SmartEdge Platform – Security

Gordon Walsh, Senior Security Architect, Industrial ASIC BU

Contents

Introduction **1**
Security Architecture 4
Security Implementation 5
Security Building Blocks 5
Security Capabilities 6
Integration Considerations **10**
Physical Security 10
Conclusion **10**
References **10**
Revision History **11**

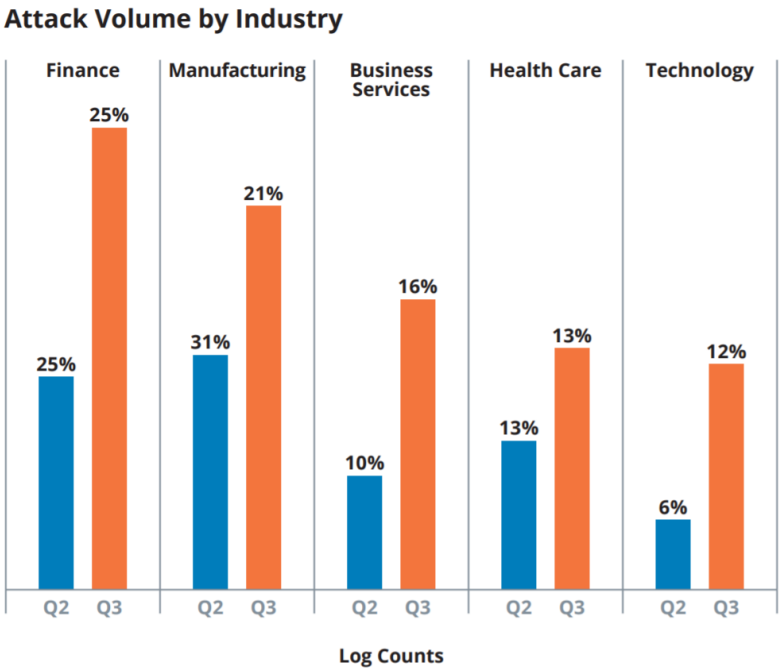
Introduction

At Arm TechCon 2016 Arm and SoftBank Chairman Masayoshi Son announced his vision for a trillion connected devices by 2035. If we truly are looking at 1 trillion connected devices by 2035 security is no longer optional. smart edge devices in the Industrial Internet of Things (IIoT) are increasingly looking to form a significant portion of those connected devices.

The marriage of operational technology (OT) and information technology (IT), the very force driving Industry 4.0, unfortunately brings new security risks across all parts of the value chain from device to cloud.

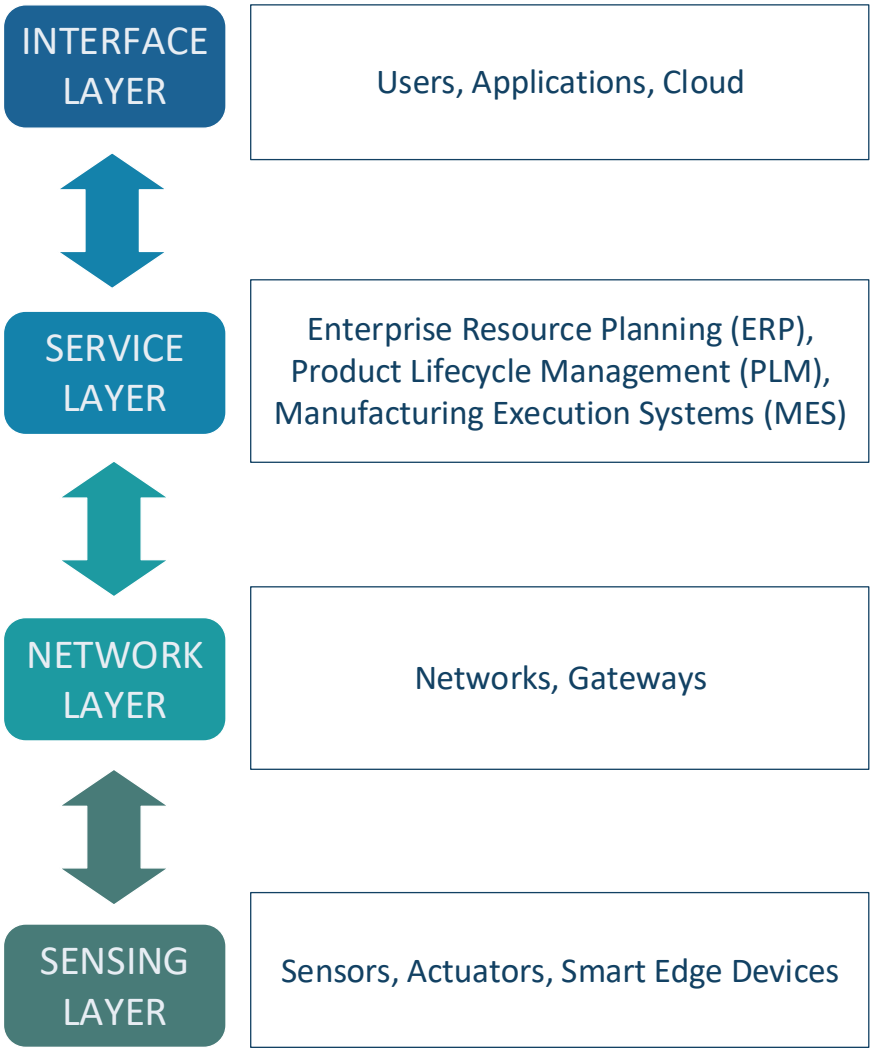
According to research by the NTT Security Global Threat Intelligence Center (GTIC) Manufacturers continue to be a key target for cybercriminals, the GTIC have observed challenges in securing manufacturing's Industry 4.0 and smart factories [1]. There was a notable amount of botnet traffic against manufacturing devices in Q3 of 2017, activity of this type may suggest that manufacturing devices – perhaps internet of things (IoT) and operational technology (OT) devices, specifically – remain unsecured [2].

