A Survey on Solutions to Enhance the Chain of Trust and Security and Privacy Considerations in Certificate Transparency Logs

EN.601.644 Network Security

Kandarp Khandwala (kkhandw1)

Johns Hopkins University - Information Security Institute

December 13, 2018

Contents

1 Problems With Certificate Authorities (CA)

1.1 Misbehavior By CAs

Thanks to modern cryptography, browsers can usually detect malicious websites that are provisioned with forged or fake Transport Layer Security (TLS) certificates. However, current cryptographic mechanisms aren't so good at detecting malicious websites if they're provisioned with mistakenly issued certificates or certificates that have been issued by a certificate authority (CA) that's been compromised or gone rogue. [4]

1.2 Subordinate Root Certificates

Some CAs have also admitted to issuing subordinate root certificates. These are used to monitor traffic on their internal network. Subordinate root certificates can be used to create TLS certificates for nearly any domain on the Internet. [6]

1.3 MITM - DigiNotar

In one case, a prominent Dutch CA (DigiNotar) was compromised and the hackers were able to use the CA's system to issue fake TLS certificates. The certificates were used to impersonate numerous sites in Iran, such as Gmail and Facebook, which enabled the operators of the fake sites to spy on unsuspecting site users. [5]

1.4 Far Reaching Consequences

In many cases, mistakenly issued certificates have been used by hackers for malicious attacks that have dire consequences, but the fallout after mitigation can be far ranging and harmful, too. The revocation and closure can cause a ripple effect as people can be denied access to government and private sites that are provisioned with the CA's TLS certificates.

2 Alternate Solutions

2.1 Key Pinning

Key Pinning associates a specific cryptographic public key with a certain web server. The public key is wrapped into a X.509 certificate which circumvents Man in the Middle (MITM) by telling the client which public key belongs to a certain web server. It is easy to setup since it only requires configuring an HTTP header. However, there are many problems with Key Pinning and in recent times it has been discounted by a lot of users. Firstly, it relies on Trust on First Use (TOFU). Secondly, non web-browser HTTP clients have a hard time configuring it. And lastly, any form of misconfiguration can lead to Denial of Service (DoS) on itself.

2.2 Attack Resilient Public-Key Infrastructure (ARPKI)

In ARPKI, the process of certificate issuance, update, revocation, and validation, are transparent and accountable. This method has resilience against impersonation attacks that involve $n - 1$ compromised entities. It can also detect domain compromise and take action against it. [1]

2.3 Certification Authority Authorization (CAA) Records

CAA Records allow domain owners to declare which certificate authorities are allowed to issue a certificate for a domain. This whitelist is maintained by domain owners and they are in control of the configuration. While this is a good solution, it has found limited support from DNS providers and hence is not widely used.

3 Certificate Transparency (CT) Logs

In recent times, Certificate Transparency Logs have become the primary data source for monitoring the public key infrastructure. It contains public, cryptographically verifiable ledgers of all browser-trusted certificates in their logs which can be viewed by anyone. These log maintainers log certificates at the time of issuance. Eventually, major browsers will only trust a certicate if it comes with a proof that it has been recorded in a public log. A certificate without such a proof will be treated as invalid. This will effectively force all CAs to register every issued certicate with one or more CT logs thus completing the circle of trust.

4 Privacy Challenges With Certificate Transparency

While CT provides a strong defense against misissuance, several privacy challenges are not addressed by the current design and may hinder wide adoption. One big problem with CT is that it is currently incompatible with private subdomains. Consider an enterprise that does not want to reveal the domains of its internal servers to the public. However, the enterprise wishes to use a public CA to issue certificates for its internal subdomains (or to log its privately issued certificates in a public log). Because domain names are publicly available in the CT log, logging certificates will reveal the servers' private domains. [3]

5 Problems with Wildcards and Redaction

There are several solutions to this privacy problem. Using wildcards in CT logs is one; while redacting the subdomain is another. Redaction of domain name labels as well as the use of wildcards carry risks. If the entirety of the domain space below the unredacted part of a domain name is not registered by a single domain owner, then the domain name may be considered by clients to be overly redacted. CAs should take care to avoid overly redacting domain names in precertificates. In addition, the CT ecosystem would be harmed if the use

of redaction becomes too widespread. [7]

6 Private Subdomain Redaction Via Commitments

6.1 Commitment Schemes

A commitment scheme is a cryptographic primitive that allows one to commit to a chosen value (or chosen statement) while keeping it hidden to others, with the ability to reveal the committed value later. Commitment schemes are designed so that a party cannot change the value or statement after they have committed to it: that is, commitment schemes have hiding and binding. [2]

