

CSE 484 : Computer Security and Privacy

# ~~Finish Cryptography; Start Web Security~~

*Certificates!*

Winter 2021

David Kohlbrenner

[dkohlbre@cs.washington.edu](mailto:dkohlbre@cs.washington.edu)

Thanks to Franzi Roesner, Dan Boneh, Dieter Gollmann, Dan Halperin, Yoshi Kohno, Ada Lerner, John Manferdelli, John Mitchell, Vitaly Shmatikov, Bennet Yee, and many others for sample slides and materials

...

# Admin

- Homework 2 due in a week (2/10)
- Final Project checkpoint #1 due in 2 weeks (2/17)
- Questions for guest speaker (pollev)

# Final Project

- Start making groups on Canvas
  - “Final Project Groups”
- Use the edstem forum to gather group members
- Goal is to have you find an interesting topic area and learn more!

[https://courses.cs.washington.edu/courses/cse484/21wi/assignments/final\\_project.html](https://courses.cs.washington.edu/courses/cse484/21wi/assignments/final_project.html)



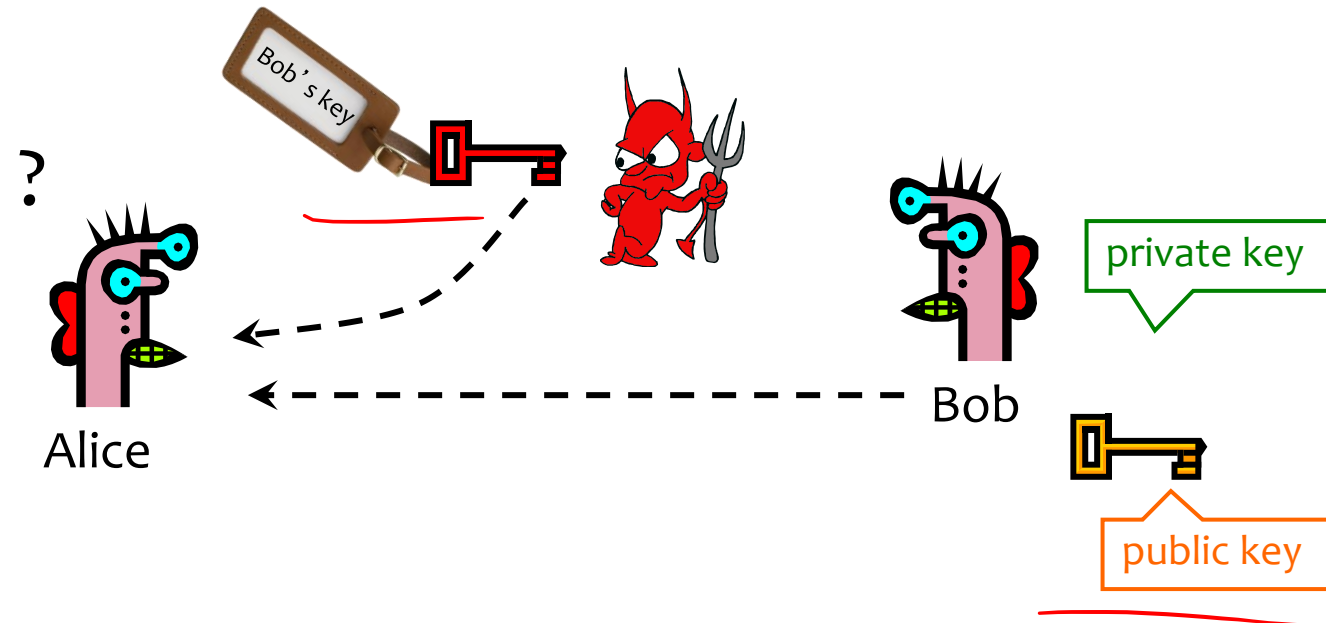
# Want More Crypto?

