What is Embedded Security?

Digital Security Solutions Marketing
Summer 2019

Agenda

1. Introduction
2. Demystifying Security
   - Crypto Overview
   - PKI
3. Security Use Cases
4. Infineon Hardware Security Overview
5. Q&A

Agenda

1. Introduction
2. Demystifying Security
   - Crypto Overview
   - PKI
3. Security Use Cases
4. Infineon Hardware Security Overview
5. Q&A

Infineon Digital Security Solutions

#1 for Embedded Security*

Authentication

Brand Protection, Anti-Counterfeiting, & Ecosystem Control

Trusted Platform Modules

Boot Protection & Key Management

Internet of Things Security

Network & Device Integrity Protection

Security Leader in Additional Markets

Smartcard

SIM

Mobile Payment

Government ID

*Source: IHS, Embedded Digital Security Report 2016, January 2016

Digital Security Solutions Overview

What we offer

Card-based and embedded security solutions

Security Solutions for **Connected Car**

Extensive **packaging and service** portfolio

Broad portfolio of **innovative** solutions for **Connected Device Security**

Core competencies

Tailored security for **best cost-performance ratio**

Contactless excellence

Embedded control

Software and security application insights

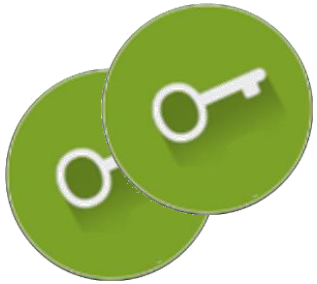
Agenda

1. Introduction
2. Demystifying Security
   - Crypto Overview
   - PKI
3. Security Use Cases
4. Infineon Hardware Security Overview
5. Q&A

Cryptography Basics

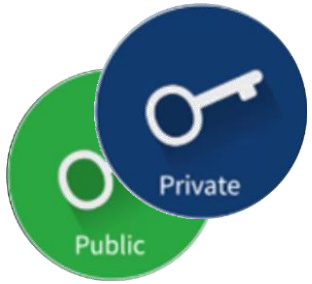
- All cryptography is based upon encoding/decoding or otherwise manipulating some bit of data using a predefined secret (key/key pair) and shared algorithm
- Good security must provide **confidentiality**, **integrity**, and **availability**
- Good security uses known, proven techniques with the keys kept secret
  - Security through obscurity does not work

Symmetric



DES **AES**

Asymmetric



RSA **ECC**

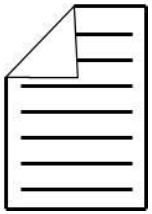
Hash Algorithms

SHA

Types of Cryptography Summary

	Hash	Symmetric	Asymetric	
	SHA	**AES**	**RSA**	**ECC**
Benefits	Fast Small result	Fast	Security	Security Smaller keys
Typ. Key	256b	128b / 256b	2kb	256b
Uses	Data Validation	Fast Encryption	Establish Trust Key Management	
	 Platform Integrity  Secure Update	 Stored Data Protection  Encrypted Communication	 Authentication  Key Management  Secure Update	

Symmetric Cryptography: DES & AES



Standards

DES
Developed: 1977
Block Size: 64
Key Size: 56 bits
Considered insecure for most applications today

AES
Developed: 2002
Block Size: 128
Key Size: 128 – 256 bits
Default standard for most IoT symmetric encryption

Benefits

Fast cryptography

Easily implemented

Small key lengths

Disadvantages

Key security

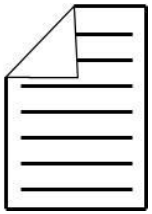
Key exchange problem

Best Uses

Stored data encryption

Encrypted communication

Asymmetric Cryptography: RSA & ECC



Standards

RSA
Developed: 1978
Key Size: 1kb – 3kb
Algorithm based on large prime numbers

ECC
Developed: 1985
Key Size: 256 – 512 bits
Algorithm based on points of an elliptic curve

Benefits

Key Security

Key Exchange

Disadvantages

Slow cryptography

Large keys

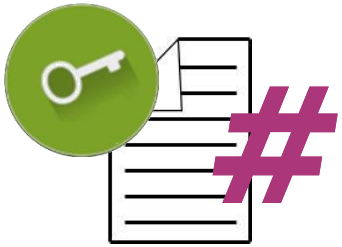
Best Uses

Authentication

Secure Key Exchange

Secure Updates

Hash Algorithms



Standards

Hash Algorithm
Publicly known algorithm
Input: Data block
Output: HASH value (256–512b)
Use: Integrity check

Benefits

Very fast

Memory efficient

Disadvantages

One-way only

Algorithm-based (no key)

