Use of Short-Term, Automatically-Renewed (STAR) Certificates to address the LURK problem
draft-sheffer-acme-star-lurk-00

Abstract

This memo proposes two mechanisms that work in concert to address the LURK problem statement, allowing a third party (e.g., a content delivery network) to terminate TLS sessions on behalf of a domain name owner (e.g., a content provider).

The proposed mechanisms are:

1. An extension to the ACME protocol to enable the issuance of short-term and automatically renewed certificates, and
2. A protocol that allows a domain name owner to delegate to a third party control over a certificate that bears its own name.

It should be noted that these are in fact independent building blocks that could be used separately to solve completely different problems.

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 April 28, 2017.
1. A Solution for the HTTPS CDN Use Case

A content provider, and Domain Name Owner (DNO), has agreements in place with one or more Content Delivery Networks (CDN) that are contracted to serve its content over HTTPS. The CDN terminates the HTTPS connection at one of its edge cache servers and needs to present its clients (browsers, set-top-boxes) a certificate whose name matches the authority of the URL that is requested, i.e. that of the DNO. However, many DNOs balk at sharing their long-term private keys with another organization and, equally, CDN providers would rather not have to handle other parties’ long-term secrets. This problem has been discussed at the IETF under the LURK (limited use of remote keys) title.

This document proposes a solution to the above problem that involves the use of short-term certificates with a DNO’s name on them, and a
scheme for handling the naming delegation from the DNO to the CDN. The generated short-term credentials are automatically renewed by an ACME Certification Authority (CA) [I-D.ietf-acme-acme] and routinely rotated by the CDN on its edge cache servers. The DNO can end the delegation at any time by simply instructing the CA to stop the automatic renewal and let the certificate expire shortly after.

Using short-term certificates makes revocation cheap and effective [Topalovic] [I-D.iab-web-pki-problems] in case of key compromise or of termination of the delegation; seamless certificate issuance and renewal enable the level of workflow automation that is expected in today’s cloud environments. Also, compared to other keyless-TLS solutions [I-D.cairns-tls-session-key-interface] [I-D.erb-lurk-rsalg], the proposed approach doesn’t suffer from scalability issues or increase in connection setup latency, while requiring virtually no changes to existing COTS caching software used by the CDN.

2. Conventions used in this document

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

TODO: glossary.

3. Protocol Flow

The protocol flow can be split into two: a LURK interface, used by CDN and DNO to agree on the name delegation, and the ACME STAR interface, used by DNO to obtain the short-term and automatically renewed certificate from the CA, which is eventually consumed by the CDN. The latter is also used to terminate the delegation, if so needed.

The following subsections describe the preconditions (Section 3.1), and the three main phases of the protocol:

- Bootstrap: the CDN requests from the DNO the delegation of a specific name and in turn DNO asks an ACME CA to create the corresponding short-term and auto-renewed (STAR) certificate (Section 3.2);
- Auto-renewal: the ACME CA periodically re-issues the short-term certificate and posts it to a public URL (Section 3.3);
- Termination: the DNO (indirectly) stops name delegation by explicitly requesting the ACME CA to discontinue the automatic renewal of the certificate (Section 3.4).
3.1. Preconditions

The protocol assumes the following preconditions are met:

- A mutually authenticated channel between CDN and DNO pre-exists. This is called "LURK channel" and all LURK protocol exchanges between CDN and DNO are run over it. It provides the guarantee that the LURK requests and responses are authentic [\[\_1: Note that, under this assumption, the key used to authenticate the CDN to the DNO becomes a critical asset for the security of the proposed protocol, and that certain interactions (e.g., CSR submission) might require a stronger authentication mechanism. For example, stacking a further authentication factor on top of CDN’s LURK key would allow to distinguish an attacker that has only managed to successfully attack the CDN’s LURK key from the legitimate CDN. --tf\]].
- CDN and DNO have agreed on a "CSR template" to use, including at a minimum:
  - Subject name (e.g., "somesite.DNO.com"),
  - Validity (e.g., 24 to 72 hours),
  - Requested algorithms,
  - Key length,
  - Key usage.

The CDN is required to use this template for every CSR created under the same delegation.
- DNO has registered through the ACME interface exposed by the Certificate Authority (CA) using the usual ACME registration procedure. The DNO shall, at the registration stage, query the ACME server for the supported STAR capabilities - for example: the minimum validity period of the issued certificate, the maximum duration of the automatic renewal process (either as a maximum number of renewal events, or as its maximum absolute life-span).

