Leibniz
Universität
Hannover

A Qualitative Comparison of SSL Validation Alternatives

OWASP AppSec Research 2013

August 22nd, 2013

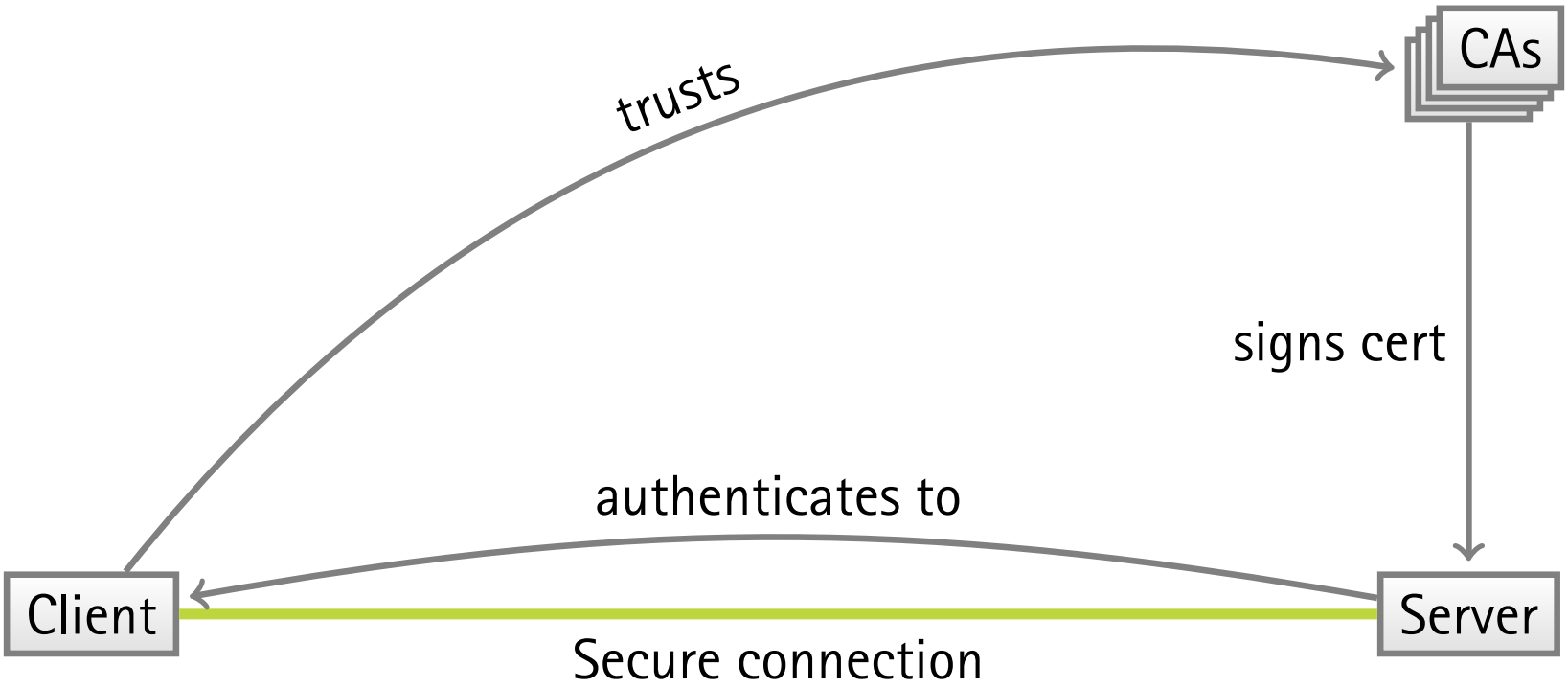
Henning Perl, Sascha Fahl, Michael Brenner, and Matthew Smith
Leibniz Universität Hannover

Outline Of This Talk

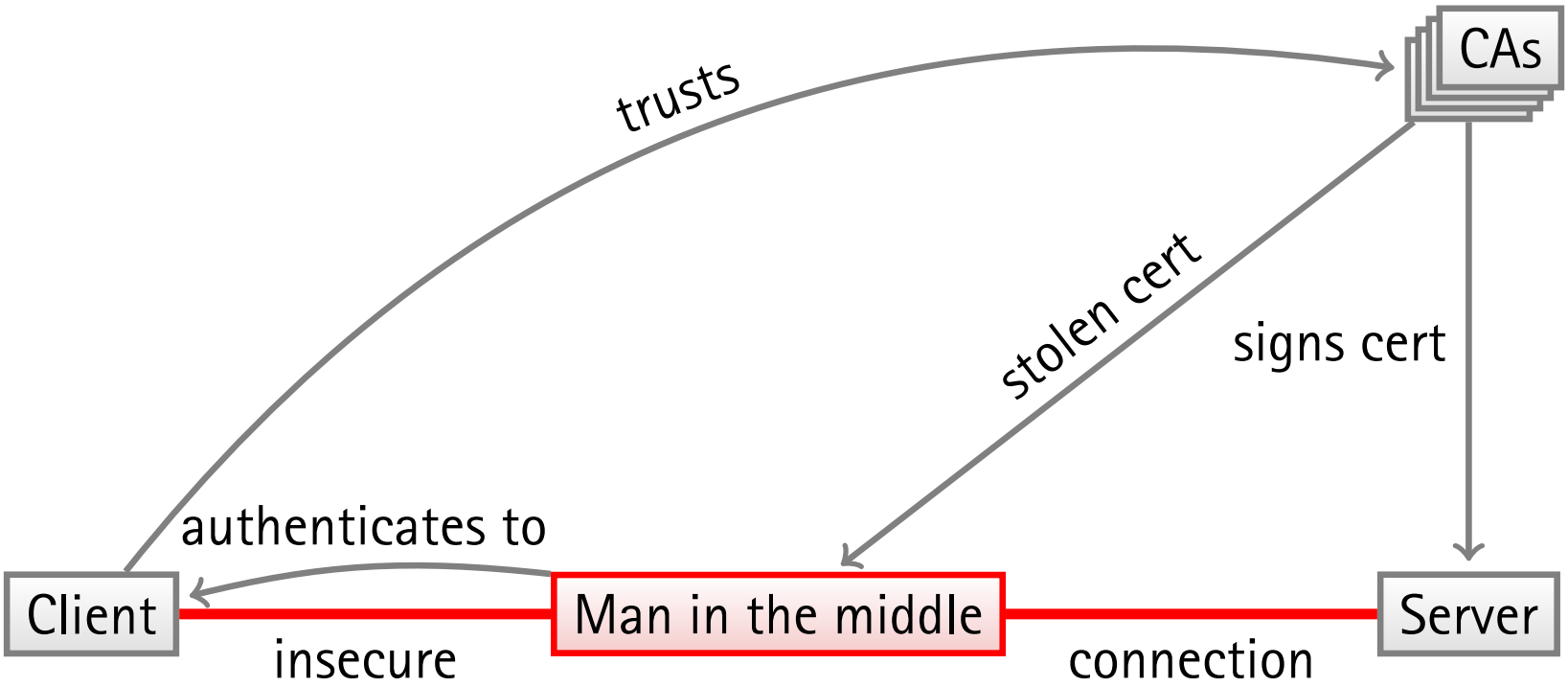
- What's SSL again?
- Things broken in SSL
- So many solutions!
- The best solution

