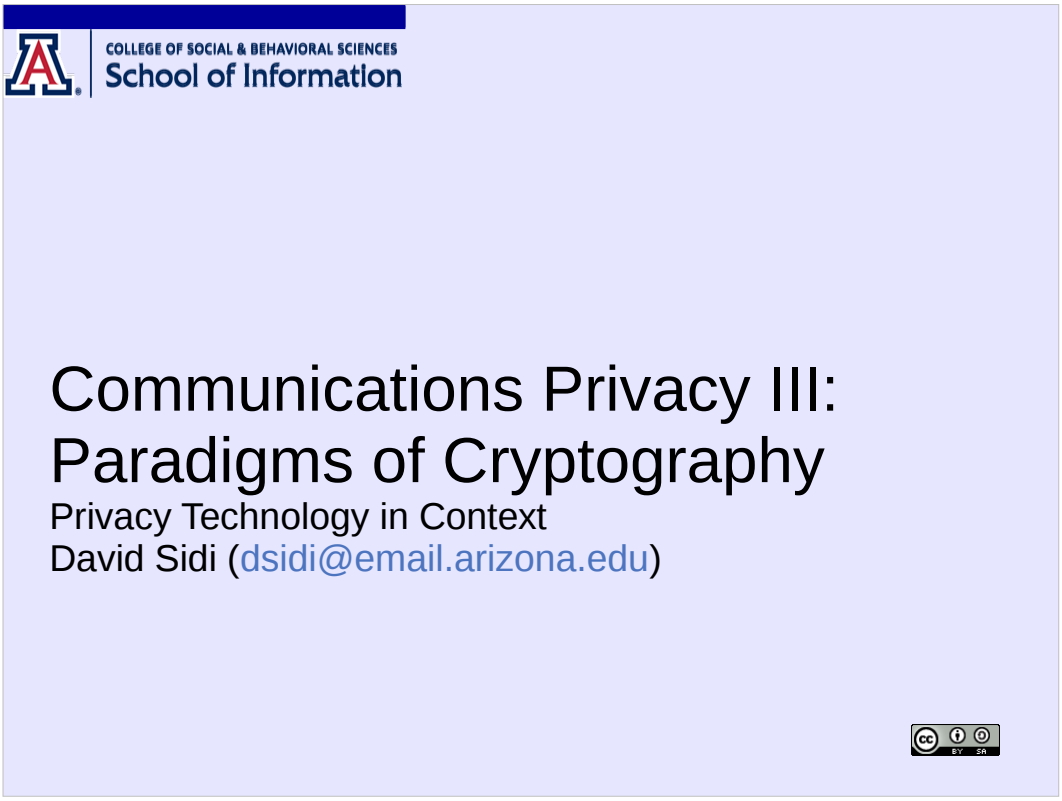
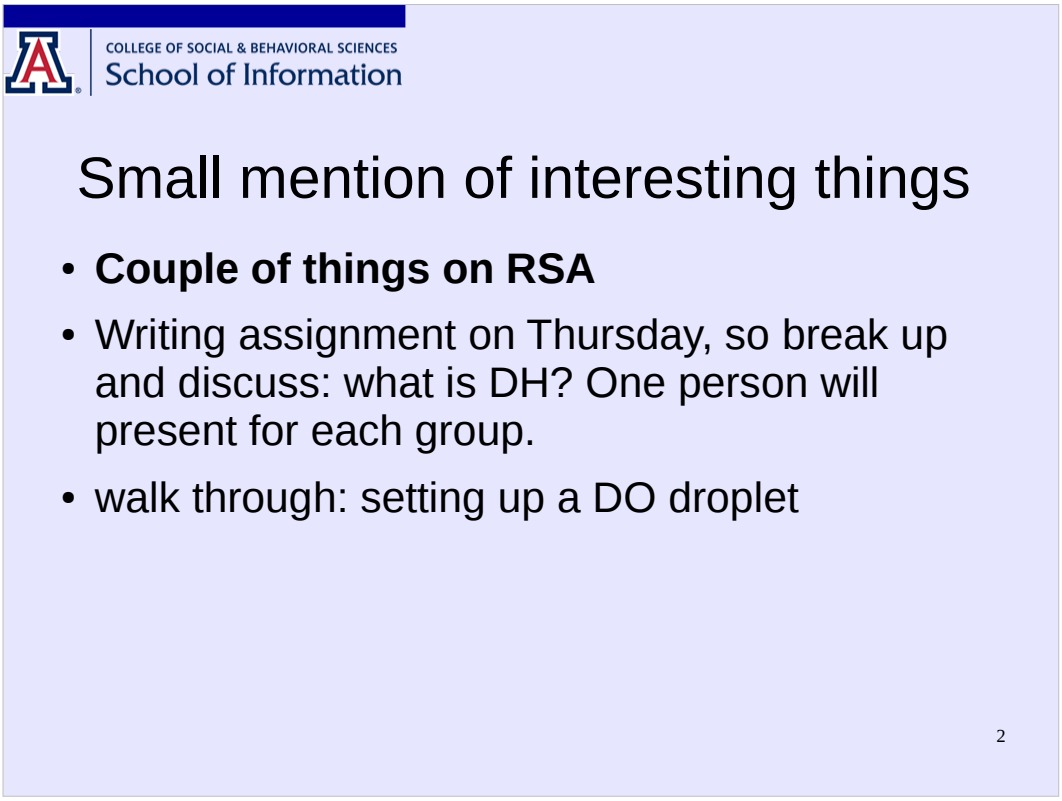


Today we'll go over some of the things that will be useful for the assignment, then talk about a new topic in communications privacy: TLS and the public key infrastructure (PKI) that supports it



