Crypto Part 3: Certificates & TLS

CMSC 23200, Winter 2024, Lecture 6

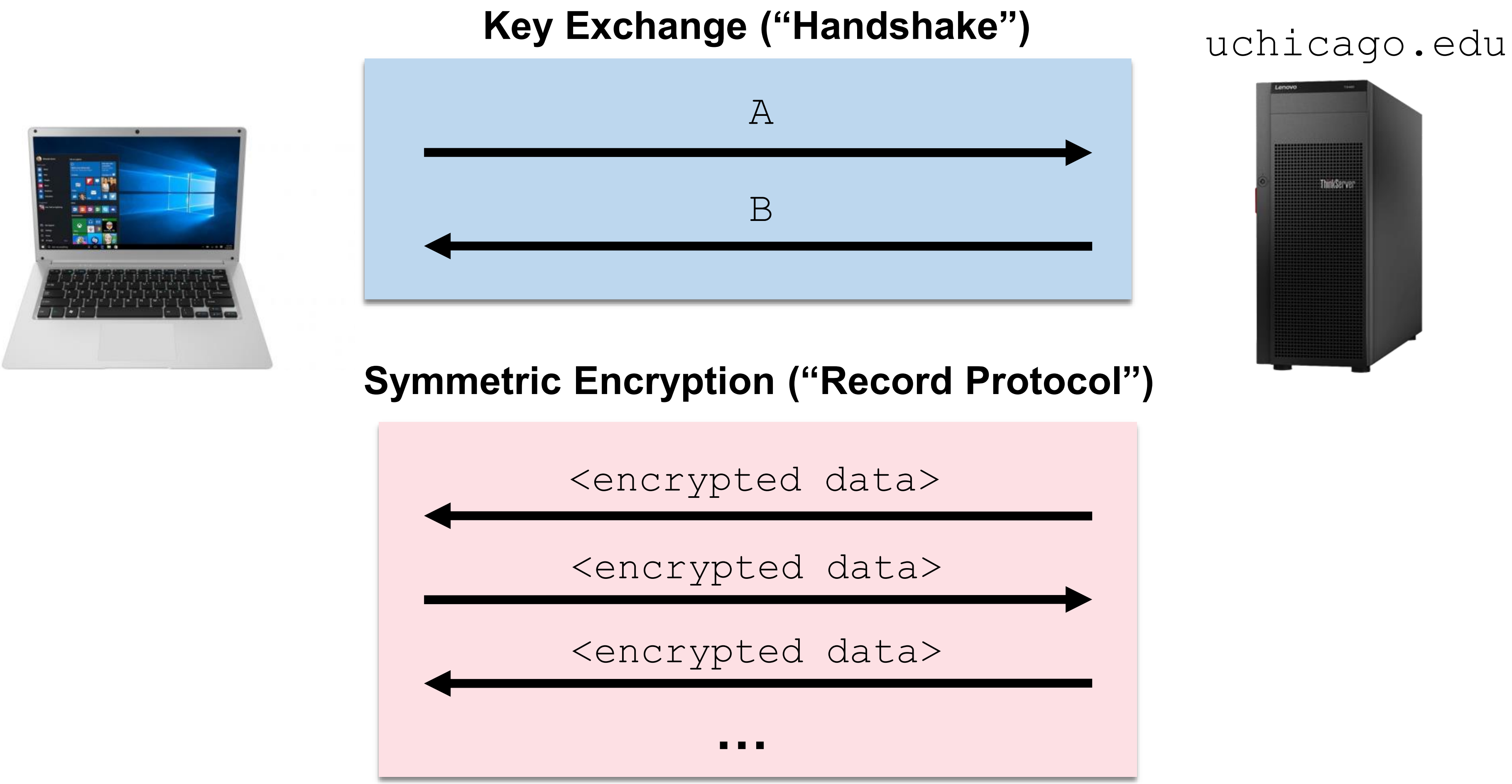
Grant Ho & Blase Ur

University of Chicago
(Slides adapted from David Cash)

Outline: Crypto Part 3

- Recap: Secure communication channels
- Authenticating endpoints: Certificates (Certs)
- Issuing Certs and Certificate Infrastructure (PKI)
- Attacks, Countermeasures
- Real World Secure Channels: SSL / TLS

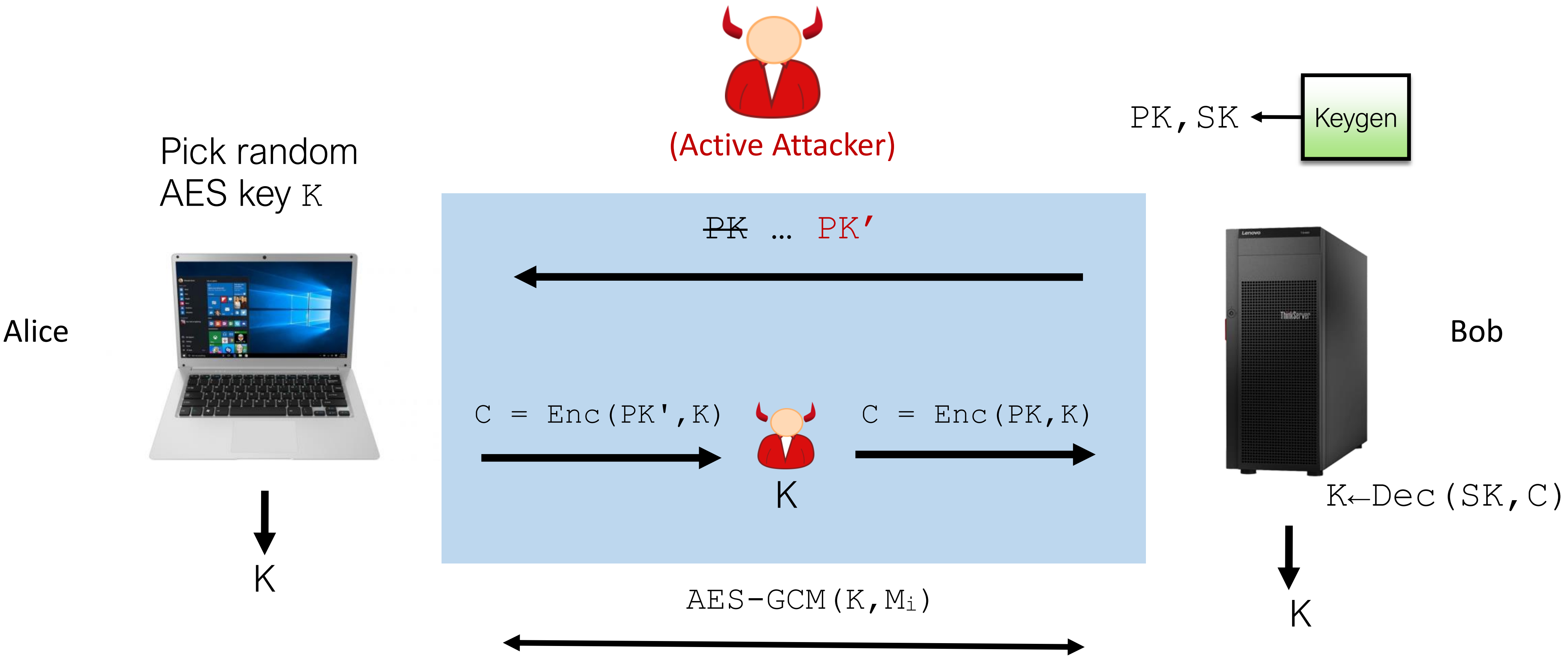
Template For Secure Channels (TLS, SSH, IPSec, …)



- Last lecture: Naïve strategy can be secure against *passive adversaries.*
- But the above template does <u>not</u> provide authentication & integrity.

Securing Key Exchange against Active Attackers

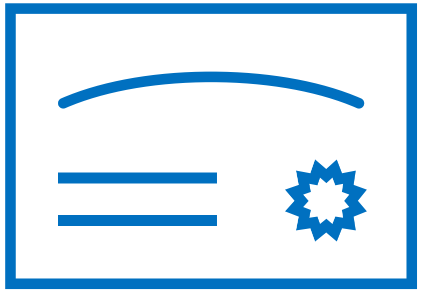
Key Challenge: Authenticity: How do we know that PK is really Bob's?



Authentication with Certificates (“Certs”)

Suppose we had a globally trusted entity, `BlaséInc`.

`BlaséInc` could issue **certificates (“certs”)** that state what other organizations’ public keys are.

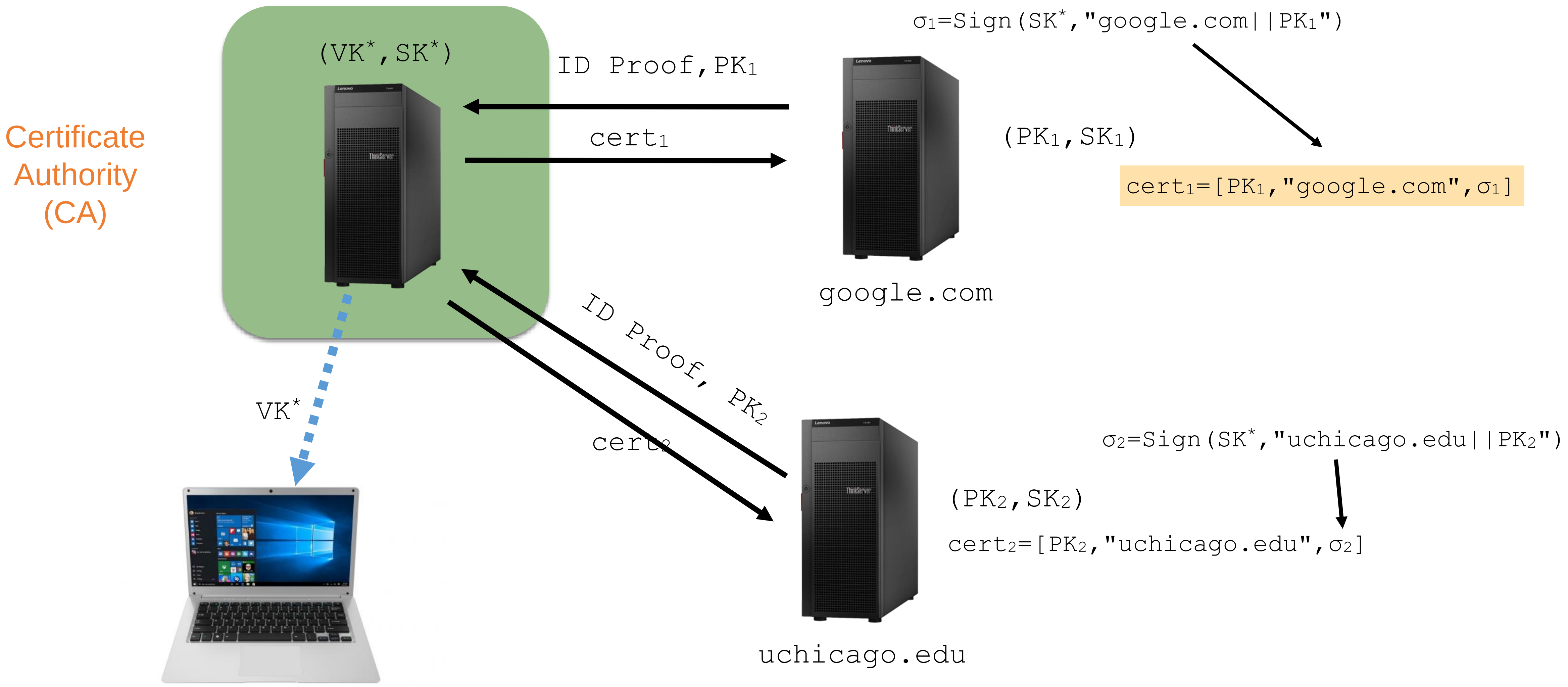


Cert = a document that says:

