RSA CONFERENCE 2014
FEBRUARY 24 – 28 | MOSCONE CENTER | SAN FRANCISCO

Share.
Learn.
Secure.
Capitalizing on
Collective Intelligence

New Ideas on CAA, CT, and Public Key Pinning for a Safer Internet

SESSION ID: TECH-T09

Moderator: Kirk Hall
Operations Director, Trust Service
Trend Micro

Panelists: Rick Andrews
Senior Technical Director for Trust Services
Symantec

Wayne Thayer
VP and GM, Security Products
GoDaddy

CT, CAA, Pinning - what are these technologies trying to do?

- Deal with mis-issued certificates from public CAs
- All of these are attempts to address weaknesses in and strengthen the existing SSL ecosystem
  - Wrongly vetted (fraud, imposter) – CA intentionally issued the cert, but in error
    - Cert still found in CA's logs, easy to find, revoke
  - "Rogue Certs" - Hackers take over CA system, issue fake certs, (Diginotar case)
    - Cert might be erased from CA's logs by hacker, can't be found, harder to revoke (added to browser CRL)

How many certificates get mis-Issued?

- Extremely low rate of mis-issuance – Compared to millions of valid certs each year. Possible sources:
  - Problems from simple CA vetting errors – almost no reports
  - CA issues intermediate cert to customer that's used to mis-issue end-entity certs (ANSSI government CA incident Dec. 2013 – revoked by browsers)
  - CA is breached, hacker issues rogue certs – few incidents, high impact:
    - 531+ fake Diginotar certs, CA logs erased - high fraud value FQDNs – *mail.google.com*, *login.yahoo.com*, *login.live.com*
    - 9 certs for 7 high-value domains in 2011 hacking incident – but CA log intact

What's the risk to the public from mis-issued certs?

- Today mis-issued certs are mainly found by monitoring groups crawling the internet, and by pinning (Google found fake Diginotar *google.com* certs this way)
- Mis-issued certs for high value FQDNs generally *can't* be used by hackers at different sites
  - The FQDN in the cert must *match* the FQDN of the web site visited or a warning is displayed to users
- But in some cases mis-issued certs can enable man-in-the middle (MITM) attacks

Example of warning from certificate mis-match

Where can a mis-issued cert be useful to a hacker?

- Anywhere the DNS can be altered or corrupted, or where the attacker can insert itself between client and server –
  - Enterprise networks at the firewall for MITM traffic interception – used to block viruses from corporate network (now outlawed by public CAs)
  - DNS spoofing, poisoning of DNS cache, redirection to spoofer's site (shows false FQDNs) – can be prevented by DNSSEC, other methods
  - Public WiFi networks – localized MITM attacks
  - Closed countries that corrupt their DNS (used to fool citizens, obtain email mail accounts, passwords, read confidential files) – ***most serious case***

RSA CONFERENCE 2014
FEBRUARY 24 – 28 | MOSCONE CENTER | SAN FRANCISCO

Certificate Transparency (CT)

Wayne Thayer

VP and GM, Security Products

GoDaddy

What problems does CT solve?

- No comprehensive way to detect mis-issuance by any one CA
  - Any Certificate Authority can issue a certificate for any domain
    - Many public CAs
  - Mis-issued certificates enable MITM attacks
  - Existing mechanisms slow to detect new certificates
  - Existing mechanisms can miss many certificates
- CA audit schemes are not sufficient to detect all compliance issues
  - Public record of issued certificates enables better oversight

How does CT solve these problems?

