ANIMA WG
Internet-Draft
Intended status: Standards Track
Expires: May 6, 2020

S. Fries
H. Brockhaus
Siemens
E. Lear
Cisco Systems

November 3, 2019

Support of asynchronous Enrollment in BRSKI
draft-fries-anima-brski-async-enroll-02

Abstract

This document discusses an enhancement of automated bootstrapping of a remote secure key infrastructure (BRSKI) to operate in domains featuring no or only timely limited connectivity to backend services offering enrollment functionality, specifically a Public Key Infrastructure (PKI). In the context of deploying new devices the design of BRSKI allows for online (synchronous object exchange) and offline interactions (asynchronous object exchange) with a manufacturer's authorization service. For this it utilizes a self-contained voucher to transport the domain credentials as a signed object to establish an initial trust between a pledge and the target deployment domain. The currently supported enrollment protocol for request and distribution of deployment domain specific device certificates provides only limited support for asynchronous PKI interactions. This memo motivates the enhancement of supporting self-contained objects for certificate management by using an abstract notation. This allows off-site operation of PKI services outside the deployment domain of the pledge. This addresses specifically scenarios, in which the final authorization of certification request of a pledge cannot be made in the deployment domain and is therefore delegated to a operator backend. The goal is to enable the usage of existing and potentially new PKI protocols supporting self-containment for certificate management.

Status of This Memo

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 May 6, 2020.

Copyright Notice

Table of Contents

1. Introduction

BRSKI as defined in [I-D.ietf-anima-bootstrapping-keyinfra] specifies a solution for secure zero-touch (automated) bootstrapping of devices (pledges) in a target deployment domain. This includes the discovery of network elements in the deployment domain, time synchronization, and the exchange of security information necessary to establish trust between a pledge and the domain and to adopt a pledge as new network and application element. Security information about the deployment domain, specifically the deployment domain certificate (domain root certificate), is exchanged utilizing vouchers as defined in [RFC8366]. These vouchers are self-contained (signed) objects, which may be provided online (synchronous) or offline (asynchronous) via the domain registrar to the pledge and originate from a manufacturer's authorization service (MASA). The manufacturer signed voucher contains the target domain certificate and can be verified by the pledge due to the possession of a manufacturer root certificate. It facilitates the enrollment of the pledge in the deployment domain and is used to establish trust from the pledge to the domain.

For the enrollment of devices BRSKI relies on EST [RFC7030] to request and distribute deployment domain specific device certificates. EST in turn relies on a binding of the certification request to an underlying TLS connection between the EST client and the EST server. According to BRSKI the domain registrar acts as EST server and is also acting as registration authority (RA) or local registration authority (LRA). The binding to TLS is used to protect the exchange of a certification request (for an LDevID certificate) and to provide data origin authentication to support the authorization decision for processing the certification request. The TLS connection is mutually authenticated and the client side authentication bases on the pledge's manufacturer issued device certificate (IDevID certificate). This approach requires an on-site availability of the RA as PKI component and/or a local asset or inventory management system performing the authorization decision based on tupel of the certification request and the pledge authentication using the IDevID certificate, to issue a domain specific certificate to the pledge. This is due to the EST server terminating the security association with the pledge and thus the binding between the certification request and the authentication of

the pledge. This type of enrollment utilizing an online connection to the PKI is considered as synchronous enrollment.

For certain use cases on-site support of a RA/CA component and/or an asset management is not available and rather provided by an operators backend and may be provided timely limited or completely through offline interactions. This may be due to higher security requirements for operating the certification authority. The authorization of a certification request based on an asset management in this case will not / can not be performed on-site at enrollment time. Enrollment, which cannot be performed in a (timely) consistent fashion is considered as asynchronous enrollment in this document. It requires the support of a store and forward functionality of certification request together with the requester authentication information. This enables processing of the request at a later point in time. A similar situation may occur through network segmentation, which is utilized in industrial systems to separate domains with different security needs. Here, a similar requirement arises if the communication channel carrying the requester authentication is terminated before the RA/CA handling the certification request. If a second communication channel is opened to forward the certification request to the issuing RA/ CA, the requester authentication information needs to be bound to the certification request. This uses case is independent from the timly limitations of the first use case. For both cases, it is assumed that the requester authentication information is utilized in the process of authorization of a certification request. There are different options to perform store and forward of certification requests including the requester authentication information:

- Providing a trusted component (e.g., an LRA) in the deployment domain, which stores the certification request combined with the requester authentication information (based on the IDevID) and potentially the information about a successful proof of possession (of the corresponding private key) in a way prohibiting changes to the combined information. Note that the assumption is that the information elements may not be cryptographically bound together. Once connectivity to the backend is available, the trusted component forwards the certification request together with the requester information (authentication and proof of possesion) to the off-site PKI for further processing. It is assumed that the off-site PKI in this case relies on the local pledge authentication result and thus performs the authorization and issues the requested certificate. In BRSKI the trusted component may be the EST server residing co-located with the registrar in the deployment domain.