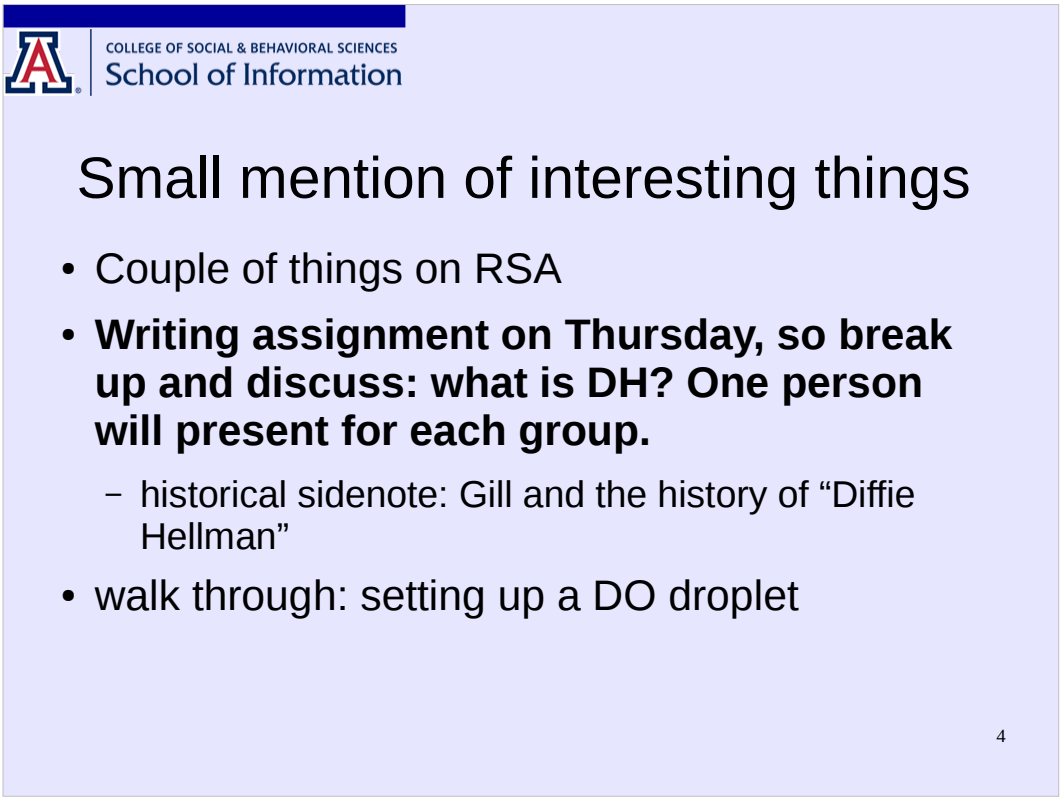
The human rights groups are arguing that British spy programs violate four key rights protected under the convention: the right to privacy; the right to a fair trial; the right to freedom of expression; and the right not to be discriminated against.

Don’t forget you’ve got a reading assignment due on Thursday! As a boost for that, take 5 minutes with a partner and explain how the Diffie-Hellman key exchange protocol works.

COLLEGE OF SOCIAL & BEHAVIORAL SCIENCES
School of Information

Generating p and q in RSA

- Now old news: ROCA. RSA weakness found in keys generated by Infineon TPMs and smart cards)
  - “I've completed a full scan of the crt.sh DB (CT log search engine), which found 171 certs with ROCA fingerprints. The list is at https://misissued.com/batch/28/”
  - mozilla-dev-security-policy@lists.mozilla.org



The human rights groups are arguing that British spy programs violate four key rights protected under the convention: the right to privacy; the right to a fair trial; the right to freedom of expression; and the right not to be discriminated against.

Don’t forget you’ve got a reading assignment due on Thursday! As a boost for that, take 5 minutes with a partner and explain how the Diffie-Hellman key exchange protocol works.

COLLEGE OF SOCIAL & BEHAVIORAL SCIENCES
School of Information

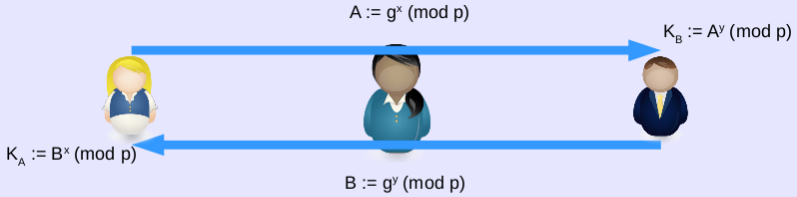
Historical Sidenote: Diffie-Hellman-Gill Key Agreement?

- Commonly known that Merkle had ideas of PKC as well as Diffie and Hellman
- Less well known: “Another potential one-way function, of interest in the analysis of algorithms, is exponentiation mod q, which was suggested to the authors by Prof. John Gill of Stanford University.”
  - From “New Directions”!

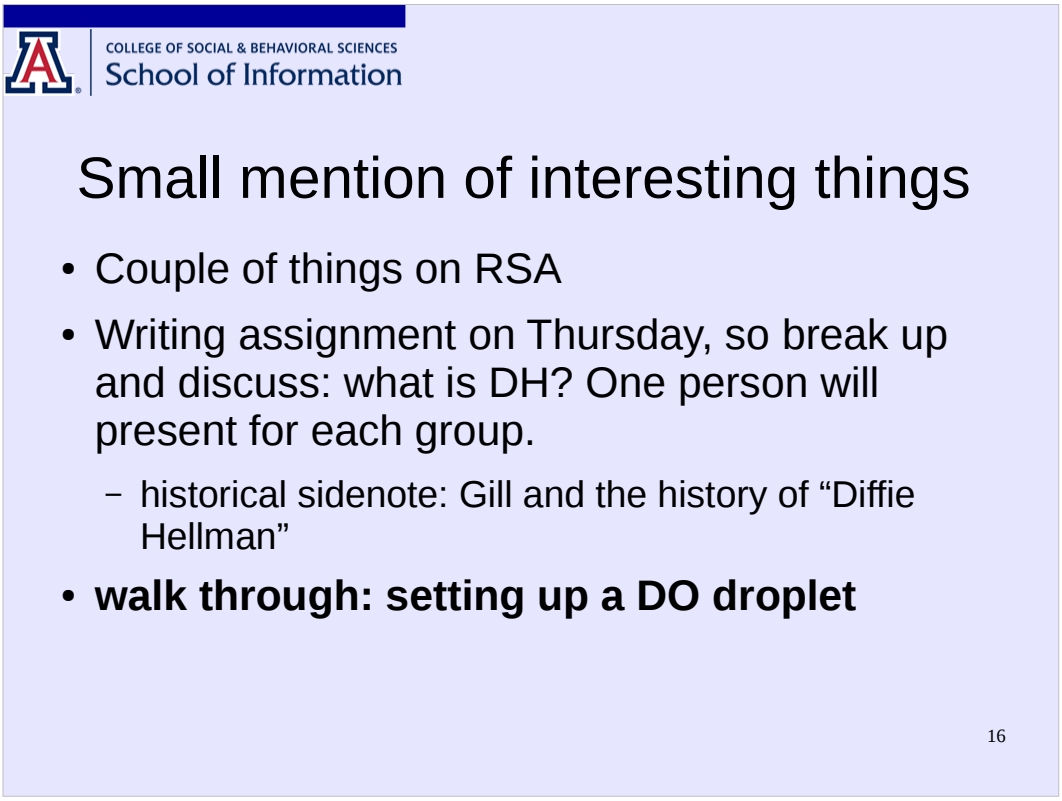
COLLEGE OF SOCIAL & BEHAVIORAL SCIENCES
School of Information