Best Uses

Secure boot

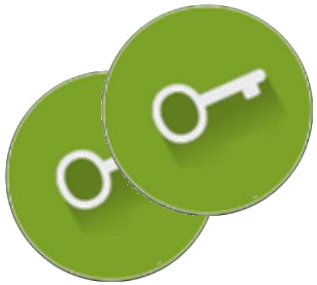
Platform integrity

Sign/verify (secure updates)

From Building Blocks to Tools

- Basic cryptographic techniques have different strengths
- Used together to create Security Tools

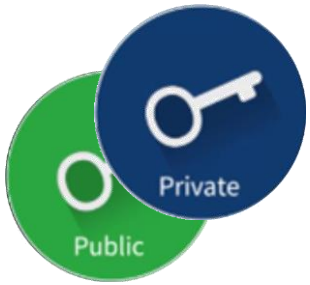
Symmetric



DES **AES**

- Fast Encryption
- Easily Implemented

Asymmetric



RSA **ECC**

- High Security
- Flexible Key Handling

Hash Algorithms

SHA

- Space Efficient
- One-way Digest

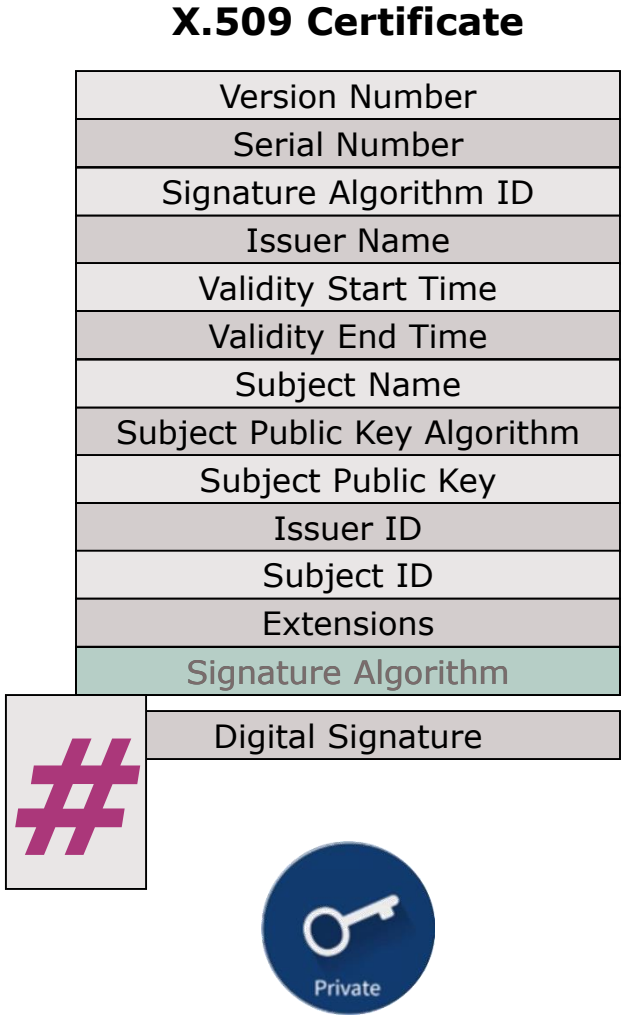
Certificates & Signatures

PKI

Key Exchange

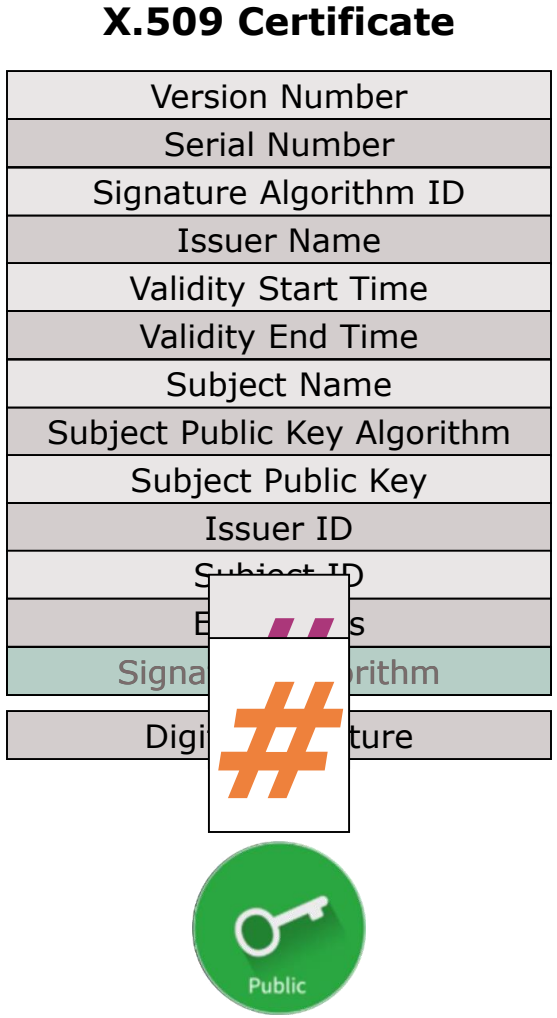
Certificates & Digital Signatures

- Certificates contain information about the device or service and can include new public keys
- Certificate hashed using referenced algorithm
- Digital signature created by signing hash value with sender's private key
- Digital signature confirms:
  - Certificate send from correct source
  - Certificate contents have not changed