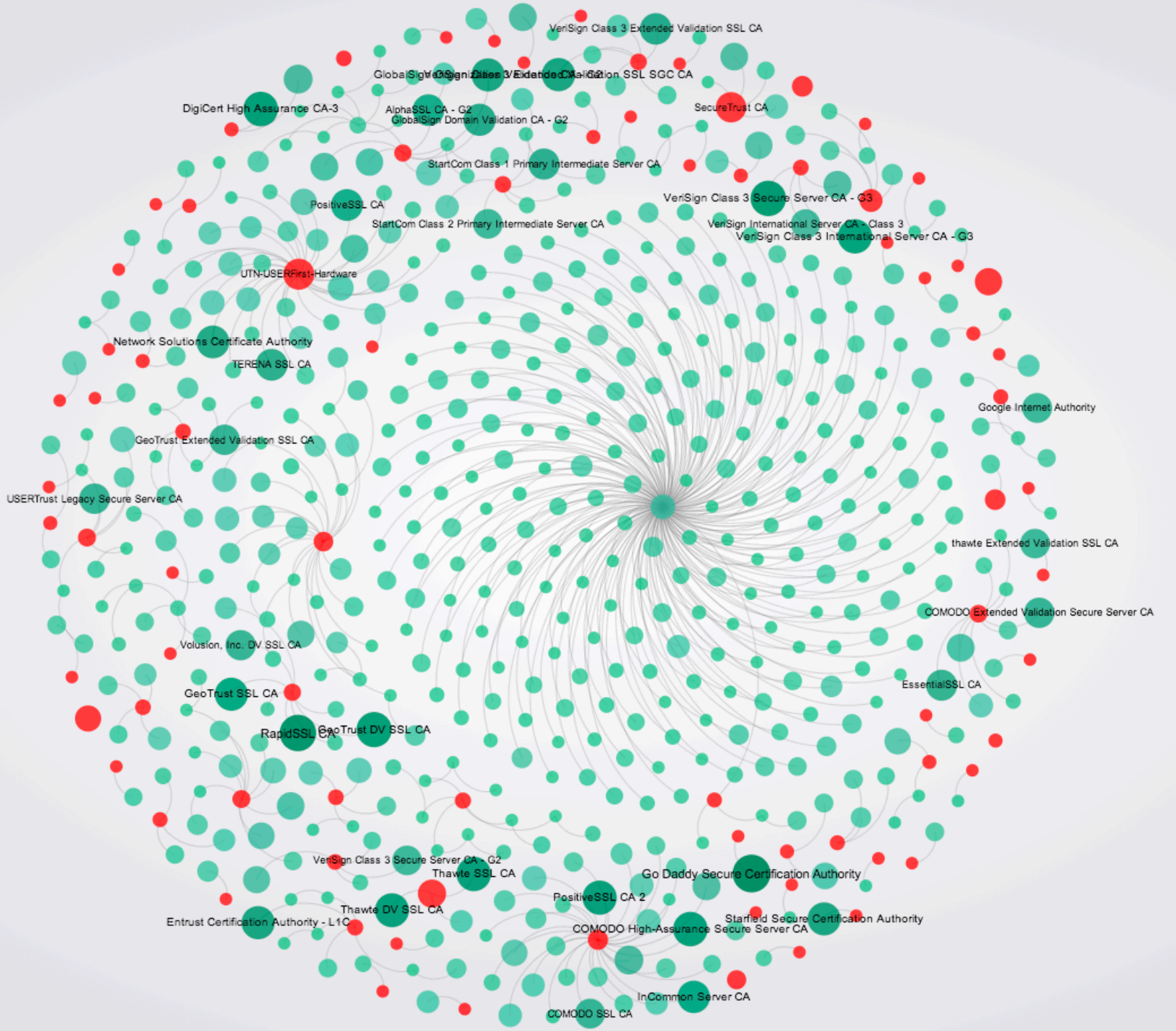
  (or why there isn't any yet)
- Our evaluation system

How SSL works



How SSL works ...and breaks





http://notary.icsi.berkeley.edu/trust-tree/

SSL CA incidents

- In 2010, VeriSign was compromised, allowing the attackers to issue arbitrary certificates.
- In March 2011, an attacker from Iran was able to compromise the Comodo CA and get certificates for www.google.com, login.yahoo.com, login.skype.com, addons.mozilla.org, and login.live.com. A MITMA attack with at least one these certificate was observed.
- In August 2011, attackers used the DigiNotar CA to issue at least 200 fraudulent certificates and used them to impersonate web servers. The breach eventually lead to the exclusion of the CA from most browsers and operating systems.

