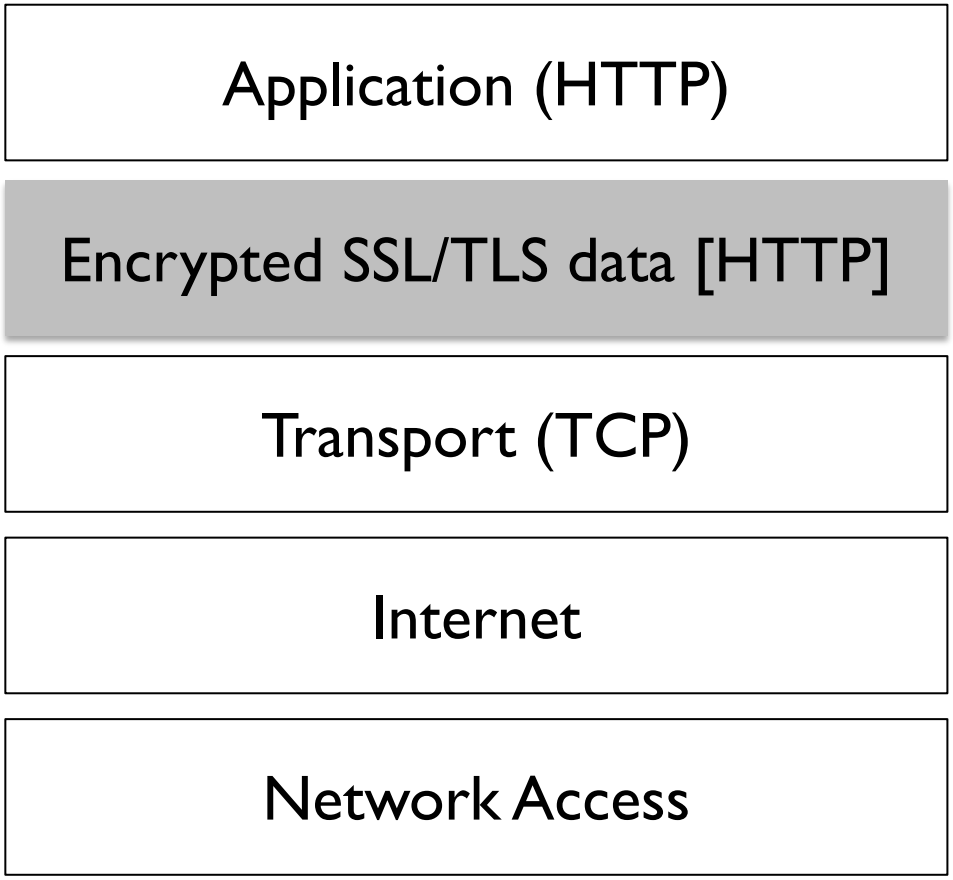
Crypto Application SSL/TLS

Issue Date:

Revision:

