# Hardware Security Requirements for Vehicle-to-Everything Communications

William Whyte, CTO, Onboard Security

2017-05-16

OnBoardSecurity

# Overview

- Provide an overview of the Connected Vehicle security subsystem and current deployments
- Describe the platform and hardware security requirements that have been considered in trial deployments to date
- Outline the current requirements for Pilot Deployments
- Provide an overview of ongoing research and specification efforts and likely timelines for a final set of requirements
- Present on current certification practices for V2X security and how they might evolve
- Note: device requirements go beyond simple crypto module specifications – we will try to illuminate the interaction between the two

# Traffic Safety

- 32,000  US road deaths, and 3,800,000 injuries
- Fatalities and injuries = $300B/year
- Congestion = $230B/year
- Leading cause of death for ages 15-34 in US

Technology Evolution

*Passive → Active → Proactive*

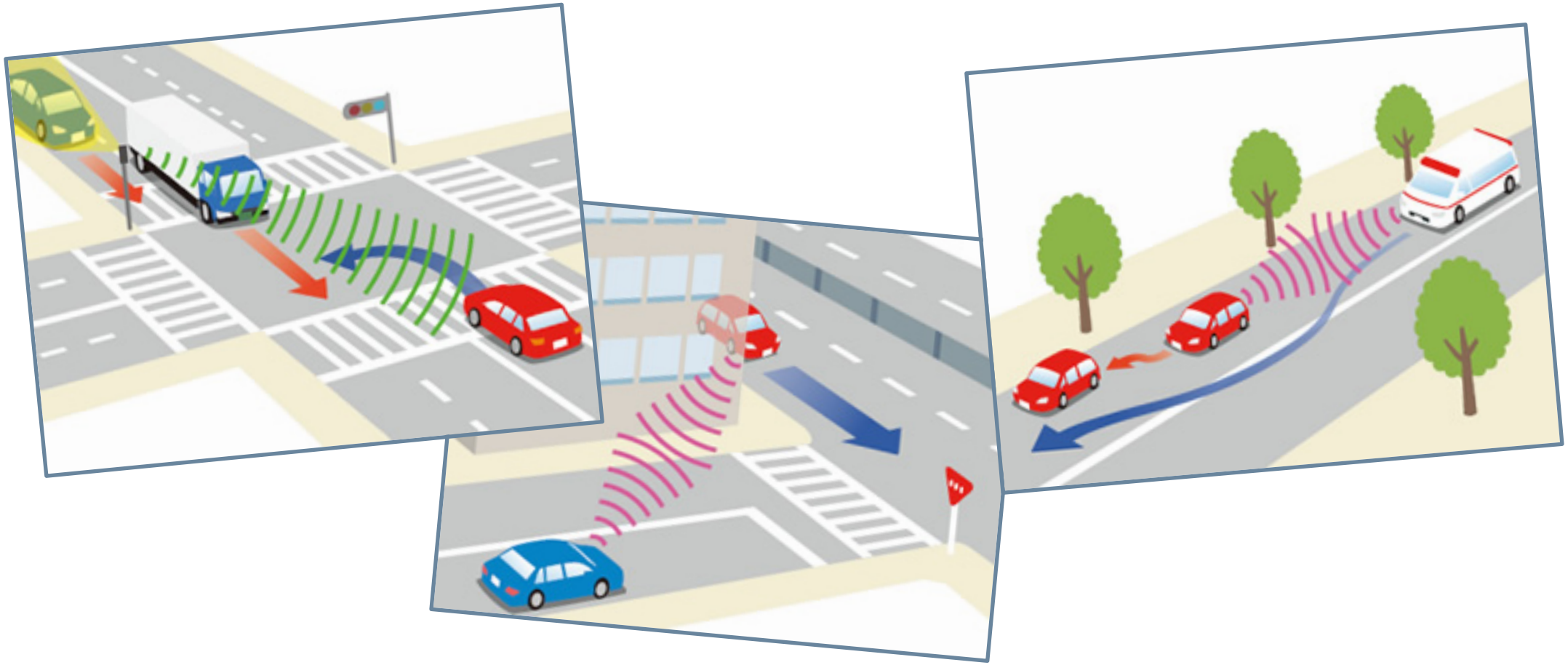
# Reduce accidents

- Drivers a causal factor for at least 80% of all crashes
- Address by short-range radar or by improved data connection
- Short-range radar: automatic driving – useful at low speed