- Creates public log(s) of all SSL certificates
  - Enables monitoring for mis-issued and non-compliant certificates
- Has a mechanism for requiring that all SSL certificates be logged
  - Browser can hard-fail if certificate isn't logged
- Tamper-resistant
  - Logs can't be modified without detection
  - Ensures that certificates are added to logs

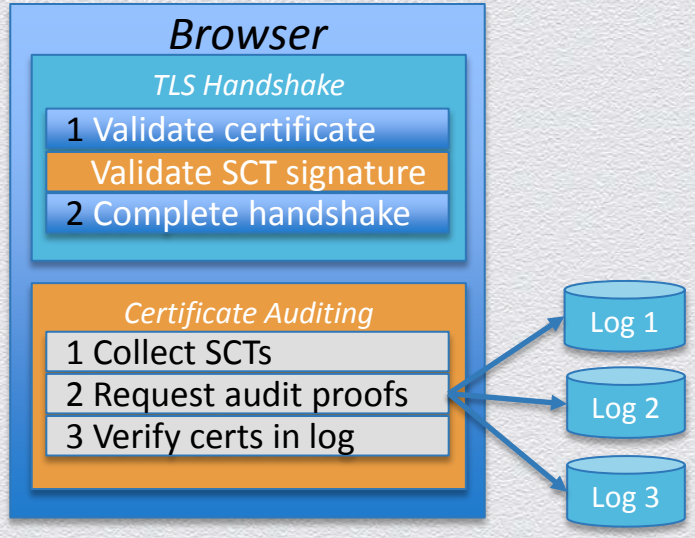
How does CT work?

- First, certificate is logged

  - Logs are append-only
  - Merkle hash trees used to detect inconsistencies
  - Certificate or “precertificate” is generated by CA and submitted to log
    - Submit to multiple logs (recommend 3 for redundancy)
  - Signed Certificate Timestamp (SCT) returned by log
  - Typically, SCTs are added to certificate via extension when issued
    - Or can deliver via TLS handshake or stapled OCSP response

How does it work?

- Browsers validate SCTs
  - SCT must be signed by a trusted log
  - No blocking connection to 3rd party
- Monitors watch logs
  - Often looking only for certain domains
    - Expect this work to be automated
  - CAs, large companies, and SSL watchdogs likely to run monitors
- Auditors verify the integrity of logs
  - Periodic verification that SCTs are found in logs



What are CTs strengths?

- Comprehensive – likely to be required for all publicly trusted SSL certificates
- Relatively mature – Experimental RFC 6962
  - Google logs deployed today; CT support in Chrome 33
- Enables early detection - certificates must appear in log before they can be used
- Deployable
  - Requires no changes on the web server to implement
  - Effective when a fraction of browsers support it

What's are CTs weaknesses?