APNIC

SSL/TLS Protocol



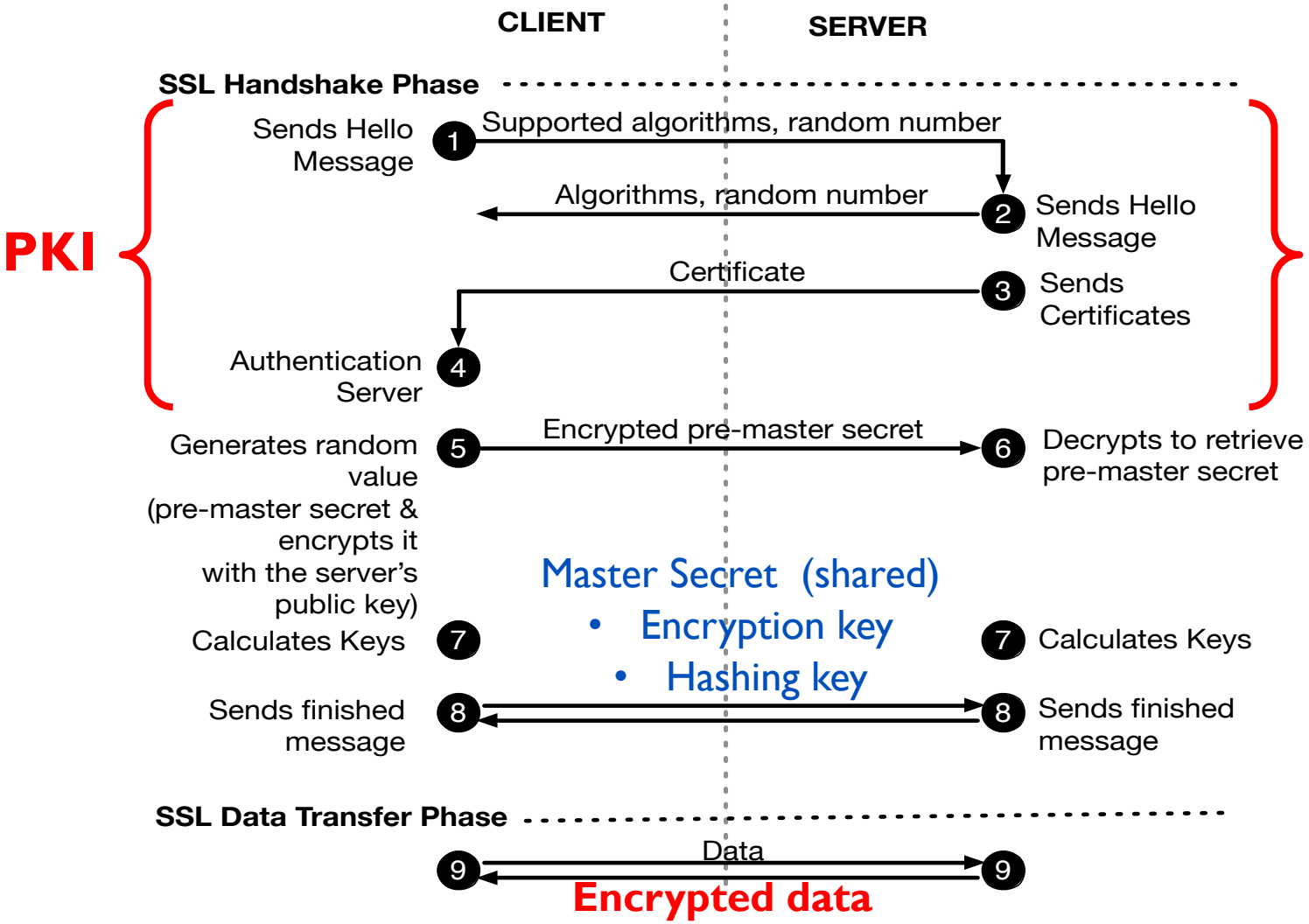
TLS/SSL: What it does

- Confidentiality
  - Encryption
- Integrity
  - Keyed hash (HMAC): TLS (authentication!)
  - Hash (MAC): SSL
- Authentication
  - certificates

SSL/TLS Operations

- Client connects to the server
  - To access a resource
- Public-key cryptography during initial handshake to authenticate and exchange session keys
  - PKI (X.509 Certificates)
- Symmetric key cryptography to encrypt and hash data
  - Master secret (shared secret) generated
  - Separate **Encryption** and **Hashing** keys from the master secret