# V2V

- Significant reduction in deaths may be possible from V2V wireless communications for 360o warning applications.
  - 300 m range, 802.11-derived medium access
  - Basic Safety Message (BSM)
    - Contains location, velocity, steering angle…
    - Transmitted up to 10x second
- Allows receiving unit to predict collisions and warn driver
  - “Prevent 80% of unimpaired 2-vehicle accidents”
- The availability of wireless communications may also enable other applications
  - Signal phase and timing
  - Point of interest notification

# Vehicle-to-anything (V2X) communications

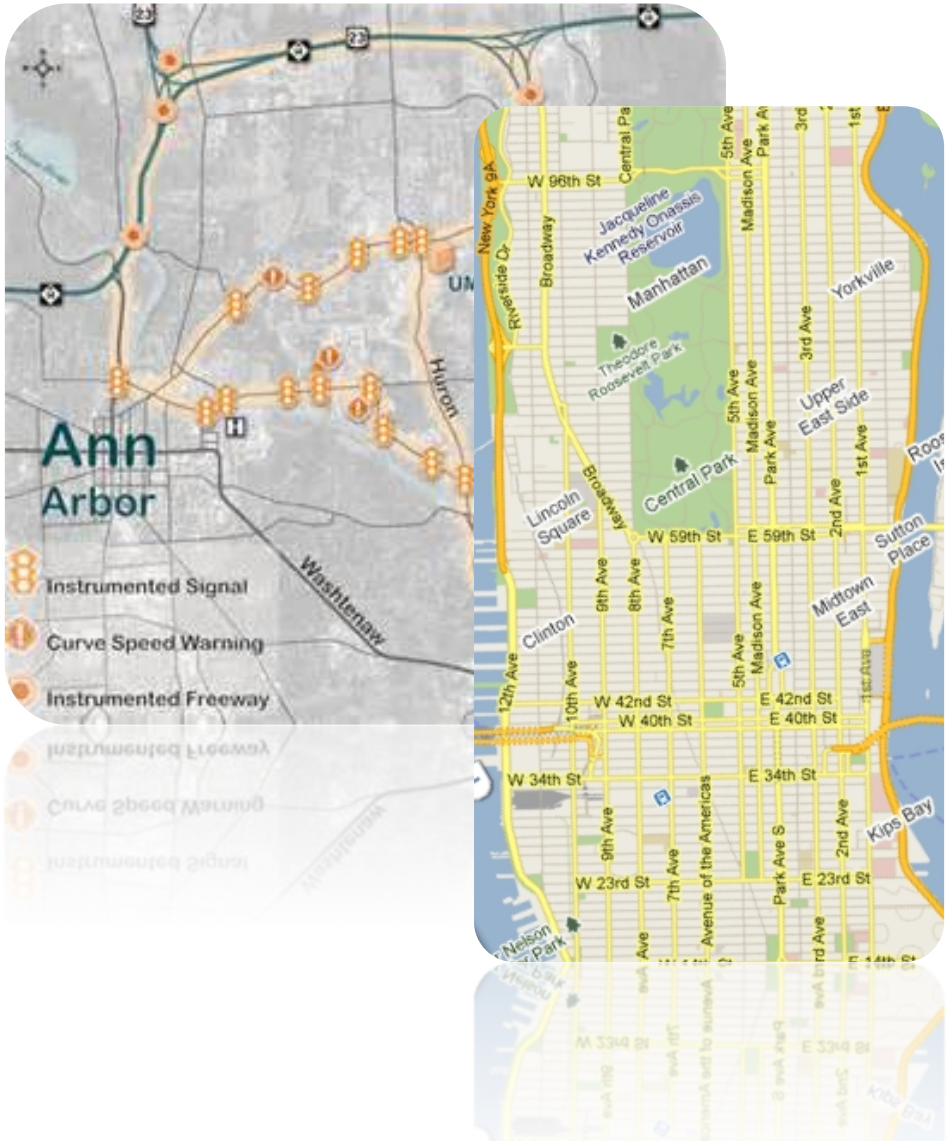




Illustrations from https://www.itsconnect-pc.org/en/about_its_connect/service.html

