Device Enrollment in IETF protocols -- a roadmap
draft-richardson-enrollment-roadmap-00

Abstract

This document provides an overview of enrollment or imprinting mechanisms in current IETF protocols.

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 July 30, 2018.

Copyright Notice

Copyright (c) 2018 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.
1. Introduction

There are numerous mechanisms being proposed to solve the problem of securely introducing a new devices into an existing managed network.

This document provides an overview of the different mechanisms showing what technologies are common. The document starts with a diagram showing the various components and how they go together to form five enrollment scenarios.
2. Components of enrollment solutions

This diagram is taken from [I-D.ietf-anima-bootstrapping-keyinfra], which is where this work started.

Five major components are described:

1. **pledge**: The node that is attempting to enroll.

2. **proxy**: A node that is within one layer-2 hop of the pledge that is helping.

3. **domain registrar**: the Join Registrar/Coordinator (JRC) that will determine eligibility of the pledge.

4. **MASA**: the representative of the manufacturer that has a pre-established trust relationship with the pledge.
5. the domain PKI (if any)

3. Map of Enrollment solution
| 6tisch | 6tisch | Transition to |
| minimal | zero | Constrained |
| security | touch | Bootstrap |
| | | BRSKI |
| | | Netconf |

* constrained voucher * * JSON format voucher *
* (CBOR) * *

* constrained object *
* security (OSCORE) *

* COSE-8152 *
* standard signature (CMS - RFC5652) *

* EST-COAPS * EST-HTTPS *
* EDHOC * w/DTLS sec. * TLS sec. *

* constrained object *
* security (OSCORE) *

* call-home *
* ssh/tls *
* .usbkey *

* IPIP proxy, stateless *

---

^ | ^ | ^ | ^

Richardson
Expires July 30, 2018
4. Components

4.1. generic voucher semantics

The abstract semantics of the voucher, described in YANG, are in [I-D.ietf-anima-voucher].

4.2. constrained voucher

The semantics of the constrained voucher, represented in CBOR, are described in [I-D.richardson-anima-ace-constrained-voucher].

This document does NOT yet have a home.

4.3. JSON format voucher

The semantics of the basic voucher, represented in JSON, are described in [I-D.ietf-anima-voucher].

4.4. COSE-8152

In constrained systems the voucher is signed using the COSE mechanism described in [RFC8152].

4.5. standard signature (CMS)

In un-constrained systems the voucher is signed using the Cryptographic Message Syntax (CMS) described in [RFC5652].

4.6. EDHOC

On constrained and challenged networks, the session key management can be formed by [I-D.selander-ace-cose-ecdhe].

This document does NOT have a home.

The CoAP-EST layer on top is described by [I-D.vanderstok-ace-coap-est]

4.7. EST-COAPS 2/DTLS sec(urity)

On unconstrained networks, the session key management is provided by [RFC6347]. The CoAP-EST layer on top is described by [I-D.vanderstok-ace-coap-est].

The ACE WG has agreed to adopt this document.
4.8. EST-HTTPS TLS security

On unconstrained networks with unconstrained nodes, the EST layer and session key management is described by [RFC7030] as modified by [I-D.ietf-anima-bootstrapping-keyinfra] (BRSKI).

4.9. constrained object security (OSCORE)

On constrained networks with constrained nodes, the CoAP transactions are secured by [I-D.ietf-core-object-security] using symmetric keys. The symmetric key may be pre-shared (for 6tisch minimal security), or MAY be derived using EDHOC.

4.10. Pledge traffic proxy mechanisms

Traffic between the Pledge and the JRC does not flow directly as the pledge does not typically have a globally reachable address, nor does it have any network access keys (whether WEP, WPA, 802.1x, or 802.15.4 keys).

Communication between the pledge and JRC is mediated by a proxy. This is primarily to protect the network against attacks. The proxy mechanism is provided by as many nodes as can afford to as a benefit to the network, and therefore MUST be as light weight as possible. There are therefore stateless mechanisms and stateful mechanisms. The costs of the various methods is analyzed in [I-D.richardson-anima-state-for-joinrouter].