- Some suggestions:

- CSE 490C (Rachel Lin): <https://courses.cs.washington.edu/courses/cse490c/20au/>
- Stanford Coursera (Dan Boneh): <https://www.coursera.org/learn/crypto>

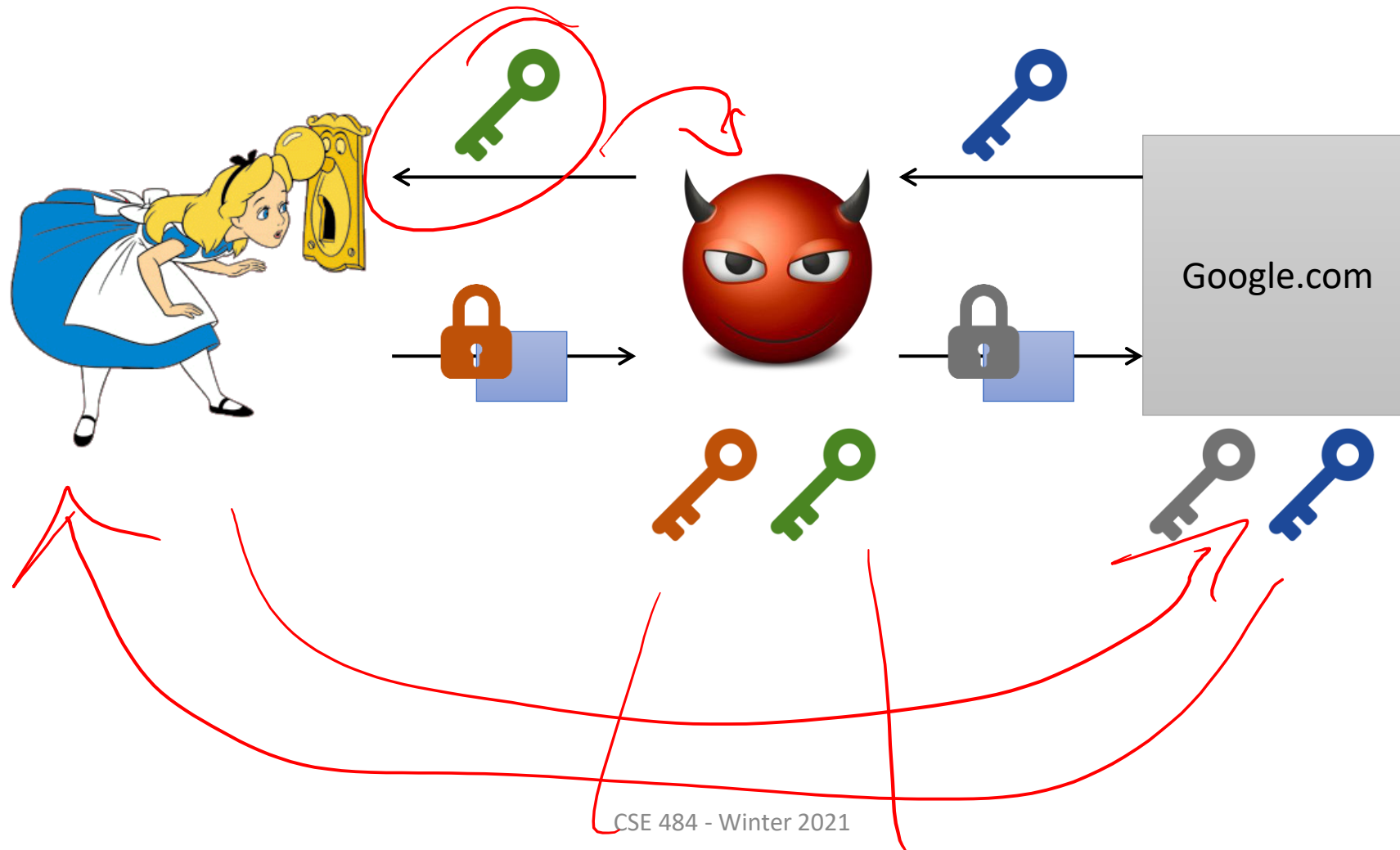
# Authenticity of Public Keys

$(n, e)$  Bob's key



Problem: How does Alice know that the public key she received is really Bob's public key?