1. An Entity (e.g., Bob) has a public key that is:
2. `pk=0x7b5532…`, where the document is
3. signed using the `BlaséInc`’s private signing key

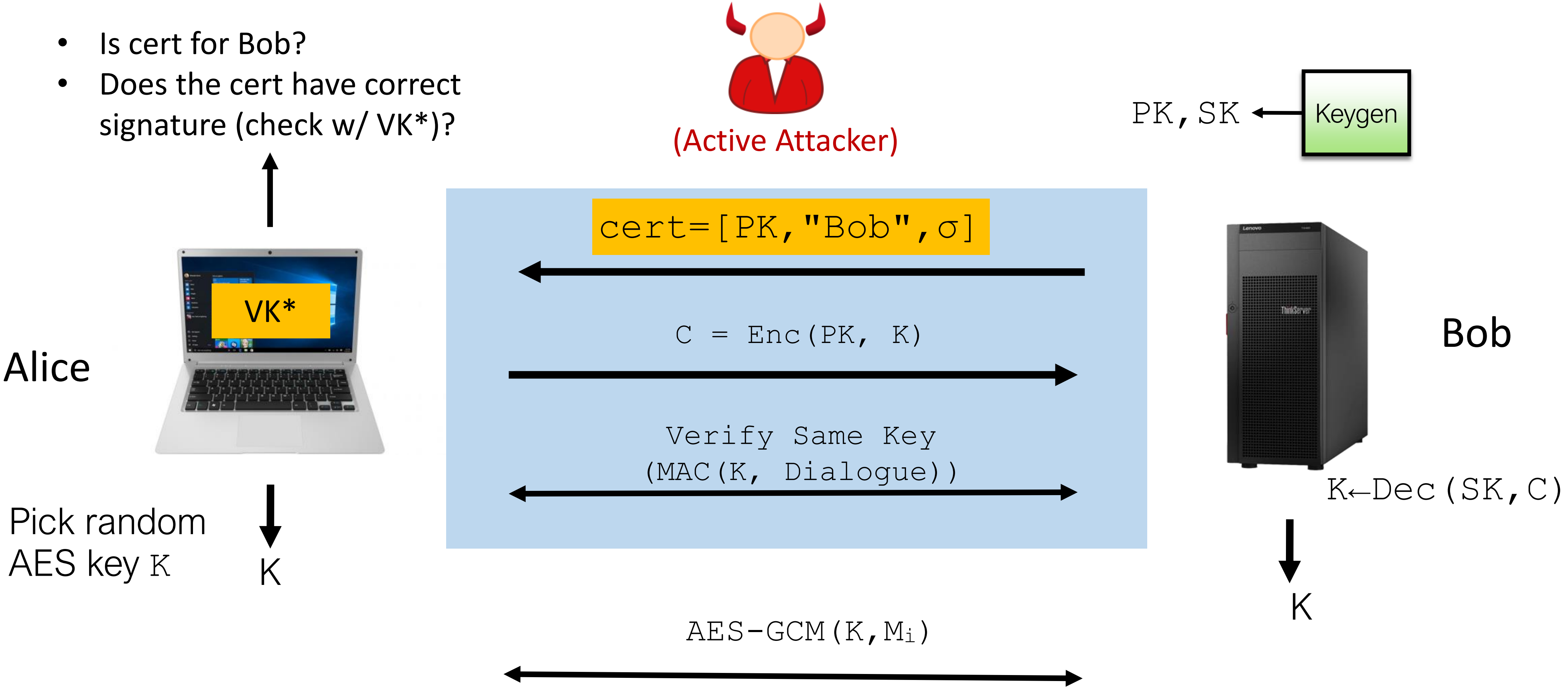
Trusted entity, `BlaséInc`, known as a **Certificate Authority (CA)**

Authentication with Certificates ("Certs")



VK* pre-installed on every machine by manufacturer or built into OS code.

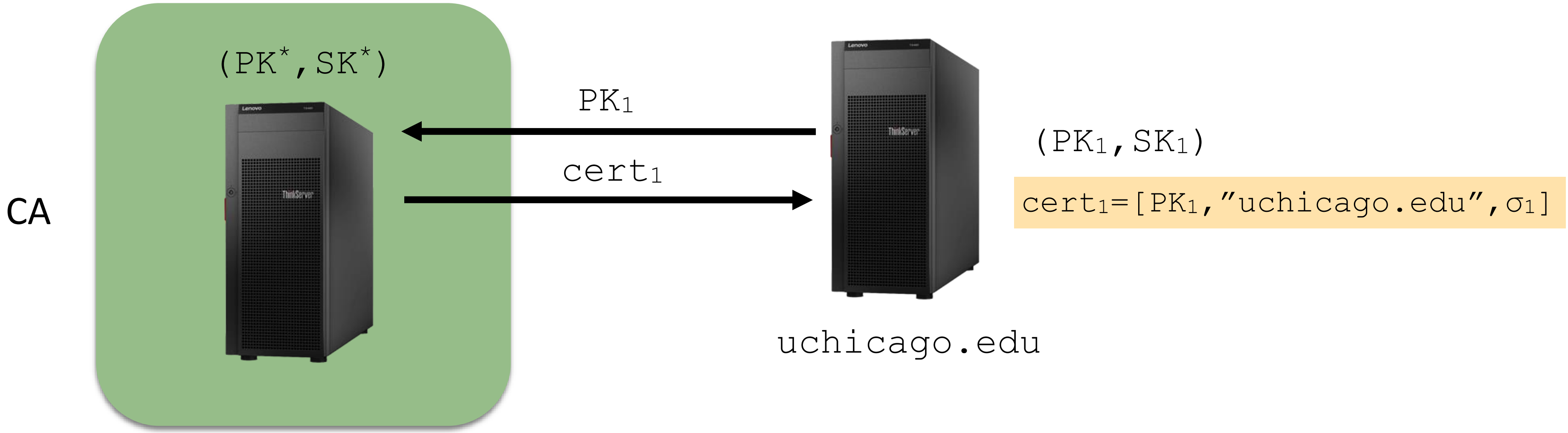
Securing Key Exchange against Active Attackers



Outline: Crypto Part 3

- Recap: Secure communication channels
- Authenticating endpoints: Certificates (Certs)
- Issuing Certs and Certificate Infrastructure (PKI)
- Attacks, Countermeasures
- Real World Secure Channels: SSL / TLS

Issuing Certificates: Validation

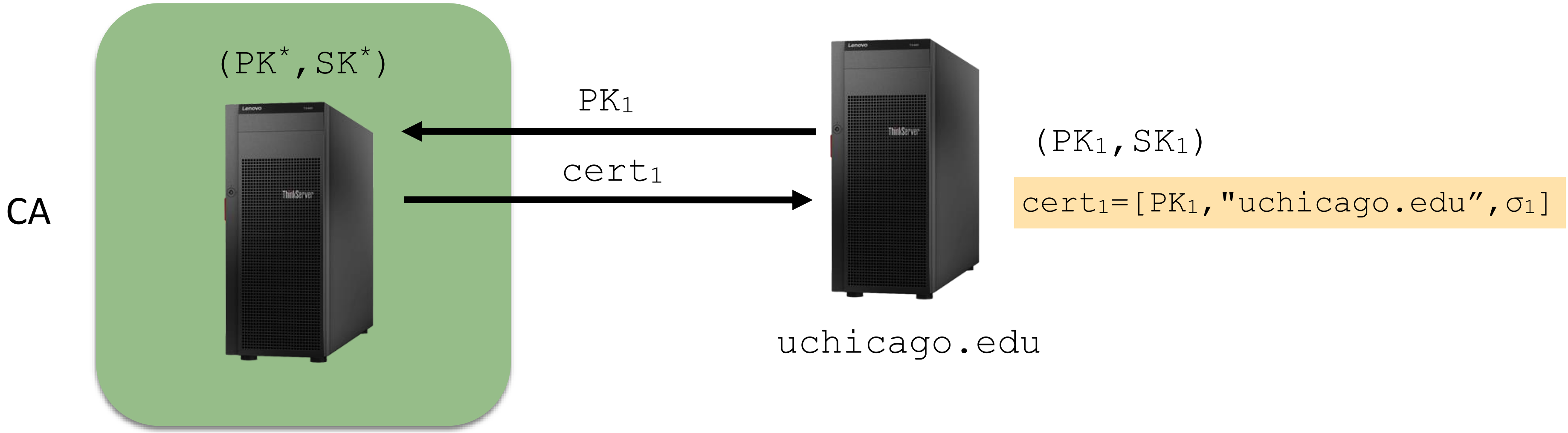


- CA must check that key PK_1 really does belong to "`uchicago.edu`"

Domain Validation (DV): Check that party with that key can control domain.

Org. Validation (OV) and **Extended Validation (EV)**: Also check company name, location etc via public records.

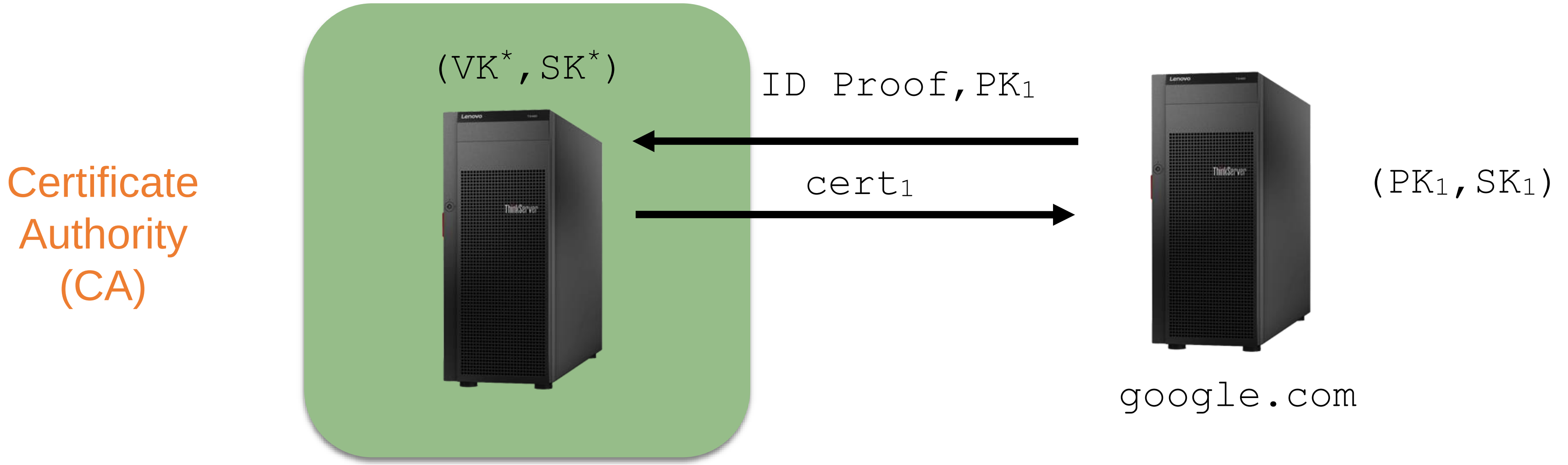
ACME Protocol by Let’s Encrypt



1. Requestor submits public key and request to CA
2. CA gives a *challenge* to requestor
3. Requestor places *challenge* on server or DNS records
4. CA checks *challenge* and then issues cert if challenge matches