Public Key Cryptography

- Key idea: Encryption key is public, decryption key is private
- “Asymmetric”
- Question: Explain how DH is a “public key distribution system”

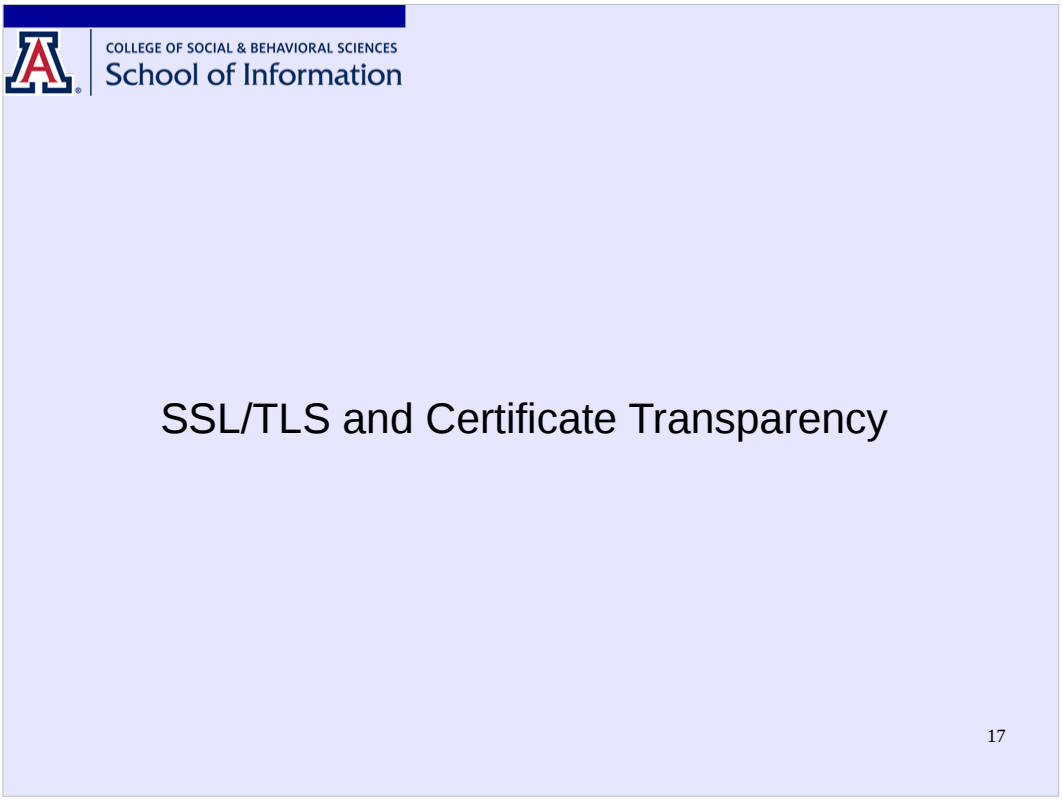


$K_A = (g^y)^x = (g^x)^y = K_B$ is the key



The human rights groups are arguing that British spy programs violate four key rights protected under the convention: the right to privacy; the right to a fair trial; the right to freedom of expression; and the right not to be discriminated against.

Don’t forget you’ve got a reading assignment due on Thursday! As a boost for that, take 5 minutes with a partner and explain how the Diffie-Hellman key exchange protocol works.



so far we've talked about cryptographic primitives, but now we're going to look at a prominent system that uses cryptography for communications privacy: TLS. "Prominent" here means "widely used:" probably in terms of bits shipped, and number of users, TLS is the number one use of crypto. Most people use it without even realizing it!

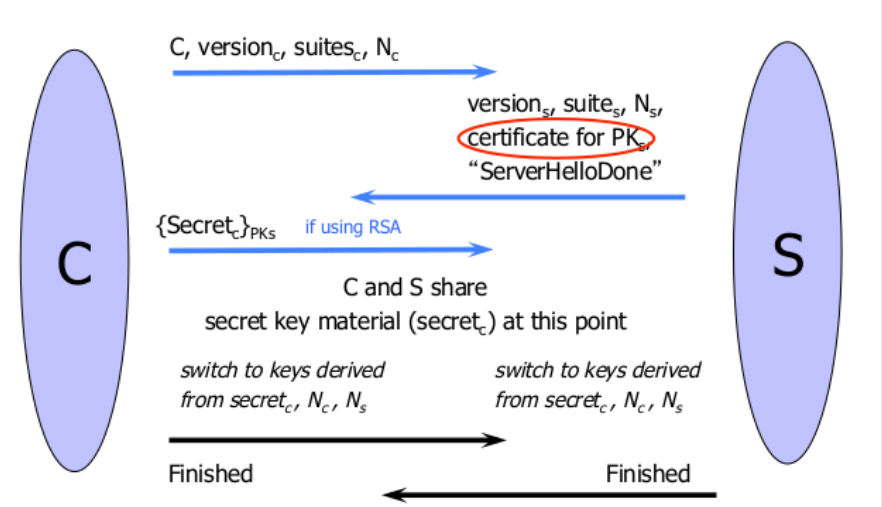
One cost of usability in the TLS case is that it gets you mixed up with the CA mafia, so there has been recent work on making the whole system a bit more trustworthy by at least providing some auditability. That work goes under the heading of 'Certificate Transparency,' and we'll talk about it as well as a lens into the way TLS works.

