Secure Pairing & Context Management

Johannes Gilger, Ulrike Meyer
Research Group IT-Security, RWTH Aachen University
E-Mail: {gilger,meyer}@umic.rwth-aachen.de

ABSTRACT

In this document, we present two of the problems of secure IoT deployment we think are interesting areas for future work. The first part of this document will talk about secure pairing protocols*, and the remainder of the document will give some thoughts on* authorization and policy frameworks *for deployed networks.*

1 INTRODUCTION

Work within the IETF CoRE WG has progressed, even on the security front, and multiple I-Ds on security and secure deployment of Smart Objects have been written. On the question of securing the connections between Smart Objects, there are numerous options (L2 security, compressed IPSec, CoAP with DTLS binding), with the CoAP/DTLS suite still being worked on. Whatever the exact result of this standardization process, it can be expected that multiple authentication models will be supported (None, CA-based, User-Verified), catering to the different needs and especially capabilities of Smart Objects.

We think that it is important to identify realistic assumptions to be made of Smart Objects in different settings (home automation, factory control, public space installations) in terms of the hardware and interfaces they posses. We want to look at the complete deployment process and classify it by the amount of user interaction needed, necessary skills for installation personnel, maximum time to bootstrap each node and resilience against attacks.

As we do not want to focus on the actual protocols used for securing communication and the key agreement process, we will focus on the steps before and after the initial key agreement takes place: Pairing and Authorization. The pairing stage will create or verify parameters to be used with existing key agreement protocols. After a security association has been established, each device has to create a security policy which specifies what other devices are allowed to do.

2 SECURE PAIRING

Secure pairing is the process of establishing a security association between two parties, in the absence of any other context. In our case, we understand pairing to happen between a newly installed Smart Object and a trust center or any other device which is already part of some system. We do not want to talk about *ad-hoc pairing* where neither party is part of a larger network.

With Smart Objects, we assume the two parties to be paired to be an *interface-constrained device* and a device with multiple user-friendly interfaces, called either the *bootstrapping device* or the *trust center*. We furthermore have the distinct situation that pairing should establish a secure connection between the two parties, but it does not have to mutually authenticate them, because the interface-constrained device can not put any meaning as to the identity of the trust center.

A lot of the literature on pairing assumes devices with some kind of interface, or *out-of-band channel* to be more precise. Some of these will certainly be applicable to Smart Objects which have either a push-button or a LED (such as BEDA or other visual pairing schemes). But we also envision devices which have absolutely no out out-of-band channel to perform secure pairing, except maybe the presence of power. We think that with some effort and certain assumptions about the hardware, secure pairing schemes can be extended to this class of devices as well, without increasing the complexity of installing these devices.

Another important aspect, especially for publicly accessible interface-constrained devices is the lack of a reset mechanism, to bring the device back into an imprintable state. How can we ensure a device can be reset (e.g. using in-band signaling), but still can not be ”stolen” and subsequently used by an attacker, either during the initial bootstrapping or after it has been deployed?

3 CONTEXT MANAGEMENT

The pairing process establishes a context between two parties, in our case a constrained device and an operator or a network. There may be several ways to support operation by a third party. If the device is sold for example, it could simply be ”killed” (reset) and imprinted again, using the pairing process. In some cases this simple sequence might not be desirable however. For one, the sheer number of devices could prohibit repeating the manual part of the pairing process. Another reason might be usage scenarios where the device is not supposed to completely destroy all context with its original owner.

As a smart object is small, very cheap and has only a limited use by itself, it will not be used like other tools. There might be no reason for anyone to assume exclusive control over a smart object. The number and size of devices will make it impossible to manually distribute access rights to these devices as they are needed by multiple parties for different purposes. Last but not least, a central authority which manages these access relationships will not always be available.

We think it is important to investigate smart object deployments which are outside the scope of current discussion and frameworks. We think that this work poses some interesting theoretical questions but will also quickly arrive at the question of usability and of how it could be implemented.