Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 25, 2020

O. Friel
Cisco
R. Shekh-Yusef
Avaya
M. Richardson
Sandelman Software Works
October 23, 2019

# BRSKI Cloud Registrar
# draft-friel-anima-brski-cloud-01

Abstract

This document specifies the behaviour of a BRSKI Cloud Registrar, and how a pledge can interact with a BRSKI Cloud Registrar when bootstrapping.

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 https://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 25, 2020.

Table of Contents



## 1. Introduction

Bootstrapping Remote Secure Key Infrastructures (BRSKI) [I-D.ietf-anima-bootstrapping-keyinfra] specifies automated bootstrapping of an Autonomic Control Plane. BRSKI Section 2.7 describes how a pledge "MAY contact a well known URI of a cloud registrar if a local registrar cannot be discovered or if the pledge's target use cases do not include a local registrar".

This document further specifies use of a BRSKI cloud registrar and clarifies operations that are not sufficiently specified in BRSKI.

Two high level deployment models are documented here:

- Local Domain Registrar Discovery: the cloud registrar is used by the pledge to discover the local domain registrar. The cloud registrar redirects the pledge to the local domain registrar, and the pledge completes bootstrap against the local domain registrar.

- Cloud Registrar Based Boostrap: there is no local domain registrar and the pledge completes boostrap using the cloud registrar. As part of boostrap, the cloud registrar may need to tell the client the domain to use for accessing services.

These deployment models facilitate multiple use cases including:

- A pledge is bootstrapping in a remote location and needs to contact a cloud registrar in order to discover its local domain.

- A pledge supports multiple deployment models and needs to discover which deployment model is in use by the operator. For example, a pledge may support connecting to a manufacturer cloud service or an operator deployed service after bootstrapping is complete, and needs to discover the deployment model in use by the pledge operator. The discovery and bootstrap mechanism should be consistent across both manufacturer cloud service and operator deployed services.

## 2. Architecture

The high level architecture is illustrated in Figure 1. The pledge connects to the cloud registrar during bootstrap. The cloud registrar may redirect the pledge to a local registrar in order to complete bootstrap against the local registrar. If the cloud registrar handles the bootstrap process itself without redirecting the pledge to a local registrar, the cloud registrar may need to inform the pledge what domain to use for accessing services once bootstrap is complete.

Finally, when bootstrapping against a local registrar, the registrar may interact with a backend CA to assist in issuing certificates to the pledge. The mechanisms and protocols by which the registrar interacts with the CA are transparent to the pledge and are out-of-scope of this document.

The architecture illustrates shows the cloud registrar and MASA as being logically separate entities. The two functions could of course be integrated into a single service.

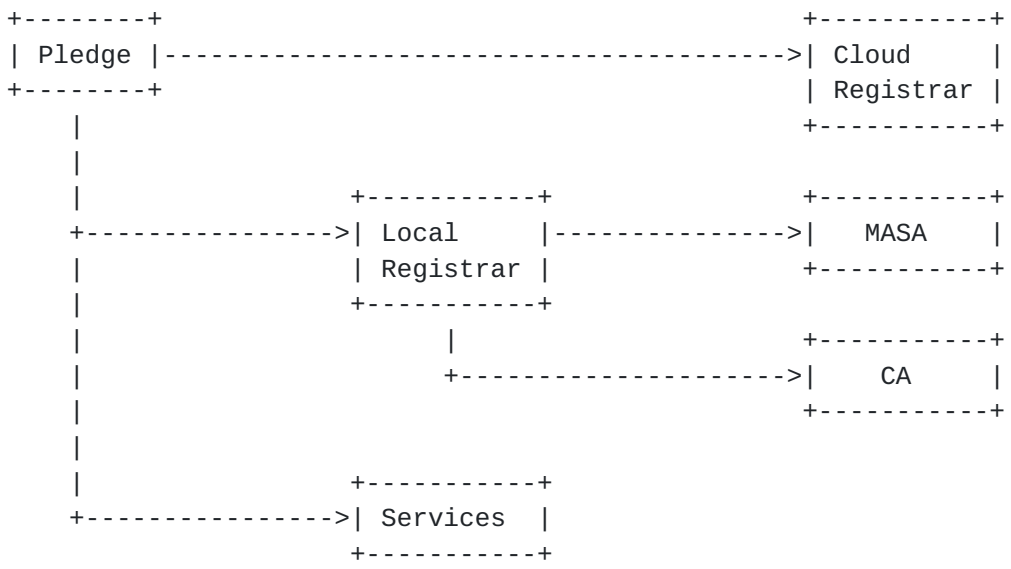



Figure 1

## 2.1. Network Connectivity

The assumption is that the pledge already has network connectivity prior to connecting to the cloud registrar. The pledge must have an IP address, must be able to make DNBS queries, and must be able to send HTTP requests to the cloud registrar. The pledge operator has already connected the pledge to the network, and the mechanism by which this has happened is out of scope of this document.