- o Utilization of self-contained objects binding the certification request and the requester authentication in a cryptographic way. This approach reduces the necessary trust in a domain component to storage and delivery. Unauthorized modifications of the requestor information (request and authentication) can be detected during the verification of the cryptographic binding of the self-contained object in the off-site PKI. An example for a self-contained object is a signed CMS wrapped object.

This document targets environments, in which connectivity to the PKI functionality is only temporary or not directly available by specifying support for handling self-contained objects supporting asynchronous enrollment. As it is intended to enhance BRSKI it is named BRSKI-AE, where AE stands for asynchronous enrollment. As BRSKI, BRSKI-AE results in the pledge storing a X.509 root certificate sufficient for verifying the domain registrar / proxy identity (LDevID CA Certificate) as well as an domain specific X.509 device certificate (LDevID EE certificate).

The goal is to enhance BRSKI to either allow other existing certificate management protocols supporting self-contained objects to be applied or to allow other types of encoding for the certificate management information exchange.

Note that in contrast to BRSKI, BRSKI-AE assumes support of multiple enrollment protocols on the infrastructure side, allowing the pledge manufacturer to select the most appropriate. Thus, BRSKI-AE can be applied for both, asynchronous and synchronous enrollment.

2. History of changes

From version 01 -> 02:

- o Update of introduction text to clearly relate to the usage of IDevID and LDevID.

- o Definition of the addressing approach used in BRSKI to allow for support of multiple enrollment protocols in Section 5.3. This section also contains a first discussion of an optional discovery mechanism to address situations in which the registrar supports more than one enrollment appraoch. Discovery should avoid that the pledge performs a trial and error of enrollment protocols.

- o Update of description of architecture elements and changes to BRSKI in Section 5.

- o Enhanced consideration of existing enrollment protocols in the context of mapping the requirements to existing solutions in Section 4.3 and in Section 7.

From version 00 -> 01:

- o Update of examples, specifically for building automation as well as two new application use cases in Section 4.2.

- o 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 5.

- o Enhancement of description of architecture elements and changes to BRSKI in Section 5.

- o Consideration of existing enrollment protocols in the context of mapping the requirements to existing solutions in Section 4.3.

- o New section starting Section 7 with the mapping to existing enrollment protocols by collecting boundary conditions.

3. 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 [RFC2119].

This document relies on the terminology defined in [I-D.ietf-anima-bootstrapping-keyinfra]. The following terms are defined additionally:

CA: Certification authority, issues certificates.

RA: Registration authority, an optional system component to which a CA delegates certificate management functions such as authorization checks.

LRA: Local registration authority, an optional RA system component with proximity to end entities.

IED: Intelligent Electronic Device (in essence a pledge).

on-site: Describes a component or service or functionality available in the target deployment domain.

off-site: Describes a component or service or functionality available in an operator domain different from the target deployment domain. This may be a central side, to which only a temporarily connection is available or which is in a different administrative domain.

asynchronous communication: Describes a timely interrupted communication between an end entity and a PKI component.

self-contained object: Describes an object, which is cryptographically bound to the IDevID EE credential of a pledge. The binding is assumed to be provided through a digital signature using the corresponding private key of the IDevID to wrap the actual object. Note that depending on the availability of a LDevID EE credential, the binding may also be achieved using corresponding private key of the LDevID. This can be utilized in for intstance in the context of an inital certification request or a certificate update.

synchronous communication: Describes a timely uninterrupted communication between an end entity and a PKI component.

4. Scope of solution

4.1. Supported environment

This solution is intended to be used in domains with limited support of on-site PKI services and comprises use cases in which:

- o there is no registration authority available in the deployment domain. The connectivity to the backend RA may only be temporarily available. A local store and forward device is used for the communication with the backend services.