$\Rightarrow$ **weakest link** security

Things broken in SSL

For sake of completeness

- **Users ignore warnings**
  (c.f. Sunshine et al., "Crying Wolf: An Empirical Study of SSL Warning Effectiveness")
- Attacks against the **cryptosystem**
  - BEAST (2011) / CRIME (2012) attacks
  - Padding oracle attack ("Lucky Thirteen", S&P 2013)
  - Attacks against RC4 (Usenix 2013)
- **SSL stripping** (Marlinspike, Black Hat 2009)
- **SSL validation** / Weakest link CA security

Things broken in SSL

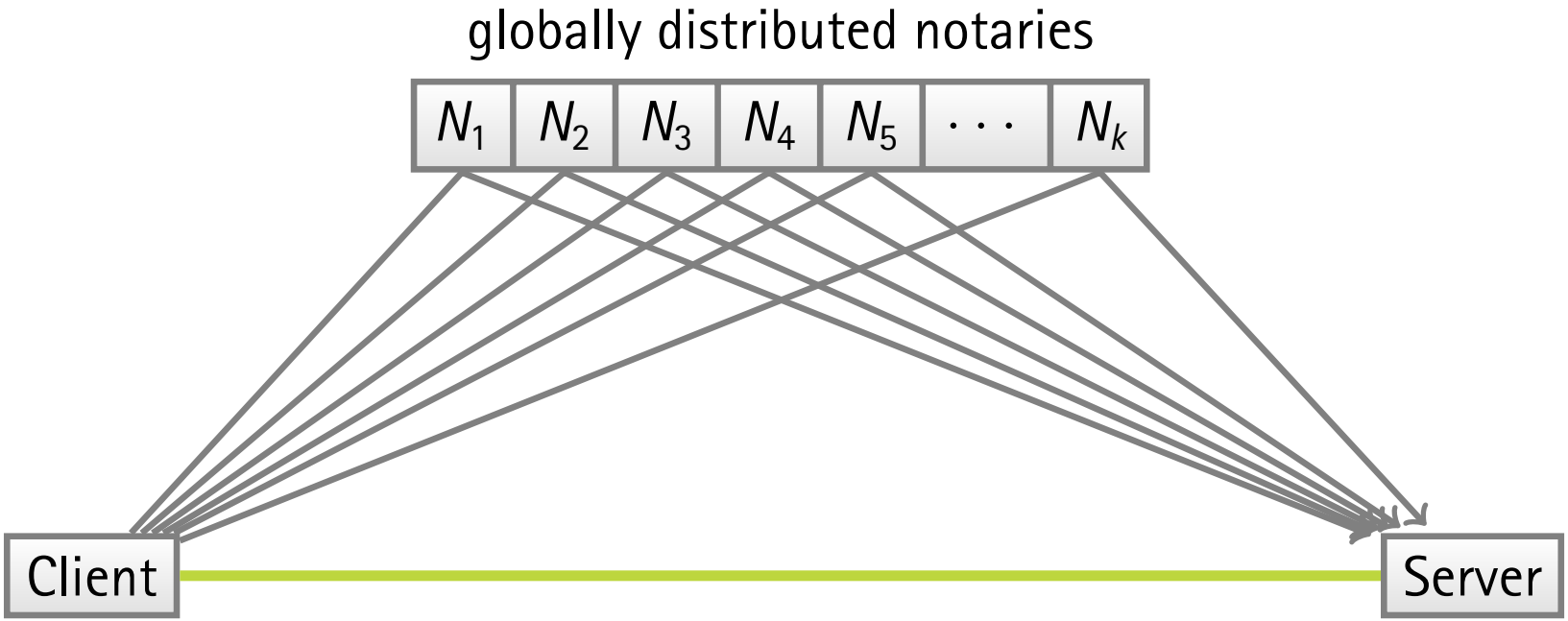
For sake of completeness

- **Users ignore warnings**
  (c.f. Sunshine et al., "Crying Wolf: An Empirical Study of SSL Warning Effectiveness")
- Attacks against the **cryptosystem**
  - BEAST (2011) / CRIME (2012) attacks
  - Padding oracle attack ("Lucky Thirteen", S&P 2013)
  - Attacks against RC4 (Usenix 2013)
- **SSL stripping** (Marlinspike, Black Hat 2009)
- **SSL validation** / Weakest link CA security

Types of solutions:

- **Use of network perspective**
  Perspectives, Convergence
- **Keep a log of certificates**
  Sovereign Keys (SK), Certificate Transparency (CT), Accountable Key Infrastructure (AKI)
- **Serve certificates over DNS**
  DANE
- **Trust on first use**
  TACK