How SSL/TLS Works – Part 1



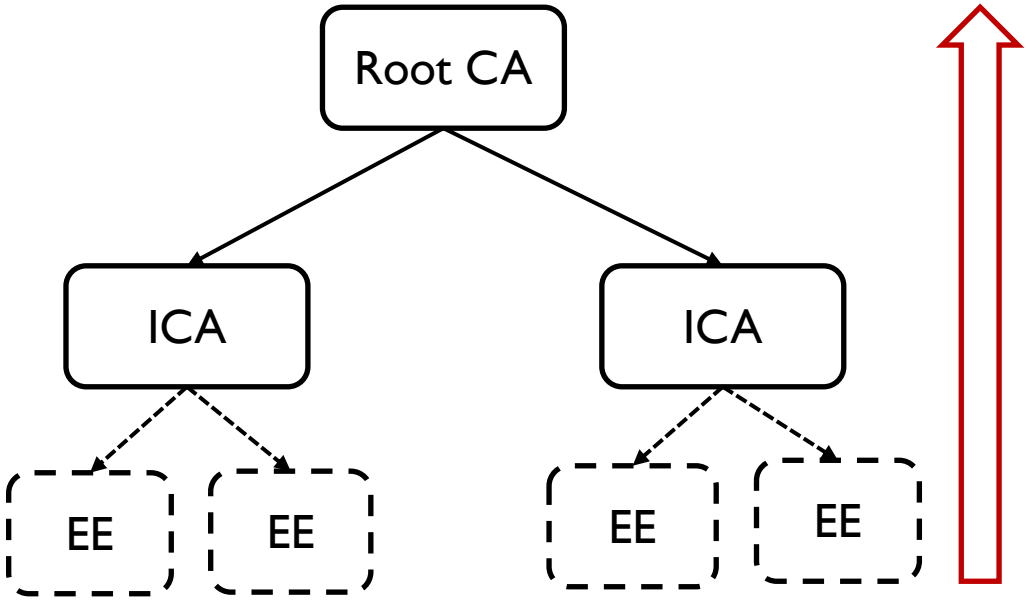
PKI – public key infra

- Digital (X.509) certificates
  - associates a public key with an individual or organization

VERSION	Version of X.509
SERIAL NUMBER	Uniquely identifies the certificate
SIGNATURE ALGORITHM	Algorithms used by the CA to sign the cert
ISSUER NAME	Id of the CA (that issued the cert)
VALIDITY PERIOD	Cert validity
SUBJECT NAME	Entity associated with the public key
SUBJECT PUBLIC KEY	Owner's public key
EXTENSIONS (ISSUER KEY ID)	Identify the pub key of issuer of the cert
EXTENSIONS (SUBJECT KEY ID)	Extra info (owner of the cert)
EXTENSIONS (CRL)	Extensions (CRL)
CA DIGITAL SIGNATURE	Certifies the binding – sings pub key of subject

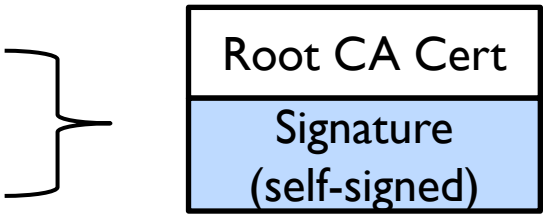
PKI – Chain of Trust

- Root CA
  - Self-signed
  - Issue and sign ICA's certificate
- Intermediate CA
  - Issue and sign EE certificate
- End Entity

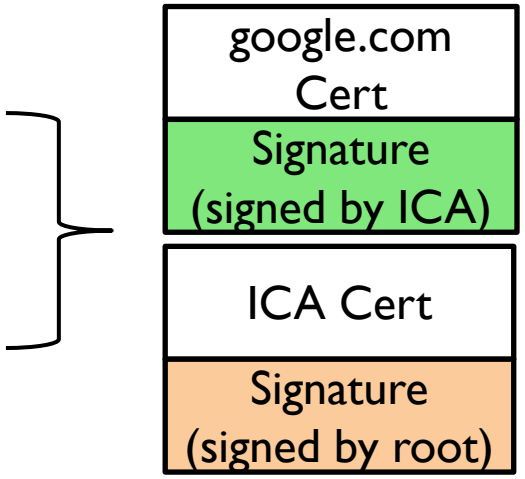


PKI – Example

- Client (browser) sends https request to `google.com`
  - browsers have trusted CA certificates stored