# 3. Initial Voucher Request

## 3.1. Cloud Registrar Discovery

BRSKI defines how a pledge MAY contact a well known URI of a cloud registrar if a local registrar cannot be discovered. Additionally, certain pledge types may never attempt to discover a local registrar and may automatically bootstrap against a cloud registrar. The details of the URI are manufacturer specific, with BRSKI giving the example "brski-registrar.manufacturer.example.com".

## 3.2. Pledge - Cloud Registrar TLS Establishment Details

The pledge MUST use an Implicit Trust Anchor database (see [RFC7030]) to authenticate the cloud registrar service as described in [RFC6125]. The pledge MUST NOT establish a provisional TLS connection (see BRSKI section 5.1) with the cloud registrar.

The cloud registrar MUST validate the identity of the pledge by sending a TLS CertificateRequest message to the pledge during TLS

session establishment. The cloud registrar MAY include a certificate_authorities field in the message to specify the set of allowed IDevID issuing CAs that pledges may use when establishing connections with the cloud registrar.

The cloud registrar MAY only allow connections from pledges that have an IDevID that is signed by one of a specific set of CAs, e.g. IDevIDs issued by certain manufacturers.

The cloud registrar MAY allow pledges to connect using self-signed identity certificates or using Raw Public Key [RFC7250] certificates.

### 3.3. Pledge Requests Voucher from the Cloud Registrar

After the pledge has established a full TLS connection with the cloud registrar and has verified the cloud registrar PKI identity, the pledge generates a voucher request message as outlined in BRSKI section 5.2, and sends the voucher request message to the cloud registrar.

## 4. Cloud Registrar Voucher Request Operation

When the cloud registrar has verified the identity of the pledge, determined the pledge ownership and has received the voucher request, there are two main options for handling the request.

- the cloud registrar can redirect the voucher request to a local domain registrar

- the cloud registrar can handle the voucher request directly by either issuing a voucher or declining the request

### 4.1. Pledge Ownership Lookup

The cloud registrar needs some suitable mechanism for knowing the correct owner of a connecting pledge based on the presented identity certificate. For example, if the pledge establishes TLS using an IDevID that is signed by a known manufacturing CA, the registrar could extract the serial number from the IDevID and use this to lookup a database of pledge IDevID serial numbers to owners.

Alternatively, if the cloud registrar allows pledges to connect using self-signed certificates, the registrar could use the thumbprint of the self-signed certificate to lookup a database of pledge self-signed certificate thumbprints to owners.

The mechanism by which the cloud registrar determines pledge ownership is out-of-scope of this document.

## 5. Voucher Request Redirected to Local Domain Registrar

Once the cloud registar has determined pledge ownership, the cloud registrar may redirect the pledge to the owner's local domain registrar in order to complete bootstrap. Ownership registration will require the owner to register their local domain. The mechanism by which pledge owners register their domain with the cloud registrar is out-of-scope of this document.

The cloud registrar replies to the voucher request with a suitable HTTP 3xx response code as per [I-D.ietf-httpbis-bcp56bis], including the owner's local domain in the HTTP Location header.

### 5.1. Pledge handling of Redirect

The pledge should complete BRSKI bootstrap as per standard BRSKI operation after following the HTTP redirect. The pledge should establish a provisional TLS connection with specified local domain registrar. The pledge should not use its Implicit Trust Anchor database for validating the local domain registrar identity. The pledge should send a voucher request message via the local domain registrar. When the pledge downloads a voucher, it can validate the TLS connection to the local domain registrar and continue with enrollment and bootstrap as per standard BRSKI operation.

## 6. Voucher Request Handled by Cloud Registrar

If the cloud registrar issues a voucher, it returns the voucher in a HTTP response with a suitable 2xx response code as per [I-D.ietf-httpbis-bcp56bis].