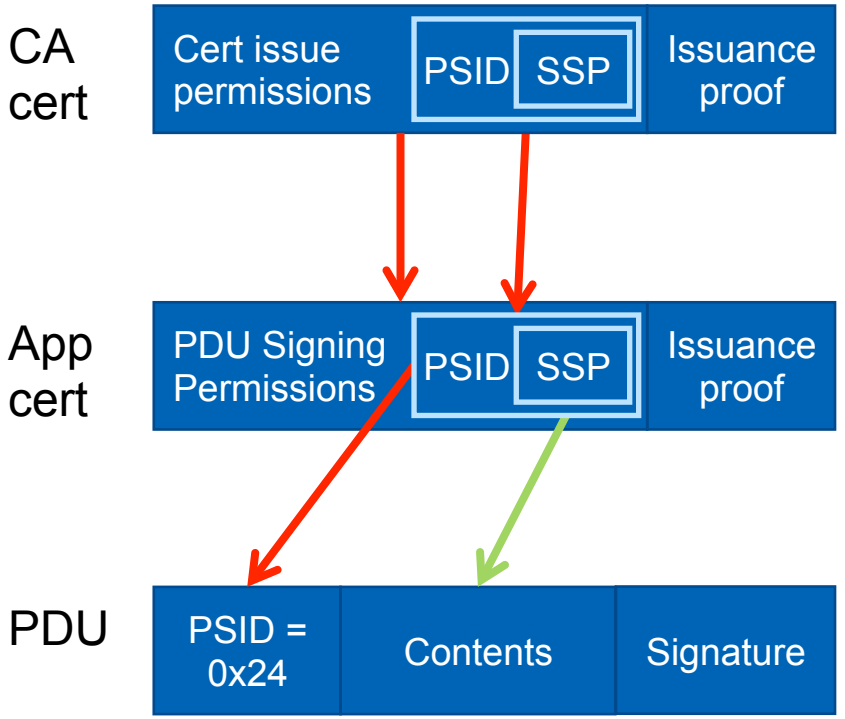
# V2X Pilots

- Ann Arbor Safety Pilot – 2,500 vehicles initially, extended to 30,000
- New York, 8,000 vehicles testing city safety
- Tampa, better freeway management
- Wyoming, improving I80 trucking efficiency
- GM deploying on 2017 Cadillac CTS
- Many EU and Asia Pacific pilots
- All major manufacturers engaged



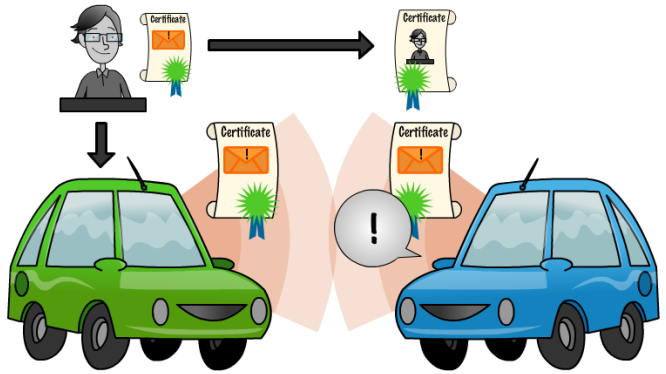
# Trust model

- IEEE 1609.2 / ETSI TS 103 097
  - Secure messages and certificates, targeted at MANET setting
- Signed PDUs are authorized by certificates
  - PSID: Identifies “application”
  - Service Specific Permissions (SSP): permissions within application
- CA ensures that sender is entitled to these permissions
  - Implications for hardware and software security, data quality
- Receiver checks PDU is consistent with permissions
- Different applications may use different mechanisms but many common applications use this approach



