FTF | FREESCALE TECHNOLOGY FORUM 2015

End-to-End Security Solutions in IoT

FTF-SNT-F1252

Jim Bridgwater | Product Line Manager

JUNE.2015

freescale™

External Use

Introduction

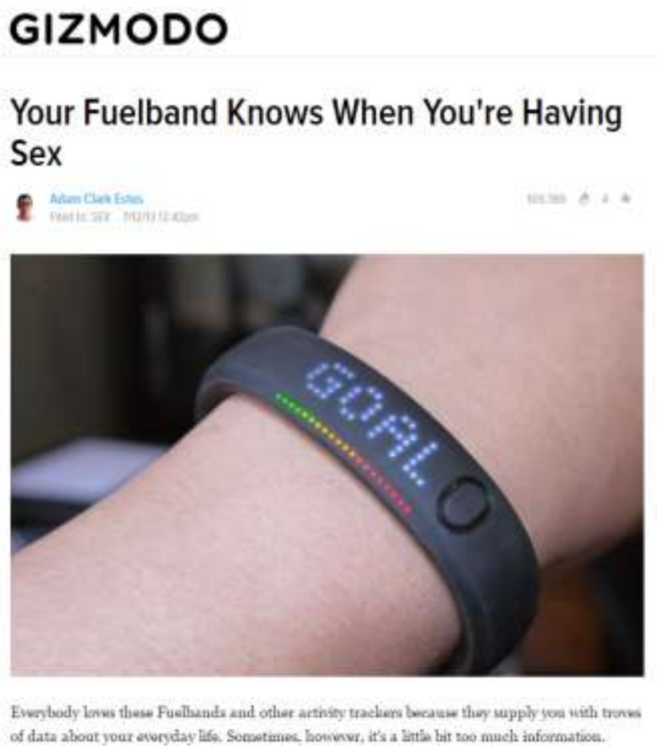
- This session will examine the trends and challenges associated with security for the so-called "Internet of Things" (IoT)
  - Presentation of security fundamentals and a layered model of the IoT
  - Discussion of potential solutions to IoT security challenges
  - Overview of how QorIQ Trust architecture can be leveraged for IoT security
- Presenter: Jim Bridgwater
  - Product Marketing Manager in Freescale's Digital Networking group

Agenda

- Introduction and Agenda Review
- Security Fundamentals
  - IoT Layers
  - IoT Security Challenges
- IoT System Examples
  - Smart Home
  - Industrial
  - Automotive
- Cloud Layer
- Device Layer
- Gateway Layer
- Secure Manufacturing Example
- Session Review and Wrap-up

Why Do We Need to Secure the Internet of Things?

- Financial
  - Eg. Tampering with a smart meter to reduce utility bills
- Safety
  - Malicious attacks on infrastructure or automobiles can endanger lives
- Operational
  - Zero downtime for industrial equipment
- Privacy
  - Loss or theft of personal information

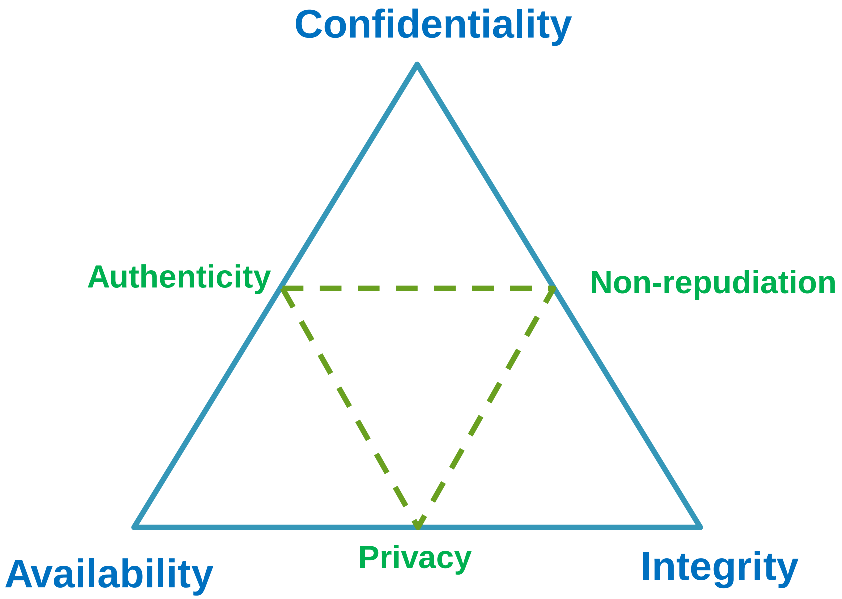

GIZMODO

Your Fuelband Knows When You're Having Sex

Adam Clark Estes
Filed to: SEX 7/12/13 12:40pm [illegible]

