TRANS (Public Notary Transparency)
Internet-Draft
Intended status: Experimental
Expires: July 21, 2017

R. Stradling
Comodo CA, Ltd.
E. Messeri
Google UK Ltd.

January 17, 2017

Certificate Transparency: Domain Label Redaction
draft-strad-trans-redaction-01

Abstract

This document defines mechanisms to allow DNS domain name labels that are considered to be private to not appear in public Certificate Transparency (CT) logs, while still retaining most of the security benefits that accrue from using Certificate Transparency.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on July 21, 2017.

Table of Contents

1. Introduction

Some domain owners regard certain DNS domain name labels within their registered domain space as private and security sensitive. Even though these domains are often only accessible within the domain owner's private network, it's common for them to be secured using publicly trusted Transport Layer Security (TLS) server certificates.

Certificate Transparency v1 [RFC6962] and v2 [I-D.ietf-trans-rfc6962-bis] describe protocols for publicly logging the existence of TLS server certificates as they are issued or observed. Since each TLS server certificate lists the domain names that it is intended to secure, private domain name labels within registered domain space could end up appearing in CT logs, especially as TLS clients develop policies that mandate CT compliance. This seems like an unfortunate and potentially unnecessary privacy leak, because it's the registered domain names in each certificate that are of primary interest when using CT to look for suspect certificates.

TODO: Highlight better the differences between registered domains and

subdomains, referencing the relevant DNS RFCs.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. Redaction Mechanisms

We propose three mechanisms, in increasing order of implementation complexity, to allow certain DNS domain name labels to not appear in public CT logs:

- Using wildcard certificates (Section 3.1) is the simplest option, but it only covers certain use cases.
- Logging a name-constrained intermediate CA certificate in place of the end-entity certificate (Section 3.2) covers more, but not all, use cases.
- Therefore, we define a domain label redaction mechanism (Section 3.3) that covers all use cases, at the cost of considerably increased implementation complexity.

We anticipate that TLS clients may develop policies that impose additional compliancy requirements on the use of the Section 3.2 and Section 3.3 mechanisms.

To ensure effective redaction, CAs and domain owners should note the privacy considerations (Section 5).

TODO(eranm): Do we need to further expand (either here or in the following subsections) on when each of the mechanisms is/isn't suitable?

TODO: Previously, these mechanisms were defined in earlier revisions of CTv2 [I-D.ietf-trans-rfc6962-bis], and nothing was said about

compatibility with CTv1. But now, given that these mechanisms have been decoupled from [I-D.ietf-trans-rfc6962-bis], and given that at least one major TLS client has announced a policy of mandatory CT compliance that will almost certainly take effect before CTv2 is widely deployed, we should consider making some or all of these mechnanisms compatible with both CTv1 and CTv2.

3.1. Using Wildcard Certificates

A certificate containing a DNS-ID [RFC6125] of "*.example.com" could be used to secure the domain "topsecret.example.com", without revealing the label "topsecret" publicly.

Since TLS clients only match the wildcard character to the complete leftmost label of the DNS domain name (see Section 6.4.3 of [RFC6125]), a different mechanism is needed when any label other than the leftmost label in a DNS-ID is considered private (e.g., "top.secret.example.com"). Also, wildcard certificates are prohibited in some cases, such as Extended Validation Certificates [EV.Certificate.Guidelines].

3.2. Using a Name-Constrained Intermediate CA

An intermediate CA certificate or intermediate CA precertificate that contains the Name Constraints [RFC5280] extension MAY be logged in place of end-entity certificates issued by that intermediate CA, as long as all of the following conditions are met:

- there MUST be a non-critical extension (OID 1.3.101.76, whose extnValue OCTET STRING contains ASN.1 NULL data (0x05 0x00)). This extension is an explicit indication that it is acceptable to not log certificates issued by this intermediate CA.

- there MUST be a Name Constraints extension, in which:

  * permittedSubtrees MUST specify one or more dNSNames.

  * excludedSubtrees MUST specify the entire IPv4 and IPv6 address ranges.

Below is an example Name Constraints extension that meets these conditions:

```
      SEQUENCE {
        OBJECT IDENTIFIER '2 5 29 30'
        BOOLEAN TRUE
        OCTET STRING, encapsulates {
          SEQUENCE {
            [0] {
              SEQUENCE {
                [2] 'example.com'
                }
              }
            [1] {
              SEQUENCE {
                [7] 00 00 00 00 00 00 00 00
                }
              SEQUENCE {
                [7]
                  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
                }
              }
            }
          }
```

}

3.2.1. Presenting SCTs, Inclusion Proofs and STHs

Each SCT (and optional corresponding inclusion proof and STH) presented by a TLS server MAY correspond to an intermediate CA certificate or intermediate CA precertificate (to which the server certificate chains) that meets the requirements in Section 3.2. This extends section TBD of CT v2 [I-D.ietf-trans-rfc6962-bis], which specifies that each SCT always corresponds to the server certificate or to a precertificate that corresponds to that certificate.

