Server-side Adoption of Certificate Transparency

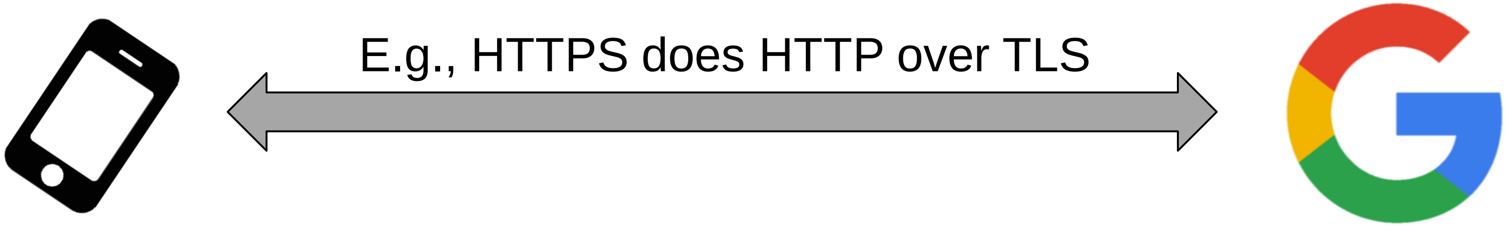
Carl Nykvist, *Linköping University*
Linus Sjöström, *Linköping University*
Josef Gustafsson, *Linköping University*
Niklas Carlsson, *Linköping University*

Proc. PAM, Berlin, Germany, Mar. 2018

LINKÖPING UNIVERSITY

Motivation and high-level problem

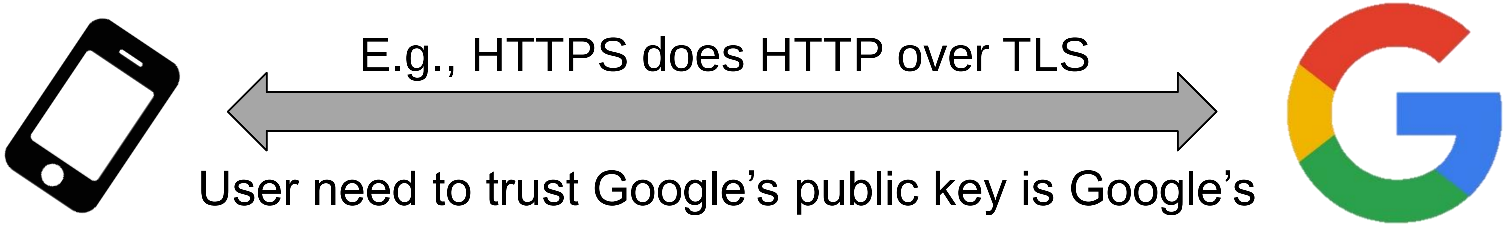
- Private and confidential communication important



E.g., HTTPS does HTTP over TLS

Motivation and high-level problem

- Private and confidential communication important

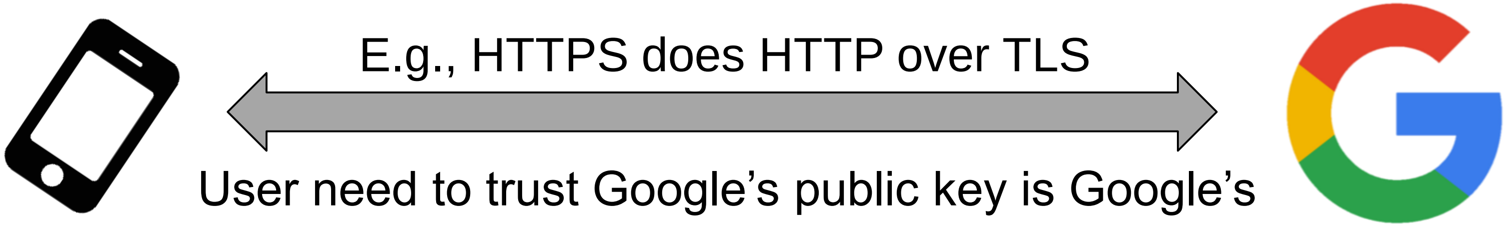


E.g., HTTPS does HTTP over TLS

User need to trust Google’s public key is Google’s

Motivation and high-level problem

- Private and confidential communication important
  - Billions of devices
  - Millions of services

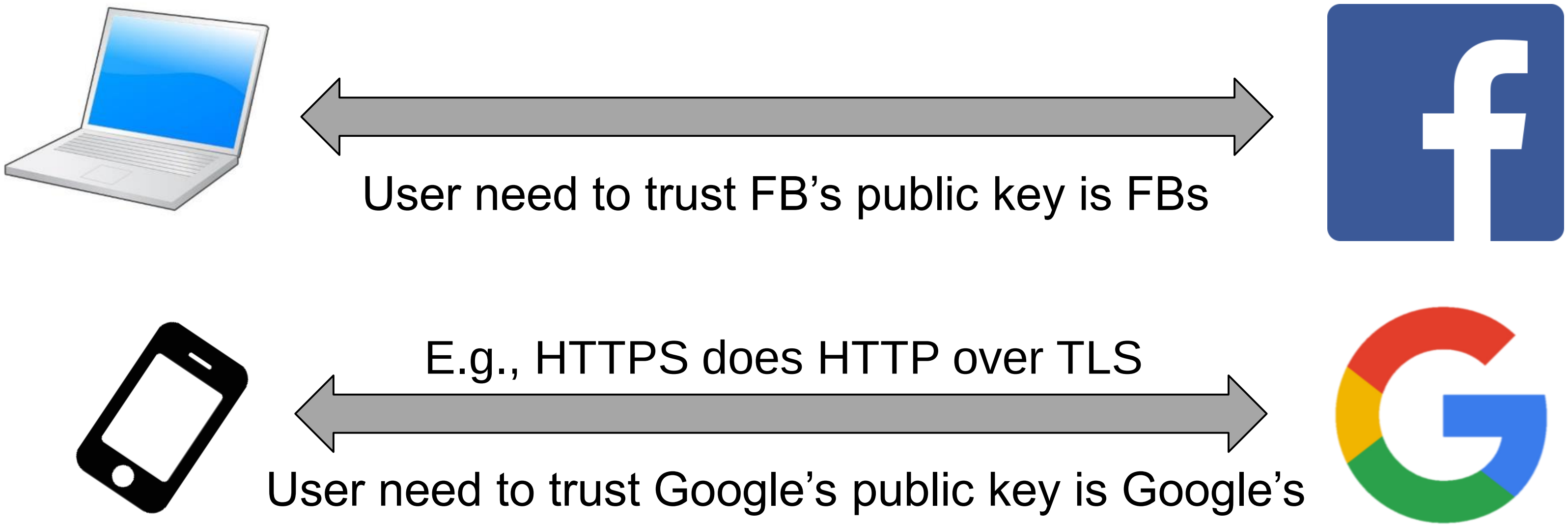


E.g., HTTPS does HTTP over TLS

User need to trust Google’s public key is Google’s

Motivation and high-level problem

- Private and confidential communication important
  - Billions of devices
  - Millions of services



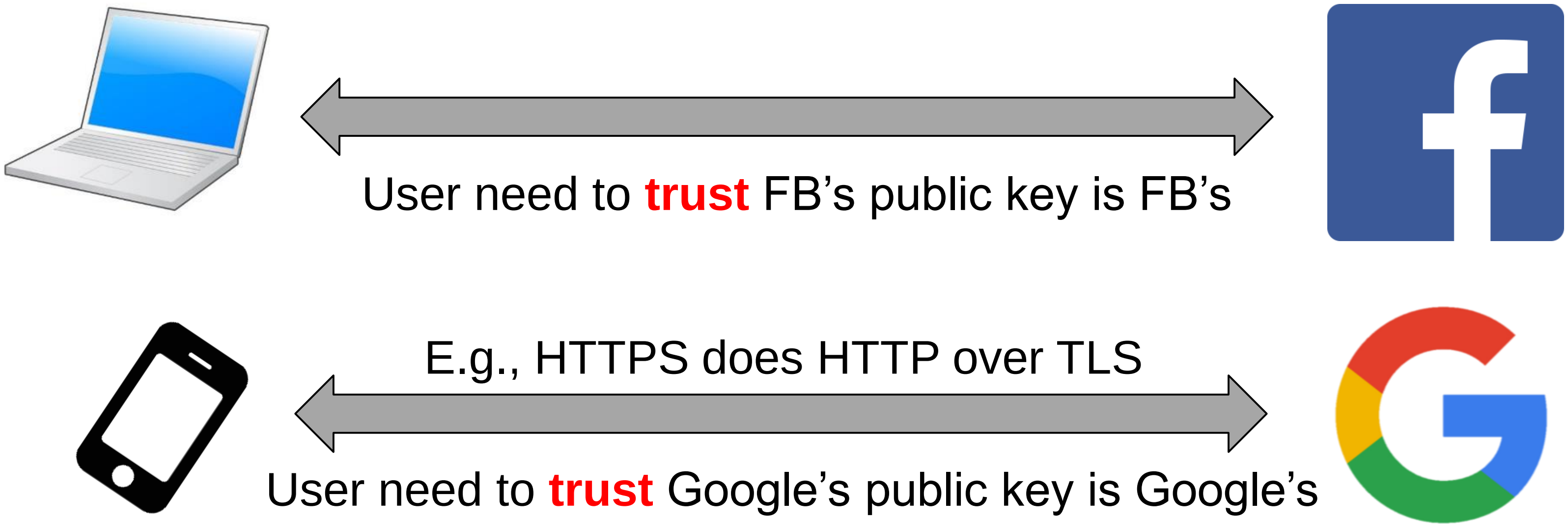
User need to trust FB's public key is FBs

E.g., HTTPS does HTTP over TLS

User need to trust Google's public key is Google's

Motivation and high-level problem

- Private and confidential communication important
  - Billions of devices
  - Millions of services
-
  -



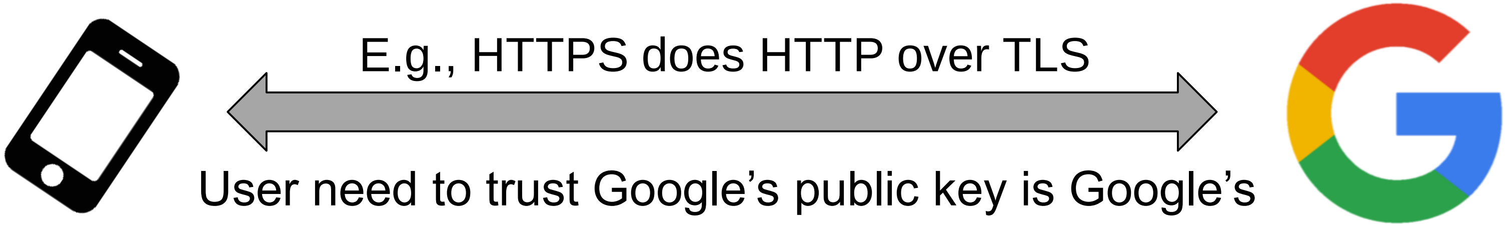
User need to **trust** FB’s public key is FB’s

E.g., HTTPS does HTTP over TLS

User need to **trust** Google’s public key is Google’s

Motivation and high-level problem

- Private and confidential communication important
  - Billions of devices
  - Millions of services
- Certification Authorities (CAs) issue certificates
  - Proof of identity (signed with their private key)

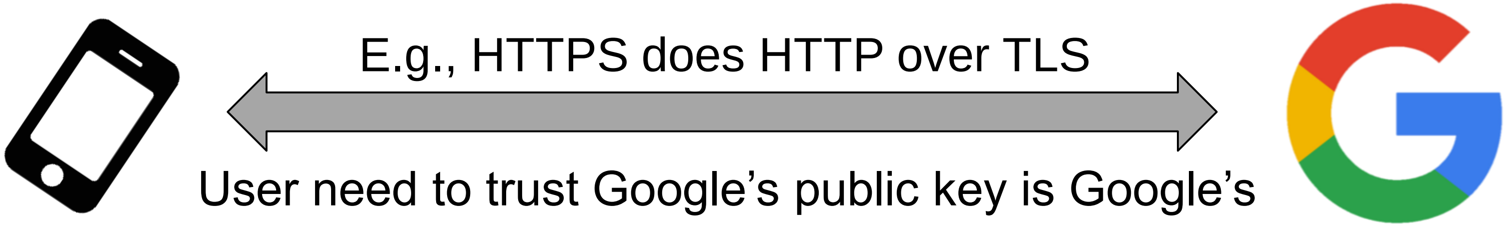


E.g., HTTPS does HTTP over TLS

User need to trust Google's public key is Google's

Motivation and high-level problem

- If CAs in our trust (root) store (e.g., Symantec/ Verisign) tells us that a public key belongs to Google, our browsers (and us) trust that this is the case

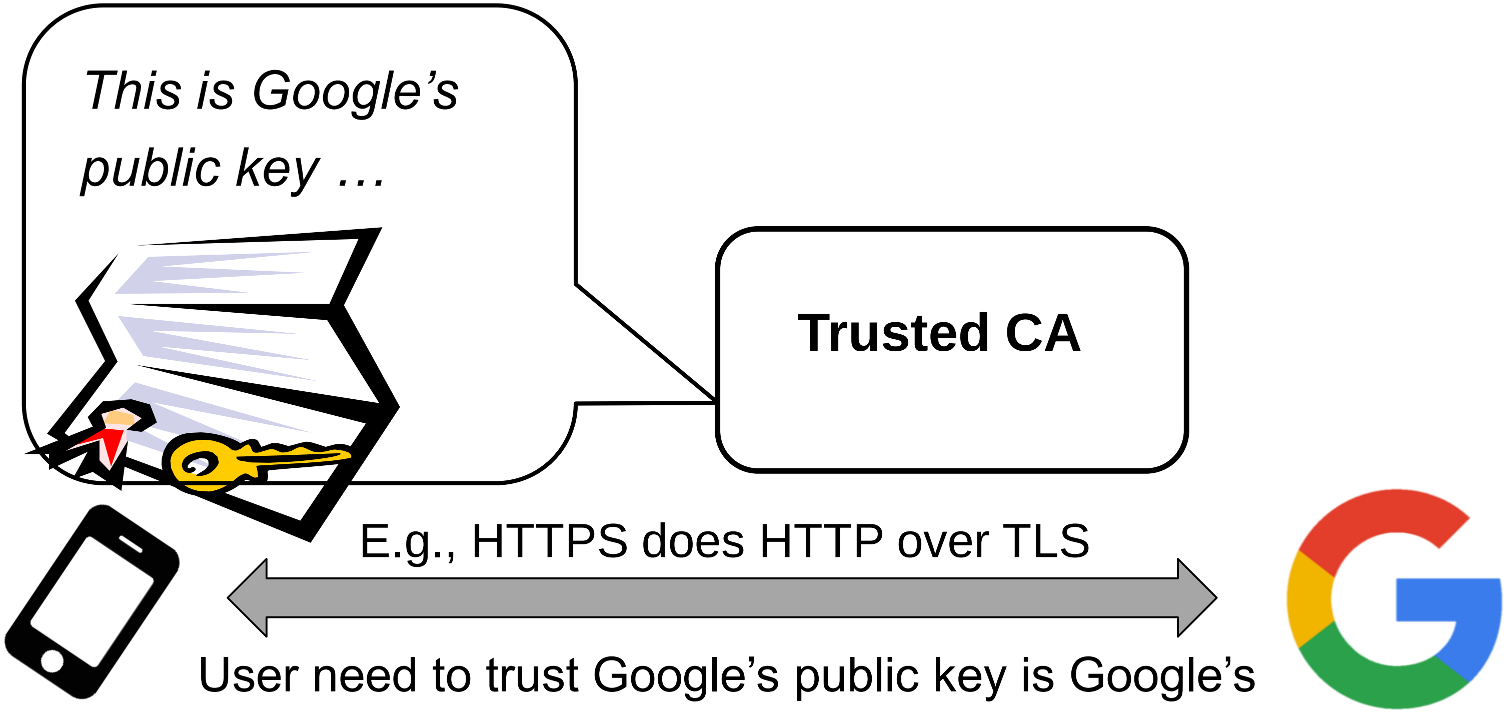


E.g., HTTPS does HTTP over TLS

User need to trust Google's public key is Google's

Motivation and high-level problem

- If CAs in our trust (root) store (e.g., Symantec/ Verisign) tells us that a public key belongs to Google, our browsers (and us) trust that this is the case



This is Google's public key …

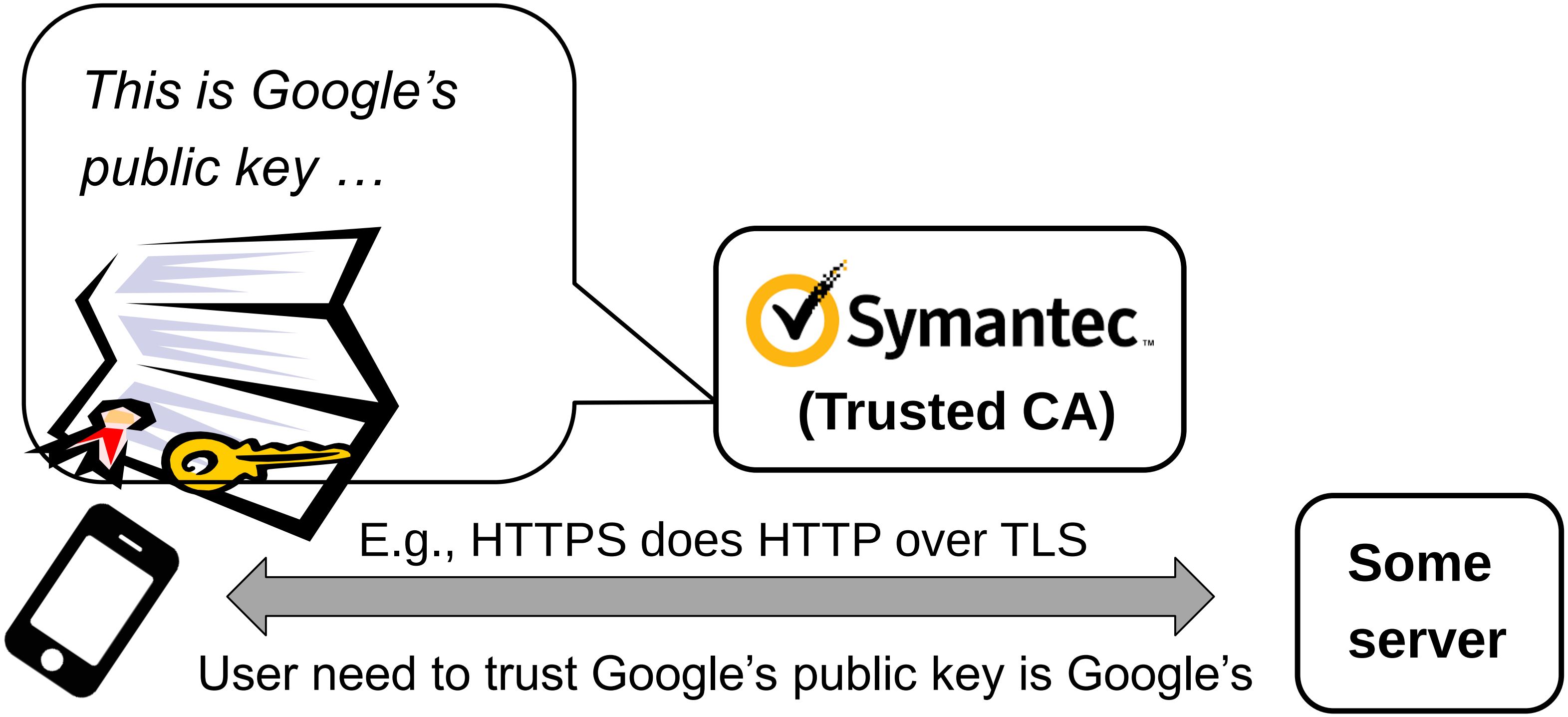
Trusted CA

E.g., HTTPS does HTTP over TLS

User need to trust Google's public key is Google's

Motivation and high-level problem

- However, mistakes happen ...
  - E.g., in Oct. 2015, Google discovered (using CT) that Symantec had issued test certificates for 76 domains that they did not own (including Google domains) and another 2,458 unregistered domains …



This is Google's public key …

Symantec
(Trusted CA)

E.g., HTTPS does HTTP over TLS

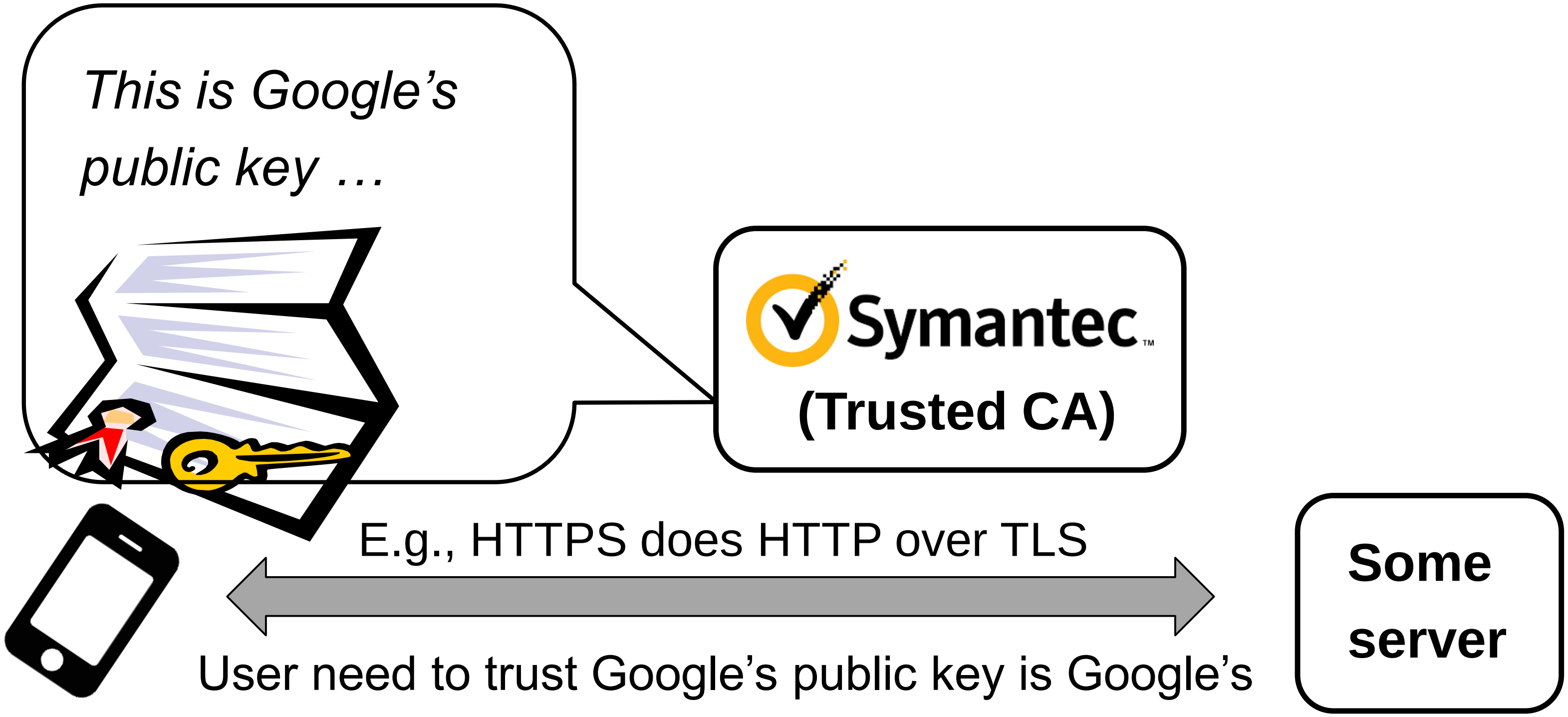
User need to trust Google's public key is Google's