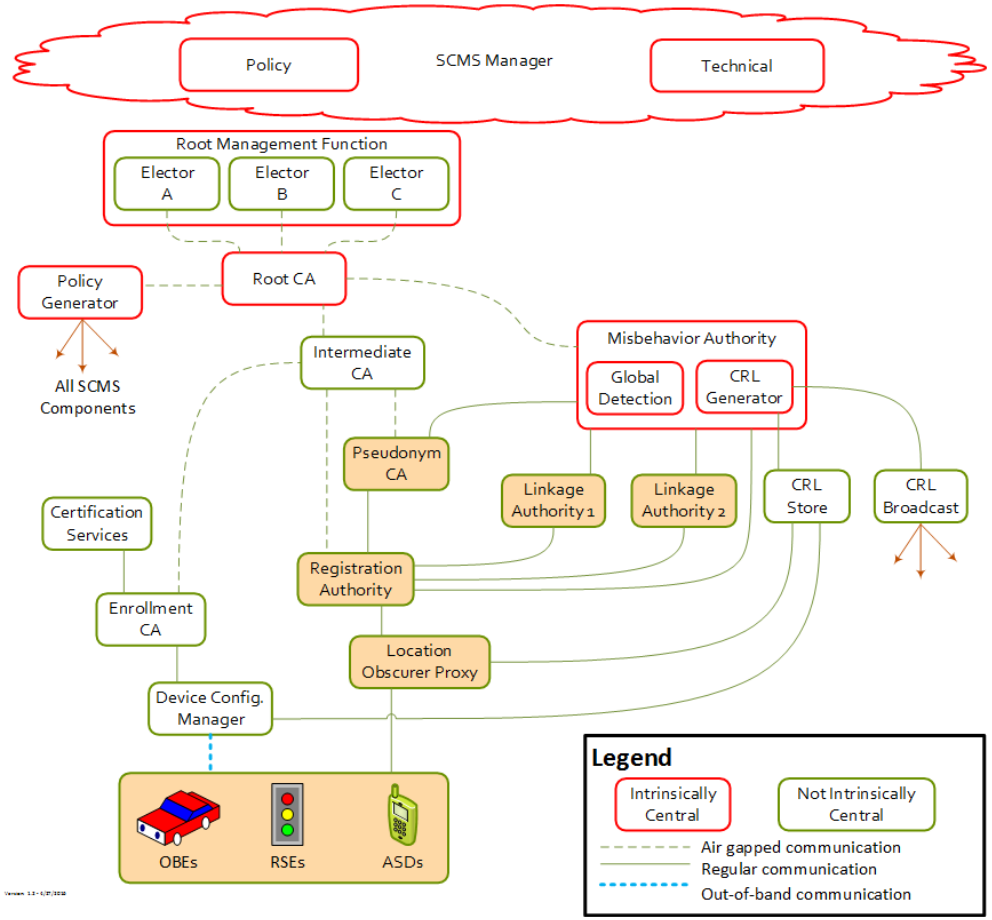
# Trust model example and implementation

- Cooperative Awareness Message (EU): “Here I am”
  - Identified by ITS-AID 0x24
- Default (NULL) SSP: cert owner can send “here I am” message only
- SSP 00 00 40: cert owner can claim to be emergency vehicle, request right of way
- Receiver of a CAM checks that CAM payload is consistent with both CAM PSID and sender-specific SSP
  - This must be carried out by CAM processing logic
  - Cannot be carried out by the security services