Each SCT (and optional corresponding inclusion proof and STH) included by a certification authority in a Transparency Information X.509v3 extension in the "singleExtensions" of a "SingleResponse" in an OCSP response MAY correspond to an intermediate CA certificate or intermediate CA precertificate (to which the certificate identified by the "certID" of that "SingleResponse" chains) that meets the requirements in Section 3.2. This extends section TBD of CT v2 [I-D.ietf-trans-rfc6962-bis], which specifies that each SCT always corresponds to the certificate identified by the "certID" of that "SingleResponse" or to a precertificate that corresponds to that certificate.

Each SCT (and optional corresponding inclusion proof and STH) included by a certification authority in a Transparency Information

X.509v3 extension in a certificate MAY correspond to an intermediate CA certificate or intermediate CA precertificate (to which the certificate chains) that meets the requirements in Section 3.2. This extends section TBD of CT v2 [I-D.ietf-trans-rfc6962-bis], which specifies that each SCT always corresponds to a precertificate that corresponds to that certificate.

TODO: Refactor this section to avoid repetition.

3.2.2. Matching an SCT to the Correct Certificate

Before considering any SCT to be invalid, a TLS client MUST attempt to validate it against the server certificate and against each of the zero or more suitable name-constrained intermediates in the chain. These certificates may be evaluated in the order they appear in the

chain, or indeed, in any order.

TODO: Shall we specify that there MUST be no more than ONE name-constrained intermediate in the chain?

TODO: Shall we specify that all presented SCTs MUST correspond to the same (end-entity or name-constrained intermediate) certificate?

3.3. Redacting Labels in Precertificates

When creating a precertificate, the CA MAY include a redactedSubjectAltName (Section 3.3.1) extension that contains, in a redacted form, the same entries that will be included in the certificate's subjectAltName extension. When the redactedSubjectAltName extension is present in a precertificate, the subjectAltName extension MUST be omitted (even though it MUST be present in the corresponding certificate).

Wildcard "*" labels MUST NOT be redacted, but one or more non-wildcard labels in each DNS-ID [RFC6125] can each be replaced with a redacted label as follows:

```
REDACT(label) = prefix || BASE32(index || _label_hash)
  _label_hash = LABELHASH(keyid_len || keyid || label_len || label)
```

"label" is the case-sensitive label to be redacted.

"prefix" is the "?" character (ASCII value 63).

"index" is the 1 byte index of a hash function in the CT hash algorithm registry (section TBD of [I-D.ietf-trans-rfc6962-bis]). The value 255 is reserved.

"keyid_len" is the 1 byte length of the "keyid".

"keyid" is the keyIdentifier from the Subject Key Identifier extension (section 4.2.1.2 of [RFC5280]), excluding the ASN.1 OCTET STRING tag and length bytes.

"label_len" is the 1 byte length of the "label".

"||" denotes concatenation.

"BASE32" is the Base 32 Encoding function (section 6 of [RFC4648]). Pad characters MUST NOT be appended to the encoded data.

"LABELHASH" is the hash function identified by "index".

3.3.1. redactedSubjectAltName Certificate Extension

The redactedSubjectAltName extension is a non-critical extension (OID 1.3.101.77) that is identical in structure to the subjectAltName extension, except that DNS-IDs MAY contain redacted labels (Section 3.3).

When used, the redactedSubjectAltName extension MUST be present in both the precertificate and the corresponding certificate.

This extension informs TLS clients of the DNS-ID labels that were redacted and the degree of redaction, while minimizing the complexity of TBSCertificate reconstruction (Section 3.3.3). Hashing the redacted labels allows the legitimate domain owner to identify whether or not each redacted label correlates to a label they know of.

TODO: Consider the pros and cons of this 'un'redaction feature. If the cons outweigh the pros, switch to using Andrew Ayer's alternative proposal of hashing a random salt and including that salt in an extension in the certificate (and not including the salt in the precertificate).

Only DNS-ID labels can be redacted using this mechanism. However, CAs can use the Section 3.2 mechanism to allow DNS domain name labels in other subjectAltName entries to not appear in logs.

TODO: Should we support redaction of SRV-IDs and URI-IDs using this mechanism?

3.3.2. Verifying the redactedSubjectAltName extension

If the redactedSubjectAltName extension is present, TLS clients MUST check that the subjectAltName extension is present, that the subjectAltName extension contains the same number of entries as the redactedSubjectAltName extension, and that each entry in the subjectAltName extension has a matching entry at the same position in the redactedSubjectAltName extension. Two entries are matching if either:

- The two entries are identical; or
- Both entries are DNS-IDs, have the same number of labels, and each label in the subjectAltName entry has a matching label at the same position in the redactedSubjectAltName entry. Two labels are matching if either:
  - The two labels are identical; or,
  - Neither label is "*" and the label from the redactedSubjectAltName entry is equal to REDACT(label from subjectAltName entry) (Section 3.3).