Certificate Receipt & Signature Verification

- Read certificate
- Create hash value using listed algorithm
- Create verification hash from signature using sender's public key
- Compare hash values
  - Certificate send from correct source
  - Certificate contents have not changed



One-Way Authentication
OPTIGA™ Trust B

Customer Specific Key Pair

Chip Unique Key Pair

Public

Public

Private

Private

Certificate

Signature

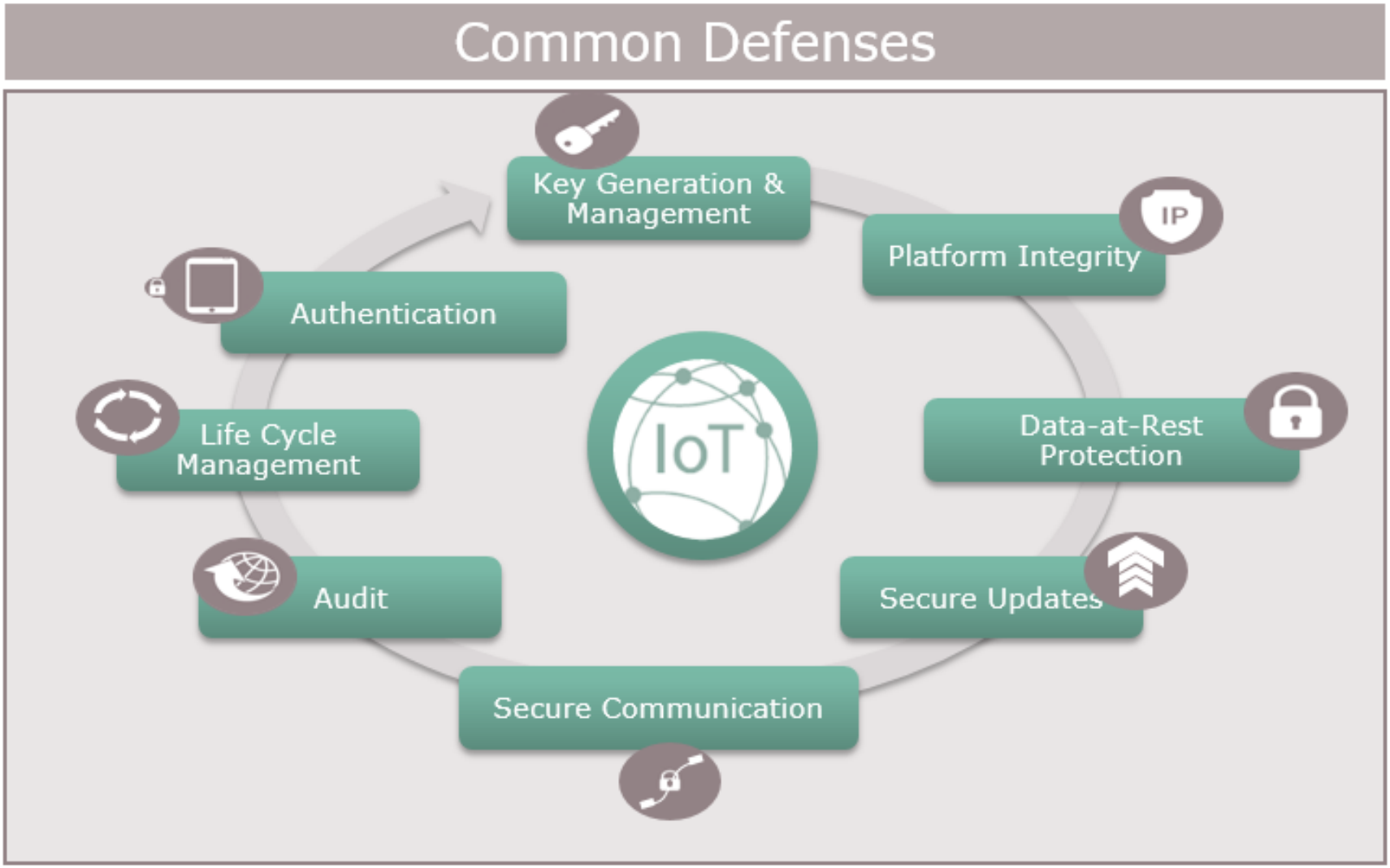
Customer
Host Device

One-Way Authentication
OPTIGA™ Trust B

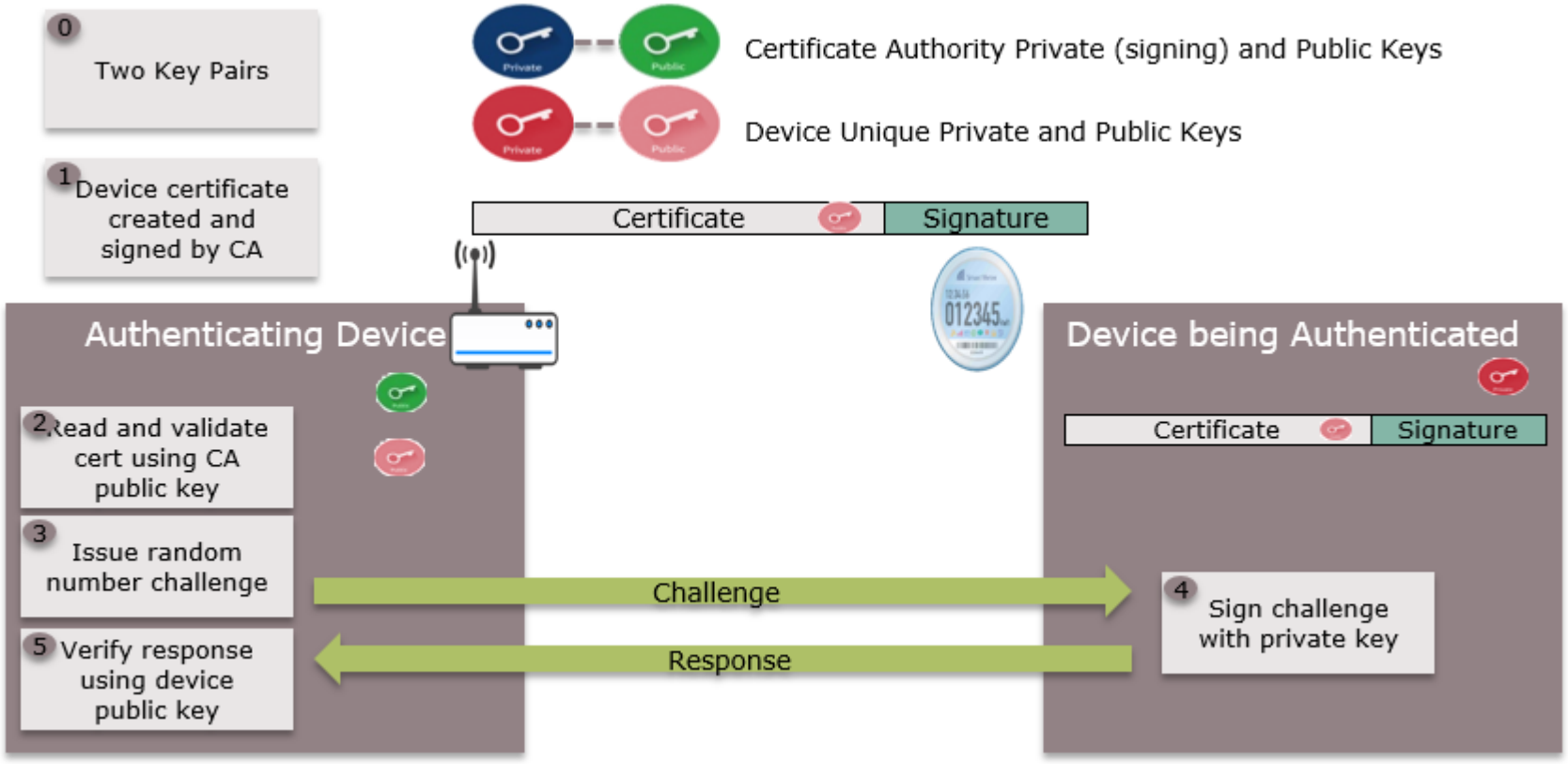
Agenda

1. Introduction
2. Demystifying Security
   - Crypto Overview
   - PKI
3. Security Use Cases
4. Infineon Hardware Security Overview
5. Q&A

