Workgroup:	ANIMA WG		
Internet-Draft:	draft-ietf-anima-brski-ae-03		
Published:	22 October 2022		
Intended Status:	Standards Track		
Expires:	25 April 2023		
Authors:	D. von Oheimb, Ed. *Siemens*	S. Fries *Siemens*	H. Brockhaus *Siemens*

BRSKI-AE: Alternative Enrollment Protocols in BRSKI

Abstract

This document enhances Bootstrapping Remote Secure Key Infrastructure (BRSKI, RFC 8995) to allow employing alternative enrollment protocols, such as CMP.

Using self-contained signed objects, the origin of enrollment requests and responses can be authenticated independently of message transfer. This supports end-to-end security and asynchronous operation of certificate enrollment and provides flexibility where to authenticate and authorize certification requests.

The RFC Editor will remove this note

About This Document

This note is to be removed before publishing as an RFC.

Status information for this document may be found at https://datatracker.ietf.org/doc/draft-ietf-anima-brski-ae/.

Source for this draft and an issue tracker can be found at https://github.com/anima-wg/anima-brski-ae.

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 25 April 2023.

Copyright Notice

Table of Contents

1. Introduction

1.1. Motivation

BRSKI, as defined in [RFC8995], specifies a solution for secure automated zero-touch bootstrapping of new devices, which are given the name *pledges*, in the domain they should operate with. This includes the discovery of the registrar representing the target domain, time synchronization or validation, and the exchange of security information necessary to establish mutual trust between pledges and the target domain. As explained in Section 2, the *target domain*, or *domain* for short, is defined as the set of entities that share a common local trust anchor.

1.1.1. Voucher Exchange for Trust Anchor Establishment

Initially, a pledge has a trust anchor only of its manufacturer, not yet of any target domain. In order for the pledge to automatically and securely obtain trust in a suitable target domain represented by its registrar, BRSKI uses vouchers defined in [RFC8366]. A voucher is a cryptographic object issued by the Manufacturer Authorized Signing Authority (MASA) of the pledge manufacturer to the specific pledge identified by the included device serial number. It is signed with the credentials of the MASA and can be validated by the manufacturer trust anchor imprinted with the pledge. So the pledge can accept the voucher contents, which indicate to the pledge that it can trust the domain identified by the given certificate.

While RFC 8995 only specifies a single, online set of protocol option to communicate the voucher between MASA, registrar, and pledge (BRSKI-EST and BRSKI-MASA, see [RFC8995], Section 2), it also describes the architecture for how the voucher may be provided in online mode (synchronously) or offline mode (asynchronously). So for the voucher exchange offline mode is basically supported because the vouchers are self-contained signed objects, such that their security does not rely on protection by the underlying transfer.

SZTP [RFC8572] is an example of another mode where vouchers may be delivered asynchronously by tools such as portable USB "thumb" drives. However, SZTP does not do signed voucher requests, so it does not allow the domain to verify the identity of the device in the same way, nor does it deploy LDevIDs to the device in the same way.

1.1.2. Enrollment of LDevID Certificate

Trust by the target domain in a pledge is established by enrolling the pledge with a domain-specific Locally significant Device IDentity (LDevID) certificate.

Recall that for certificate enrollment it is crucial to authenticate the entity requesting the certificate. Checking both the identity and the authorization of the requester is the job of a registration authority (RA). With BRSKI-EST, there is only one RA instance, co-located with the registrar.

The certification request of the pledge is signed using its IDevID secret. It can be validated by the target domain (e.g., by the domain registrar) using the trust anchor of the pledge manufacturer, which needs to pre-installed in the domain.

For enrolling devices with LDevID certificates, BRSKI specifies how Enrollment over Secure Transport (EST) [RFC7030] can be used. EST has its specific characteristics, detailed in Appendix A. In particular, it requires online on-site availability of the RA for performing the data origin authentication and final authorization decision on the certification request. This type of enrollment can be called 'synchronous enrollment'. EST, BRSKI-EST, and BRSKI-MASA as used in RFC 8995 are tied to a specific transport, TLS, which may not be suitable for the target use case outlined by the examples in Section 1.3. Therefore deployments may require different transport, see Constrained Voucher Artifacts for Bootstrapping Protocols [I-D.ietf-anima-constrained-voucher] and EST-coaps [RFC9148].

Since EST does not support offline enrollment, it may be preferable for the reasons given in this section and depending on application scenarios as outlined in Section 1.3 and Appendix B to use alternative enrollment protocols such as the Certificate Management Protocol (CMP) [RFC4210] profiled in [I-D.ietf-lamps-lightweight-cmp-profile] or Certificate Management over CMS (CMC) [RFC5272]. These protocols are more flexible, and by representing the certification request messages as authenticated self-contained objects, they are designed to be independent of the transfer mechanism.

Depending on the application scenario, the required components of an RA may not be part of the BRSKI registrar. They even may not be available on-site but rather be provided by remote backend systems. The RA functionality may also be split into an on-site local RA (LRA) and a central RA component in the backend, referred to as PKI RA. For certification authorities (CAs) it is common to be located in the backend. The registrar or its deployment site may not have an online connection with these RA/CA components or the connectivity may be intermittent. This may be due to security requirements for operating the backend systems or due to deployments where on-site or always-online operation may be not feasible or too costly. In such scenarios, the authentication and authorization of certification requests will not or can not be performed on-site.

In this document, enrollment that is not performed over an online connection is called 'asynchronous enrollment'. Asynchronous enrollment means that messages need to be forwarded through offline methods (e.g., Sneakernet/USB sticks) and/or at some point in time only part of the communication path is available. Messages need to be stored, along with the information needed for authenticating their origin, in front of an unavailable segment for potentially long time (e.g., days) before they can be forwarded. This implies that end-to-end security between the parties involved can not be provided by an authenticated (and often confidential) communications channel such as TLS used in EST/BRSKI-EST/BRSKI-MASA.

Application scenarios may also involve network segmentation, which is utilized in industrial systems to separate domains with different security needs -- see also Appendix B.5. Such scenarios lead to similar requirements if the TLS channel carrying the requester authentication is terminated and thus request messages need to be forwarded on further channels before the registrar or RA can authorize the certification request. In order to preserve the requester authentication, authentication information needs to be retained and ideally bound directly to the certification request.

There are basically two approaches for forwarding certification requests along with requester authentication information:

- The component in the target domain that forwards the certification request, such as a local RA being part of the registrar, combines the certification request with the validated identity of the requester (e,g., its IDevID certificate) and an indication of successful verification of the proof of possession (of the corresponding private key) in a way preventing changes to the combined information. This implies that it must be trusted by the PKI. When connectivity is available, the trusted component forwards the certification request together with the requester information (authentication and proof of possession) for further processing. This approach offers hop-by-hop security, but not end-to-end security.

In BRSKI, the EST server, being co-located with the registrar in the domain, is such a component that needs to be trusted by the backend PKI components. They must rely on the local pledge authentication result provided by that component when performing the final authorization of the certification request.

- A trusted intermediate domain component is not needed when involved components use authenticated self-contained objects for the enrollment, directly binding the certification request and the requester authentication in a cryptographic way. This approach supports end-to-end security, without the need to trust in intermediate domain components. Manipulation of the request and the requester identity information can be detected during the validation of the self-contained signed object.

  Note that with this approach the way in which enrollment requests are forwarded by the registrar to the backend PKI components does not contribute to their security and therefore does not need to be addressed here.

Focus of this document is the support of alternative enrollment protocols that allow the second approach, i.e., using authenticated self-contained objects for device certificate enrollment. This enhancement of BRSKI is named BRSKI-AE, where AE stands for **A**lternative **E**nrollment and for **A**synchronous **E**nrollment. This specification carries over the main characteristics of BRSKI, namely that the pledge obtains trust anchor information for authenticating the domain registrar and other target domain components as well as a domain-specific X.509 device certificate (the LDevID certificate) along with the corresponding private key (the LDevID secret) and certificate chain.