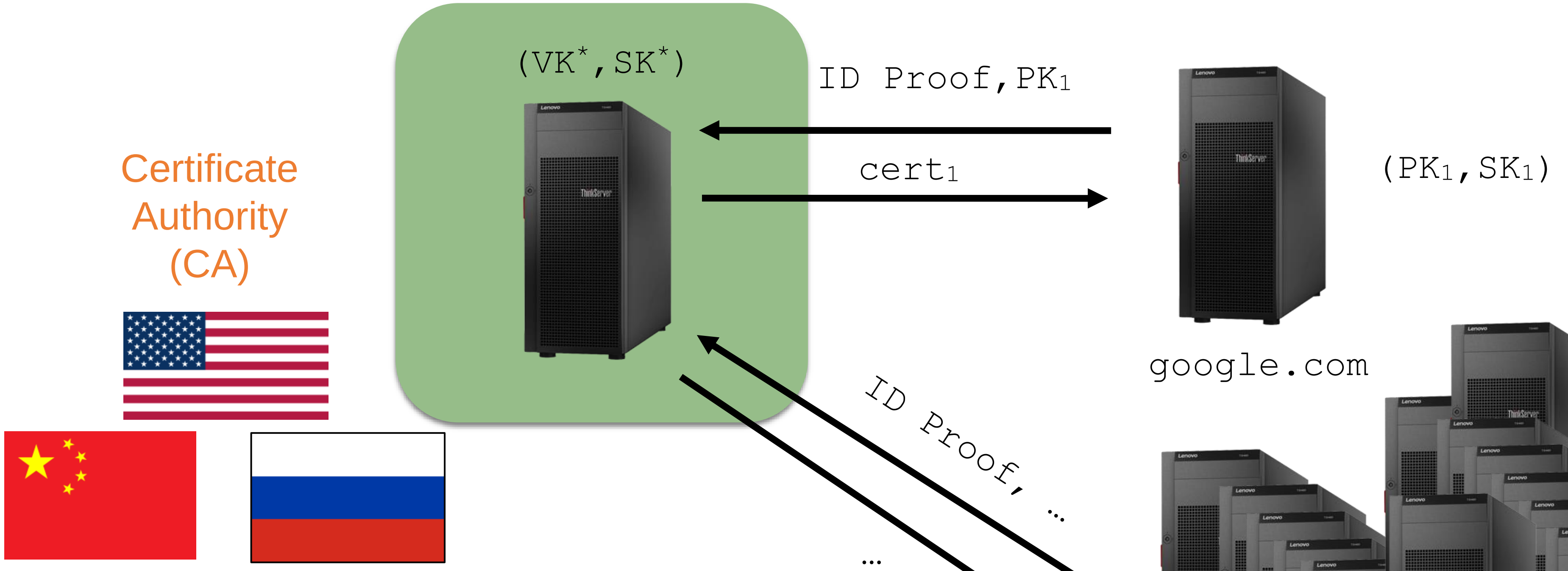
Let’s Encrypt certbot

Scaling Certificates to the Internet



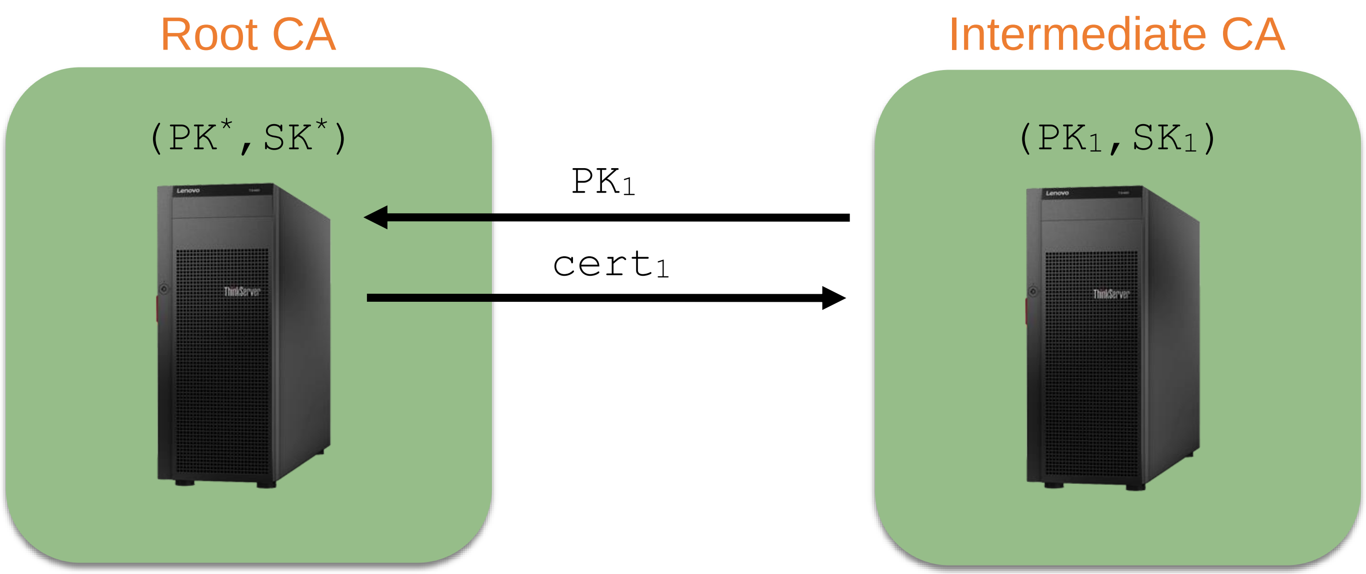
Having one CA works fine if the Internet has just a few entities and everyone agrees that the CA is trustworthy.

Scaling Certificates to the Internet



But the Internet has billions of devices…
And not everyone agrees on a trusted party (CA)…

Scaling: Intermediate CAs and Cert Chains



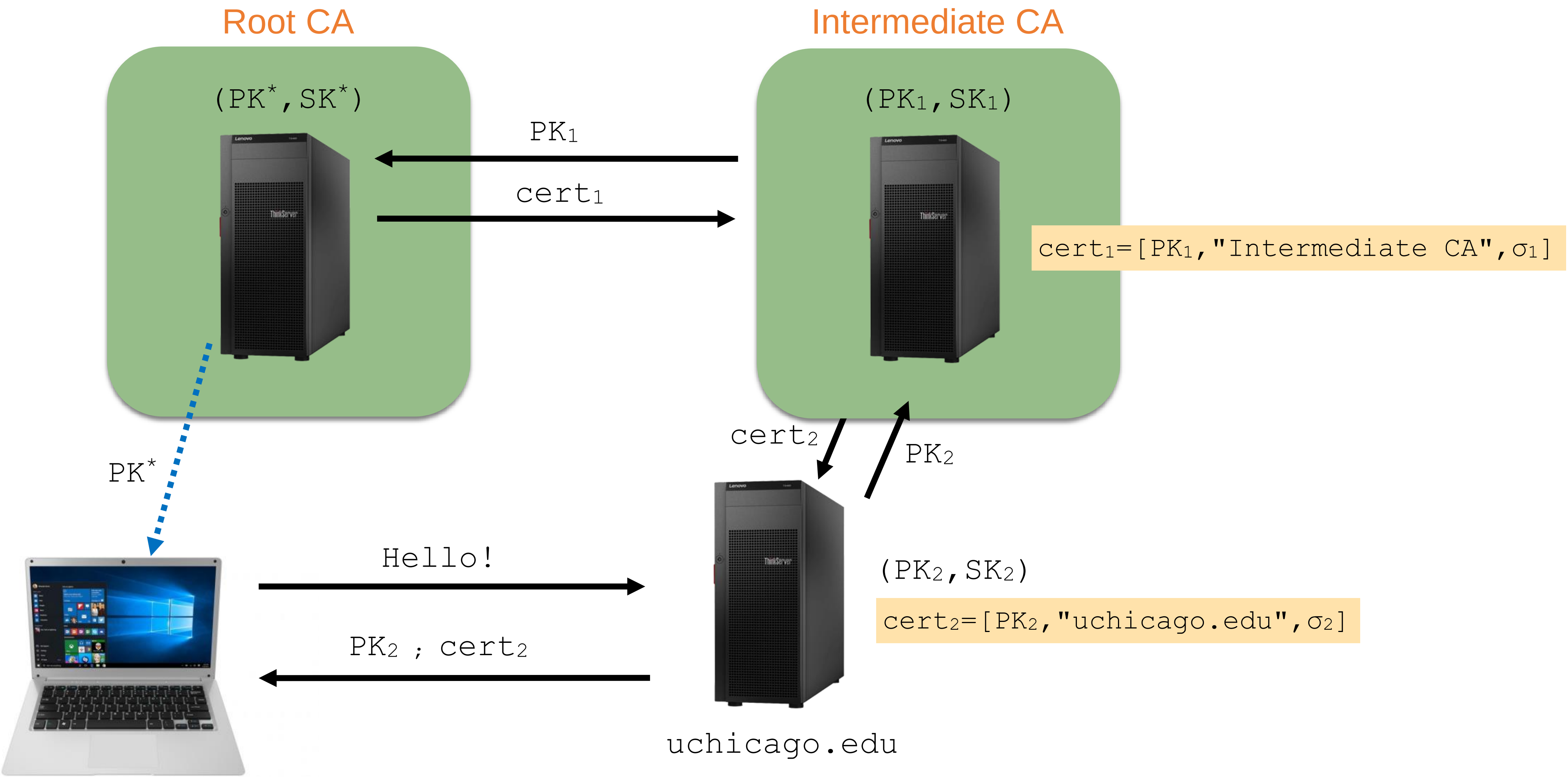
To handle scaling:

- Allow a trusted Root CA to delegate their trust to multiple intermediate CA's
- Any of these intermediate CA's can then create a certificate for someone
  - 100's of intermediate CA's on the Internet

Scaling: Intermediate CAs and Cert Chains

To check PK_2 recursive validation:

1) Check $cert_2$ to make sure PK_2 for uchicago.edu
2) Get PK_1 and $cert_1$ to check sig of $cert_2$
3) If $cert_1$ issued by root CA, use PK* to its check sig.