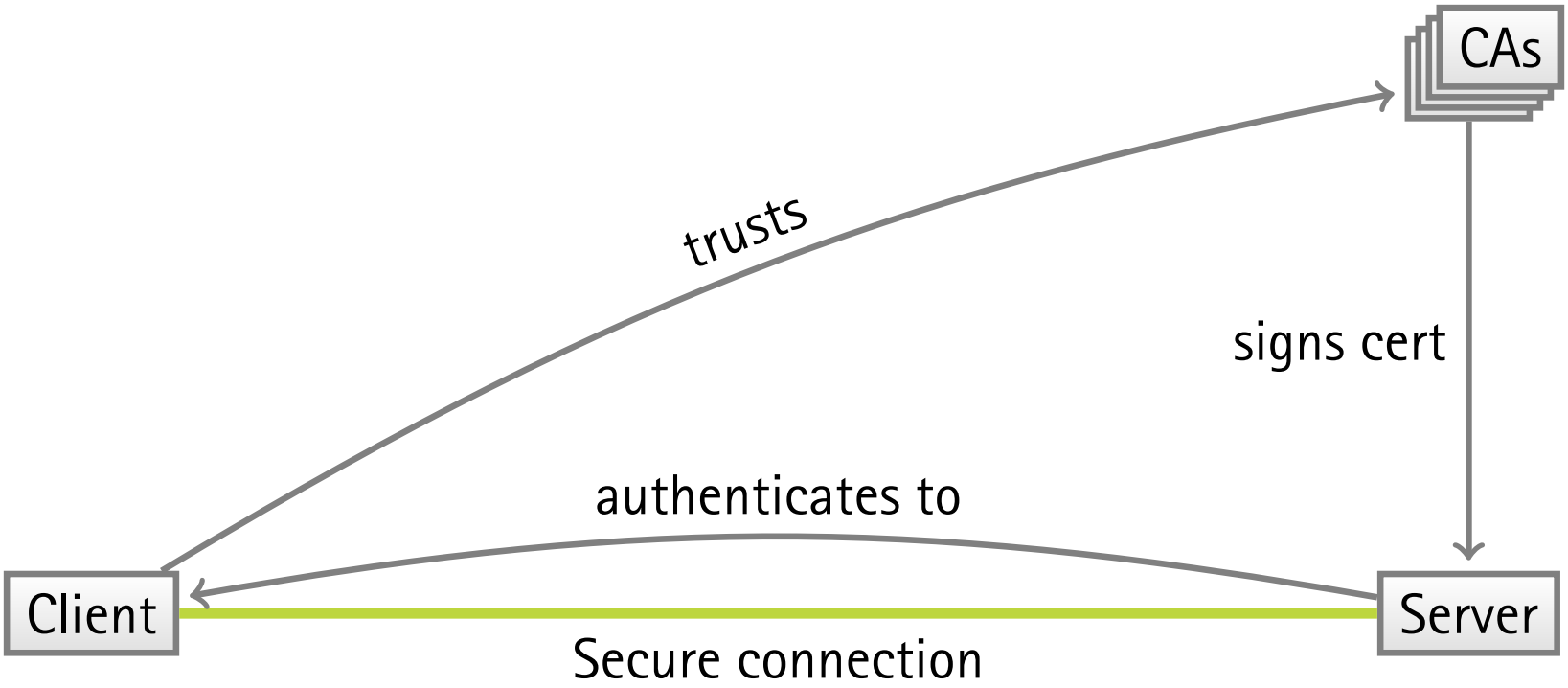
Network Perspective (Perspectives, Convergence)



