Everything you always wanted to know about Certificate Transparency

(but were afraid to ask)

SBA Research

Martin who?

$whoami:

- member c^{3}wien
- online privacy, network security & digital forensics
- researcher at SBA Research
- @Fr333k

Agenda

State of the Foo

The Big Picture

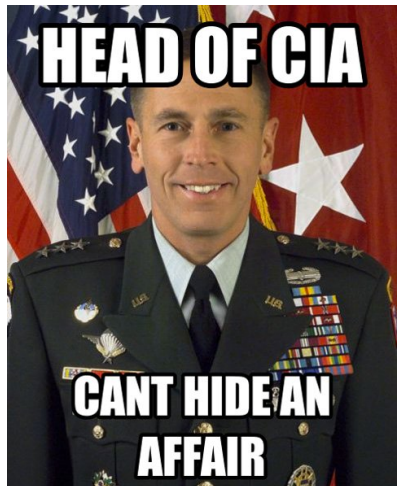
Under the hood

Show me the data!

Disclaimer

Keep in mind:

- this is a serious topic
- memes & cat pics are just means to an end
- people literally depend on decent HTTPS



DigiNotar

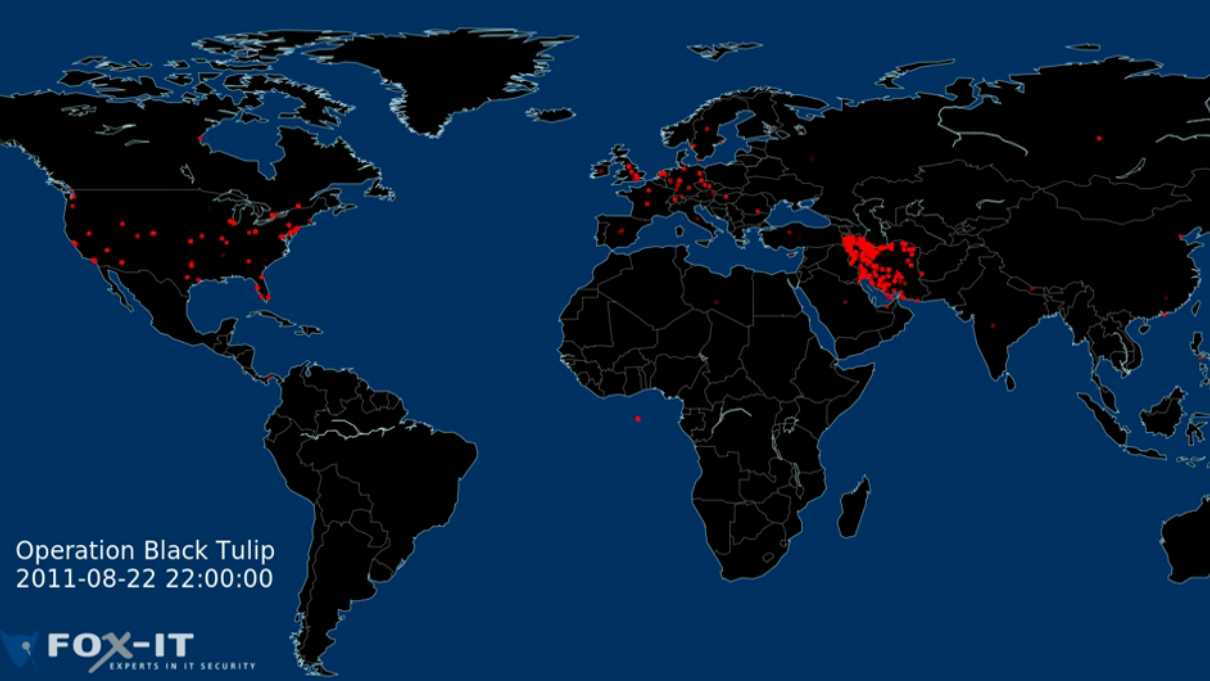
It all started with a hack:

- July 10th-20th, 2011
- CA DigiNotar pwned
- 531 fraudulent certificates issued
- among them: *.google.com, *.windowsupdate.com, *.mozilla.com, *.torproject.org ...