What is SSL/TLS?

- Secure Sockets Layer and Transport Layer Security protocols
  - Set up a secure channel first with asymmetric key, then transfer a secret for symmetric key encryption of the rest of communication
- SSL/TLS is very widely used for internet security
  - “zero configuration” for most
  - see: HTTP over TLS (i.e., HTTPS)

COLLEGE OF SOCIAL & BEHAVIORAL SCIENCES
School of Information

Handshake protocols



credit: Vitaly Shmatikov

TLS involves two protocols

- Handshake protocol: set up the session key
- Record protocol: symmetric cryptography with the session key to provide a secure authenticated channel
-
- TLS is worried about MiTM and passive attackers.

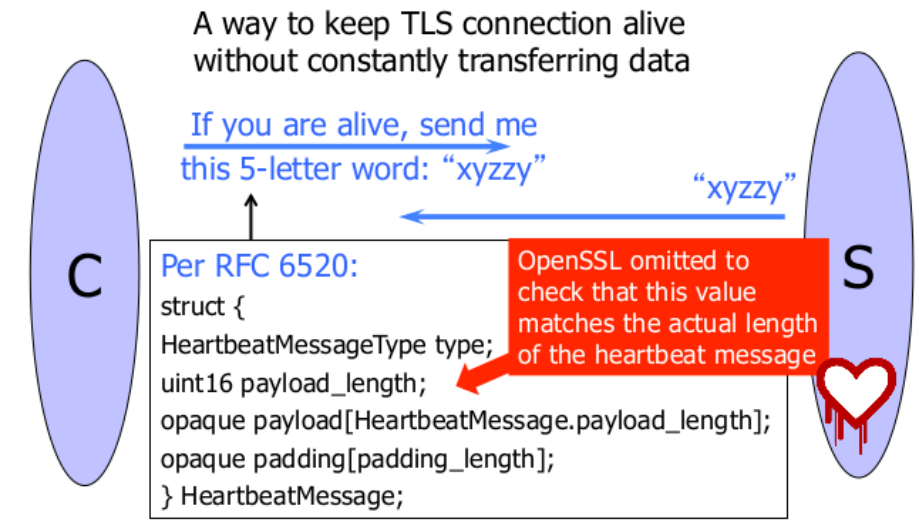
COLLEGE OF SOCIAL & BEHAVIORAL SCIENCES
School of Information

TLS Records

+	Byte +0	Byte +1	Byte +2	Byte +3
Byte 0	Content type			
Bytes 1..4	Version		Length	
	(Major)	*(Minor)*	*(bits 15..8)*	*(bits 7..0)*
Bytes 5..(*m*-1)	Protocol message(s)			
Bytes *m*..(*p*-1)	MAC (optional)			
Bytes *p*..(*q*-1)	Padding (block ciphers only)			

COLLEGE OF SOCIAL & BEHAVIORAL SCIENCES
School of Information

heartbeat extension



http://heartbleed.com/

image credit: Vitaly Shmatikov

Many things can go wrong with TLS: one famous recent thing was the Heartbleed vulnerability

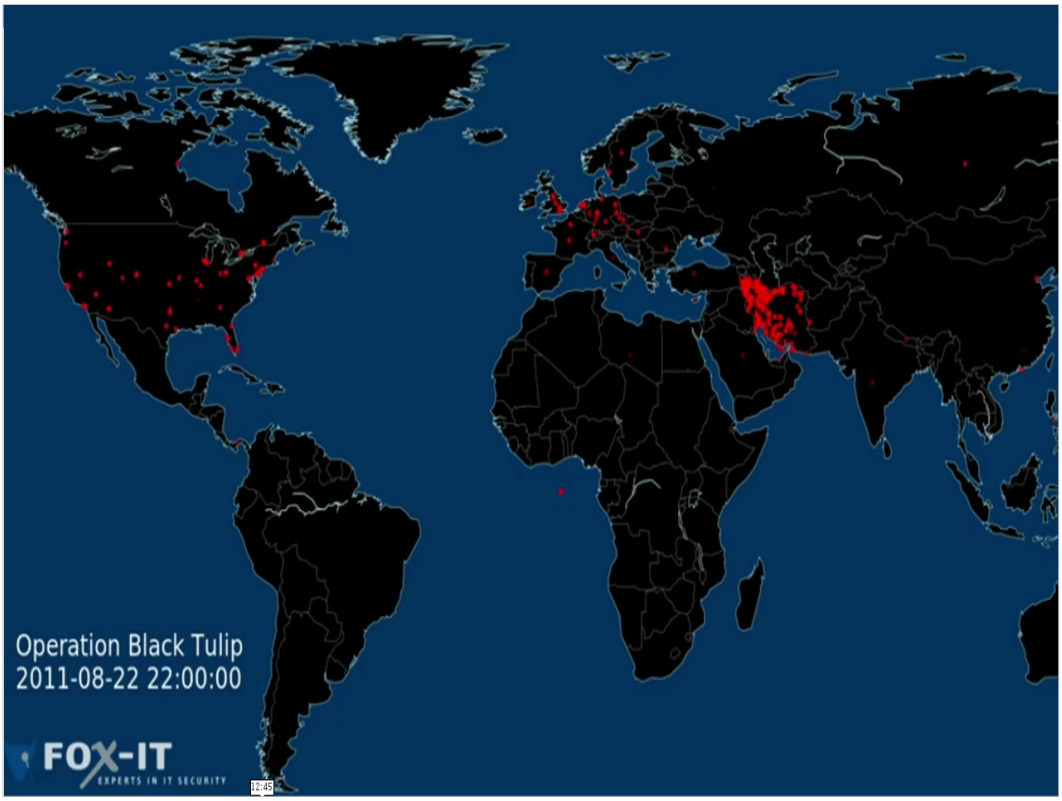
There are more fundamental problems, too, to do with the trust model (draw architecture on the board)