[[ TODO: it is TBD which of the following three options should be used. Possibly 1 or 2 of them, maybe all 3. It is possible that some options will be explicitly NOT recommended. There are standards implications too as two of the options require including a DNS-ID in a Voucher. ]]

There are a few options here:

- Option 1: the pledge completes EST enroll against the cloud registrar. Once EST enrol is complete, we need a mechanism to tell the pledge what its service domain is. This could be by including a service domain in the voucher.

- Option 2: the pledge attempts EST enrol against the cloud registrar and the cloud registrar responds with a 3xx redirecting the pledge to the local domain RA in order to complete cert

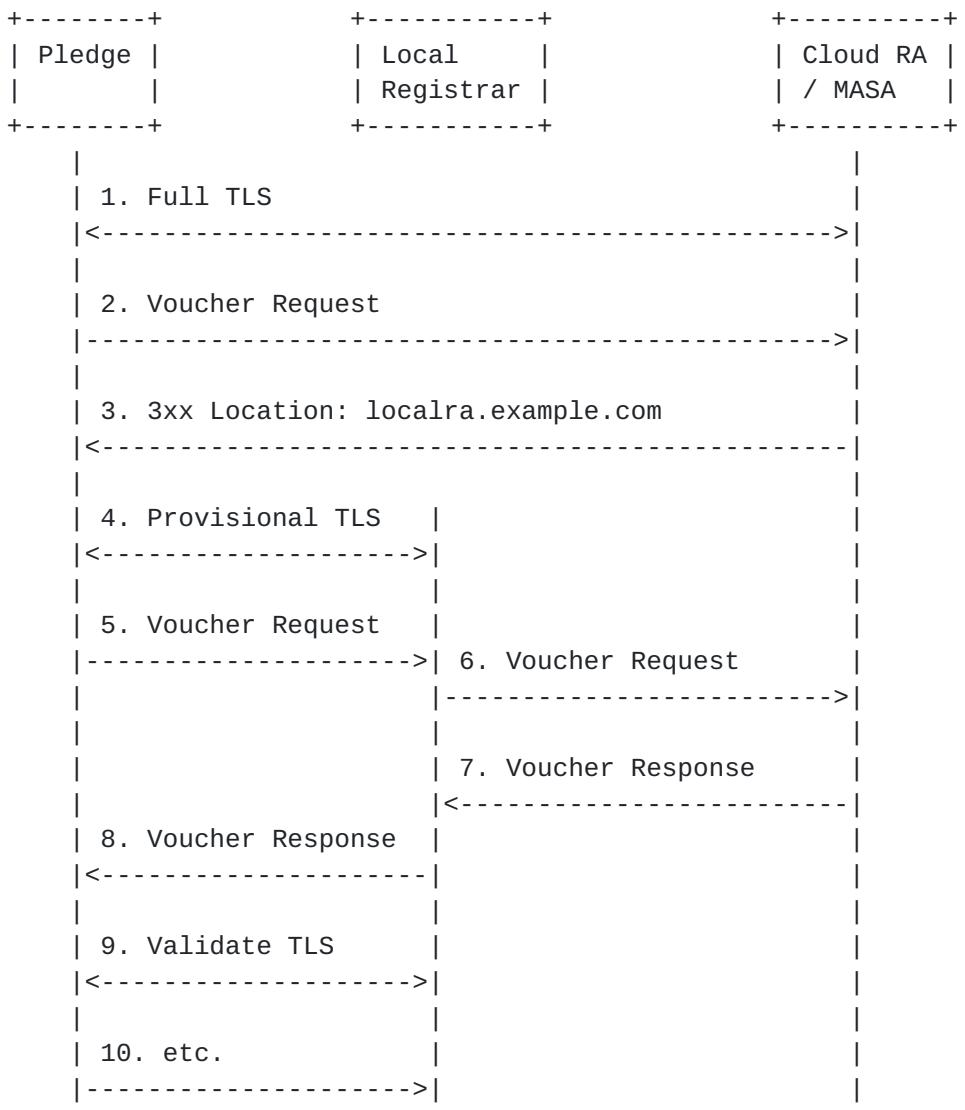
enrollment. The pledge assumes that services are off the local domain. This does not require adding an FQDN to the voucher.