The goals are to provide an interpretation of BRSKI using enrollment protocols alternatively to EST that

- support end-to-end security for LDevID certificate enrollment and
- make it applicable to scenarios involving asynchronous enrollment.

This is achieved by

- extending the well-known URI approach of BRSKI and EST message with an additional path element indicating the enrollment protocol being used, and
- defining a certificate waiting indication and handling, for the case that the certifying component is (temporarily) not available.

This specification can be applied to both synchronous and asynchronous enrollment.

As an improvement over BRSKI, this specification supports offering multiple enrollment protocols on the infrastructure side, which enables pledges and their developers to pick the preferred one.

1.2. Supported Environments

BRSKI-AE is intended to be used in like the following.

- Scenarios indirectly excluding the use of EST for certificate enrollment, such as the requirement for end-to-end authentication of the requester.
- Scenarios having implementation restrictions that speak against using EST for certificate enrollment, such as the use of a library that does not support EST but CMP.
- Pledges and/or the target domain already having an established certificate management approach different from EST that shall be reused (e.g., in brownfield installations where CMP is used).
- No RA being available on site in the target domain. Connectivity to an off-site PKI RA is intermittent or entirely offline. A store-and-forward mechanism is used for communicating with the off-site services.
- Authoritative actions of a local RA being not sufficient for fully authorizing certification requests by pledges. Final authorization then is done by a PKI RA residing in the backend.

1.3. List of Application Examples

Bootstrapping can be handled in various ways, depending on the application domains. The informative Appendix B provides illustrative examples from various industrial control system environments and operational setups. They motivate the support of alternative enrollment protocols, based on the following examples of operational environments:

- Rolling stock
- Building automation
- Electrical substation automation
- Electric vehicle charging infrastructures
- Infrastructure isolation policy
- Sites with insufficient level of operational security

2. Terminology

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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

This document relies on the terminology defined in [RFC8995] and [IEEE.8802.1AR_2014]. The following terms are defined partly in addition.

asynchronous communication: time-wise interrupted communication between a pledge and a registrar or PKI component.

authenticated self-contained object: data structure that is cryptographically bound to the IDevID certificate of a pledge. The binding is assumed to be provided through a digital signature of the actual object using the IDevID secret.

backend: same as off-site

BRSKI-AE: Variation of BRSKI [RFC8995] in which BRSKI-EST, the enrollment protocol between pledge and the registrar including the RA, is replaced by alternative enrollment protocols such as Lightweight CMP. To this end a new URI scheme used for performing the certificate enrollment. BRSKI-AE enables the use of other enrollment protocols between pledge and registrar and to any backend RA components with end-to-end security.

CA: Certification Authority, which is the PKI component that issues certificates and provides certificate status information.

domain: shorthand for target domain

IDevID: Initial Device IDentifier, provided by the manufacturer and comprising of a private key, an X.509 certificate with chain, and a related trust anchor.

LDevID: Locally significant Device IDentifier, provided by the target domain and comprising of a private key, an X.509 certificate with chain, and a related trust anchor.

local RA (LRA): RA that is on site with the registrar and that may be needed in addition to an off-site RA.

on-site: locality of a component or service or functionality in the local target deployment site of the registrar.

off-site: locality of component or service or functionality in an operator site different from the target deployment site. This may be a central site or a cloud service, to which only a temporary connection is available.

PKI RA: off-site RA in the backend of the target domain

pledge: device that is to be bootstrapped to the target domain. It requests an LDevID using an IDevID installed by its manufacturer.

RA: Registration Authority, which is the PKI component to which a CA typically delegates certificate management functions such as authenticating requesters and performing authorization checks on certification requests.

site: the locality where an entity, e.g., pledge, registrar, RA, CA, is deployed. Different sites can belong to the same target domain.

synchronous communication: time-wise uninterrupted communication between a pledge and a registrar or PKI component.

target domain: the set of entities that the pledge should be able to operate with and that share a common local trust anchor, independent of where the entities are deployed.

3. Requirements and Mapping to Solutions

3.1. Basic Requirements

There are two main drivers for the definition of BRSKI-AE:

- The solution architecture may already use or require a certificate management protocol other than EST. Therefore, this other protocol should be usable for requesting LDevID certificates.
- The domain registrar may not be the (final) point that authenticates and authorizes certification requests and the pledge may not have a direct connection to it. Therefore, certification requests should be self-contained signed objects.

Based on the intended target environment described in Section 1.2 and the application examples described in Appendix B, the following requirements are derived to support authenticated self-contained objects as containers carrying certification requests.

At least the following properties are required:

- *Proof of possession*: demonstrates access to the private key corresponding to the public key contained in a certification request. This is typically achieved by a self-signature using the corresponding private key.
- *Proof of identity*, also called *proof of origin*: provides data origin authentication of the certification request. Typically this is achieved by a signature using the pledge IDevID secret over some data, which needs to include a sufficiently strong identifier of the pledge, such as the device serial number typically included in the subject of the IDevID certificate.

The rest of this section gives an non-exhaustive list of solution examples, based on existing technology described in IETF documents:

3.2. Solution Options for proof of Possession

Certification request objects: Certification requests are data structures protecting only the integrity of the contained data and providing proof of possession for a (locally generated) private key. Examples for certification request data structures are:

- PKCS#10 [RFC2986]. This certification request structure is self-signed to protect its integrity and prove possession of the private key that corresponds to the public key included in the request.
- CRMF [RFC4211]. This certificate request message format also supports integrity protection and proof of possession, typically by a self-signature generated over (part of) the structure with the private key corresponding to the included public key. CRMF also supports further proof-of-possession methods for types of keys that do not support any signature algorithm.

The integrity protection of certification request fields includes the public key because it is part of the data signed by the corresponding private key. Yet note that for the above examples this is not sufficient to provide data origin authentication, i.e., proof of identity. This extra property can be achieved by an additional binding to the IDevID of the pledge. This binding to the source authentication supports the authorization decision of the certification request. The binding of data origin authentication to the certification request may be delegated to the protocol used for certificate management.

3.3. Solution Options for proof of Identity

The certification request should be bound to an existing authenticated credential (here, the IDevID certificate) to enable a proof of identity and, based on it, an authorization of the certification request. The binding may be achieved through security options in an underlying transport protocol such as TLS if the authorization of the certification request is (completely) done at the next communication hop. This binding can also be done in a transport-independent way by wrapping the certification request with a signature employing an existing IDevID. In the BRSKI context, this will be the IDevID. This requirement is addressed by existing enrollment protocols in various ways, such as:

- EST [RFC7030] utilizes PKCS#10 to encode the certification request. The Certificate Signing Request (CSR) optionally provides a binding to the underlying TLS session by including the tls-unique value in the self-signed PKCS#10 structure. The tls-unique value results from the TLS handshake. Since the TLS handshake includes certificate-based client authentication and the pledge utilizes its IDevID for it, the proof of identity is provided by such a binding to the TLS session. This can be supported using the EST /simpleenroll endpoint. Note that the binding of the TLS handshake to the CSR is optional in EST.

  [RFC7030], Section 2.5 sketches wrapping the CSR with a Full PKI Request message sent to the /fullcmc endpoint. This would allow for source authentication at message level as an alternative to indirectly binding to the underlying TLS authentication in the transport layer.
- SCEP [RFC8894] supports using a shared secret (passphrase) or an existing certificate to protect CSRs based on SCEP Secure Message Objects using CMS wrapping ([RFC5652]). Note that the wrapping using an existing IDevID in SCEP is referred to as *renewal*. This way SCEP does not rely on the security of the underlying message transfer.
- CMP [RFC4210] supports using a shared secret (passphrase) or an existing certificate, which may be an IDevID credential, to authenticate certification requests via the PKIProtection structure in a PKIMessage. The certification request is typically encoded utilizing CRMF, while PKCS#10 is supported as an alternative. Thus CMP does not rely on the security of the underlying message transfer.
- CMC [RFC5272] also supports utilizing a shared secret (passphrase) or an existing certificate to protect certification requests, which can be either in CRMF or PKCS#10 structure. The proof of identity can be provided as part of a FullCMCRequest, based on CMS [RFC5652] and signed with an existing IDevID secret. Thus also CMC does not rely on the security of the underlying message transfer.