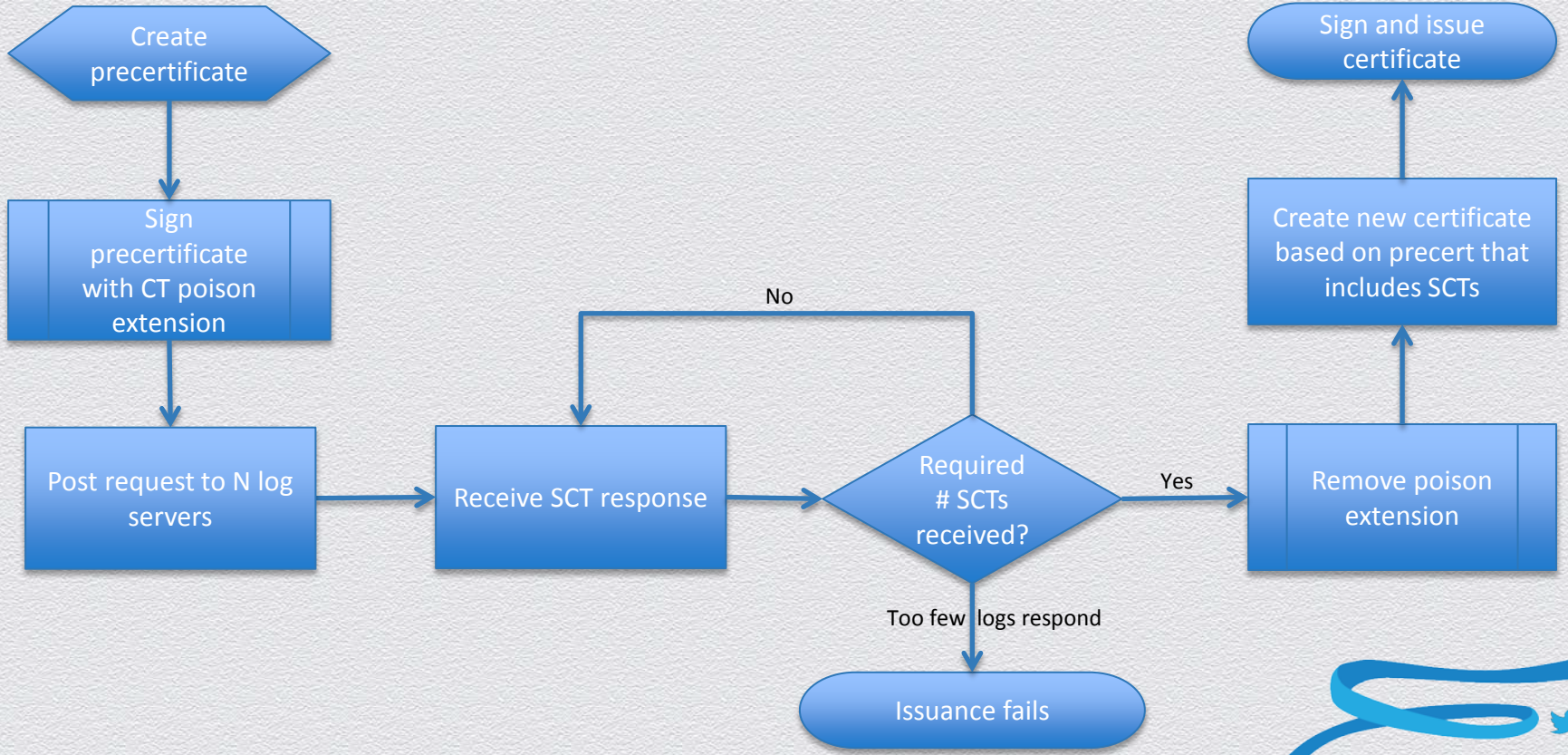
- It only works if someone is monitoring for a particular domain
- Monitors have potential to create lots of false alerts
- It can't prevent or mitigate an attack (e.g. Diginotar) – only detect
- It adds unknown cost and complexity for CAs
- Interrupts current cert issuance processing; could introduce vulnerabilities
- Logs must be highly available – they can block cert issuance
- Public log of all certificates creates privacy & data leakage concerns
- Increases TLS payload

1. CA submits precertificate to N logs

2. Log operators provide SCTs

3. CA confirms integrity of SCTs

4. CA issues certificate with embedded SCTs



The future of Certificate Transparency

- Google plans to require CT for Extended Validation certificates
  - EV certificates issued after July must contain SCTs
  - Google may require CT for all SSL certificates at a later date
- Some CAs adding CT support and deploying logs
- Need to determine:
  - Who will perform monitoring, and how?
  - What happens when a monitor or auditor detects a problem?
  - Which logs will be trusted by which browsers?
  - How will the number of trusted logs be managed?

RSA CONFERENCE 2014
FEBRUARY 24 – 28 | MOSCONE CENTER | SAN FRANCISCO

Certificate Authority Authorization (CAA)

Rick Andrews

Senior Technical Director

for Trust Services, Symantec

What problems does CAA solve?

- Web site owners have no way today to indicate their preference of CAs (authorized CAs) for their domains to prevent mis-issuance by a non-authorized CA

How does CAA solve these problems?