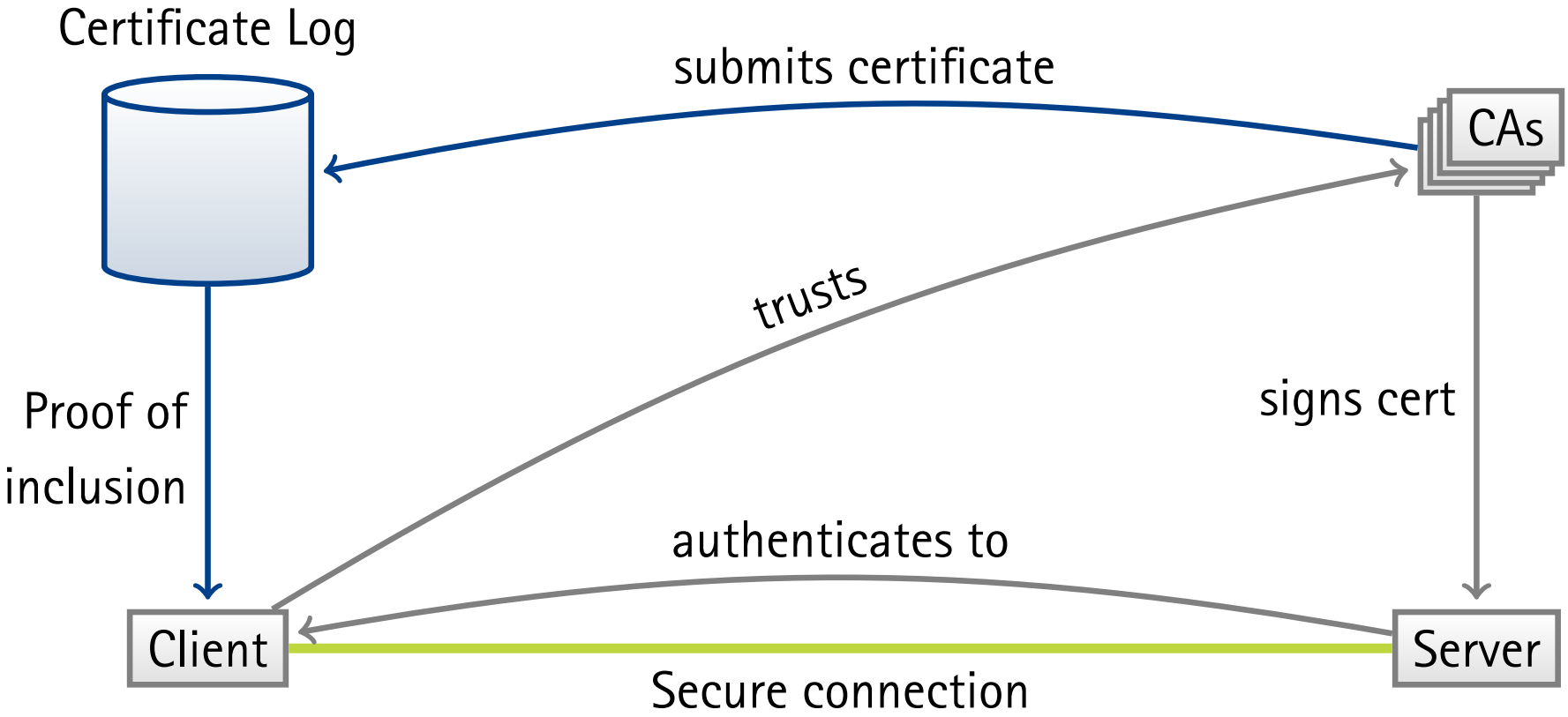
✓ No extra software on server

✗ Network delay ✗ Privacy

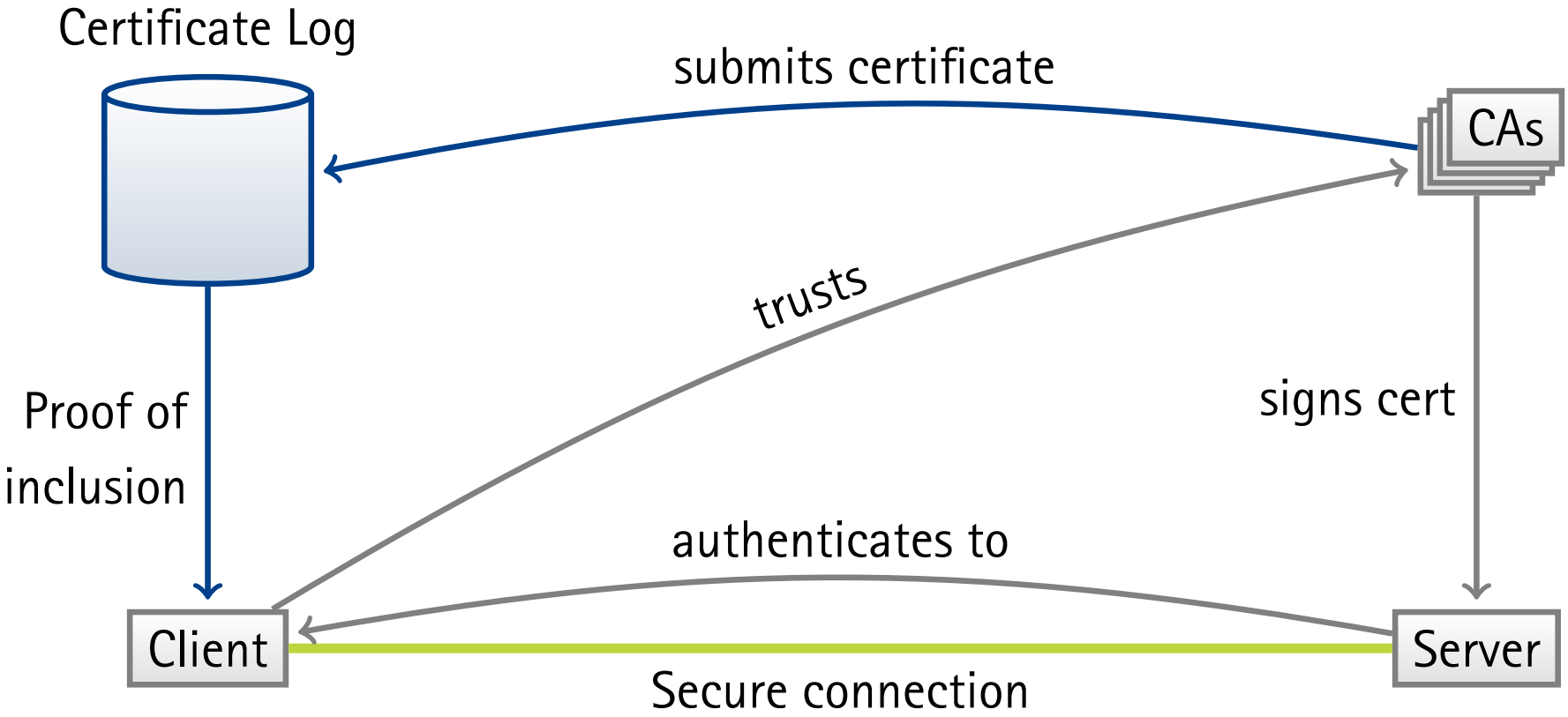
Keep A Log Of Certificates SK, CT, AKI