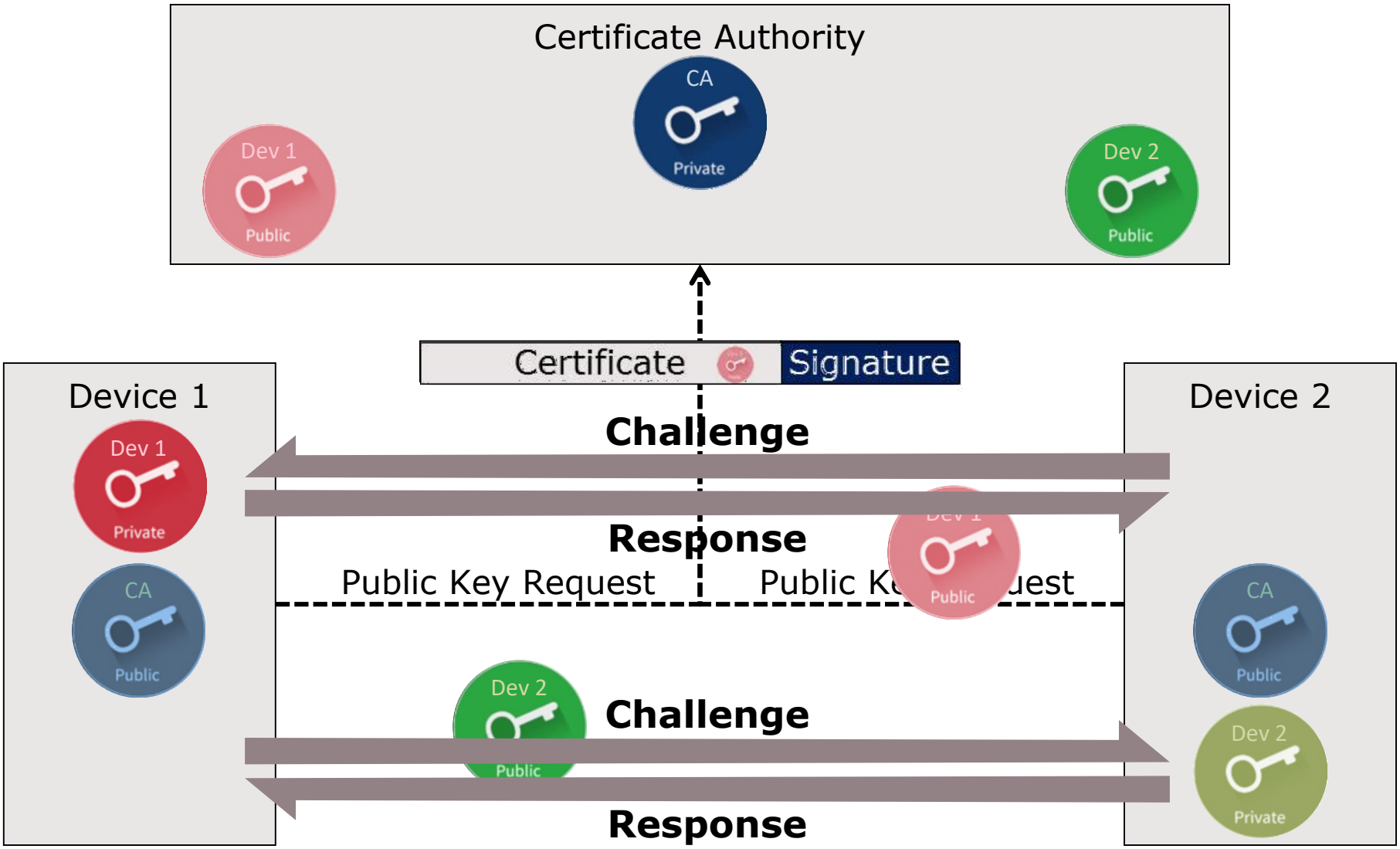
Standard Security Use Cases



Strong authentication

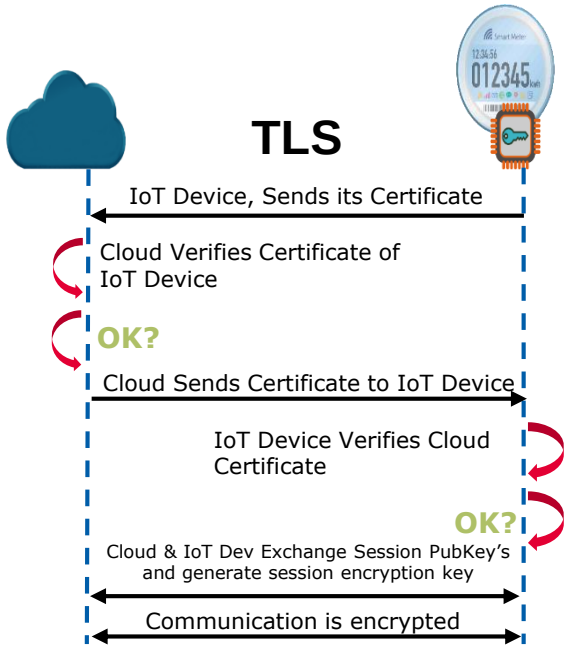


Mutual Authentication



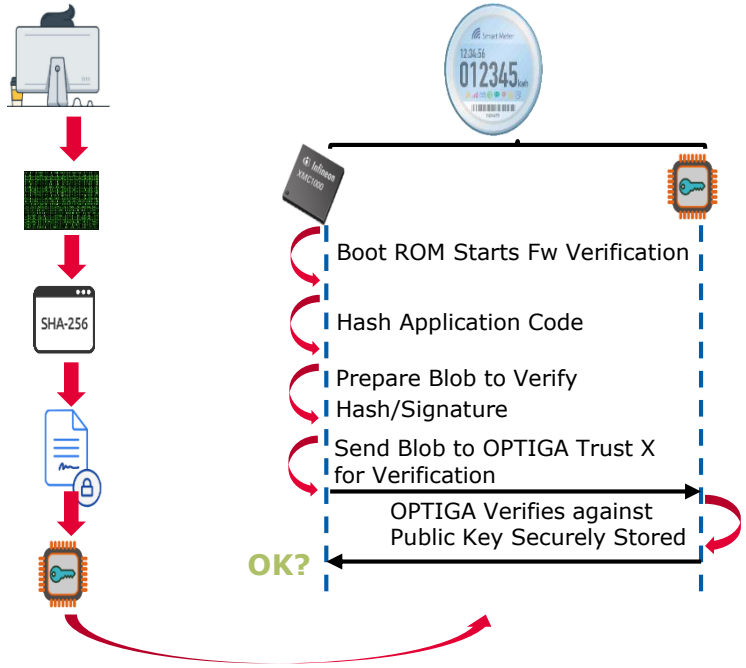
Secure Communication

- Secure channel/session established between 2 end points
- Ensures the nodes themselves are authenticated
- Authentication can occur in both directions – mutual
- Authentication, encryption/decryption occurs at the end points