- o authoritive actions of a LRA are limited and may not comprise authorization of certification requests of pledges. Final authorization is done at the RA residing in the backend operator domain.

- o the target deployment domain already uses a certificate management approach that shall be reused to be consistent throughout the lifecycle.

4.2. Application Examples

The following examples are intended to motivate the support of different enrollment approaches in general and asynchronous enrollment specifically, by introducing industrial applications

cases, which could leverage BRSKI as such but also require support of asynchronous operation as intended with BRSKI-AE.

4.2.1. Rolling stock

Rolling stock or railroad cars contain a variety of sensors, actuators, and controller, which communicate within the railroad car but also exchange information between railroad cars building a train or with a backend. These devices are typically unaware of backend connectivity. Managing certificates may be done during maintenance cycles of the railroad car, but can already be prepared during operation. The preparation may comprise the generation of certification requests by the components, which are collected and 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 operators asset/inventory information in the backend.

4.2.2. Building automation

In building automation a use case can be described by a detached building or the basement of a building equipped with sensor, actuators, and controllers connected, but with only limited or no connection to the centralized building management system. This limited connectivity may be during the installation time but also during operation time. During the installation in the basement, a service technician collects the necessary information from the basement network and provides them to the central building management system, e.g., using a laptop or even a mobile phone to transport the information. This information may comprise parameters and settings required in the operational phase of the sensors/actuators, like a certificate issued by the operator to authenticate against other components and services.

4.2.3. Substation automation

In substation automation a control center typically hosts PKI services to issue certificates for Intelligent Electronic Devices (IED)s in a substation. Communication between the substation and control center is done through a proxy/gateway/DMZ, which terminates protocol flows. Note that NERC CIP-005-5 [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 different enrollment protocols to facilitate the capabilities of IEDs from different vendors. The IEC standard IEC62351-9 [IEC-62351-9] specifies the mandatory support of two enrollment protocols, SCEP

[I-D.gutmann-scep] and EST [RFC7030] for the infrastructure side, while the IED must only support one of the two.

4.2.4. Electric vehicle charging infrastructure

For the electric vehicle charging infrastructure protocols have been defined for the interaction between the electric vehicle (EV) 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 authenticates uses an X.509 certificate to authenticate in the context of a TLS connection between the EV and the charging point. The management of this certificate depends (beyond others) on the selected backend connectivity protocol. Specifically in case of OCPP it is intended as single 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 is intended to be handled in-band of OCPP. This requires to be able to encapsulate the certificate management exchanges in a transport independent way. Self-containment will ease this by allowing the transport without a separate communication protocol. For the purpose of certificate management CMP [RFC4210]is intended to be used.

4.2.5. 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 resources will be allowed in carefully controlled short periods of time, for example when a batch of new devices are deployed, but impossible at other times.

4.2.6. Less operational security in the deployment domain

The registration point performing the authorization of a certificate request is a critical PKI component and therefore implicates higher operational security than other components utilizing the issued certificates for their security features. CAs may also demand higher security in the registration procedures. Especially the CA/Browser forum currently increases the security requirements in the certificate issuance procedures for publicly trusted certificates. There may be the situation that the deployment domain does not offer enough security to operate a registration point and therefore wants to transfer this service to a backend.

4.3. Requirement discussion and mapping to solution elements

For the requirements discussion it is assumed that the entity receiving the self-contained object in the deployment domain is not the authorization point for the certification request contained in the object. If the entity is the authorization point, BRSKI can be used directly. Note that BRSKI-AE could also be used in this case.

Based on the supported deployment environment described in Section 4.1 and the motivated application examples described in Section 4.2 the following base requirements are derived to support self-contained objects as container carrying the certification request and further information to support asynchronous operation. Moreover, potential solution examples (not complete) based on existing technology are provided with the focus on existing IETF standards track documents:

- o Certification requests are structures protecting at least integrity of the contained data combined with a proof-of-private-key-possession for locally generated key pairs. Examples for certification requests are:

  - * PKCS#10 [RFC2986]: Defines a structure for a certification request. The structure must be signed to ensure integrity protection and proof-of-private-key-possession. Hence, the signature is performed by using the private key of the requestor (corresponding to the contained public key).

  - * CRMF [RFC4211]: Defines a structure for the certification request. The structure typically contains an integrity protection and a proof of possession, in which a signature value is generated by using the corresponding private key to the contained public key. This self-signature may also be replaced by the RA after verification, if the RA intends to update or alter the request message.