PK^* bound to Root ⇒ PK_1 bound to CA ⇒ PK_2 bound to uchicago.edu

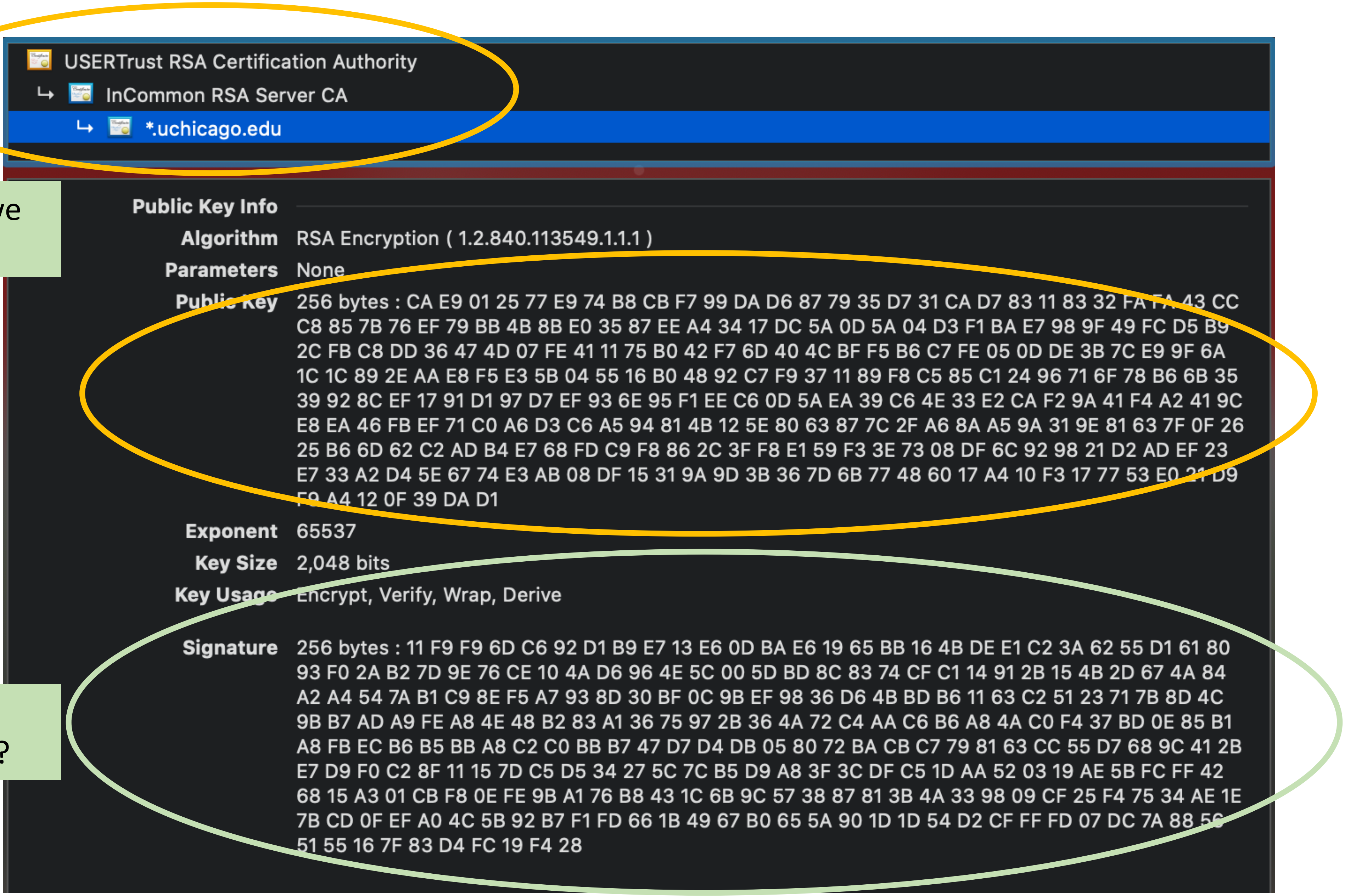
X.509 Certificates

Cert Content Includes:

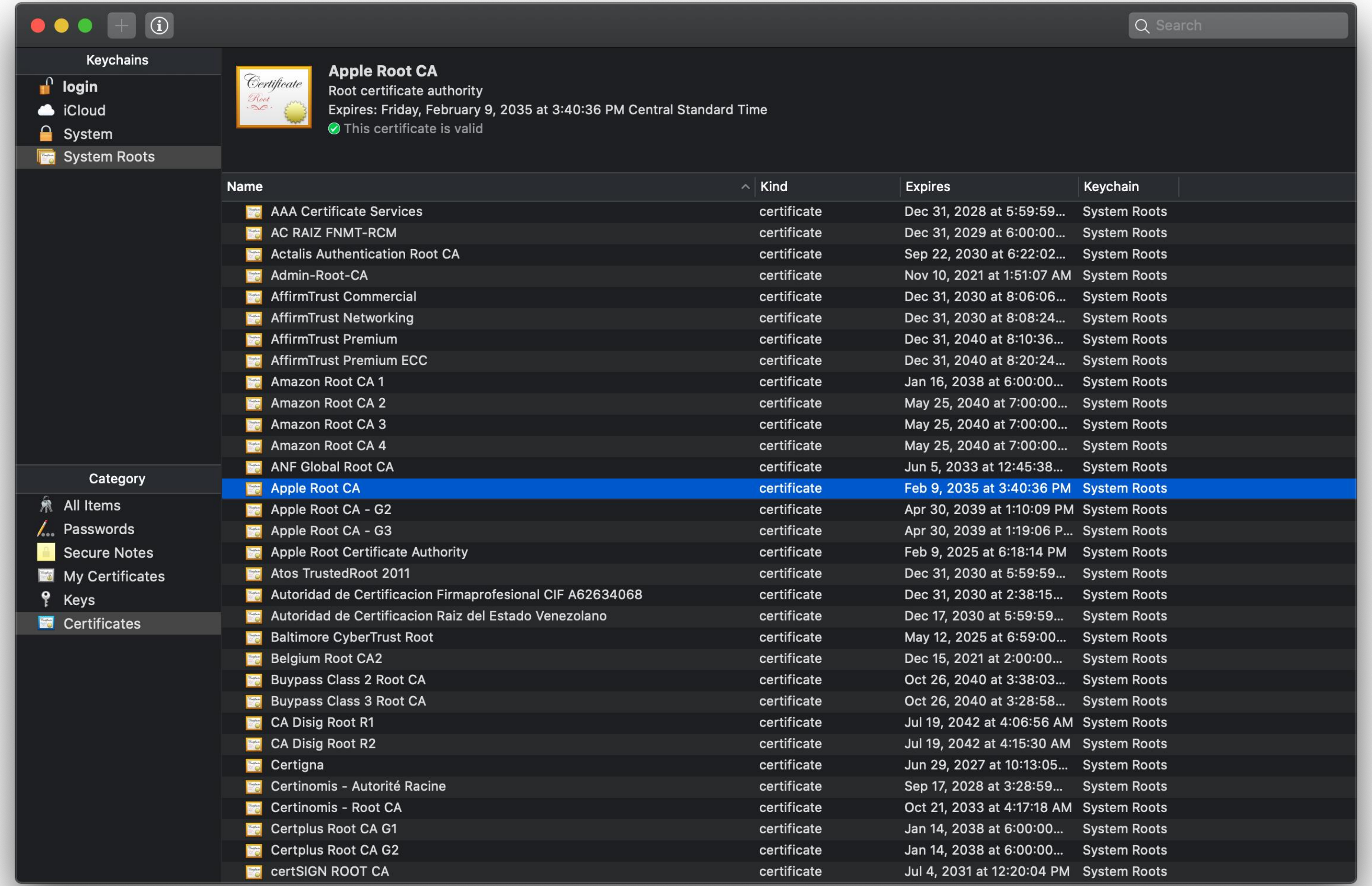
- Cert’s Serial number
- Cert’s Expiration date
- Common name of subject (e.g., Bob [google.com])
- Public key of subject
- Extensions (possibly many)
- CA info (name of CA that is issuing the cert, etc.)
- CA’s Signature (on hash of cert)

Who are we trusting?

Who’s signature?



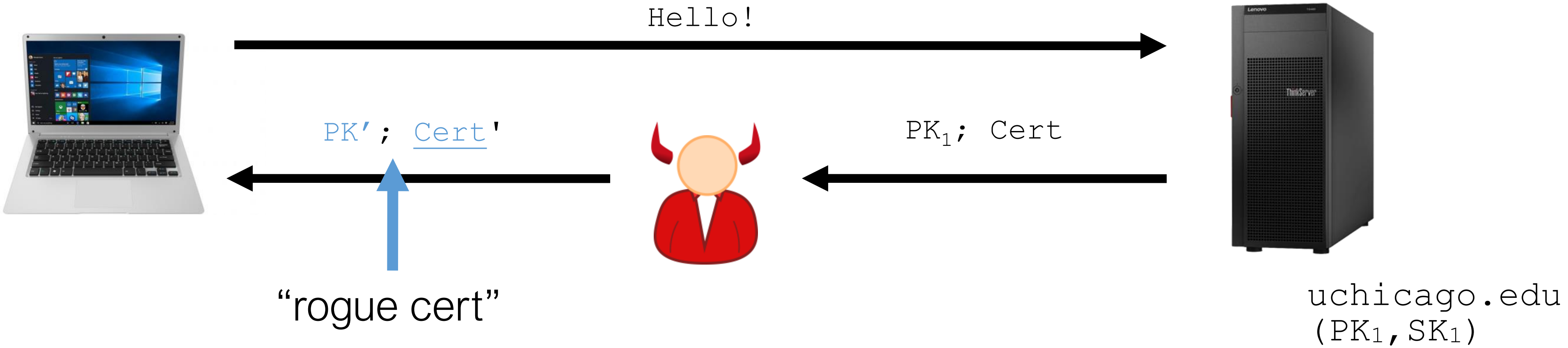
Root CA’s & Root Certificates



Outline: Crypto Part 3

- Recap: Secure communication channels
- Authenticating endpoints: Certificates (Certs)
- Issuing Certs and Certificate Infrastructure (PKI)
- Attacks, Countermeasures
- Real World Secure Channels: SSL / TLS

What if attacker got a “valid” cert for uchicago.edu that has their malicious key?



- “Machine-in-the-middle” can read/change all traffic undetected


TECHNOLOGY | August 31, 2011

Google warns of man-in-the-middle attacks on Iranian users

Intrusion into Dutch SSL provider led to cyber snooping

CA Security

Some common attacks to get rogue certificate:

- Fool or bypass a CA's validation process
- Compromise a CA organization and generate malicious cert's

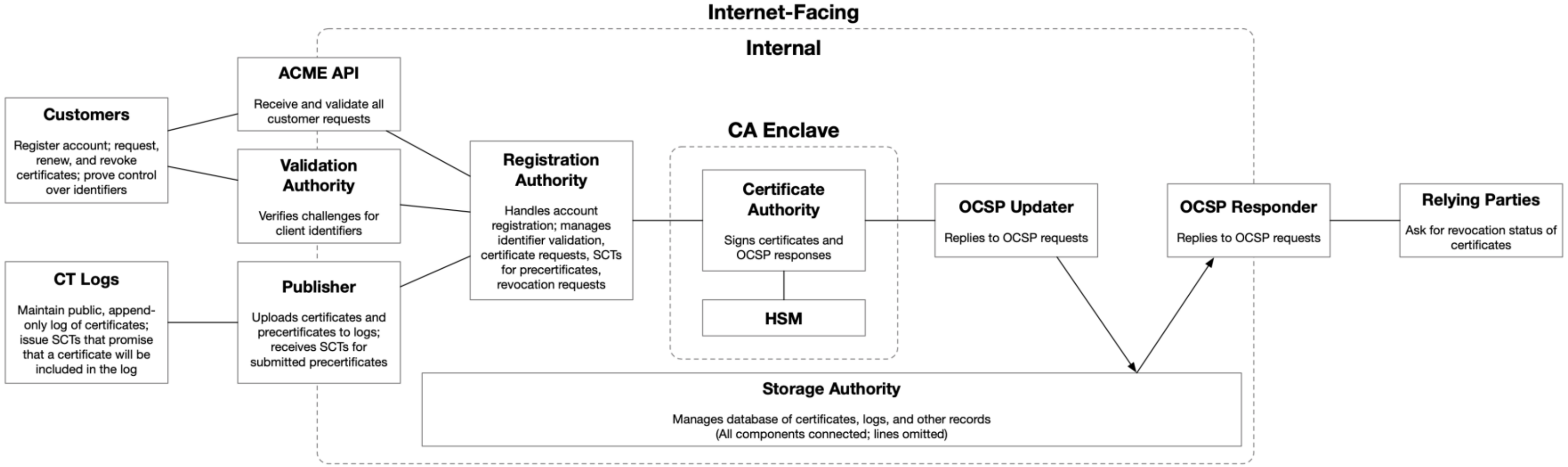


Figure 3: Boulder architecture. Let's Encrypt developed and operates a Go-based open-source CA software platform named Boulder, which is composed of single-purpose components that communicate over gRPC, as illustrated here. The certificate lifecycle unfolds roughly from left to right in the diagram.

