Poster: First Results in Analyzing the Certificate Transparency Ecosystem

Karoline Busse
Usable Security and Privacy Group
University of Bonn
busse@cs.uni-bonn.de

Christian Tiefenau
Usable Security and Privacy Group
University of Bonn
tiefenau@cs.uni-bonn.de

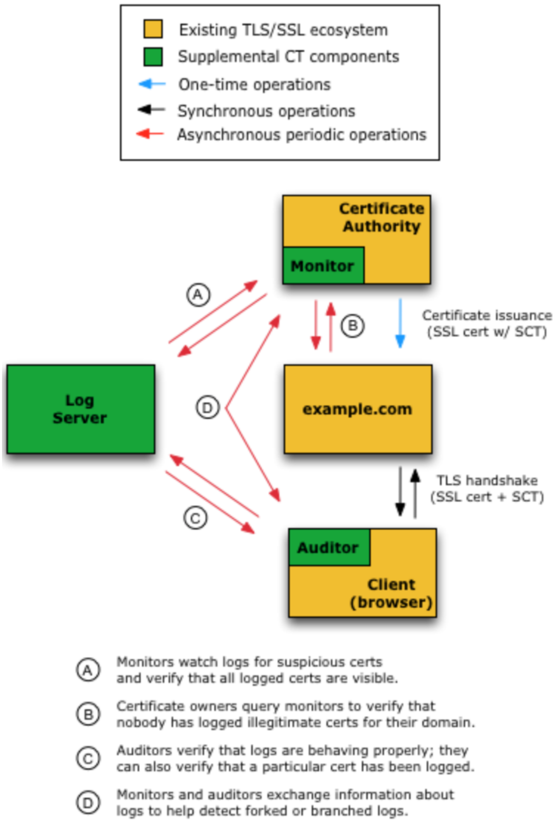
Matthew Smith
Usable Security and Privacy Group
University of Bonn
smith@cs.uni-bonn.de

***Abstract*—SSL is a widely used technology to secure connections all over the world. But it faces one main weakness: Potentially corrupt or rogue Certificate Authorities that illegitimately issue certificates for arbitrary domains to enable Man-in-the-Middle-Attacks for associated criminals or mass-surveying governments. Certificate Transparency has been implemented and deployed as a technology that aims to fix this weakness of the SSL ecosystem. By publicly logging and monitoring all issued SSL certificates, Certificate Transparency aims to provide documentation of certificate issuing and misuse, therefore enabling administrators to quickly identify misbehaving CAs. Until now, CT data has been published and collected in text-only format which lacks clear indicators of "bad" behavior. Our contribution is a scalable web-interface on top of a CT monitor that provides useful visualization and new insights into the certificate data collected by all public CT logs.**

I. Introduction

With Edward Snowden's revelations about public mass surveillance in 2013, the need to communicate privately gets more and more important. When surfing the web, connections are secured using SSL/TLS [1] which relies on X.509 certificates that are issued by trusted third parties, so called Certificate Authorities (CA). In reality, these CAs pose the biggest threat to secure SSL/TLS connections, because when controlling a certificate issuer, an adversary is able to obtain technically legitimate certificates for domains they do not own. Since those certificates are usually accepted by all major web browsers, the forged certificates allow for subtle Man-in-the-Middle-Attacks (MitMAs) on a large scale. There are scenarios where this technology for example can be used to hijack secure connections through certificates that are issued of by intelligence agencies [**?**].

The Certificate Transparency (CT) technology makes those attacks visible and greatly reduces the time in which such a misissued certificate can be used unnoticed. During the issuance process, a certificate must be submitted to at least one publicly auditable, append-only Log server. When establishing a secure connection, the server is obliged to provide a list of Logs in which the certificate is published, so the client can check its validity. In addition, domain owners can run Monitors that periodically query selected Logs for illegitimately issued certificates for their sites [2]. Figure 1 illustrates this process.



🟨 Existing TLS/SSL ecosystem
🟩 Supplemental CT components
← One-time operations
← Synchronous operations
← Asynchronous periodic operations
Certificate Authority
Monitor
Log Server
example.com
Auditor
Client (browser)
A
B
C
D
Certificate issuance (SSL cert w/ SCT)
TLS handshake (SSL cert + SCT)
A Monitors watch logs for suspicious certs and verify that all logged certs are visible.
B Certificate owners query monitors to verify that nobody has logged illegitimate certs for their domain.
C Auditors verify that logs are behaving properly; they can also verify that a particular cert has been logged.
D Monitors and auditors exchange information about logs to help detect forked or branched logs.

