Considerations for a lightweight, usable, and quantum-secure IoT

Oscar Garcia-Morchon, Ronald Rietman, Ludo Tolhuizen, Sauvik Bhattcharya, Jose-Luis Torre-Arce

Philips Group Innovation, Research, Eindhoven, The Netherlands

Abstract. This paper discusses the IoT security design space and the corresponding trade-offs between usability, cost, performance, and security for successfully deploying secure IoT applications. We discuss priorities for different functions such as encryption, integrity or key management in the context of some existing use cases and indicate some pragmatic solutions. We argue that for many IoT use cases, the efficient management of keys/credentials of devices is and will remain one of the key challenges in IoT scenarios due to both performance issues and constraints imposed by the use case and application lifecycle. For this key challenge, key pre-distribution schemes such with good collusion and quantum-resistance features can offer a good trade-off between security and performance.

Keywords Lightweight Cryptography, Post-Quantum Cryptography, Security Architecture, Device Lifecycle, IoT.

1 Introduction

The Internet of Things (IoT) is making more and more devices "smart" by connecting them while providing them with computing, sensing and acting capabilities.The purpose of making devices smart is to automate processes and manage (personal) information in a way that it can optimize services. A first IoT example refers to wireless medical sensors enabling ubiquitous healthcare monitoring, collecting health information in a variety of scenarios 24/7. This IoT application provides preventive healthcare and allows alerting the patient and his family of any potential health issues. Another example is about a smart lighting system that can automatically adapt the light settings to the existing lighting conditions and according to the user's preferences, thus increasing comfort and safety while reducing energy consumption.

Security plays a key role in the above mentioned applications for several reasons. First and most important, it ensures the privacy and safety of the users and their data. Second, it guarantees that the image and IPR of companies delivering new innovative solutions are not negatively affected. Third, security holes in IoT systems could even be exploited by cyber terrorists to carry out large scale attacks affecting whole nations. Finally, any of the above events will negatively affect

the world economy since it will prevent the wide-deployment of new innovative applications driving economic growth.
In order for any IoT application to be successfully deployed and commercially viable, the value that it brings has to be higher than capital and operational expenses. The usage of any security algorithm requires a given amount of memory, hardware, communication overhead and energy budget affecting the total cost of ownership. Another key consideration is that the impact of security procedures and cryptography on the normal operation of the IoT application is minimal since otherwise users will not adopt it.
Therefore, the challenge is to find a proper trade-off between security, usability, and cost.
To find this trade-off, the design of a secure IoT system enabling a new application requires a careful assessment of the business and privacy impacts of a security breach as well as the associated risks. This allows identifying the threats that can represent a relevant risk to the confidentiality, integrity or availability of the system or the exchanged or stored data. Once the risks are identified, mitigation strategies – based on cryptographic building blocks – can be selected to become part of the secure communication and software architecture that enables secure communication links and security functionality such as verified boot or secure firmware updates.
The cryptographic building blocks can fulfill different functions. On the one hand, there is functionality aiming at encryption, authentication and integrity of data, either in transit or at rest. Most frequently this is achieved by means of symmetric-key primitives such as hash functions, block ciphers, or stream ciphers relying on symmetric-keys. On the other hand, these keys need to be properly managed during their entire lifecycle in order to have a secure and usable IoT system. This is where architectural aspects and the usage of other (asymmetric) cryptographic primitives for key exchange and authentication play a fundamental role.

2 Security and cryptography design space

In order for the IoT application to be secure, operate as desired, and be cost effective, the designer of a new secure IoT system has to consider several aspects: The first one refers to the use case and the application lifecycle that are to be implemented. A new system enabling certain functionality should be viable business-wise, this means that its cost cannot be higher than the expected revenue. The technical system should not affect the usability in a negative manner, for instance, by requiring a complex configuration process or leading to unacceptable delays. Often, the successful deployment of a new application requires scalable solutions, i.e., regarding network size. Finally, many applications require long-term operation, either because hardware cannot be replaced, or because otherwise it is not commercially viable due to the associated costs.
This affects the performance of crypto primitives and the achieved security level due to a number of reasons. The second one refers to the available financial

budget. Even if almost any smart device could have a very sophisticated energy supply, this is not the case due to cost reasons. Therefore, many devices run on batteries. Since replacing batteries is both expensive and user unfriendly, it is desired that the smart devices are capable of running on batteries during their entire lifecycle that can be ten years or even more. Although devices could be outfitted with powerful communication interfaces, this is mostly not the case. Communication is usually wireless, low power and long distance to fit the required use case flexibility and lifecycle. Cost and longevity further drive the CPU choice to make sure that the smart object can survive during its whole lifecycle. For cost reasons many smart objects also lack a tamper resistant unit that can protect (long-term) secrets. Energy consumption and price limit the amount of RAM. Even though it increases over time long term storage tends to be limited for cost reasons.