# Threat: Person-in-the Middle

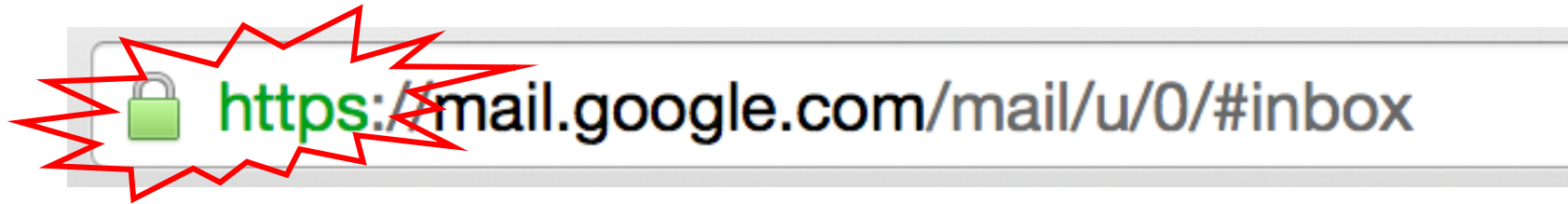


# Distribution of Public Keys

CA  $\text{Sign}[(n, e), \text{"Bob"}]$

- Public announcement or public directory
  - Risks: forgery and tampering
- Public-key certificate
  - Signed statement specifying the key and identity
    - $\text{sig}_{\text{CA}}(\text{"Bob"}, \text{PK}_B)$
- Common approach: certificate authority (CA)
  - Single agency responsible for certifying public keys
  - After generating a private/public key pair, user proves his identity and knowledge of the private key to obtain CA's certificate for the public key (offline)
  - Every computer is pre-configured with CA's public key

You encounter this every day...



SSL/TLS: Encryption & authentication for connections




# SSL/TLS High Level

TLS 1.3

- SSL/TLS consists of two protocols
  - Familiar pattern for key exchange protocols
- Handshake protocol
  - Use public-key cryptography to establish a shared secret key between the client and the server
- Record protocol
  - Use the secret symmetric key established in the handshake protocol to protect communication between the client and the server

# Example of a Certificate

GeoTrust Global CA  
↳ Google Internet Authority G2  
↳ \*.google.com

 **\*.google.com**  
Issued by: Google Internet Authority G2  
Expires: Monday, July 6, 2015 at 5:00:00 PM Pacific Daylight Time  
✓ This certificate is valid


▼ Details

Subject Name	
Country	US
State/Province	California
Locality	Mountain View
Organization	Google Inc
Common Name	*.google.com
Issuer Name	
Country	US
Organization	Google Inc
Common Name	Google Internet Authority G2
Serial Number	6082711391012222858
Version	3

Signature Algorithm	<del>SHA-1</del> with RSA Encryption ( 1.2.840.113549.1.1.5 )
Parameters	none
Not Valid Before	Wednesday, April 8, 2015 at 6:40:10 AM Pacific Daylight Time
Not Valid After	Monday, July 6, 2015 at 5:00:00 PM Pacific Daylight Time
Public Key Info	
Algorithm	Elliptic Curve Public Key ( 1.2.840.10045.2.1 )
Parameters	Elliptic Curve secp256r1 ( 1.2.840.10045.3.1.7 )
Public Key	65 bytes : 04 CB DD C1 CE AC D6 20 ...
Key Size	256 bits
Key Usage	Encrypt, Verify, Derive
Signature	256 bytes : 34 8B 7D 64 5A 64 08 5B ...

Certificate

General Details Certification Path

**Certificate Information**

**This certificate is intended for the following purpose(s):**

- Proves your identity to a remote computer
- Ensures the identity of a remote computer
- 1.3.6.1.4.1.5923.1.4.3.1.1
- 2.23.140.1.2.2

\* Refer to the certification authority's statement for details.

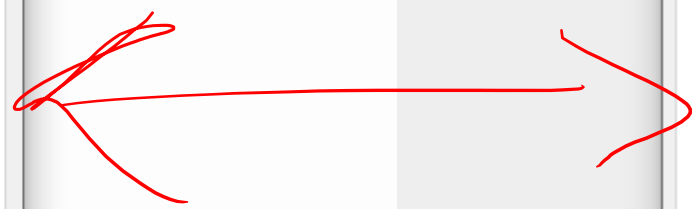
**Issued to:** \*.cs.washington.edu

**Issued by:** InCommon RSA Server CA

**Valid from** 3/19/2020 **to** 3/20/2022


Issuer Statement

OK



Certificate

General Details Certification Path

**Certificate Information**

**This certificate is intended for the following purpose(s):**

- Proves your identity to a remote computer
- Ensures the identity of a remote computer
- 1.3.6.1.4.1.5923.1.4.3.1.1
- 2.23.140.1.2.2

\* Refer to the certification authority's statement for details.