Fig. 1. CT Structure [**?**]

Until now, 10 logs have been set up across 7 organizations, collecting a total of 13245364 certificates of which 6385173 are currently valid. Given the data provided by Censys[1] that includes all currently active certificates, this corresponds to $14.44\ \%$ of all certificates used for HTTPS connections in the IPv4 part of the internet. This huge amount of data promises to hold many interesting insights about the SSL ecosystem, especially about suspicious behavior that could indicate the compromise of a CA. Current query tools like `crt.sh`[2] provide simple search access to Log data, but lack a proactive presentation of interesting behavior. We implement a so called CT Observatory that consists of a fully-featured query interface for all known CT logs and provides visualizations that can help domain owners, Certificate Authorities and interested researchers with new insights into the Certificate Transparency ecosystem.

[1] https://www.censys.io

[2] https://github.com/crtsh

After briefly presenting the technology behind the CT Observatory in section II, we report our findings in section III and give a look on the next steps of development in section IV.

II. Technology

The CT Observatory consists of a Postgres database derived from the `crt.sh` project which contains CT Log data. All logs listed on the official Certificate Transparency website [3] are periodically queued for updates. The web application backend is based on Django, a widespread web framework written in Python [4]. Visualizations on the frontend are produced with d3.js, bootstrap and jQuery. The Observatory is intended to help administrators and researchers in accessing the rapidly growing amount of Log data. The Observatory will be made accessible to the general public in the near future, together with its source code.

III. Findings

A. Log Coverage

CT encourages the submission of certificates to multiple Logs, since individual Logs can become inoperative without further notice. In addition, clients can blacklist individual logs, so when CT is considered mandatory for SSL/TLS connections, including a single certificate in multiple logs is highly encouraged.

Our analysis shows that 911640 certificates from the total of 13245364 certificates are listed in only one log, which equals 6.89 %. 17.64 percent of the certificates are incorporated in two logs, while 75.47 percent are included in more than two logs. Table I illustrates these numbers. The single biggest Log is Google's Pilot Log, which incorporates 12594447 certificates in total and 6 % of certificates which aren't listed in multiple Logs.

Occurences in logs	Total	Percentage
1	911640	6.89
2	2335147	17.64
>2	9988612	75.47

TABLE I. Certificate distribution in logs

B. Signature Algorithms

The signature algorithms used in submitted certificates are not as diverse as the key sizes. Table II shows that Certificates either use RSA (11061364, which equals 83.51 percent of all and 85.80 percent of valid certificates), or EC (2183634, that represents 16.49 % and currently 14.20 % of all active certificates). The deprecated DSA algorithm is still used in 0.004 percent of all valid certificates (261). These would be a perfect candidate for getting informed through our CT observatory.

	All certificates		Currently valid	
Signature Algorithm	Total	Percentage	Total	Percentage
RSA	11061364	83.51	5478262	85.80
EC	2183634	16.49	906453	14.20
DSA	366	0.003	261	0.004

TABLE II. Signature Algorithm Distribution

C. Key Length Distribution

The key length parameter is another interesting subject for closer inspection when looking at certificates that are using RSA as the signature algorithm. Currently 99.96% of all active certificates in the logs are using a key length of at least 2048 bit. Only 1 still valid certificate is using use a key length of 512 bits. The widely encouraged key size of 4096 or more Bit is still a minority with 346302 of all valid certificates. Besides there are some certificates that are using key lengths other than in the common power-of-two-pattern. Table III illustrates these numbers.

	All certificates		Currently valid	
Key size (bit)	Total	Percentage	Total	Percentage
< 1024	1188	0.01	1	0.00
< 2048	185165	1.67	2146	0.04
< 4096	10335015	93.43	5129813	93.64
> 4096	539996	4.88	346302	6.32

TABLE III. Key Length Distribution using RSA

IV. Conclusion and Future Work

The presented analyses represent our first steps toward an internet-wide observatory of the SSL ecosystem through the use of Certificate Transparency infrastructure. With the recent launch of the Let's Encrypt CA that issues SSL certificates automatically, a significant increase in issued and logged certificates is expected.

Regarding future work, a major task for the CT Observatory is to detect suspicious behavior in the Log data. This can include:

- Switching a CA
- Key change
- Same certificate except interesting new extensions / interesting values in existing extensions
- First wildcard certificate for a domain
- Sudden weakening: New certificate using weaker crypto than existing ones

In addition, we want to provide an in-depth analysis of historical certificate data for researchers and other interested entities. The extension of a notification for site owners in case of suspicious behavior is also considered.

References

[1] T. Dierks and E. Rescorla, "RFC 5246: The Transport Layer Security (TLS) Protocol Version 1.2," 2008.

[2] B. Laurie, A. Langley, and E. Kasper, "RFC 6962: Certificate Transparency," 2013.

[3] Google Sites. Certificate transparency. [Online]. Available: https://www.certificate-transparency.org

[4] Django Software Foundation. Django – The Web Framework for Perfectionists with Deadlines. [Online]. Available: https://djangoproject.com/