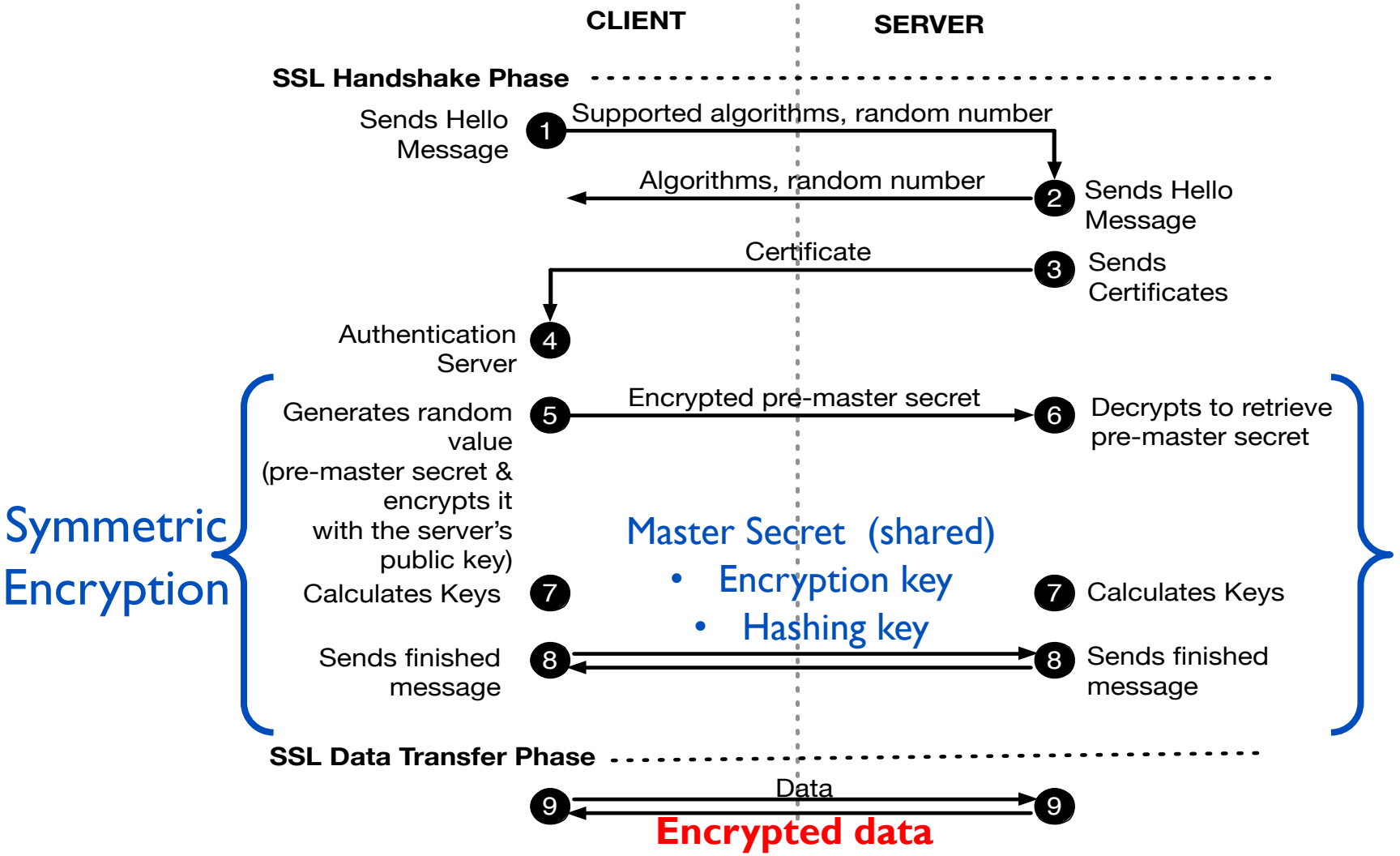


- Web server sends back `google.com`'s certificate
  - Signed by Google ICA, plus
  - Google ICA's certificate signed by root CA (GeoTrust)



- Verify the certificates up the chain of trust
  - Once successfully verified, use the public key