Some server

CT: Emerging trust-monitoring solution

- Since then, Google has demanded that Symantec logs all their certificates in public (append-only) CT logs
- Since Jan. 2015, the Chrome browser requires all EV certificates be logged in 1 Google log and 1 other log
  - Mozilla planning to make similar demands
  - Both Chrome and Mozilla expected to implement policies for DV certificates too …

CT: Emerging trust-monitoring solution



This is Google's public key …

Symantec

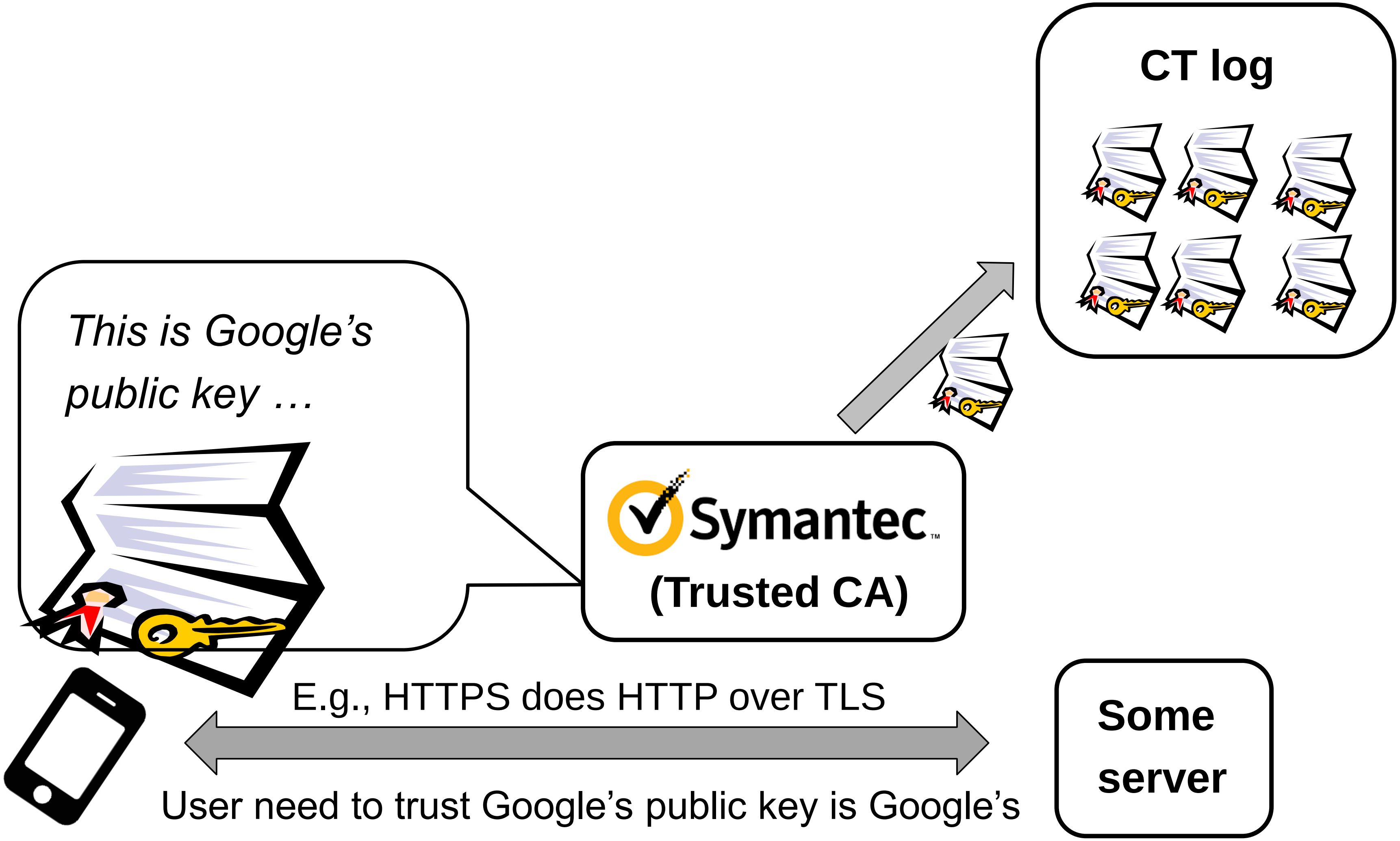
(Trusted CA)

E.g., HTTPS does HTTP over TLS

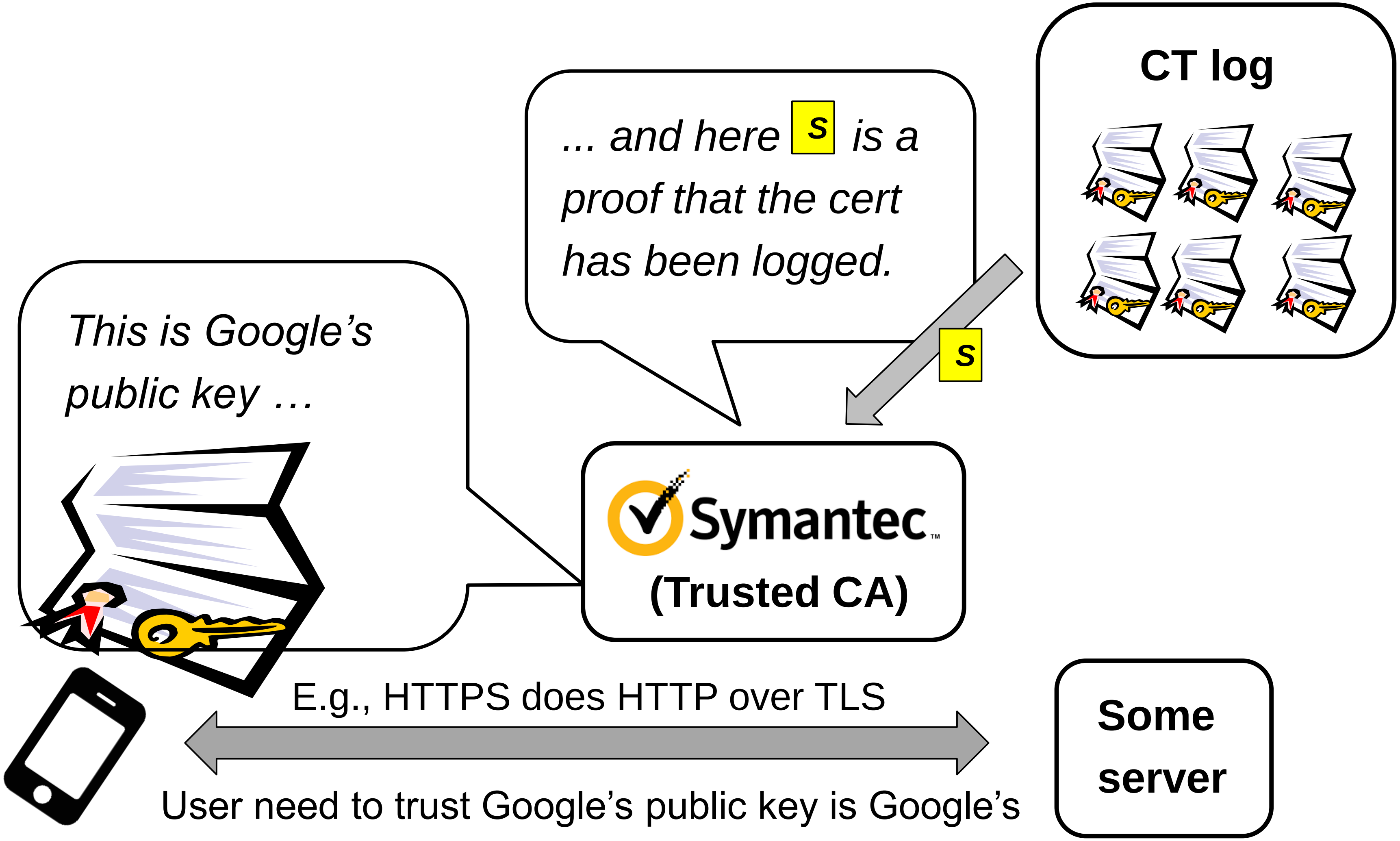
User need to trust Google's public key is Google's

Some server

CT: Emerging trust-monitoring solution


CT log
This is Google's public key …
Symantec
(Trusted CA)
E.g., HTTPS does HTTP over TLS
User need to trust Google's public key is Google's
Some server

CT: Emerging trust-monitoring solution



CT log

... and here **S** *is a proof that the cert has been logged.*

This is Google's public key …

S

Symantec

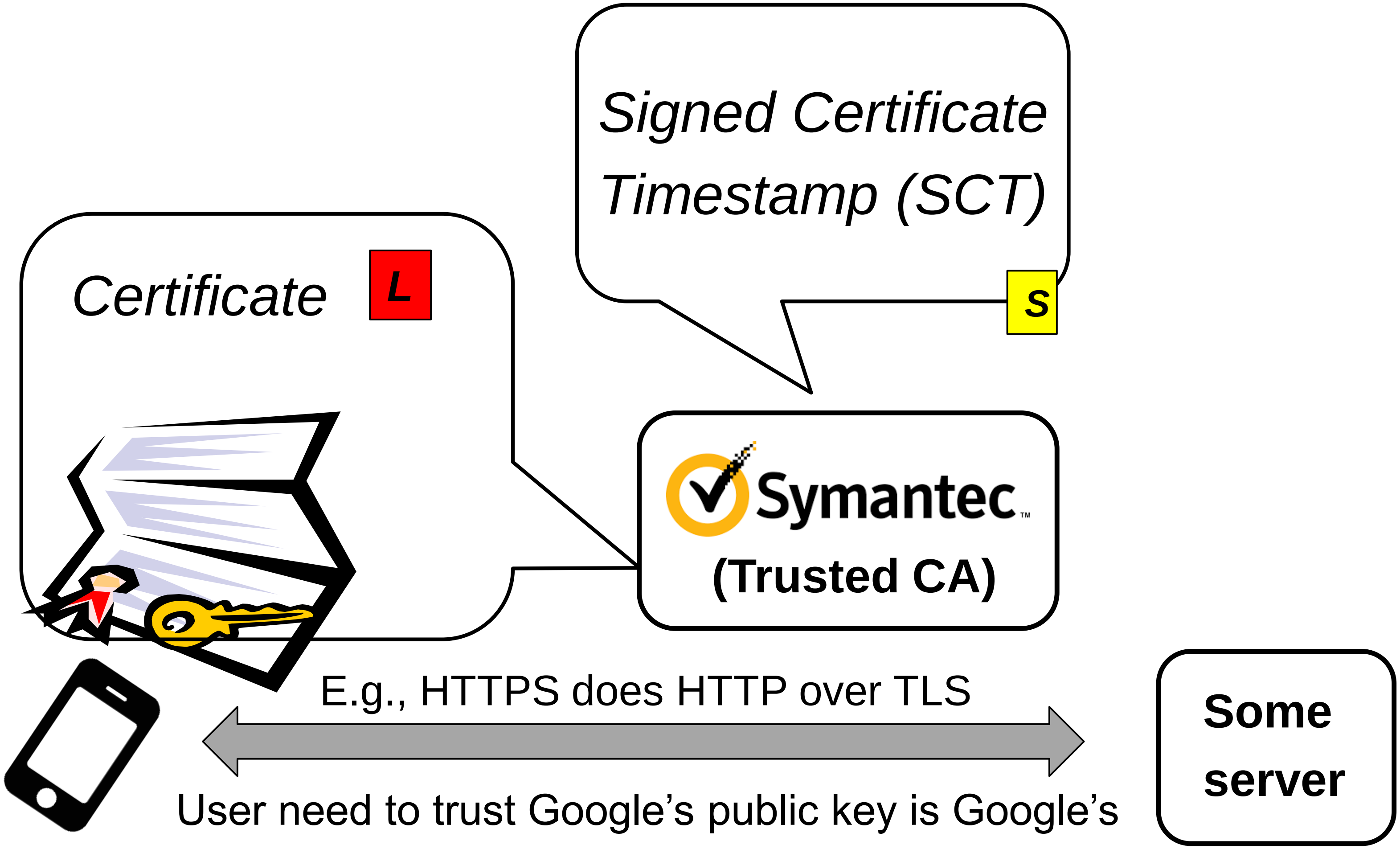
(Trusted CA)

E.g., HTTPS does HTTP over TLS

Some server

User need to trust Google's public key is Google's

CT: Emerging trust-monitoring solution



Signed Certificate Timestamp (SCT)

S

Certificate

L

Symantec

(Trusted CA)

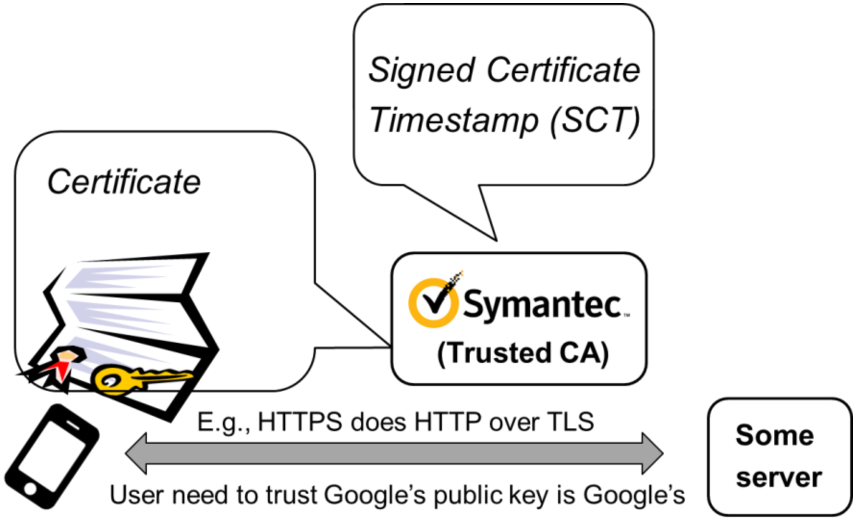
E.g., HTTPS does HTTP over TLS

Some server

User need to trust Google's public key is Google's

Signed Certificate Timestamps (SCTs)

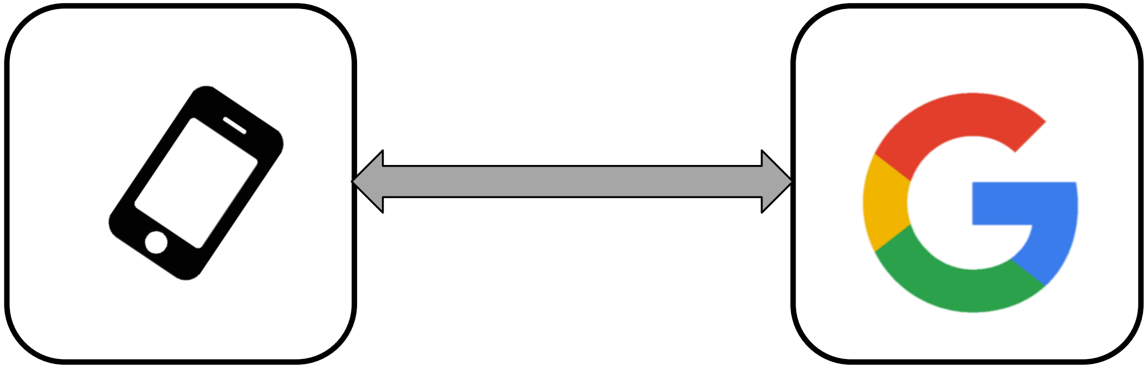
- SCTs delivered three different ways
  - X.509v3 extension
  - TLS extension
  - OSCP stapling
- In this paper, we characterize and compare
  - Server-side usage of these methods
  - Client-side performance of these methods


Signed Certificate Timestamp (SCT)
Certificate
Symantec
(Trusted CA)
E.g., HTTPS does HTTP over TLS
User need to trust Google's public key is Google's
Some server

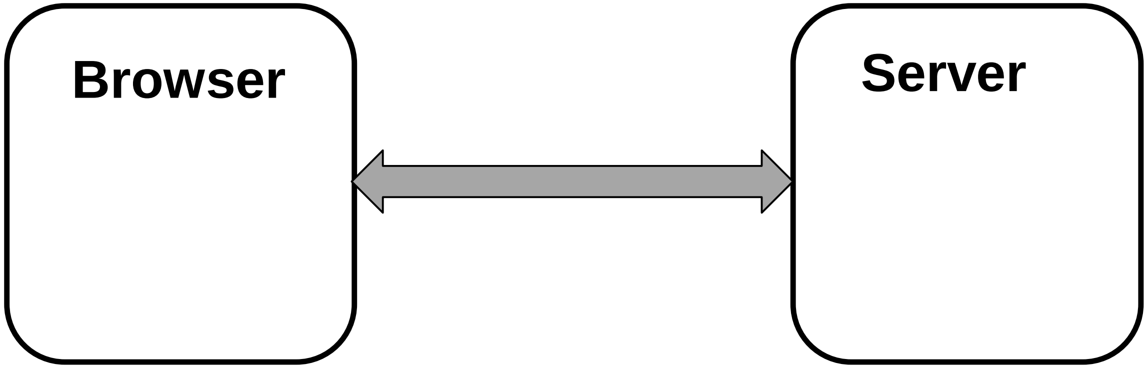
Background

Certification of public keys

Certification of public keys

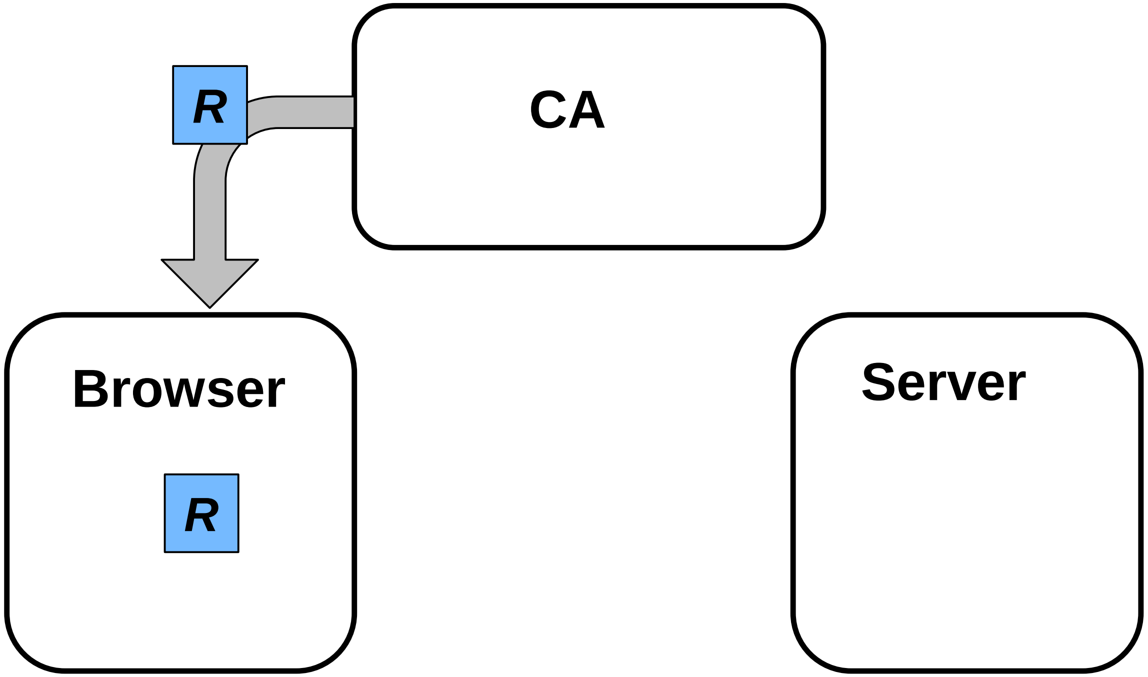


Certification of public keys



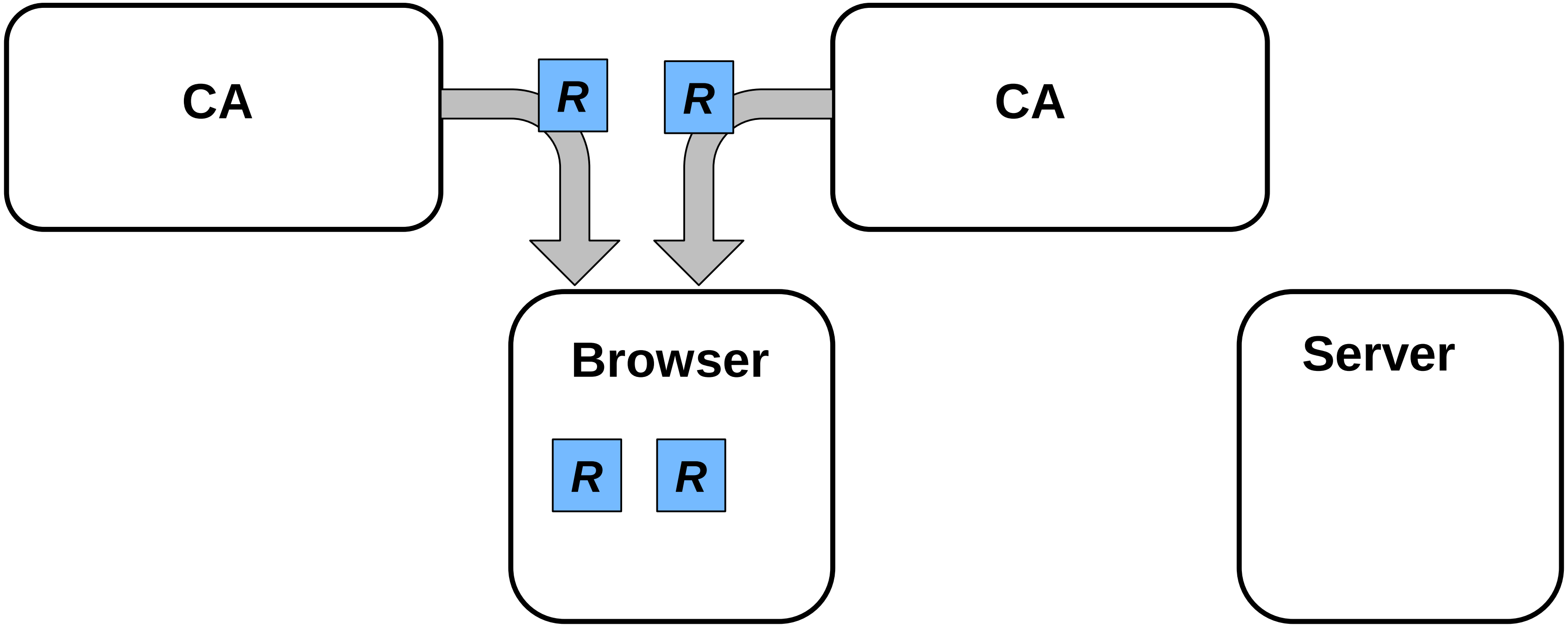
Certification of public keys

- Browsers have trust stores with root certs (of CAs)