- CAs would check for the web site's CAA record in DNS before issuing a cert
- If the CA is included in the list of preferred CAs, it can issue the cert
- If the CA is not clearly included, it should discuss with the site owner (business rules not mandated by the spec)
- If the web site owner has not listed any preferred CAs in the DNS, the CA can issue the cert

What are the strengths of CAA?

- It can prevent mis-issuance, not just detect it after the fact
- Low cost of implementation for customers who are concerned about mis-issuance
- Low cost of implementation for CAs, and no cost for applications like browsers
- No cost for customers who are not concerned about mis-issuance
- Easily expandable to include multiple CAs, preference easily changed
- Reporting mechanism can alert site owners when mis-issuance is attempted

What are the weaknesses of CAA?

- Current spec gives CAs a lot of leeway on how to respond if the CA is not listed in the web site's CAA record
- Large customers may have multiple cert buyers, not the same people who maintain the company's web sites/DNS records (coordination issues)
- Possible competition issues, CAA could make it hard for new CAs to get business if a customer has indicated a different preference
- To be effective, we need broad adoption among the majority of CAs
- CAA is not yet supported in many DNS implementations
- Most secure with DNSSEC, which is not yet widely deployed (but can be used with DNS)

What does CAA not do compared to CT and Pinning/HPKP?

- CAA does not attempt to publish all issued certificates
- CAA does not attempt to determine if the cert presented by a web server is the legitimate cert for that domain name

RSA CONFERENCE 2014
FEBRUARY 24 – 28 | MOSCONE CENTER | SAN FRANCISCO

Certificate Pinning

Rick Andrews

Senior Technical Director

for Trust Services, Symantec

How does Pinning work?

- Domain owner pins hash of one or more public keys in the cert chain to the website
- First time visiting a site, site returns public key pins to Browser via HTTP headers
- Browser checks that at least one pin is valid for the cert chain presented
- Browser caches pins in case none are received on next visit

What problems does Pinning solve?

- Reduces the incidents of MITM attacks due to compromised CAs by having the browser compare cached hashes of known valid keys for a particular web site with the hashes of the keys securing the web site currently being visited
- If no match, a report is sent or access is blocked, or both

Further details on Pinning

- Browser must check that at least two different pins are included (so there is at least one “backup pin” to cover transition from expiring cert, etc.)
- Browsers cache pins for the max-age defined in each pin (determined by web site owner)
- Browsers hard-fail if there is no intersection between cached pins and subject public key info of all certs in the validated chain
- A pin can be “report only” (report pin failures but don’t block access)

What are the strengths of Pinning?

- Site owners who care most about mis-issued certs (e.g., top fraud targets) have sophisticated IT groups capable of implementing Pinning
- Allows each site owner to optionally pin one or more keys
- Site owners can pin keys for end-entity, intermediate or root certs

What are the strengths of Pinning?

- Backup pins allow for a transition from old to new key, in cases of compromise or normal key replacement
- “includeSubDomains” directive can effectively block access to a rogue site unknown to the site owner
- Chrome’s hard-coded pins have successfully detected mis-issued certs (e.g., Diginotar)
- Pinning can scale beyond pins currently hard-coded in browsers like Chrome

