IoT Security in the context of Semantic Interoperability

Darshak Thakore, CableLabs

Abstract: IoT promises to be a multi-billion dollar industry that will, over the next decade, deliver connected ecosystems of everyday things and dramatically change how we use technology in our daily lives. The two major hurdles/issues that are pointed out in almost any analysis of IoT are, "interoperability" and "security". In most cases, existing work on IoT has focused on these issues separately. Security in most cases is implemented in the form of a separate layer or a protocol that is bolted-on in the architecture. In this paper, we discuss the possibility of semantic interoperability of security itself, i.e. can the semantic information about a model also include its security characteristics as a first class member?

Introduction: The promise of IoT has resulted in a tremendous amount of work being done to develop standards and protocols that would allow manufacturers to build and deploy IoT products quickly and cost-effectively. However this "race-to-be-first" has resulted in a plethora of standards (and non-standards) being developed, to the point that the most salient feature of IoT Standards today is that there are so many to choose from. Needless to say, any article that talks about IoT invariably points out the issue of interoperability between these various devices (that conform to different standards). This is a painfully obvious problem and one that is getting prominent day by day. The problem of interoperability between ecosystems is aggravated by the fact that different ecosystems use different architectures for communication. For example the popularity of a RESTful architecture in web and web-based applications has resulted in IoT specific RESTful protocol stacks to be developed (e.g. CoAP). In contrast, certain protocols and stacks like MQTT, DDS etc existed for device-to-device communication even before IoT became the hype it is today. This has resulted in IoT data models being tightly coupled to the protocol stacks they are implemented over.

Besides interoperability the other issue that is invariably discussed is security. This is, in some regard an even bigger issue compared to interoperability. Lack of security can be the "Achilles heel" of the IoT promise simply due to the fact that an insecure IoT ecosystem can lead to actual physical harm. Interestingly, "interoperability" and "security" in most cases get discussed in a separate context. This is largely due to the fact that most architectures and protocol stacks that were used as a basis for developing IoT ecosystems had security as a separate layer that was applied/bolted on. In the sections below, we look at why semantic interoperability of IoT may require us to look at security, not as a separate problem but as the problem of interoperable security.

Current State of Security: Most IoT architectures today are a result of the evolution and/or modification of existing protocol architectures. For example, even though CoAP is a new protocol developed specifically for IoT, it was modeled based on the use of HTTP for a RESTful architecture. Similarly the AllJoyn framework is an evolution/adaptation over the D-Bus architecture that was primarily designed for IPC. These architectures/protocols were developed to provide functionality for a specific layer of the communication protocol stack. Security was considered a separate layer that was applied at a specific interface point and in most cases the protocols were designed to be ignorant of explicit security requirements. The same approach is

being taken in the design of the corresponding IoT architectures. From a software design point of view where the primary goal was for application software components to interact, this makes sense since it allows for separation of concerns and provides modularity.

Modeling of Things: IoT, however is a massive medley of "things". This means, in most cases the software component is tightly coupled with an actual physical entity. A physical entity is always "owned" by someone. The owner always provides some form of protection to the physical entity and decides who can do what with the thing. The model of a thing needs to encapsulate the notion of the "owner" within its expression. Current data models for IoT attempt to express this notion using existing security constructs (e.g. ACL's, authorization tokens) that were used for access control of software components. This poses the problem where even if the interoperability of the data model were solved, the varied security construct being used may still break interoperable communication since the security construct of one thing may not be compatible with the security construct of the other thing. For example, a thermostat that uses message bus architecture for its communication may need to communicate with a thermometer that is based on a RESTful architecture. In this scenario, we not only need to map the semantics of the data model produced by the thermometer to the semantics of the data model consumed by the thermostat, but we also need to ensure that the DTLS based security implemented by the thermometer is translated to the security construct used in the message bus. Most data model driven semantic interoperability approaches do not account for the mapping of the security constructs used by the data models.

Information modeling of "secure" things: Given that data models are typically too closely related to the underlying communication architecture, the information modeling of things might be the appropriate place where the security aspects of the thing can be expressed in an interoperable manner. For example, in a RESTful architecture, the data models are typically in the form of encoded resource representations associated with specific RESTful methods that can be invoked on the resource. The resource itself is considered an abstract concept. However, we may want to brainstorm the following:

1. What would the information model of the resource itself look like? (as opposed to its representation)
2. Would it be possible to model the resource itself such that the expression not only captures the characteristics/data of the resource but also the semantics of how the resource can be securely operated upon

In general, as we start thinking about semantic interoperability for IoT, it may we worthwhile to consider how the semantics of security of a thing can also be expressed in an interoperable manner.