GTIC analysis of NTT Security monitoring data indicates the top five industries targeted were finance, manufacturing, business services, health care and technology. The graph below represents the top targeted industries based on attack volume in comparison to Q2'17. The column heights represent the number of attacks and the percentages represent the overall attack volume per industry in that quarter [2].



All of this points to the need to address security of smart edge devices.

The diagram below shows an example representation of an IIoT Architecture. Security must be addressed at each layer as well as at the interface between each layer. Providing security capabilities within the smart edge devices is the first step in protecting the value chain and provides a foundation upon which to build the remainder of the security solution.

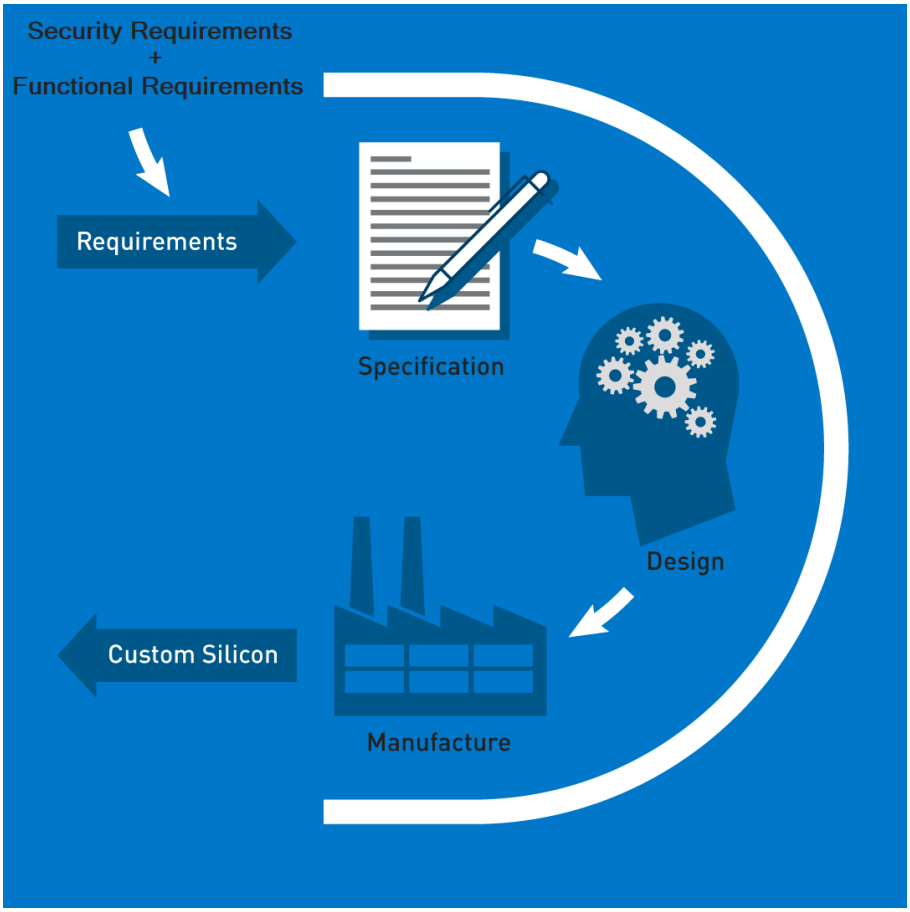

INTERFACE LAYER
Users, Applications, Cloud
SERVICE LAYER
Enterprise Resource Planning (ERP), Product Lifecycle Management (PLM), Manufacturing Execution Systems (MES)
NETWORK LAYER
Networks, Gateways
SENSING LAYER
Sensors, Actuators, Smart Edge Devices

Security Architecture

Security Architecture definition relies on having a clearly defined set of Security Objectives. These are expected to be provided by a Client's Security Expert/Consultant and are typically the result of Threat Modelling and Security Analysis across the entire value chain from device to cloud (see the example IIoT Architecture above).

Renesas, formerly Dialog's ASIC & IP division will work with a Client to assess the Security Objectives to determine which are applicable to the silicon and will agree a set of Security Requirements for the custom silicon.

The Security Requirements will feed into the specification of the custom silicon along with the Functional Requirements.



Security Implementation

Security Building Blocks

Once the Security Requirements are understood the process of selecting the appropriate Security Building Blocks can begin. Adesto's ASIC & IP division partner with expert Security IP vendors (such as Synopsys and ARM) to choose the appropriate security certified hardware building blocks.

Some examples of such hardware Security Building Blocks are:

- Hardware Root of Trust
- Processor with Trusted Execution Mode
- Dedicated Security Processor
- Cryptographic Accelerator
- Public Key Accelerator
- Processor Bus Infrastructure with Access Control
- Memory Controller with Access Control
- Debug Controller with Access Control
- True Random Number Generator
- Physically Unclonable Function

The Hardware Root of Trust ensures the device starts from reset in an expected state and its firmware is intact and has not been tampered with. The Hardware Root of Trust is foundation up which Secure Boot is based and typically triggers verification of the various processor boot stages (e.g. Bootloader, OS, Application …).

Encryption/Decryption Cryptographic Algorithms consist of symmetric (same key for encryption/decryption) and asymmetric algorithms (secret Private Key for encryption and Public Key for decryption). There are advantages and disadvantages of each as outlined in the table below.

Type	Advantages	Disadvantages	Usage Example
Symmetric	Faster	Single Key system is vulnerable to key exposure during key distribution	Using a short live session key to encrypt firmware update payload (data + signature) during transport using a AES 256 hardware accelerator
	Lower Hardware Area	Not suitable for signature generation or validation	
Asymmetric	Private/Public Key system supports secure key distribution since Private Key is kept secret	Slower	Creating a signature for a firmware update by encrypting a 256bit hash of the firmware update data using a software Elliptic Curve Cryptography (ECC) algorithm
	Suitable for signature generation and validation	Higher Hardware Area	

Careful consideration must be made to choose the appropriate cryptographic algorithm based on the above characteristics.

Note that a subset of these hardware building blocks are often integrated, along with other components, into a dedicated Security Subsystem within the device. For example, when coupled with an appropriate software architecture these Security Subsystems can be used to create a Trusted Execution Environment [3]. These Security Subsystems are also known as Complex Cryptography Engines/Accelerators and it is becoming popular for Security IP vendors to provide these integrated solutions in their IP portfolios.