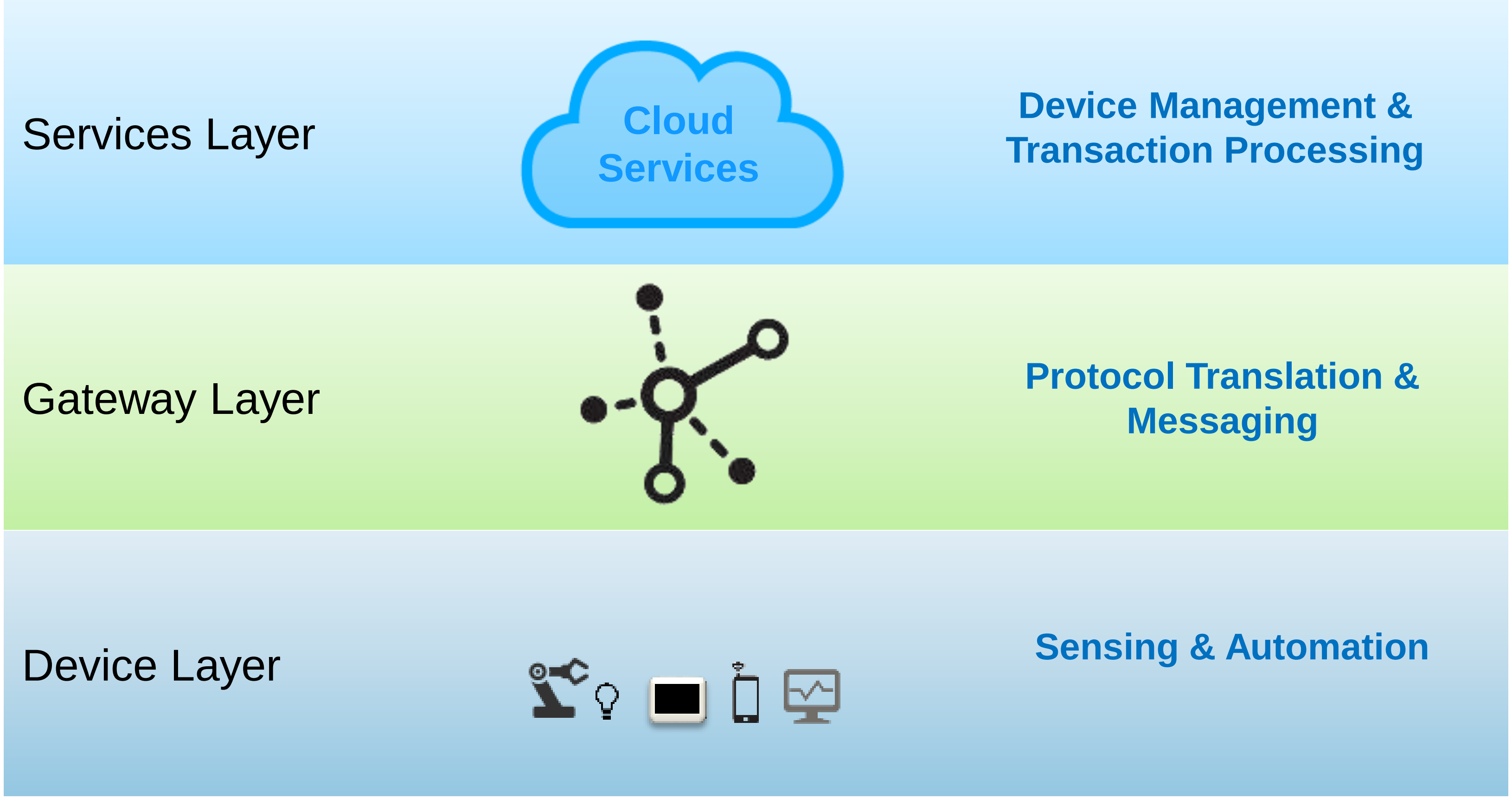
Everybody loves these Fuelbands and other activity trackers because they supply you with troves of data about your everyday life. Sometimes, however, it's a little bit too much information.

Security Fundamentals



- Confidentiality – data is only disclosed to authorized parties
- Integrity – data is trusted
- Availability – data is accessible when and where needed
- Non-repudiation – a trusted audit trail is provided
- Authenticity – node’s identity can be verified
- Privacy – service does not automatically see customer data

IoT Layer Examples



Services Layer

Cloud Services

Device Management & Transaction Processing

Gateway Layer

Protocol Translation & Messaging

Device Layer

Sensing & Automation

Security Questions for IoT Nodes

- Who are you?
- Can I trust you?
- Can you protect yourself against malware infection?
- Can you stay healthy?
- Can you detect malware infections?
- Can you recover from infections?
- Can you maintain secrets while infected?
- Can you protect yourself against hardware tampering?
- Can you protect the confidentiality of data from tampering?
- Can you protect integrity of data from tampering?
- Can you protect computation from tampering?
- Can you maintain the confidentiality, integrity, and availability of data at rest?
- Can you prepare a device for resale or decommissioning?
- Can you safely engage in cryptographic protocols?
- Can you support common models of provisioning?
- Can you securely maintain evidence?
- Can you be managed remotely?
- Can you secure legacy hardware?

What are the Challenges with IoT Security?

