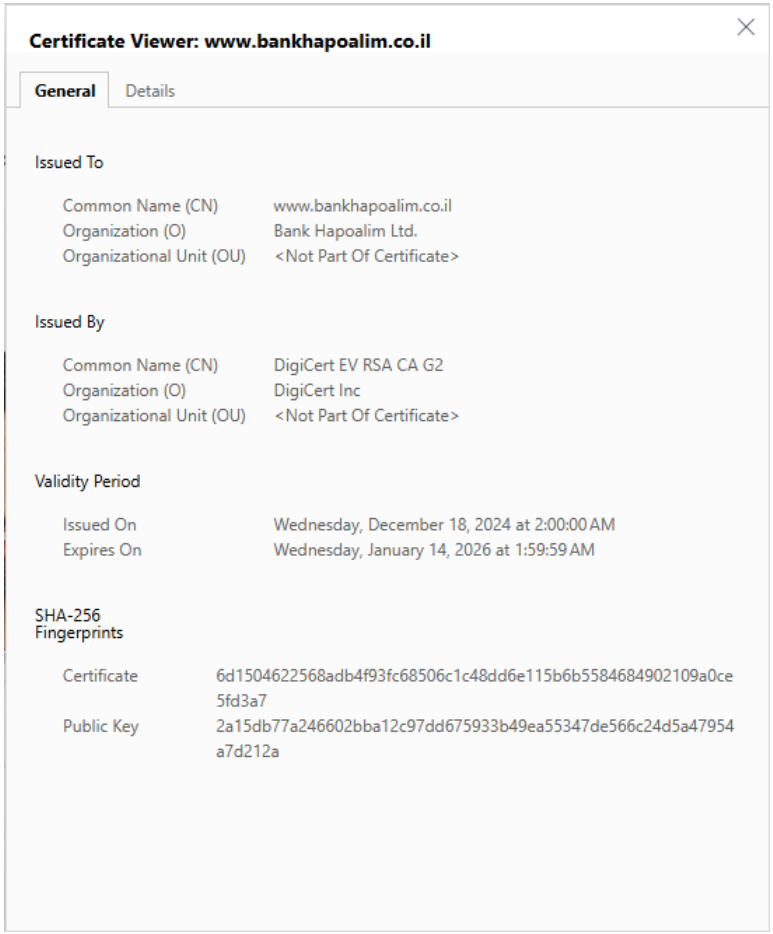
To Dig a Bit Deeper



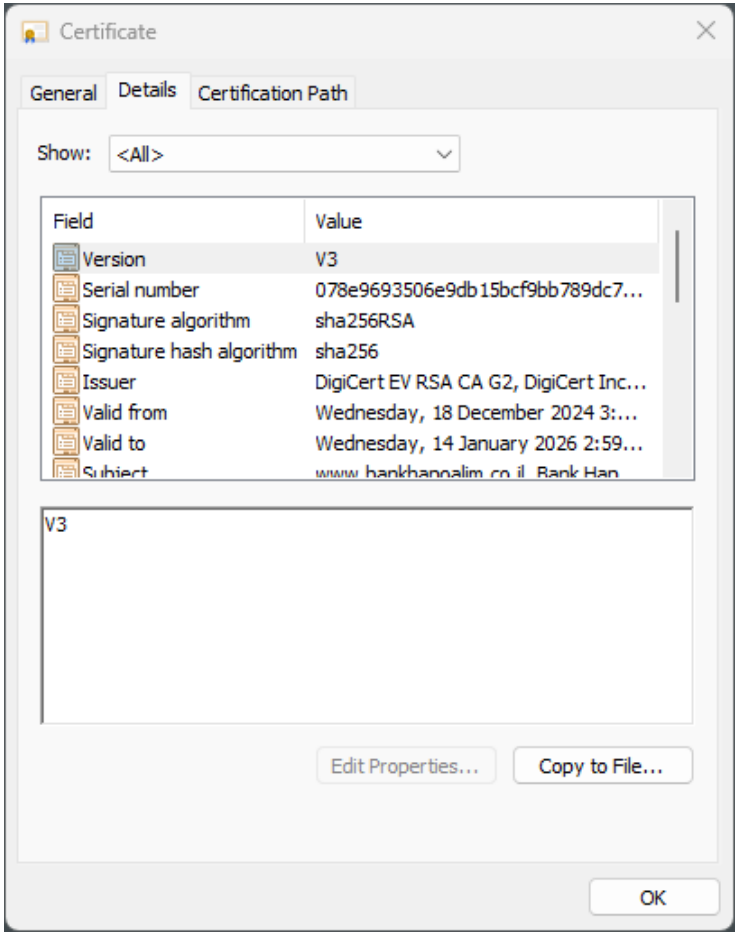


Certificate Tour

Bank Hapoalim Summary

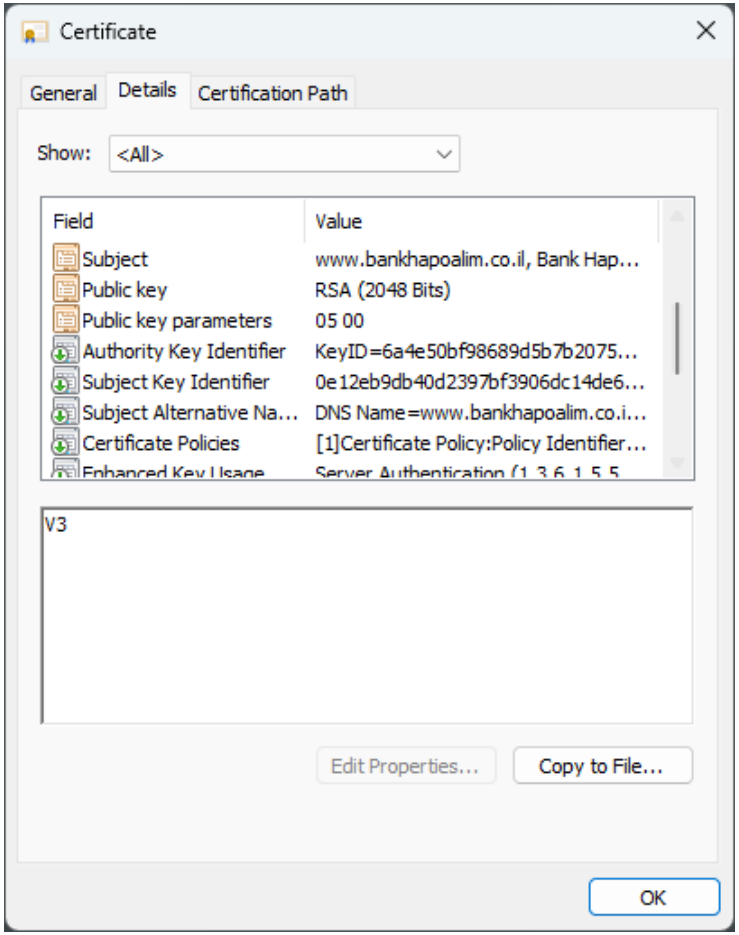


Some fields

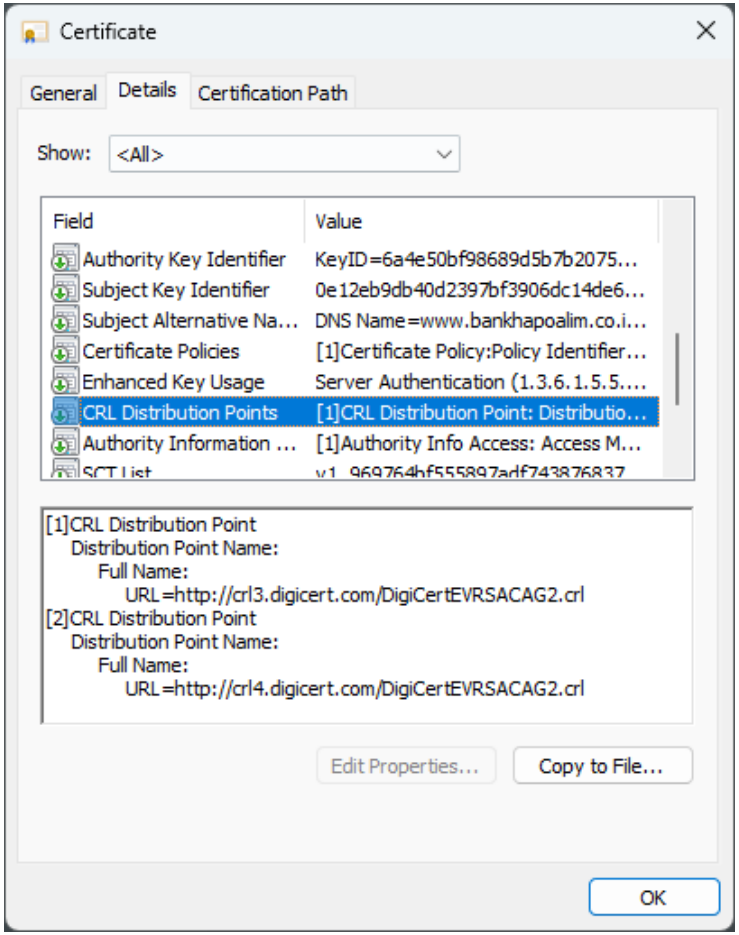


Certificate Tour

More fields

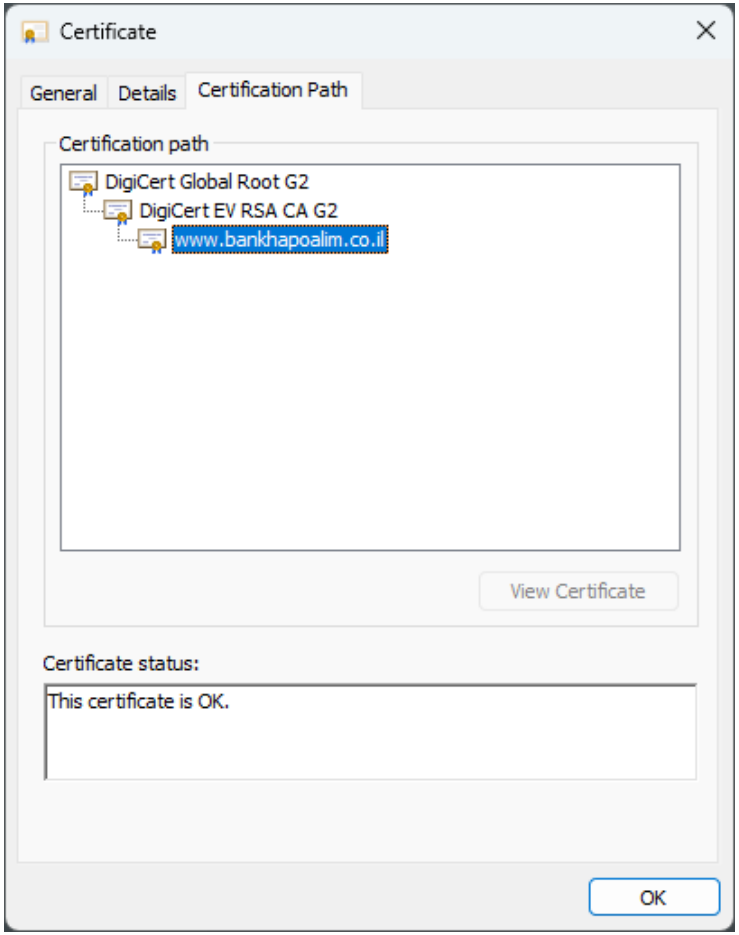


Certificate Revocation List

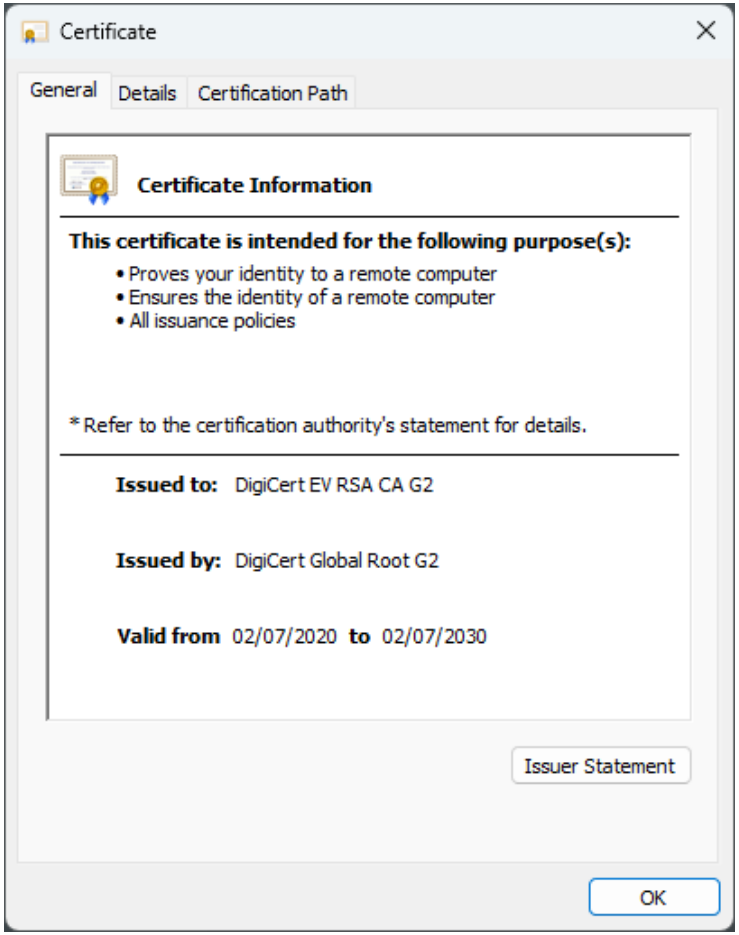


Certificate Tour

Hierarchy/Chain

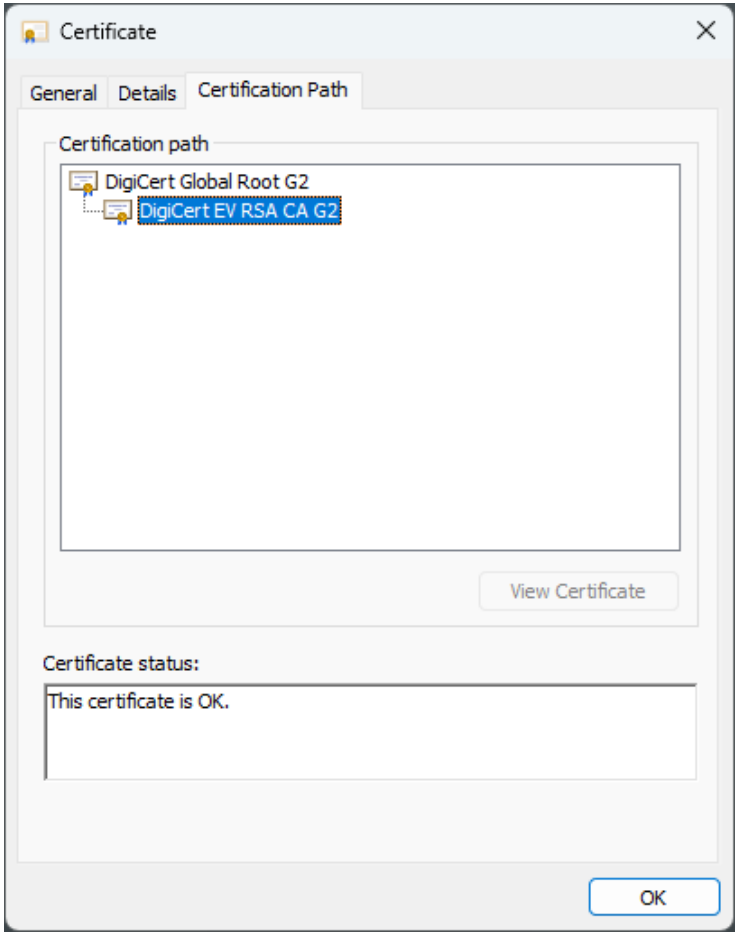