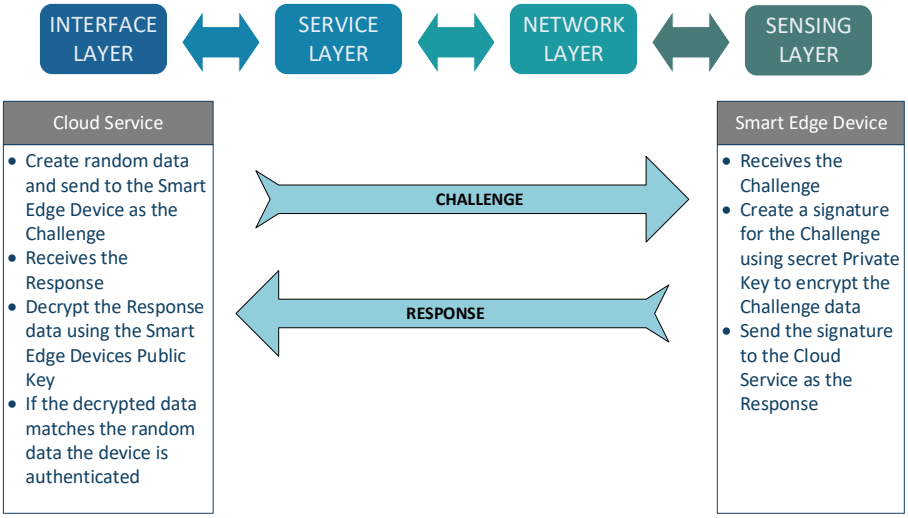
Security Capabilities

The integration of the above building blocks into the silicon will enable the desired Security Capabilities called out in the Security Objectives. Some examples of such Security Capabilities are:

- Secure Boot
- Device Provisioning
- Device Authentication
- Code Protection
- Data Protection
- Secure Communication
- Secure Update
- Secure Debug

It is typical that a Security Capability is reliant on elements outside of the silicon to meet the Security Objectives. The following examples show how elements in the various layers shown above work together to deliver a Security Capability.

Device Authentication



The following Security Building Blocks will typically be used on the Adesto SmartEdge™ device to deliver the Device Authentication capability:

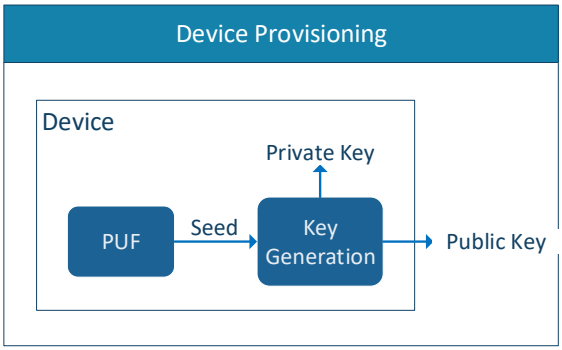
- Processor in Trusted Execution Mode or Dedicated Security Processor – used to process the Device Authentication Challenge
- Memory Controller with Access Control – used to unsure that the received Challenge data is protected from modification during and after reception, also used to ensure that the Processor in Trusted Execution Mode / Dedicated Security Processor is the only entity allowed to access and send the Response data (signature)
- Public Key Accelerator – used to accelerate the generation of the signature

Note: The generation and distribution of the Public Key and secret Private Key pair for the SmartEdge™ is typically done as part of Device Provisioning. Due to security concerns around the generation, distribution and storage of the secret Private Key the use of a Physically Unclonable Function (PUF) for key generation is becoming popular. This is described in the following section.

PUF Key Generation

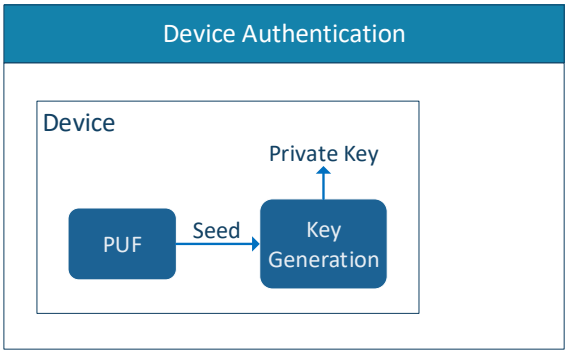
A Physically Unclonable Function (PUF) is a secure silicon fingerprint that cannot be modified or duplicated and is guaranteed to be unique from one device to another.

Thanks to the nature of a PUF it is possible to use the data from a PUF to generate a unique Public Key and secret Private Key pair for individual Smart Edge Devices that can then be used for Device Authentication.



During Device Provisioning, the following occurs:

- The PUF response is used as seed into a Public/Private Key Generation algorithm in the silicon
- The generated Public Key is output to the provisioning agent and provided to any agent that needs to authenticate a device
- The generated Private Key is not output to the provisioning agent and is not stored permanently in the device, when required the Key Generation is re-seeded and the Private Key is only stored temporarily and is erased when no longer required.