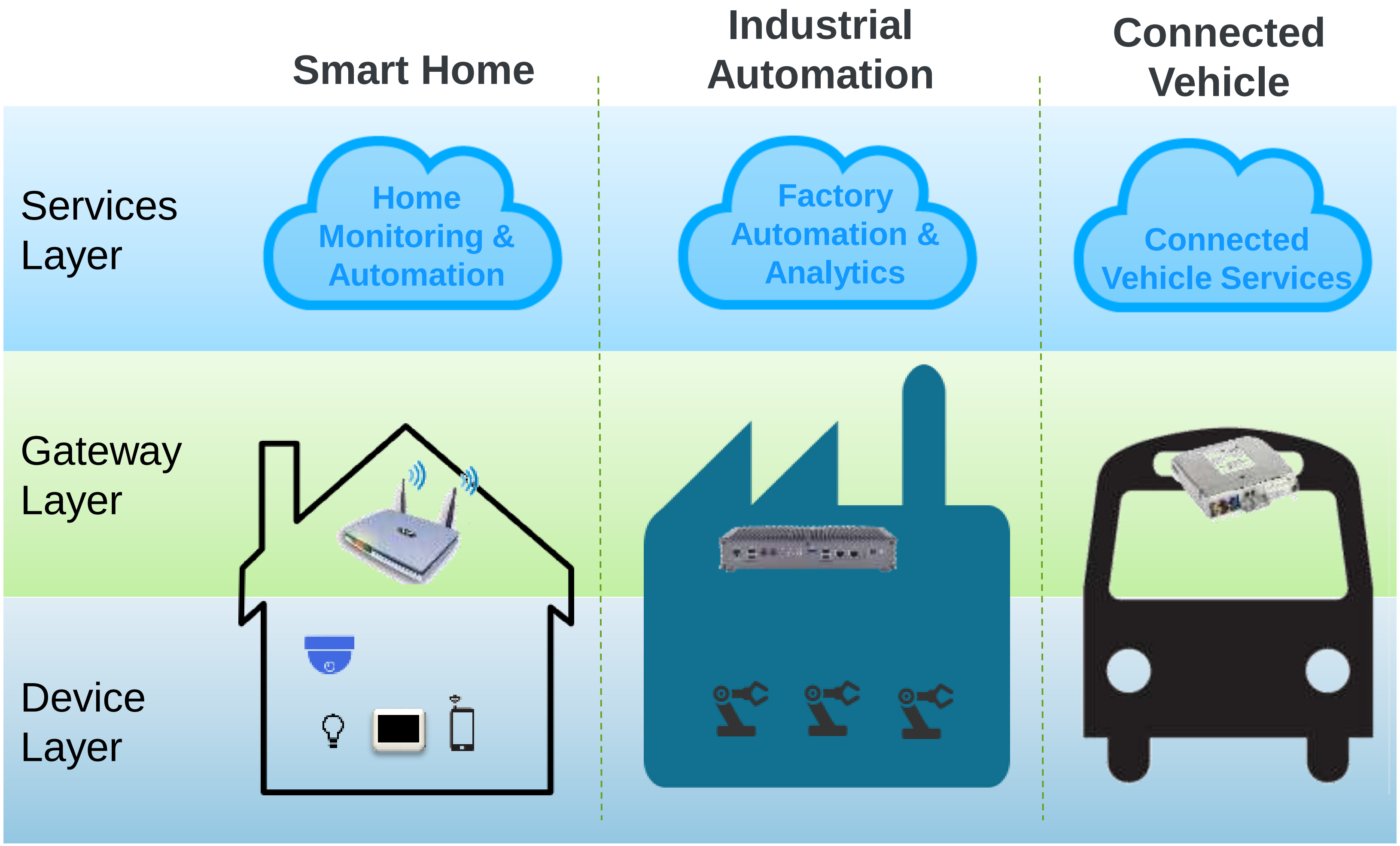
- Cloud / Service Layer
  - Sheer number of transactions to service 50B IoT devices
  - Separation of customer data in shared cloud hosting model
  - Attack prevention/detection/recovery

- Gateway Layer
  - Isolation of WAN from LAN, only permitted communication passes
  - Separation of customer applications from service provider SW
  - Remote lifecycle management of keys

- Device Layer
  - Low power, low cost security
  - Interoperability, ease of use
  - Many different protocols

