IoT Security and the role of Manufacturers: A Story of Unrealistic Design Expectations

Mohit Sethi
NomadicLab, Ericsson Research, Finland
mohit@piuha.net

Tuomas Aura
Aalto University, Finland
tuomas.aura@aalto.fi

ABSTRACT

The *Internet of Things (IoT)* refers to an interconnected world where physical devices in our ambient environment are seamlessly integrated into the Internet. Although the needs for securing IoT devices is generally well understood and accepted, many steps are still needed for ensuring secure IoT deployments. Several new security protocols that rely on the active participation of IoT device manufacturers have emerged. In this position paper, we document common expectations of security protocols from IoT device manufacturers. Thereafter, we discuss the potential drawbacks of relying on manufacturers for security critical tasks. Finally, we discuss some important design considerations that protocol developers can use when specifying protocols that require active participation of the device manufacturers.

KEYWORDS

IoT, security, design, manufacturer

1 INTRODUCTION

Internet of Things (IoT) refers to the phenomena where physical devices in our ambient environment are seamlessly integrated into the Internet. The need for IoT security is widely accepted, and the Internet Engineering Task Force (IETF) has made several efforts. For example, the specification of Constrained Application Protocol (CoAP) secured with Datagram Transport Layer Security (DTLS). RFC 8576 [5] provides a detailed summary of all the IETF efforts towards making the Internet of Things more secure.

A trusted third-party for establishing security is sometimes necessary. Many security protocols at the IETF use X.509 certificates and Certificate Authorities (CAs) as trusted third parties. For example, Transport Layer Security (TLS) and Internet Key Exchange version 2 (IKEv2) are typically deployed with certificates and CAs for protecting large parts of the Internet infrastructure. While there are many national and regional certificate authorities, the market for globally trusted TLS/SSL server certificates is consolidated and lies in the hands of a few enterprises. We at the IETF have accepted our use of such trusted third parties and taken several steps to improve its security (for example, with certificate transparency [6]) and scalability (for example, with free certificates from Let's Encrypt[1]).

2 ROLE OF MANUFACTURERS IN SECURITY PROTOCOLS

None of the security protocols developed at the IETF thus far require active participation from the manufacturer of the devices running the protocol. Obviously, the manufacturers must implement IETF protocols as described in specifications. But we have not assigned additional responsibilities to the device manufacturers (as well as the operating system vendors running on top of those devices). Contrary to this practice, a plethora of new IoT security solutions currently being developed at the IETF (and elsewhere) are putting arguably unrealistic expectations on manufactures to provide a variety of additional services critical for security. A non-exhaustive list of expectations from IoT device manufacturers in some protocols include:

- *An online server*: The manufacturer may be expected to run an always online server. This server maybe involved during the initial configuration and bootstrapping of the IoT devices. In the worst case, the manufacturer may have significant additional responsibility, such as, tracking the ownership of the devices across ownership handovers.
- *Device certificates*: Manufacturers must purchase and install certificates. These certificates are not only required for any servers that the manufacturer is expected to operate, but also in many cases for each individual IoT device manufactured. Given that the number of IoT devices may range from a few hundreds to tens of thousands, the cost of having individual device certificates can be substantial.
- *Companion app on a smartphone*: Since IoT devices are often limited in the amount user interfaces available, protocols now require a companion device (such as a smart phone) running an application developed by the IoT device manufacturer for the protocol to correctly complete and for the IoT device to become operational. Typically, this entails that manufacturers must develop and maintain smartphone applications for both Android and iOS.
- *Software update*: IoT device manufacturers may need to run servers which provide software updates for their devices. This requirement is justified and essential since software bugs in devices are inevitable.

Advocates of protocols that rely heavily on the active participation of IoT device manufacturers might argue that this is necessary. Some might even claim that just as we managed to get global CAs to work reasonably well as trusted third parties, manufacturers will eventually improve their practices and provide services such as those listed above. Various steps to improve the security practices of IoT manufacturers are also underway. For example, the United Kingdom (UK) Department for Culture, Media and Sport (DCMS) and the National Cyber Security Centre (NCSC) for example have launched a "Code of Practice" for manufacturers of consumer IoT devices [1]. This code of practice provides basic guidelines such as discouraging the use of default passwords. Additionally, they are conducting a survey of consumer IoT device manufacturers and retailers [2, 3] to decide on how they can regulate the market better.

[1] https://letsencrypt.org/

Schneier [7] also writes in his blog about a new law in California which comes into effect in January 2020 and requires all "connected devices" to have a "reasonable security feature". Schneier correctly notes that the term "reasonable security" is unfortunately broad and can allow manufacturers to avoid implementing robust security in their IoT devices.

3 WHY ARE THESE EXPECTATIONS UNREALISTIC?

In this paper, we argue that requiring manufacturers to provide security critical services may be acceptable when the devices are deployed in large enterprise settings. These manufacturers already have experience in deploying secure devices and generally have a service agreement with the end customer. Service agreements typically take into account the cost of deploying secure code on devices and also guarantee that support services (such as online servers to track assets added and removed from the enterprise deployment) are provided for the agreed period. However, having the same expectations from all manufacturers is unrealistic and bound to fail. This is especially true for manufacturers of devices which are deployed in small office and home (SOHO) because of the following reasons:

- Not all manufacturers are willing or capable to do security critical tasks: Many manufacturers have expertise in manufacturing hardware at the lowest possible cost. Setting up servers and offering secure services is not something they are accustomed with.
- Costs can be excessive: As stated above, the costs of installing certificates on each IoT device can be prohibitive in some cases. It is also important to note that the cost of the device certificates themselves are marginal when compared to the cost of having a secure supply chain where key pairs and certificates are provisioned on each individual IoT device.
- Not all manufacturers are interested in, or have the capability of building and maintaining a dedicated smartphone application: Many manufacturers may initially provide a smartphone application for presenting the user with data collected from the IoT device (such as step count or calories burnt). However, they are mostly interested in providing a nice User Interface (UI) and rarely have any interest or experience in implementing security critical protocol specifications. They typically rely on the operating system (OS) to do this, for example, by using the Bluetooth implementation of the OS vendor to setup the secure communication. Security protocols that require a dedicated smartphone application from the device manufacturer also run the risk that the IoT devices can never be operational again once the manufacturer ceases to exist.
- Many IoT device manufacturers are startups launching products through crowd-funding platforms such as Kickstarter[2] and Indiegogo[3]. Obviously they are running on tight budgets. Some folks would argue that such gimmicky devices should be forbidden from getting deployed in any case. But we believe that such blanket bans have negative consequences as it limits the new entrants in the markets and gives the dominant players an unfair advantage. Additionally, while our politicians love to provide lip service on how entrepreneurship and small businesses must be encouraged, their actions and regulations have often favored the existing dominant players. The Internet Architecture Board (IAB) has recently been discussing this problem of Internet consolidation [4].

[2]https://www.kickstarter.com/
[3]https://www.indiegogo.com/

4 DESIGN CONSIDERATIONS FOR PROTOCOLS

We are having a hard time getting manufacturers to securely implement IETF protocols. Adding additional burden must come with more consideration. We can choose to define protocols that support a few dominant IoT device manufacturers and live happily in our closed gardens. Or we can introspect, and support open permissionless innovation (something in which we take a lot of pride at the IETF). This position paper suggests that the following considerations should be taken into account when developing protocols for IoT devices that involve some level of support of manufacturers:

(1) Companion app: does the protocol absolutely need a custom application from the manufacturer. Can it simply work with the standard web browser and camera application already available on the device. If information needs to be communicated out-of-band (OOB), for example via QR codes, is a custom URI format necessary? Can it just be a https web link that opens in the browser? Native camera apps on both iOS and Android now support scanning of QR codes and opening web links in browsers. Having a simple HTTPS url for example will require no native apps from the manufacturer. Also, OS vendors typically provide fairly robust security in their browsers.
(2) When considering the use of device certificates, it is important to consider the additional cost of individual device certificates and having a secure supply chain. Simply making security someone else's problem is definitely advisable when designing protocols. It is also important to consider the lifetime of the certificates? What happens when they expire or the cryptographic algorithms they rely on are broken?
(3) Reduce the number of things that need to be hardwired into the device for the protocol. Does the protocol absolutely require a URL or an IP address to be hardwired? This can potentially fail due to misconfiguration at the time of device manufacture and also once the company manufacturing looses the domain because of carelessness or when it ceases to exist.
(4) If an online server is required from the manufacturer, how easily can the role of this server be transferred to some place else? An idea protocol should support such ownership transfers since it is common for organizations to merge or get acquired.
(5) Limit the number of services that need to be provided by the manufacturer for the protocol to complete. Is it enough if the manufacturer attests that it is a verified device? Does the manufacturer also need to know and track the current user/owner of the device. It is also important to consider the

privacy consequence of revealing such operational information to the manufacturer.

(6) Build upon existing standard protocols. Many of them have lot of stable open source code available which the manufacturers can easily use.

(7) Allow freedom and flexibility to the device user/owner to choose someone other than manufacturer for providing the security critical services. It is equally plausible that some enterprises might want to run their own servers rather than relying on the device manufacturer for any services.

(8) Less manufacturer code on device. IoT devices are resource constrained in the terms of memory, computation and energy. Less code is not only beneficial because of these constraints, it also implies that less code needs to be updated by the manufacturer.

5 CONCLUSION

In this position paper, we discussed what kind of expectations new IoT security protocols may have from the device manufacturer. We then argued that these expectations may not be realistic for all types of deployments. Finally, we presented some design considerations that protocol developers could use. These design considerations specifically recommend that the manufacturer should not be treated as the bad guy who doesn't know what he is doing. Even though many of them have had woefully bad security practices with default passwords, we must develop protocols that don't require tons of people, services, servers, money, and code from manufacturers.

REFERENCES

[1] 2018. Code of Practice for Consumer IoT Security. https://www.gov.uk/government/publications/code-of-practice-for-consumer-iot-security. Accessed: April 2019.

[2] 2019. Regulating Security on the Internet of Things - for manufacturers. https://www.smartsurvey.co.uk/s/IoT_manufacturers/. Accessed: April 2019.

[3] 2019. Regulating Security on the Internet of Things - for Retailers. https://www.smartsurvey.co.uk/s/IoT_retailers/. Accessed: April 2019.

[4] Jari Arkko, Brian Trammell, Mark Nottingham, Christian Huitema, Martin Thomson, Jeff Tantsura, and Niels ten Oever. 2019. *Considerations on Internet Consolidation and the Internet Architecture.* Internet-Draft draft-arkko-iab-internet-consolidation-01. Internet Engineering Task Force. https://datatracker.ietf.org/doc/html/draft-arkko-iab-internet-consolidation-01 Work in Progress.

[5] Oscar Garcia-Morchon, Sandeep Kumar, and Mohit Sethi. 2019. Internet of Things (IoT) Security: State of the Art and Challenges. http://tools.ietf.org/rfc/rfc8576.txt. RFC 8576.

[6] Ben Laurie, Adam Langley, and Emilia Kasper. 2013. Certificate transparency. http://tools.ietf.org/rfc/rfc6962.txt. RFC 6962.

[7] Bruce Schneier. 2018. New IoT Security Regulations. https://www.schneier.com/blog/archives/2018/11/new_iot_securit.html. Accessed: April 2018.