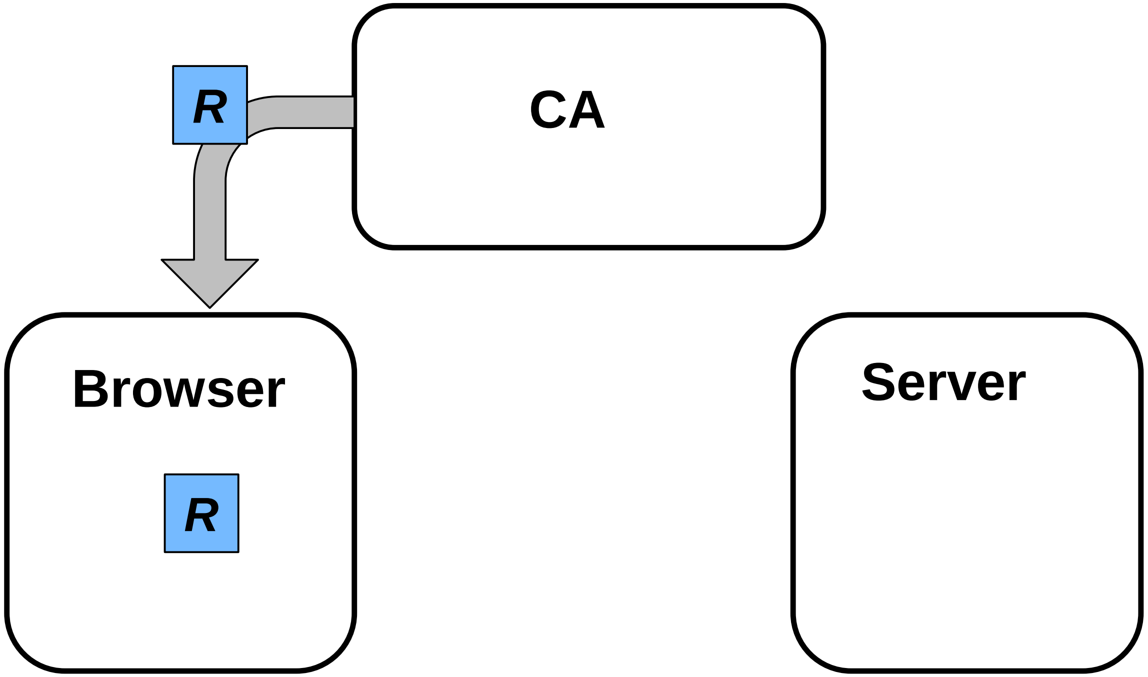
Certification of public keys

- Browsers have trust stores with root certs (of CAs)



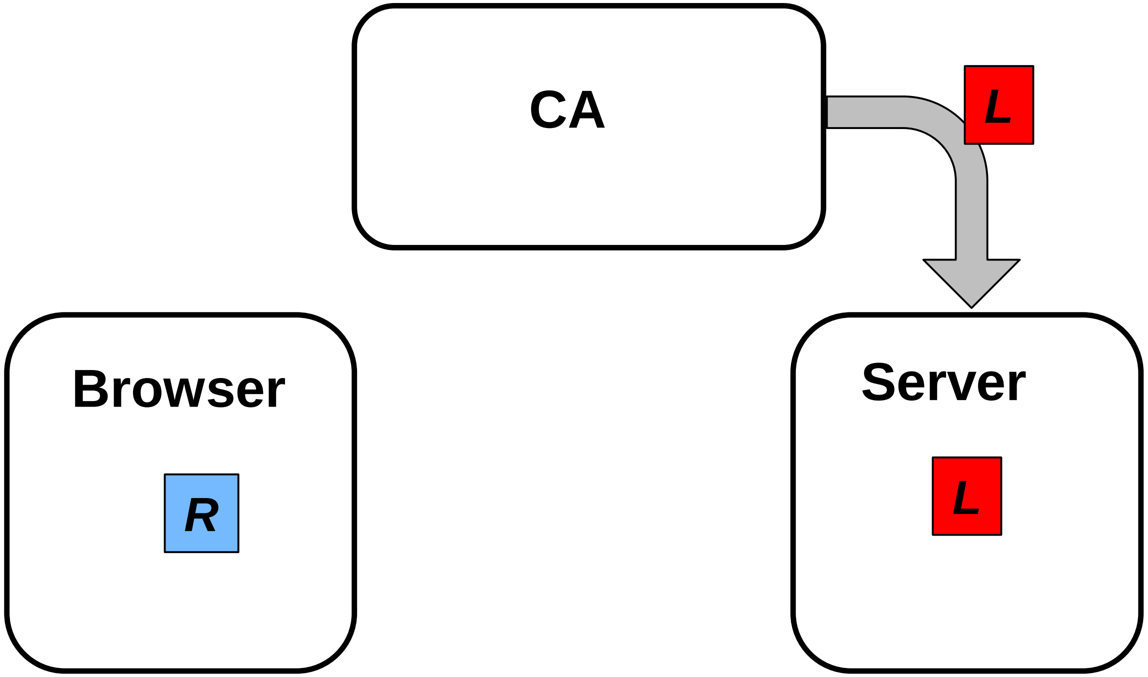
Certification of public keys

- Browsers have trust stores with root certs (of CAs)



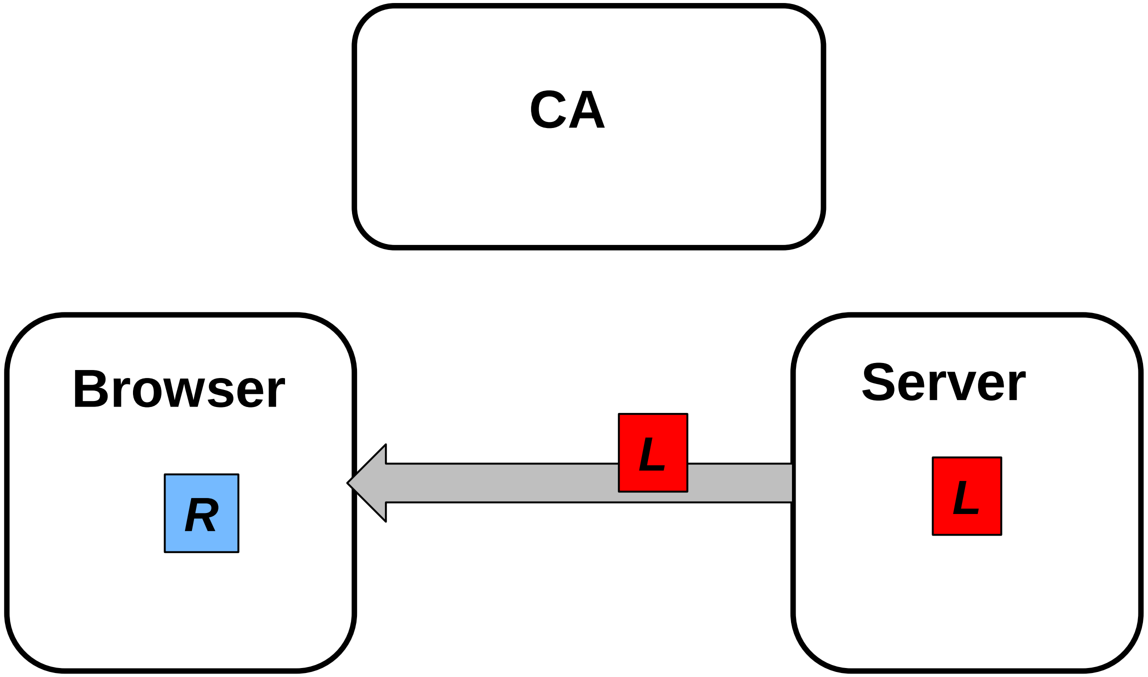
Certification of public keys

- Browsers have trust stores with root certs (of CAs)
- CAs use private key to sign certs for servers/domains
  - Certs are proof that public key belongs to server/domain



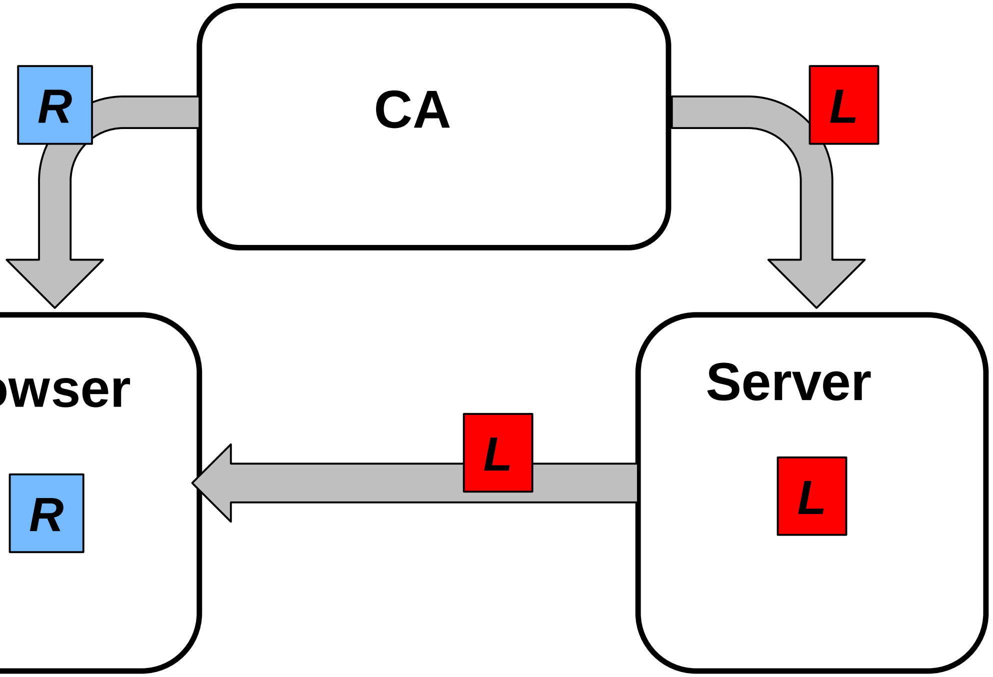
Certification of public keys

- Browsers have trust stores with root certs (of CAs)
- CAs use private key to sign certs for servers/domains
  - Certs are proof that public key belongs to server/domain
  - Signature of certs can be validated using keys in root store

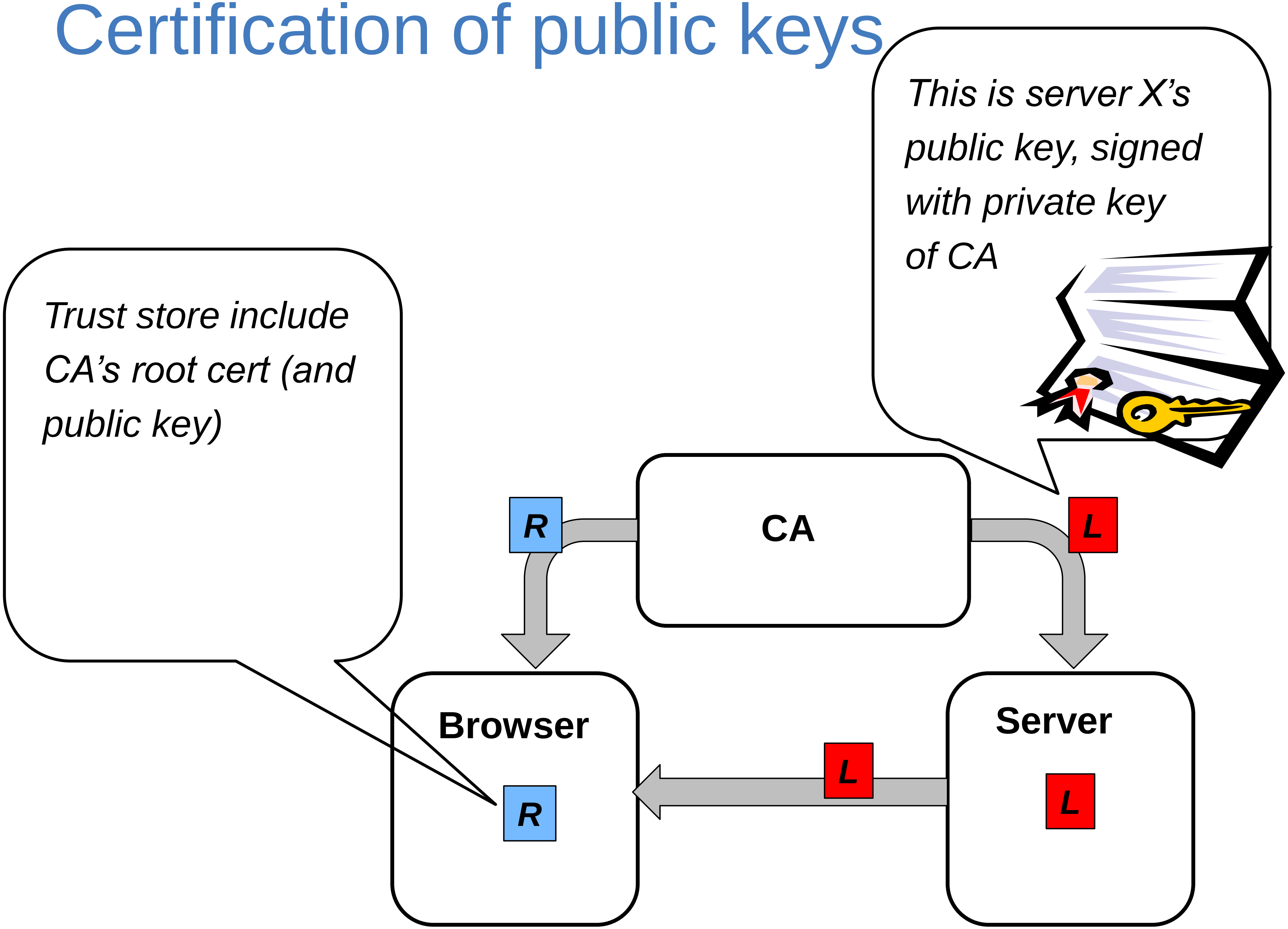


Certification of public keys

- Browsers have trust stores with root certs (of CAs)
- CAs use private key to sign certs for servers/domains
  - Certs are proof that public key belongs to server/domain
  - Signature of certs can be validated using keys in root store



Certification of public keys

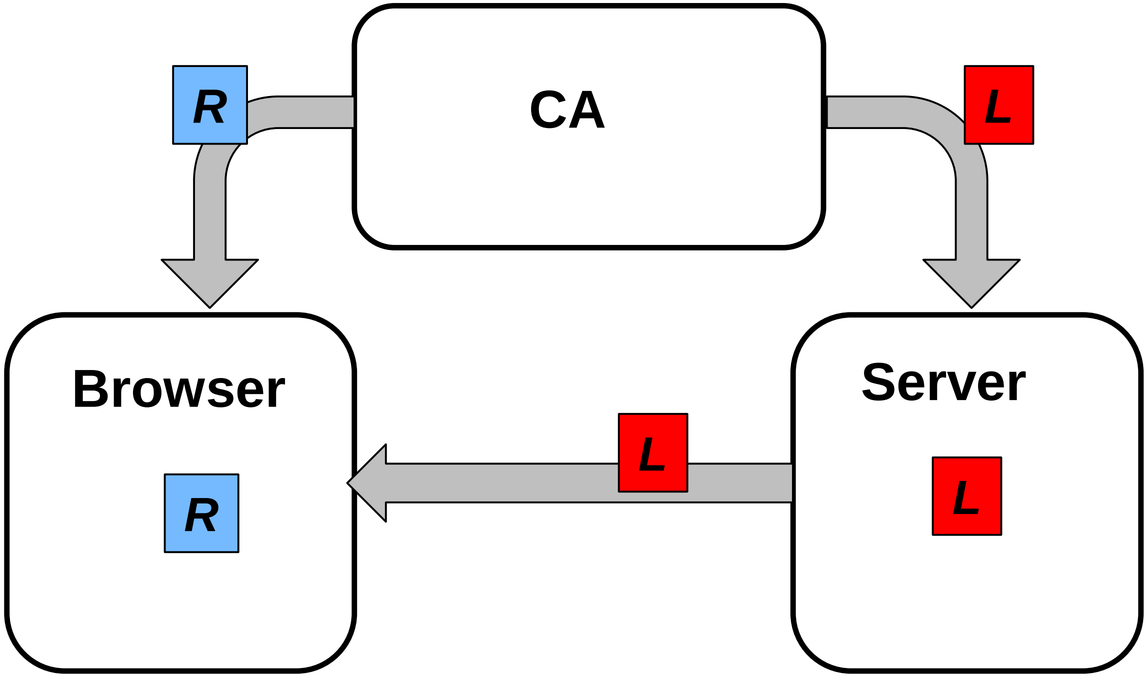


Trust store include CA's root cert (and public key)

This is server X's public key, signed with private key of CA

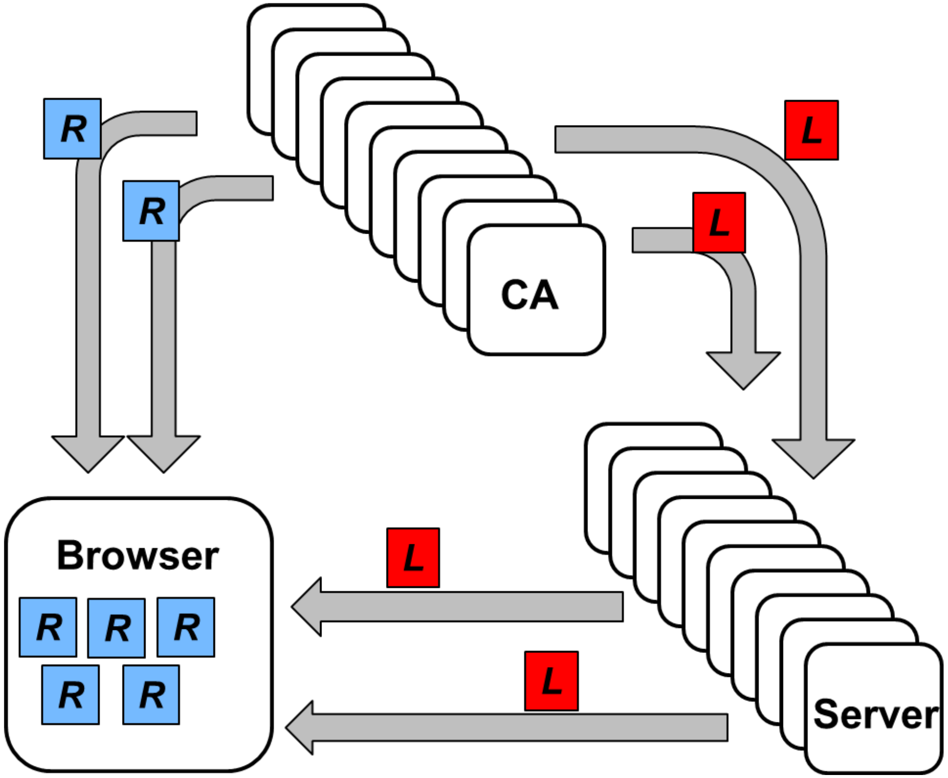
Certification of public keys

- Browsers have trust stores with root certs (of CAs)
- CAs use private key to sign certs for servers/domains
  - Certs are proof that public key belongs to server/domain
  - Signature of certs can be validated using keys in root store
-
  -
  -

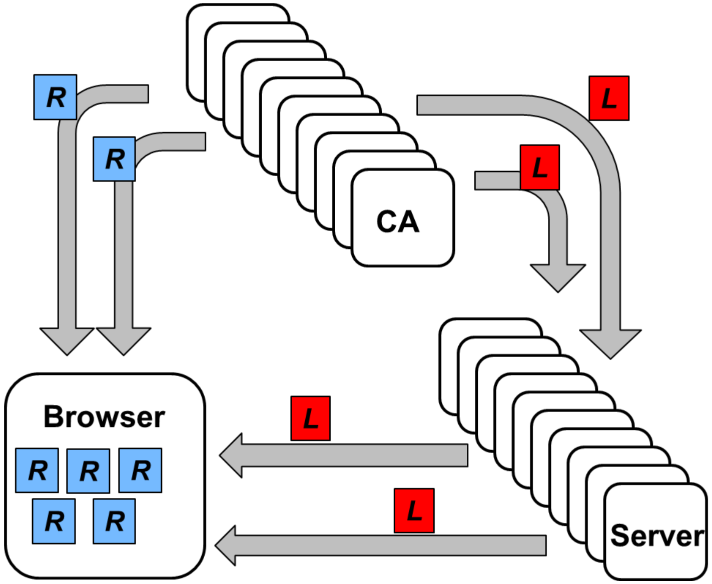


Certification of public keys

- Browsers have trust stores with root certs (of CAs)
- CAs use private key to sign certs for servers/domains
  - Certs are proof that public key belongs to server/domain
  - Signature of certs can be validated using keys in root store
- In practice, many
  - Many CAs, servers
  - Varying trust+security