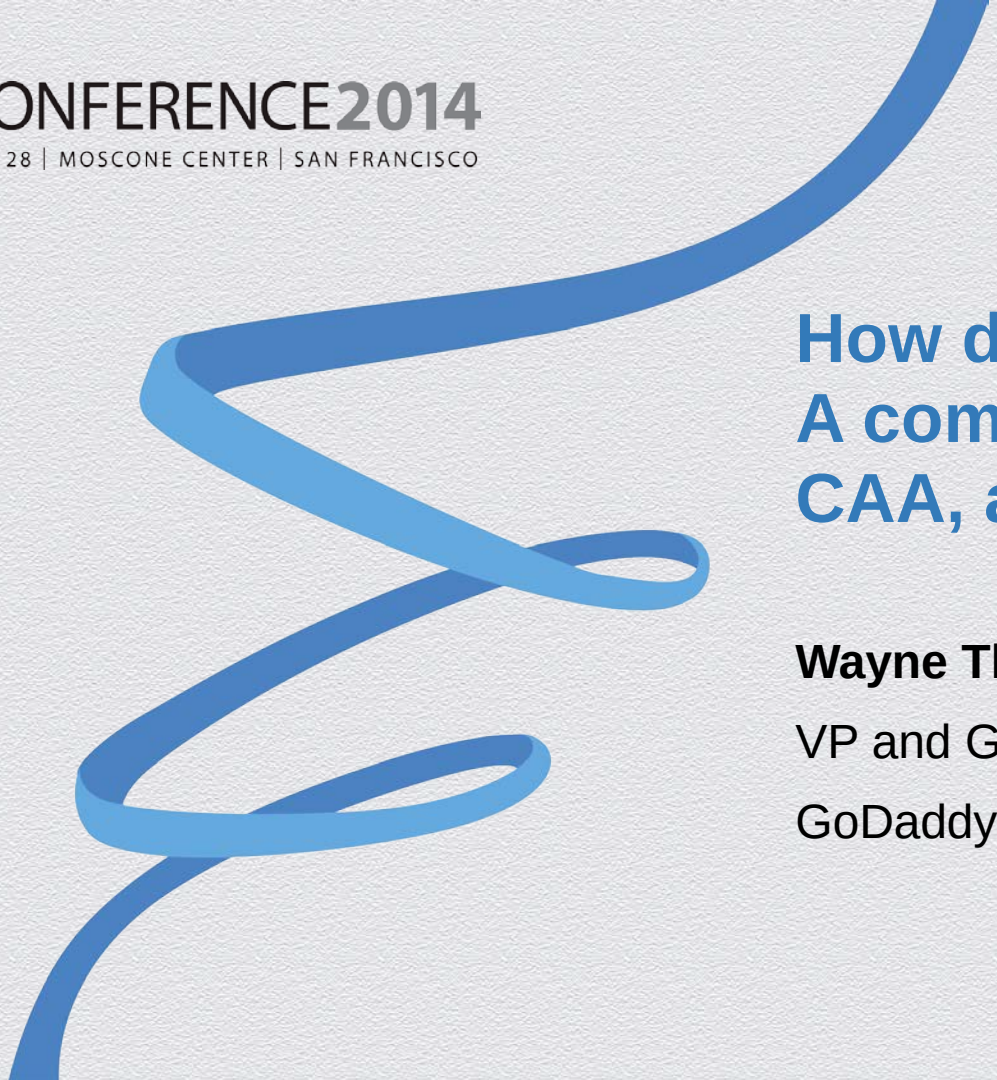
What are the weaknesses of Pinning?

- Requires Trust On First Use – preloaded pins address this, but aren't scalable
- Incorrect pin set can block all access to a site ("bricking")
- May be beyond the technical capabilities of many site operators, possible incorrect implementation
- "includeSubDomains" directive, if not used carefully, can block access to legitimate sites
- Could be abused to allow tracking of users

What does Pinning not do compared to CT and CAA?

- Pinning does not prevent mis-issuance by a compromised CA, but it can block all access to sites with mis-issued certs (neither CT nor CAA can block mis-issued certs)
- Pin checks can be carried out entirely by browsers; no action is needed by CAs
- Pinning can be limited to those web sites whose owners worry about mis-issued certs (e.g., top fraud targets), no others need to take any action