A typical IoT example: *Consider a smart home with a few dozens of wireless devices realizing a functionality such as energy or lighting control. Such an application limits material costs since otherwise it would not be commercially viable. Thus, the devices use a low power and low data rate radio with an AES128 hardware coprocessor. The devices have a resource constrained CPU (e.g, a few tens of Megahertz, 16 bit) and limited memory (10 KB of RAM and 256 KB of Flash). The smart devices form a wireless mesh network in which some devices (gateways) access the Internet.*

We argue that the main ***crypto/security challenges today*** *for such a scenario are twofold, namely: (i) the efficient security configuration of devices during their introduction in the network, since this involves the usage of some kind of network access protocol enabling authentication and authorization and later key transport; (ii) the efficient management of the keys and credentials of the devices within the network so that devices can interact with each other securely and according to their respective access rights. For this scenario, message confidentiality and integrity can be achieved efficiently by the available AES128 hardware coprocessor, that is able to run even in energy harvesting devices such as light switches.*

Once a quantum computer is available, the ***future crypto/security challenges*** *will increase. First, the security configuration of devices during their introduction in the network will become more challenging since most quantum-resistant primitives for key exchange and authentication have longer keys. Second, managing the keys and credentials will remain challenging for the same reasons. And thirdly, AES128 will only provide 64 bits of security due to Grover's algorithm [Grover] so that many devices in the field be using too short keys.*

3 Discussion and specific proposals

For many applications, the main constraints in the design and development of a secure IoT system are imposed by the use case and the application lifecycle.
The "low" performance of cryptographic primitives for encryption and integrity is determined by the budget constraints (that determine hardware components). While a budget increase can give access to more powerful CPUs, faster radio,... so that encryption and integrity checks can be realized even in a more efficient manner, use case features such as the interaction model – between smart things and people – or scalability are the ones really limiting the solution deployability. Thus, the use case (lifecycle) also impacts the choice of cryptographic primitives and architectural design related to key and credential management.

For many (existing) applications, AES128 is and might still be a working solution performance-wise for the years to come. Still, it is fully true that having more lightweight alternatives than existing block ciphers (AES) or hash functions (SHA3) will definitively increase the lifetime of devices and performance. The main issue might be that in the advent of quantum computers we will require support of 256-bits keys. Given the large amount of deployed devices, most of them implementing an AES128 hardware coprocessor, we ask ourselves how to address this problem in a pragmatic way. Deploying a software update with AES256 (or any future alternative) might not always be feasible since device memories will be full and a software-only implementation will be likely too slow, or at least, slower than a solution based on a hardware co-processor. Therefore, next to research into new very lightweight solutions, it might be interesting to consider other known constructions that could reuse existing AES128 HW co-processor in deployed devices to achieve a longer effective key size.
Most importantly, we believe that the efficient management of keys/credentials of devices is and will remain the key challenge in IoT scenarios due to both performance issues and the constraints of use case and application lifecycle. This situation will only become more dramatic if quantum-resistant alternatives need to be supported since most existing quantum-resistant candidates have very long keys that will limit their applicability to IoT scenarios. We believe that the easiest solution for managing keys and credentials is to "simply" allow any pair of devices to directly share a common key and directly verify their credentials. Extensive research into key pre-distribution schemes such as polynomial schemes [Blundo] was carried out in the last decades, enabling this functionality and even providing information-theoretic security against collusion attacks up to a given threshold of colluding (or corrupted) nodes. New schemes, such as HIMMO, mimicking the operational features of Blundo's scheme have been designed with the advantage that they can achieve collusion resistance and quantum resistance if they are configured with large enough parameters. For smaller parameters, HIMMO inherits the information theoretic threshold that ensures that an attacker cannot learn about non-compromised keys since he lacks information. At the same time, HIMMO's design principles can be reused to efficiently realize other functionality such as a pseudo-random number generator from which other crypto primitives, e.g., encryption function can be derived.