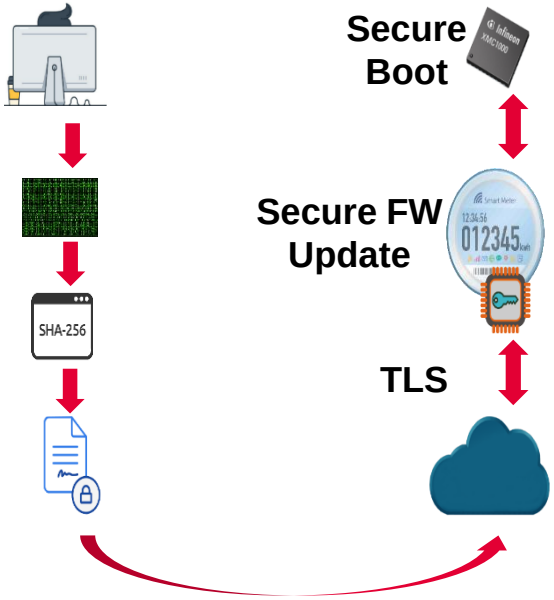
Boot Protection

- Prevent unauthorized builds
- Ensure firmware has not been altered by unauthorized party
- Verify system integrity



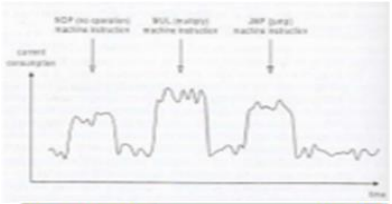
Secure Updates

- Secure Updates is a combination of:
  - Secure Communication
  - Secure Boot
- Ensures that downloaded software updates really came from OEM
- Ensures that downloaded software has not been modified by ”Men In The Middle”