"Let's Encrypt: An Automated Certificate Authority to Encrypt the Entire Web", CCS 2019

Sample of CA Security Incidents

- 2011, Root CA Comodo: Login credentials stolen. Hacker issues certs for mail.google.com, login.live.com, www.google.com, login.yahoo.com…
- 2011, Root CA DigiNotar: Hacker issues rogue cert for *.google.com, others. Used to MitM by Iranian government.
- 2013, Root CA TurkTrust: Accidentally issues intermediate CA cert, used to issue gmail.com cert.
- …
- 2019, Root CA Comodo: Pushes email login credentials to public GitHub repo…

(Slide inspiration: Dan Boneh)

Countermeasure: Public-Key Pinning

- **Goal:** Eliminate Root / Intermediate CA's with bad hygiene or who you don't trust

- Server (e.g., website) can tell client (e.g., browser) to only accept certs signed by certain CA's
  - Code trusted CA keys into client app (e.g., Chrome only trusts certs signed by Google's CA), or
  - Send special application message telling client what to pin (More common)
- Helped discover some rogue certs from previous slide

- What are some problems with this defense?
  - If server hacked… attacker can pin a malicious key/cert: will only connect w/ attacker cert!
  - Website error: pin wrong or broken key… website inaccessible!

Now deprecated because of these issues

Countermeasure: Revocation

Publicly list bad (revoked) certificates so they are no longer accepted

- CA or Server (that was issued cert) can revoke

Mass Revocation: Millions of certificates revoked by Apple, Google & GoDaddy

The DarkMatter debate is already having industry-wide ramifications

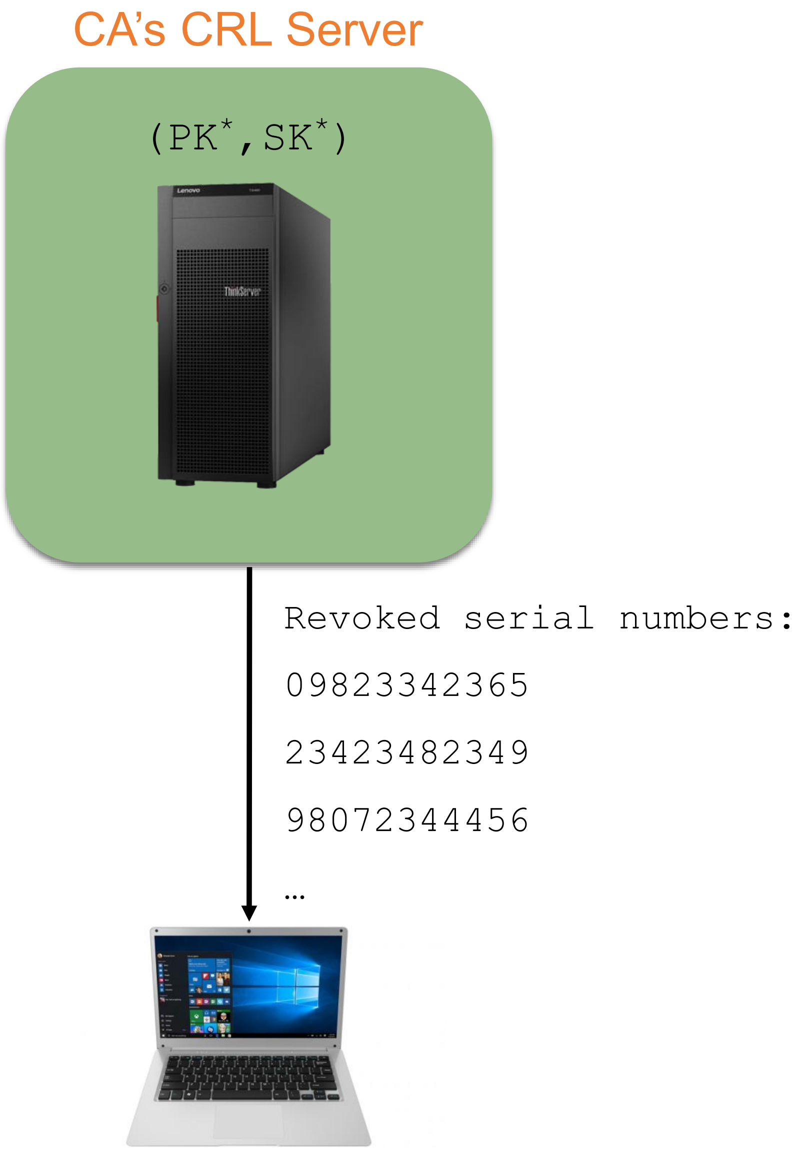
Millions of SSL/TLS certificates – among other digital certificates – are being revoked right now as a result of an operational error that caused the generation of non-compliant serial numbers.

March 3, 2020 0

Let's Encrypt to Revoke 3 Million SSL Certificates on March 4

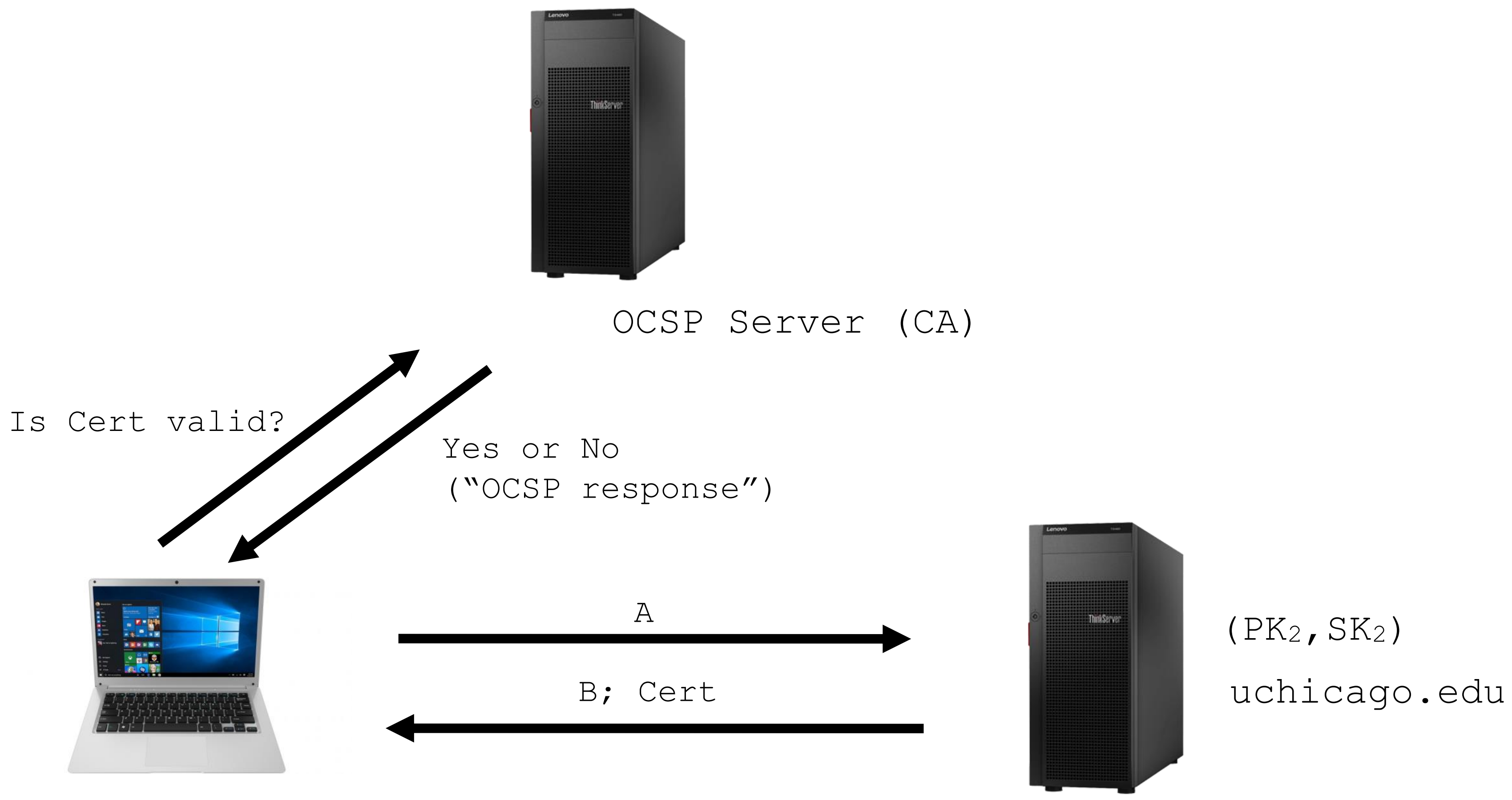
The world's leading free SSL provider announces that millions of certificates are being revoked due to a bug they discovered days ago – giving subscribers potentially only hours to respond

Cert Revocation Lists (CRLs)



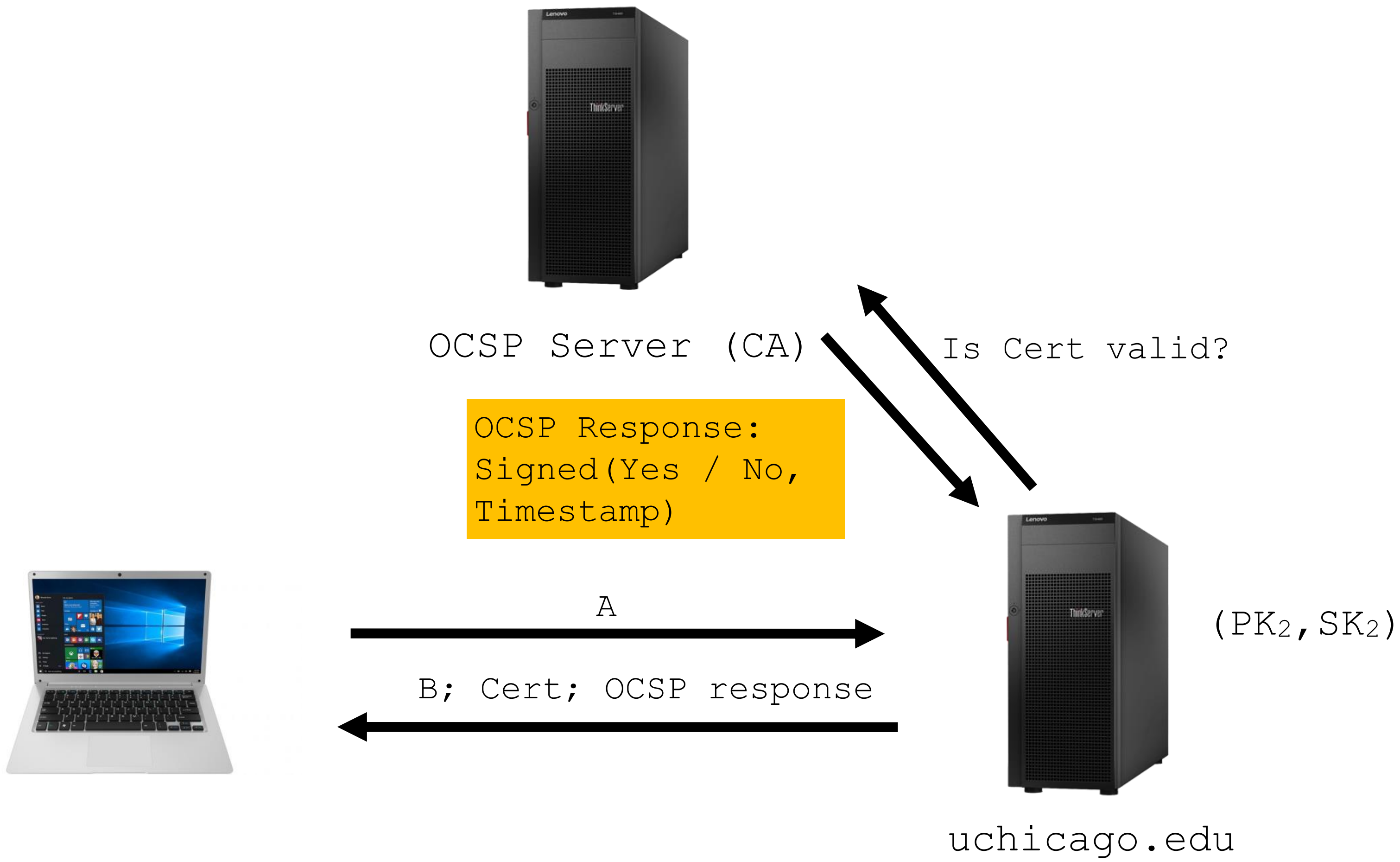
- Each CA provides a list of revoked cert's
- Clients can download CRL and check cert's they receive against the list
- Problems:
    - List will get too large
    - Difficult to keep current