  Note that the integrity of the certification request is bound to the public key contained in the certification request by performing the signature operation with the corresponding private key. In the considered application examples, this is not sufficient and needs to be bound to the existing credential of the pledge (IDevID). This binding supports the authorization decision for the certification request. The binding of data origin authentication to the certification request may be delegated to the management protocol.

- o The container carrying the certification request should support a binding to an existing credential (here IDevID) known to the peer

performing the authoriation of the certification request as proof of identity. The binding may be transport dependent if the endpoint at the next communication hop is authorizing the certification request. This requirements is addressed by existing enrollment protocols in different ways, for instance:

* EST [RFC7030]: Utilizes PKCS#10 to encode the certification request. The Certificate Signing Request (CSR) may contain a binding to the underlying TLS by including the tls-unique value in the self-signed CSR structure. The tls-unique value is one result of the TLS handshake. As the TLS handshake is performed mutually authenticated and the pledge utilized its IDevID for it, the proof of identity can be provided by the binding to the TLS session.

* SCEP [I-D.gutmann-scep]: Provides the option to utilize either an existing secret (password) or an existing certificate to protect the CSR based on SCEP Secure Message Objects using CMS ([RFC5652]). Note that the wrapping using an existing IDevID credential is referred to as re-enroll.

* CMP [RFC4210] Provides the option to utilize either an existing secret (password) or an existing certificate to protect the PKIMessage containing the certification request. The certification request is encoded utilizing CRMF. PKCS#10 is optionally supported. The proof of identity of the PKIMessage containing the certification request can be achieved by using IDevID credentials to calculate a signature over the header and the body of the PKIMessage utilizing the protectionAlg signaled in the PKIMessage header and the PKIProtection carrying the actual signature value.

* CMC [RFC5272] Provides the option to utilize either an existing secret (password) or an existing certificate to protect the certification request (either in CRMF or PKCS#10) based on CMS [RFC5652]). Here a FullCMCRequest can be used, which allows signing with an existing IDevID credential to provide a proof of identity.

o The container carrying the certification request should support transport independent protection using an existing credential of the pledge verifiable at the authorization point of the certification request (typically the RA in conjunction with an inventory). This requirements is addressed by existing enrollment protocols in different ways, for instance:

* EST [RFC7030]: Not supported natively. Requires support of FullCMCRequest.

* SCEP [I-D.gutmann-scep]: Not specified in SCEP, could be done using message wrapping with signature (based on CMS). Note that in the current definition of SCEP this could be supported using a re-enroll request.

* CMP [RFC4210]: Message wrapping with signature.

* CMC [RFC5272]: Message wrapping with signature.

Note that besides the already exisiting enrollment protocols there ongoing work in the ACE WG to define an encapsulation of EST in OSCORE to result in a TLS independent way of protecting EST. This approach [I-D.selander-ace-coap-est-oscore] is intended to be considered in the future as well.

5. Architectural Overview

To support asynchronous enrollment, the base system architecture defined in BRSKI [I-D.ietf-anima-bootstrapping-keyinfra] is changed to allow for off-site operation of the PKI components. The assumption for BRSKI-AE is that the authorization for a certification request is performed based on a self-contained object binding the the certification request to the authentication using the IDevID. In addition, the authorization may be handled by an inventory or asset management system residing in the backend of the domain operator as described in Section 4.1. This leads to changes in the placement or enhancements of the logical elements as shown in Figure 1.

```
                                          +-----------------------+
      +--------------Drop Ship--------------->| Vendor Service        |
      |                                   +-----------------------+
      |                                   | M anufacturer|        |
      |                                   | A uthorized  |Ownership|
      |                                   | S igning     |Tracker  |
      |                                   | A uthority   |        |
      |                                   +--------------+--------+
      |                                                  ^
      |                                                  |
      V                                                  |
   +--------+     ......................................  |
   |        |     .                                    .  |
   |        |     .  +------------+       +------------+  .  | BRSKI-
   |        |     .  |            |       |            |  .  | MASA
   | Pledge |     .  |   Join     |       | Domain     <-----+
   |        |     .  |   Proxy    |       | Registrar/ |  .
   |        <-------->............<------> Proxy      |  .
   |        |     .  |        BRSKI-AE    |            |  .
   | IDevID |     .  |            |       +------^-----+  .
   |        |     .  +------------+              |        .
   |        |     .                              |        .
   +--------+     ...............................|........
                   "on-site domain" components   |
                                                 |
                                                 |
    ............................................|.....................
    . +---------------------------+     +--------v-----------------+ .
    . | Public Key Infrastructure |<----+ PKI RA                   | .
    . | PKI CA                    |---->+ [(Domain) Registrar (opt)]| .
    . +---------------------------+     +--------+--^---------------+ .
    .                                            |  |                 .
    .                                   +--------v--+---------------+ .
    .                                   | Inventory (Asset)         | .
    .                                   | Management                | .
    .                                   +---------------------------+ .
    ...................................................................
            "off-site domain" components
```