4.10.1. COAP proxy, stateless

The CoAP proxy mechanism uses the OSCORE Context Hint to statelessly store the address of the proxy within the CoAP structure. It is described in [I-D.ietsf-6tisch-minimal-security].

4.11. DTLS proxy

There has been no specific DTLS specific stateless proxy described, although the mechanism described by the Thread Group is being considered, if it can be referenced easily.

4.12. IPIP proxy, stateless

An IPIP proxy mechanism uses a layer of IP-in-IP header (protocol 98) to encapsulate the traffic between Join Proxy and JRC. It has some complexities to implement on typical POSIX platforms. It is intended to be described in [I-D.ietsf-6tisch-dtsecurity-zerotouch-join], in an Appendix. Another home for the text is also desired.
4.13. circuit proxy stateful

   The circuit proxy method utilizes either an application layer
gateway (which in canonical 1990-era implementation requires a
process per connection), or the use of NAT66. It maintains some
state for each connection whether TCP or UDP.

   It is this most expensive and most easily abused, but also the most
widely available, code-wise.

5. call-home ssh/tls/usbkey

   The NETCONF call-home mechanism assumes that the device can get basic
connectivity, enough for an out "outgoing" TCP connection to the
manufacturer.

6. manufacturer authorized signing authority (MASA)

   The MASA is the manufacturers anchor of the manufacturer/pledge trust
relationship that is established at the factory where the pledge is
built.

7. Enrollment Mechanisms

7.1. NETCONF

   The NETCONF WG is describing this in [I-D.ietf-netconf-zerotouch]
document.

7.2. BRSKI

   The ANIMA WG is describing this in

7.3. Transition to Constrained Bootstrap

   The bulk of this work has no home as yet. It is distinguished from
BRSKI in that it uses DTLS (rather than TLS) and constrained (CBOR)
vouchers.
   It is distinguished from 6tisch Zero Touch in that uses CMS to sign
rather than COSE.

   The ACE WG is going to adopt [I-D.vanderstok-ace-coap-est].
7.4. 6tisch Zero Touch

The 6tisch WG is describing this in

7.5. 6tisch minimal security

The 6tisch WG is describing this in
[I-D.ietf-6tisch-minimal-security] document. This mechanism does
enrollment in a single request/response message, but requires at
least one "touch" to pre-share symmetric keys.

All other methods are considered zero "touch".

8. Security Considerations

This document includes a tradeoff of the security attributes of the
different protocols, and so the entire document contains security
advice.

9. IANA Considerations

This document does not define any new protocols, and therefore does
not have any IANA Considerations.

10. Acknowledgements

TBD

11. References

11.1. Normative References

[I-D.ietf-6tisch-dtsecurity-zerotouch-join]
Richardson, M. and B. Damm, "6tisch Zero-Touch Secure Join
protocol", draft-ietf-6tisch-dtsecurity-zerotouch-join-01
(work in progress), October 2017.

[I-D.ietf-6tisch-minimal-security]
Vucinic, M., Simon, J., Pister, K., and M. Richardson,
"Minimal Security Framework for 6TiSCH", draft-ietf-
6tisch-minimal-security-04 (work in progress), October
2017.
[I-D.ietf-anima-bootstrapping-keyinfra]

[I-D.ietf-anima-voucher]

[I-D.ietf-core-object-security]

[I-D.ietf-netconf-zerotouch]

[I-D.richardson-anima-ace-constrained-voucher]
Richardson, M., "Constrained Voucher Profile for Bootstrapping Protocols", draft-richardson-anima-ace-constrained-voucher-02 (work in progress), December 2017.

[I-D.selander-ace-cose-ecdhe]

[I-D.vanderstok-ace-coap-est]

[ieee802-1AR]

Informative References


Michael Richardson
Sandelman Software Works

Email: mcr+ietf@sandelman.ca