Parent Certificate

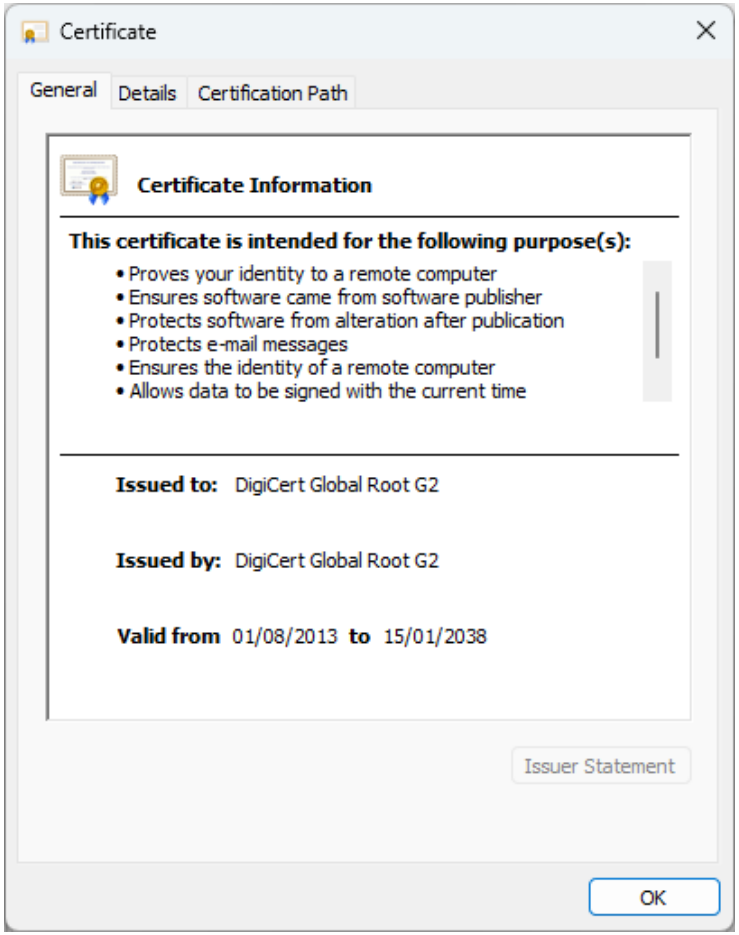


Certificate Tour

Hierarchy/Chain



Grandparent Certificate



Top-level Certificates

To check an X.509 certificate, one needs to have the public key of the issuer.

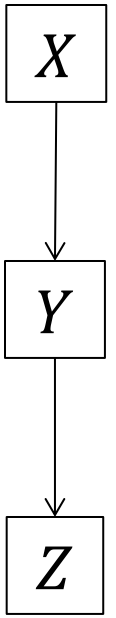
Such certificates can be “self-signed” by top-level, trusted CAs

In practice, companies like DigiCert pay web browser developers to include such certificates in browser releases.