Revocation: Online Certificate Status Protocol (OCSP)



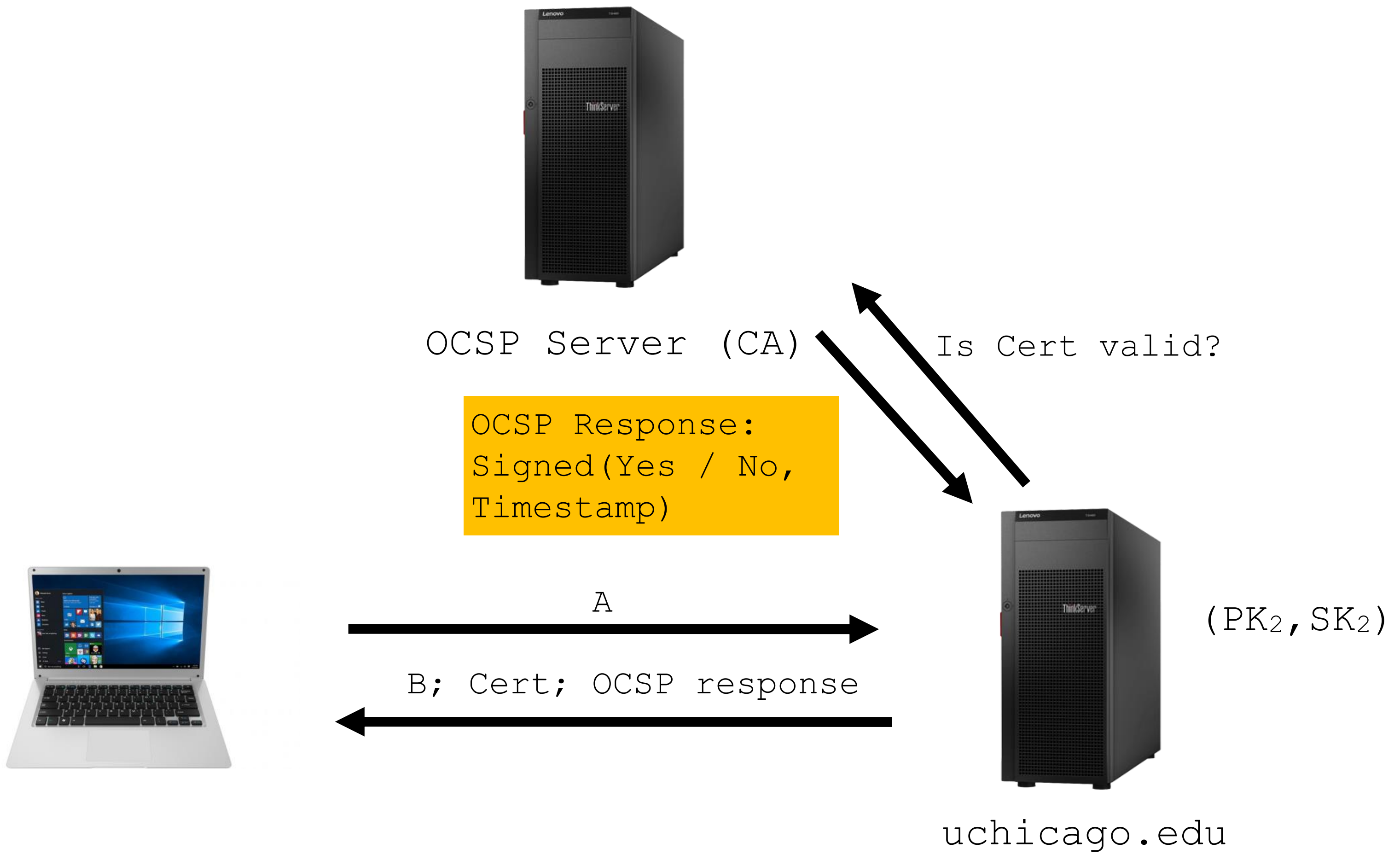
- Add another server to connect to, slowing connection
- What if OCSP server times out?
- Privacy problem?

Revocation: OCSP Stapling



- TLS Extension that allows for OCSP response to be included with cert
- Client checks CA signature and time-stamp on response (~hours old).
- Certs can have “must staple” extension.

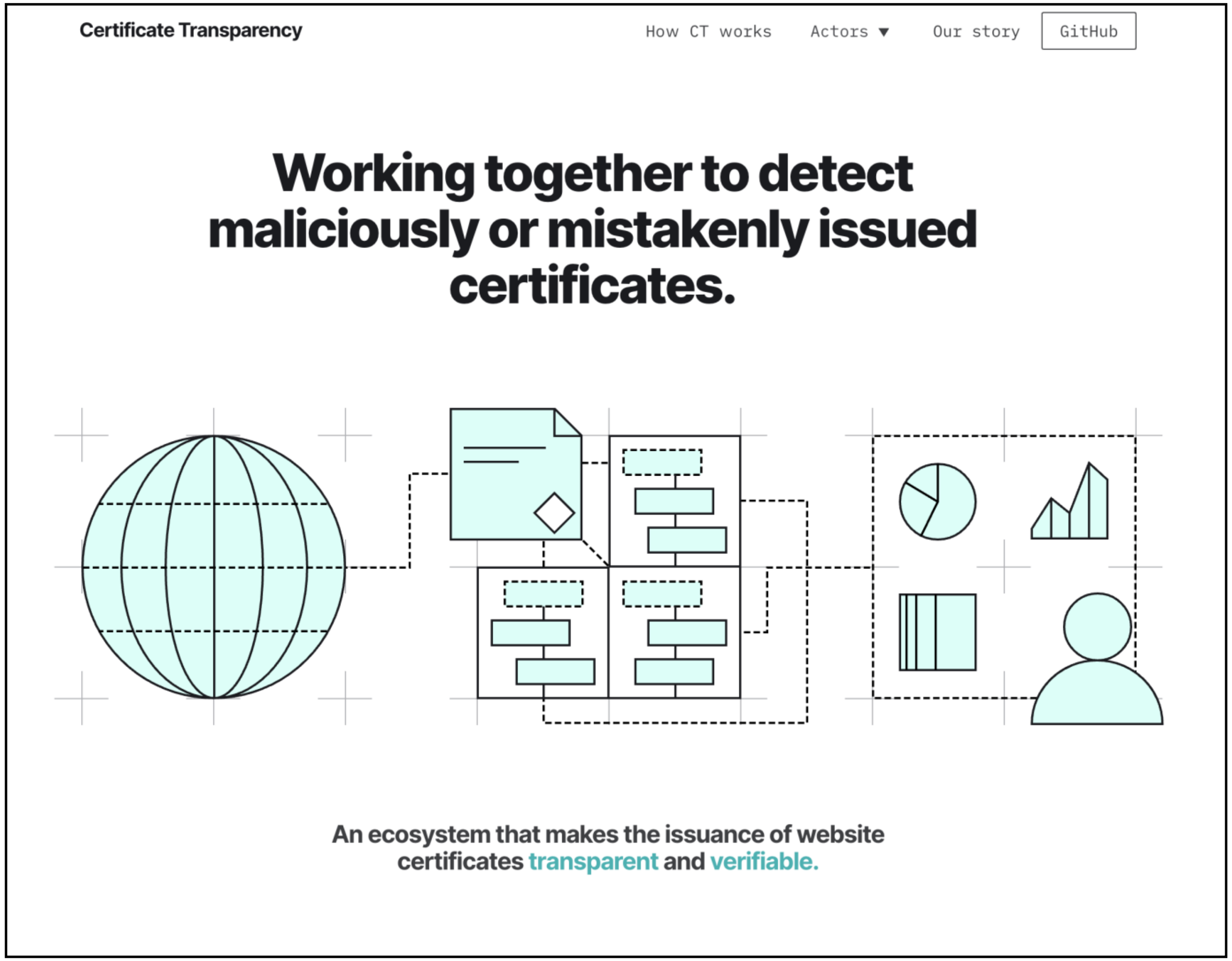
Revocation: OCSP Stapling



Problems?

- OCSP server goes down => uchicago.edu goes down (no OCSP response to attach to cert)

Certificate Transparency (CT) : How do we find rogue certs?



Scenario: Attackers compromise a CA and create rogue certs for `google.com` that have
(1) attacker's public keys and
(2) valid CA signature

How does Google or the CA discover these rogue certs were issued or in use?

Cert Transparency:

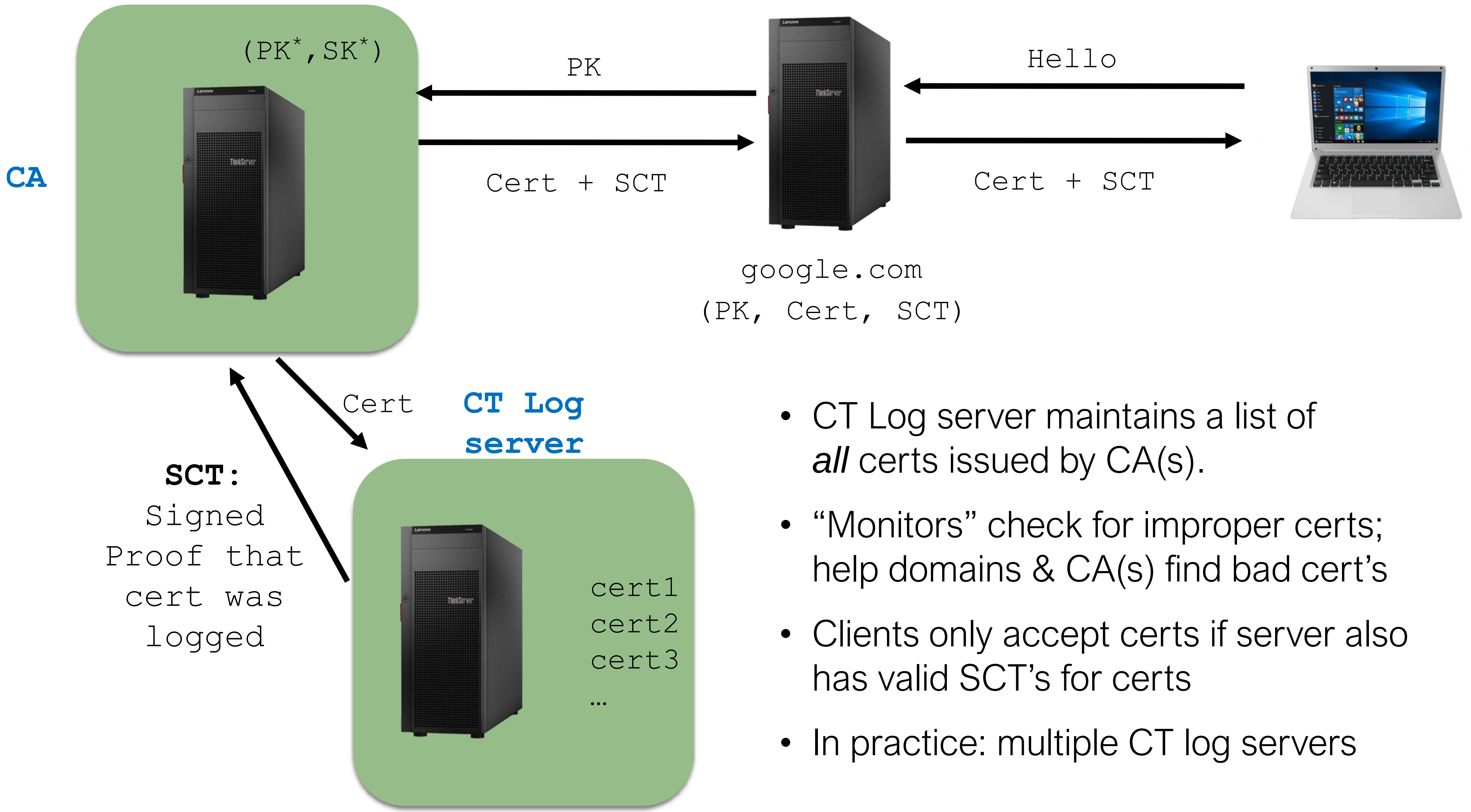
- Require all cert's added to public audit logs
- Domains & CA's can check audit logs for rogue certs & revoke them