If any of these checks fail, the certificate MUST NOT be considered compliant.

3.3.3. Reconstructing the TBSCertificate

Section TBD of [I-D.ietf-trans-rfc6962-bis] describes how TLS clients can reconstruct the TBSCertificate component of a precertificate from a certificate, so that associated SCTs may be verified.

If the redactedSubjectAltName extension (Section 3.3.1) is present in the certificate, TLS clients MUST also:

- Verify the redactedSubjectAltName extension against the subjectAltName extension according to Section 3.3.2.
- Once verified, remove the subjectAltName extension from the TBSCertificate.

4. Security Considerations

4.1. Avoiding Overly Redacted Domain Names

Redaction of domain name labels (Section 3.3) carries the same risks as the use of wildcards (e.g., section 7.2 of [RFC6125]). If the entirety of the domain space below the unredacted part of a domain

name is not registered by a single domain owner (e.g., REDACT(label).com, REDACT(label).co.uk and other [Public.Suffix.List] entries), 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. It is expected that monitors will treat precertificates that contain overly redacted domain names as potentially misissued. TLS clients MAY consider a certificate to be non-compliant if the reconstructed TBSCertificate (Section 3.3.3) contains any overly redacted domain names.

TODO(eranm): Describe how the CT ecosystem would be harmed if the use of redaction becomes too widespread.

5. Privacy Considerations

5.1. Ensuring Effective Redaction

Although the mechanisms described in this document remove the need for private labels to appear in CT logs, they do not guarantee that this will never happen. For example, anyone who encounters a certificate could choose to submit it to one or more logs, thereby rendering the redaction futile.

Domain owners are advised to take the following steps to minimize the likelihood that their private labels will become known outside their closed communities:

- Avoid registering private labels in public DNS.
- Avoid using private labels that are predictable (e.g., "www", labels consisting only of numerical digits, etc). If a label has insufficient entropy then redaction will only provide a thin layer of obfuscation, because it will be feasible to recover the label via a brute-force attack.
- Avoid using publicly trusted certificates to secure private domain space.
- Avoid enabling unrestricted access for DNS zone transfer (AXFR) requests (see section 5 of [RFC5936]).

CAs are advised to carefully consider each request to redact a label using the Section 3.3 mechanism. When a CA believes that redacting a particular label would be futile, we advise rejecting the redaction

request. TLS clients may have policies that forbid redaction, so

label redaction should only be used when it's absolutely necessary and likely to be effective.

6. Acknowledgements

The authors would like to thank Andrew Ayer and TBD for their valuable contributions.

A big thank you to Symantec for kindly donating the OIDs from the 1.3.101 arc that are used in this document.

7. References

7.1. Normative References

[I-D.ietf-trans-rfc6962-bis]
Laurie, B., Langley, A., Kasper, E., Messeri, E., and R. Stradling, "Certificate Transparency Version 2.0", draft-ietf-trans-rfc6962-bis-24 (work in progress), December 2016.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, DOI 10.17487/RFC4648, October 2006, <http://www.rfc-editor.org/info/rfc4648>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <http://www.rfc-editor.org/info/rfc5280>.

[RFC5936] Lewis, E. and A. Hoenes, Ed., "DNS Zone Transfer Protocol (AXFR)", RFC 5936, DOI 10.17487/RFC5936, June 2010, <http://www.rfc-editor.org/info/rfc5936>.

[RFC6125] Saint-Andre, P. and J. Hodges, "Representation and Verification of Domain-Based Application Service Identity within Internet Public Key Infrastructure Using X.509 (PKIX) Certificates in the Context of Transport Layer Security (TLS)", RFC 6125, DOI 10.17487/RFC6125, March 2011, <http://www.rfc-editor.org/info/rfc6125>.

[RFC6962] Laurie, B., Langley, A., and E. Kasper, "Certificate Transparency", RFC 6962, DOI 10.17487/RFC6962, June 2013, <http://www.rfc-editor.org/info/rfc6962>.

7.2. Informative References

[EV.Certificate.Guidelines]
CA/Browser Forum, "Guidelines For The Issuance And Management Of Extended Validation Certificates", 2007, <https://cabforum.org/wp-content/uploads/EV_Certificate_Guidelines.pdf>.

[Public.Suffix.List]
Mozilla Foundation, "Public Suffix List", 2016, <https://publicsuffix.org>.

Authors' Addresses

Rob Stradling
Comodo CA, Ltd.

Email: rob.stradling@comodo.com

Eran Messeri
Google UK Ltd.

Email: eranm@google.com