How SSL/TLS Works – Part 2



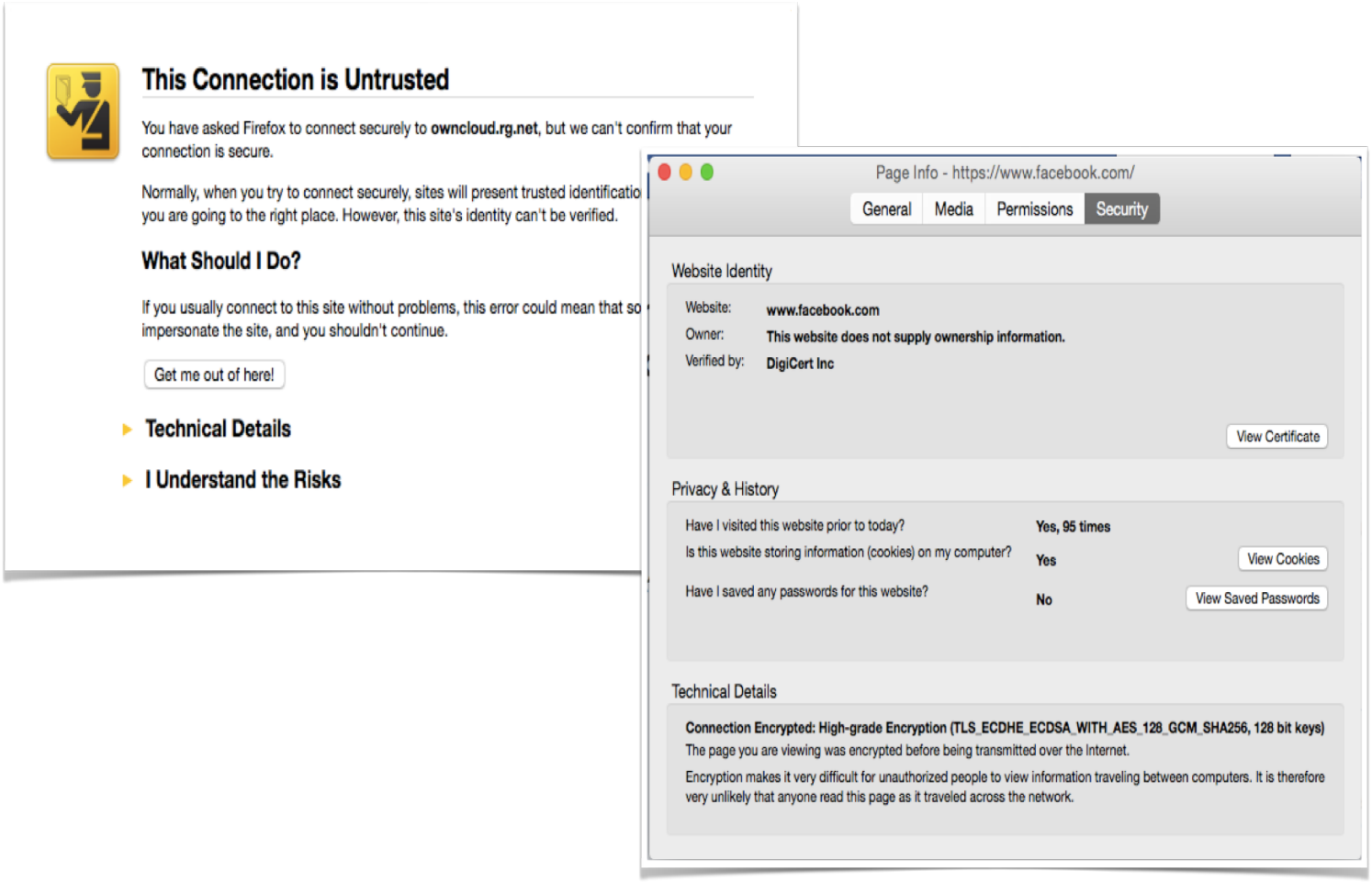
Symmetric Encryption

- Once the server's public key is verified up the chain of trust
  - Client generates a pre-master secret (`C-random & S-random`)
  - Sends to the server encrypted (with server's public key)