DigiNotar

Who got MitM'd:

- at least 300.000 unique IPs
- $>$ 99% from Iran
- identified using OCSP requests
- others: Tor, VPN, proxies ...

credit: Martin Schmiedecker

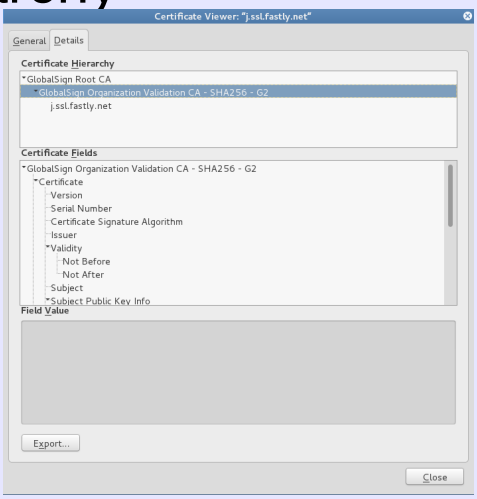


Logged OCSP requests for fraudulent certs

COLLEGE OF SOCIAL & BEHAVIORAL SCIENCES
School of Information

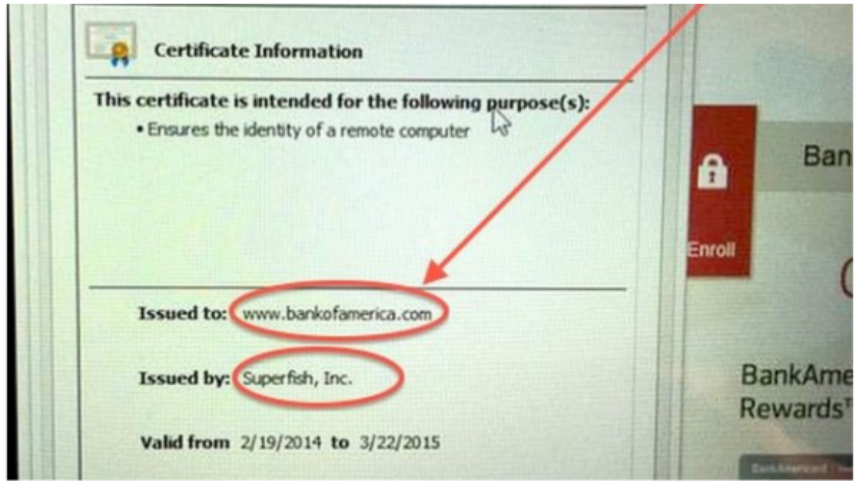
Certificate authorities form a hierarchy

- CA is a verifier of identity and domain ownership, to avoid problems arising from impersonation
- Root CAs sign certificates for Intermediate CAs, who sign for lower-level CAs: trust chain