Certificate Chains

- Notation: $X\ \langle\langle Y \rangle\rangle$ means the certificate of principal Y issued by authority X.

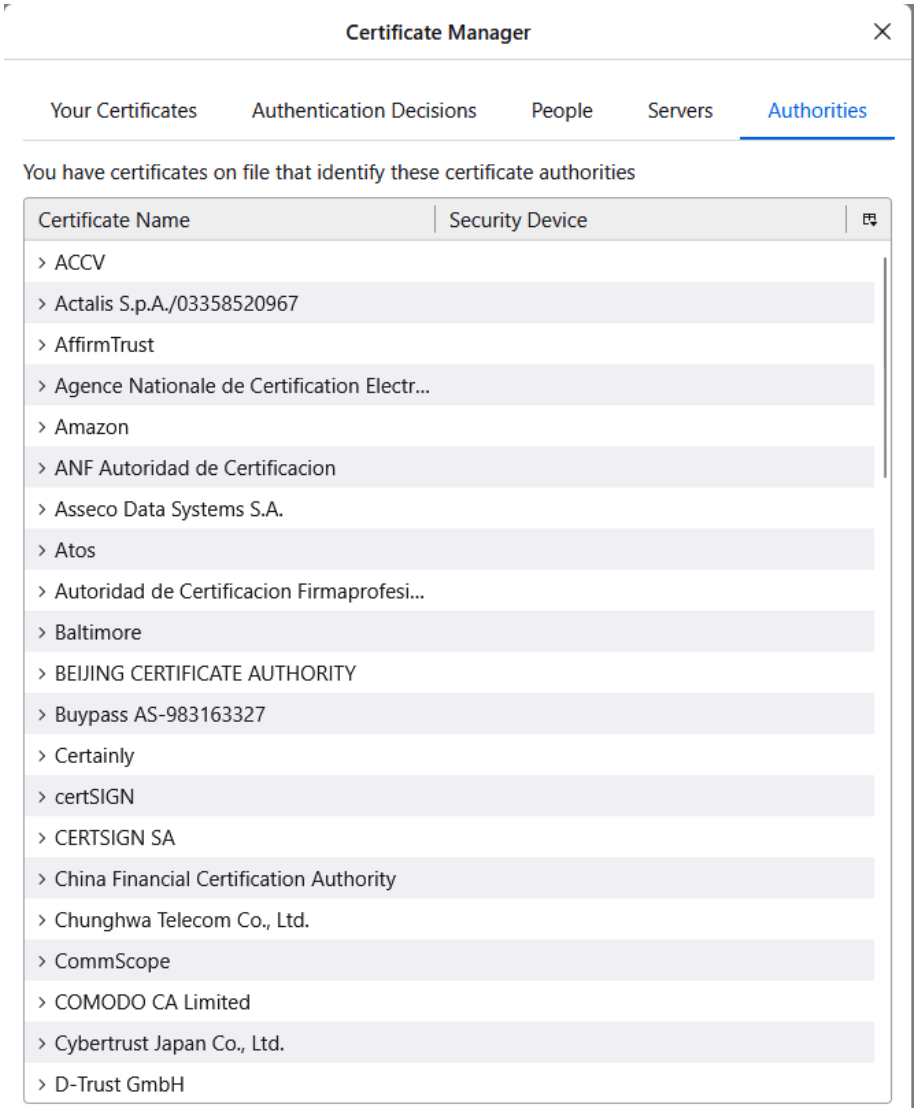
- One can create *certificate chains* to delegate authentication duties among principals:



- Example: $X\langle\langle Y \rangle\rangle, Y\langle\langle Z \rangle\rangle$
  - These two certificates together allow a principal who trusts X to verify the authenticity of the identity of Z.