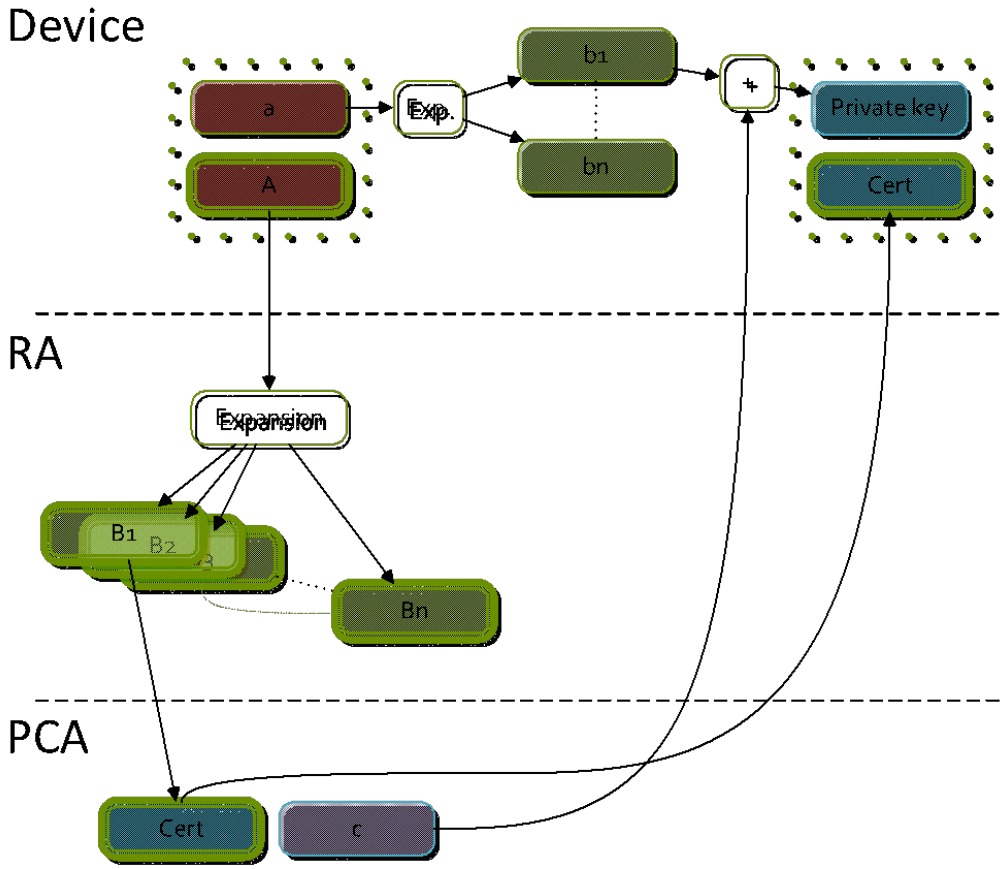

# Certificate issuance

- Secure Credential Management System (SCMS – think PKI-on-steroids) for V2V includes privacy-preserving mechanisms
- Shuffle at RA to protect against CA learning certificates
- Linkage authorities to allow tracing misbehaving devices without revealing their identity, and revoking in a way that only allows them to be tracked after revocation
- Organization separation ensures no single insider / no single database breach can track any car