RSA CONFERENCE 2014
FEBRUARY 24 – 28 | MOSCONE CENTER | SAN FRANCISCO



How do they stack up? A comparison of CT, CAA, and Pinning

Wayne Thayer

VP and GM, Security Products

GoDaddy

Issue	CT	CAA	Pinning
Ability to *prevent* rogue cert issuance	**None**	**Moderate** – depending on CAA business rules, compliance by all CAs	**None**
Ability to *detect* rogue certs after issuance	**High** – but only if target domain owners monitor all CT logs for rogue certs (potential delay in detection)	**None**	**High** -- Chrome's hard-coded pins have successfully detected serious cases of mis-issuance
Ability to *detect* rogue certs after issuance – *countries with closed or controlled DNS*	**High** – cert must be included in multiple public logs or else browser will hard fail	**None**	**Moderate** – browser will hard fail but may not be able to report failure
Hard fail to protect users?	**Yes** (if cert not signed by CT logs) – but rogue certs signed by CT logs will be treated as valid, no hard fail	**No**	**Yes**
Revocability of rogue certs	Improves potential to detect mis-issued cert, but only if domain owner is monitoring CT logs	No change from present system – no easy way for owner or user to detect mis-issued cert	HPKP (assuming hard fail) is equivalent to revocation of mis-issued cert (any cert not pinned to the website)

Issue	CT	CAA	Pinning
Potential latency/performance issues	**None** as to the user agents, but CT logs must be high-availability or CAs can't issue certs (creates a new external dependency)	**None**	**None**
DOS Issues	**Potential issue** – if CT logs are blocked, certs can't be issued and CT logs can't be monitored during crucial periods – but multiple CT logs will exist	**None**	**None**
Scalability Issues	**Significant** - New high-availability infrastructure will be required, but scalable once established	**None**	**None** - HPKP can scale beyond pins currently hard-coded in browsers like Chrome
User Privacy Issues	**High** - All issued certs would instantly become public and capable of copying	**Low** - CA preferences for domains are listed in publicly viewable DNS record	**Low** - Hash for website's public keys are publicly viewable in domain's DNS record). But theoretical privacy issues stated at *HPKP and Privacy – IETF WebSec*

Issue	CT	CAA	Pinning
Requirements on CAs	**High** (complex, cost unknown, creates external dependencies)	**Moderate** (depending on business rules adopted). Some extra customer communication needed, potential competition issues	**Low** – CAs will have to teach customers how to use, deal with impact when changing intermediate or root certs (if pinned to the CA)
Requirements on Browsers	**High** (change user agent to monitor certs for CT log signatures using 3 methods, choose CT logs to trust, audit CT logs)	**None**	**Moderate** – browser user agents must be modified to check user's key hash against pinning information in DNS, cache pins, display warnings or hard fail
Requirements on Domain Owners	**Moderate** (Owners who care must monitor CT logs or pay for monitoring service – all enterprises must keep a central record of all valid certs for their organization) CT requires all domains owners to participate by listing their certs in public CT logs	**Moderate** for participating domain owners - must list permitted CAs in all DNS entries Participation by domain owners is purely voluntary	**High** for participating domain owners – domain owner must keep pinning records for all valid certs updated on all servers, could block access to site Participation by domain owners is purely voluntary

Issue	CT	CAA	Pinning
Other dependencies	**High** - Multiple CT logs must be established – cost, security, CT logs must be authorized and master CT log lists created Owners must monitor all CT logs, or pay for monitoring service Who will provide log audit functions?	**Unclear** – Most effective if all CAs are monitoring CAA records and complying. How will CAA be enforced (depends on business rules adopted). Audited? Vulnerable to DNS attacks – best with DNSSEC	**Unclear** – Pinning failures (warning to users) must be reported to someone to detect mis-issuance of certs or incorrect pinning for valid cert
Overall burden of required system changes	**Major** – CAs must reprogram, change flow for cert issuance, CT logs must be created, monitors and auditors must be created, domain owners must build and maintain lists of their valid certs	**Minor** – domain owners must modify DNS records for protected domains, CAs must consult DNS record before issuing certs, contact customer if not listed (works best with DNSSEC, not widely deployed)	**Moderate** – domain owners must pin all valid certs to website, continuously update

THANK YOU!

Audience questions and

comments?

For more info: check CA Security Council
www.CAsecurity.org