- Option 3: we enhance the voucher definition to include local RA domain info, and the pledge implicitly knows that it if received a voucher from the cloud registrar, and that voucher included a local domain FQDN, the pledge knows to do EST enroll against the local domain. i.e. it got a 200OK from the cloud registrar, and knows to send the next HTTP request to the EST domain specified in the voucher. The pledge assumes that services are off the local domain specified in the voucher.

## 7. Protocol Details

[[ TODO ]] Missing detailed BRSKI steps e.g. CSR attributes, logging, etc.

### 7.1. Voucher Request Redirected to Local Domain Registrar

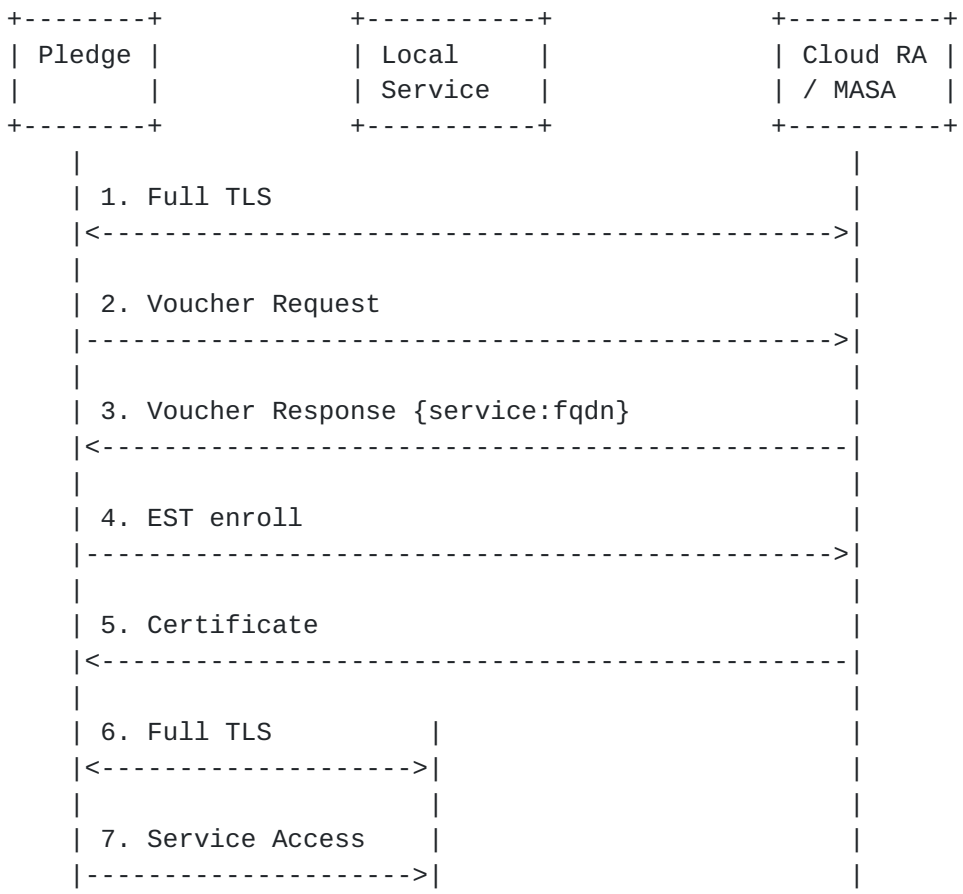



## 7.2. Voucher Request Handled by Cloud Registrar

[[ TODO: it is TBD which of the following three options should be used. Possibly 1 or 2 of them, maybe all 3. It is possible that some options will be explicitly NOT recommended. There are standards implications too as two of the options require including a DNS-ID in a Voucher. ]]