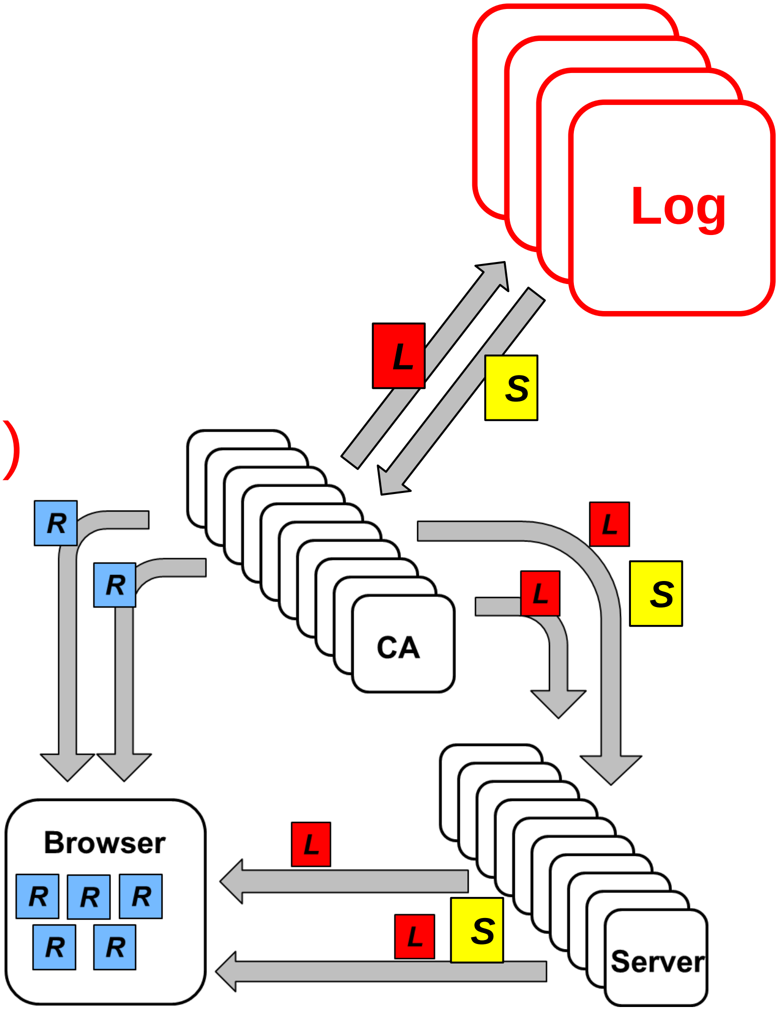


Certification Transparency (CT)



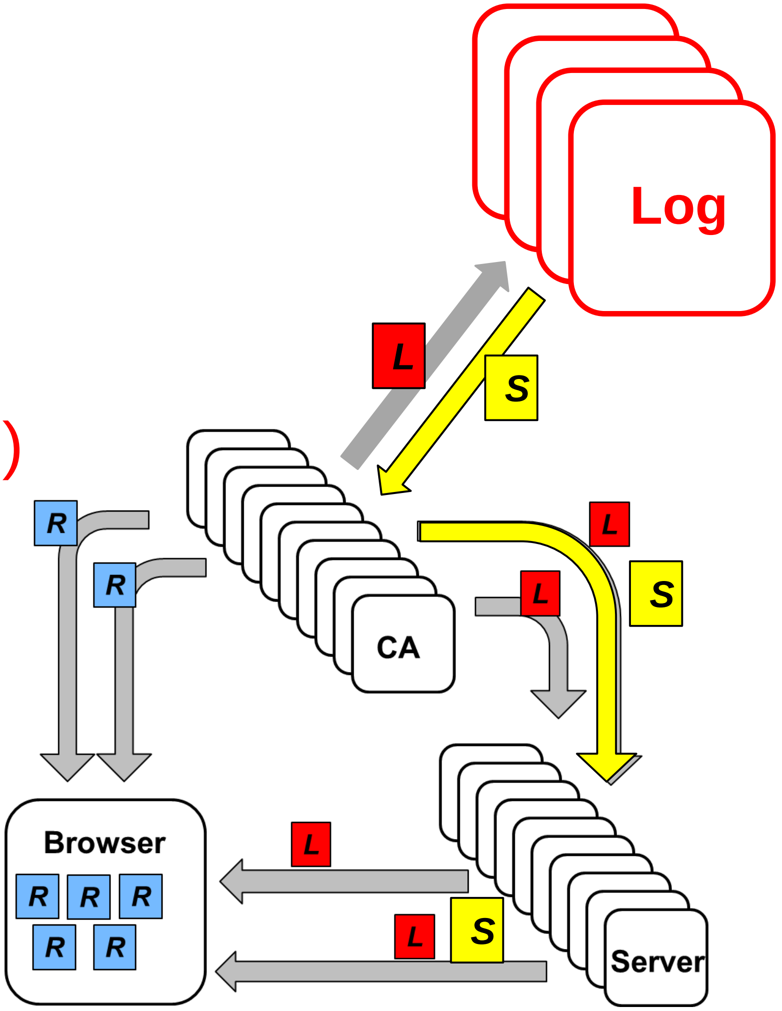
Certification Transparency (CT)

- Logs
  - Public record of certs
  - Append only (Merkle trees)
  - Create SCTs
-
  -



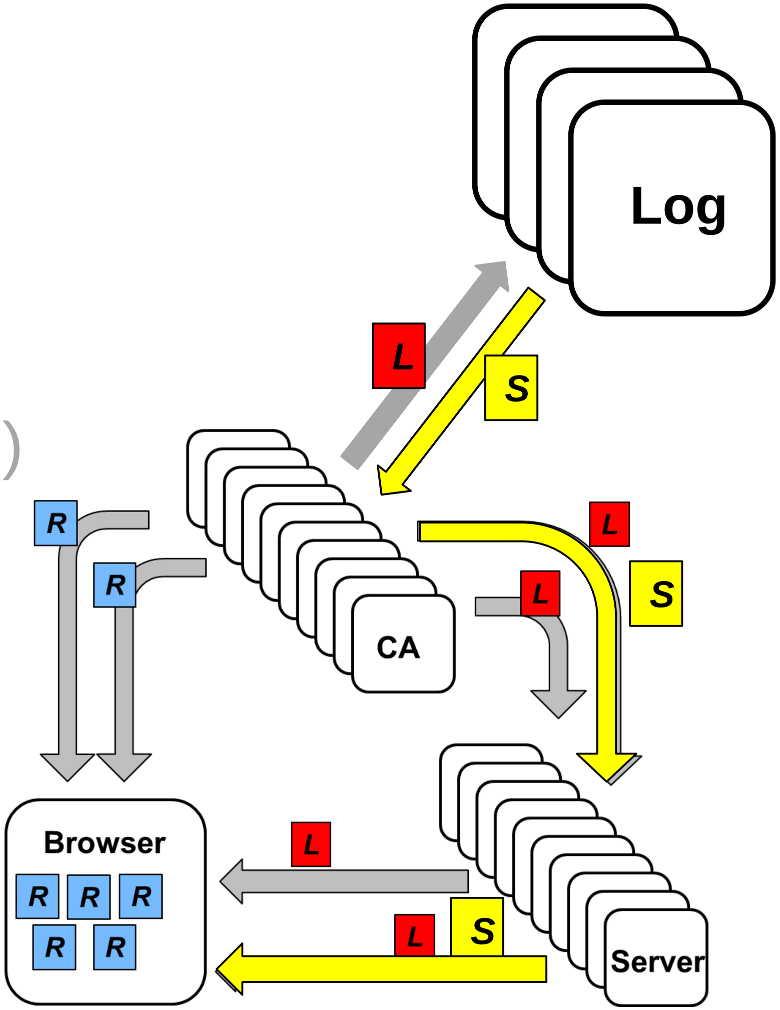
Certification Transparency (CT)

- Logs
  - Public record of certs
  - Append only (Merkle trees)
  - Create SCTs
-
  -



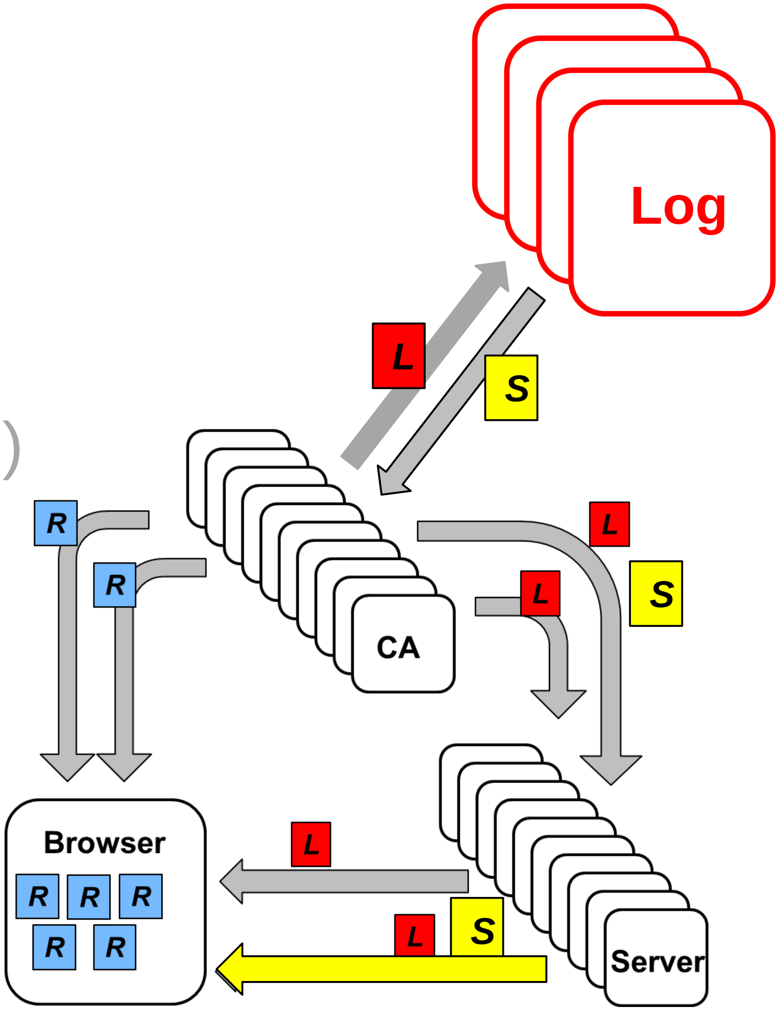
Certification Transparency (CT)

- Logs
  - Public record of certs
  - Append only (Merkle trees)
  - Create SCTs
- SCTs
  - Proof cert is logged

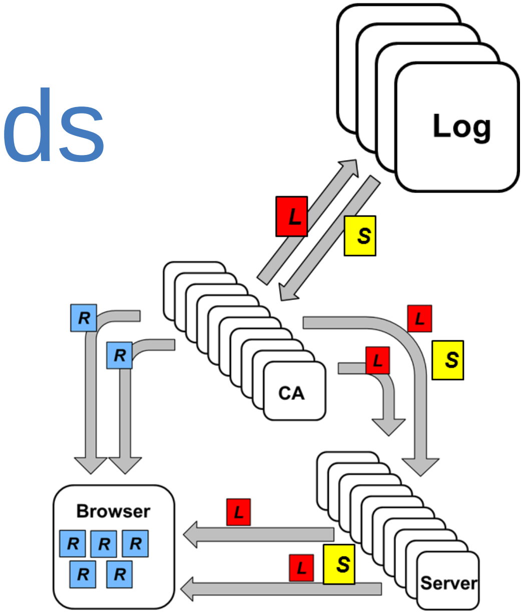


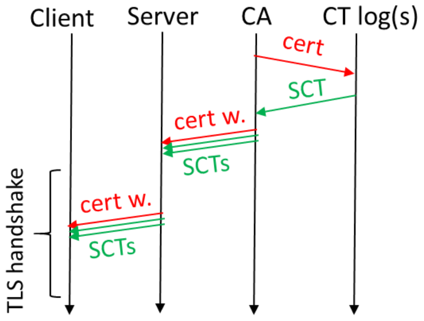
Certification Transparency (CT)

- Logs
  - Public record of certs
  - Append only (Merkle trees)
  - Create SCTs
- SCTs
  - Proof cert is logged