3.2. Bootstrap

CDN (LURK client) generates a key-pair, wraps it into a Certificate Signing Request (CSR) according to the agreed CSR template, and sends it to the DNO (LURK server) over the pre-established LURK channel. The DNO uses the CDN identity provided on the LURK template to look up the CSR template that applies to the requesting CDN and decides whether or not to accept the request. (TBD: This is probably a case that would require a further authentication stage over the one provided by the mutual-authenticated LURK channel?) Assuming everything is in order, it then "forwards" the CDN request to the ACME CA by means of the usual ACME application procedure.
Specifically, DNO, in its role as a ACME/STAR client, requests the CA a STAR certificate, i.e., one that:

- Has a short validity (e.g., 24 to 72 hours);
- Is automatically renewed by the CA for a certain period of time;
- Is downloadable from a (highly available) public link without requiring any special authorisation.

Other than that, the ACME protocol flows as normal between DNO and CA, in particular DNO is responsible for satisfying the requested ACME challenges until the CA is willing to issue the requested certificate. The DNO is given back a unique identifier for the issued STAR certificate to be used in subsequent interaction with the CA (e.g., if the certificate needs to be terminated.)

Concurrently, a 202 response has been sent back to the CDN with an endpoint to poll for completion of the certificate generation process.

The bootstrap phase ends when the DNO obtains the OK from the ACME CA and posts the certificate’s URL to the "completion endpoint" where the CDN can retrieve it. The information that is passed on to the CDN at this stage also includes details about how much time before the certificate expires can the CDN expect the replacement to be ready.
3.3. Refresh

The CA automatically re-issues the certificate (using the same CSR) before it expires and publishes it to the URL that the CDN has come to know at the end of the bootstrap phase. The CDN downloads and installs it. This process goes on until either:
3.4. Termination

DNO requests termination of the STAR certificate by including the previously obtained identifier in a STAR certificate termination request to the ACME interface. After CA receives and verifies the request, it shall:

- Cancel the automatic renewal process for the LURK certificate;
- Change the certificate publication resource to return an error indicating the termination of the delegation to external clients, including the CDN;

Note that it is not necessary to explicitly revoke the short-term certificate.
4. Security Considerations

- CDN’s client certificate key is first order security asset and MUST be protected. Absent 2FA/MFA, an attacker that can compromise the key might be able to obtain certificates bearing DNO’s identity.
- Consider collusion of two or more CDNs with contracts with the same DNO (?)

4.1. Restricting CDNs to the Delegation Mechanism

Currently there are no standard methods for the DNO to ensure that the CDN cannot issue a certificate through mechanisms other than the one described here, for the URLs under the CDN’s control. For example, regardless of the STAR solution, a rogue CDN employee can use the ACME protocol (or proprietary mechanisms used by various CAs) to create a fake certificate for the DNO’s content.

The best solution currently being worked on would consist of several related configuration steps:

- Make sure that the CDN cannot modify the DNS records for the domain. Typically this would mean that the content owner establishes a CNAME resource record from a subdomain into a CDN-managed domain.
o Restrict certificate issuance for the domain to specific CAs that comply with ACME. This assumes universal deployment of CAA [RFC6844] by CAs, which is not the case yet.

o Deploy ACME-specific methods to restrict issuance to a specific authorization key which is controlled by the content owner [I-D.landau-acme-caa], and/or to specific ACME authorization methods.

This solution is recommended in general, even if an alternative to the mechanism described here is used.

5. Acknowledgments

This work is partially supported by the European Commission under Horizon 2020 grant agreement no. 688421 Measurement and Architecture for a Middleboxed Internet (MAMI). This support does not imply endorsement.

6. References

6.1. Normative References

[I-D.ietf-acme-acme]


6.2. Informative References

[I-D.cairns-tls-session-key-interface]

[I-D.erb-lurk-rsalg]


Authors' Addresses

Yaron Sheffer
Intuit
EMail: yaronf.ietf@gmail.com

Diego Lopez
Telefonica I+D
EMail: diego@telefonica.es

Oscar Gonzalez de Dios
Telefonica I+D
EMail: oscar.gonzalezdedios@telefonica.com

Thomas Fossati
Nokia
EMail: thomas.fossati@nokia.com