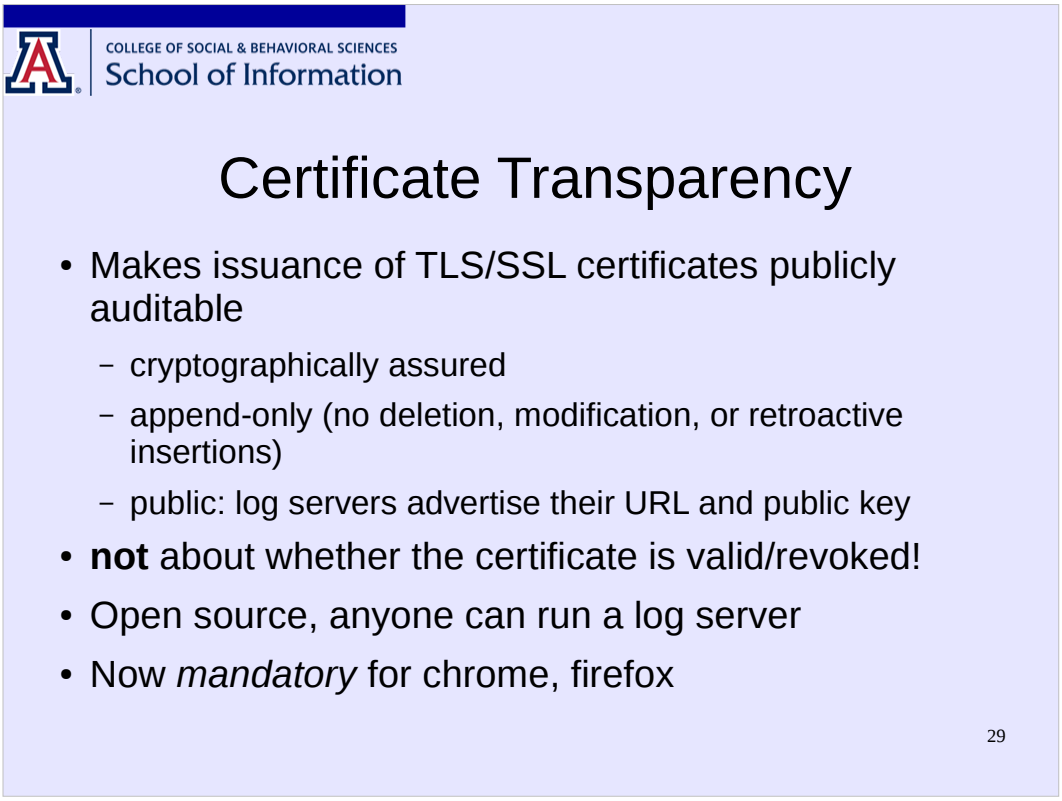


the word is ‘delegation’---we’ve seen it before

Superfish and the trouble with certs



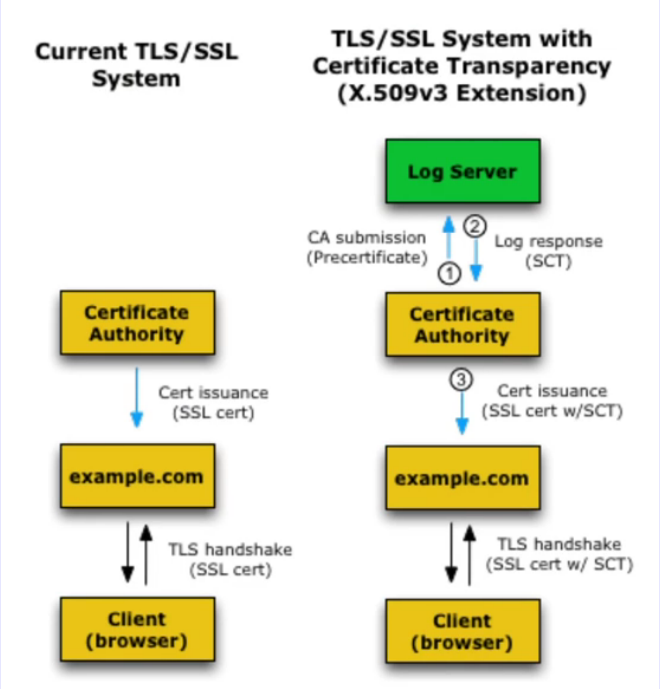
Much more is wrong:
https://www.eff.org/deeplinks/2010/03/researchers-reveal-likelihood-governments-fake-ssl



It would be GREAT if CAs made public all certs that they issued. Unfortunately, sometimes CAs have historically done under-the-table stuff, like sell certs to state actors or businesses that want to MiTM traffic, and they don't want this to be known. Also, sometimes CA's are compromised (see: DigiNotar)

So better we should introduce some technology to ensure transparency: that's RFC 6962. Helps to detect misbehaving CAs and identify fraudulent certs.

This has caught Symantec, WoSign, CNNIC cheating, so now they have to run CT (forced by Google)



the CA issues a cert and the website installs it in the right place, so that the clients can check it

with CT, nothing different for the client. The CA now not only issues the cert, but first sends a precert to the log server, which returns it with an SCT. The SCT is included in the certificate that is shipped to the client.

Signed Certificate Timestamp

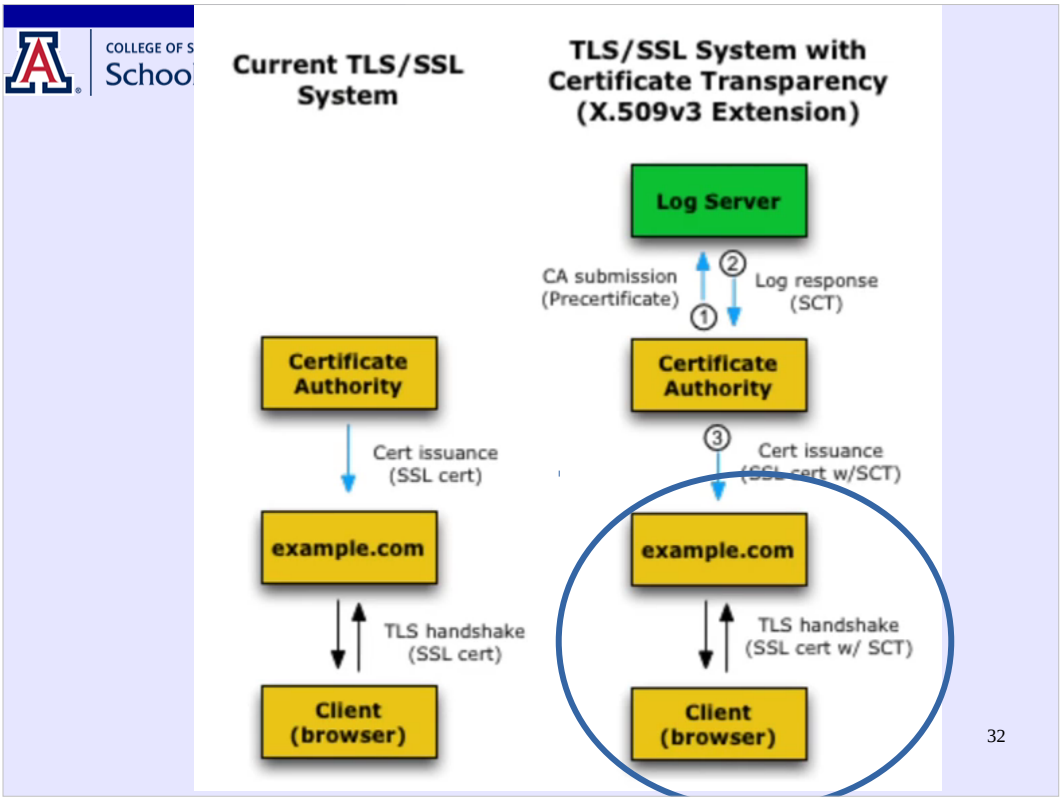
3.2. Structure of the Signed Certificate Timestamp

```
enum { certificate_timestamp(0), tree_hash(1), (255) }
  SignatureType;

enum { v1(0), (255) }
  Version;

  struct {
      opaque key_id[32];
  } LogID;

opaque TBSCertificate<1..2^24-1>;

  struct {
    opaque issuer_key_hash[32];
    TBSCertificate tbs_certificate;
  } PreCert;

  opaque CtExtensions<0..2^16-1>;
```