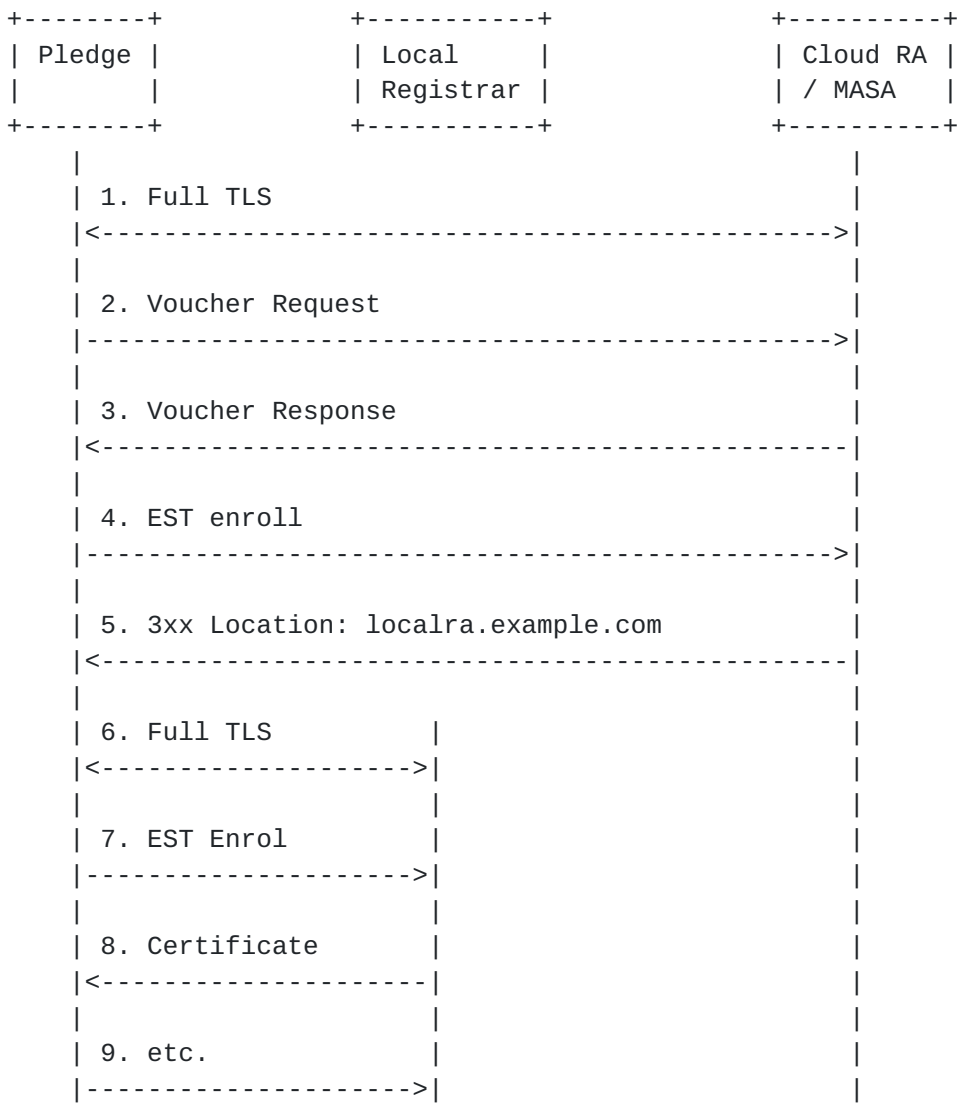
### 7.2.1. Option 1: EST enroll completed against cloud registrar

The Voucher includes the service domain to use after EST enroll is complete.