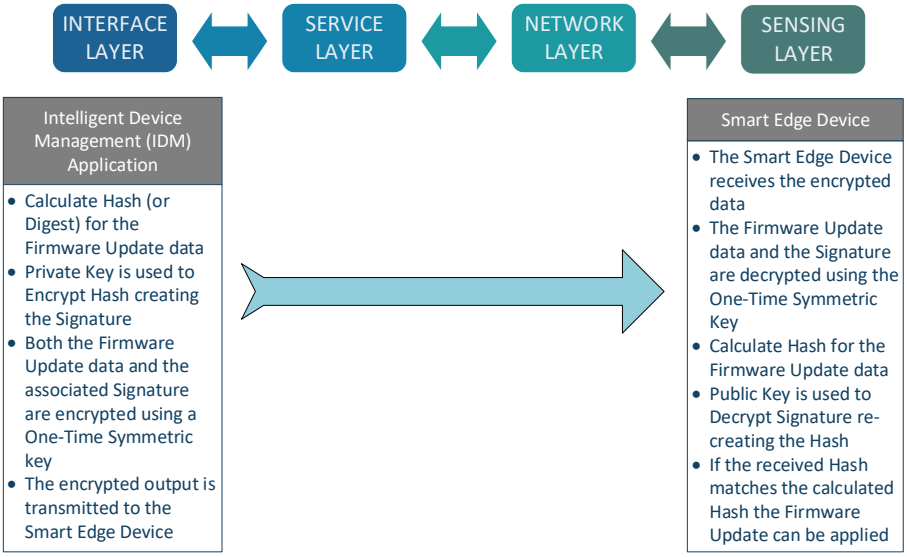


During Device Authentication, the following occurs:

- The PUF response is used to re-seed the Public/Private Key Generation algorithm in the silicon
- The generated Private Key is used to create the Response data (signature) for the received Challenge and the subsequently Private Key erased from temporary storage.
- The authenticating agent then verifies the Response using the Public Key that it was provided during Device Provisioning

Note: PUFs can be used to support other Security Capabilities such as Data Protection.

Secure Update



The following Security Building Blocks will typically be used on the SmartEdge™ to deliver the Secure Update capability:

- Processor in Trusted Execution Mode or Dedicated Security Processor – used to validate and apply the Firmware Update
- Memory Controller with Access Control – used to unsure that the received Firmware Update data is protected from modification during and after reception, also used to ensure that the Processor in Trusted Execution Mode / Dedicated Security Processor is the only entity allowed to apply the Firmware Update to the existing software
- Cryptographic Accelerator – used to accelerate the generation of the Hash and the Symmetric decryption
- Public Key Accelerator – used to accelerate the validation of the signature

Integration Considerations

Integration of any building block must also consider functional constraints such as area, performance and power. For example, real-time performance requirements will likely drive the use of Cryptographic Accelerators but area constraints may dictate the use of a Processor with Trusted Execution Mode rather than using a Dedicated Security Processor.

Physical Security

There are often security concerns relating to physical attacks on the device. There are capabilities that can be provided in the device to defend against certain classes of physical attacks, for example:

- **Power Supply Voltage Attacks**: generally used to trigger bits within the device to toggle which could force a secret encryption key to a known value or force a device to accept a firmware update even when the signature check has failed – e.g. voltage monitoring within the device is one way to protect against this type of attack.
- **Pin/Interface Attack**: generally used to gain unrestricted access the device where pins are exposed to the outside world – e.g. Secure Debug can be used to prevent unauthorized access to the JTAG pins.
- **Chassis Intrusion Attack**: generally used to gain unrestricted access the device by opening the casing of the product – e.g. a pin can be used to trigger an event into the device when the casing is opened which would automatically trigger erasure of secure collateral such as secret keys and place the device in secure parked state.

Conclusion

In conclusion, Renesas have the knowledge and working partnerships necessary to ensure your custom silicon delivers on the agreed Security Requirements enabling an IIoT Architecture that meets your Security Objectives.

References

[1] GTIC 2017 Q2 Threat Intelligence Report:
http://it.nttdata.com/fileadmin/web_data/country/it/NTT_Security_Q2_2017_FINAL.pdf

[2] GTIC 2017 Q3 Threat Intelligence Report: https://www.nttsecurity.com/docs/librariesprovider3/default-document-library/ntt-security-GTIC-2017-q3-threat-intelligence-report.pdf

[3] Trusted Execution Environment: https://en.wikipedia.org/wiki/Trusted_execution_environment

Revision History

Revision	Date	Description
1.0	Mar 01, 2018	Initial release
1.1	Dec 22, 2021	Re-brand