DigiNotar

Who got MitM'd:

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


Operation Black Tulip
2011-08-22 22:00:00
FOX-IT
EXPERTS IN IT SECURITY


VULNERABILITIES
VULNERABILITIES EVERYWHERE
imgflip.com

DigiNotar

Lessons learned (for CAs):

- patch your software!
- use antivirus!
- strong passwords for admin accounts!
- not all eggs in one basket (domain)
- report incidents

Other Incidents

Fraudulent CAs:

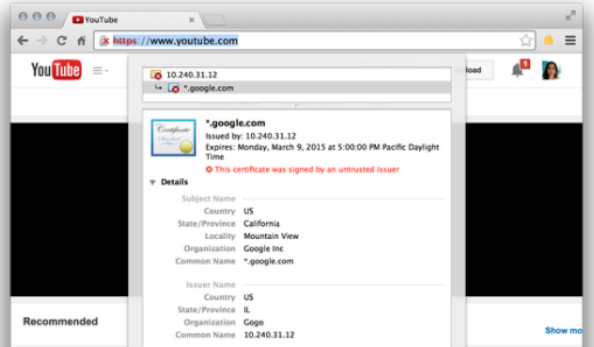
- Trustwave 2011: sub-CA for introspection
- Lenovo Superfish 2015: local MitM-CA
- CNNIC 2015: sub-CA for introspection
- Symantec 2016: test certificates (with CT)
- …

Gogo inflight wifi

Adrienne Porter Felt 
@__apf__

Follow

hey @Gogo, why are you issuing *.google.com certificates on your planes?



More Problems

(Some) weaknesses of TLS:

- certificate revocation is tricky
- all CAs for all CommonNames
- 1800+ CAs and sub-CAs (in 2013) [1]
- 1/3 never used for issuing HTTPS certificate (in 2014) [2]

More Problems

Implementation issues:

- different trust stores per OS/browser
- low entropy during key generation
- “goto fail;”

Deployment issues:

- SSLv2, SHA-1, CipherSuites, …
- STARTTLS, no PFS, …

More Problems

Implementation issues:

- different trust stores per OS/browser
- low entropy during key generation
- “goto fail;”

Deployment issues:

- SSLv2, SHA-1, CipherSuites, …
- STARTTLS, no PFS, …

More Problems

More Problems


CERTIFICATE TRANSPARENCY
TO THE RESCUE
memecrunch.com

Basic Idea of CT

Wouldn't it be nice, if ...

- CAs would publish all their business?
- problems could be detected upon issuance?
- there was punishment for misbehaving CAs?

Google is like ...

Why Google?

Uniquely positioned:

- control over client-run software
- pinned their certs
- $>$ 50% market share
- also, common target

Solution

RFC 6962:

- public, append-only cert logging
- cryptographically assured
- open for all

Goals:

- detect misbehaving CAs
- quickly identify fraudulent certs

Solution

RFC 6962:

- public, append-only cert logging
- cryptographically assured
- open for all

Goals:

- detect misbehaving CAs
- quickly identify fraudulent certs

RFC 6962

CT Entities:

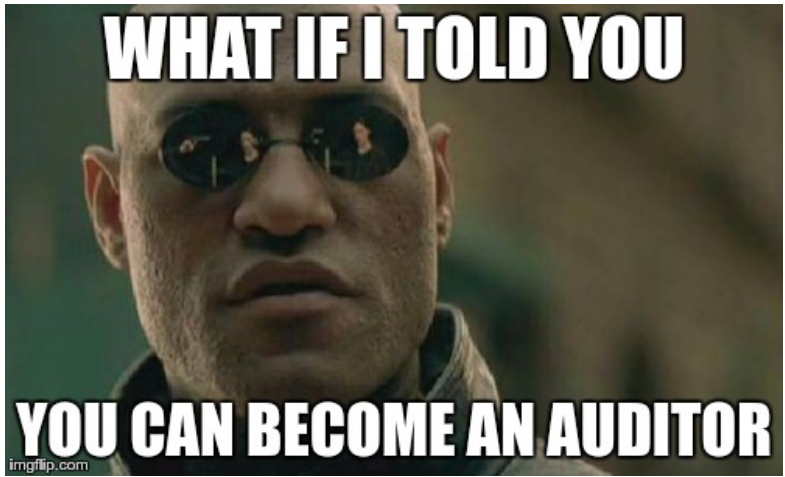
- *Logs*: collect certificates
- *Monitors*: identify suspicious certs
- *Auditors*: identify misbehaving logs

RFC 6962

Monitors:

- periodically fetch all logged certs
- look for suspicios certs or permissions
- e.g. sub-CAs, submitted cert not visible, …
- most commonly CAs
- also, identify misbehaving log operators

RFC 6962



RFC 6962

Auditors:

- verify log integrity
- e.g. no old certs removed, back-dated certs, ...
- query logs with signed cert timestamp (SCT)
- verify log proofs
- most commonly browsers

RFC 6962

CT openess:

- anyone can run any software
- e.g. CA can run all three:
  - log, monitor & auditor
  - also for other CAs and logs
- ideally, all gossip with each other
- difficulty: mostly availability, and log size