Figure 1: Architecture overview of BRSKI-AE

The architecture overview in Figure 1 utilizes the same logical elements as BRSKI but with a different placement in the deplayment architecture for some of the elements. The main difference is the placement of the PKI RA/CA component, which is actually performing the authorization decision for the certification request message. Also shown is the connectivity of the RA/CA with an inventory management system, which is expected to be utilized in the

authorization decision. Note that this may also be an integrated functionality of the RA. Both components are placed in the off-site domain of the operator (not the deployment site directly), which may have no or only temporary connectivity to the deployment domain of the pledge. This is to underline the authorization decision for the certification request in the backend rather than in the deployment domain itself. The following list describes the components in the deployment domain:

- o Join Proxy: same functionality as described in BRSKI

- o Domain Registrar / Proxy: In general the domain registrar / proxy has a similar functionality regarding the imprinting of the pledge in the deployment domain to facilitate the communication of the pledge with the MASA and the PKI. Different is the authorization of the certification request. BRSKI-AE allows to perform this in the operators backend (off-site), even if the deployment domain has only temporary or no connectivity to an operator domain.

  - * Voucher exchange: The voucher exchange with the MASA via the domain registrar is performed as described in BRSKI [I-D.ietf-anima-bootstrapping-keyinfra] .

  - * Certificate enrollment: For the pledge enrollment the domain registrar in the deployment domain supports the adoption of the pledge to be part of the domain, but not necessarily to authorize the certification request provided during enrollment. This may be due to lack of authorization information in the deployment domain. If the authorization is done in the operator domain, the domain registrar is used to forward the certification request to the RA. Thus it basically works as a proxy. In the case of no connectivity, the domain registrar stores the certification request and forwards it to the RA upon connectivity. As this requires the certification request to be self-contained, the domain registrar needs functionality enhancements with respect to the support of alternative enrollment approaches supporting self-containment. To support alternative enrollment approaches (protocol options, protocols, encodings), it is necessary to enhance the addressing scheme at the domain registrar. This is addressed in section Section 5.3.

The following list describes the vendor related components/service outside the deployment domain:

- o MASA: general functionality as described in BRSKI. Assumption that the interaction with the MASA may be synchronous (voucher

request with nonce) or asynchronous (voucher request without nonce).

- o Ownership tracker: as defined in BRSKI.

The following list describes the operator related components/service operated in the backend:

- o PKI RA: Performs certificate management functions (validation of certification requests, interaction with inventory/asset management for authorization of certification requests, etc.) for issuing, updating, and revoking certificates for a domain as a centralized infrastructure for the operator.

- o PKI CA: Performs certificate generation by signing the certificate structure provided in the certification request.

- o Inventory (asset) management: contains information about the known devices belonging to the operator. Specifically, the inventory is used to provide the information to authorize issuing a certificate based on the certification request of the pledge. Note: the communication between the inventory (asset) management and the PKI components (RA/CA) are out of scope of this document.

- o (Domain) registrar: Optional component if the deployment domain does not feature a domain registrar but only a proxy. In this case it is involved in the certification request processing and is assumed to be co-located with the PKI RA.

5.1. Behavior of a pledge

The behavior of a pledge as described in [I-D.ietf-anima-bootstrapping-keyinfra] is kept with one exception. After finishing the imprinting phase (4) the enrollment phase (5) is performed with a method supporting self-contained objects. Using EST with simpleenroll as in BRSKI cannot be applied here, as it binds the pledge authentication with the existing IDevID to the transport channel rarther than the certification request object. This authentication is not visible / verifiable at the authorization point in the off-site domain. Section Section 7 discusses potential protocols and EST protocol options applicable.

5.2. Secure Imprinting using Vouchers

The described approach in [I-D.ietf-anima-bootstrapping-keyinfra] is kept as is.