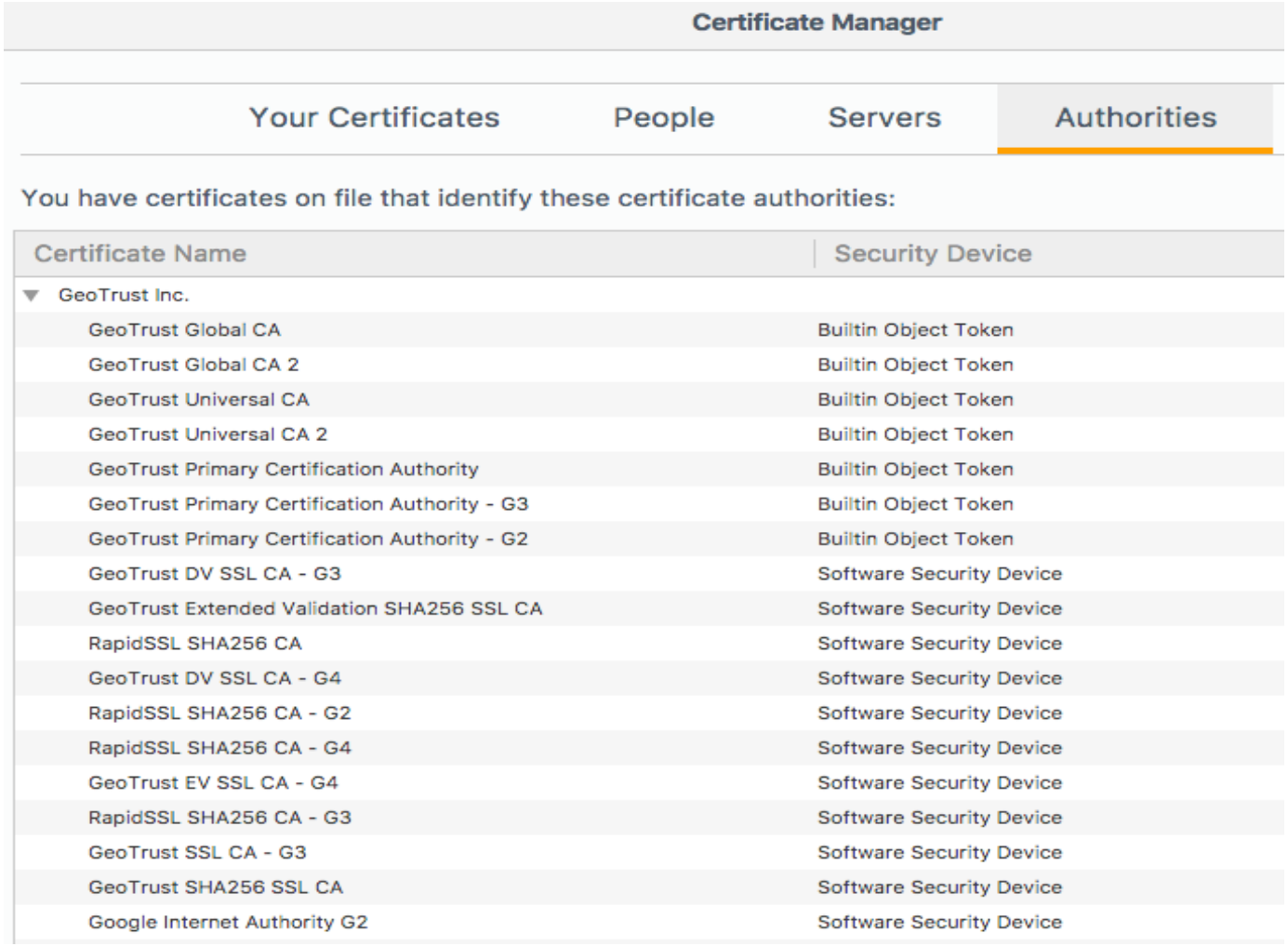
- Both client and server generates the Master Secret
  - Uses the `pre-master secret`, `C-random`, and `S-random` with the agreed `key exchange cipher` (eg: DH)

- Separate Encryption and Hashing keys generated from the Master secret
  - All future communication hashed and encrypted using the symmetric keys