IoT System Examples

IoT System Examples

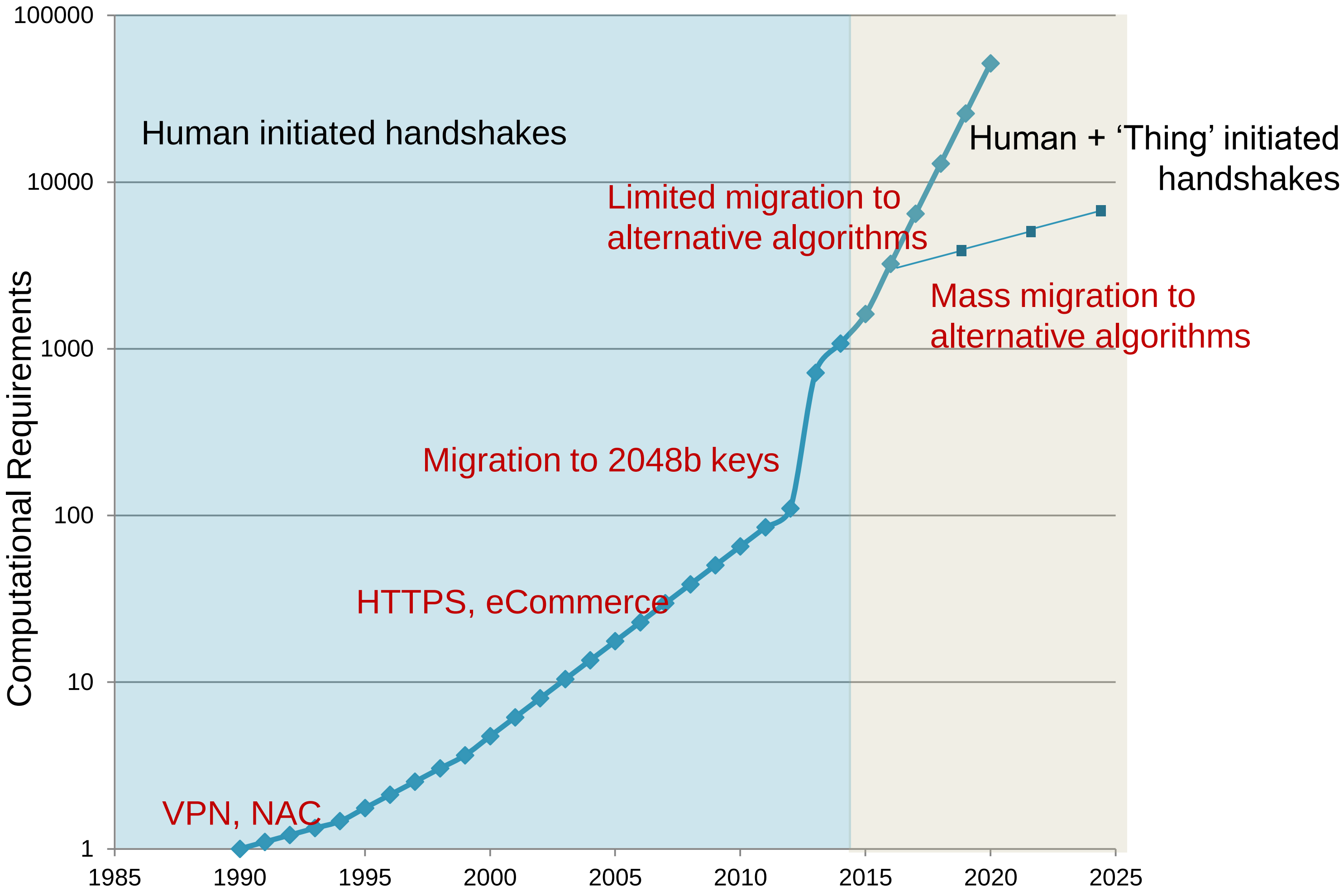


Cloud Layer

Transport Layer Security (TLS) Protocol: How Internet Transactions are Secured

- Uses X.509 certificates and asymmetric cryptography to authenticate the counterparty and negotiate a symmetric session key
- Session key is then used to encrypt data flowing between the parties
- Certificate authorities and a public key infrastructure are necessary to verify the relation between a certificate and its owner, as well as to generate, sign, and administer the validity of X.509 certificates
- Asymmetric cryptography uses “big number” maths – currently 2048-bit private keys are the norm – computationally intensive!
- Relies on private keys being kept private

Secure Handshaking



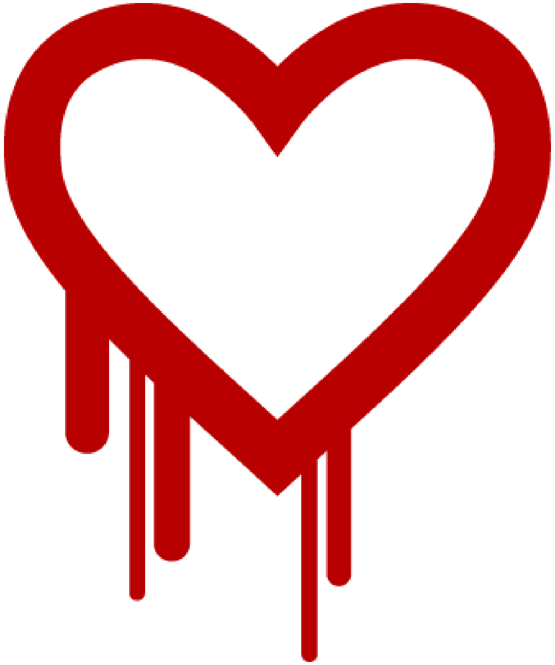
Shaking Hands with a Cloud

- Can a cloud of commodity hardware systems service all these handshakes?
  - Yes, at ~10 handshakes/sec/watt (2048b)
  - Yes, at high risk of keys being exposed
    - Criminal organizations are selling SSL keys as low as $1,000

- Hardware Security Modules (HSMs) can address these deficiencies
  - Provide accelerated cryptographic services within a hardened boundary
    - >200 handshakes/sec/watt (2048b)
  - Protect and manage provisioned keys; keys cleared if HSM tampered

- A new type of HSM proposes to be fast & power efficient, while also cheap enough for wide deployment

“Heartbleed” Bug



- Bug found in the “Heartbeat” extension in the OpenSSL cryptographic library, a widely used implementation of TLS
- 17% (around half a million) of the Internet's secure web servers were believed to be vulnerable to the attack
- Heartbleed enabled “memory scraping” of the target, i.e. downloading any data in the server’s memory, potentially including private keys and user’s passwords and cookies
- A fixed version of OpenSSL was released within 24hrs of the public announcement

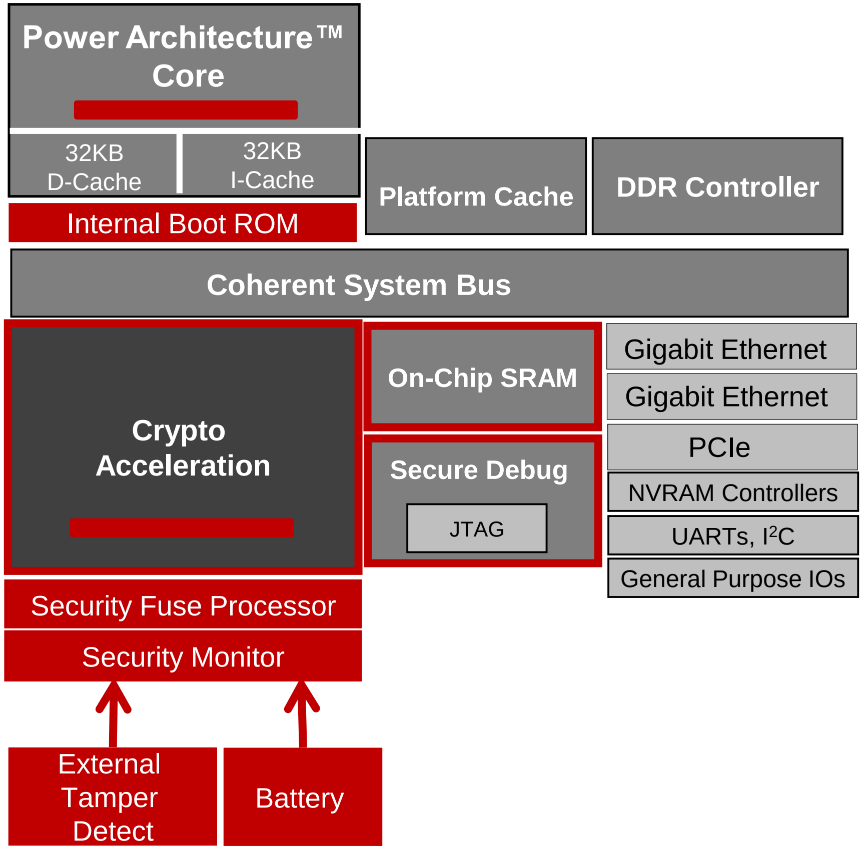
Freescale C29x Trust Architecture & Key Protection

Factory provisioned with:

- OTP and/or battery backed secret key
- Root of Trust for verification (secure boot public key)
- Options for additional credentials

In the field:

- Performs secure boot, continuously checks for HW & SW security violations
- Generates public & private keys
- Exports public keys
- Stores private keys as AES-256 protected key blobs
- Can use credentials to create secure tunnel with Key Management Server; download many additional SSL keys
- Server software can request sign & verify services with C29x HSM defending the keys even against physical attacks.