**Issued to:** \*.cs.washington.edu

**Issued by:** InCommon RSA Server CA

**Valid from** 3/19/2020 **to** 3/20/2022

Issuer Statement

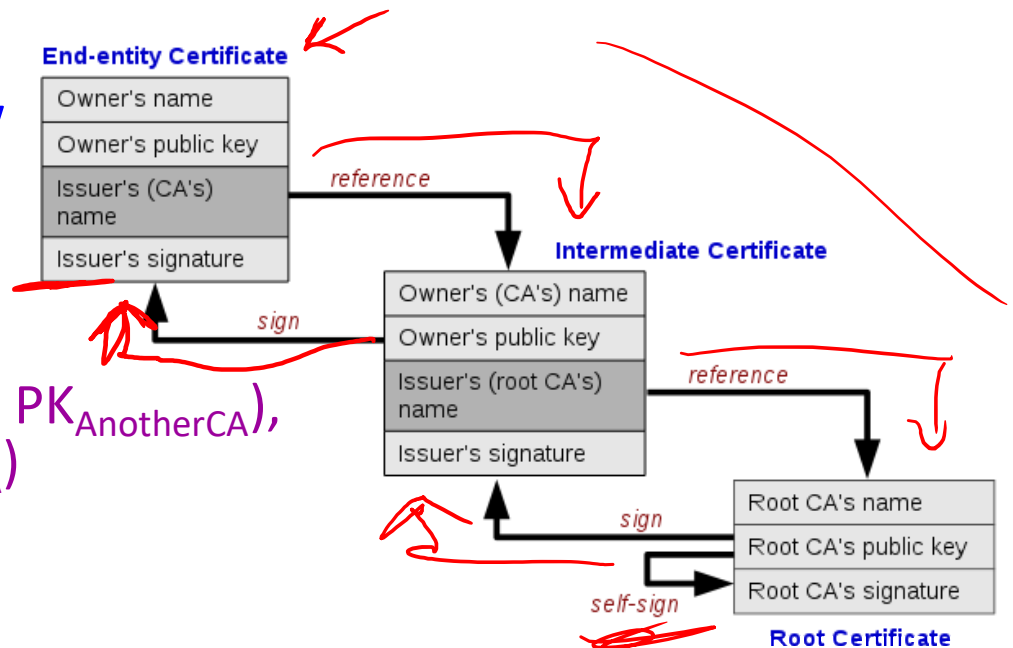
OK

# Hierarchical Approach

- Single CA certifying every public key is impractical
- Instead, use a trusted **root authority** (e.g., Verisign)