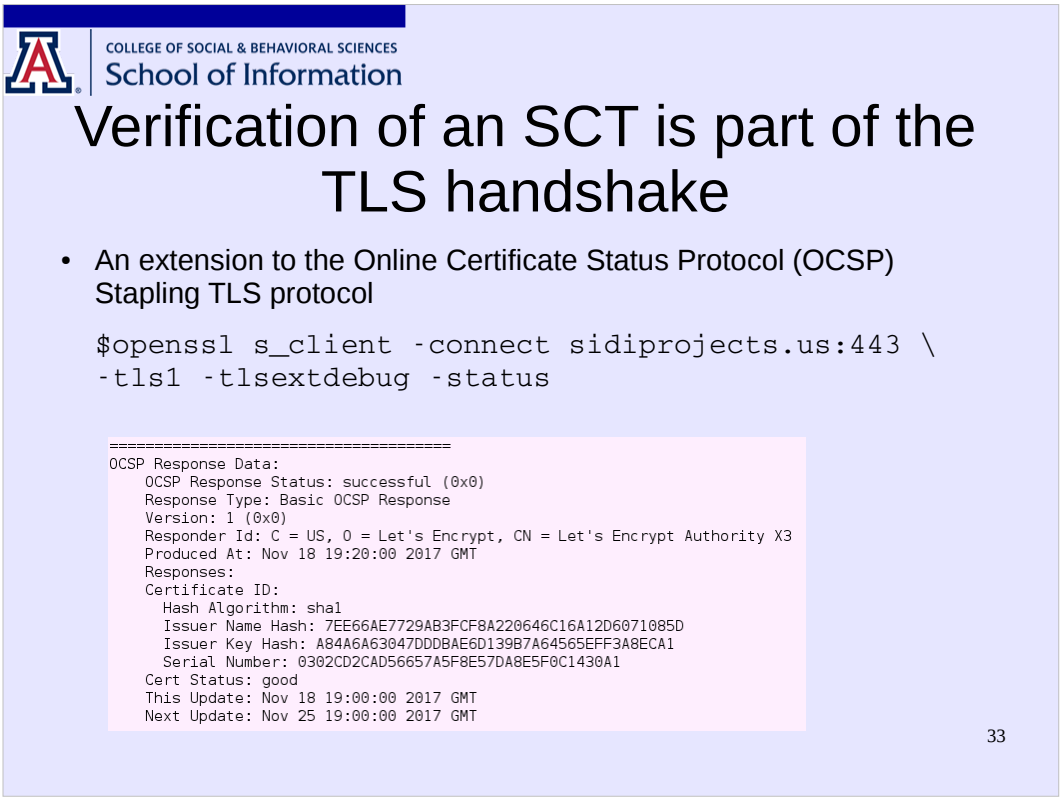
- subject of the certificate's name
- issuer's name

- public key of the subject
- validity period
- version number and a serial number

RFC 5280 https://tools.ietf.org/html/rfc5280#section-4.1.2



let’s look at the client-side for a second

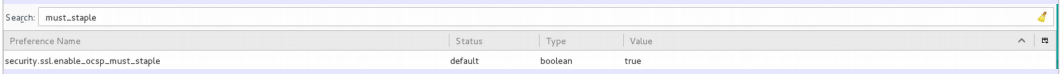


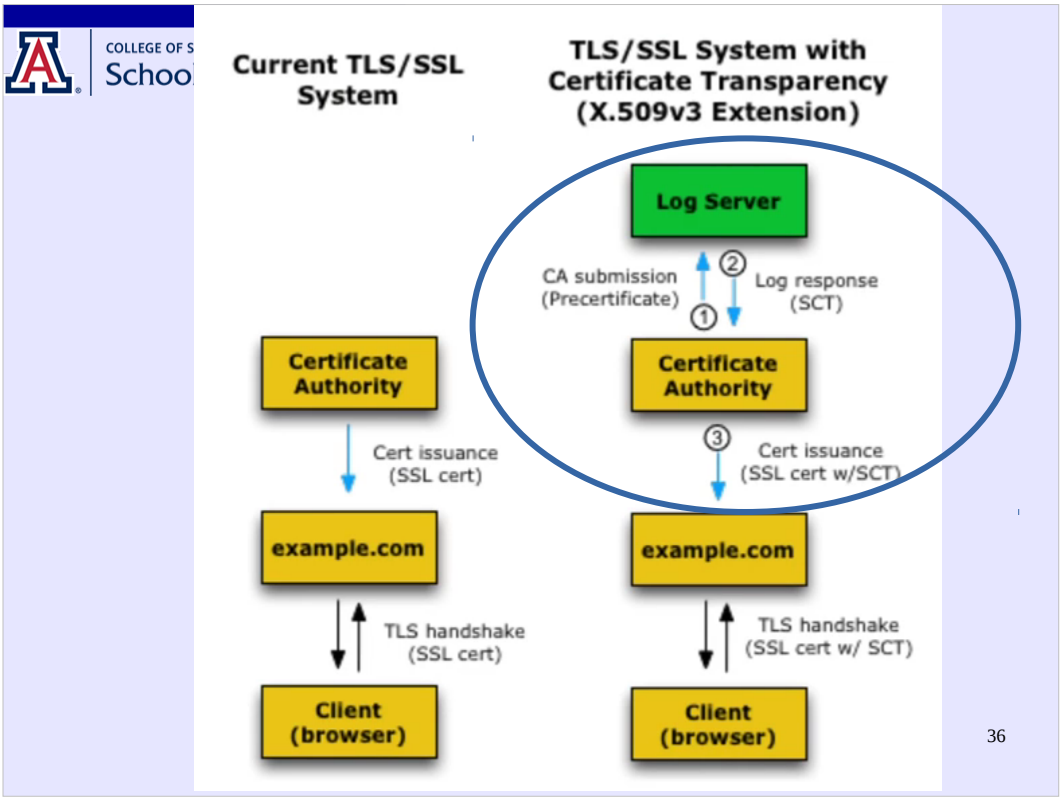
the ocsp response is combined with the certificate and sent to the client

ocsp is used to check other aspects of cert validity; CT is an extension

OCSP stapling is better than the alternatives

- There are other ways for the client to verify the SCT in the TLS handshake (x509v3 certificate extensions, or TLS extensions)
- OCSP stapling does not require going out to the CA
  - the OCSP request, signed by the CA, is combined with the certificate and sent to the client
  - SCT can be included as part of this stapling
- Why might contacting the CA be a negative thing?