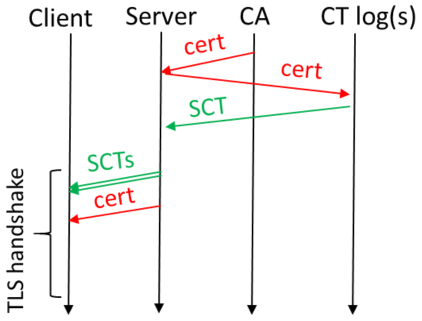


Three SCT delivery methods

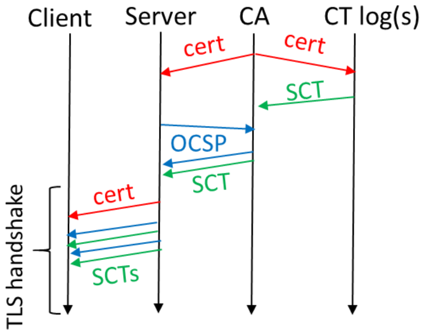




(a) X.509v3 extension

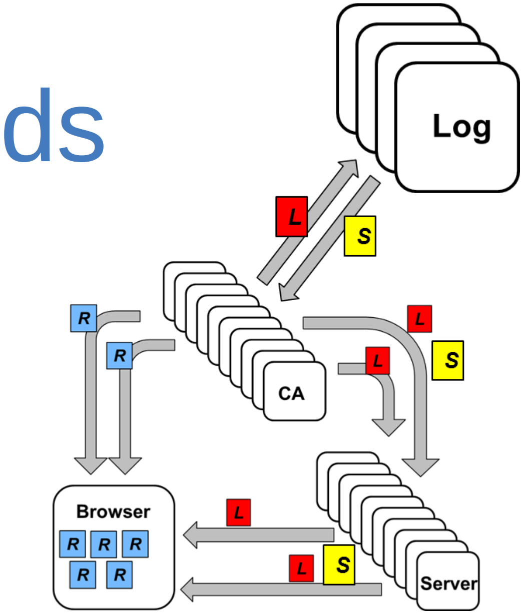


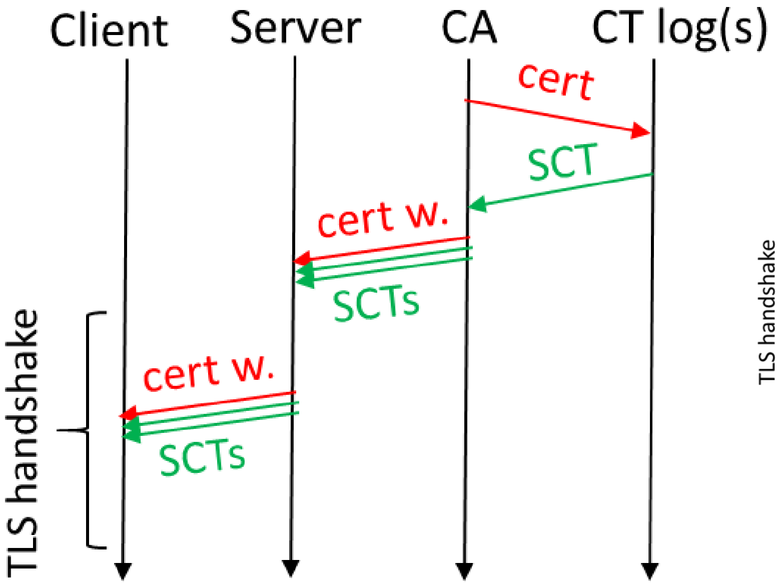
(b) TLS extension



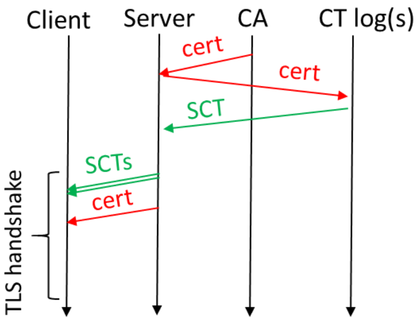
(c) OCSP stapling

Three SCT delivery methods

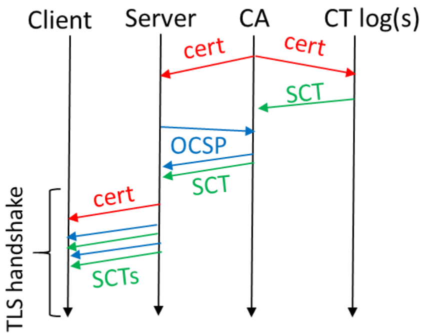




(a) X.509v3 extension

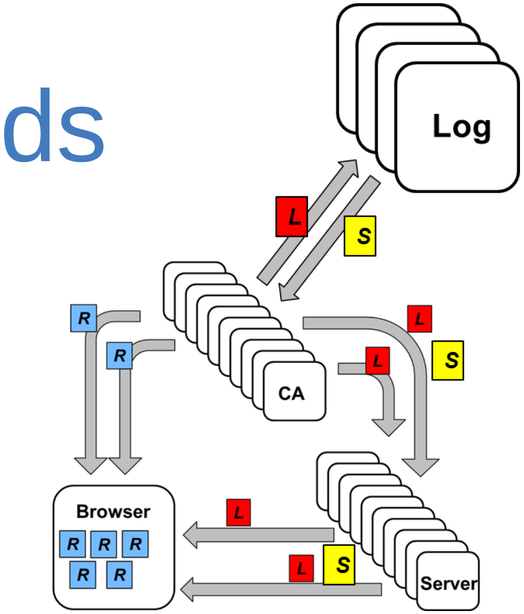


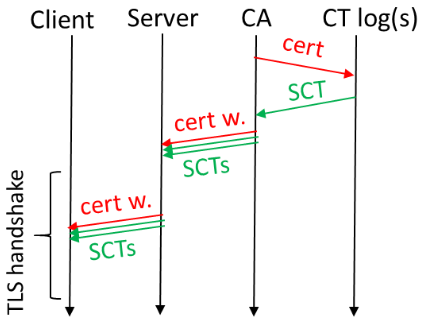
(b) TLS extension



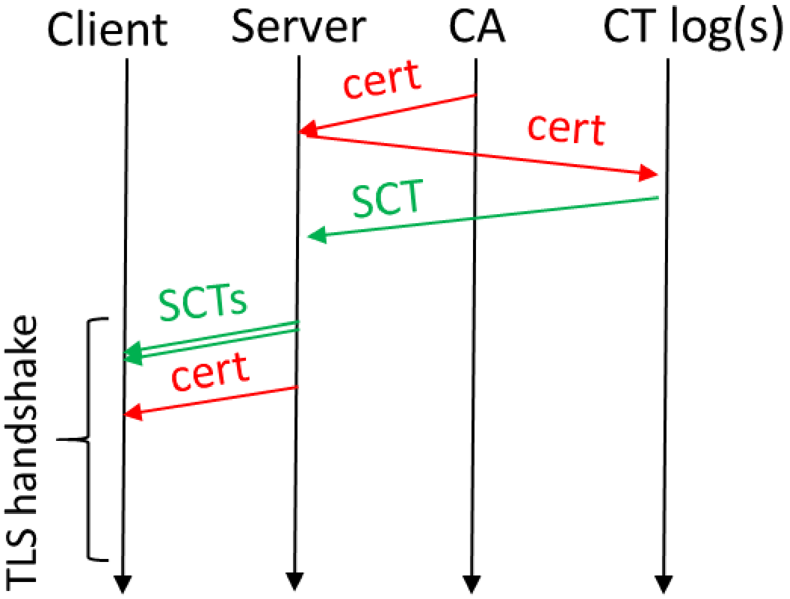
(c) OCSP stapling

Three SCT delivery methods

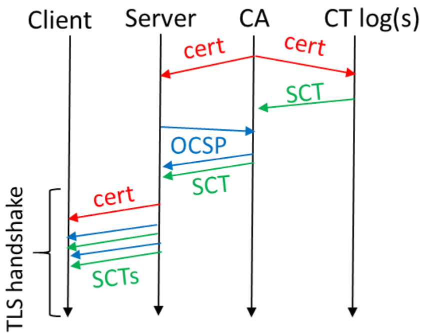




(a) X.509v3 extension

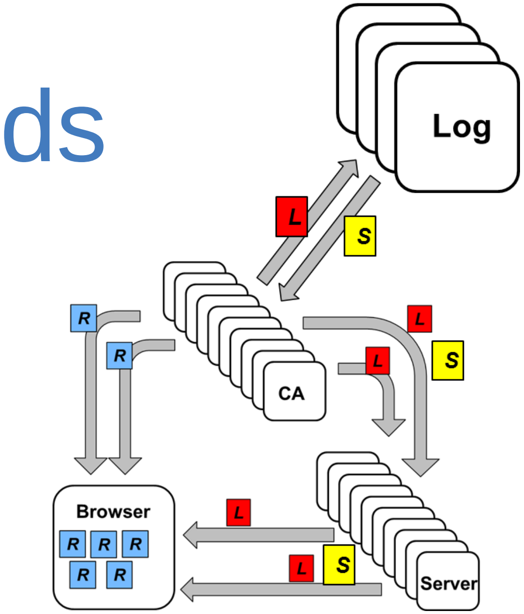


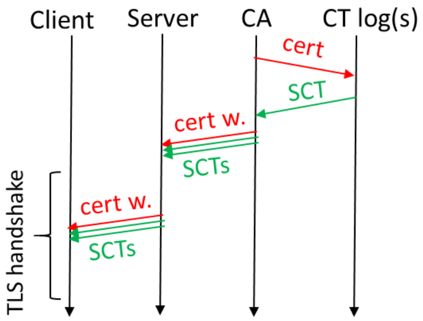
(b) TLS extension



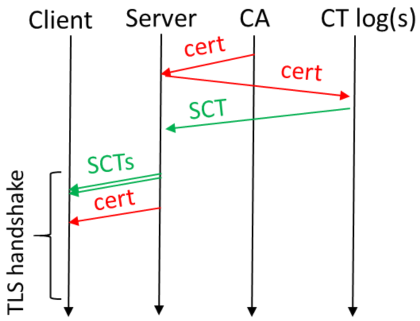
(c) OCSP stapling

Three SCT delivery methods

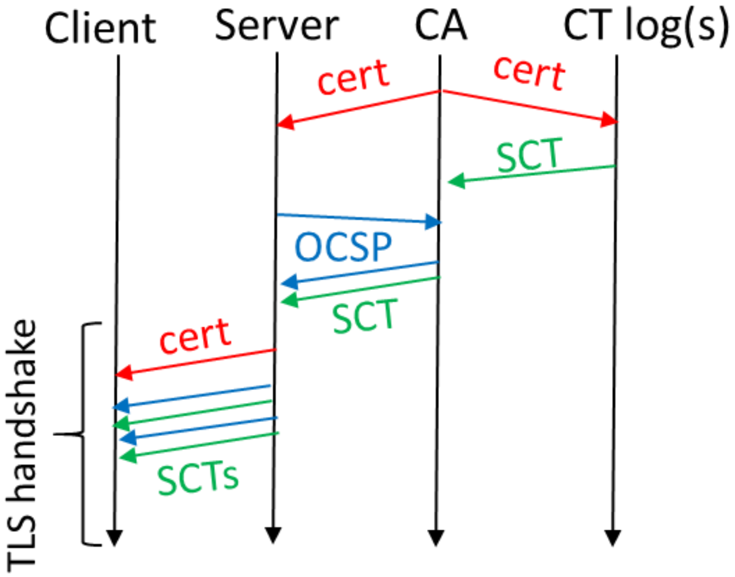




(a) X.509v3 extension



(b) TLS extension

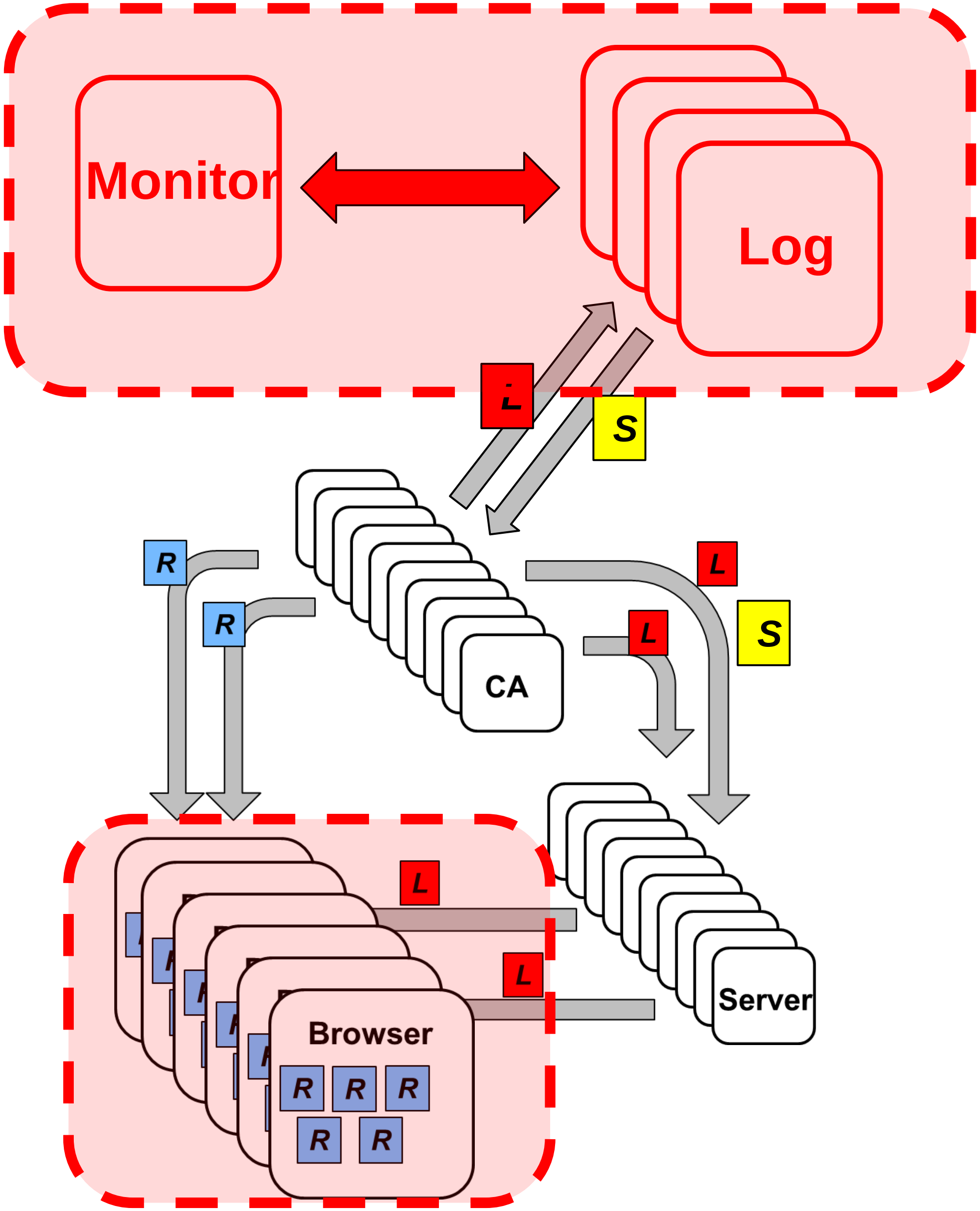


(c) OCSP stapling

Bigger picture

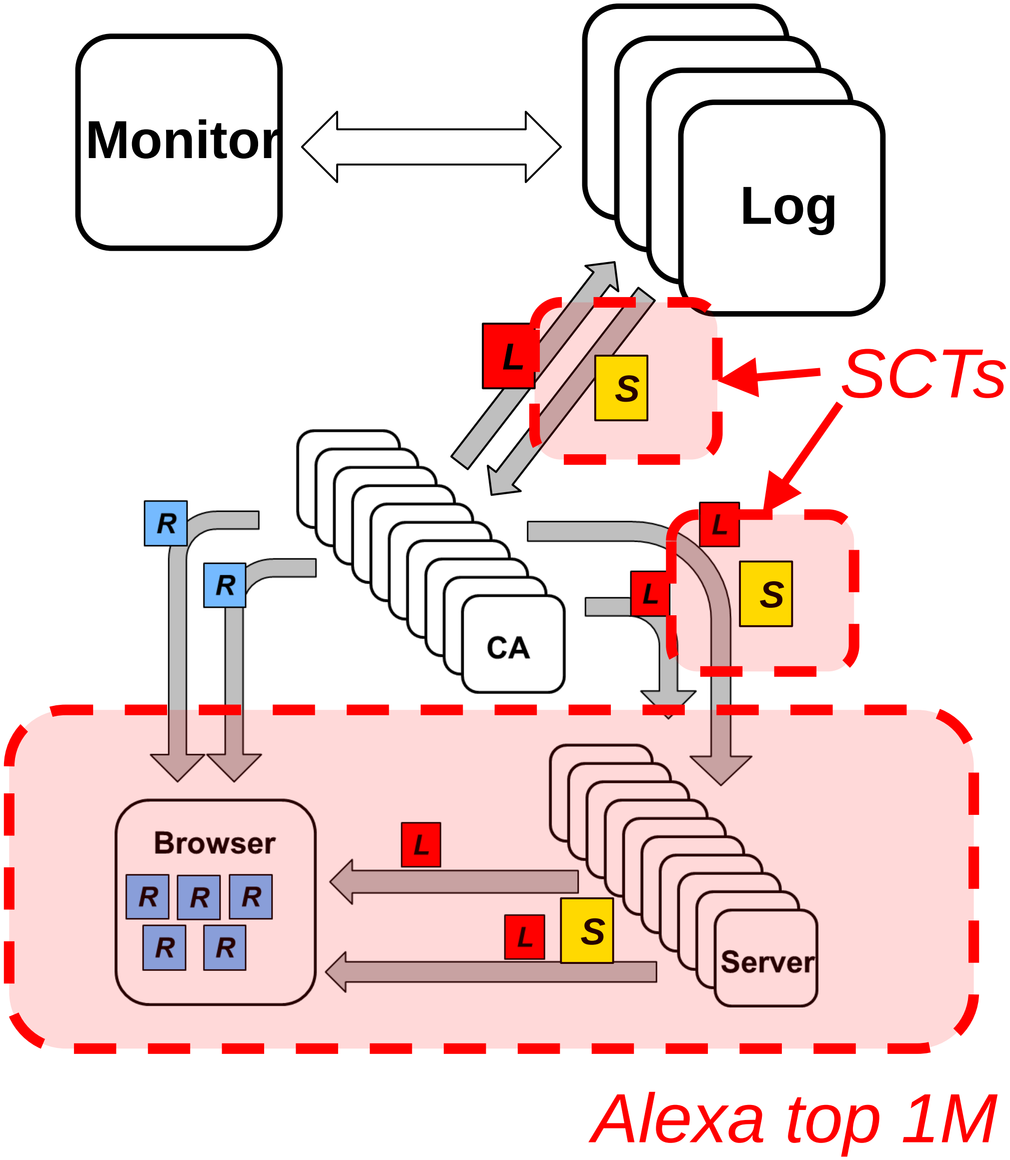
Bigger picture

- Last year's (PAM '17)
  - Monitor: All public logs
  - Campus measurements: All HTTPS sessions for a week
- This paper (PAM '18)
  - Server-side SCT usage
  - Client-side performance
- Other related work
  - Gasser et al. (PAM '18), Amann et al. (IMC '17), VanderSloot et al.(IMC '16)



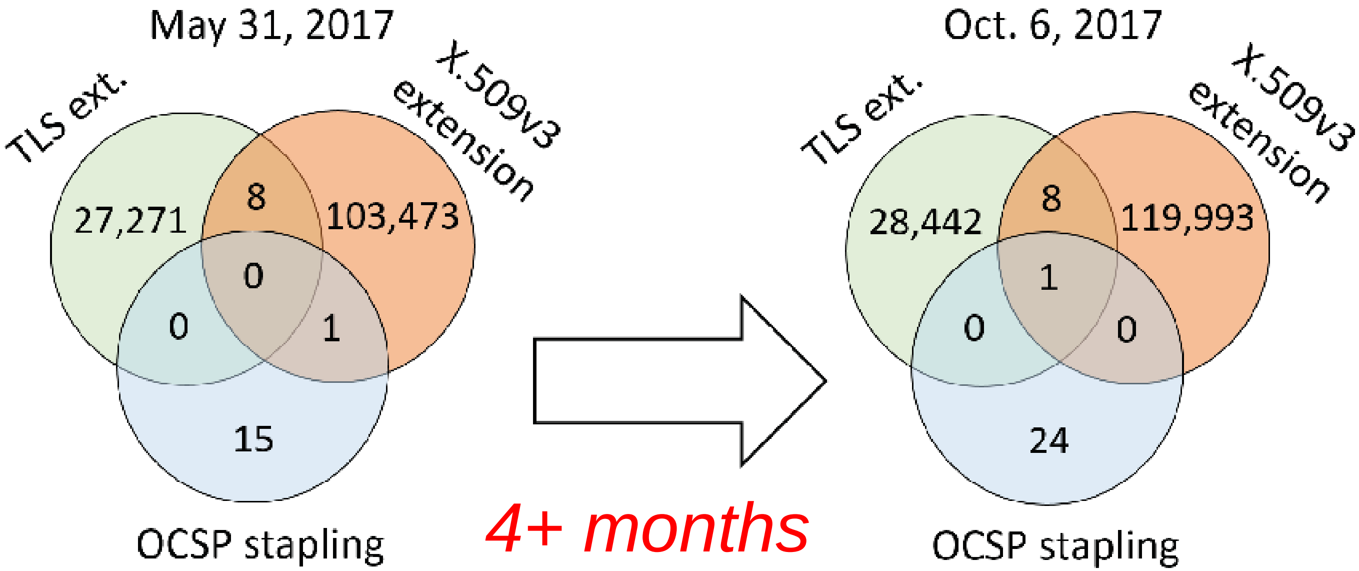
Bigger picture

- Last year's (PAM '17)
  - Monitor: All public logs
  - Campus measurements: All HTTPS sessions for a week
- This paper (PAM '18)
  - Server-side SCT usage
  - Client-side performance
- Other related work
  - Gasser et al. (PAM '18), Amann et al. (IMC '17), VanderSloot et al.(IMC '16)



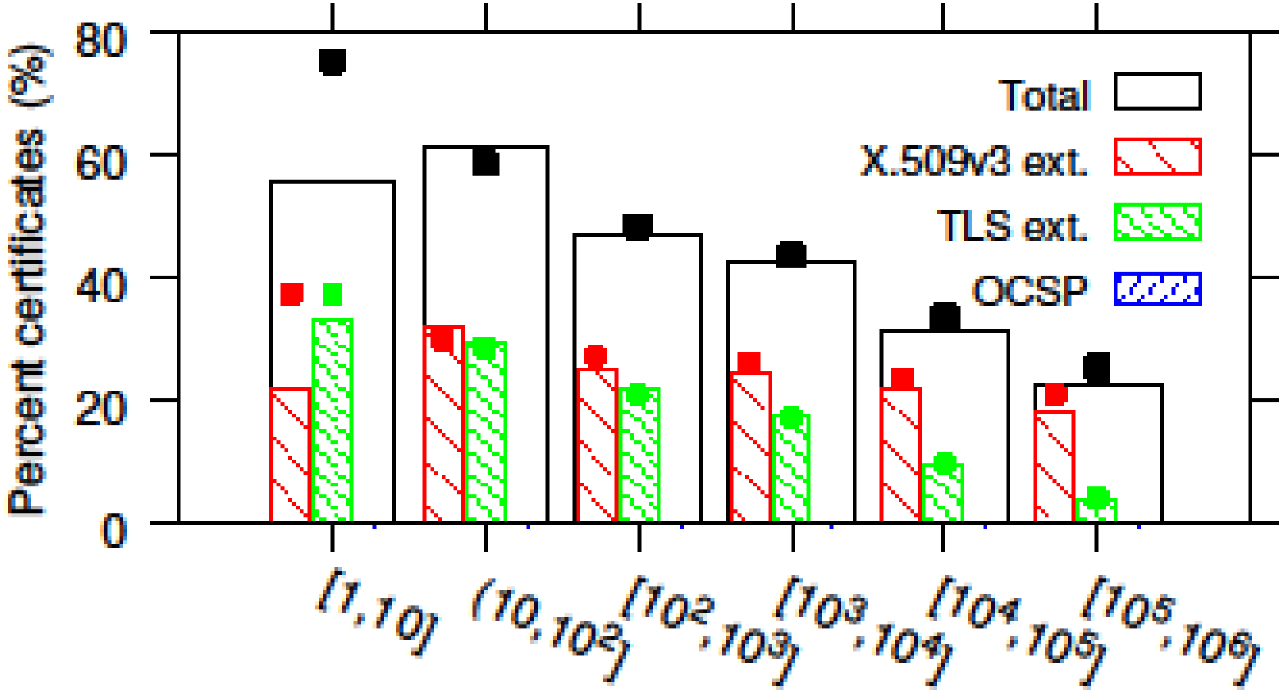
Results

Dataset overview

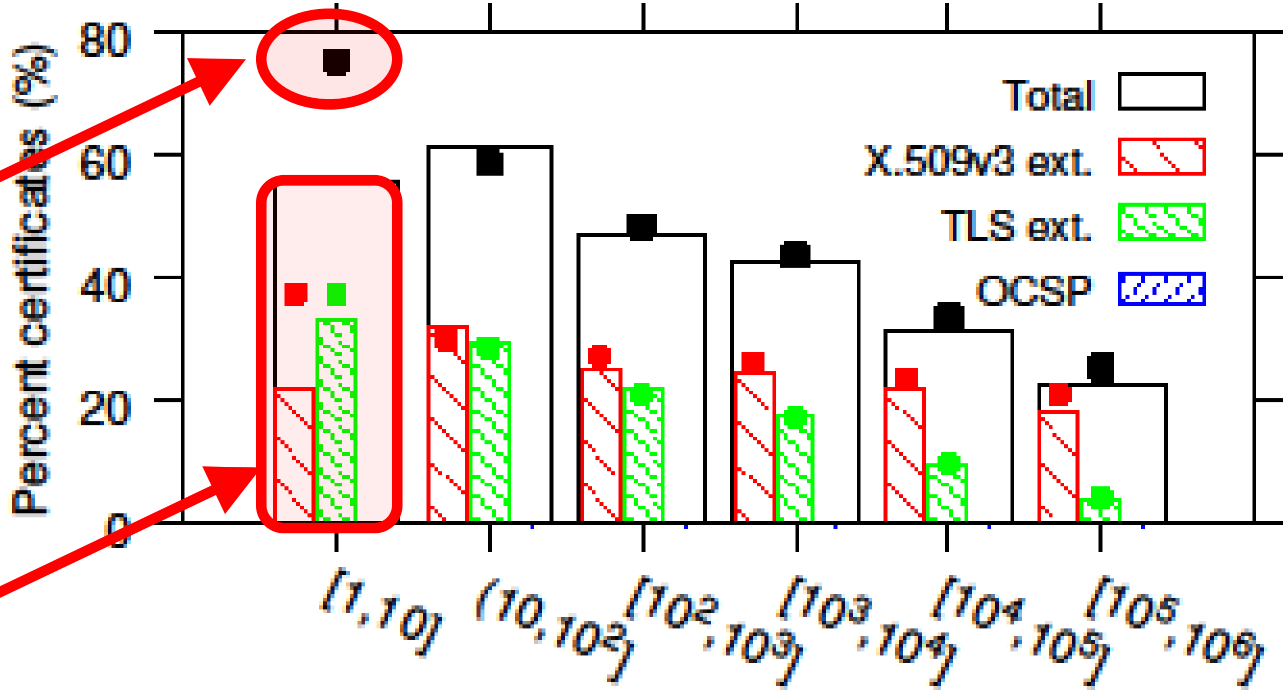


- Method
    - Alexa top-1M
    - Two snapshots: May 31 (2017) and Oct. 6 (2017)
    - Single machine, 600 parallel threads (approx. 4 hours)
- SCT usage increase across all methods
- X.509v3 dominates (easiest method for server domains)

Popularity-based breakdown



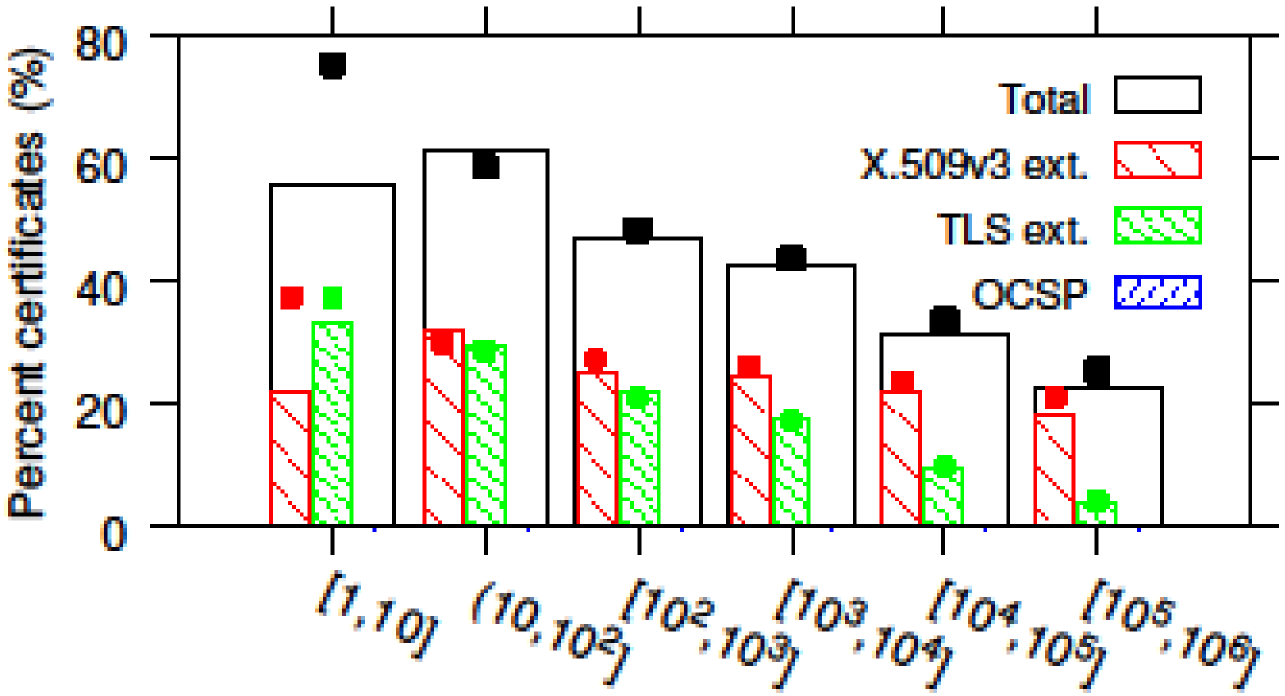
Popularity-based breakdown



Oct 2017

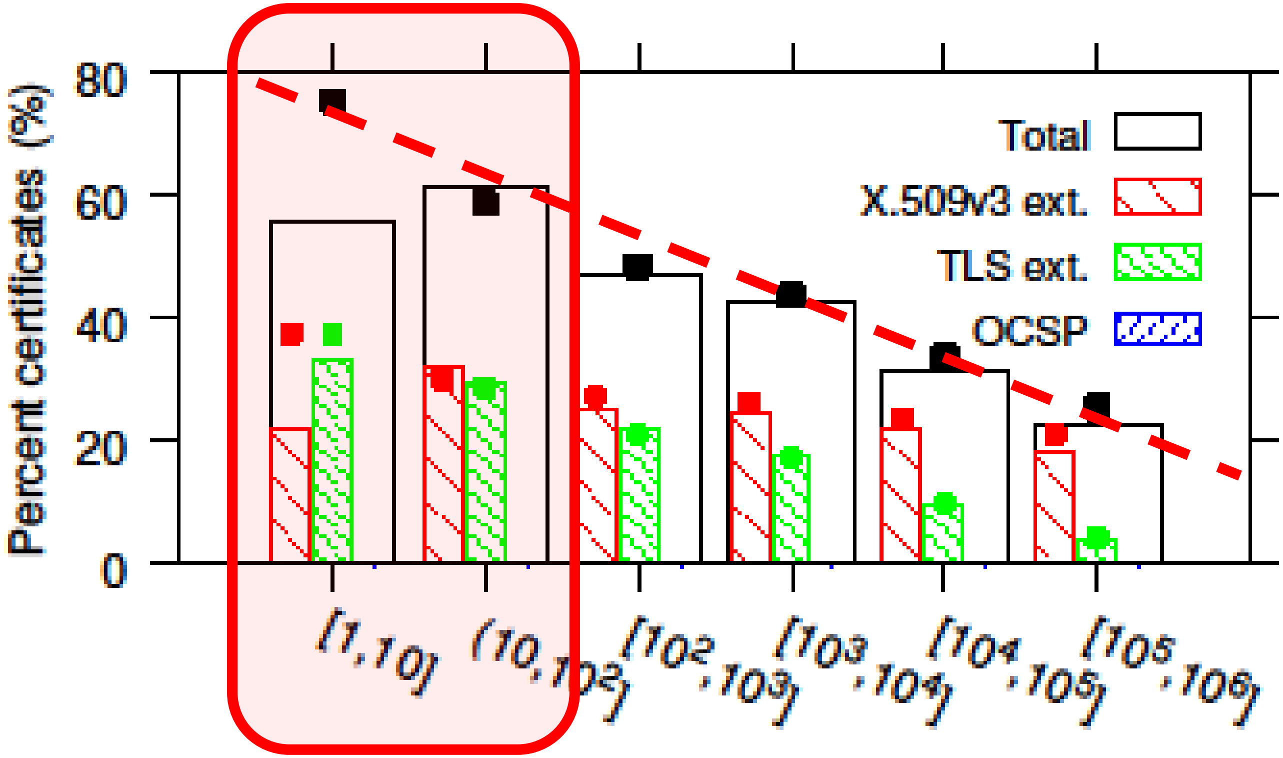
May 2017

Popularity-based breakdown



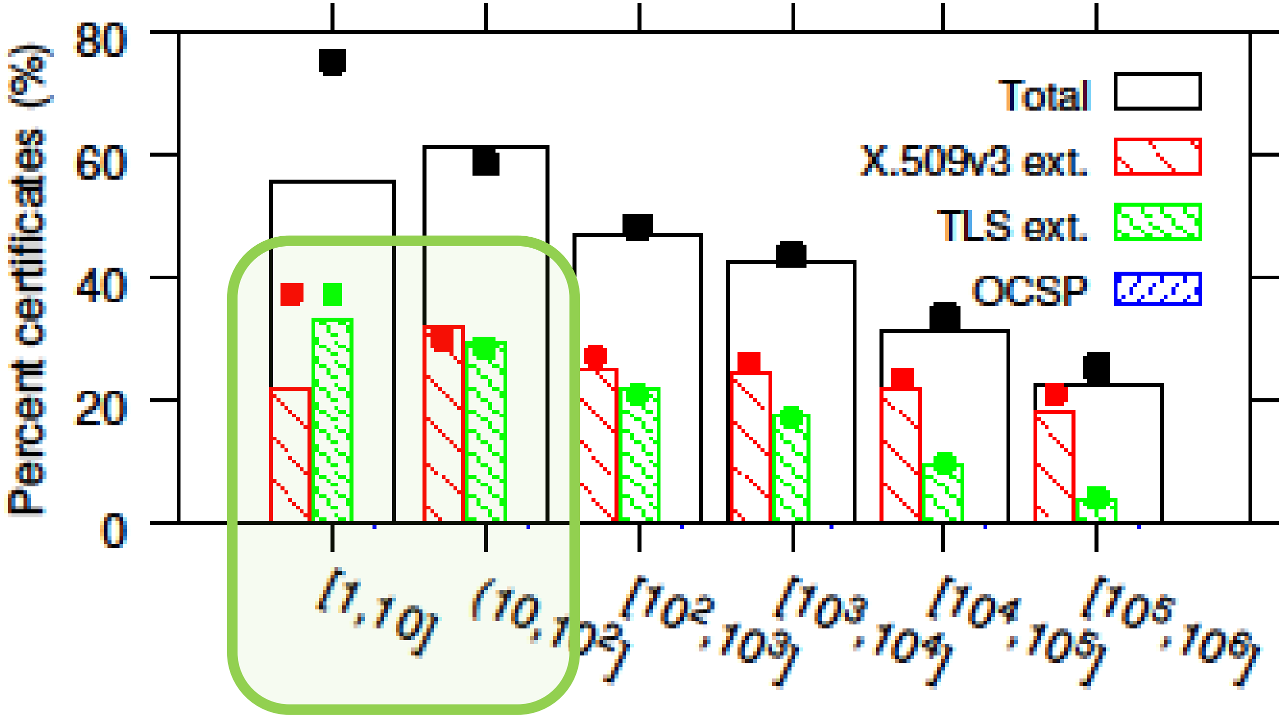
- SCT usage highest among most popular domains
- TLS usage highest among most popular domains