C29x can perform RSA private key operations up to 20X faster than a high end server

Device Layer

Challenges to Securing IoT Devices

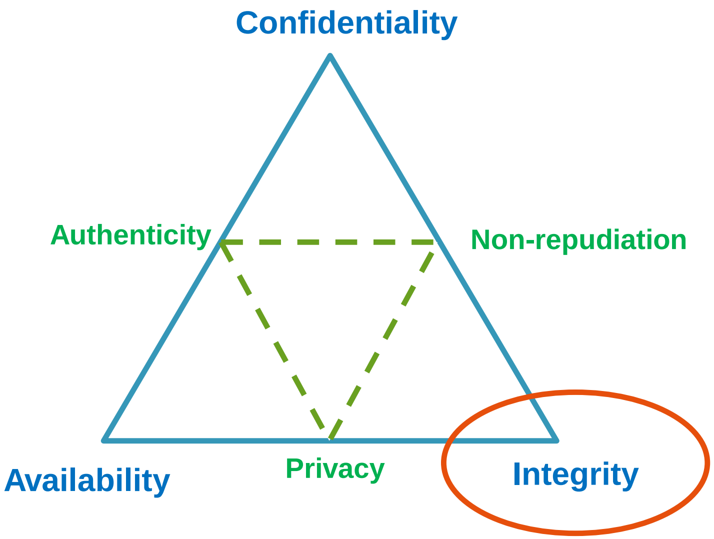
- Example IoT devices include: light bulb, thermostat, utility meter, door open sensor, wind turbine, vending machine, etc.

- Requirements which may conflict with security include:
  - Resource constrained / battery-powered operation
  - Remote location / inaccessibility
  - Ease of installation / use
  - Interoperability
  - Ad hoc / wireless connectivity
  - Low cost

Key Management Lifecycle Stages – as Defined by U.S. National Institute of Standards & Technology

- User Registration
- System and User Initialization
- Keying Material Installation
- Key Establishment
- Key Registration
- Operational Use
- Storage of Keying Material
- Key Update
- Key Recovery
- Key De-registration and Destruction
- Key Revocation

IoT Device Layer – Full Security May Not be Possible

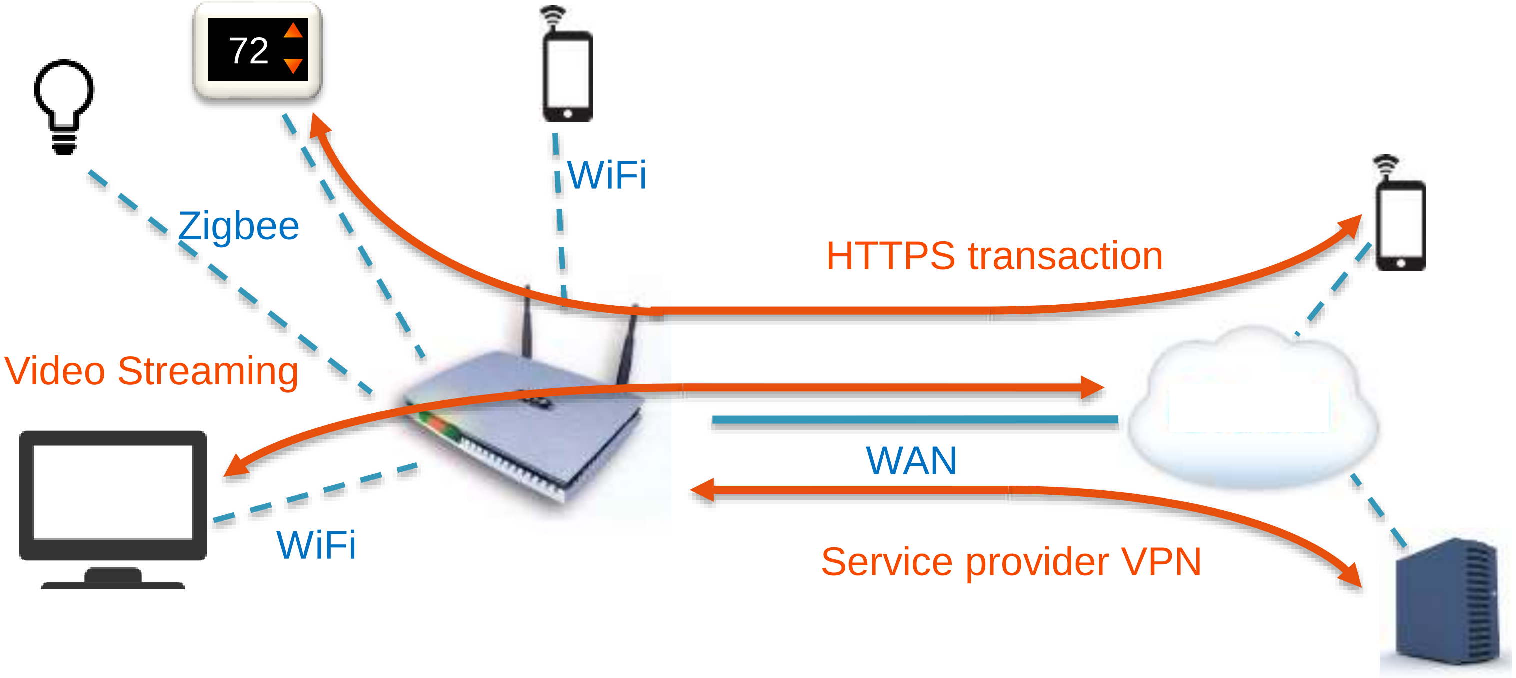


- Focus on most important / achievable aspects
  - Data can be trusted within the scope of the device's capability
- Authenticity & availability may be impossible to guarantee given the operating constraints
- Gateway layer can add confidentiality, authenticity and maybe availability (with use of redundant sensors)

Gateway Layer

What Does an IoT Gateway Do?