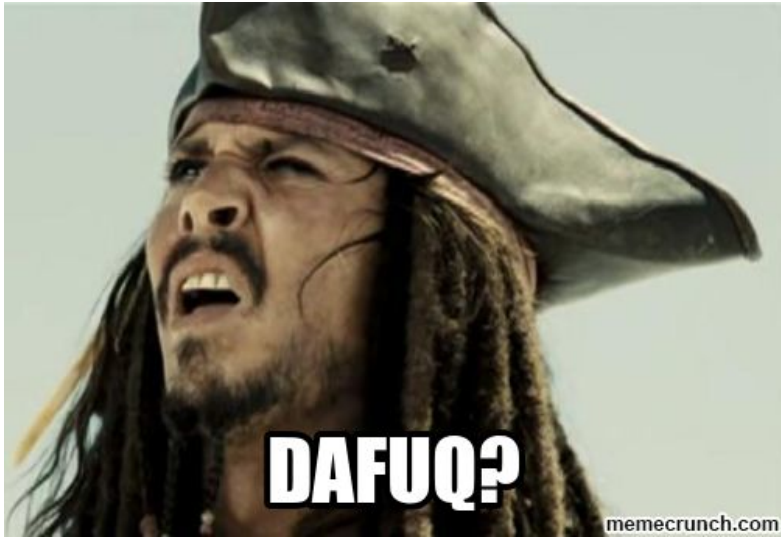
Under the hood

Under the Hood

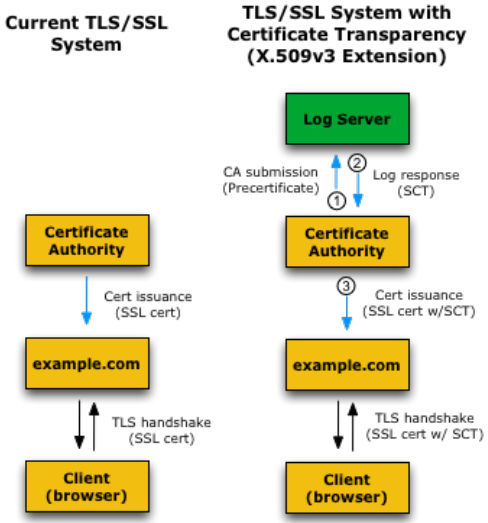
But how does it work?

- CAs send (pre-)cert to log
- immediately get a signed SCT back
- log promises to add the cert
- servers deliver SCT with cert

Under the Hood



Under the Hood



Under the Hood

Signed Certificate Timestamp (SCT):

- contains timestamp and cert
- LogID
- signed by log
- 3 methods available for clients

non-exclusive, a cert may have multiple SCT

Under the Hood

X.509v3 extensions:

- send pre-certificate to log
- get SCT valid for cert
- obtain certificate from CA
- SCT is part of certificate
- works on all current servers

Under the Hood

OCSP stapling:

- part of the OCSP information

OR as part of the TLS handshake:

- as TLS extension
- part of the ClientHello

Merkle Tree

Merkle Hash Tree:

- foundation for CT logs
- binary tree
- hash of a node depends on all children
- CT uses SHA-256

More funky terms:

- *maximum merge delay* usually 24h
- *signed tree head* (STH)

Merkle Tree

Merkle Hash Tree:

- foundation for CT logs
- binary tree
- hash of a node depends on all children
- CT uses SHA-256

More funky terms:

- *maximum merge delay* usually 24h
- *signed tree head* (STH)

Merkle Tree

Why MHT:

- order of included elements important
- signed tree hash == all elements
- proofs are hashes of inner nodes
- number is small

not possible to unnoticeably:

- back-date elements
- remove elements
- add elements

Merkle Tree

Why MHT:

- order of included elements important
- signed tree hash == all elements
- proofs are hashes of inner nodes
- number is small

not possible to unnoticeably:

- back-date elements
- remove elements
- add elements

Merkle Tree

Growing trees:

- logs add new certs e.g. every hour
- build a separate tree
- merge it with main tree
- all previous elements still there
- minimal hashes needed for verification

Merkle Tree

Merkle Consistency Proof:

- a.k.a “Hey log, u be cheating?”
- get STH, certs, inner nodes
- have: old STH
- verify STH and signature