4. Adaptations to BRSKI

In order to support alternative certificate enrollment protocols, asynchronous enrollment, and more general system architectures, BRSKI-AE provides some generalizations on BRSKI [RFC8995]. This way, authenticated self-contained objects such as those described in Section 3 above can be used for certificate enrollment, and RA functionality can be distributed freely in the target domain.

The enhancements needed are kept to a minimum in order to ensure reuse of already defined architecture elements and interactions. In general, the communication follows the BRSKI model and utilizes the existing BRSKI architecture elements. In particular, the pledge initiates communication with the domain registrar and interacts with the MASA as usual.

4.1. Architecture

The key element of BRSKI-AE is that the authorization of a certification request **MUST** be performed based on an authenticated self-contained object. The certification request is bound in a self-contained way to a proof of origin based on the IDevID. Consequently, the authentication and authorization of the certification request **MAY** be done by the domain registrar and/or by other domain components. These components may be offline or reside in some central backend of the domain operator (off-site) as described in Section 1.2. The registrar and other on-site domain components may have no or only temporary (intermittent) connectivity to them. The certification request **MAY** also be piggybacked on another protocol.

This leads to generalizations in the placement and enhancements of the logical elements as shown in Figure 1.

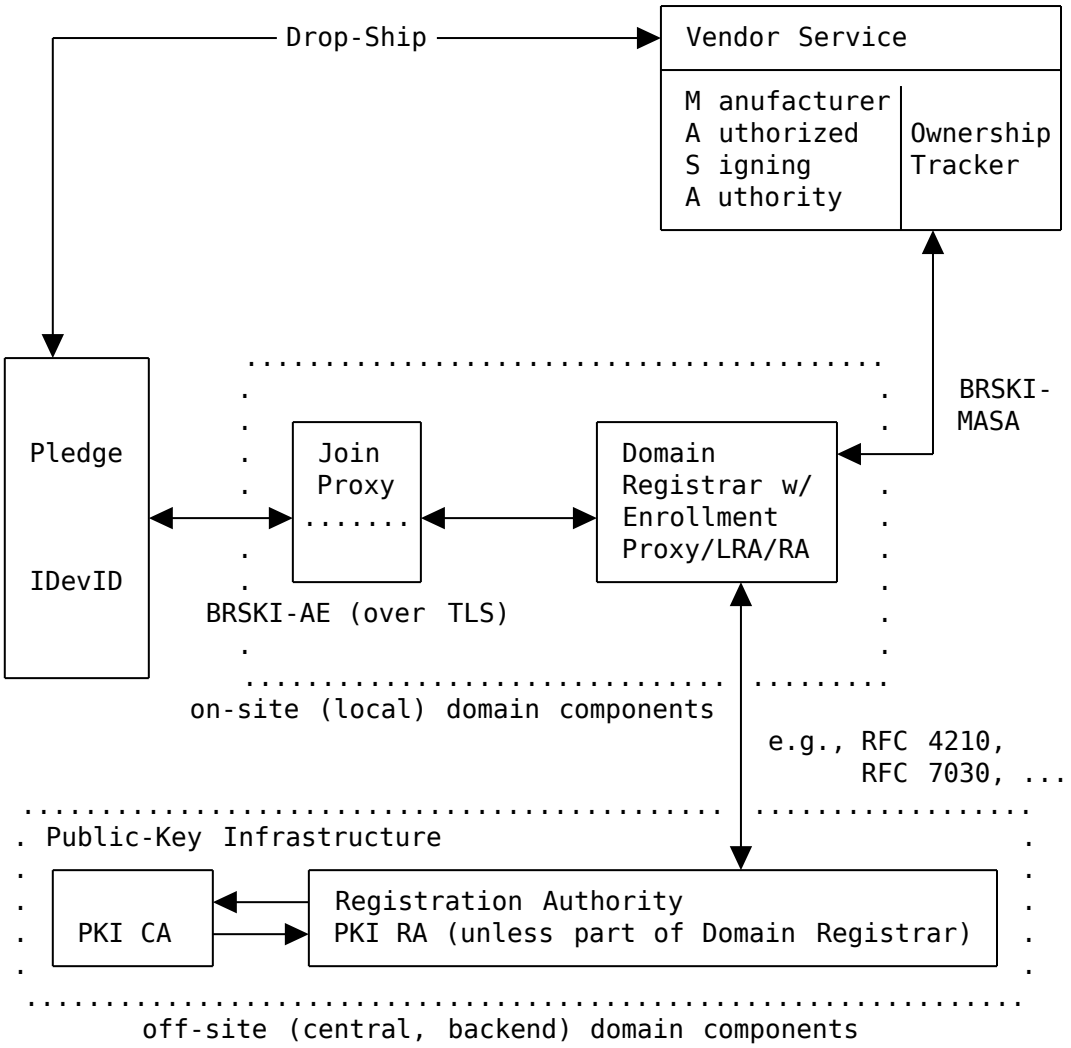

Figure 1: Architecture Overview Using Off-site PKI Components

The architecture overview in Figure 1 has the same logical elements as BRSKI, but with more flexible placement of the authentication and authorization checks on certification requests. Depending on the application scenario, the registrar **MAY** still do all of these checks (as is the case in BRSKI), or part of them, or none of them.

The following list describes the on-site components in the target domain of the pledge shown in Figure 1.

- Join Proxy: same functionality as described in BRSKI [RFC8995], Section 4

- Domain Registrar including RA, LRA, or Enrollment Proxy: in BRSKI-AE, the domain registrar has mostly the same functionality as in BRSKI, namely to facilitate the communication of the pledge with the MASA and the PKI. Yet there are two generalizations:
  1. The registrar **MUST** support at least one certificate enrollment protocol that uses for certificate requests authenticated self-contained objects. To this end, the URI scheme for addressing the endpoint at the registrar is generalized (see Section 4.3).

     To support the end-to-end proof of identity of the pledge, the registrar **MUST** use for the upstream certificate enrollment message exchange with backend PKI components the same enrollment protocol as used by the pledge. Between the pledge and the registrar the enrollment request messages are tunneled over the TLS channel already established between these entities. The registrar optionally checks the requests and then passes them on to the PKI. On the way back, it forwards responses by the PKI to the pledge on the existing TLS channel.
  2. The registrar **MAY** also delegate all or part of its certificate enrollment support to a separate system. That is, alternatively to having full RA functionality, the registrar may act as a local registration authority (LRA) or just as an enrollment proxy. In such cases, the domain registrar may forward the certification request to some off-site RA component, also called PKI RA here, that performs the remaining parts of the enrollment request validation and authorization. This also covers the case that the registrar has only intermittent connection and forwards certification requests to off-site PKI components upon re-established connectivity.

     Still all certificate enrollment traffic goes via the registrar, such that from the pledge perspective there is no difference in connectivity and the registrar is involved in all steps. The final step of BRSKI, namely the enrollment status telemetry, is also kept.

The following list describes the components provided by the vendor or manufacturer outside the target domain.

- MASA: functionality as described in BRSKI [RFC8995]. The voucher exchange with the MASA via the domain registrar is performed as described in BRSKI.

  Note: From the definition of the interaction with the MASA in [RFC8995], Section 5 follows that it may be synchronous (voucher request with nonce) or asynchronous (voucher request without nonce).
- Ownership tracker: as defined in BRSKI.

The following list describes the target domain components that can optionally be operated in the off-site backend of the target domain.