5.3. Addressing Scheme for the Enrollment

The realization of BRSKI-AE requires enhancements to the addressing scheme defined in [I-D.ietf-anima-bootstrapping-keyinfra]. This is due to the additions of self-contained object handling to BRSKI. BRSKI itself utilizes EST as enrollment protocol, which can be enabled to support self-contained objects by utilizing the FullCMCRequest instead of the simple enroll. Besides EST there are further enrollment protocols, which also support the handling of self-contained objects and which can be employed here. The approach of BRSKI-AE is to allow additional enrollment options to be supported. For the provisioning of different enrollment options at the domain registrar, the addressing approach of BRSKI using a "/.well-known" tree from [RFC5785] is enhanced.

The current addressing scheme in BRSKI for the client certificate request function during the enrollment is using the definition from EST [RFC7030] "/.well-known/est/simpleenroll" This approach is generalized to the following notation: "/.well-known/enrollment-protocol/request" in which enrollment-protocol may be an already existing protocol or a newly defined approach. Note that enrollement is considered here as a sequence of at least a certification request and a certification response. In case of existing enrollment protocols the following notation is used ptoving compatibility to BRSKI:

- o enrollment-protocol: references EST [RFC7030] as in BRSKI directly or CMP, CMC, SCEP, or newly defined approaches as alternatives for support in BRSKI-AE.

- o request: depending on the utilized enrollment protocol, the request describes the required operation at the registrar side. For BRSKI the request would be a "simpleenroll" for the base behavior and a "FullCMCRequest" for the support of self-contained objects

/* to be done:

- o Consideration of different transport options. BRSKI utilizes EST over HTTP but there is also the definition of EST over CoAP. This has been defined in the draft from the ACE WG and utilizes coaps instead in https in the URI.

- o Definition of a mandatory enrollment protocol to be supported to ensure interoperability.

5.3.1. Discovery of Enrollment Protocol Support

If the registrar supports multiple enrollment protocols, specifically beyond the required mechanisms, it is more efficient to also support an optional discovery mechanism. By querying the registrar, the pledge gets an enumeration of potential options, based on the defined namespace.

/* the discover mechanism needs to be defined in terms of message exchanges. */

6. Protocol Flows

Based on BRSKI and the architectural changes the original protocol flow is divided into three phases showing commonalities and differences to the original approach as depicted in the following.

- o Discovery phase (same as BRSKI)
- o Voucher exchange with deployment domain registrar (same as BRSKI).
- o Enrollment phase (changed to accompany the application of self-contained objects for the enrollment).

6.1. Pledge - Registrar discovery and voucher exchange

The discovery phase is applied as specified in [I-D.ietf-anima-bootstrapping-keyinfra]. /* for discussion: is a reference to BRSKI sufficient here or is it helpful to provide additional information and the figure? */

```
+--------+         +---------+    +------------+     +------------+
| Pledge |         | Circuit |    | Domain     |     | Vendor     |
|        |         | Join    |    | Registrar  |     | Service    |
|        |         | Proxy   |    |  (JRC)     |     | (MASA)     |
+--------+         +---------+    +------------+     +------------+
  |                    |                  |           Internet |
  |<-RFC4862 IPv6 addr |                  |                    |
  |<-RFC3927 IPv4 addr | Appendix A       |  Legend            |
  |------------------->|                  |  C - circuit       |
  | optional: mDNS query| Appendix B      |      join proxy    |
  | RFC6763/RFC6762    |                  |  P - provisional   |
  |<-------------------|                  |    TLS connection  |
  | GRASP M_FLOOD      |                  |                    |
  |   periodic broadcast|                 |                    |
  |<------------------>C<----------------->|                    |
  |              TLS via the Join Proxy    |                    |
  |<--Registrar TLS server authentication---|                    |
[PROVISIONAL accept of server cert]        |                    |
  P---X.509 client authentication---------->|                    |
  P                    |                  |                    |
  P--Voucher Request (w/nonce for voucher)->|                    |
  P                    |       /--->      |                    |
  P                    |       |      see Figure 3 below        |
  P                    |       \---->     |                    |
  P<------voucher--------------------------|                    |
[verify voucher, imprint]                  |                    |
  |----------------------------------------->|                    |
  |      [voucher status telemetry]        |<-device audit log--|
  |                    |       [verify audit log and voucher]   |
  |<------------------------------------------>|                    |
```