- Chains can be arbitrarily long.
  - CAs can attest to each other's identities via peering agreements

Firefox Roots (75 total)



Windows 11 Roots (74 total)

Issued To	Issued By	Expiration Date	Intended Purposes	Friendly Name
AAA Certificate Services	AAA Certificate Services	01/01/2029	Client Authentication, Code Signing, Encr...	Sectigo (AAA)
Actalis Authentication Root CA	Actalis Authentication Root CA	22/09/2030	Client Authentication, Code Signing, Sec...	Actalis Authenticati...
AddTrust External CA Root	AddTrust External CA Root	30/05/2020	Client Authentication, Code Signing, Encr...	Sectigo (AddTrust)
Amazon Root CA 1	Amazon Root CA 1	17/01/2038	Client Authentication, Document Signing...	Amazon Root CA 1
Baltimore CyberTrust Root	Baltimore CyberTrust Root	13/05/2025	Client Authentication, Code Signing, Sec...	DigiCert Baltimore ...
Certification Authority of WoSign	Certification Authority of WoSign	08/08/2039	Client Authentication, Code Signing, Sec...	WoSign
Certum CA	Certum CA	11/06/2027	Client Authentication, Code Signing, Sec...	Certum
Certum Trusted Network CA	Certum Trusted Network CA	31/12/2029	Client Authentication, Code Signing, Encr...	Certum Trusted Net...
Class 3 Public Primary Certificat...	Class 3 Public Primary Certificatio...	02/08/2028	Client Authentication, Code Signing, Sec...	VeriSign Class 3 Pu...
COMODO RSA Certification Au...	COMODO RSA Certification Auth...	19/01/2038	<All>	<None>
Copyright (c) 1997 Microsoft C...	Copyright (c) 1997 Microsoft Corp.	31/12/1999	Time Stamping	Microsoft Timesta...
DigiCert Assured ID Root CA	DigiCert Assured ID Root CA	10/11/2031	Client Authentication, Code Signing, Sec...	DigiCert
DigiCert CS RSA4096 Root G5	DigiCert CS RSA4096 Root G5	15/01/2046	Code Signing, Time Stamping	DigiCert CS RSA409...
DigiCert Global Root CA	DigiCert Global Root CA	10/11/2031	Client Authentication, Code Signing, Sec...	DigiCert
DigiCert Global Root G2	DigiCert Global Root G2	15/01/2038	Client Authentication, Code Signing, Sec...	DigiCert Global Roo...
DigiCert Global Root G3	DigiCert Global Root G3	15/01/2038	Client Authentication, Code Signing, Sec...	DigiCert Global Roo...
DigiCert High Assurance EV Ro...	DigiCert High Assurance EV Root ...	10/11/2031	Time Stamping, Secure Email, Code Signi...	DigiCert
DigiCert Trusted Root G4	DigiCert Trusted Root G4	15/01/2038	Client Authentication, Code Signing, Sec...	DigiCert Trusted Ro...
DST Root CA X3	DST Root CA X3	30/09/2021	Client Authentication, Document Signing...	DST Root CA X3
Entrust Code Signing Root Certi...	Entrust Code Signing Root Certifi...	30/12/2040	Code Signing, Time Stamping	Entrust Code Signin...
Entrust Root Certification Auth...	Entrust Root Certification Authority	27/11/2026	Client Authentication, Code Signing, Encr...	Entrust
Entrust Root Certification Auth...	Entrust Root Certification Authori...	18/12/2037	Client Authentication, Code Signing, Sec...	Entrust Root Certifi...
Entrust Root Certification Auth...	Entrust Root Certification Authori...	07/12/2030	Client Authentication, Code Signing, Encr...	Entrust.net
Entrust.net Certification Author...	Entrust.net Certification Authority...	24/07/2029	Client Authentication, Code Signing, Encr...	Entrust (2048)
ePKI Root Certification Authority	ePKI Root Certification Authority	20/12/2034	Client Authentication, Code Signing, Encr...	Chunghwa Teleco...
GeoTrust Global CA	GeoTrust Global CA	21/05/2022	Client Authentication, Code Signing, Sec...	GeoTrust Global CA
GlobalSign	GlobalSign	18/03/2029	<All>	<None>
GlobalSign	GlobalSign	18/03/2029	Client Authentication, Code Signing, Encr...	GlobalSign Root CA...
GlobalSign	GlobalSign	10/12/2034	Client Authentication, Code Signing, Doc...	GlobalSign Root CA...
GlobalSign	GlobalSign	19/01/2038	Client Authentication, Secure Email, Serve...	GlobalSign
GlobalSign	GlobalSign	19/01/2038	Client Authentication, Code Signing, Sec...	GlobalSign ECC Ro...

So Far

- Certificates and PKI
- Certificate Types
- Certificate Validation
  - OCSP
  - Certificate Pinning
  - Certificate Transparency

Certificate Revocation

- What if the CA needs to revoke a certificate?
  - Key compromise
  - Name expired
- Happens all the time (can get a new certificate in a few minutes)
- Some options:

Wait for expiration	Manual Notification • works in small systems	Revocation Certificate • Like regular Certificate, with a “Revoke” note inside

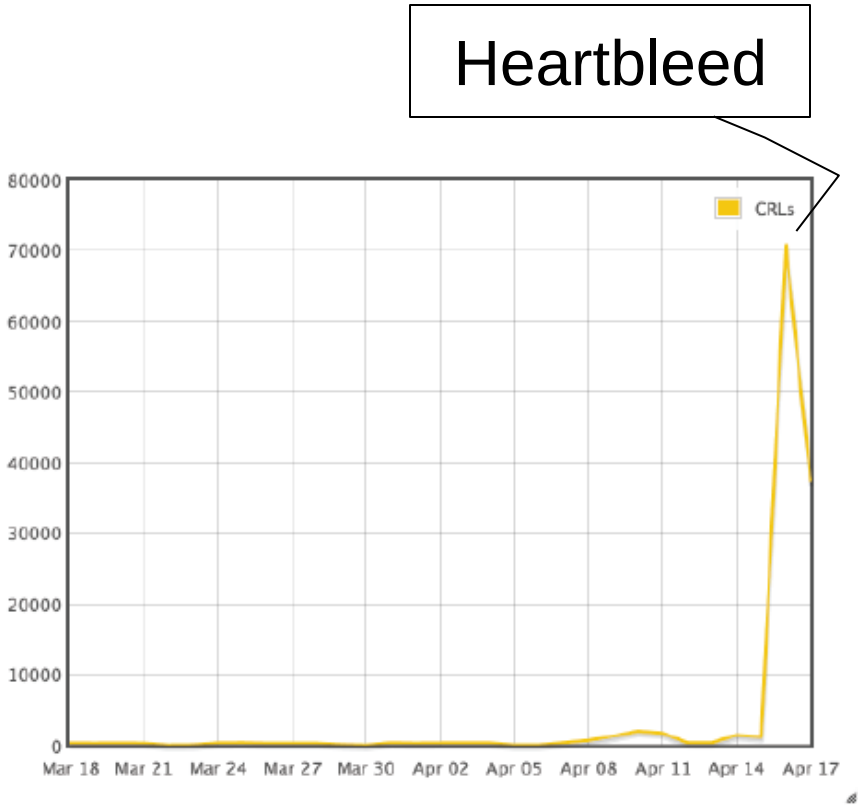
Public file of revoked keys

- Certificate Revocation List (CRL)