- PKI RA: Performs certificate management functions for the domain as a centralized public-key infrastructure for the domain operator. As far as not already done by the domain registrar, it performs the final validation and authorization of certification requests. Otherwise, the RA co-located with the domain registrar directly connects to the PKI CA.
- PKI CA: Performs certificate generation by signing the certificate structure requested in already authenticated and authorized certification requests.

Based on the diagram in BRSKI [RFC8995], Section 2.1 and the architectural changes, the original protocol flow is divided into four phases showing commonalities and differences to the original approach as follows.

- Discovery phase: same as in BRSKI steps (1) and (2).
- Voucher exchange phase: same as in BRSKI steps (3) and (4).
- Certificate enrollment phase: the use of EST in step (5) is changed to employing a certificate enrollment protocol that uses an authenticated self-contained object for requesting the LDevID certificate.

  Still for transporting certificate enrollment request and response messages between the pledge and the registrar, the TLS channel established between them via the join proxy is used. So the enrollment protocol **MUST** support this. Due to this architecture, the pledge does not need to establish an additional connection for certificate enrollment and the registrar retains control over the certificate enrollment traffic.
- Enrollment status telemetry phase: the final exchange of BRSKI step (5).

4.2. Message Exchange

The behavior of a pledge described in BRSKI [RFC8995], Section 2.1 is kept with one exception. After finishing the Imprint step (4), the Enroll step (5) **MUST** be performed with an enrollment protocol utilizing authenticated self-contained objects. Section 5 discusses selected suitable enrollment protocols and options applicable.

Figure 2: BRSKI-AE Abstract Protocol Overview

4.2.1. Pledge - Registrar Discovery

The discovery is done as specified in [RFC8995].

4.2.2. Pledge - Registrar - MASA Voucher Exchange

The voucher exchange is performed as specified in [RFC8995].

4.2.3. Pledge - Registrar - RA/CA Certificate Enrollment

The certificate enrollment phase may involve several exchanges of requests and responses. Which of the message exchanges marked **OPTIONAL** in the below Figure 3 are potentially used, or are actually required or prohibited to be used, depends on the application scenario and on the employed enrollment protocol.

These **OPTIONAL** exchanges cover all those supported by the use of EST in BRSKI. The last **OPTIONAL** one, namely certificate confirmation, is not supported by EST, but by CMP and other enrollment protocols.

The only generally MANDATORY message exchange is for the actual certificate request and response. As stated in Section 3, the certificate request **MUST** be performed using an authenticated self-contained object providing not only proof of possession but also proof of identity (source authentication).

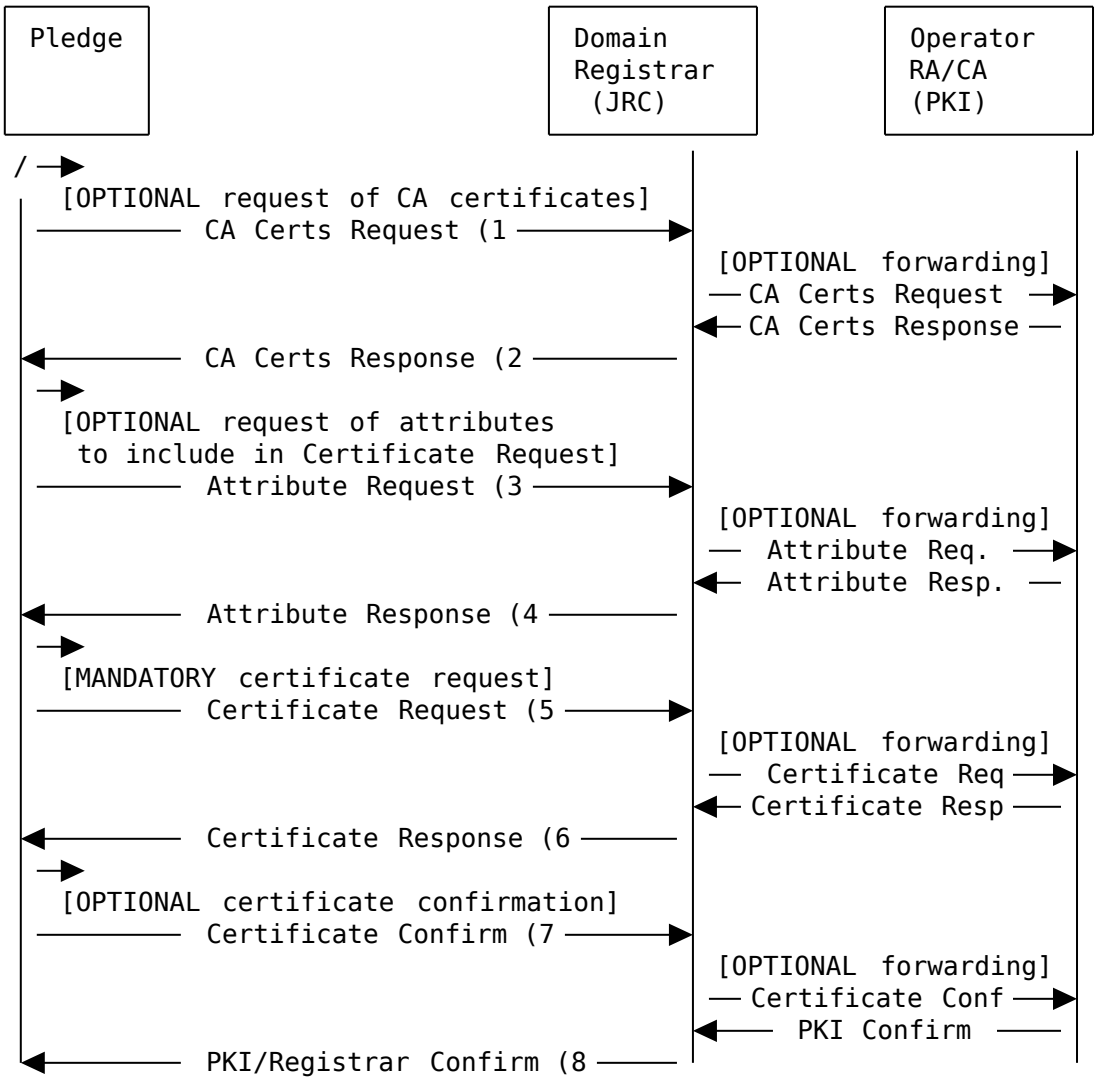

Figure 3: Certificate Enrollment

Note that the various connections between the registrar and the PKI components of the operator (RA/CA) may be intermittent or off-line. Messages are to be sent as soon as sufficient transfer capacity is available.

The label "[**OPTIONAL** forwarding]" means that on receiving from a pledge a request of the given type (or certificate confirmation), depending on the application scenario, the enrollment protocol being used, and the capabilities of the registrar and the local RA possibly co-located with it, the registrar **MAY** answer the request itself. For CA certificates request/response this would require for example explicit provisioning of the certificates at the registrar.

Unless providing the response itself, the registrar **MUST** forward the request to a backend PKI component and forward any resulting response back to the pledge. The registrar **MAY** cache responses containing CA certificates or attributes and use them later for responding directly, as far as suitable.

Note: Typically, certificate requests will be forwarded to the backend PKI, but even for these the registrar may answer part of them if adequate, such as returning an error response in case the registrar determines that the request is not properly authenticated or not authorized.

The following list provides an abstract description of the flow depicted in Figure 3.

- CA Certs Request (1): The pledge optionally requests the latest relevant CA certificates. This ensures that the pledge has the complete set of current CA certificates beyond the pinned-domain-cert (which is contained in the voucher and may be just the domain registrar certificate).
- CA Certs Response (2): This **MUST** contain the current root CA certificate, which typically is the LDevID trust anchor, and any additional certificates that the pledge may need to validate certificates.
- Attribute Request (3): Typically, the automated bootstrapping occurs without local administrative configuration of the pledge. Nevertheless, there are cases in which the pledge may also include additional attributes specific to the target domain into the certification request. To get these attributes in advance, the attribute request can be used.