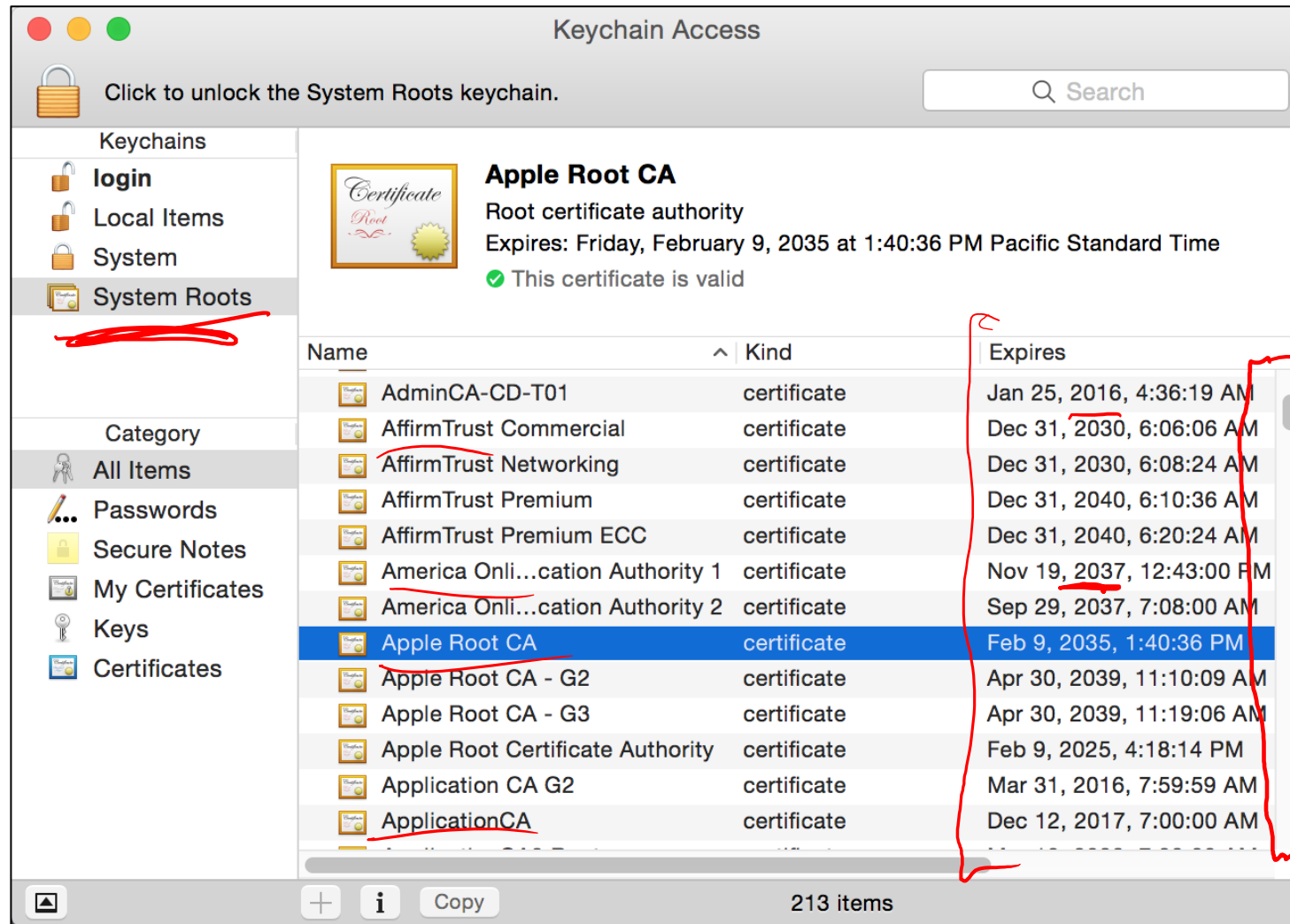
- Everybody must know the root's public key
- Instead of single cert, use a **certificate chain**

- $\text{sig}_{\text{Verisign}}(\text{"AnotherCA"}, \text{PK}_{\text{AnotherCA}}),$   
 $\text{sig}_{\text{AnotherCA}}(\text{"Alice"}, \text{PK}_A)$



- What happens if root authority is ever compromised?

# Trusted(?) Certificate Authorities



# Turtles All The Way Down...



The saying holds that the world is supported by a chain of increasingly large turtles. Beneath each turtle is yet another: it is "turtles all the way down".

[Image from Wikipedia]