Certificate Revocation Lists

- CRLs are a common mechanism
- Must be signed by the CA
  - Why?
- Include timestamp and refreshed regularly
  - Why?
- When the CRL gets large, segment it
  - Put up only new revocations (*delta-CRL*)
  - Divide up the CRL by reason for revocation
  - Pre-assign each certificate to a given CRL “bucket” in case of revocation

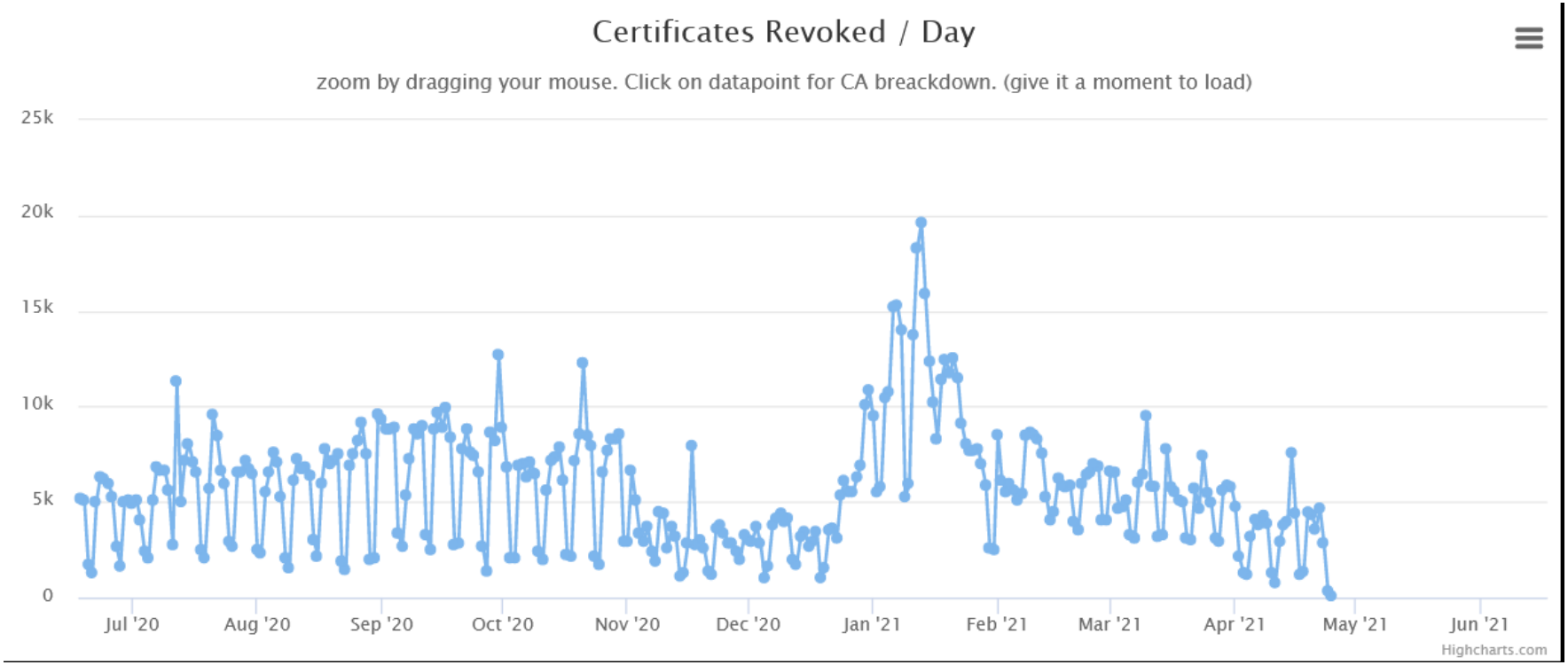
CRLs Cost Money



Source: isc.sans.edu

- After Heartbleed (2014), CloudFare revoked and reissued all certificates → CRL for GlobalSign grew from 22KB to 4.7MB
- 40Gpbs of new traffic
  - Costs $400,000
  - Using Amazon's AWS: $952,992.40 per month

CRLs Up and Downs



Source: isc.sans.edu

So Far

- Certificates and PKI
- Certificate Types
- Certificate Validation
  - OCSP
  - Certificate Pinning
  - Certificate Transparency

OCSP

- Revocation is hard:
  - Certificate Revocation Lists (CRLs)
    - Most certificates are revoked for non-security reasons
  - ...so CRLs are often huge (problem especially on mobile)
    - Download MBs of CRL for each SSL connection?

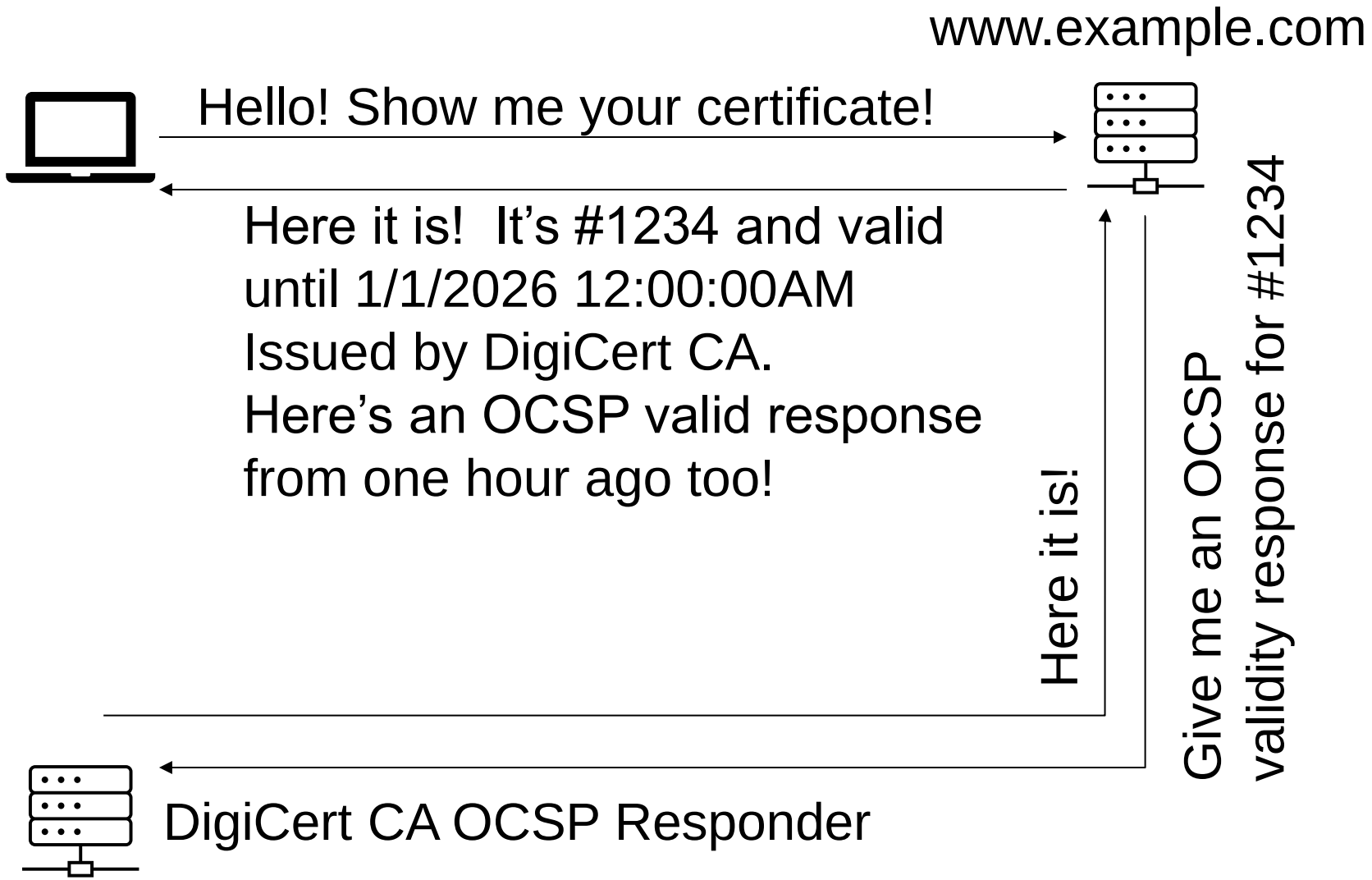
- *Online Certificate Status Protocol (OCSP):*
  - Online means now (TLS handshake opens side channel to OCSP responder)
  - … so what do you do if the online responder is not responding?
    - Could an attacker just block the side channel?
    - Typical response time: 430ms
  - OCSP implemented by IE, Firefox, turned off by Chrome (privacy)