Keep A Log Of Certificates SK, CT, AKI



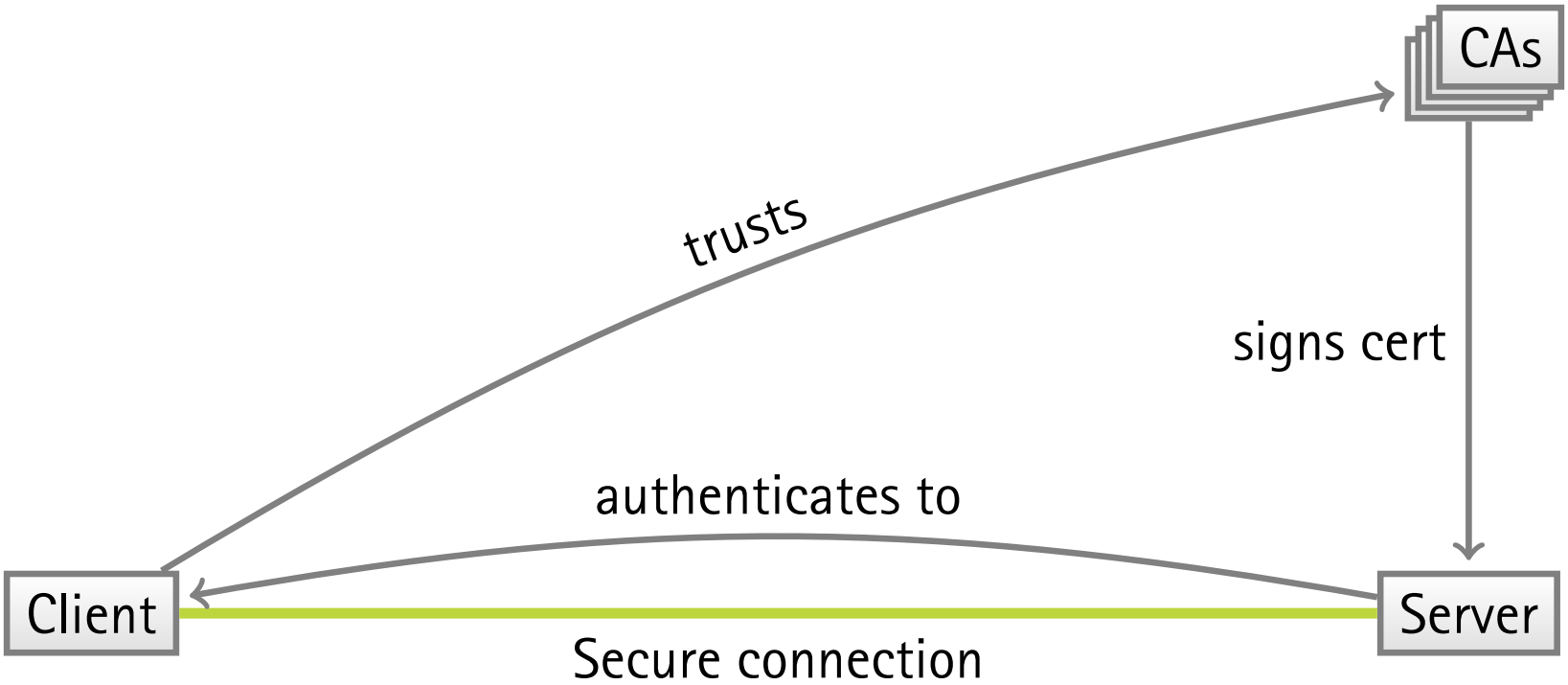
Keep A Log Of Certificates SK, CT, AKI



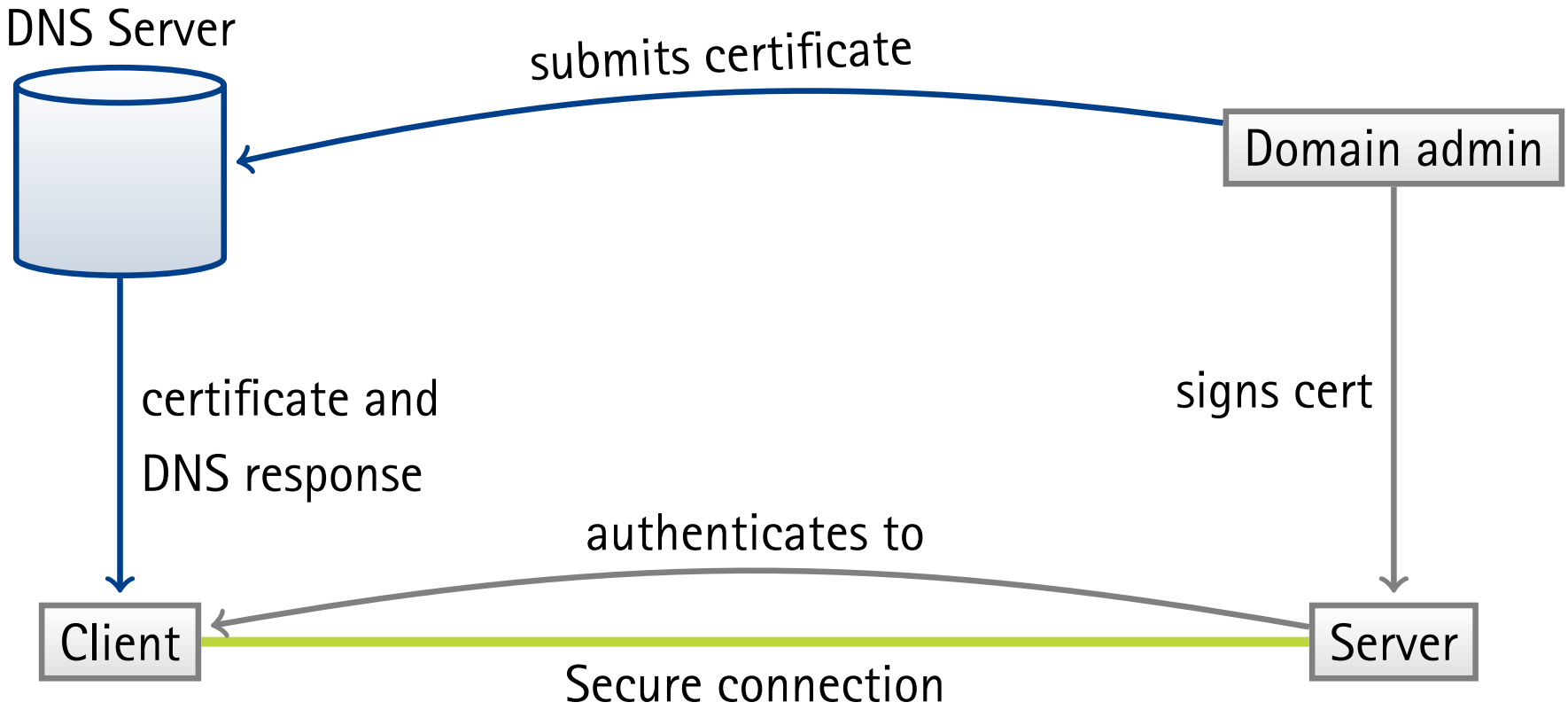
✓ No extra software on server ✓ no extra network delay