- Bridges Local Area Networks and protocols to the Wide Area Network of the Internet
- Speaks Internet Security (SSL, IPSec) to the WAN
- Speaks local security to IoT device layer (BLE, 802.15.4, etc.)



Critical Security Roles of the Gateway Layer

- Provide an isolating layer between cloud layer and device layer
  - Assume all IoT Devices are potentially compromised
  - IoT devices cannot gain access to cryptographic secrets from other devices or layers

- Classification-based control
  - IoT devices are only allowed to send / receive data that is appropriate to the class of device
    - For example:
      - An occupancy sensor should not be trying to browse the web
      - A factory robot should only connect to the company cloud via VPN

- Managing Quality of Service (QoS)
  - Ensure connections do not “hog” all bandwidth
  - Critical messages must get through

Security Requirements for IoT Gateways

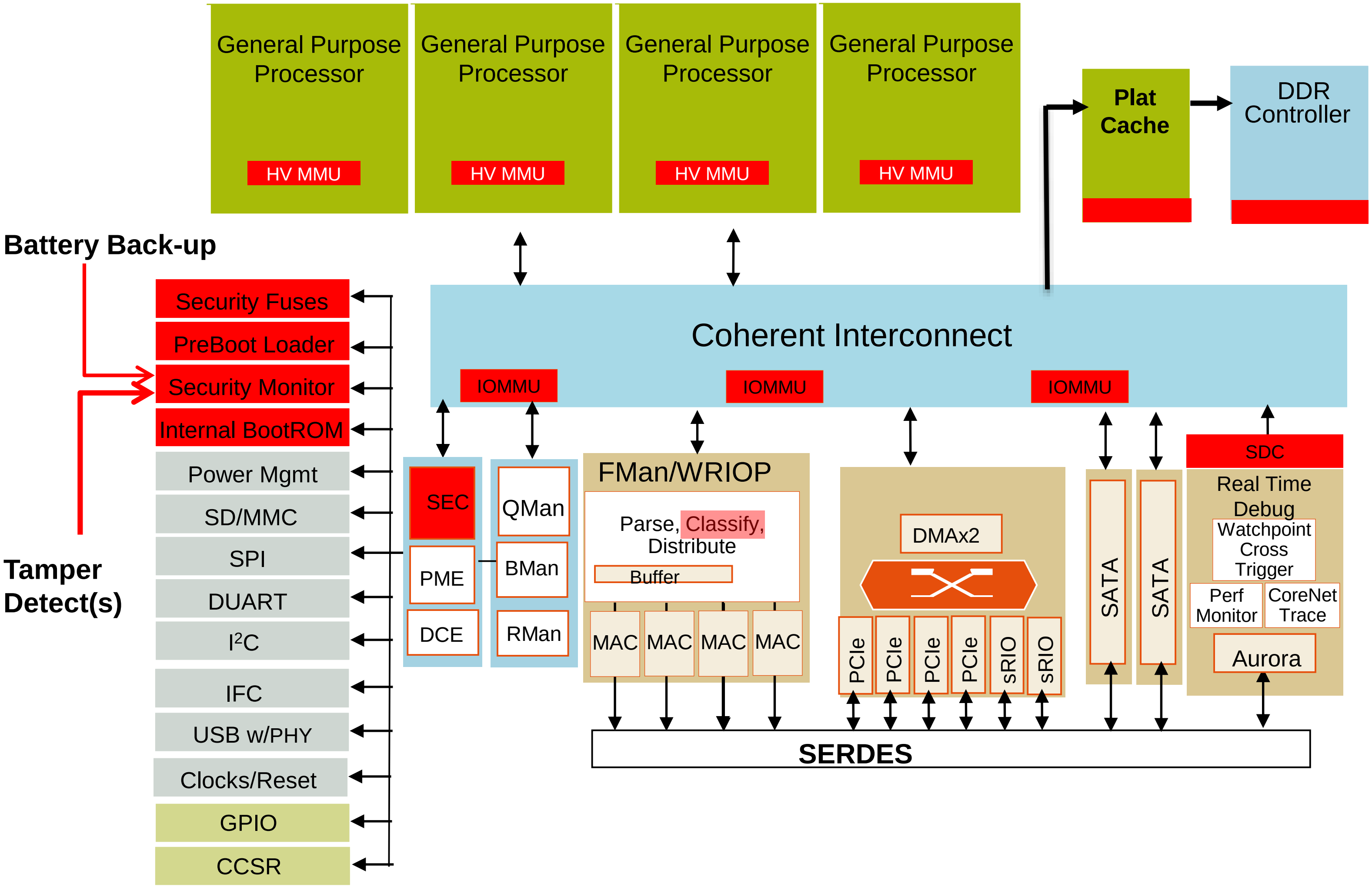
- Cryptographic acceleration
  - Symmetric & asymmetric crypto
  - Random number generation
  - Protocol acceleration

- Lifecycle management of cryptographic keys
  - Provisioning
  - Storage
  - Operation
  - Revocation / Destruction

- Hardware root of trust
  - Secure boot, secure debug, etc.