Popularity-based breakdown



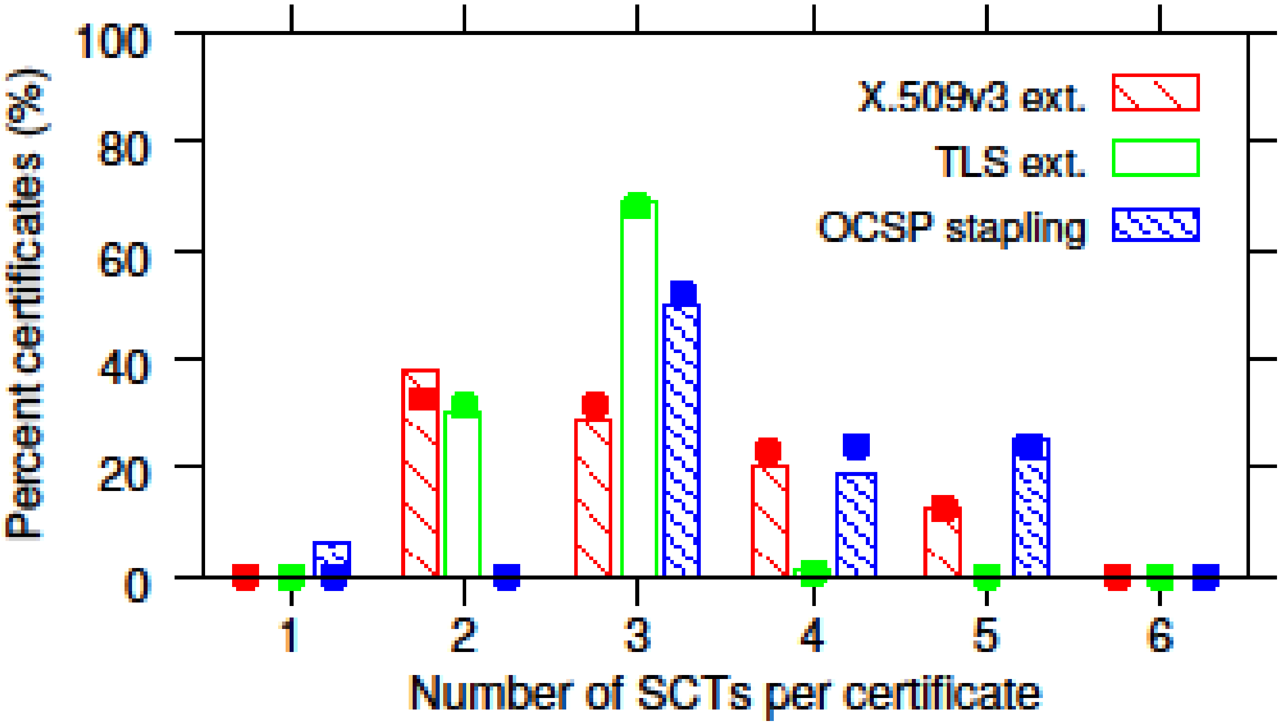
- SCT usage highest among most popular domains
- TLS usage highest among most popular domains

Popularity-based breakdown



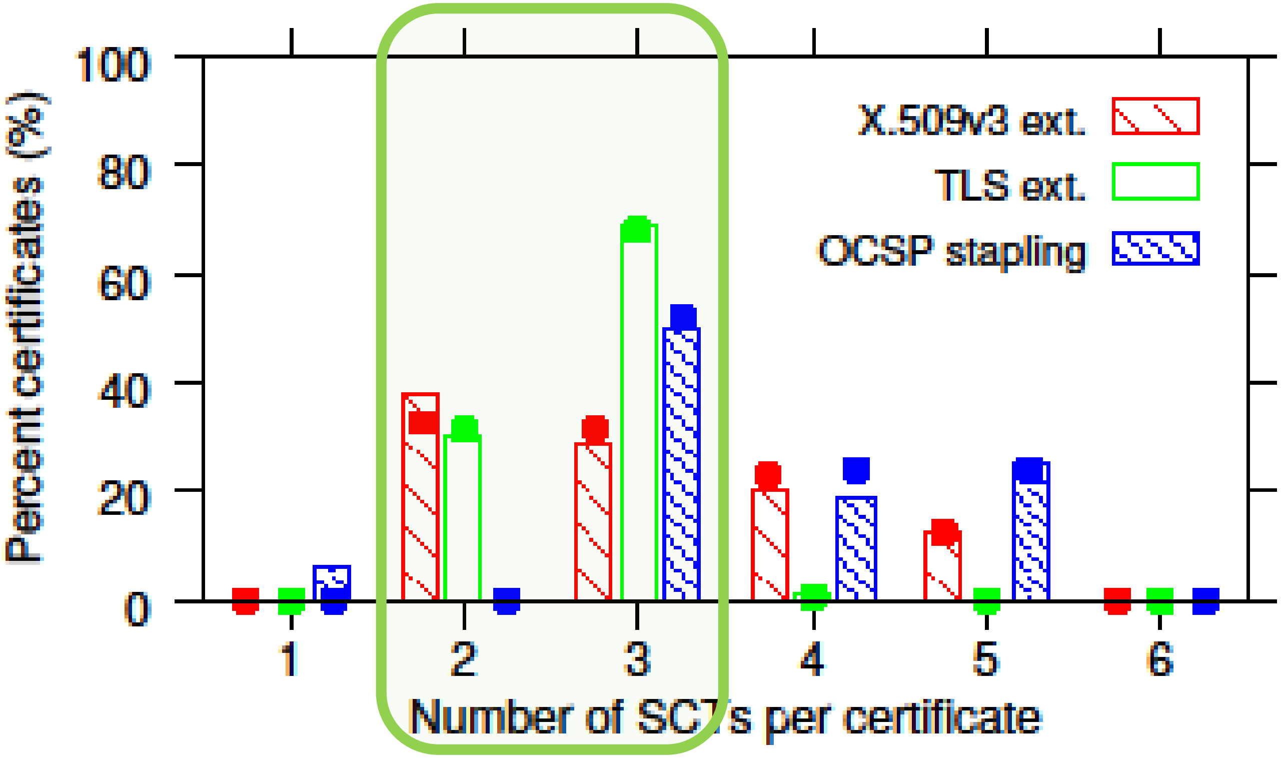
- SCT usage highest among most popular domains
- TLS usage highest among most popular domains

Number of SCTs per certificate



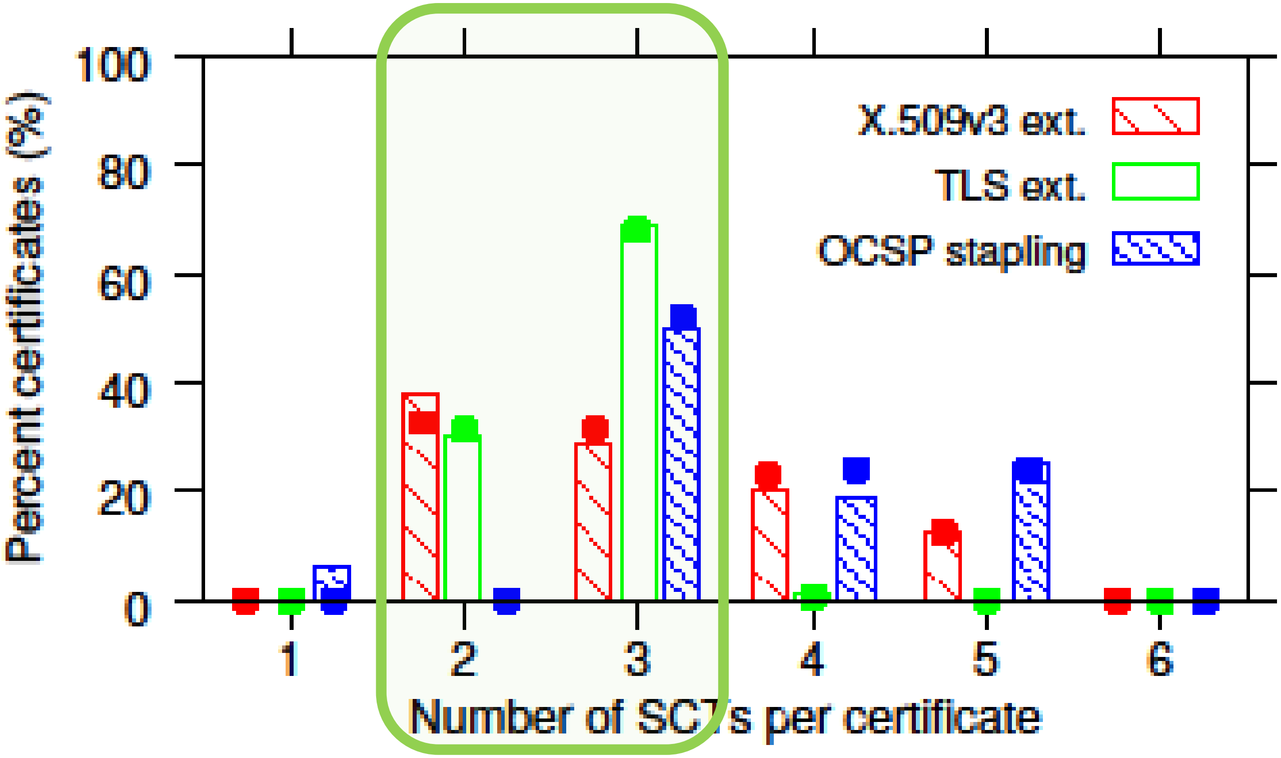
- Many SCTs per certificate
  - TLS typically has 2 or 3
  - The other have higher diversity

Number of SCTs per certificate



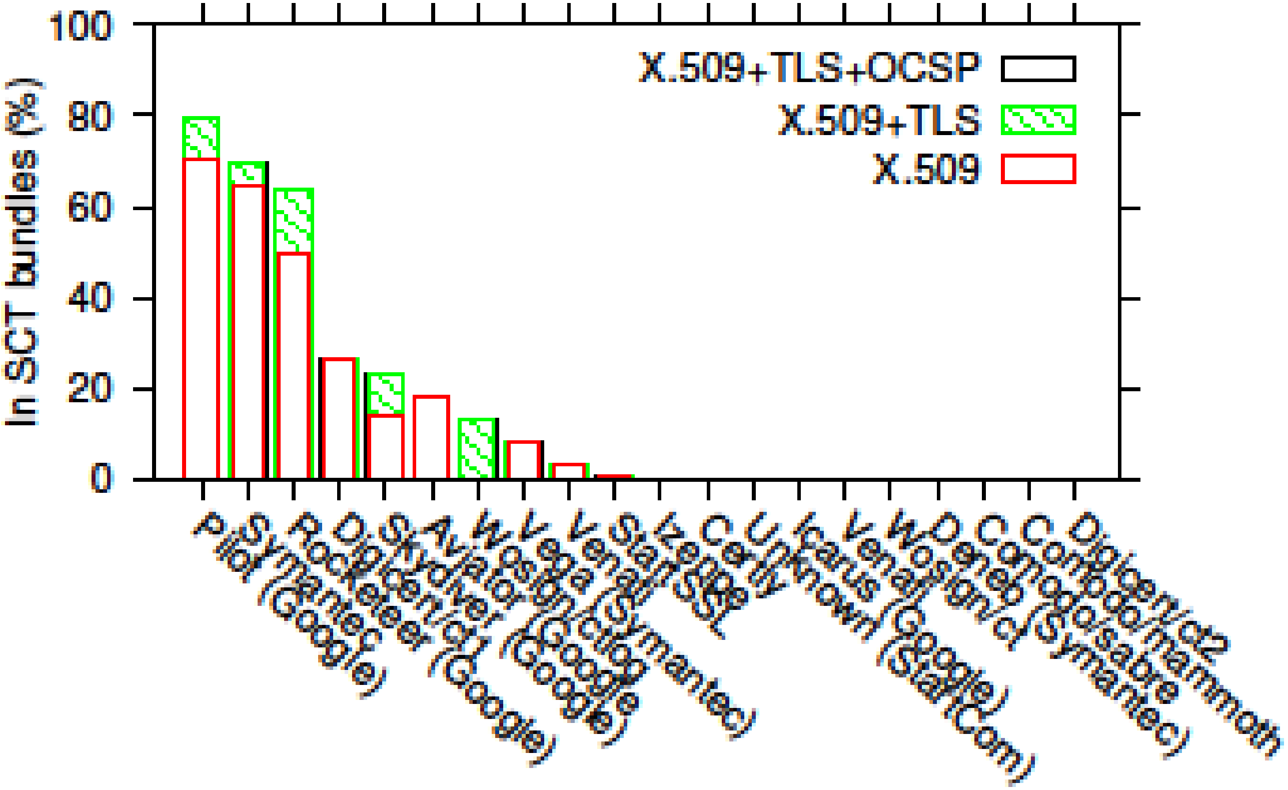
- Many SCTs per certificate
  - TLS typically has 2 or 3
  - The other have higher diversity

Number of SCTs per certificate



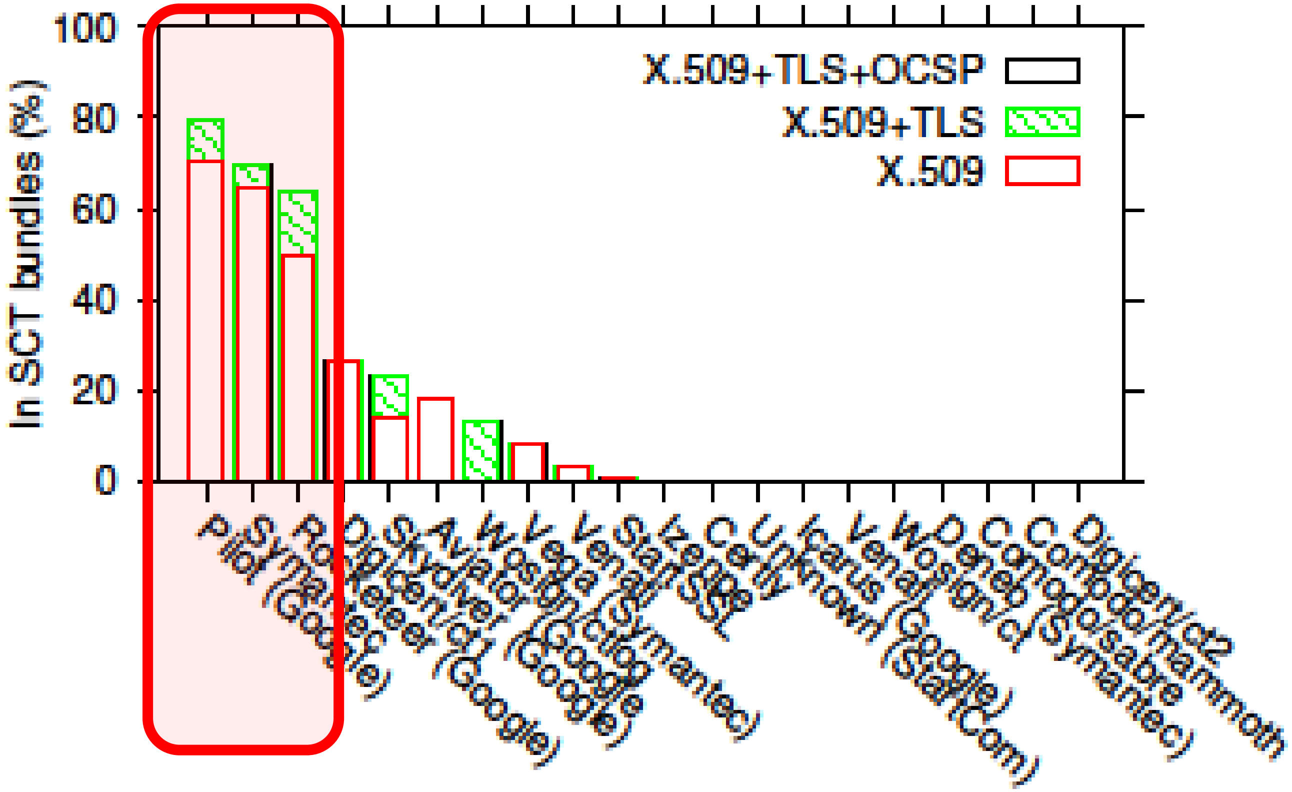
- Many SCTs per certificate
  - TLS typically has 2 or 3
  - The other have higher diversity

Log usage



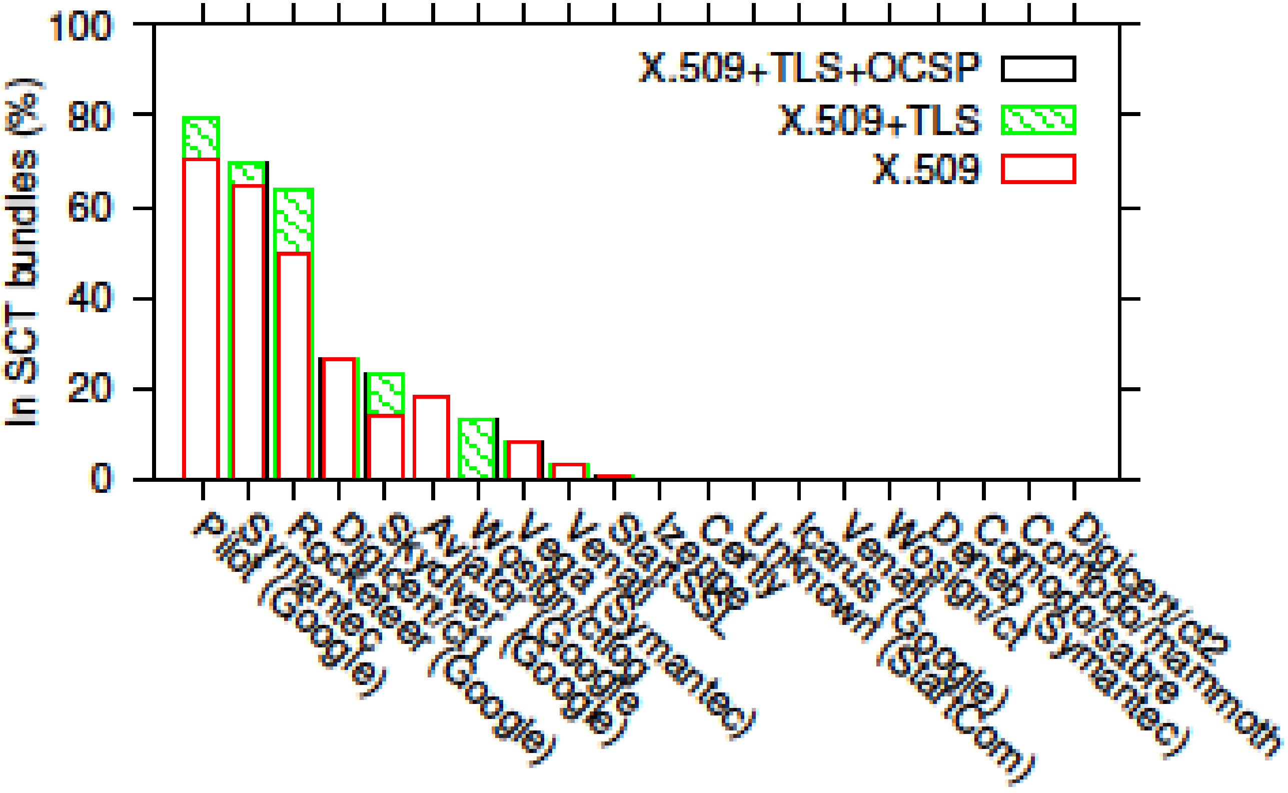
- A few dominating logs
- Big differences in TLS frequency among CA logs
  - Wosign almost only TLS
- Aviator (frozen on Nov 29, 2016) almost only X.509v3
  - Again, TLS is increasing (but way behind)