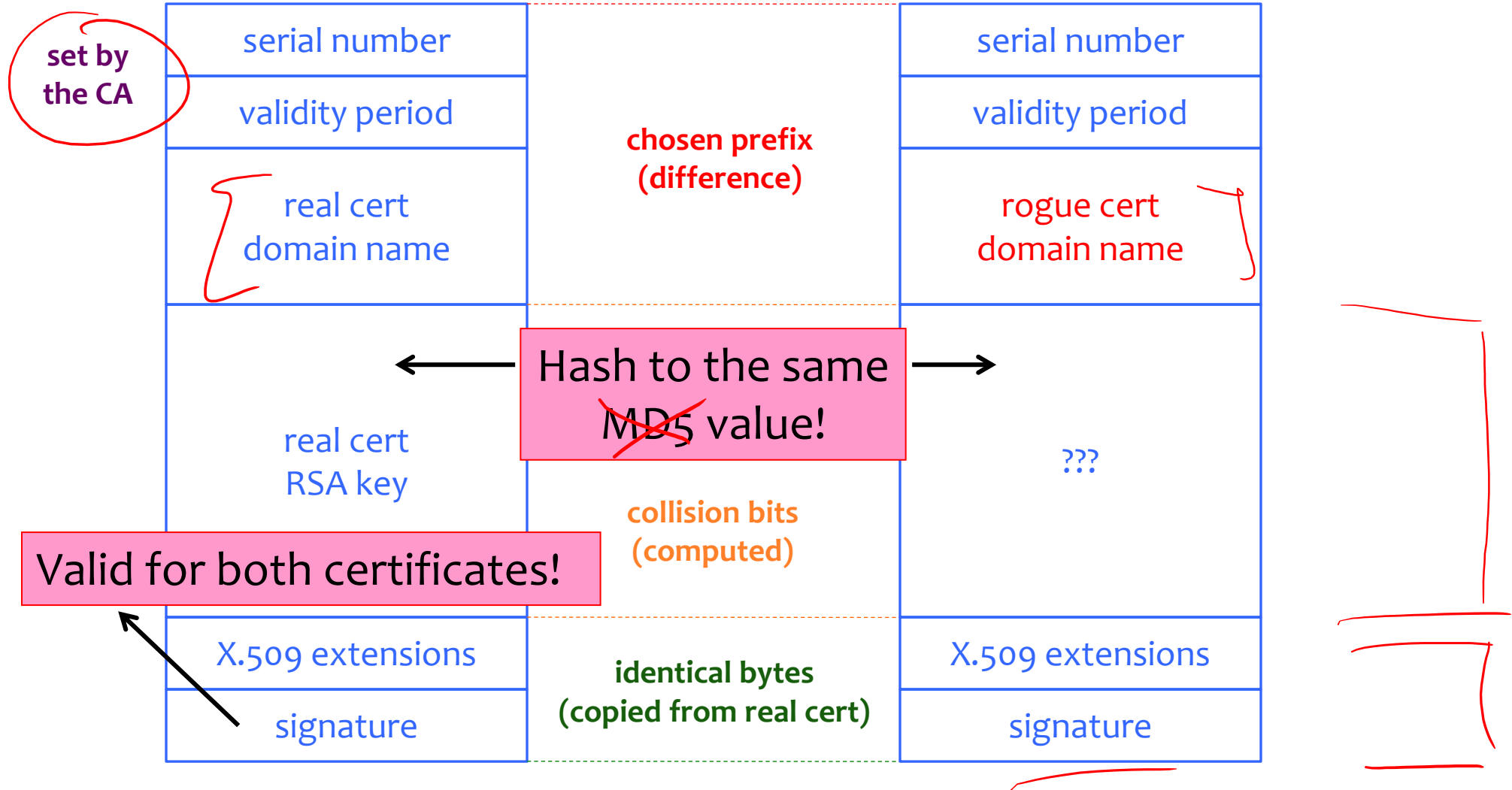
# Many Challenges...

- Hash collisions
- Weak security at CAs
  - Allows attackers to issue rogue certificates
- Users don't notice when attacks happen
  - We'll talk more about this later in the course
- How do you revoke certificates?

#1

C.S. - Washington  
\*, Washington.

# Colliding Certificates





DigiNotar is a Dutch Certificate Authority. They sell SSL certificates.



Somehow, somebody managed to get a rogue SSL certificate from them on **July 10th, 2011**. This certificate was issued for domain name **.google.com**.

What can you do with such a certificate? Well, you can impersonate Google — assuming you can first reroute Internet traffic for google.com to you. This is something that can be done by a government or by a rogue ISP. Such a reroute would only affect users within that country or under that ISP.

## Attacking CAs

### Security of DigiNotar servers:

- All core certificate servers controlled by a single admin password (Prod@dm1n)
- Software on public-facing servers out of date, unpatched
- No anti-virus (could have detected attack)

# Consequences

- Attacker needs to first divert users to an attacker-controlled site instead of Google, Yahoo, Skype, but then...
  - For example, use DNS to poison the mapping of mail.yahoo.com to an IP address
- ... “authenticate” as the real site
- ... decrypt all data sent by users
  - Email, phone conversations, Web browsing

# More Rogue Certs

website ← intermediate  
↑  
root



- In Jan 2013, a rogue \*.google.com certificate was issued by an intermediate CA that gained its authority from the Turkish root CA TurkTrust
  - TurkTrust <sup>''</sup>accidentally issued intermediate CA certs to customers who requested regular certificates
  - Ankara transit authority used its certificate to issue a fake \*.google.com certificate in order to filter SSL traffic from its network
- This rogue \*.google.com certificate was trusted by every browser in the world

# Bad CAs

- **DarkMatter** (<https://groups.google.com/g/mozilla.dev.security.policy/c/nnLVNfqgz7g/m/TseYqDzaDAAJ> and [https://bugzilla.mozilla.org/show\\_bug.cgi?id=1427262](https://bugzilla.mozilla.org/show_bug.cgi?id=1427262))
  - Security company wanted to get CA status
  - Questionable practices
- **Symantec!** ([https://wiki.mozilla.org/CA:Symantec\\_Issues](https://wiki.mozilla.org/CA:Symantec_Issues))
  - Major company, regular participant in standards
  - Poor practices, mismanagement 2013-2017
  - CA distrusted in Oct 2018