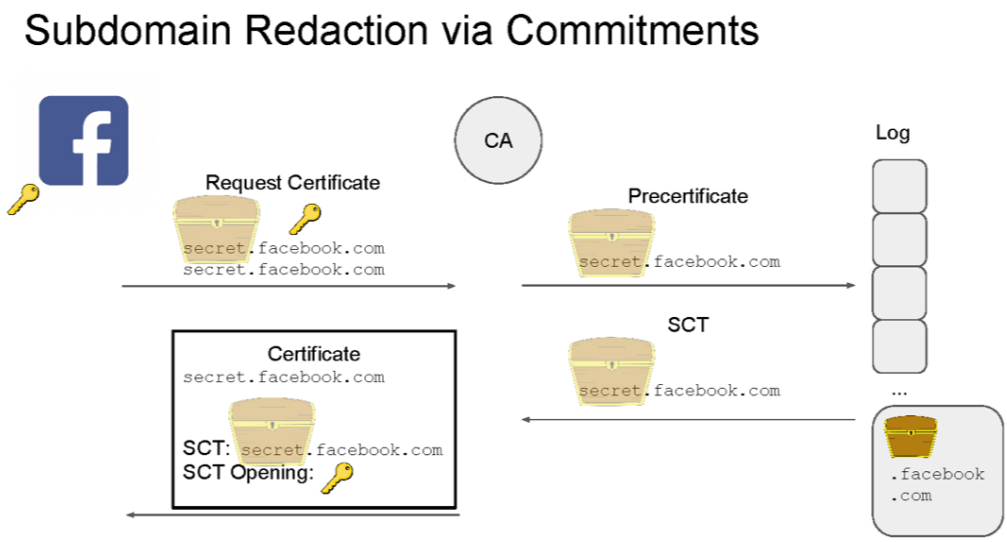

Figure 1: Private Subdomain Redaction via Commitments

6.2 Domain Owner Issuing Pre-Certificate

The domain owner D generates a commitment C_d to subdomain d (e.g. for the domain name $secret.facebook.com$, we have $D = facebook.com$ and $d = secret$) with decommitment randomness r. The domain owner sends (d, D, C_d, r) to the CA. The CA checks that C_d is

computed correctly and passes (D, C_d) to the log in the signed pre-certificate.

6.3 User Visiting The Domain

Any visitor to the site (or any auditor/monitor) is given the certificate and the Signed Certificate Timestamp (SCT). The visitor verifies that the commitment C_d in the SCT is in fact a commitment to d with decommitment randomness r. Monitors who audit the logs can check that the correct number of example.com certicates are present on the logs and that no spurious certicates have been issued.

7 Conclusion

7.1 Complete Domain Hiding

This technique achieves private subdomains in CT. Is it also possible to efficiently make entire domain names private? Such a scheme would be useful in practice for enterprises who wish to register domains for new projects before announcing them publicly.

7.2 Modification Pain

The implementation of this scheme, while trivial involves modifications to the CA and the CT logs which might slow the adoption rate of this change.

7.3 Better Web of Trust/Blockchain

There is absolutely no doubt that the current PKI needs work and while there are many ways to implement public verifiability using CT Logs or a decentralized implementation like Blockchain, there must be serious effort to do so while also maintaining the security and privacy of the domain owners as well as the user.

References

[1] David Basin et al. "ARPKI: Attack resilient public-key infrastructure". In: *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. ACM. 2014, pp. 382–393.

[2] Gilles Brassard, David Chaum, and Claude Crépeau. "Minimum disclosure proofs of knowledge". In: *Journal of Computer and System Sciences* 37.2 (1988), pp. 156–189.

[3] Saba Eskandarian et al. "Certificate transparency with privacy". In: *Proceedings on Privacy Enhancing Technologies* 2017.4 (2017), pp. 329–344.

[4] Certificate Transparency Group. *What is Certificate Transparency?* URL: `https://www.certificate-transparency.org/what-is-ct`.

[5] Hans Hoogstraaten. *Black Tulip Report of the investigation into the DigiNotar Certificate Authority breach*. URL: `https://www.researchgate.net/publication/269333601_Black_Tulip_Report_of_the_investigation_into_the_DigiNotar_Certificate_Authority_breach/download`.

[6] Global Sign. *What Are Subordinate CAs and Why Would You Want Your Own?* URL: `https://www.globalsign.com/en/blog/what-is-an-intermediate-or-subordinate-certificate-authority/`.

[7] R. Stradling and E. Messeri. "Certificate Transparency: Domain Label Redaction". In: *Internet Draft* 2017 ().