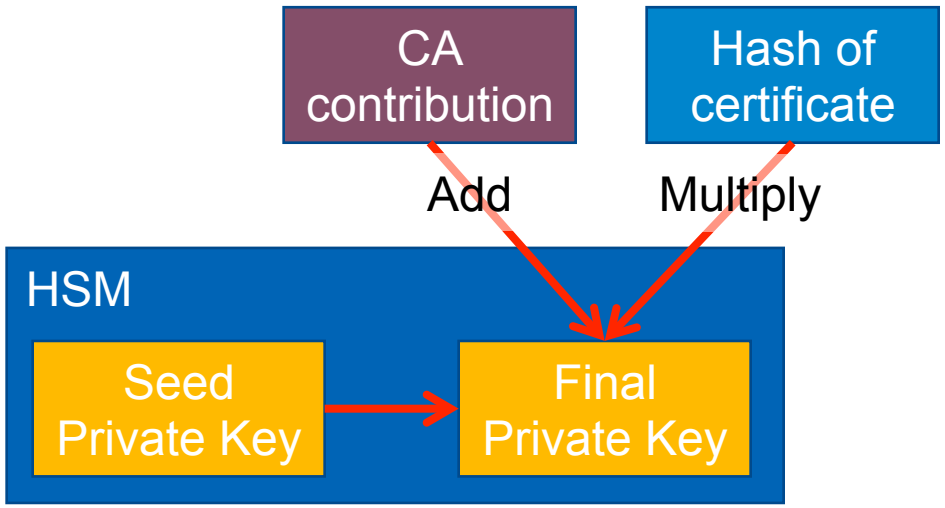
# Unique aspects of V2V hardware

- Butterfly keys



# Unique aspects of V2V hardware

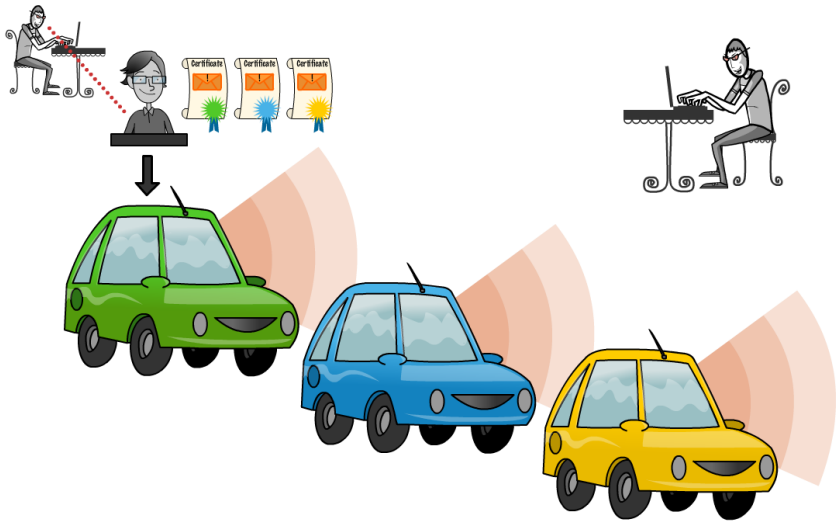
- Butterfly keys
- Implicit certificates




- CA has private/public key (u, U = uG)
- For each request, CA:
  - Generates random integer c
  - Calculates C = cG
  - Calculates “Reconstruction Value” R = [A + f(i,j) G] + C
- Public key associated with cert: H(Cert) * R + U
- Private key: H(Cert)*r + u
  - = H*(a + f(i,j) + c) + u
- CA provides the cert and H*c + u to the device:
  - Device can calculate H*(a + f(i,j) locally and so recover private key
  - No-one else knows a, so no-one knows private key
  - The value H*c completely hides u, i.e. no information is leaked to the device about the CA’s private key

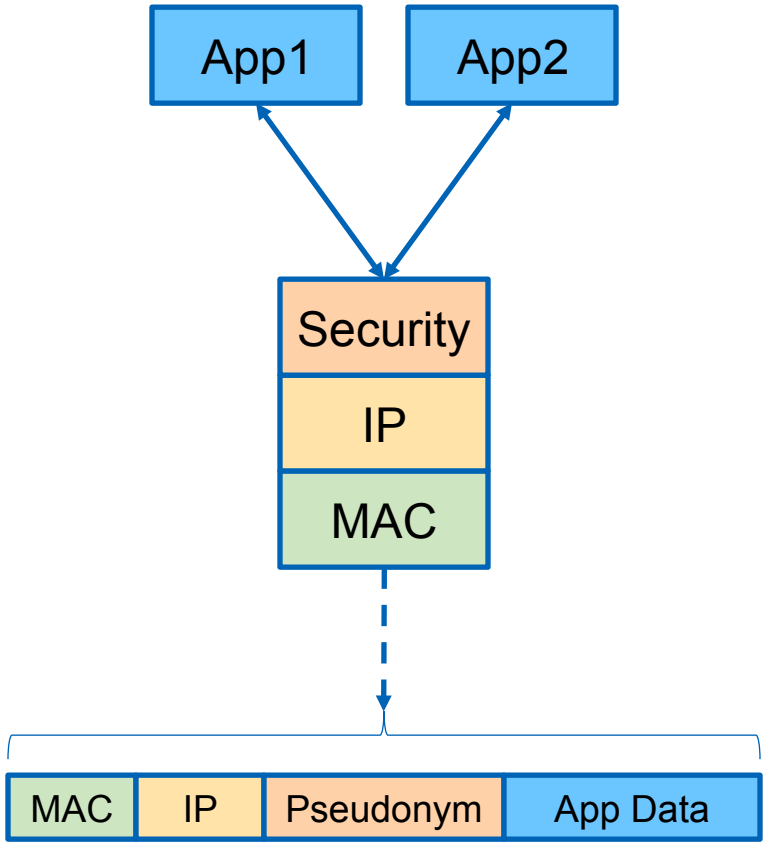
# Unique aspects of V2V hardware

- Butterfly keys
- Implicit certificates
- Lots of certificates for a single application