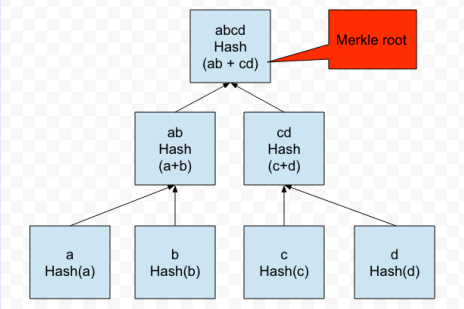


now let's look at the CA and log server side

COLLEGE OF SOCIAL & BEHAVIORAL SCIENCES
School of Information

Log servers use Merkle Hash Trees to keep track of the certificates

- Binary tree
- Calculated from the leaves: combine children's hashes to get the parent hash
- Can check integrity of a whole lot of hashes by checking one hash!
- All changes are auditable



credit

Nothing is deleted, ever

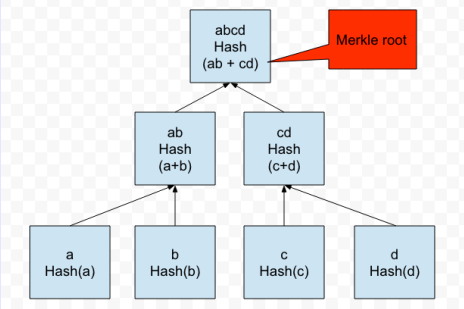
Still MTs are efficient, and prevent retroactive additions, removals

Batches are integrated periodically, and the global MT is recalculated, and the tree head signed

COLLEGE OF SOCIAL & BEHAVIORAL SCIENCES
School of Information

Log servers use Merkle Hash Trees to keep track of the certificates

- Can catch CA's that are adding and removing illicit certificates
- Can catch cheating log servers
- Not enough to just calculate the root value to audit the log once new hashes are added. Why not?



credit

Idea is to prevent a cert from being offered for a domain without the domain owner knowing about it

COLLEGE OF SOCIAL & BEHAVIORAL SCIENCES
School of Information

Walk through: auditing a log addition

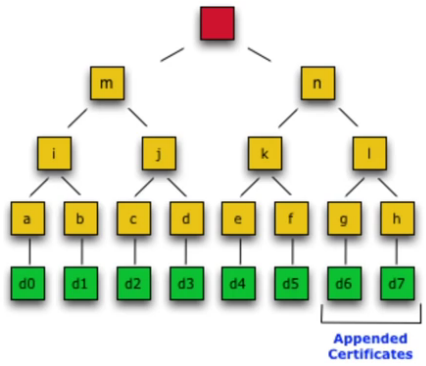


Figure 3

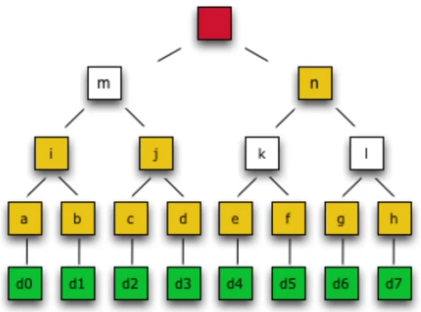
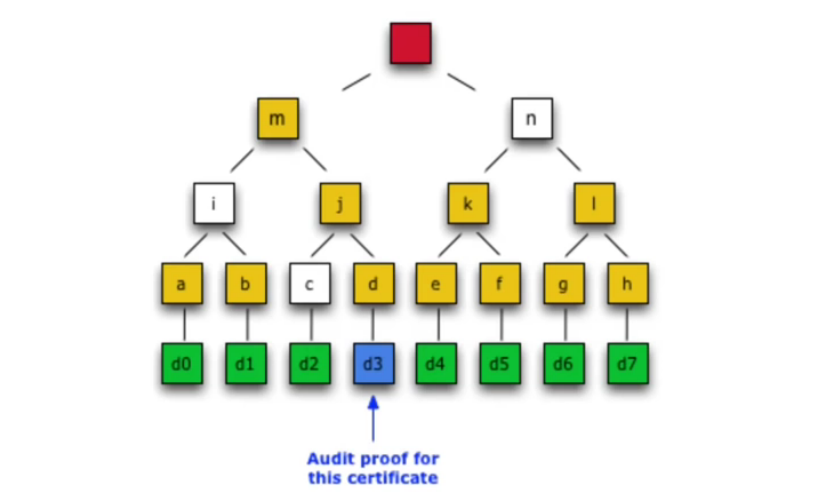


Figure 4

validate that d6 and d7 have been added in the tree correctly---notice the problem when you go from one tree to another with an addition. It's not enough to just calculate the root hash. (why not?) So in addition, send the nodes highlighted. Still not too many; this is efficient.

COLLEGE OF SOCIAL & BEHAVIORAL SCIENCES
School of Information

Walk through: Auditing for presence of a particular certificate



not just overall consistency of the log, but that a particular certificate is present. The SCT gives the promise to add the certificate, this proves that the certificate made it in.

Log servers are still centralized in practice

- In theory, anyone can run a log
- In practice, there are only a few
  - Digicert: the first
  - Google: their idea; they run the big ones

```
$curl ct.googleapis.com/icarus/ct/v1/get-sth
{"tree_size":148531007,
"timestamp":1511196824947,
"sha256_root_hash":"bRmJZDeJZIs/WTOYZ3pA+MyJuOEZ9m+XGZIRU9fnViI=
",
"tree_head_signature":"BAMASDBGAiEAk+md3GDvKIPyuQ27UnLdDhKoVB5hn
zVDA8ZX1Dkx/JgCIQCDmYMAi6oqpAXk+LV/vIKwfrhyaCNrXl7N37moFv/BfA=="
}
```

- Use crt.sh to search manually from the browser. Certspotter can help you monitor your domains (https://sslmate.com/certspotter/)

```
Cert Spotter has discovered the following certificates for domain(s) on
your watch list:

            Issuer: C=US, O=Let's Encrypt, CN=Let's Encrypt Authority X3
          DNS Name: sidiprojects.us
          DNS Name: www.sidiprojects.us
       Fingerprint:
09354b0d8b549e5ec5eaffe1f4fe740b9b9be946603d6df95ec150500ab3710f
```

Other ways to fix TLS

- Using GPG, with monkeysphere
  - http://web.monkeysphere.info/
- Flexible trust model of WoT used for PKI
- Problem: goes out to the keyserver for failing requests