Figure 2: Pledge discovery of domain registrar discovery and voucher exchange

6.2. Registrar - MASA voucher exchange

The voucher exchange is performed as specified in [I-D.ietf-anima-bootstrapping-keyinfra]. /* for discussion: is a reference to BRSKI sufficient here or is it helpful to provide additional information and the figure? */

```
+--------+         +---------+    +------------+     +------------+
| Pledge |         | Circuit |    | Domain     |     | Vendor     |
|        |         | Join    |    | Registrar  |     | Service    |
|        |         | Proxy   |    |  (JRC)     |     | (MASA)     |
+--------+         +---------+    +------------+     +------------+
  P                     |       /--->       |                    |
  P                     |       |      [accept device in domain] |
  P                     |       |      [contact Vendor]          |
  P                     |       |           |--Pledge ID-------->|
  P                     |       |           |--Domain ID-------->|
  P                     |       |           |--optional:nonce--->|
  P                     |       |           |     [extract DomainID]
  P                     |    optional:      |     [update audit log]
  P                     |      can occur in advance if nonceless |
```

Figure 3: Domain registrar - MASA voucher exchange

6.3. Pledge - Registrar - RA/CA certificate enrollment

The enrollment for BRSKI-AE will be performed using a self-contained object. According to the abstract requirements from [I-D.ietf-anima-bootstrapping-keyinfra]. This object containes the certification request and shall support at least the following properties:

- o Proof of Possession: utilizing the private key corresponding to the public key contained in the certification request.

- o Proof of Identity: utilizing the existing IDevID credential to generate a signature of the inital certification request. certificate updates may utilize the LDevID credential.

- o /* further parameter to be specified if necessary */.

```
   +--------+         +---------+    +------------+     +------------+
   | Pledge |         | Circuit |    | Domain     |     | Operator   |
   |        |         | Join    |    | Registrar  |     | RA/CA      |
   |        |         | Proxy   |    |  (JRC)     |     | (OPKI)     |
   +--------+         +---------+    +------------+     +------------+
     /-->                                       |                    |
     |---------- Request CA Certs ------------>|                    |
     |              [if connection to operator domain is available] |
     |                                          |-Request CA Certs ->|
     |                                          |<- CA Certs Response|
     |<-------- CA Certs Response--------------|                    |
     |---------- Attribute Request ----------->|                    |
     |              [if connection to operator domain is available] |
     |                                          |Attribute Request ->|
     |                                          |<-Attribute Response|
     |<--------- Attribute Response -----------|                    |
     /-->                                       |                    |
     |-------------- Cert Request ------------>|                    |
     |              [if connection to operator domain is available] |
     |                                          |--- Cert Request -->|
     |                                          |<-- Cert Response --|
     /-->                                       |                    |
     |          [if connection to operator domain is not available] |
     |                                          |                    |
     |<---------- Cert Waiting ----------------|                    |
     |-- Cert Polling (with orig request ID) ->|                    |
     |              [if connection to operator domain is available] |
     |                                          |--- Cert Request -->|
     |                                          |<-- Cert Response --|
     /-->                                       |                    |
     |<------------- Cert Response ------------|                    |
     |-------------- Cert Confirm ------------>|                    |
     |                                         /-->                 |
     |                                          |[optional]          |
     |                                          |--- Cert Confirm -->|
     |                                          |<-- PKI Confirm ----|
     |<------------- PKI/Registrar Confirm ----|                    |
```

Figure 4: Certificate enrollment

The following list provides an abstract description of the flow depicted in Figure 4.

o CA Cert Request: The pledge SHOULD request the full distribution of CA Certificates message. This ensures that the pledge has the complete set of current CA certificates beyond the pinned-domain-cert.

- o Attribute Request: Typically, the automated bootstrapping occurs without local administrative configuration of the pledge. Nevertheless, there are cases, in which the pledge should also include additional attributes specific to the deployment domain into the certification request. To get these attributes in advance, the attribute request SHOULD be used.

- o Cert Request: certification request message (to be done: reference to PKCS#10 or CRMF, proof of possession, pledge authentication)

- o Cert Response: certification response message containing the requested certificate and potentially further information like certificates of intermediary CAs on the certification path.

- o Cert Waiting: waiting indication for the pledge to retry after a given time. For this a request identifier is necessary. This request identifier may bei either part of the enrollment protocol or build based on the certification request.