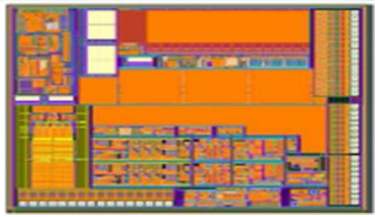


Why Hardware Security? Summary

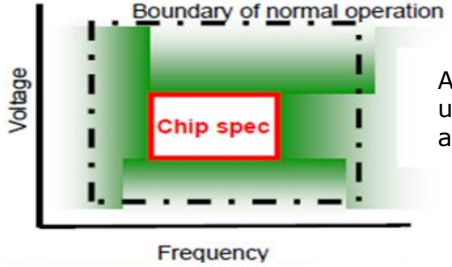
Standard Micro



Attacker
can read data by
monitoring
current consumption

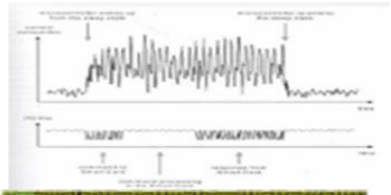


Attacker can capture
data by probing
metal patterns

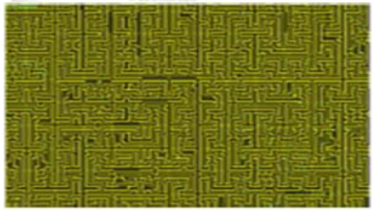


Attacker can read data
under
abnormal conditions

Secure Element

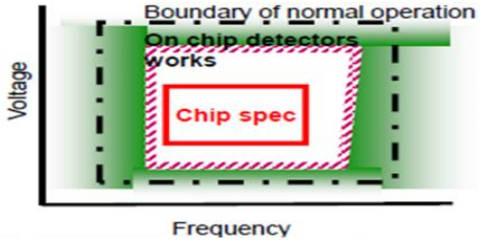


Current consumption is scrambled by **dynamically generated noise** so that Data cannot be extracted by current monitoring.



Chip is protected with:

- **"Active" metal shield** to prevent data capture
- **Randomized layout**

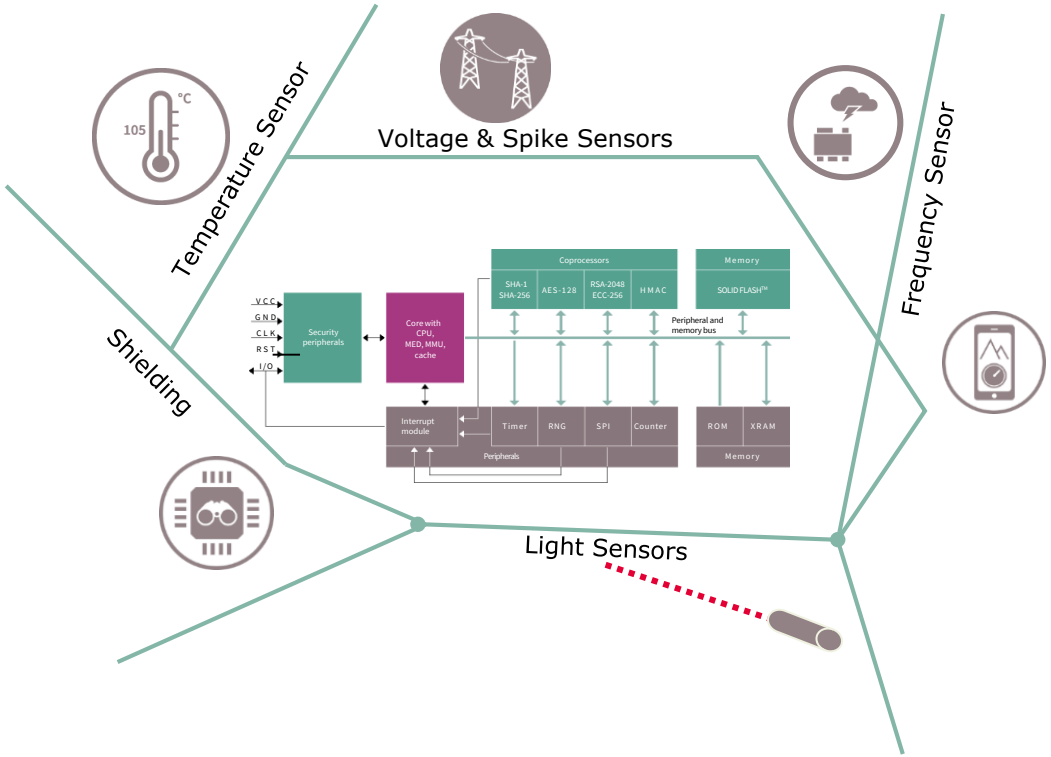


On **chip sensors** force
to stop Operation under
Abnormal conditions

Agenda

1. Introduction
2. Demystifying Security
   - Crypto Overview
   - PKI
3. Security Use Cases
4. Infineon Hardware Security Overview
5. Q&A

Underlying technologies:
Tamper / side channel attack resistance



Tamper-resistant Security

- Security by design: Shielding, Firewalling, encrypted buses
- Rich set of security sensors

- Memory encryption
- True random number generator

- Securely coded soft/firmware

- Developed, produced, programmed and personalized in secured environment

OPTIGA™ Family

	OPTIGA™ Trust B	OPTIGA™ Trust E	OPTIGA™ Trust X	OPTIGA™ Trust P	OPTIGA™ TPM
					
Security Level	Basic	CC EAL 6+ *	CC EAL 6+ *	CC EAL 5+ *	CC EAL 4mod
Functionality	Authentication	Authentication	Connected device security	Programmable	TCG standard
NVM (Data)	64Byte	3kByte	10kByte	150kByte **	6kByte
Cryptography Private key stored in secure HW	ECC131	ECC256	ECC384	ECC521 RSA2K	ECC256 RSA2K
Type of Host System	MCU without OS / proprietary OS / RTOS		Embedded Linux		Windows / Linux
Interface	SWI	I2C	I2C	UART	I2C, SPI, LPC
System integration	✓	✓	✓	✓	Platform vendor