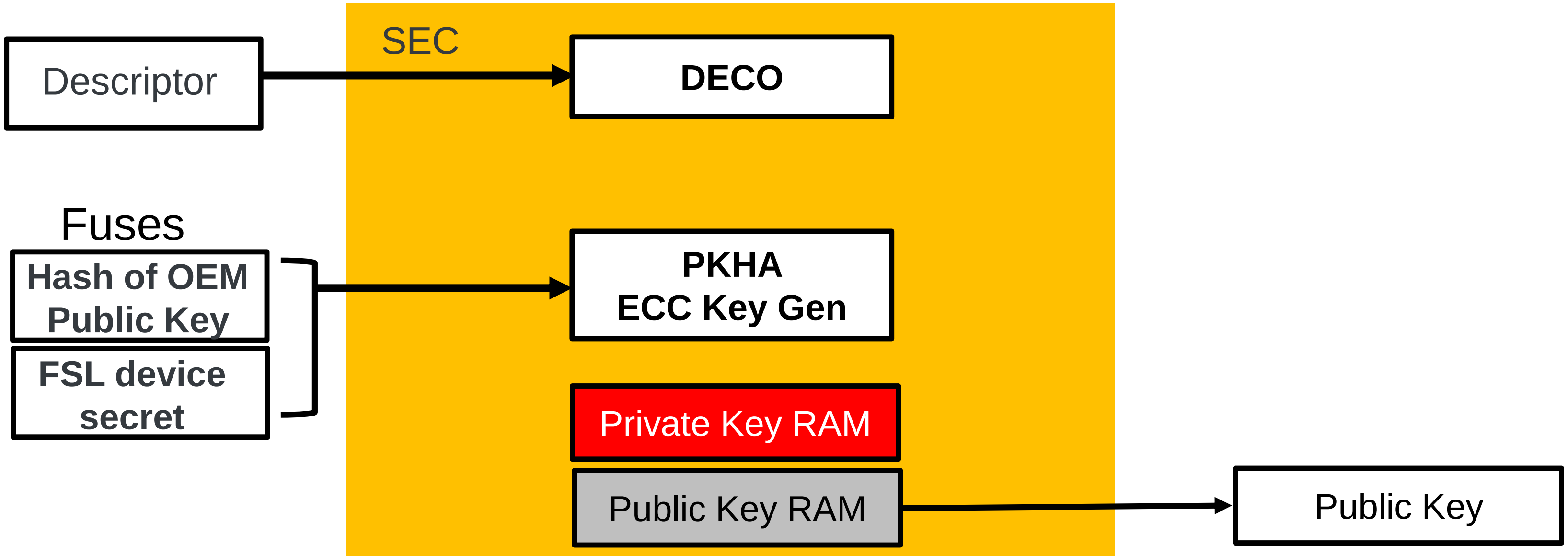
- Virtualization support
  - Isolate applications and dataflows

QorIQ Trust Architecture – Features Supporting Gateway Security



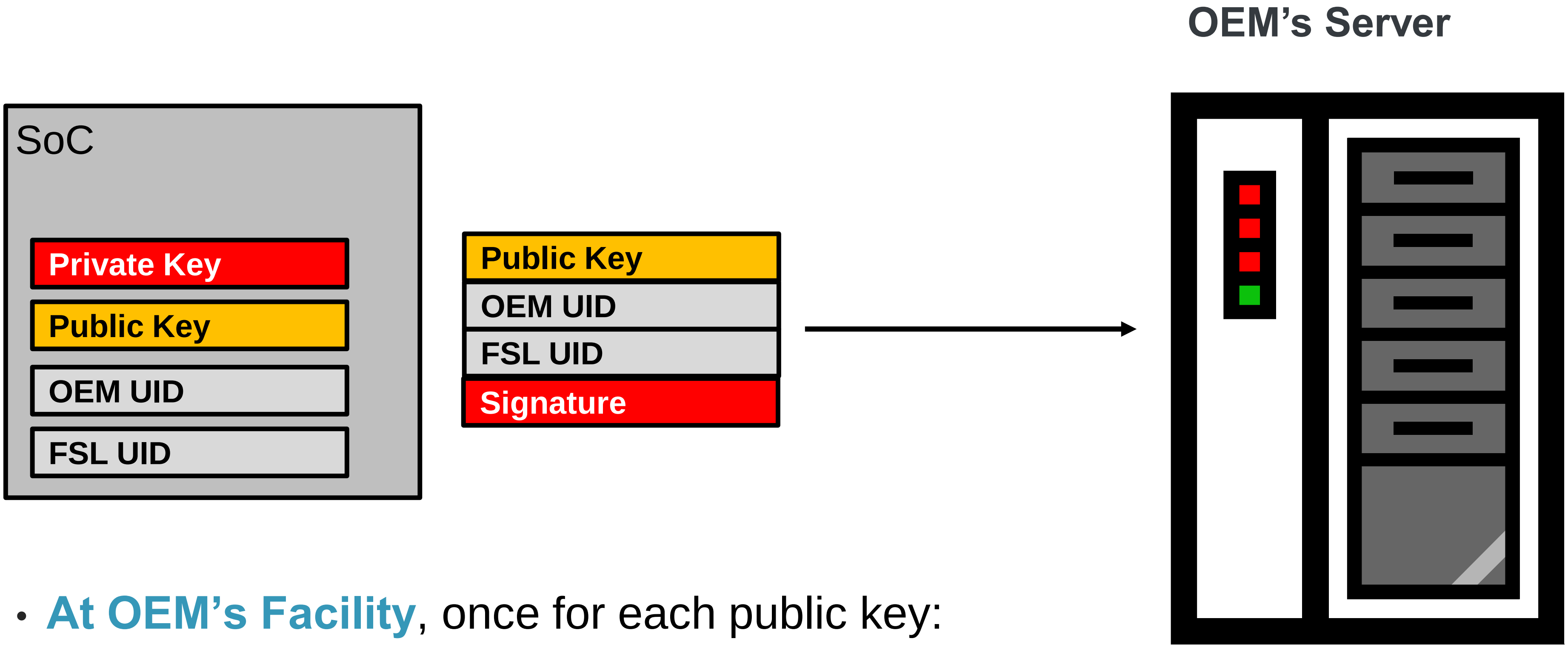
QorIQ Trust Architecture Secure Manufacturing Flow