Certificate Transparency (CT)

Simplified strategy to find certificates we should revoke:

- An auditor maintains a list (log) of every certificate ever issued
- Whenever a CA issues a cert, they submit (add) cert to this log
- Clients only accept a server's cert if it appears on the log
- Each server (domain) can now monitor the logs to see if anyone (and who) issued a rogue certificate for them
  - If so, add the rogue cert to revocation lists
  - If CA has pattern of issuing rogue cert's, ban them

Certificate Transparency (CT)

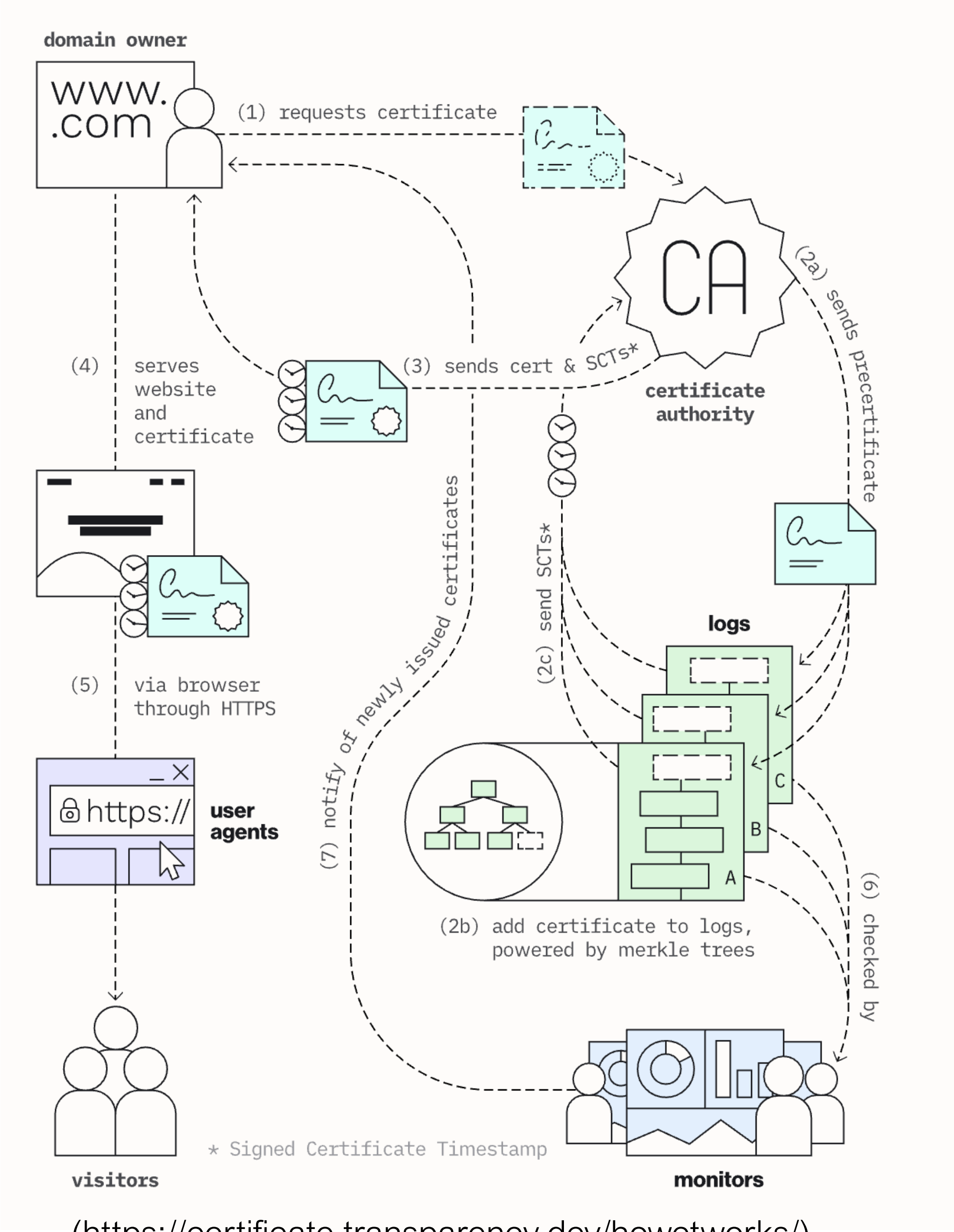


- CT Log server maintains a list of *all* certs issued by CA(s).
- "Monitors" check for improper certs; help domains & CA(s) find bad cert's
- Clients only accept certs if server also has valid SCT's for certs
- In practice: multiple CT log servers

Challenges with CT

- List is huuuuge (every issued cert… solution: temporal sharding)
- Trust the CT Log?
- (Monitors) Who checks the logs?
- Privacy (e.g., enterprise has private servers)?

Cert Transparency & OCSP



(https://certificate.transparency.dev/howctworks/)

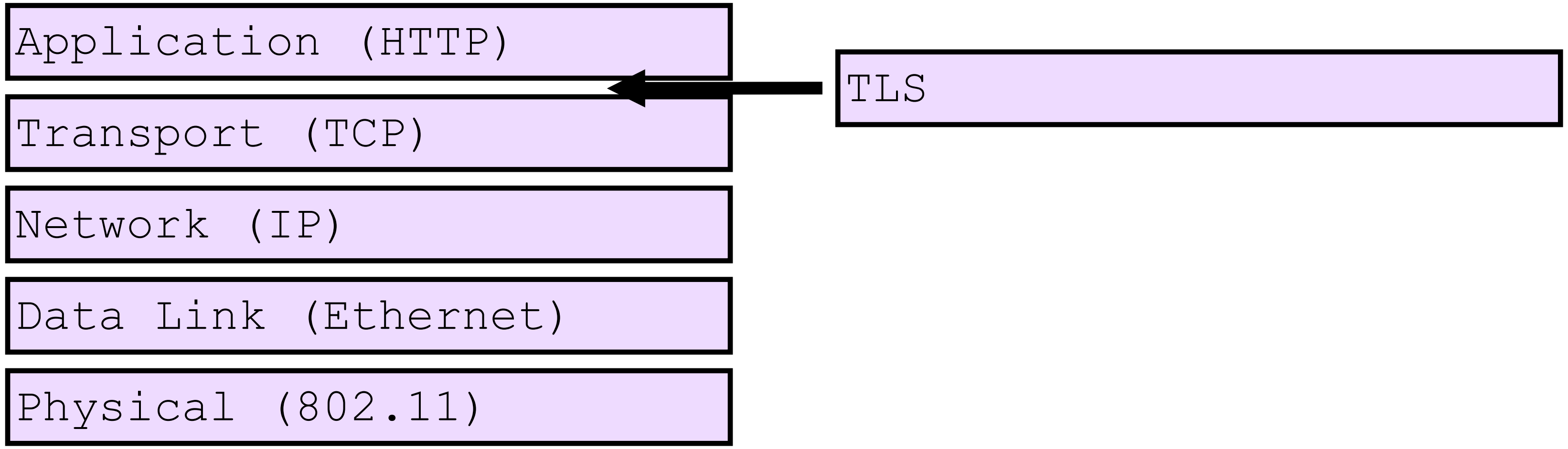
How do CT and OCSP compare?

- OCSP: Allows clients to determine if a cert is valid
- CT: Allows domains (cert owners) and CA’s to find malicious cert’s

Outline: Crypto Part 3

- Recap: Secure communication channels
- Authenticating endpoints: Certificates (Certs)
- Issuing Certs and Certificate Infrastructure (PKI)
- Attacks, Countermeasures
- Real World Secure Channels: SSL / TLS