- OCSP stapling: not widely deployed
  - The CA issues a time-stamped OCSP validity claim which is sent along with the SSL handshake

OCSP Basics


www.example.com
Hello! Show me your certificate!
Here it is! It's #1234 and valid
until 1/1/2026 12:00:00AM
Issued by DigiCert CA.
Is certificate #1234
still valid?
Yes
DigiCert CA OCSP Responder

OCSP Stapling Basics

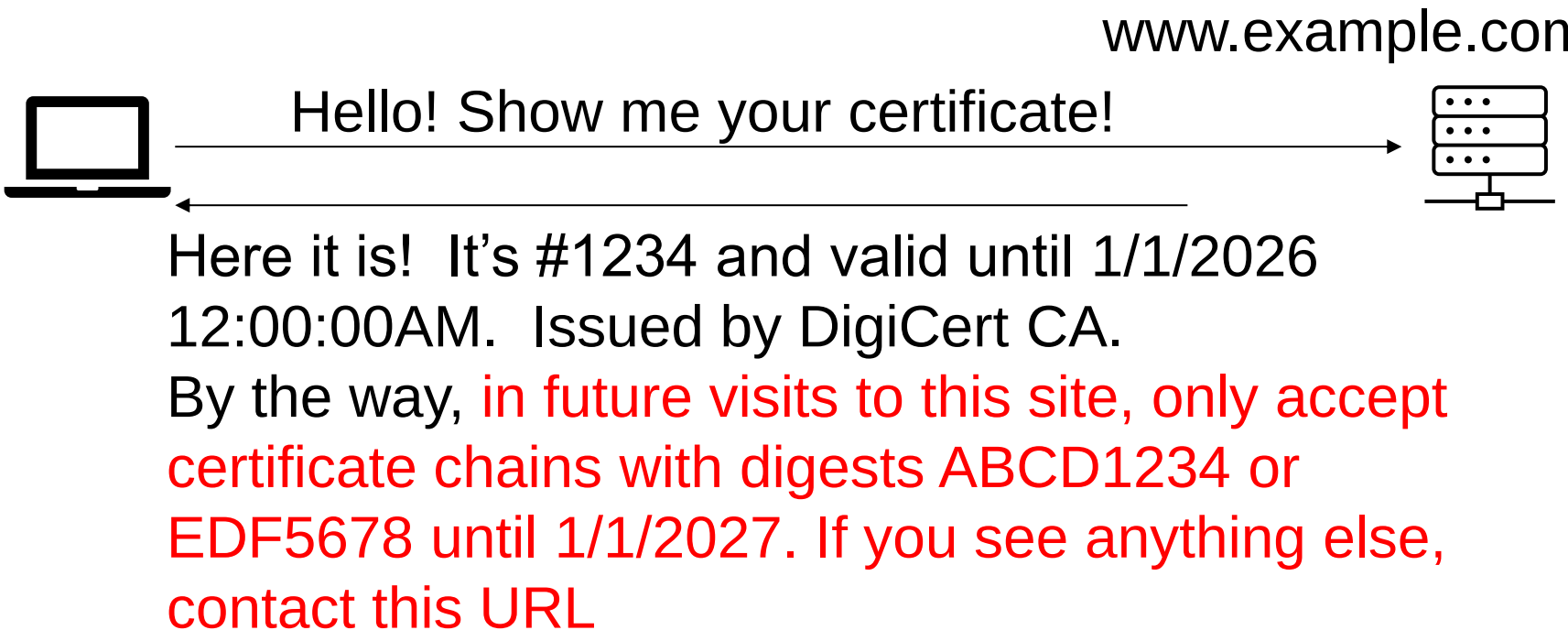


So Far

- Certificates and PKI
- Certificate Types
- Certificate Validation
  - OCSP
  - Certificate Pinning
  - Certificate Transparency

Certificate Pinning

- Called HTTP Public Key Pinning (no longer in use)
- Basic idea



So Far

- Certificates and PKI
- Certificate Types
- Certificate Validation
  - OCSP
  - Certificate Pinning
  - Certificate Transparency

Certificate Transparency

What it does:

1. Make all public end-entity TLS certificates public knowledge

2. Hold CAs publicly accountable for all certificates they issue.

Explicit anti-goal:

Certificate Transparency will not introduce another trusted third party.

There already are too many “trusted parties” out there

Big problem with certificates

Browsers trust several hundred root CA certificates

Any CA can issue on behalf of any domain.

Any CA can issue intermediate CA certificates that can issue on behalf of every domain.

100s of equally trusted third parties!

Corollary: Website owners must trust every single CA, even the ones they don't do business with.

Cautionary tale 1

DigiNotar
Internet Trust Services

- July 19th, 2011: DigiNotar CA finds evidence of compromise through routine daily check.

- Evidence of large-scale Man in the Middle attack in July
  - By the end of July, thought everything was under control.

- Rogue certificates hit OCSP responders `*.google.com` pinning failure externally reported August 28th.
  - Certificate pinning: The browser knows what certificate authority or certificate should be provided in SSL handshake (may be installed or stored the first time)
- Rogue `*.google.com` cert revoked and Chrome updated August 29th.