  For example, [RFC8994], Section 6.11.7.2 specifies how the attribute request is used to signal to the pledge the acp-node-name field required for enrollment into an ACP domain.
- Attribute Response (4): This **MUST** contain the attributes to be included in the subsequent certification request.
- Certificate Request (5): This **MUST** contain the authenticated self-contained object ensuring both proof of possession of the corresponding private key and proof of identity of the requester.
- Certificate Response (6): This **MUST** contain on success the requested certificate and **MAY** include further information, like certificates of intermediate CAs.
- Certificate Confirm (7): An optional confirmation sent after the requested certificate has been received and validated. It contains a positive or negative confirmation by the pledge to the PKI whether the certificate was successfully enrolled and fits its needs.
- PKI/Registrar Confirm (8): An acknowledgment by the PKI that **MUST** be sent on reception of the Cert Confirm.

The generic messages described above may be implemented using any certificate enrollment protocol that supports authenticated self-contained objects for the certificate request as described in Section 3 and tunneling over TLS. Examples are available in Section 5.

Note that the optional certificate confirmation by the pledge to the PKI described above is independent of the mandatory enrollment status telemetry done between the pledge and the registrar in the final phase of BRSKI-AE, described next.

4.2.4. Pledge - Registrar Enrollment Status Telemetry

The enrollment status telemetry is performed as specified in [RFC8995].

In BRSKI this is described as part of the enrollment step, but due to the generalization on the enrollment protocol described in this document its regarded as a separate phase here.

4.3. Enhancements to the Endpoint Addressing Scheme of BRSKI

BRSKI-AE provides generalizations to the addressing scheme defined in BRSKI [RFC8995], Section 5 to accommodate alternative enrollment protocols that use authenticated self-contained objects for certification requests. As this is supported by various existing enrollment protocols, they can be employed without modifications to existing PKI RAs/CAs supporting the respective enrollment protocol (see also Section 5).

The addressing scheme in BRSKI for certification requests and the related CA certificates and CSR attributes retrieval functions uses the definition from EST [RFC7030], here on the example of simple enrollment: `"/.well-known/est/simpleenroll"`. This approach is generalized to the following notation: `"/.well-known/<enrollment-protocol>/<request>"` in which `<enrollment-protocol>` refers to a certificate enrollment protocol. Note that enrollment is considered here a message sequence that contains at least a certification request and a certification response. The following conventions are used to provide maximal compatibility with BRSKI:

- `<enrollment-protocol>`: **MUST** reference the protocol being used. Existing values include EST [RFC7030] as in BRSKI and CMP as in [I-D.ietf-lamps-lightweight-cmp-profile] and Section 5.1 below. Values for other existing protocols such as CMC or SCEP or CMC, or for newly defined protocols, require their own specifications for their use of the `<enrollment-protocol>` and `<request>` URI components and are outside the scope of this document.
- `<request>`: if present, this path component **MUST** describe, depending on the enrollment protocol being used, the operation requested. Enrollment protocols are expected to define their request endpoints, as done by existing protocols (see also Section 5).

Well-known URIs for various endpoints on the domain registrar are already defined as part of the base BRSKI specification or indirectly by EST. In addition, alternative enrollment endpoints **MAY** be supported at the registrar.

A pledge **SHOULD** use the endpoints defined for the enrollment protocol(s) that it is capable of. It will recognize whether its preferred protocol or the request that it tries to perform is supported by the domain registrar by sending a request to its preferred enrollment endpoint according to the above addressing scheme and evaluating the HTTP status code in the response.

The following list of endpoints provides an illustrative example for a domain registrar supporting several options for EST as well as for CMP to be used in BRSKI-AE. The listing contains the supported endpoints to which the pledge may connect for bootstrapping. This includes the voucher handling as well as the enrollment endpoints. The CMP-related enrollment endpoints are defined as well-known URIs in CMP Updates [I-D.ietf-lamps-cmp-updates] and the Lightweight CMP Profile [I-D.ietf-lamps-lightweight-cmp-profile].

```
</brski/voucherrequest>,ct=voucher-cms+json
</brski/voucher_status>,ct=json
</brski/enrollstatus>,ct=json
</est/cacerts>;ct=pkcs7-mime
</est/csrattrs>;ct=pkcs7-mime
</est/fullcmc>;ct=pkcs7-mime
</cmp/getcacerts>;ct=pkixcmp
</cmp/getcertreqtemplate>;ct=pkixcmp
</cmp/initialization>;ct=pkixcmp
</cmp/p10>;ct=pkixcmp
```

5. Instantiation to Existing Enrollment Protocols

This section maps the requirements to support proof of possession and proof of identity to selected existing enrollment protocols and provides further aspects of instantiating them in BRSKI-AE.

5.1. BRSKI-CMP: Instantiation to CMP

Note: Instead of referring to CMP as specified in [RFC4210] and [I-D.ietf-lamps-cmp-updates], this document refers to the Lightweight CMP Profile [I-D.ietf-lamps-lightweight-cmp-profile] because the subset of CMP defined there is sufficient for the functionality needed here.

When using CMP, the following specific implementation requirements apply (cf. Figure 3).

- CA Certs Request
  - Requesting CA certificates over CMP is **OPTIONAL**.
    If supported, it **SHALL** be implemented as specified in [I-D.ietf-lamps-lightweight-cmp-profile], Section 4.3.1.

- Attribute Request
  - Requesting certificate request attributes over CMP is **OPTIONAL**.
    If supported, it **SHALL** be implemented as specified in [I-D.ietf-lamps-lightweight-cmp-profile], Section 4.3.3.

Note that alternatively the registrar **MAY** modify the contents of requested certificate contents as specified in [I-D.ietf-lamps-lightweight-cmp-profile], Section 5.2.3.2.

- Certificate Request
  - Proof of possession **SHALL** be provided as defined in the Lightweight CMP Profile [I-D.ietf-lamps-lightweight-cmp-profile], Section 4.1.1 (based on CRMF) or [I-D.ietf-lamps-lightweight-cmp-profile], Section 4.1.4 (based on PKCS#10).
    In certificate response messages the `caPubs` field, which generally in CMP may convey CA certificates to the requester, **SHOULD NOT** be used.
  - Proof of identity **SHALL** be provided by using signature-based protection of the certification request message as outlined in [I-D.ietf-lamps-lightweight-cmp-profile], Section 3.2 using the IDevID secret.

- Certificate Confirm
  - Explicit confirmation of new certificates to the RA/CA **MAY** be used as specified in the Lightweight CMP Profile [I-D.ietf-lamps-lightweight-cmp-profile], Section 4.1.1.
    Note that independently of certificate confirmation within CMP, enrollment status telemetry with the registrar will be performed as described in BRSKI [RFC8995], Section 5.9.4.

- If delayed delivery of responses (for instance, to support asynchronous enrollment) within CMP is needed, it **SHALL** be performed as specified in the Lightweight CMP Profile [I-D.ietf-lamps-lightweight-cmp-profile], Section 4.4 and [I-D.ietf-lamps-lightweight-cmp-profile], Section 5.1.2.
- Due to the use of self-contained signed request messages providing end-to-end security and the general independence of CMP of message transfer, the way in which messages are exchanged by the registrar with backend PKI (RA/CA) components is out of scope of this document. It can be freely chosen according to the needs of the application scenario (e.g., using HTTP). CMP Updates [I-D.ietf-lamps-cmp-updates] and the Lightweight CMP Profile [I-D.ietf-lamps-lightweight-cmp-profile] provide requirements for interoperability.

BRSKI-AE with CMP can also be combined with Constrained BRSKI [I-D.ietf-anima-constrained-voucher], using CoAP for enrollment message transport as described by CoAP Transport for CMPV2 [I-D.ietf-ace-cmpv2-coap-transport]. In this scenario, of course the EST-specific parts of [I-D.ietf-anima-constrained-voucher] do not apply.