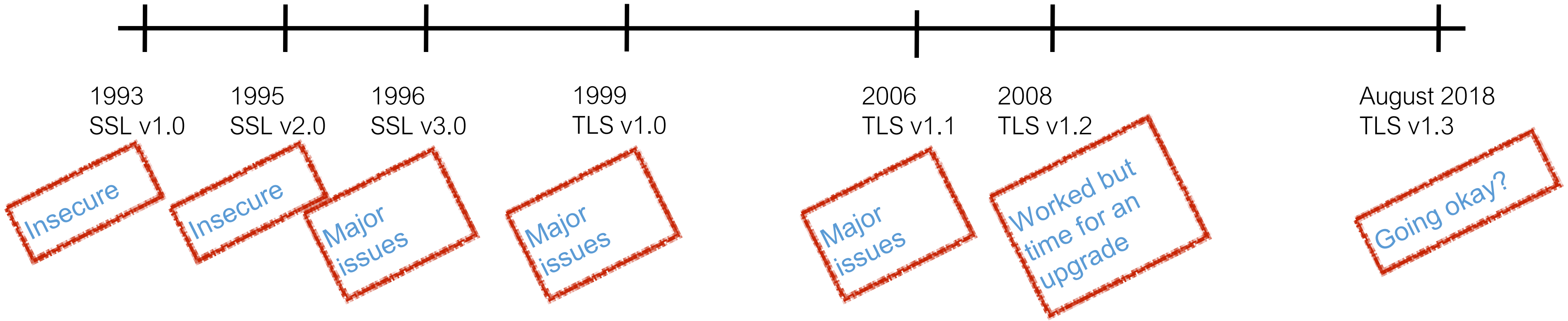
TLS in the Protocol Stack



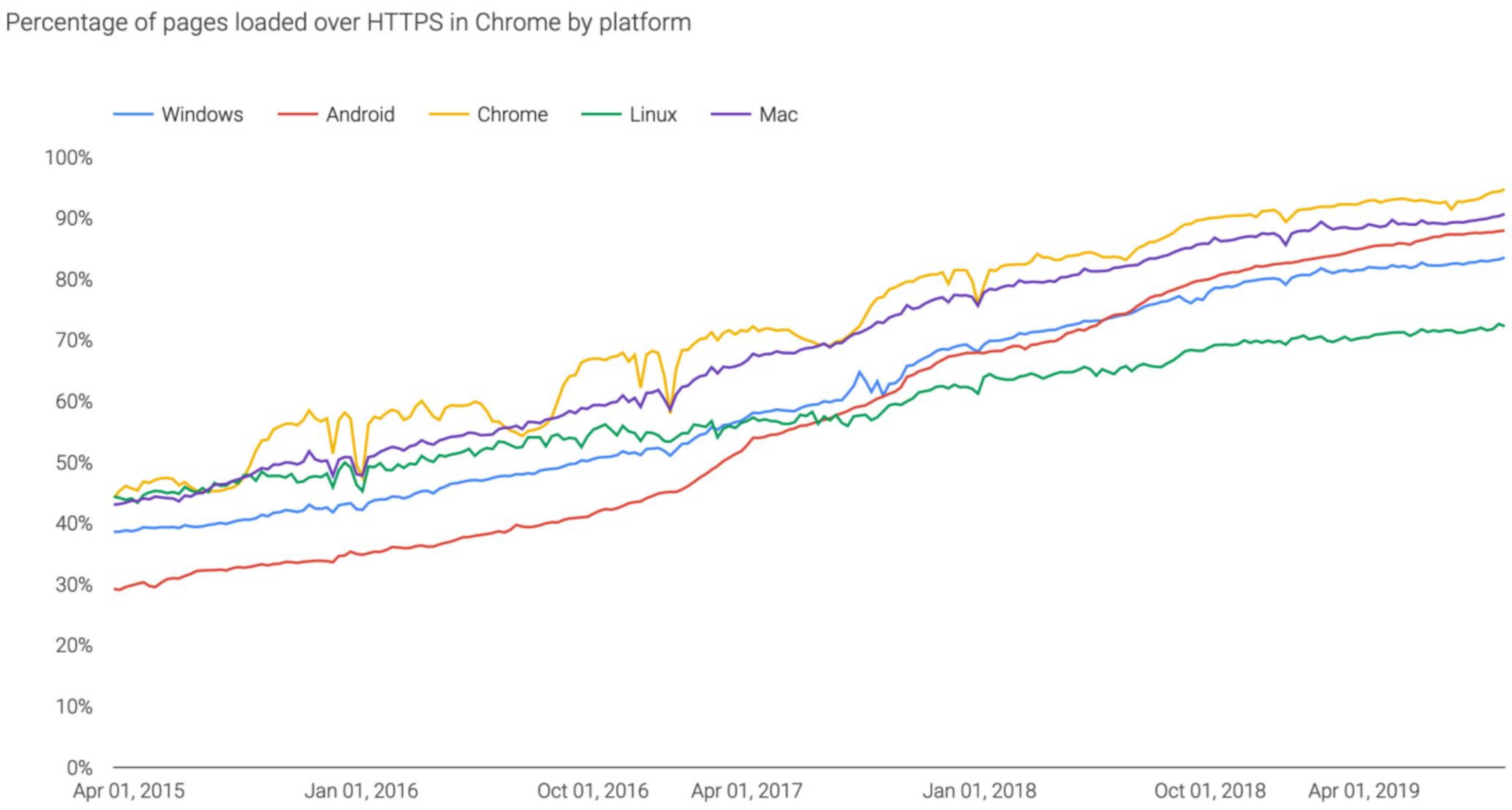
- **Goal:** Allow any application using TCP to transmit data with E2E security
- TLS takes requests from applications (e.g. browser speaking HTTP) and transmits them securely to another host on the Internet

History: SSL/TLS

- SSL = "Secure Sockets Layer"
- TLS = "Transport Layer Security" (renaming of SSL)

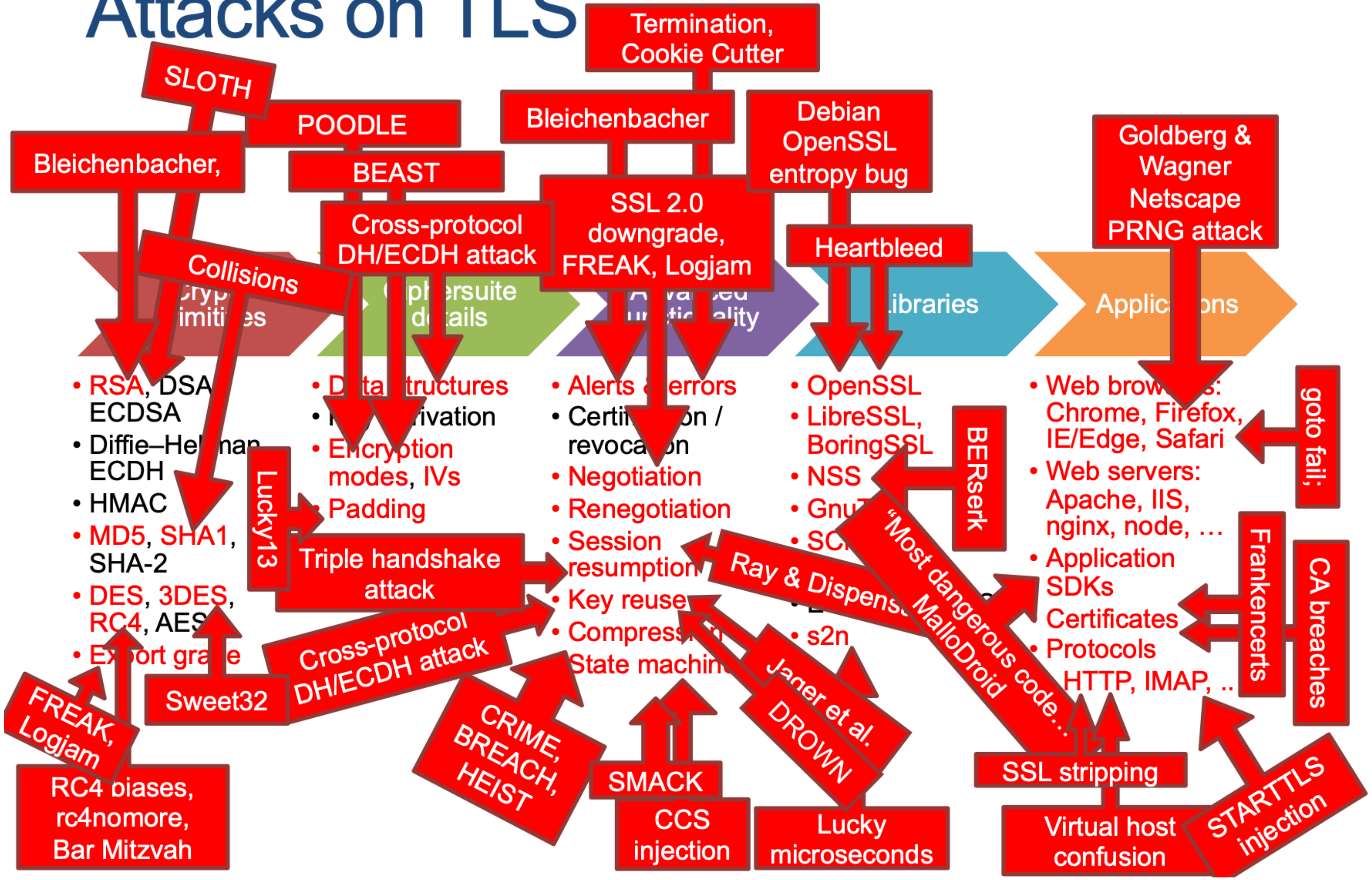


TLS Adoption (HTTPS)



(Source: transparencyreport.google.com, via Matt Green)

Attacks on TLS



Security

It's official: TLS 1.3 approved as standard while spies weep

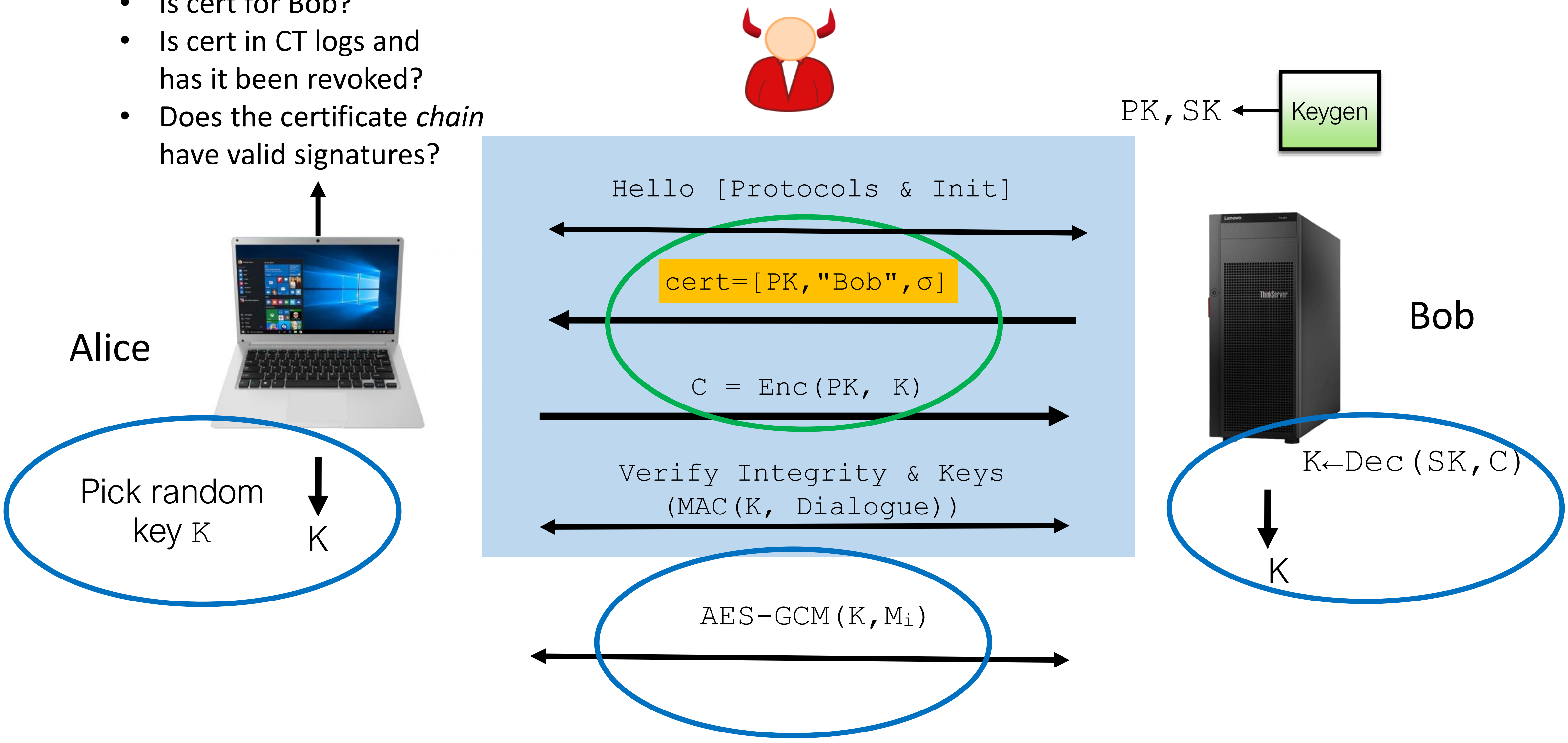
Now all you lot have to actually implement it

By Kieren McCarthy in San Francisco 13 Aug 2018 at 22:19 26 SHARE ▼

An overhaul of a critical internet security protocol has been completed, with TLS 1.3 becoming an official standard late last week.

TLS Protocol: Very Similar to Our Template

- Is cert for Bob?
- Is cert in CT logs and has it been revoked?
- Does the certificate *chain* have valid signatures?



Registrar has set
Final Exam Schedule

Wed, Mar 6 from 8-10 PM

(For **BOTH** sections)

The End