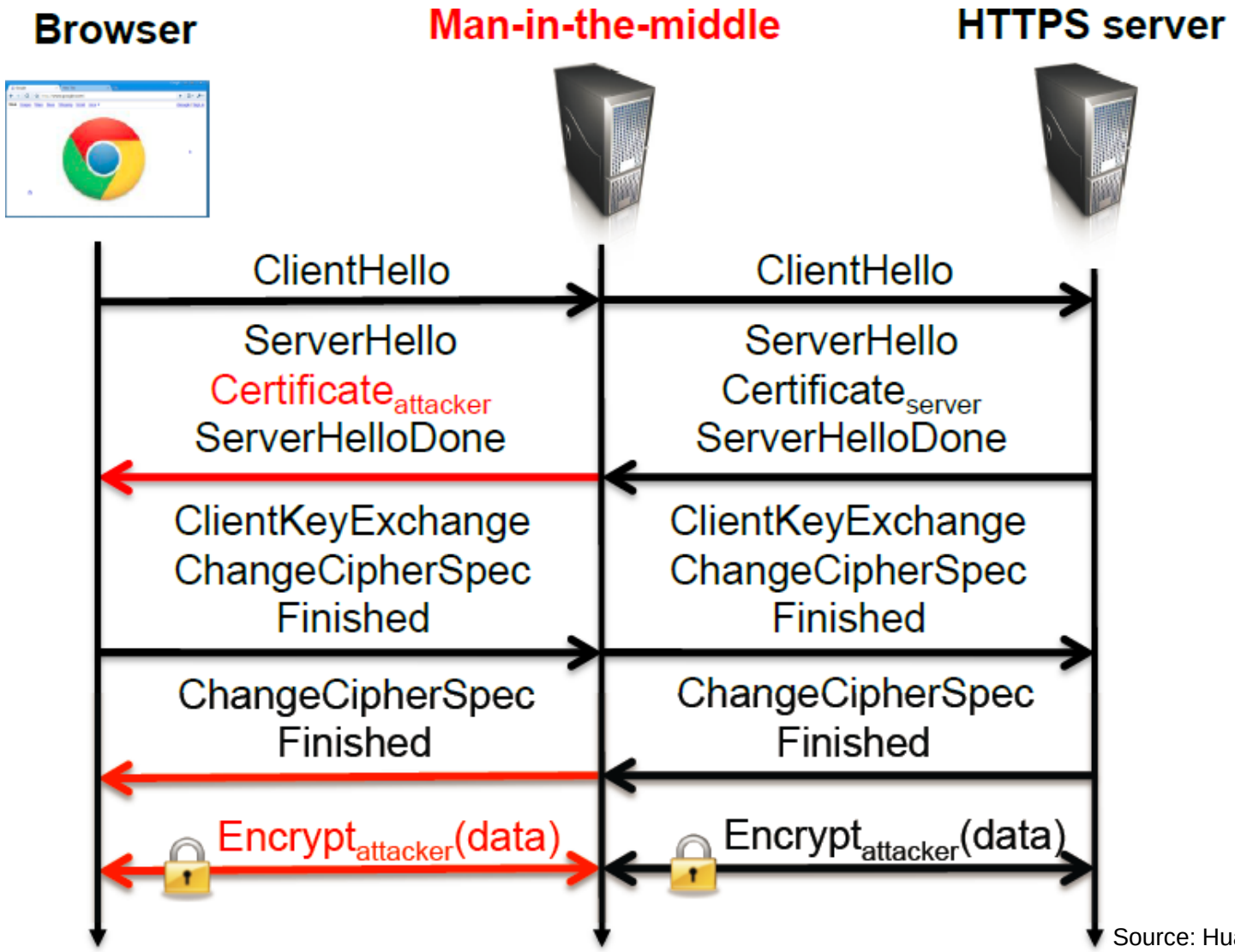
Cautionary Tale 2

- August 2011: TURKTRUST CA mistakenly issues two intermediate CA certificates
  - `CA = true` is just one bit in a regular certificate.

- `*.google.com` certificate issued by the intermediate detected December 24, 2012.

- Certificate revoked and Chrome updated December 25, 2012.

SSL Man In The Middle



Source: Huang, et al. 2014

Yes, it happened

IMAGE: SCOTT RODGERSON VIA UNSPLASH

Alexander Martin

October 23rd, 2023

News Privacy Technology

Alleged covert wiretap on Russian messaging service blown by expired TLS certificate

Security researchers have discovered what they believe may be a government attempt to covertly wiretap an instant messaging service in Germany — an attempt that was blown because the potential intercepting authorities failed to reissue a TLS certificate.

The suspected man-in-the-middle attack was identified when the administrator of jabber.ru, the largest Russian XMPP service, received a notification that one of the servers' certificates had expired.

However, jabber.ru found no expired certificates on the server — as explained in a blog post by ValdikSS, a pseudonymous anti-censorship researcher based in Russia who collaborated on the investigation.

The expired certificate was instead discovered on a single port being used by the service to establish an encrypted Transport Layer Security (TLS) connection with users. Before it had expired, it would have allowed someone to decrypt the traffic being exchanged over the service.

https://therecord.media/jabber-ru-alleged-government-wiretap-expired-tls-certificate

What went wrong?

Huge delay between incident and (public) response.

- Long window of opportunity for the attacker.

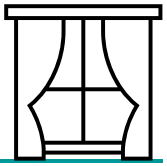
No incident detection mechanism

- First signs of DigiNotar Man in the Middle in the wild were for non-Google domains without pinning.

No automated incident reporting mechanism.

- Pinning failure reported manually (user sent it)
- Pinning reporting has improved since; but only the pinned domain can get reports for it (ex. Google can only get pinning reports for Google domains)
- Not all domains collect pinning data or reports

Goal: Reduce the Window



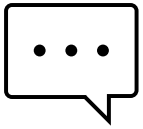
Minimize window between incident and response.

- May cost millions $ to get a forged certificate
- Can't prevent attacks, but we can make them more expensive by giving the attacker only one, short-lived shot.

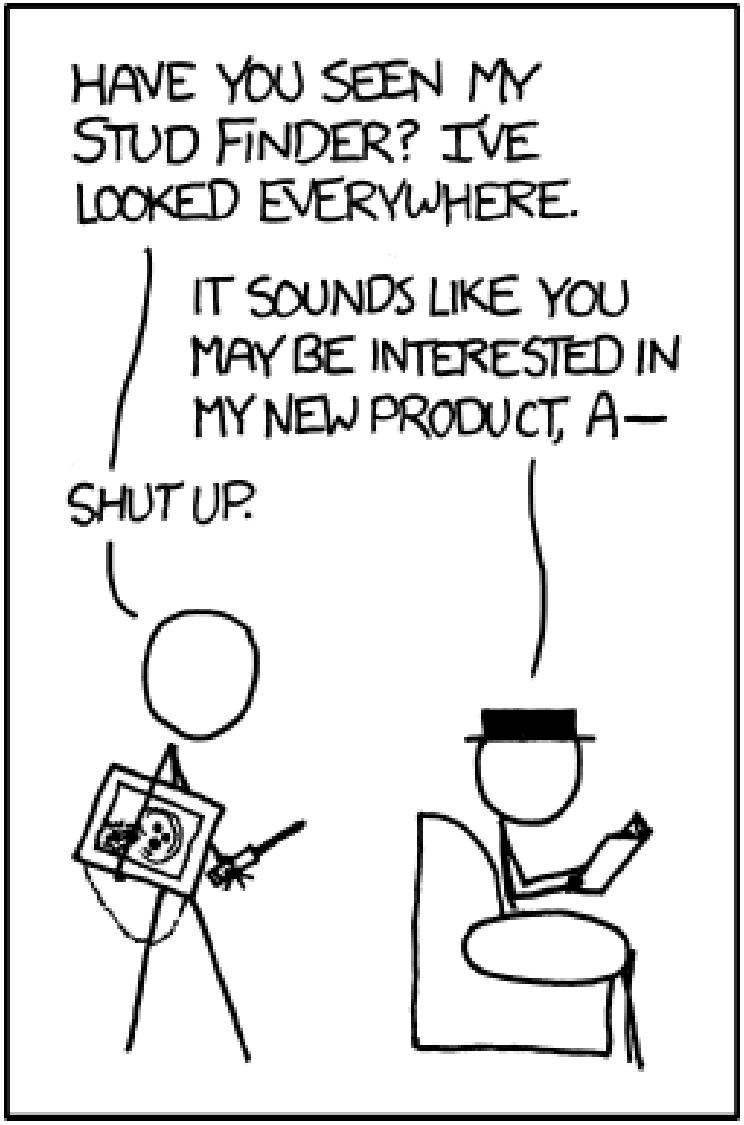
Only domain owners know which certificates are legitimate, so give them power

Make the computers *gossip*.