5.2. Other Instantiations of BRSKI-AE

Further instantiations of BRSKI-AE can be done. They are left for future work.

In particular, CMC [RFC5272] (using its in-band source authentication options) and SCEP [RFC8894] (using its 'renewal' option) could be used.

The fullCMC variant of EST sketched in [RFC7030], Section 2.5 might also be used here. For EST-fullCMC further specification is necessary.

6. IANA Considerations

This document does not require IANA actions.

7. Security Considerations

The security considerations as laid out in BRSKI [RFC8995] apply for the discovery and voucher exchange as well as for the status exchange information.

The security considerations as laid out in the Lightweight CMP Profile [I-D.ietf-lamps-lightweight-cmp-profile] apply as far as CMP is used.

8. Acknowledgments

We thank Eliot Lear or his contributions as a co-author at an earlier draft stage.

We thank Brian E. Carpenter, Michael Richardson, and Giorgio Romanenghi for their input and discussion on use cases and call flows.

Moreover, we thank Michael Richardson and Rajeev Ranjan for their reviews.

9. References

9.1. Normative References

[I-D.ietf-ace-cmpv2-coap-transport] Sahni, M. and S. Tripathi, "CoAP Transfer for the Certificate Management Protocol", Work in Progress, Internet-Draft, draft-ietf-ace-cmpv2-coap-transport-05, 19 September 2022, <https://www.ietf.org/archive/id/draft-ietf-ace-cmpv2-coap-transport-05.txt>.

[I-D.ietf-anima-constrained-voucher] Richardson, M., Van der Stok, P., Kampanakis, P., and E. Dijk, "Constrained Bootstrapping Remote Secure Key Infrastructure (BRSKI)", Work in Progress, Internet-Draft, draft-ietf-anima-constrained-voucher-18, 11 July 2022, <https://www.ietf.org/archive/id/draft-ietf-anima-constrained-voucher-18.txt>.

[I-D.ietf-lamps-cmp-updates] Brockhaus, H., von Oheimb, D., and J. Gray, "Certificate Management Protocol (CMP) Updates", Work in Progress, Internet-Draft, draft-ietf-lamps-cmp-updates-23, 29 June 2022, <https://www.ietf.org/archive/id/draft-ietf-lamps-cmp-updates-23.txt>.

[I-D.ietf-lamps-lightweight-cmp-profile] Brockhaus, H., von Oheimb, D., and S. Fries, "Lightweight Certificate Management Protocol (CMP) Profile", Work in Progress, Internet-Draft, draft-ietf-lamps-lightweight-cmp-profile-14, 5 October 2022, <https://www.ietf.org/archive/id/draft-ietf-lamps-lightweight-cmp-profile-14.txt>.

[IEEE.8802.1AR_2014] IEEE, "ISO/IEC/IEEE International Standard for Information technology -- Telecommunications and information exchange between systems -- Local and metropolitan area networks -- Part 1AR: Secure device identity", IEEE 8802.1AR-2014, DOI 10.1109/ieeestd.2014.6739984, 13 February 2014, <http://ieeexplore.ieee.org/servlet/opac?punumber=6739982>.

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

[RFC4210] Adams, C., Farrell, S., Kause, T., and T. Mononen, "Internet X.509 Public Key Infrastructure Certificate Management Protocol (CMP)", RFC 4210, DOI 10.17487/RFC4210, September 2005, <https://www.rfc-editor.org/info/rfc4210>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8366] Watsen, K., Richardson, M., Pritikin, M., and T. Eckert, "A Voucher Artifact for Bootstrapping Protocols", RFC 8366, DOI 10.17487/RFC8366, May 2018, <https://www.rfc-editor.org/info/rfc8366>.

[RFC8995] Pritikin, M., Richardson, M., Eckert, T., Behringer, M., and K. Watsen, "Bootstrapping Remote Secure Key Infrastructure (BRSKI)", RFC 8995, DOI 10.17487/RFC8995, May 2021, <https://www.rfc-editor.org/info/rfc8995>.

9.2. Informative References

[IEC-62351-9] International Electrotechnical Commission, "IEC 62351 - Power systems management and associated information exchange - Data and communications security - Part 9: Cyber security key management for power system equipment", IEC 62351-9, May 2017.

[ISO-IEC-15118-2] International Standardization Organization / International Electrotechnical Commission, "ISO/IEC 15118-2 Road vehicles - Vehicle-to-Grid Communication Interface - Part 2: Network and application protocol requirements", ISO/IEC 15118-2, April 2014.

[NERC-CIP-005-5] North American Reliability Council, "Cyber Security - Electronic Security Perimeter", CIP 005-5, December 2013.

[OCPP] Open Charge Alliance, "Open Charge Point Protocol 2.0.1 (Draft)", December 2019.

[RFC2986] Nystrom, M. and B. Kaliski, "PKCS #10: Certification Request Syntax Specification Version 1.7", RFC 2986, DOI 10.17487/RFC2986, November 2000, <https://www.rfc-editor.org/info/rfc2986>.

[RFC4211] Schaad, J., "Internet X.509 Public Key Infrastructure Certificate Request Message Format (CRMF)", RFC 4211, DOI 10.17487/RFC4211, September 2005, <https://www.rfc-editor.org/info/rfc4211>.

[RFC5272] Schaad, J. and M. Myers, "Certificate Management over CMS (CMC)", RFC 5272, DOI 10.17487/RFC5272, June 2008, <https://www.rfc-editor.org/info/rfc5272>.

[RFC5652] Housley, R., "Cryptographic Message Syntax (CMS)", STD 70, RFC 5652, DOI 10.17487/RFC5652, September 2009, <https://www.rfc-editor.org/info/rfc5652>.

[RFC5929] Altman, J., Williams, N., and L. Zhu, "Channel Bindings for TLS", RFC 5929, DOI 10.17487/RFC5929, July 2010, <https://www.rfc-editor.org/info/rfc5929>.

[RFC6402] Schaad, J., "Certificate Management over CMS (CMC) Updates", RFC 6402, DOI 10.17487/RFC6402, November 2011, <https://www.rfc-editor.org/info/rfc6402>.

[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>.

[RFC8572] Watsen, K., Farrer, I., and M. Abrahamsson, "Secure Zero Touch Provisioning (SZTP)", RFC 8572, DOI 10.17487/RFC8572, April 2019, <https://www.rfc-editor.org/info/rfc8572>.

[RFC8894] Gutmann, P., "Simple Certificate Enrolment Protocol", RFC 8894, DOI 10.17487/RFC8894, September 2020, <https://www.rfc-editor.org/info/rfc8894>.

[RFC8994] Eckert, T., Ed., Behringer, M., Ed., and S. Bjarnason, "An Autonomic Control Plane (ACP)", RFC 8994, DOI 10.17487/RFC8994, May 2021, <https://www.rfc-editor.org/info/rfc8994>.

[RFC9148] van der Stok, P., Kampanakis, P., Richardson, M., and S. Raza, "EST-coaps: Enrollment over Secure Transport with the Secure Constrained Application Protocol", RFC 9148, DOI 10.17487/RFC9148, April 2022, <https://www.rfc-editor.org/info/rfc9148>.

[UNISIG-Subset-137] UNISIG, "Subset-137; ERTMS/ETCS On-line Key Management FFFIS; V1.0.0", December 2015, <https://www.era.europa.eu/sites/default/files/filesystem/ertms/ccs_tsi_annex_a_-_mandatory_specifications/set_of_specifications_3_etcs_b3_r2_gsm-r_b1/index083_-_subset-137_v100.pdf>. http://www.kmc-subset137.eu/index.php/download/

Appendix A. Using EST for Certificate Enrollment

When using EST with BRSKI, pledges interact via TLS with the domain registrar, which acts both as EST server and as the PKI RA. The TLS channel is mutually authenticated, where the pledge uses its IDevID certificate issued by its manufacturer.