Merkle Tree

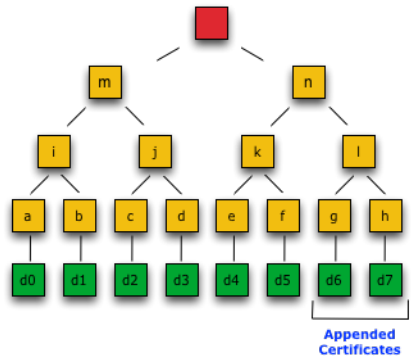


Figure 3

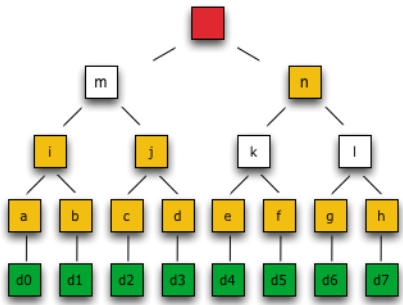


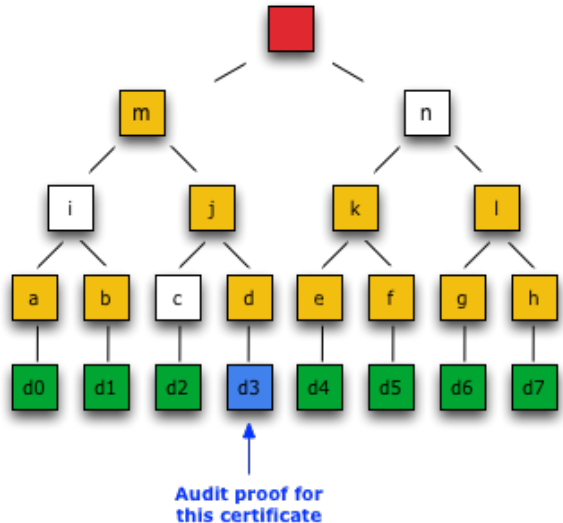
Figure 4

Merkle Tree

Audit proofs:

- specific certificate is in the log?
- no need to obtain all certificates
- only (few) inner node hash values
- reconstruct STH

Merkle Tree


m
n
i
j
k
l
a
b
c
d
e
f
g
h
d0
d1
d2
d3
d4
d5
d6
d7
Audit proof for
this certificate

Merkle Tree

Ever-growing logs:

- nothing is forever
- need to rotate the logs
- old logs get “frozen”
- e.g. aviator, 46M certs
- needs to remain online until last cert expires

Gossip

Information exchange:

- logs should chatter
- exchange STH
- detect malicious logs
- split-world attack e.g. gouvernments
- piggybacked in handshake [3]

Gossip

Who is logging

Who runs the logs:

- Google: 5 logs
  - 3 open for all
  - 1 let's encrypt
  - 1 non-let's encrypt
- Digicert: among the first
- Symantec, WoSign, CNNIC: caught cheating
- some smaller ones

In Browsers

Support by browsers:

- Google mandates 2 SCT for EV certs
- also checks it chrome://net-internals
- Firefox will gradually include

Does it work?

Symantec incident:

- issued google.com
- 23 test certificates
- CT logs had another 164 certs
- another 2.5k certs for non-existing domains

Downsides

Privacy:

- people can learn your internal hosts
- great for reconaissance!
- popular: Let’s encrypt

Downsides

Log entries must contain entire chain up to root, thus:

- excludes self-signed
- exludes DANE

“… until some mechanism to control spam is found. The authors welcome suggestions.”

Show me the data!

Show me the Data

For all logs:

- https://URL/ct/v1/get-sth
- gives no. of certs, timestamp, root hash and signature

Show me the Data


```
{"tree_size":46466472,
"timestamp":1480512258330,
"sha256_root_hash":"LcGcZRsm+LGYmrlyC5LXhV1T6OD8iH5dNlb0sEJl9bA=",
"tree_head_signature":"BAMASDBGAiEA/M0Nvt77aNe
+9eYbKsv6rRpTzFTKa5CGqb56ea4hnt8CIQCJDE7pL6xgAewMd5i3G1lrBWgFooT2kd3+zliEz5Rw8w=="}
```

Show me the Data

Other proofs:

- https://URL/ct/v1/get-sth-consistency
- https://URL/ct/v1/get-proof-by-hash

Push certs:

- POST https://URL/ct/v1/add-chain

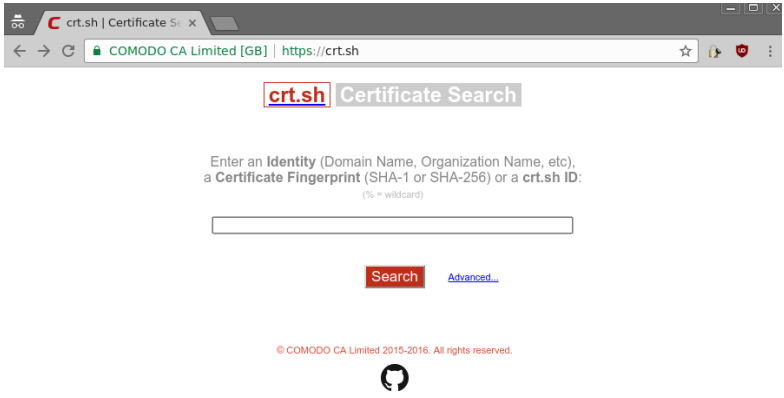
Show me the Data

Chrome net-internals:

```
t=4690 [st=148]        SIGNED_CERTIFICATE_TIMESTAMPS_RECEIVED
                       --> embedded_scts = "APEAdgDd6x0reg1PpiCLga2BaHB+Lo6dAdVciI09EcT
                       --> scts_from_ocsp_response = ""
                       --> scts_from_tls_extension = ""
t=4690 [st=148]        SIGNED_CERTIFICATE_TIMESTAMPS_CHECKED
                       --> scts = [{"extensions":"","hash_algorithm":"SHA-256","log_id"
t=4690 [st=148]        EV_CERT_CT_COMPLIANCE_CHECKED
                       --> certificate =
                             -----BEGIN CERTIFICATE-----
                             MIIGdDCCBVygAwIBAgIQat1vXCh8QJJNXRO5v+zigjANBgkqhkiG9w0BA
                             MQswCQYDVQQGEwJVUzEdMBsGA1UEChMUU3ltYW50ZWMgQ29ycG9yYXRpb
```

Show me the Data

https://crt.sh



Show me the Data

CT covers:

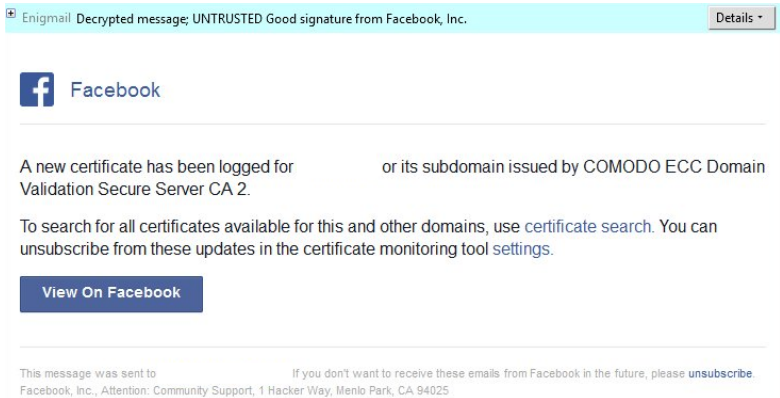
- 99.3% of Cloudflare,
- 100% of lets Ecnrypt
- paper by Halderman et al. [4]
- IPv4 scanning misses two thirds!
- CT misses GoDaddy, cPanel, Thawtec…

Show me the Data

Facebook Monitor:

- allows to monitor domains
- get email on cert update
- based on CT data

Facebook Notification



Future of CT

Whats next?

- moar logs and certs
- many moar auditors
- creating incentives for running it?

Future of CT

Far future:

- software releases?
- key management?
- alternatives to blockchains?

Generalize:

- “Verifiable Data Structures”
- *Trillian*

Future of CT

Far future:

- software releases?
- key management?
- alternatives to blockchains?

Generalize:

- “Verifiable Data Structures”
- *Trillian*

Future of CT

Thx for the attention!

[1] Zakir Durumeric, James Kasten, Michael Bailey, and J Alex Halderman.
Analysis of the https certificate ecosystem.
In *Proceedings of the 2013 conference on Internet measurement conference*, pages 291–304. ACM, 2013.

[2] Henning Perl, Sascha Fahl, and Matthew Smith.
You won't be needing these any more: On removing unused certificates from trust stores.
In *International Conference on Financial Cryptography and Data Security*, pages 307–315. Springer, 2014.

[3] Laurent Chuat, Pawel Szalachowski, Adrian Perrig, Ben Laurie, and Eran Messeri.

Efficient gossip protocols for verifying the consistency of certificate logs.
In *Communications and Network Security (CNS), 2015 IEEE Conference on*, pages 415–423. IEEE, 2015.

[4] Benjamin VanderSloot, Johanna Amann, Matthew Bernhard, Zakir Durumeric, Michael Bailey, and J Alex Halderman.
Towards a complete view of the certificate ecosystem.
In *Proceedings of the 2016 ACM on Internet Measurement Conference*, pages 543–549. ACM, 2016.