✗ needs new infrastructure

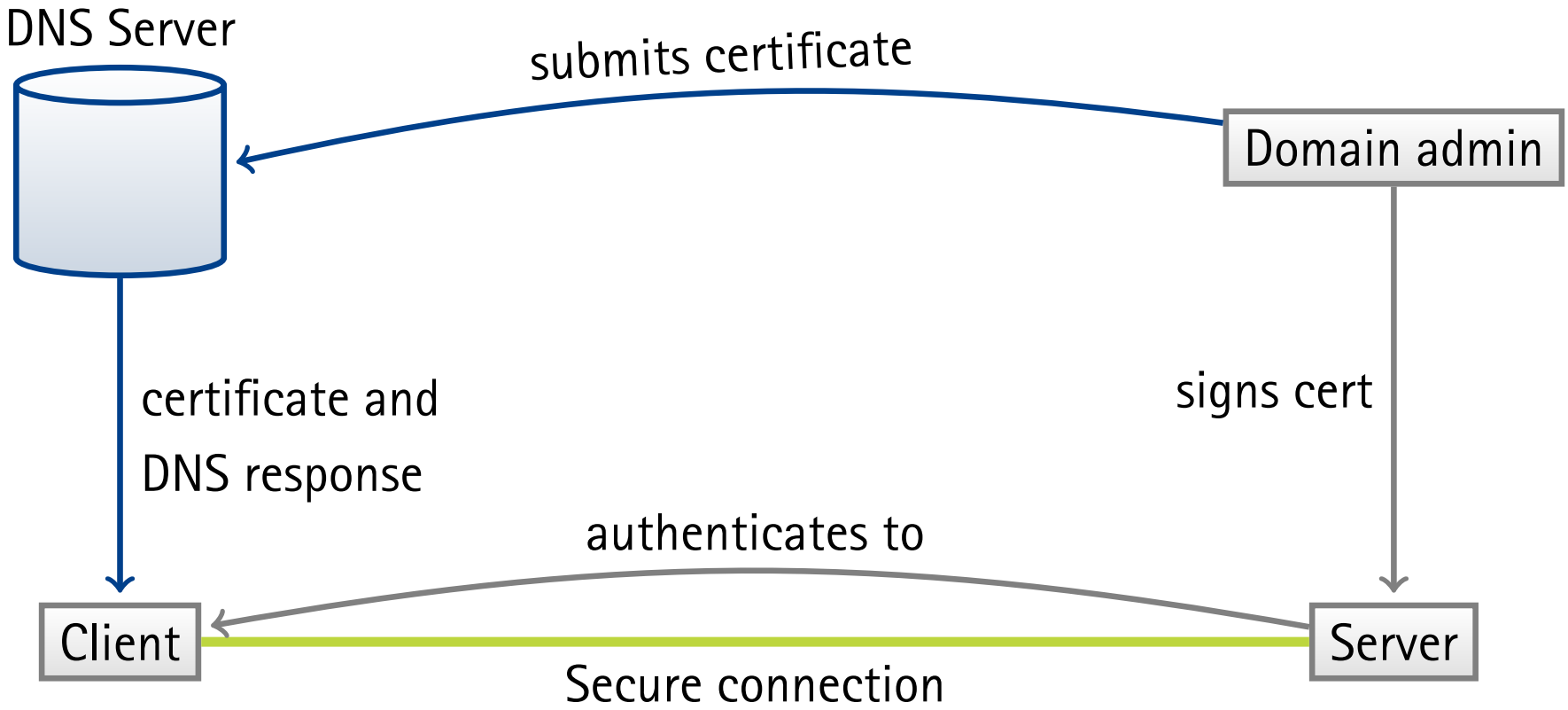
Serve Certificates Over DNS DANE



Serve Certificates Over DNS DANE



Serve Certificates Over DNS DANE

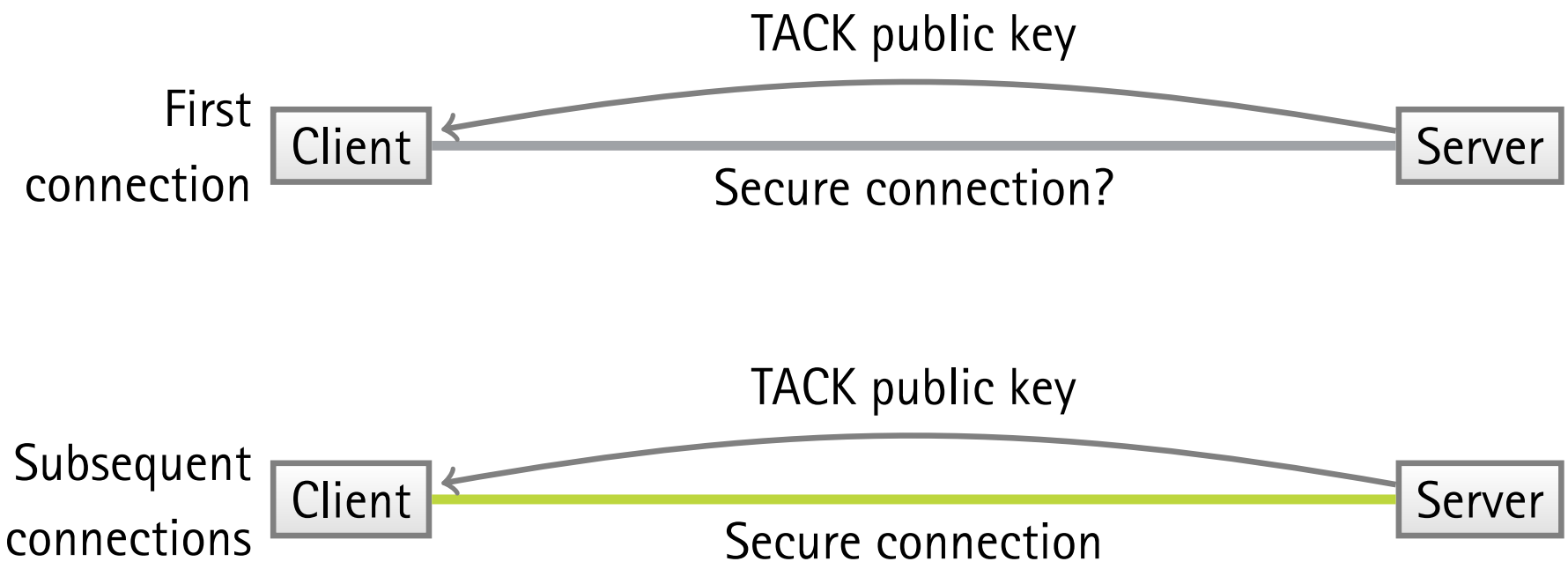


✓ No extra software on server ✓ reuses infrastructure

✗ DNSSEC

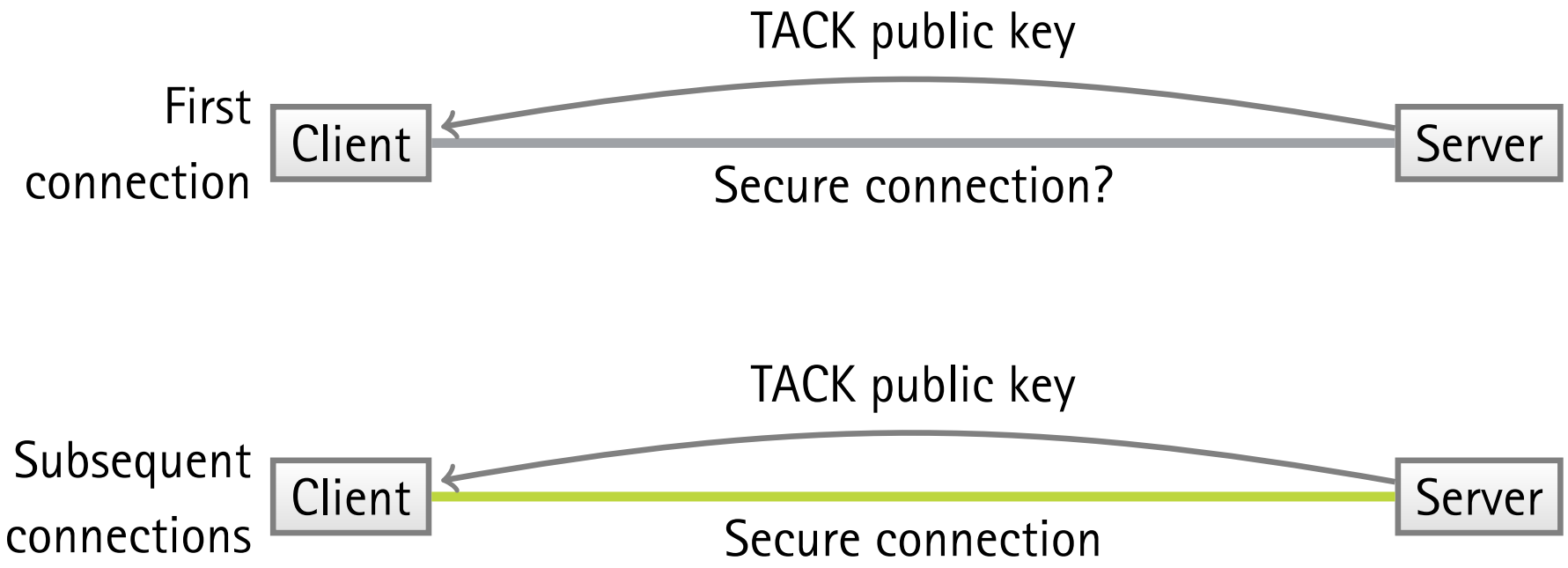
Pinning TACK

Pinning on TACK public key; TACK secret key signs actual cert.



Pinning TACK

Pinning on TACK public key; TACK secret key signs actual cert.



✓ No extra software on server ✓ no CAs (just selfsign)

✗ No protection on first visit

What do we draw from this?

Our Evaluation Scheme

Goals:

- Tool to compare solution
- Discussion about which properties are important
- Organize, formalize the debate

Our Evaluation Scheme

Goals:

- Tool to compare solution
- Discussion about which properties are important
- Organize, formalize the debate

Structure:

- One large table
- 12 Deployability Benefits
- 9 Security and Privacy Benefits
- Adversary Capabilities
  - Active MITMA required
  - Trusted CA certificate required
  - Compromising user chosen third parties required

Scheme	Ref.	Deployability benefits													Security benefits							Capabilities				
		No-User-Cost	No-Server-Cost	Server-Compatible	Browser-Compatible	Incrementally-Deployable	Negligible-Communication-Overhead	Negligible-Computational-Overhead	No-Additional-Infrastructure	Trusted-Root-CA-support	Custom-Root-CA-support	Selfsigned-Certificate-support	No-Out-Of-Band-Connection	X.509-Compatible	Built-In-Revocation	OCSP-or-CRL-Compatibility	