Using BRSKI-EST has the advantage that the mutually authenticated TLS channel established between the pledge and the registrar can be reused for protecting the message exchange needed for enrolling the LDevID certificate. This strongly simplifies the implementation of the enrollment message exchange.

Yet the use of TLS has the limitation that this cannot provide auditability nor end-to-end authentication of the CSR by the pledge at a remote PKI RA/CA because the TLS session is transient and terminates at the registrar. This is a problem in particular if the enrollment is done via multiple hops, part of which may not even be network-based.

With enrollment protocols that use for CSRs self-contained signed objects, logs of CSRs can be audited because CSRs can be third-party authenticated in retrospect, whereas TLS connections can not.

Furthermore, the BRSKI registrars in each site have to be hardened so that they can be trusted to be the TLS initiator of the EST connection to the PKI RA/CA, and in result, their keying material needs to be managed with more security care than that of pledges because of trust requirements, for example they need to have the id-kp-cmcRA extended key usage attribute according to [RFC7030], see [RFC6402]. Impairment to a BRSKI registrar can result in arbitrarily many fake certificate registrations because real authentication and authorization checks can then be circumvented.

Relying on TLS authentication of the TLS client, which is supposed to be the certificate requester, for a strong proof of origin for the CSR is conceptually non-trivial and can have implementation challenges. EST has the option to include in the certification request, which is a PKCS#10 CSR, the so-called tls-unique value [RFC5929] of the underlying TLS channel. This binding of the proof of identity of the TLS client to the proof of possession for the private key requires specific support by TLS implementations.

The registrar terminates the security association with the pledge at TLS level and thus the binding between the certification request and the authentication of the pledge. In BRSKI [RFC8995], the registrar typically doubles as the PKI RA and thus also authenticates the CSR and filters/denies requests from non-authorized pledges. If the registrar cannot do the final authorization checks on the CSR and needs to forward it to the PKI RA, there is no end-to-end proof of identity and thus the decision of the PKI RA must trust on the pledge authentication performed by the registrar. If successfully authorized, the CSR is passed to the PKI CA, which will issue the domain-specific certificate (LDevID). If in this setup the protocol between the on-site registrar and the remote PKI RA is also EST, this approach requires online or at least intermittent connectivity between registrar and PKI RA, as well as availability of the PKI RA for performing the final authorization decision on the certification request.

A further limitation of using EST as the certificate enrollment protocol is that due to using PKCS#10 structures in enrollment requests, the only possible proof-of-possession method is a self-signature, which excludes requesting certificates for key types that do not support signing. CMP, for instance, has special proof-of-possession options for key agreement and KEM keys, see [RFC4210], Section 5.2.8.

Appendix B. Application Examples

This informative annex provides some detail to the application examples listed in Section 1.3.

B.1. Rolling Stock

Rolling stock or railroad cars contain a variety of sensors, actuators, and controllers, which communicate within the railroad car but also exchange information between railroad cars building a train, with track-side equipment, and/or possibly with backend systems. These devices are typically unaware of backend system connectivity. Managing certificates may be done during maintenance cycles of the railroad car, but can already be prepared during operation. Preparation will include generating certification requests, which are collected and later forwarded for processing, once the railroad car is connected to the operator backend. The authorization of the certification request is then done based on the operator's asset/inventory information in the backend.

UNISIG has included a CMP profile for enrollment of TLS client and server X.509 certificates of on-board and track-side components in the Subset-137 specifying the ETRAM/ETCS on-line key management for train control systems [UNISIG-Subset-137].

B.2. Building Automation

In building automation scenarios, a detached building or the basement of a building may be equipped with sensors, actuators, and controllers that are connected with each other in a local network but with only limited or no connectivity to a central building management system. This problem may occur during installation time but also during operation. In such a situation a service technician collects the necessary data and transfers it between the local network and the central building management system, e.g., using a laptop or a mobile phone. This data may comprise parameters and settings required in the operational phase of the sensors/actuators, like a component certificate issued by the operator to authenticate against other components and services.

The collected data may be provided by a domain registrar already existing in the local network. In this case connectivity to the backend PKI may be facilitated by the service technician's laptop. Alternatively, the data can also be collected from the pledges directly and provided to a domain registrar deployed in a different network as preparation for the operational phase. In this case, connectivity to the domain registrar may also be facilitated by the service technician's laptop.

B.3. Substation Automation

In electrical substation automation scenarios, a control center typically hosts PKI services to issue certificates for Intelligent Electronic Devices (IEDs) operated in a substation. Communication between the substation and control center is performed through a proxy/gateway/DMZ, which terminates protocol flows. Note that [NERC-CIP-005-5] requires inspection of protocols at the boundary of a security perimeter (the substation in this case). In addition, security management in substation automation assumes central support of several enrollment protocols in order to support the various capabilities of IEDs from different vendors. The IEC standard IEC62351-9 [IEC-62351-9] specifies mandatory support of two enrollment protocols: SCEP [RFC8894] and EST [RFC7030] for the infrastructure side, while the IED must only support one of the two.

B.4. Electric Vehicle Charging Infrastructure

For electric vehicle charging infrastructure, protocols have been defined for the interaction between the electric vehicle and the charging point (e.g., ISO 15118-2 [ISO-IEC-15118-2]) as well as between the charging point and the charging point operator (e.g. OCPP [OCPP]). Depending on the authentication model, unilateral or mutual authentication is required. In both cases the charging point uses an X.509 certificate to authenticate itself in TLS channels between the electric vehicle and the charging point. The management of this certificate depends, among others, on the selected backend connectivity protocol. In the case of OCPP, this protocol is meant to be the only communication protocol between the charging point and the backend, carrying all information to control the charging operations and maintain the charging point itself. This means that the certificate management needs to be handled in-band of OCPP. This requires the ability to encapsulate the certificate management messages in a transport-independent way. Authenticated self-containment will support this by allowing the transport without a separate enrollment protocol, binding the messages to the identity of the communicating endpoints.

B.5. Infrastructure Isolation Policy

This refers to any case in which network infrastructure is normally isolated from the Internet as a matter of policy, most likely for security reasons. In such a case, limited access to external PKI services will be allowed in carefully controlled short periods of time, for example when a batch of new devices is deployed, and forbidden or prevented at other times.

B.6. Sites with Insufficient Level of Operational Security

The RA performing (at least part of) the authorization of a certification request is a critical PKI component and therefore requires higher operational security than components utilizing the issued certificates for their security features. CAs may also demand higher security in the registration procedures from RAs, which domain registrars with co-located RAs may not be able to fulfill. Especially the CA/Browser forum currently increases the security requirements in the certificate issuance procedures for publicly trusted certificates, i.e., those placed in trust stores of browsers, which may be used to connect with devices in the domain. In case the on-site components of the target domain cannot be operated securely enough for the needs of an RA, this service should be transferred to an off-site backend component that has a sufficient level of security.