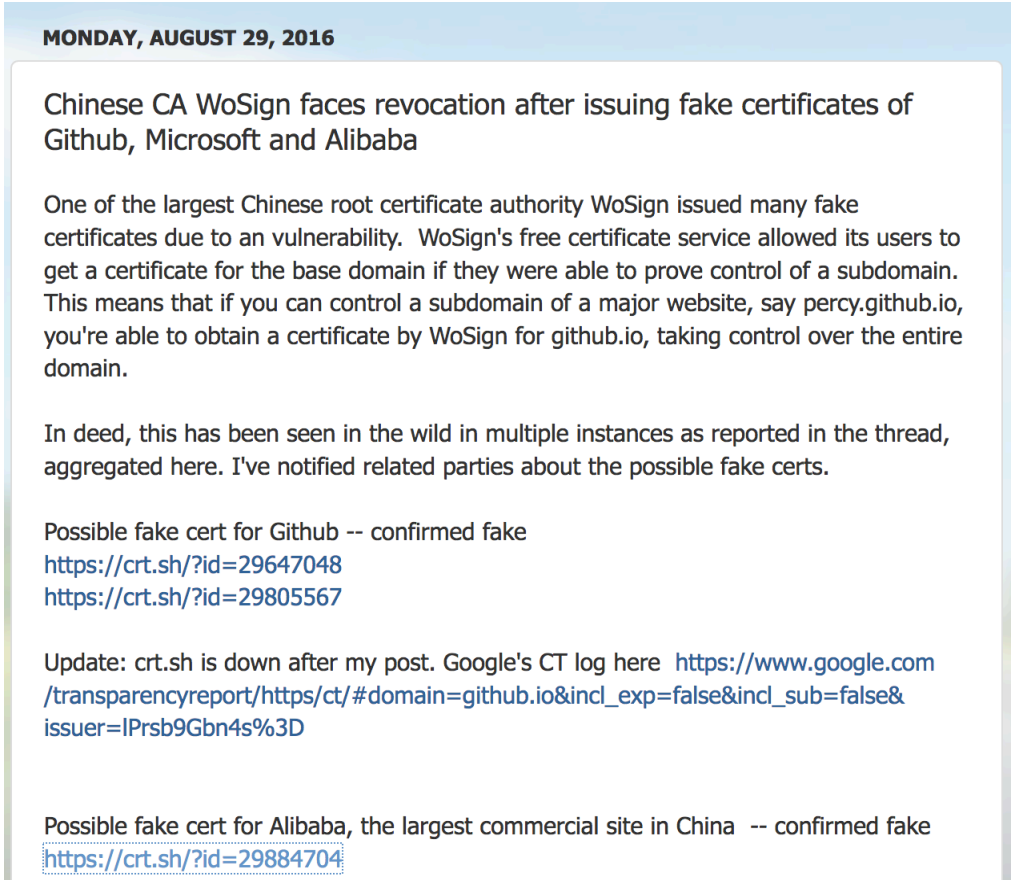
Trusted vs Non-trusted Certificate



Certificate Authority



WoSign



MONDAY, AUGUST 29, 2016

Chinese CA WoSign faces revocation after issuing fake certificates of Github, Microsoft and Alibaba

One of the largest Chinese root certificate authority WoSign issued many fake certificates due to an vulnerability. WoSign's free certificate service allowed its users to get a certificate for the base domain if they were able to prove control of a subdomain. This means that if you can control a subdomain of a major website, say percy.github.io, you're able to obtain a certificate by WoSign for github.io, taking control over the entire domain.

In deed, this has been seen in the wild in multiple instances as reported in the thread, aggregated here. I've notified related parties about the possible fake certs.

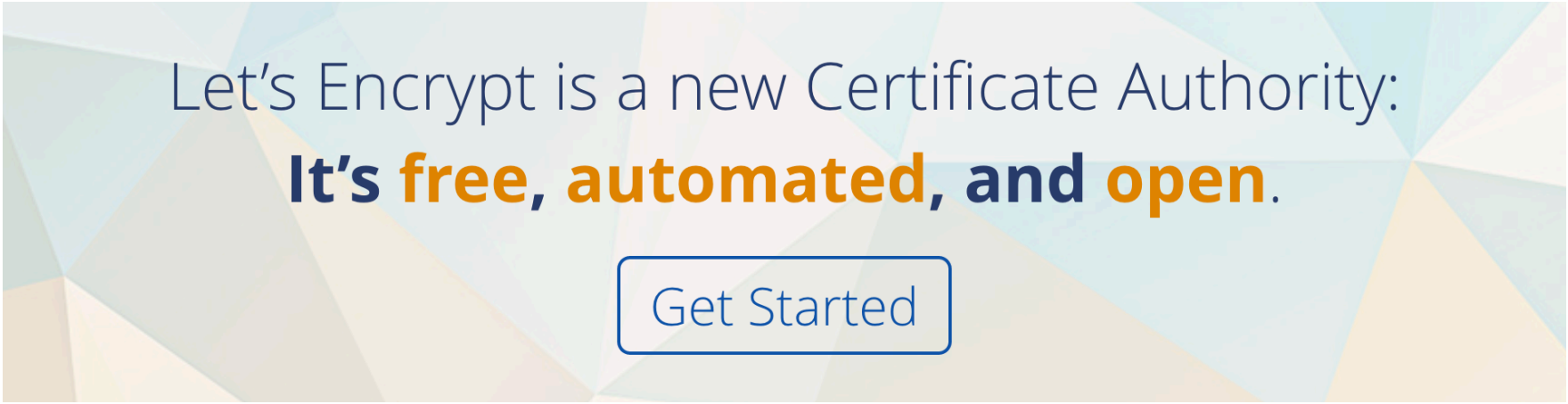
Possible fake cert for Github -- confirmed fake
https://crt.sh/?id=29647048
https://crt.sh/?id=29805567