Log usage



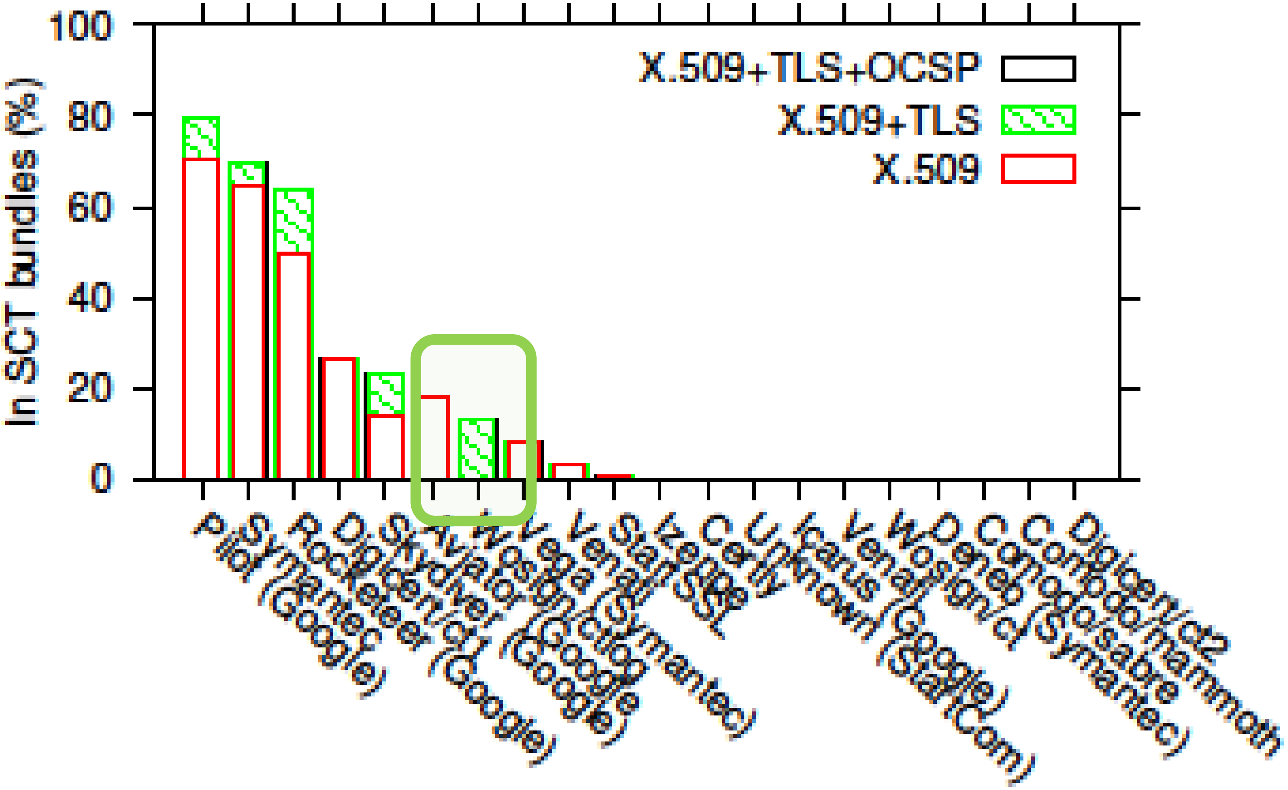
- A few dominating logs
- Big differences in TLS frequency among CA logs
  - Wosign almost only TLS
- Aviator (frozen on Nov 29, 2016) almost only X.509v3
  - Again, TLS is increasing (but way behind)

Log usage



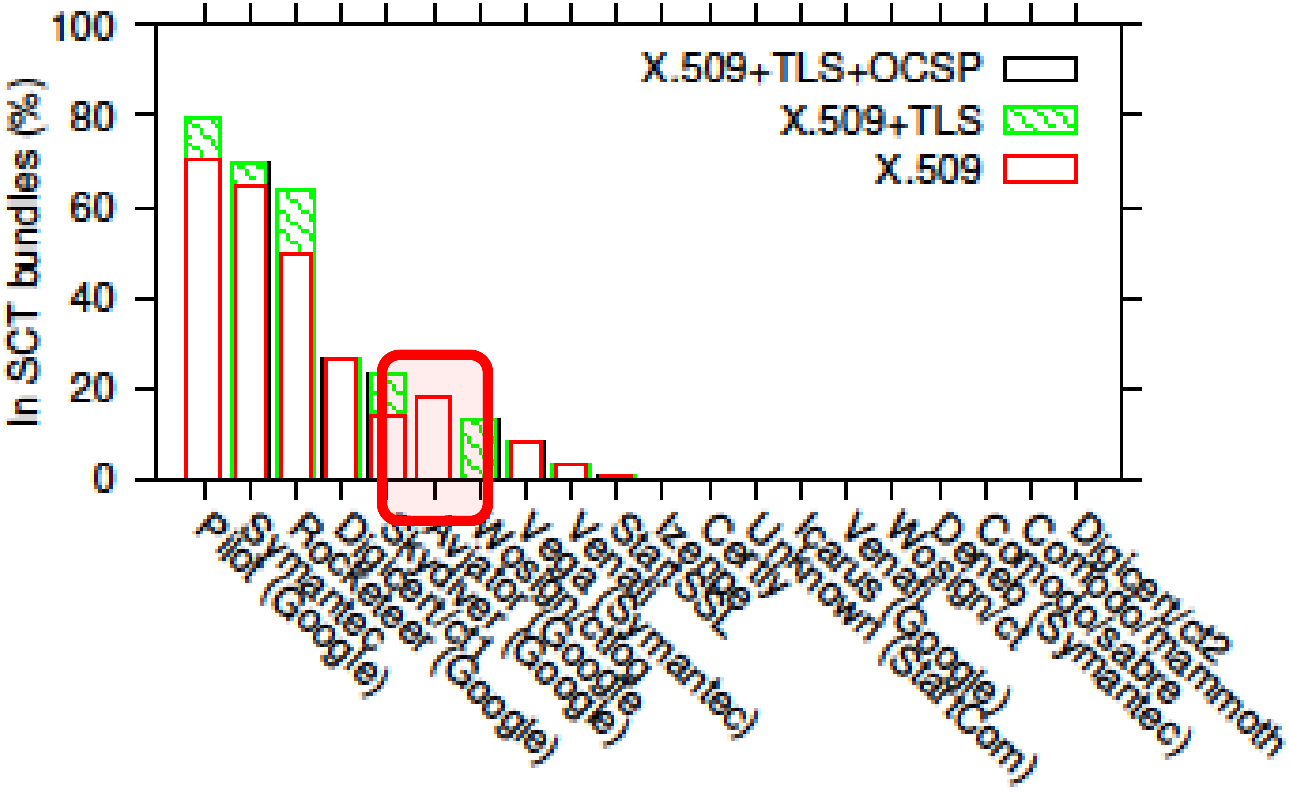
- A few dominating logs
- Big differences in TLS frequency among CA logs
  - Wosign almost only TLS
- Aviator (frozen on Nov 29, 2016) almost only X.509v3
  - Again, TLS is increasing (but way behind ...)

Log usage



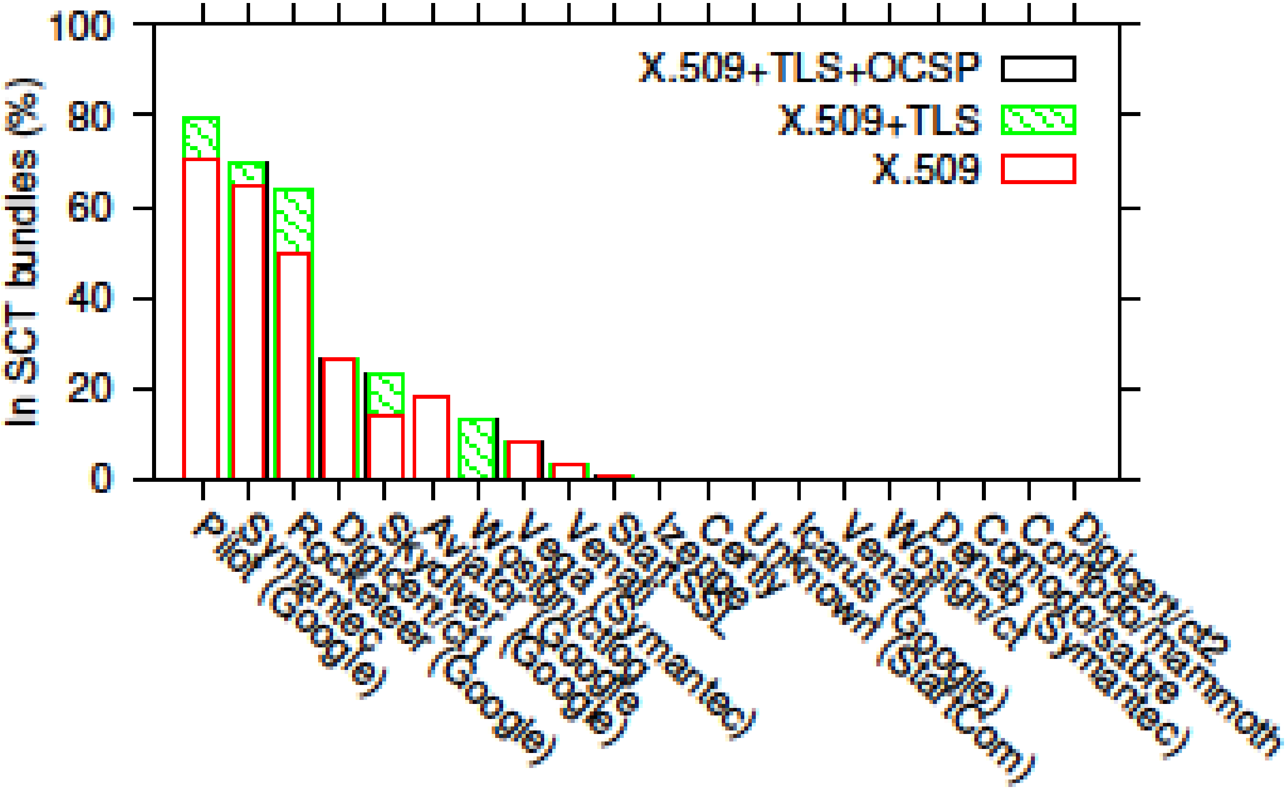
- A few dominating logs
- Big differences in TLS frequency among CA logs
  - Wosign almost only TLS
- Aviator (frozen on Nov 29, 2016) almost only X.509v3
  - Again, TLS is increasing (but way behind ...)

Log usage



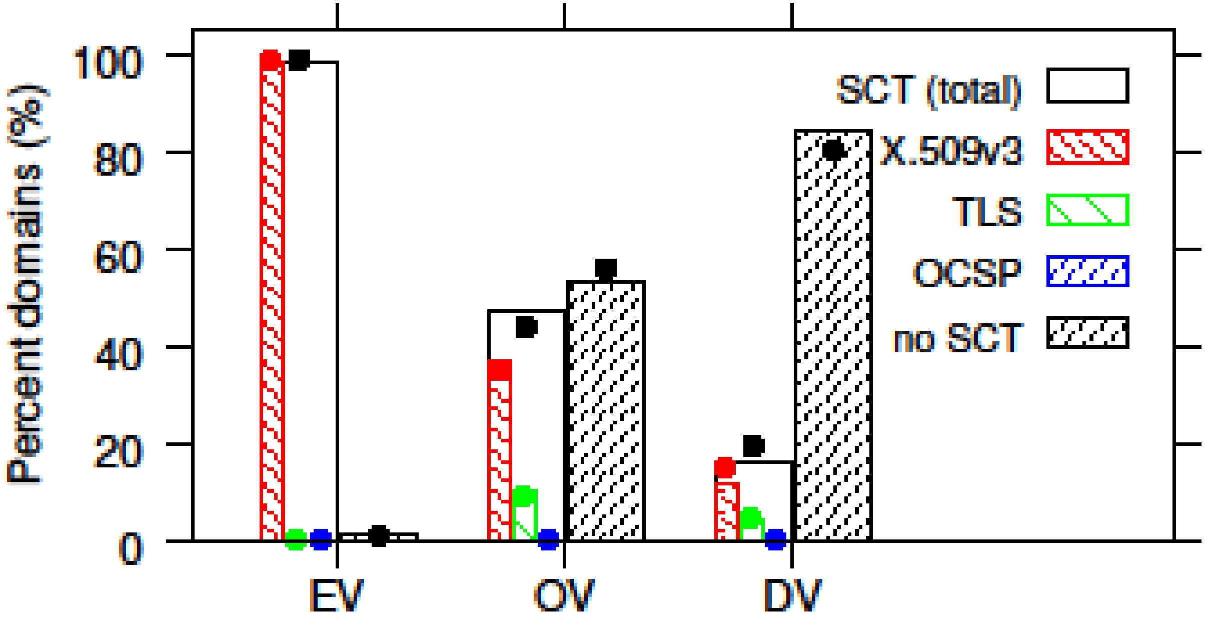
- A few dominating logs
- Big differences in TLS frequency among CA logs
  - Wosign almost only TLS
- Aviator (frozen on Nov 29, 2016) almost only X.509v3
  - Again, TLS is increasing (but way behind ...)

Log usage



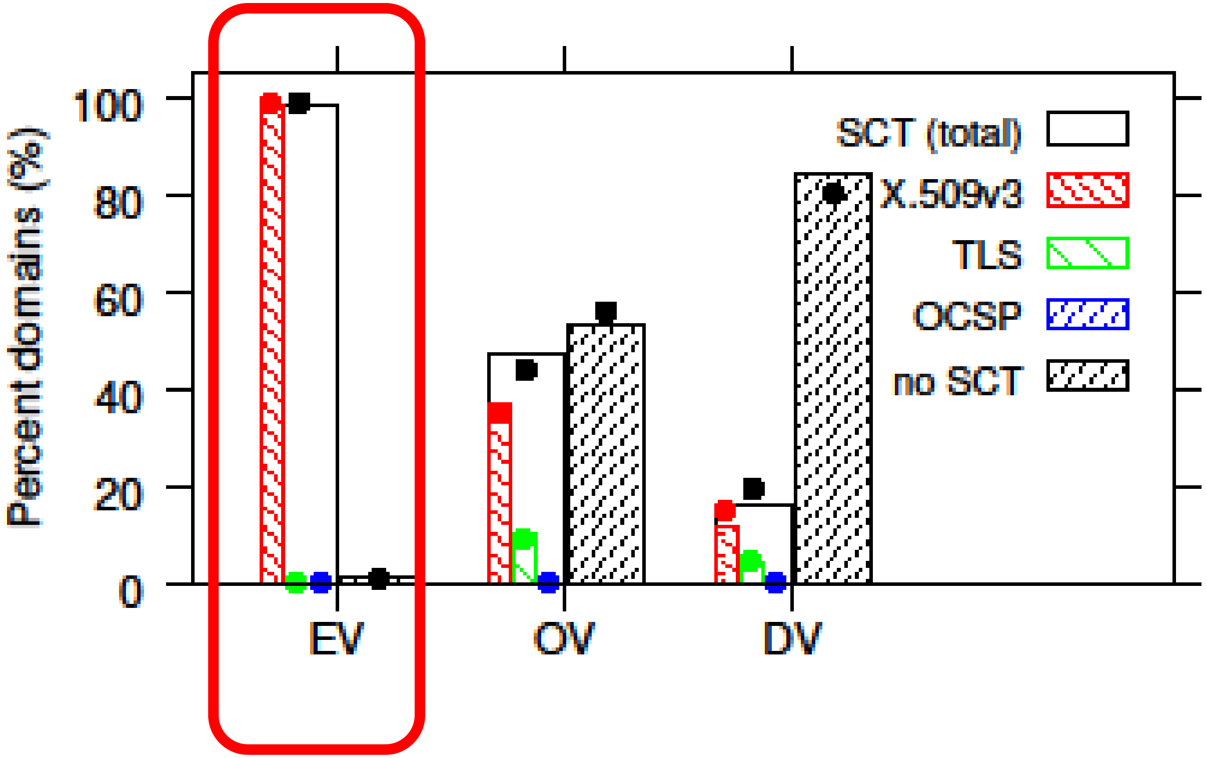
- A few dominating logs
- Big differences in TLS frequency among CA logs
  - Wosign almost only TLS
- Aviator (frozen on Nov 29, 2016) almost only X.509v3
  - Again, TLS is increasing (but way behind ...)

Certificate type



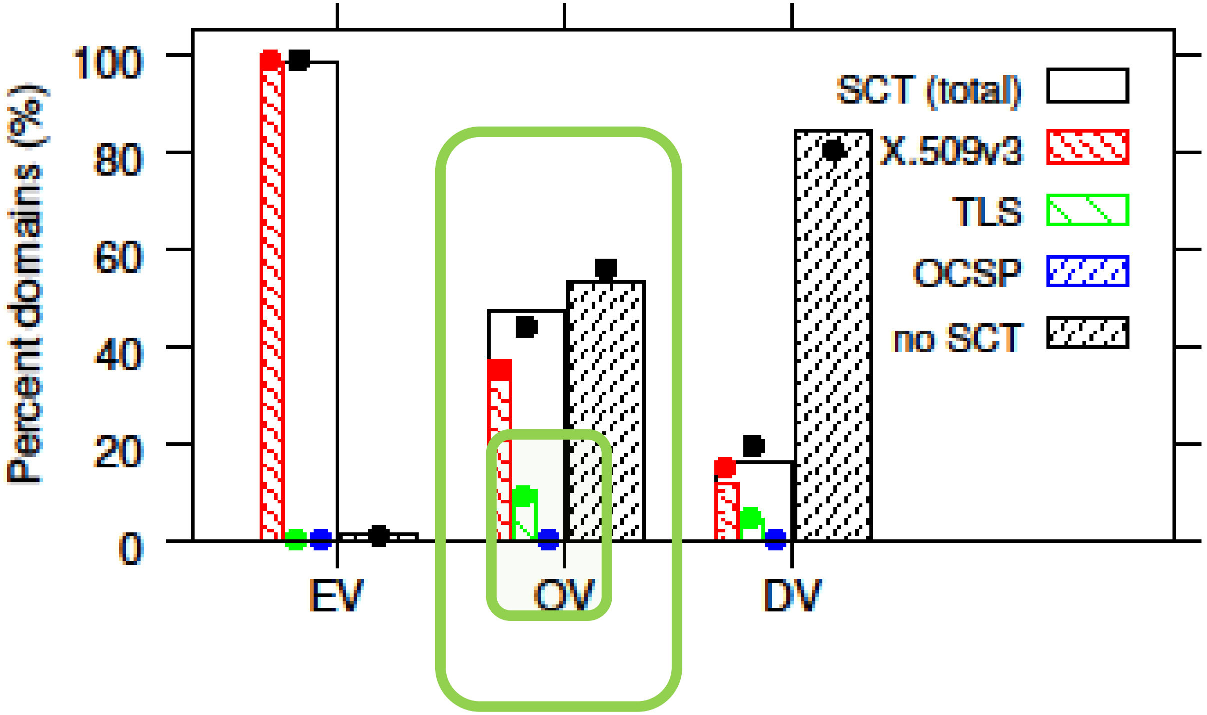
- X.509v3 dominates EV
  - Rush to get a solution ...
  - Simplest method
- OV certificates have highest fraction TLS
  - Google issued domains largest fraction here (7,858 / 8,374)
  - Comodo dominates TLS in DV (19,458 / 21,378)

Certificate type



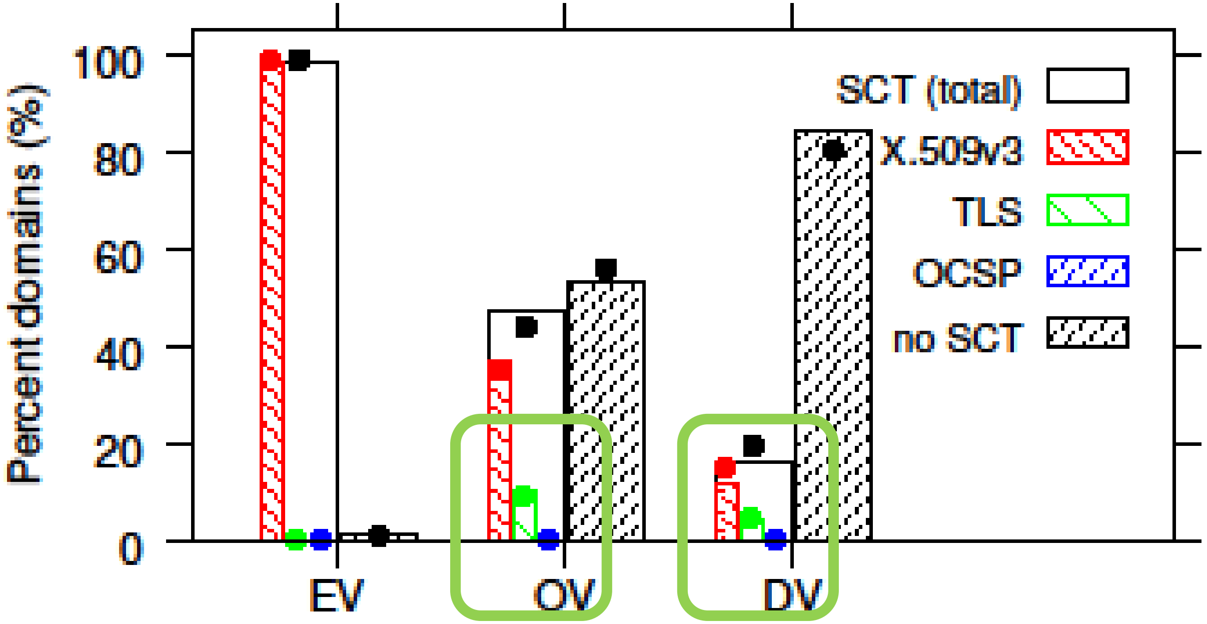
- X.509v3 dominates EV
  - Rush to get a solution ...
  - Simplest method
- OV certificates have highest fraction TLS
  - Google issued domains largest fraction here (7,858 / 8,374)
  - Comodo dominates TLS in DV (19,458 / 21,378)

Certificate type



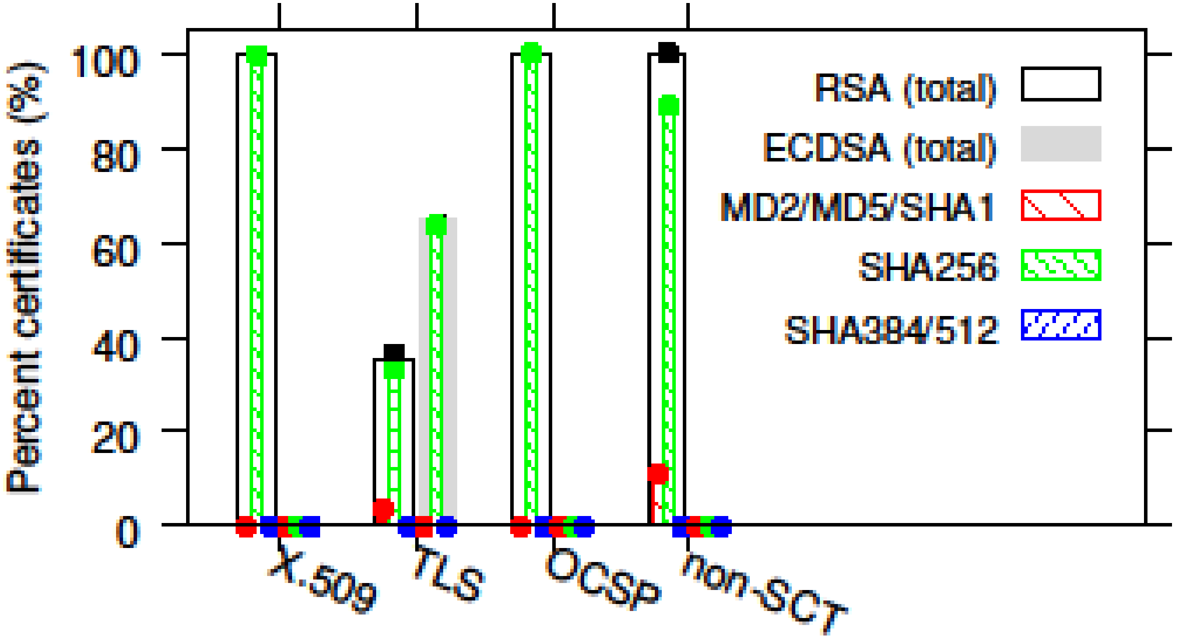
- X.509v3 dominates EV
  - Rush to get a solution ...
  - Simplest method
- OV certificates have highest fraction TLS
  - Google issued domains largest fraction here (7,858 / 8,374)
  - Comodo dominates TLS in DV (19,458 / 21,378)