Hardware Key Pair Generation



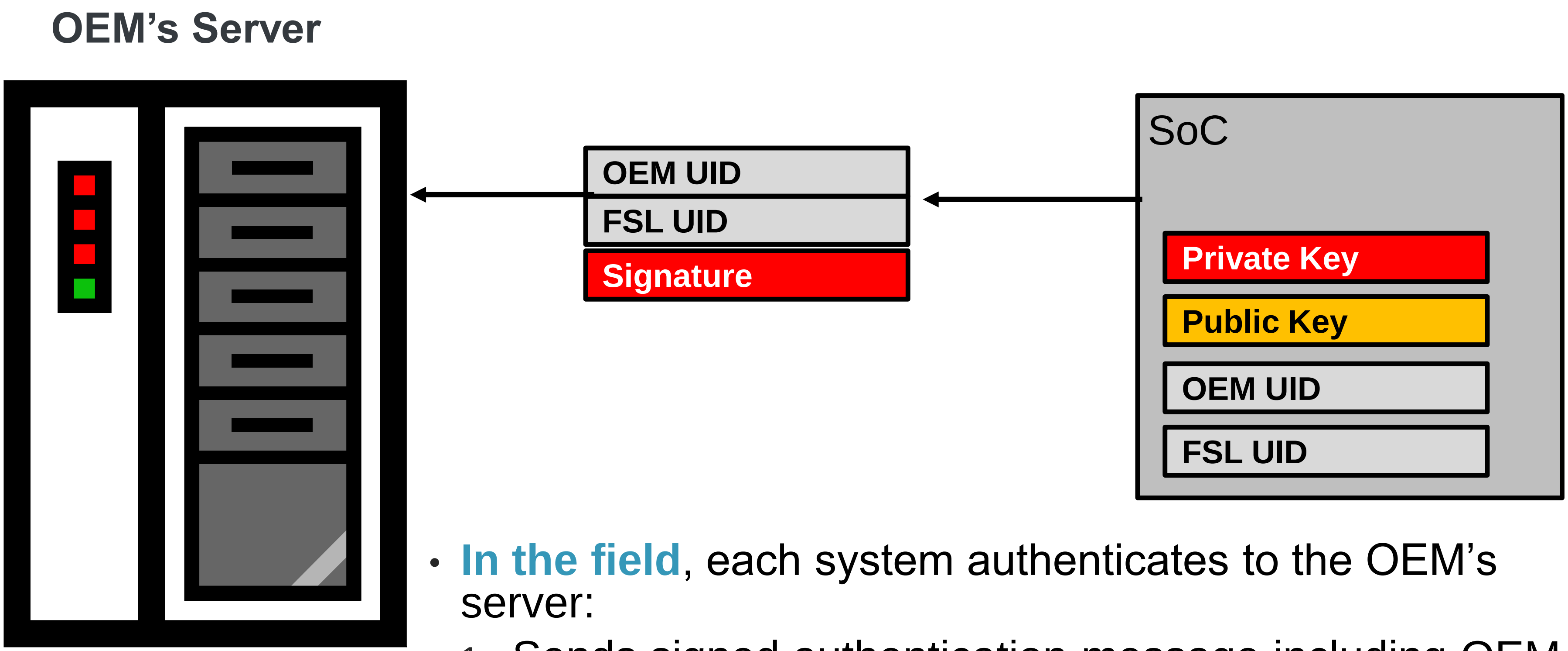
- Following successful secure boot, the SEC can be commanded to generate an ECC public/private key pair.
- The hash of the OEM programmed Public Key and a Freescale Secret Value are the inputs to the Key Gen process.
- Once the Hardware Key Pair is generated, the Public Key is optionally output. The Private Key isn’t readable by software, and cannot be output. It can only be used by the SEC.
- The same Hardware Key Pair is generated each time the Hardware Key Pair Generation is executed. The Keys are locked out & cleared in response to a security violation.

Trusted Manufacturing – Part 1



- **At OEM's Facility**, once for each public key:
  1. Program all fuses, including OEM Unique ID
  2. Perform the Hardware Key Pair Generation operation
  3. Export a file containing OEM & FSL Unique ID and public key, sign file with private key

Trusted Manufacturing – Part 2



- **In the field**, each system authenticates to the OEM's server:
  1. Sends signed authentication message including OEM & FSL Unique IDs
  2. Server verifies the message, confirms IDs match the public key used to verify the message
  3. Server completes systems provisioning (TFTP of additional software), checks this system off list to detect cloning attempts

Summary

IoT Security Summary

- The three different layers of the IoT have different security challenges
  - Cloud layer: transactions/sec & protection against cyber attacks
  - Device layer: often not feasible to implement fully secure device
  - Gateway layer: key role to isolate domains and translate local security protocols to internet formats
- Systems designers should consider what aspects of security are most important and focus on those
  - Proper isolation so that weakest node cannot compromise the whole
- QorIQ Trust Architecture is a great place to start!

More Information

Related FTF Sessions

- FTF-DES-F1239: SEC Software Tuning, Debug and Performance for QorIQ Processors
- FTF-SNT-F1234: Overview of Secure Embedded Processing in QorIQ Platforms
- FTF-DES-F1216: Security 101: Introduction to Cryptographic Accelerators and Security Software for QorIQ Processors
- FTF-SNT-F1311: Configuring Secure Boot and Secure Virtualization in QorIQ Processors
- FTF-SNT-F1312: Secure your system with QorIQ LS Series: Implement Secure Boot and ARM TrustZone® Technology
- FTF-DES-F1186: QorIQ Trust Architecture 202 – Advanced Techniques
- FTF-SNT-F1233: Security Solutions for High Performance Data-Center Applications

Other Useful Links

- NIST Key Management Guidelines: http://csrc.nist.gov/groups/ST/toolkit/key_management.html
- http://www.trustedcomputinggroup.org/resources/securing_the_internet_of_things_new_ways_to_deploy_trust_in_enterprise_computing_beyond_the_pc

Freescale Has World Class Support….and MORE

Global Technical Information Center
Design & Support Resource

Networking Applications Team
Depth of Expertise & Knowledge

Design With Freescale, Freescale Technology Forum
Training

Networking Software and Services Group

- Commercial Solutions
- Engineering Services
- Guaranteed Performance
- Service Level Agreement Support…and MORE

- Visit Pedestal 415 in the Technology Lab freescale™

www.Freescale.com