Update: crt.sh is down after my post. Google's CT log here https://www.google.com/transparencyreport/https/ct/#domain=github.io&incl_exp=false&incl_sub=false&issuer=lPrsb9Gbn4s%3D

Possible fake cert for Alibaba, the largest commercial site in China -- confirmed fake
https://crt.sh/?id=29884704

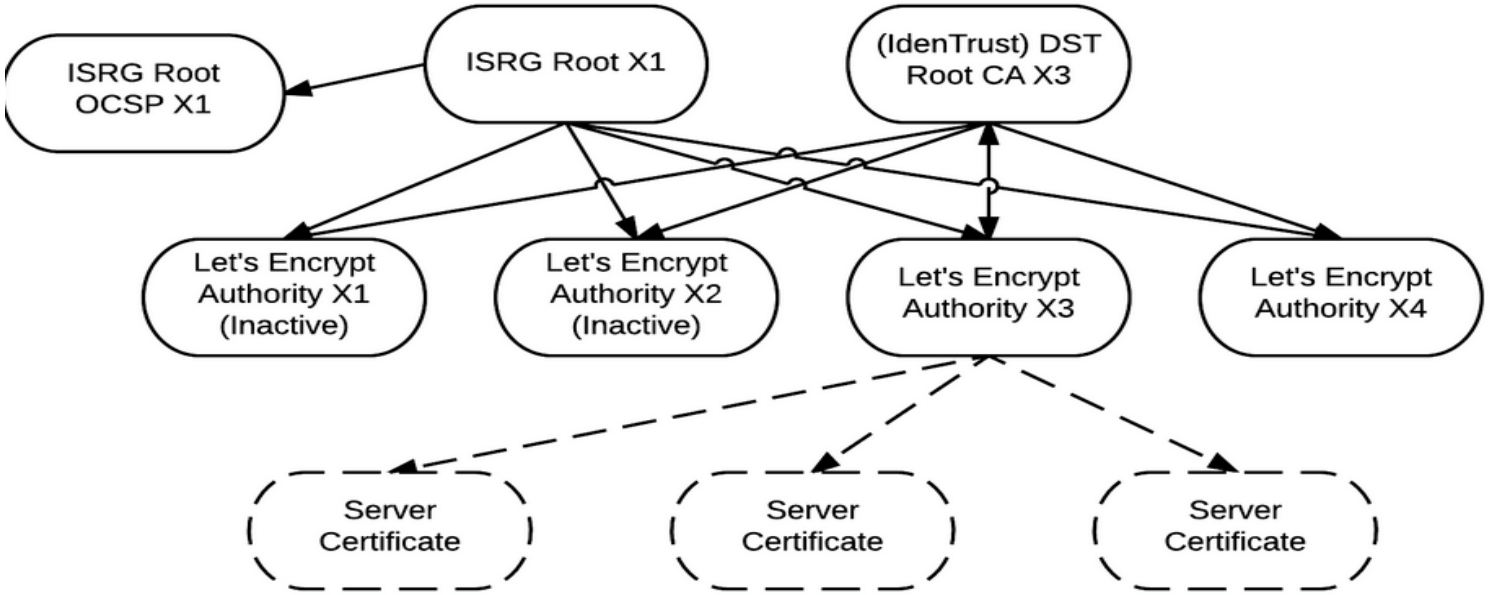
https://groups.google.com/forum/m/#!topic/mozilla.dev.security.policy/k9PBmyLCi8I/discussion

Introducing Let's Encrypt



- An open source CA
  - Proof your domain to get your digital (TLS/SSL) certificate
  - https://letsencrypt.org

Let's Encrypt chain



- Let's Encrypt ICA (X3) cross-signed by DST (IdenTrust)
  - Until ISRG (Internet Security Research Group) is trusted by everyone

https://letsencrypt.org/2018/08/06/trusted-by-all-major-root-programs.html

Introducing Let's Encrypt

- Browsers and OS
  - https://community.letsencrypt.org/t/which-browsers-and-operating-systems-support-lets-encrypt/4394

- Check your browser
  - https://wiki.apnictraining.net
    - (signed by Let's Encrypt)

Lab Exercise

Lab Exercise

- Follow lab manual **ssl_lab.pdf**