Network Working Group
Internet-Draft
Intended status: Informational
Expires: August 23, 2021

E. Lear
Cisco Systems

February 19, 2021

Discovering And Accessing Software Bills of Materials
draft-lear-iotops-onboard-intr-00

Abstract

With various onboarding methods being built out, one aspect that has been overlooked is the trust relationship between the deployment and the manufacturer. This document asks questions about how that trust should be established, and how it can be leveraged.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

A number of network onboarding technologies are currently beginning to mature in the market place. The questions they all seek to answer are these:

- o How does the device know that it should join a particular network?

- o How does the network know that it should trust a particular device?

Let's examine two protocols to understand how these questions are answered and what questions may also be interesting to ask. We will begin with looking at the Wifi Alliance's Device Provisioning Protocol, and then take a gander at Bootstrapping Remote Secure Key Infrastructure (BRSKI)[I-D.ietf-anima-bootstrapping-keyinfra], with a quick review of the operation model of each. We focus our attention to zero touch provisioning.

Zero touch provisioning in this context means that the device receives network credentials that it will use to bidirectionally authenticate with an appropriate network without any human direction or validation at the time that the device is first powered at the deployment.

1.1. Device Provisioning Protocol

As is described in its specification, Device Provisioning Protocol or DPP makes use of a public/private key pair. The private key is stored in the device, and the public key is provided to the user out of band.

In the current specification, either side may initiate communciation, but for battery saving reasons, it is generally preferred that the

endpoint initiate, and thus be in a position to disable its transceiver at other times.

Several validations then take place. The network is able to prove to the endpoint that it has the correct public key, and the device is able to prove to the network that it has the correct private key. Thus, mutual authentication has taken place. The next exchange allows for appropriate credentials to be provisioned in the device, such as a trust anchor or an SAE password.

As previously mentioned, the public key is provided to the user out of band. In this sense, the public key is effectively a password, in that anyone who holds it may onboard the device. If the user would ned to scan the public key from a QR code or via OCR, we would not call such a step "hands free". If the user needs to agree to onboarding a device at the time it is enabled, then here again we would not call this "hands free".

This leaves one additional possibility: communication of the public key via electronic means for the device having been deployed. The DPP standard doesn't discuss this method. This may provide a means for zero-touch deployment. In this context, we might consider the device that receives and stores public keys a form of a registrar.

While Internet connectivity is not required for DPP to function on its own, transmission of an public key would require some connectivity at some point.

The assumed endstate in all of this is that the device will be able to authenticate to the network without the Internet being available. While this may not be important in some cases, such as with devices whose applications require Internet availability to function, it would will be critical for other cases, such as disconnected environments with critical control functions.

1.2. Bootstrapping Remote Key Infratructure (BRSKI)

BRSKI's flow is somewhat different. In this case, the endpoint sends a voucher request to a registrar in the local deployment, who adorns the request with is public key information. This request is then forwarded to the device's manufacturer, who can take whatever choices it will to determine whether the device belongs in a particular local deployment. Those choices include:

- Nothing. It can just sign the voucher request, logging the request.

o Validation that a particular device belongs in a particular deployment.

The first step is problematic for wireless deployments because a device would simply join the first network it heard, if it could authenticate, and there is no reason to believe that the first network would be the correct network.

The second step is not standardized. It is possible that an out-of-band introduction is taking place on the first transaction, but that would not be a zero-touch flow.

At some point, the deployment must also assign establish that the device belongs on its network. This too is not specified by the standard.

2. Discussion

In the case of both BRSKI and DPP, once the device is onboarded by the network, no Internet connectivity is required. This is important, as a matter of resiliency.

BRSKI and DPP barely differ the gaps they have to get to zero-touch provisioning. In BRSKI's case, it's about establishing trust between the registrar and the manufacturer, a one time affair, and then then later the registrar having some reason to believe that particular device belongs within a deployment. In the case of DPP, once one decides that keys are going to be delivered in advance, whatever service receives them has to have been introduced to the service sending them.

One could perhaps envision a system in which credentials are transmitted at time of sale to a registrar. In the case of BRSKI, this would involve careful configuration of voucher requests, in which the voucher is "nonceless", but yet bound to a particular deployment.

This leads us to another concern.

3. Resale and Transfer

We should assume that device ownership and use will change over time. This presents some additional problems. We assume, for purposes of this discussion that both DPP and BRSKI have some form of a registrar function. We also assume that zero-touch may or may not require a device reset (a'la pin in the hole, type).

There are several ways to think about this problem:

- seller transmits something to the buyer registrar as part of a transaction.

- seller provides buyer an artifact as part of a sale.

- original manufacturer records the sale.

- original manufacturer simply notes the transfer.

Let's dispense with the last option. It suffers all the same wireless problems as the original case. Therefore, it is not further considered in this discussion.

If the seller transmits something to the buyer registrar as part of a transaction, this means that the seller must have identified the buyer registrar. That in itself may be a trick. If the seller provides the buyer with an artifact, the buyer must do something with it. Both of these methods suffer a particular problem: if the original owner went out of business, there is no chance for any transfer to take place.

If the original manufacturer records the sale, this means that the new owner would have to either know to query the manufacturer, or that the manufacturer would have enough information to send an appropriate credential to the new owner. This _also_ means that the manufacturer is still in business.

3.1. Choices, Choices

The above models are not necessarily mutually exclusive. That is, it might be possible to rely on automated means when they exist, and otherwise rely on less automated means when they do not exist. This is somewhat problematic for BRSKI, in that somehow or another a voucher needs to be generated.

Alternatively we might ponder an additional actor whose role it is to safeguard transfer credentials in the form of an escrow agent. The existence of such an actor introduces a number of questions:

- How would the buyer know to use a particular escrow agent?

- Is there an incentive model that would bring such an agent into creation? After all, someone has to pay for such a service.

- How is transaction privacy maintained (or is _that_ how the service gets paid for)?

Another model, and the author writes this with great trepidation, is the use of Merkle trees to record a transaction. This has the benefit of being able to establish previous ownership, but has the risks that the previous owner is no longer in existence to provide an assertion that the device has transfered. On the third hand, perhaps a claim by a known party is enough. Such transactions have privacy implications as well, and there has to be an incentive model to maintain a distributed ledger.

4. Beware too much mechanism

As this area of work advances, there will be the temptation to add a vast amount of mechanism. The previous supposition of the use of Merkel trees is a great example of just that. From an IoT device standpoint, it's all but interolerable. We are, after all, talking about a codespace that is generally considered to be constrainted.

5. This is really only about network onboarding

But that may not be the only problem. Rekeying will occasionally happen for various reasons. How this takes places will depend on the authentication mechanism used in a deployment. If EAP-TLS or TEAP are used, the presumption is that there will be an EST server or similar available for credential renewal.

For private shared keys, there currently is no great answer to this question.

6. 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-45 (work in progress), November 2020.

[I-D.ietf-anima-brski-async-enroll]
Fries, S., Brockhaus, H., Lear, E., and T. Werner, "Support of asynchronous Enrollment in BRSKI (BRSKI-AE)", draft-ietf-anima-brski-async-enroll-01 (work in progress), January 2021.

Appendix A. Changes from Earlier Versions

Draft -00:

- o Initial revision

Author's Address

Eliot Lear
Cisco Systems
Richtistrasse 7
Wallisellen CH-8304
Switzerland

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