- Vaccination effect: not everyone has to participate for everyone to benefit.



Another Layer

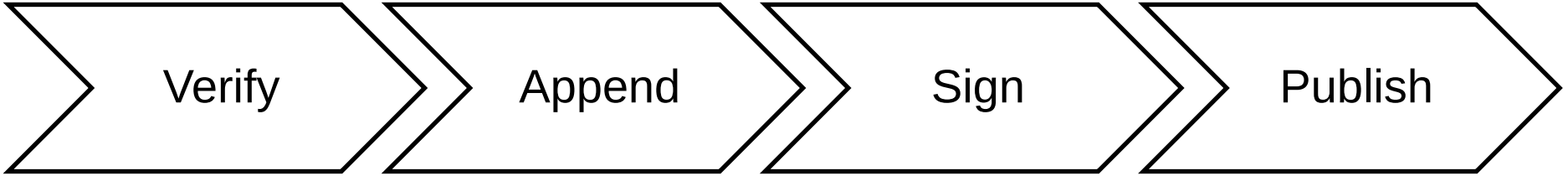


Source: xkcd.com

How does it work?

Central Feature: An append-only log of certificates

The log server:



- Verifies certificate chain.
- CA attribution for certificate mis-issuance
- Spam control
  - i.e. ignore self signed ones

- Periodically append new certs to append-only log
- Merkle Trees to prove nothing removed

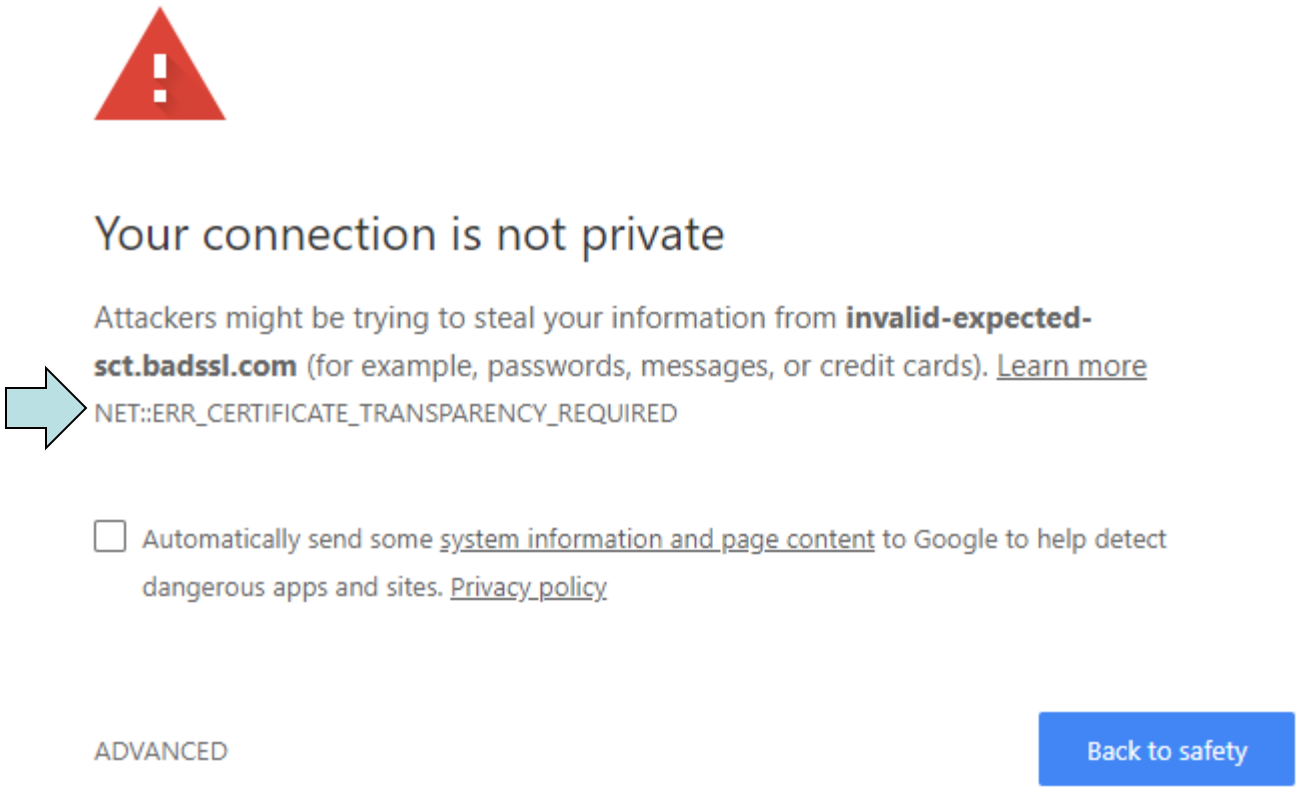
- Signs the log

- Publishes all updates of signed list of certs (“the log”) to the world.

Try it yourself

- https://no-sct.badssl.com/

When it fails (Chrome)



Some browsers ignore this error.

So the log is a TTP?

A CT log is designed to not be “super CA”:

Does **not** testify to the “goodness” of certificates; it just notes they were seen.

Public: everyone can inspect all the certificates.

Log is not trusted: since the log is signed, the fact that everyone sees the same list of certificates is cryptographically verifiable.

Sample log https://crt.sh/

CT Today

Mozilla Blog May 2025

Browser Requirements

Google Chrome 107 and later requires CT log inclusion for all certificates issued with a notBefore date of after 30 April 2018. Users will be prevented from visiting sites using non-compliant TLS certificates. Chrome had previously required CT inclusion for *Extended Validation* (EV) and Symantec-issued certificates.

Apple requires a varying number of SCTs in order for Safari and other servers to trust server certificates.

Firefox desktop from version 135 requires CT log inclusion for all certificates issued by certificate authorities in Mozilla's Root CA Program. Firefox for Android does not currently require CT log inclusion.

https://developer.mozilla.org/en-US/docs/Web/Security/Certificate_Transparency

TLDR:

- Google requires it
- Edge requires it
- Firefox requires it
  - But not Android

https://educatedguesswork.org/posts/transparency-part-2/

TLDR:

- CT is useful, but it's over engineered.
- Gossiping didn't work

Conclusion

- Certificates and PKI
- Certificate Types
- Certificate Validation
  - OCSP
  - Certificate Pinning
  - Certificate Transparency