Resilient-To-DOS-Attacks	User-Privacy-Preserving	Secure-Key-Migration	Secure-Key-Migration-After-Credential-Theft	Secure-Domain-Migration	Active MITMA required	Trusted CA certificate required (weakest link)	Compromising user chosen third parties required (strongest link)	First-Contact-Protection	Connection-Protection
SSL with CA-PKI	—	●	◐	●	●	●	●	●	●	●	◐	○	●	●	○	●	●	●	●	○	○	➡			●	●
(90's)																							➡		○	○
																								Σ0⊃	○	○
Perspectives	[13]	●	●	●	○	●	○	●	○	●	●	●	○	●	○	●	○	○	◐	◐	◐	➡			●	●
(2008)																							Σ⇨		●	●
																								◼n▶	○	○
DANE	[5]	●	◐	●	○	○	●	●	◐	●	●	●	●	●	○	●	●	●	●	●	●	➡			●	●
(2010)																							Σ⇨		●	●
																								◼1▶	○	○
Convergence	[9]	●	●	●	○	●	○	●	○	●	●	●	○	●	○	●	○	●	◐	◐	◐	➡			●	●
(2011)																							Σ⇨		●	●
																								◼n▶	○	○
Sovereign Keys	[3]	●	◐	○	○	◐	○	●	○	●	○	○	○	●	●	●	○	○	●	●	●	➡			●	●
(2011)																							➡		●	●
																								◼1▶	◐	◐
Certificate Transparency	[7]	●	◐	●	○	◐	●	●	○	●	○	○	◐	●	○	●	◐	●	●	●	●	➡			●	●
(2012)																							➡		●	●
																								◼1▶	◐	◐
TACK	[8]	●	●	○	○	●	●	●	●	●	●	●	●	●	○	○	●	●	●	◐	◐	➡			○	●
(2012)																							Σ⇨		○	●
																								Σ0⊃	○	●
AKI	[6]	●	◐	○	○	●	●	●	○	●	●	○	●	○	●	○	○	●	●	●	●	➡			●	●
(2013)																							➡		●	●
																								◼1▶	◐	◐

Conclusion

- All proposals solve weakest link problem
- ...but in very different ways
- No clear winner
- Do we want/need/have to have CAs?
- Deployment is challenging
- Question: *When to fail hard?*