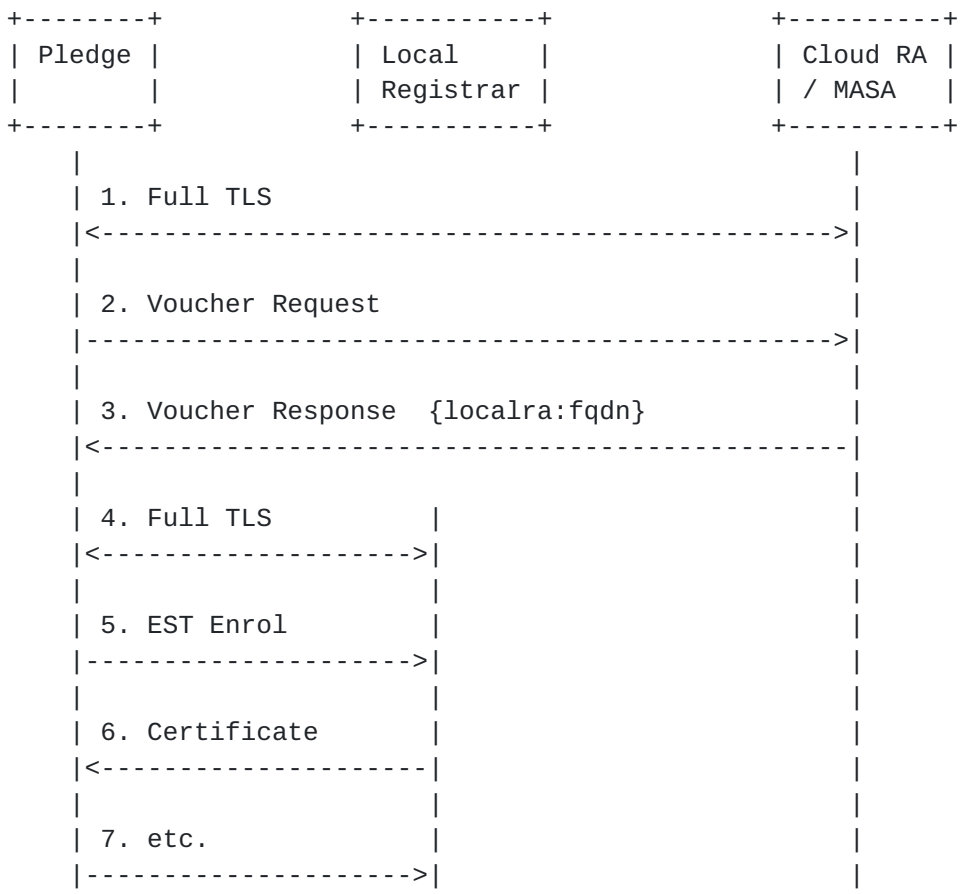

### 7.2.2. Option 2: EST redirect by cloud registrar

As trust is already established via the Voucher, the pledge does a full TLS handshake against the local RA.




### 7.2.3. Option 3: Voucher includes EST domain

The Voucher includes the EST domain to use for EST enroll. It is assumed services are accessed at that domain too. As trust is already established via the Voucher, the pledge does a full TLS handshake against the local RA.




## 8. Pledge Certificate Identity Considerations

BRSKI section 5.9.2 specifies that the pledge MUST send a CSR Attributes request to the registrar. The registrar MAY use this mechanism to instruct the pledge about the identities it should include in the CSR request it sends as part of enrollment. The registrar may use this mechanism to tell the pledge what Subject or Subject Alternative Name identity information to include in its CSR request. This can be useful if the Subject must have a specific value in order to complete enrollment with the CA.

For example, the pledge may only be aware of its IDevID Subject which includes a manufacturer serial number, but must include a specific fully qualified domain name in the CSR in order to complete domain ownership proofs required by the CA. As another example, the registrar may deem the manufacturer serial number in an IDevID as personally identifiable information, and may want to specify a new random opaque identifier that the pledge should use in its CSR.

## 9. IANA Considerations

[[ TODO ]]

## 10. Security Considerations

[[ TODO ]]

## 11. Informative References

[I-D.ietf-anima-bootstrapping-keyinfra]
Pritikin, M., Richardson, M., Eckert, T., Behringer, M., and K. Watsen, "Bootstrapping Remote Secure Key Infrastructures (BRSKI)", draft-ietf-anima-bootstrapping-keyinfra-28 (work in progress), September 2019.

[I-D.ietf-httpbis-bcp56bis]
Nottingham, M., "Building Protocols with HTTP", draft-ietf-httpbis-bcp56bis-08 (work in progress), November 2018.

[IEEE802.1AR]
IEEE, ., "Secure Device Identity", 2017.

[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, <https://www.rfc-editor.org/info/rfc6125>.

[RFC7030] Pritikin, M., Ed., Yee, P., Ed., and D. Harkins, Ed., "Enrollment over Secure Transport", RFC 7030, DOI 10.17487/RFC7030, October 2013, <https://www.rfc-editor.org/info/rfc7030>.

[RFC7250] Wouters, P., Ed., Tschofenig, H., Ed., Gilmore, J., Weiler, S., and T. Kivinen, "Using Raw Public Keys in Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", RFC 7250, DOI 10.17487/RFC7250, June 2014, <https://www.rfc-editor.org/info/rfc7250>.

## Authors' Addresses

Owen Friel
Cisco

Email: ofriel@cisco.com

Rifaat Shekh-Yusef
Avaya

Email: rifaat.ietf@gmail.com

Michael Richardson
Sandelman Software Works

Email: mcr+ietf@sandelman.ca