Appendix C. History of Changes TBD RFC Editor: please delete

List of reviewers (besides the authors):

- Toerless Eckert (document shepherd)
- Michael Richardson
- Rajeev Ranjan

From IETF draft ae-02 -> IETF draft ae-03:

- In response to review by Toerless Eckert,
  - many editorial improvements and clarifications as suggested, such as the comparison to plain BRSKI, the description of offline vs. synchronous message transfer and enrollment, and better differentiation of RA flavors.
  - clarify that for transporting certificate enrollment messages between pledge and registrar, the TLS channel established between these two via the join proxy is used and the enrollment protocol **MUST** support this.
  - clarify that the enrollment protocol chosen between pledge and registrar **MUST** also be used for the upstream enrollment exchange with the PKI.
  - extend the description and requirements on how during the certificate enrollment phase the registrar **MAY** handle requests by the pledge itself and otherwise **MUST** forward them to the PKI and forward responses to the pledge.
- Change "The registrar **MAY** offer different enrollment protocols." to "The registrar **MUST** support at least one certificate enrollment protocol ..."
- In response to review by Michael Richardson,
  - slightly improve the structuring of the Message Exchange Section 4.2 and add some detail on the request/response exchanges for the enrollment phase
  - merge the 'Enhancements to the Addressing Scheme' Section 4.3 with the subsequent one: 'Domain Registrar Support of Alternative Enrollment Protocols'
  - add reference to SZTP (RFC 8572)
  - extend venue information
  - convert output of ASCII-art figures to SVG format
  - various small other text improvements as suggested/provided
- Remove the tentative informative instantiation to EST-fullCMC
- Move Eliot Lear from co-author to contributor, add him to the acknowledgments
- Add explanations for terms such as 'target domain' and 'caPubs'
- Fix minor editorial issues

From IETF draft ae-01 -> IETF draft ae-02:

- Architecture: clarify registrar role including RA/LRA/enrollment proxy
- CMP: add reference to CoAP Transport for CMPV2 and Constrained BRSKI
- Include venue information

From IETF draft 05 -> IETF draft ae-01:

- Renamed the repo and files from anima-brski-async-enroll to anima-brski-ae
- Added graphics for abstract protocol overview as suggested by Toerless Eckert
- Balanced (sub-)sections and their headers
- Added details on CMP instance, now called BRSKI-CMP

From IETF draft 04 -> IETF draft 05:

- David von Oheimb became the editor.
- Streamline wording, consolidate terminology, improve grammar, etc.
- Shift the emphasis towards supporting alternative enrollment protocols.
- Update the title accordingly - preliminary change to be approved.
- Move comments on EST and detailed application examples to informative annex.
- Move the remaining text of section 3 as two new sub-sections of section 1.

From IETF draft 03 -> IETF draft 04:

- Moved UC2-related parts defining the pledge in responder mode to a separate document. This required changes and adaptations in several sections. Main changes concerned the removal of the subsection for UC2 as well as the removal of the YANG model related text as it is not applicable in UC1.
- Updated references to the Lightweight CMP Profile.
- Added David von Oheimb as co-author.

From IETF draft 02 -> IETF draft 03:

- Housekeeping, deleted open issue regarding YANG voucher-request in UC2 as voucher-request was enhanced with additional leaf.
- Included open issues in YANG model in UC2 regarding assertion value agent-proximity and CSR encapsulation using SZTP sub module).

From IETF draft 01 -> IETF draft 02:

- Defined call flow and objects for interactions in UC2. Object format based on draft for JOSE signed voucher artifacts and aligned the remaining objects with this approach in UC2 .
- Terminology change: issue #2 pledge-agent -> registrar-agent to better underline agent relation.
- Terminology change: issue #3 PULL/PUSH -> pledge-initiator-mode and pledge-responder-mode to better address the pledge operation.
- Communication approach between pledge and registrar-agent changed by removing TLS-PSK (former section TLS establishment) and associated references to other drafts in favor of relying on higher layer exchange of signed data objects. These data objects are included also in the pledge-voucher-request and lead to an extension of the YANG module for the voucher-request (issue #12).
- Details on trust relationship between registrar-agent and registrar (issue #4, #5, #9) included in UC2.
- Recommendation regarding short-lived certificates for registrar-agent authentication towards registrar (issue #7) in the security considerations.
- Introduction of reference to agent signing certificate using SKID in agent signed data (issue #11).

- Enhanced objects in exchanges between pledge and registrar-agent to allow the registrar to verify agent-proximity to the pledge (issue #1) in UC2.
- Details on trust relationship between registrar-agent and pledge (issue #5) included in UC2.
- Split of use case 2 call flow into sub sections in UC2.

From IETF draft 00 -> IETF draft 01:

- Update of scope in Section 1.2 to include in which the pledge acts as a server. This is one main motivation for use case 2.
- Rework of use case 2 to consider the transport between the pledge and the pledge-agent. Addressed is the TLS channel establishment between the pledge-agent and the pledge as well as the endpoint definition on the pledge.
- First description of exchanged object types (needs more work)
- Clarification in discovery options for enrollment endpoints at the domain registrar based on well-known endpoints in Section 4.3 do not result in additional /.well-known URIs. Update of the illustrative example. Note that the change to /brski for the voucher-related endpoints has been taken over in the BRSKI main document.
- Updated references.
- Included Thomas Werner as additional author for the document.

From individual version 03 -> IETF draft 00:

- Inclusion of discovery options of enrollment endpoints at the domain registrar based on well-known endpoints in Section 4.3 as replacement of section 5.1.3 in the individual draft. This is intended to support both use cases in the document. An illustrative example is provided.
- Missing details provided for the description and call flow in pledge-agent use case UC2, e.g. to accommodate distribution of CA certificates.
- Updated CMP example in Section 5 to use Lightweight CMP instead of CMP, as the draft already provides the necessary /.well-known endpoints.
- Requirements discussion moved to separate section in Section 3. Shortened description of proof-of-identity binding and mapping to existing protocols.
- Removal of copied call flows for voucher exchange and registrar discovery flow from [RFC8995] in Section 4 to avoid doubling or text or inconsistencies.
- Reworked abstract and introduction to be more crisp regarding the targeted solution. Several structural changes in the document to have a better distinction between requirements, use case description, and solution description as separate sections. History moved to appendix.

From individual version 02 -> 03:

- Update of terminology from self-contained to authenticated self-contained object to be consistent in the wording and to underline the protection of the object with an existing credential. Note that the naming of this object may be discussed. An alternative name may be attestation object.
- Simplification of the architecture approach for the initial use case having an offsite PKI.

- Introduction of a new use case utilizing authenticated self-contain objects to onboard a pledge using a commissioning tool containing a pledge-agent. This requires additional changes in the BRSKI call flow sequence and led to changes in the introduction, the application example,and also in the related BRSKI-AE call flow.
- Update of provided examples of the addressing approach used in BRSKI to allow for support of multiple enrollment protocols in Section 4.3.

From individual version 01 -> 02:

- Update of introduction text to clearly relate to the usage of IDevID and LDevID.
- Definition of the addressing approach used in BRSKI to allow for support of multiple enrollment protocols in Section 4.3. This section also contains a first discussion of an optional discovery mechanism to address situations in which the registrar supports more than one enrollment approach. Discovery should avoid that the pledge performs a trial and error of enrollment protocols.
- Update of description of architecture elements and changes to BRSKI in Section 4.1.
- Enhanced consideration of existing enrollment protocols in the context of mapping the requirements to existing solutions in Section 3 and in Section 5.

From individual version 00 -> 01:

- Update of examples, specifically for building automation as well as two new application use cases in Appendix B.
- Deletion of asynchronous interaction with MASA to not complicate the use case. Note that the voucher exchange can already be handled in an asynchronous manner and is therefore not considered further. This resulted in removal of the alternative path the MASA in Figure 1 and the associated description in Section 4.1.
- Enhancement of description of architecture elements and changes to BRSKI in Section 4.1.
- Consideration of existing enrollment protocols in the context of mapping the requirements to existing solutions in Section 3.
- New section starting Section 5 with the mapping to existing enrollment protocols by collecting boundary conditions.

Contributors

Eliot Lear
Cisco Systems
Richtistrasse 7
CH-8304 Wallisellen
Switzerland
Phone: +41 44 878 9200
Email: lear@cisco.com

Authors' Addresses

David von Oheimb (EDITOR)
Siemens AG
Otto-Hahn-Ring 6
81739 Munich
Germany
Email: david.von.oheimb@siemens.com
URI: https://www.siemens.com/

Steffen Fries
Siemens AG
Otto-Hahn-Ring 6
81739 Munich
Germany
Email: steffen.fries@siemens.com
URI: https://www.siemens.com/

Hendrik Brockhaus
Siemens AG
Otto-Hahn-Ring 6
81739 Munich
Germany
Email: hendrik.brockhaus@siemens.com
URI: https://www.siemens.com/