Certificate type



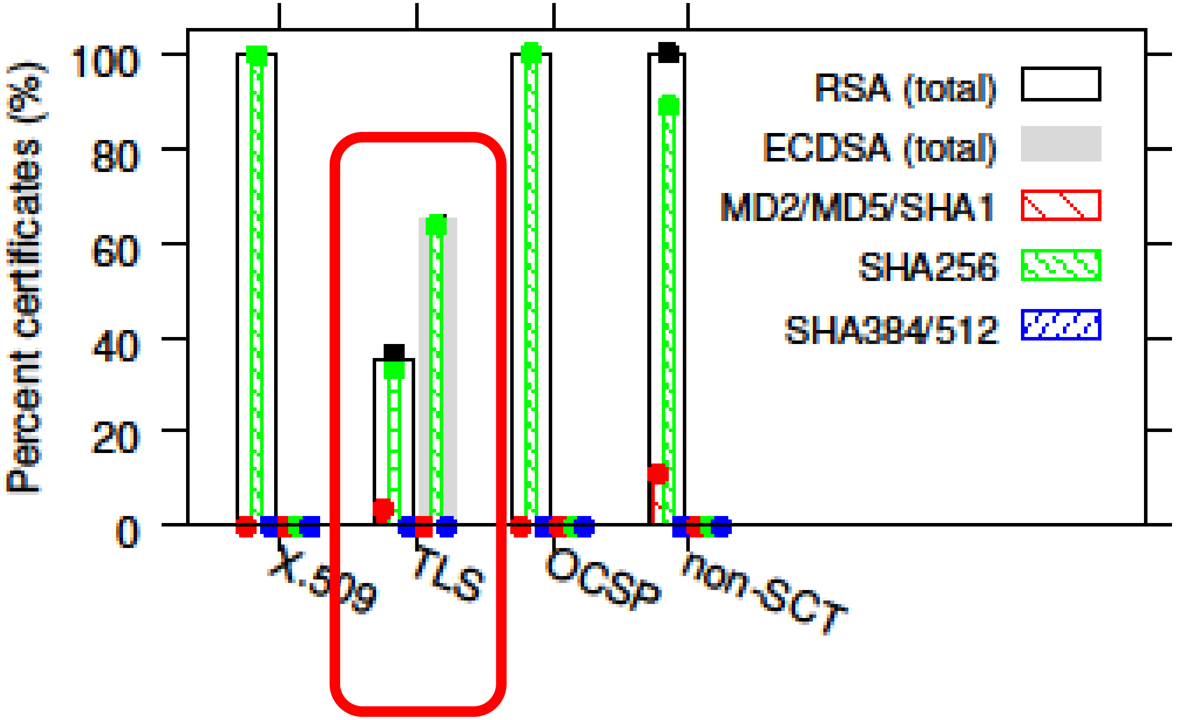
- X.509v3 dominates EV
  - Rush to get a solution ...
  - Simplest method
- OV certificates have highest fraction TLS
  - Google issued domains largest fraction here (7,858 / 8,374)
  - Comodo dominates TLS in DV (19,458 / 21,378)

Signatures (and keys)



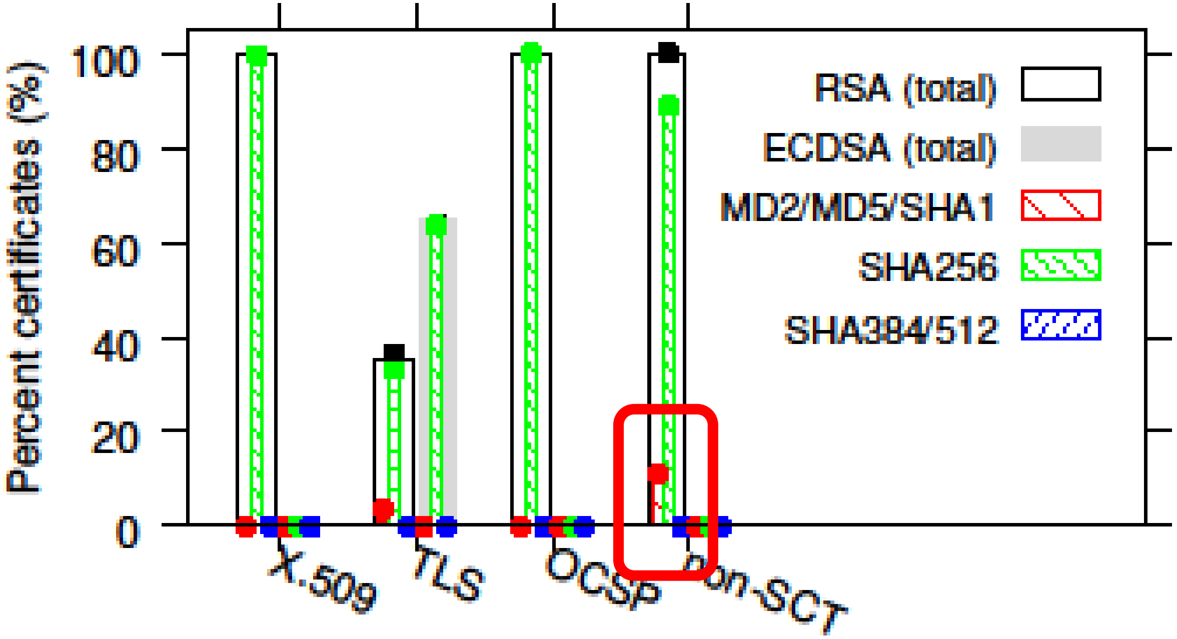
- RSA dominates all but TLS
  - TLS include 65% ECDSA signed and Elliptic Curve (EC) keys
- Non-SCTs weaker signatures (and shorter keys)

Signatures (and keys)



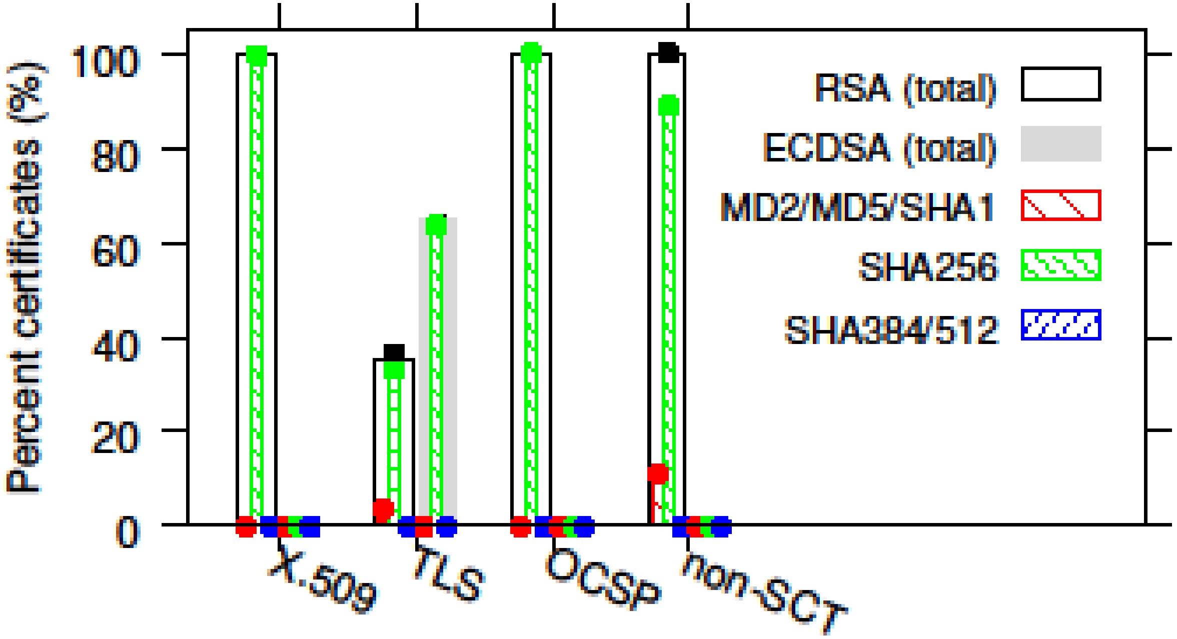
- RSA dominates all but TLS
  - TLS include 65% ECDSA signed and Elliptic Curve (EC) keys
- Non-SCTs weaker signatures (and shorter keys)

Signatures (and keys)



- RSA dominates all but TLS
  - TLS include 65% ECDSA signed and Elliptic Curve (EC) keys
- Non-SCTs weaker signatures (and shorter keys)

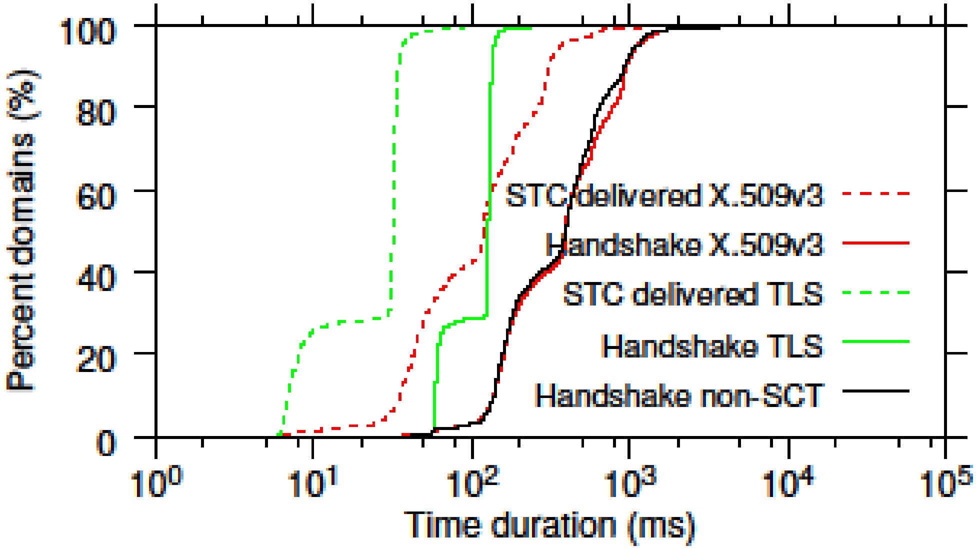
Signatures (and keys)



- RSA dominates all but TLS
  - TLS include 65% ECDSA signed and Elliptic Curve (EC) keys
- Non-SCTs weaker signatures (and shorter keys)

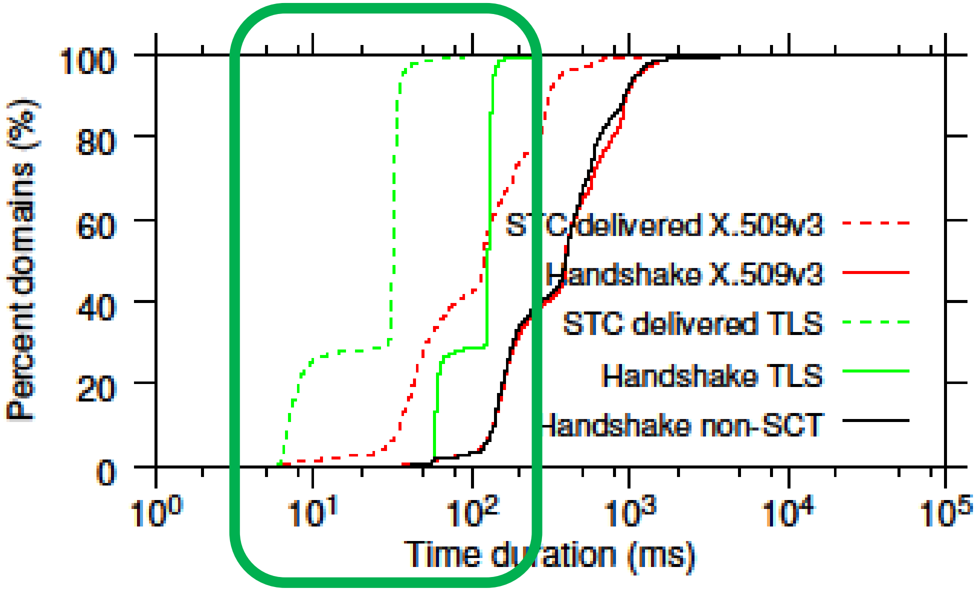
Handshake and delivery times

- TLS much faster than the other methods
- X.509v3 similar to non-SCT



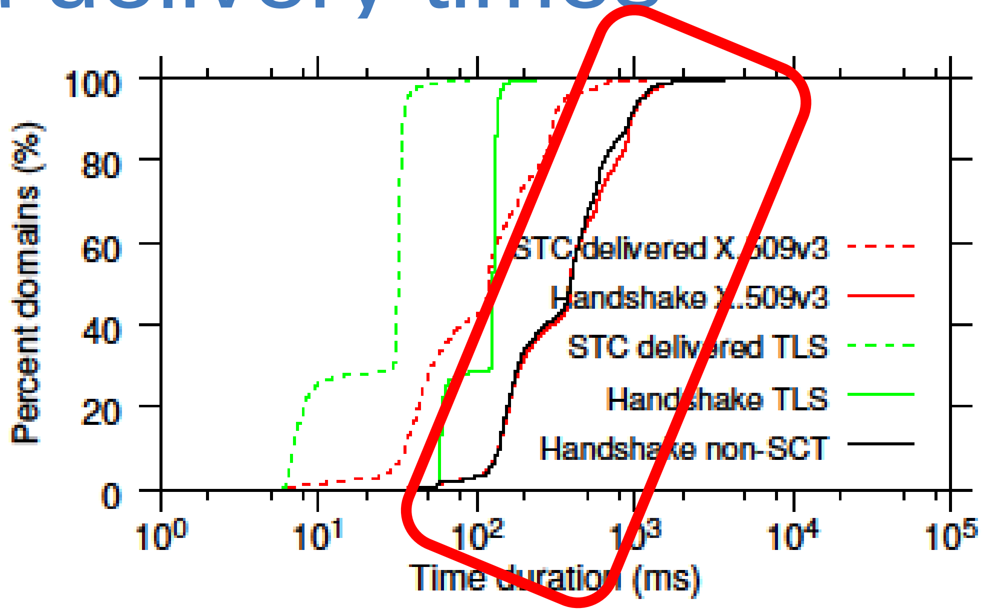
Handshake and delivery times

- TLS much faster than the other methods
- X.509v3 similar to non-SCT



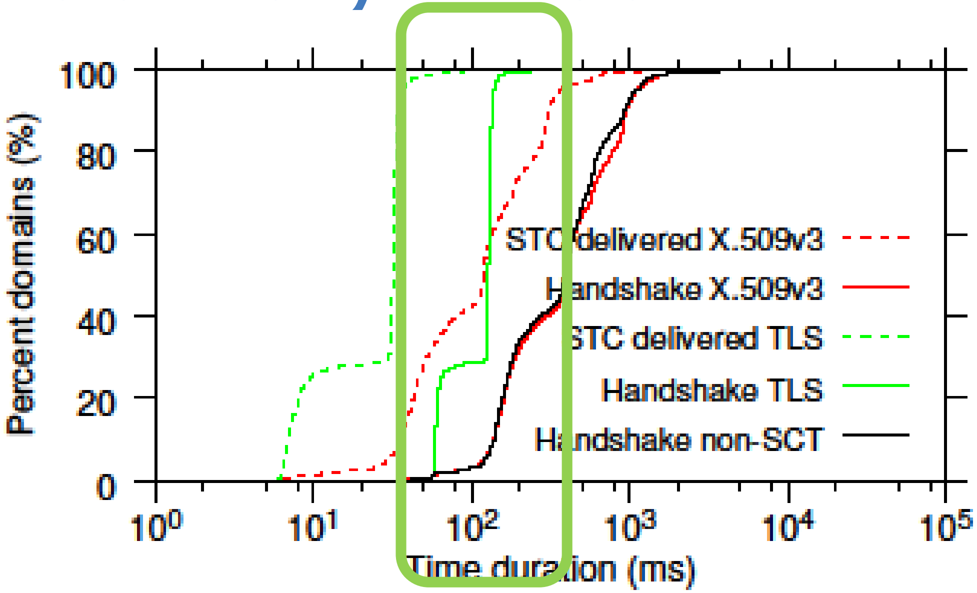
Handshake and delivery times

- TLS much faster than the other methods
- X.509v3 similar to non-SCT

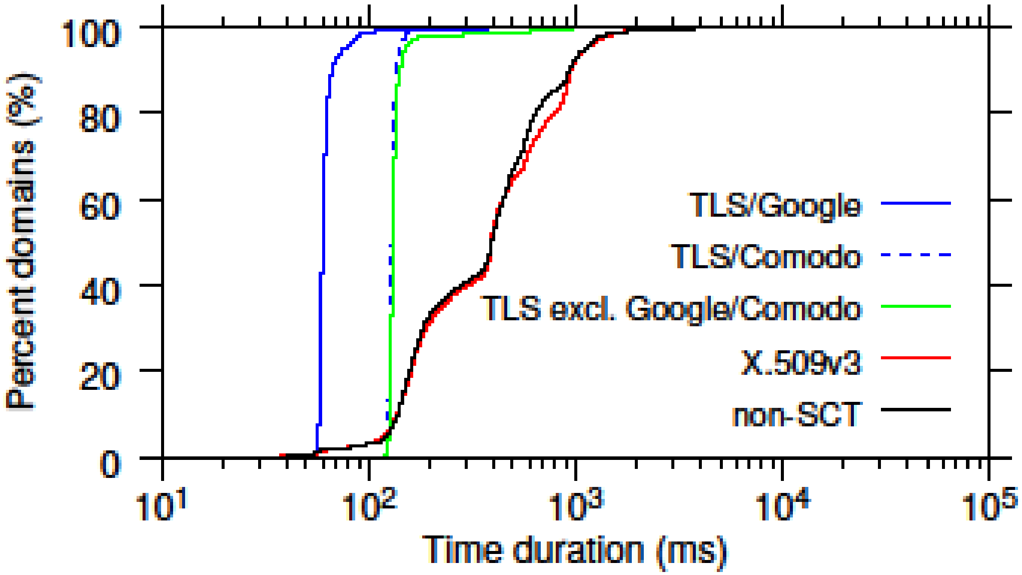


Handshake and delivery times

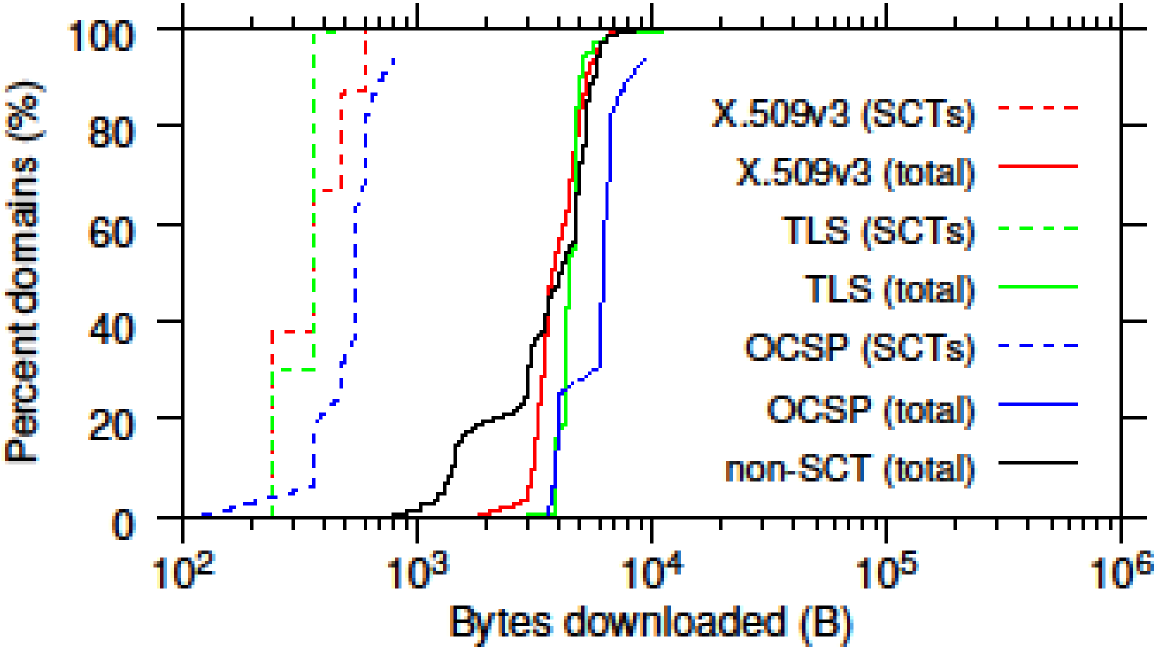
- TLS much faster than the other methods
- X.509v3 similar to non-SCT



- Google fastest, with short tail
- Comodo and other TLS domains both outperform X.509 domains



Byte overhead



- The SCT bundles have negligible byte overhead
- Otherwise SCT byte differences mostly due to bundle sizes and other differences dominated by the certificates themselves (keys included)

Conclusions

- SCT analysis: current status and trend
  - Two snapshots (May and Oct. 2017) of Alexa top-1M
- SCT usage is highest among the very top domains, hopefully pushing others to follow
  - Majority of domains selects simplest solution (X.509v3)
  - Fastest delivery method (TLS) is used by organizations (e.g., Google) that appear to provide much faster connection establishment and handshake times
- SCT delivery has low overhead
- Positive and encouraging trends in the adoption
  - Overall increase in use of SCTs
  - Use of SCTs goes hand-in-hand with a reduced use of weak signatures and public keys
  - Big players such as Google are pushing the adoption

Thanks for listening!

Server-side Adoption of Certificate Transparency

Carl Nykvist, Linus Sjöström, Josef Gustafsson, Niklas Carlsson

LINKÖPING UNIVERSITY

Niklas Carlsson (niklas.carlsson@liu.se)
www.ida.liu.se/~nikca/