# Certificate Revocation

- Revocation is very important
- Many valid reasons to revoke a certificate
  - Private key corresponding to the certified public key has been compromised
  - User stopped paying his certification fee to this CA and CA no longer wishes to certify him
  - CA's private key has been compromised!
- Expiration is a form of revocation, too
  - Many deployed systems don't bother with revocation
  - Re-issuance of certificates is a big revenue source for certificate authorities

# Certificate Revocation Mechanisms

side  
↑  
int  
↑  
root

- Certificate revocation list (CRL)
  - CA periodically issues a signed list of revoked certificates
    - Credit card companies used to issue thick books of canceled credit card numbers
  - Can issue a “delta CRL” containing only updates
- Online revocation service
  - When a certificate is presented, recipient goes to a special online service to verify whether it is still valid
    - Like a merchant dialing up the credit card processor

Attempt to Fix CA Problems:

# Certificate Transparency

- **Problem:** browsers will think nothing is wrong with a rogue certificate until revoked
- **Goal:** make it impossible for a CA to issue a bad certificate for a domain *without the owner of that domain knowing*
  - (Then what?)
- **Approach:** auditable certificate logs

[www.certificate-transparency.org](http://www.certificate-transparency.org)

Attempt to Fix CA Problems:

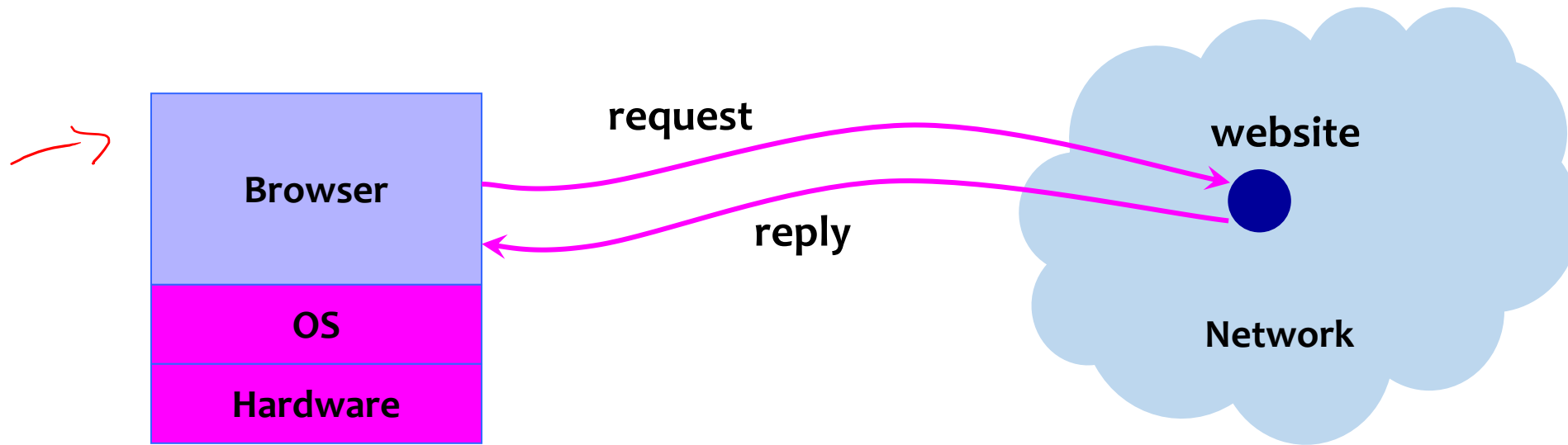
# Certificate Pinning

- **Trust on first access:** tells browser how to act on subsequent connections
- ~~HPKP~~ – HTTP Public Key Pinning *dead*
  - Use these keys!
  - HTTP response header field "Public-Key-Pins"
- HSTS – HTTP Strict Transport Security *✓ dying?*
  - Only access server via HTTPS
  - HTTP response header field "Strict-Transport-Security"



*Next Major Topic!*  
Web+Browser Security

# Big Picture: Browser and Network



# Where Does the Attacker Live?