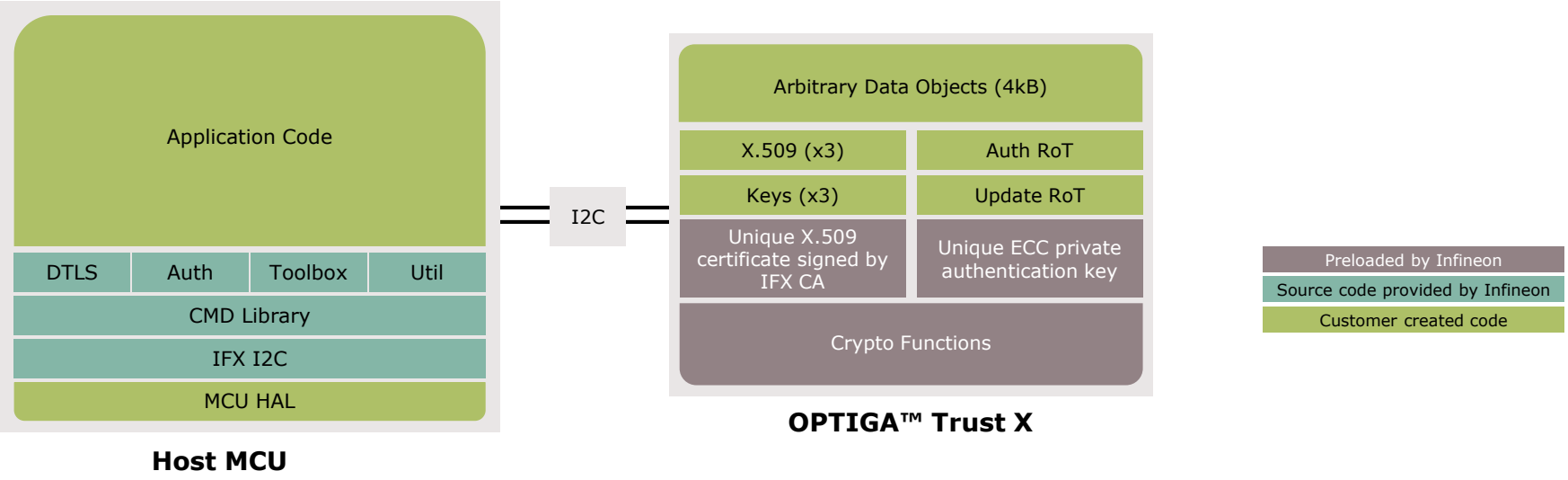
✓ Done by IFX
* Based on certified HW
** Code & Data

✓ Customer Implementation, support by IFX

Security and Complexity

OPTIGA™ Trust X Implementation

- Full turnkey solution with pre-loaded keys and certificates
- Host interface and crypto libraries provided
- Easy integration into device application code



OPTIGA™ Trust on GitHub

Hostcode repository

- GitHub is a world's leading software development platform for Open Source projects
- github.com/Infineon - Official Infineon GitHub account
- github.com/Infineon/optiga-trust-x
  OPTIGA™ Trust X **Application Notes Framework**
  - Wiki Page describing the structure, performance, code footprint and porting guide
  - **Not** a new release of hostcode software. The difference:
    - Folder and file structure mimics upcoming OPTIGA™ Trust M
    - User API mimics upcoming OPTIGA™ Trust M
    - Hostcode is under MIT Open Source license
- Application notes are based on this framework via *git submodule* (cross reference)

Agenda

1. Introduction
2. Demystifying Security
   - Crypto Overview
   - PKI
3. Security Use Cases
4. Infineon Hardware Security Overview
5. Q&A

Support Links

Collaterals and Brochures 	– Product Info Page & Product Briefs. – Video Links	www.Infineon.com/optiga https://www.infineon.com/cms/en/product/security-smart-card-solutions/optiga-embedded-security-solutions/#!videos
Technical Material 	– Datasheets – Demo Boards	https://www.infineon.com/cms/en/product/security-smart-card-solutions/optiga-embedded-security-solutions/optiga-trust/ **TPM Data Sheets are located in My ICP** TrustX www.Infineon.com/optiga#!boards
Contact 	– Infineon Product Marketing	East: Patrick.Carrier@Infineon.com West: Terry.Kreifels@Infineon.com

Part of your life. Part of tomorrow.