- o Cert Polling: querying the registrar, if the certificate request was already processed; can be answered either with another Cert Waiting, or a Cert Response.

- o Cert Confirm: confirmation message from pledge after receiving and verifying the certificate.

- o PKI/Registrar Confirm: confirmation message from PKI/registrar about reception of the pledge's certificate confirmation.

/* to be done:

- o Investigation into handling of certificate request retries.

- o Message exchange description.

- o Confirmation message (necessary? optional? from Registrar and/or PKI?).

7. Example mappings to existing enrollment protocols

This sections maps the requirements and the approach described in Section 6.3 to already existing enrollment protocols. Note that that the work in the ACE WG described in [I-D.selander-ace-coap-est-oscore] may be considered here as well, as it also adresses the encapsulation of EST in a way to make it independent from the underlying TLS using OSCORE resulting in a self-contained object.

7.1. EST Handling

When using the EST protocol [RFC7030], the following constrains should be observed:

- o Proof of possession is provided by using the specified PKCS #10 structure in the request method.

- o For proof of identit only the /fullcmc endpoint should be used with a fullcmc request. This contains sufficient information for the RA/CA to make an authorization decision on the received certification request. Note that EST references CMC [RFC5272] for the definition of the full PKI request. For proof of identity, the signature of the SignedData of the Full PKI Request would be calculated using the IDEVID credential of the pledge. /*TBD: in this case the binding to the underlying TLS connection may not be necessary */

- o When the RA/CA is not available, as per [RFC7030] Section 4.2.3, a 202 return code should be returned by the Join Registrar. The pledge in this case would retry with the same PKCS#10 request as in the initial simpleentroll run. Note that if the TLS connection is teared down for the waiting time, the PKCS#10 request would need to be rebuild as it contains the unique identifier (tls_unique) from the underlying TLS connection for the binding.

7.2. CMP Handling

When using the CMP protocol [RFC4210], the following constrains should be observed:

- o For proof of possession, the defined approach in CMP [RFC4210] section 4.3 should be supported. This can be achieved by using either CRMF or PKCS#10 to specify the certification request.

- o Proof of identity can be provided by using the MSG_SIG_ALG to protect the certificate request message with signatures as outlined in section D.5.

- o When the CA/CA is not available, as per [RFC4210] Section 5.2.3, a waiting indication should be returned in the PKIStatus by the Join Registrar. The pledge in this case would retry using the PollReqContent with a request identifier certReqId provided in the initial CertRequest message as specified in section 5.3.22.

8. IANA Considerations

This document requires the following IANA actions:

/* to be done: IANA consideration to be included for the defined namespaces in Section 5.3. */

9. Privacy Considerations

/* to be done: clarification necessary */

10. Security Considerations

/* to be done: clarification necessary */

11. Acknowledgements

We would like to thank the various reviewers for their input, in particular Brian E. Carpenter, Giorgio Romanenghi, Oskar Camenzind for their input and discussion on use cases and call flows.

12. References

12.1. Normative 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-29 (work in progress), October 2019.

[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>.

[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>.

[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>.

12.2. Informative References

[I-D.gutmann-scep]
Gutmann, P., "Simple Certificate Enrolment Protocol", draft-gutmann-scep-14 (work in progress), June 2019.

[I-D.selander-ace-coap-est-oscore]
Selander, G., Raza, S., Furuhed, M., and M. Vucinic, "Protecting EST payloads with OSCORE", draft-selander-ace-coap-est-oscore-02 (work in progress), March 2019.

[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 , 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", April 2018.

[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>.

[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>.

[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>.

[RFC5785] Nottingham, M. and E. Hammer-Lahav, "Defining Well-Known Uniform Resource Identifiers (URIs)", RFC 5785, DOI 10.17487/RFC5785, April 2010, <https://www.rfc-editor.org/info/rfc5785>.

Authors' Addresses

Steffen Fries
Siemens AG
Otto-Hahn-Ring 6
Munich, Bavaria 81739
Germany

Email: steffen.fries@siemens.com
URI: http://www.siemens.com/

Hendrik Brockhaus
Siemens AG
Otto-Hahn-Ring 6
Munich, Bavaria 81739
Germany

Email: hendrik.brockhaus@siemens.com
URI: http://www.siemens.com/

Eliot Lear
Cisco Systems
Richtistrasse 7
Wallisellen CH-8304
Switzerland

Phone: +41 44 878 9200
Email: lear@cisco.com