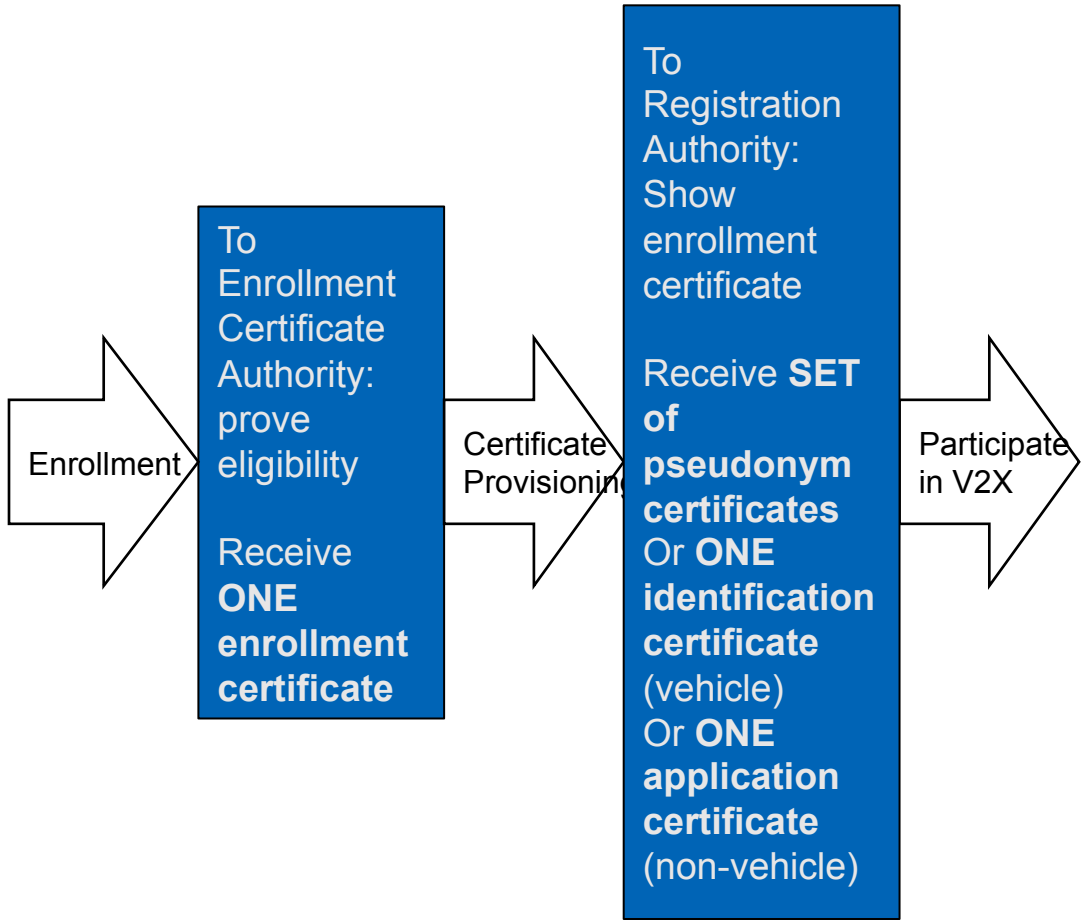
# Unique aspects of V2V hardware

- Butterfly keys
- Implicit certificates
- Lots of certificates for a single application
- Different applications with different certificates



# Unique aspects of V2V hardware

- Butterfly keys
- Implicit certificates
- Lots of certificates for a single application
- Different applications with different certificates
- Certificates automatically issued



# Unique aspects of V2V hardware

- Butterfly keys
- Implicit certificates
- Lots of certificates for a single application
- Different applications with different certificates
- Certificates automatically issued

- Many vendors already produce HSMs that support the necessary operations
  - Autotalks
  - NXP
  - Renesas

# Getting a certificate means you're trusted to be secure enough

Where do security requirements come from?

What are they?

# Deriving security requirements

- Threat model is application specific
- Connected Vehicle applications – not just vehicles!
  - Traffic signals, signs, gates, toll plazas, back-end systems, …
  - Signal preemption, border crossing, pedestrian in signalized intersection warning...
- Many devices playing many different roles in different applications

# Threat model for collision avoidance

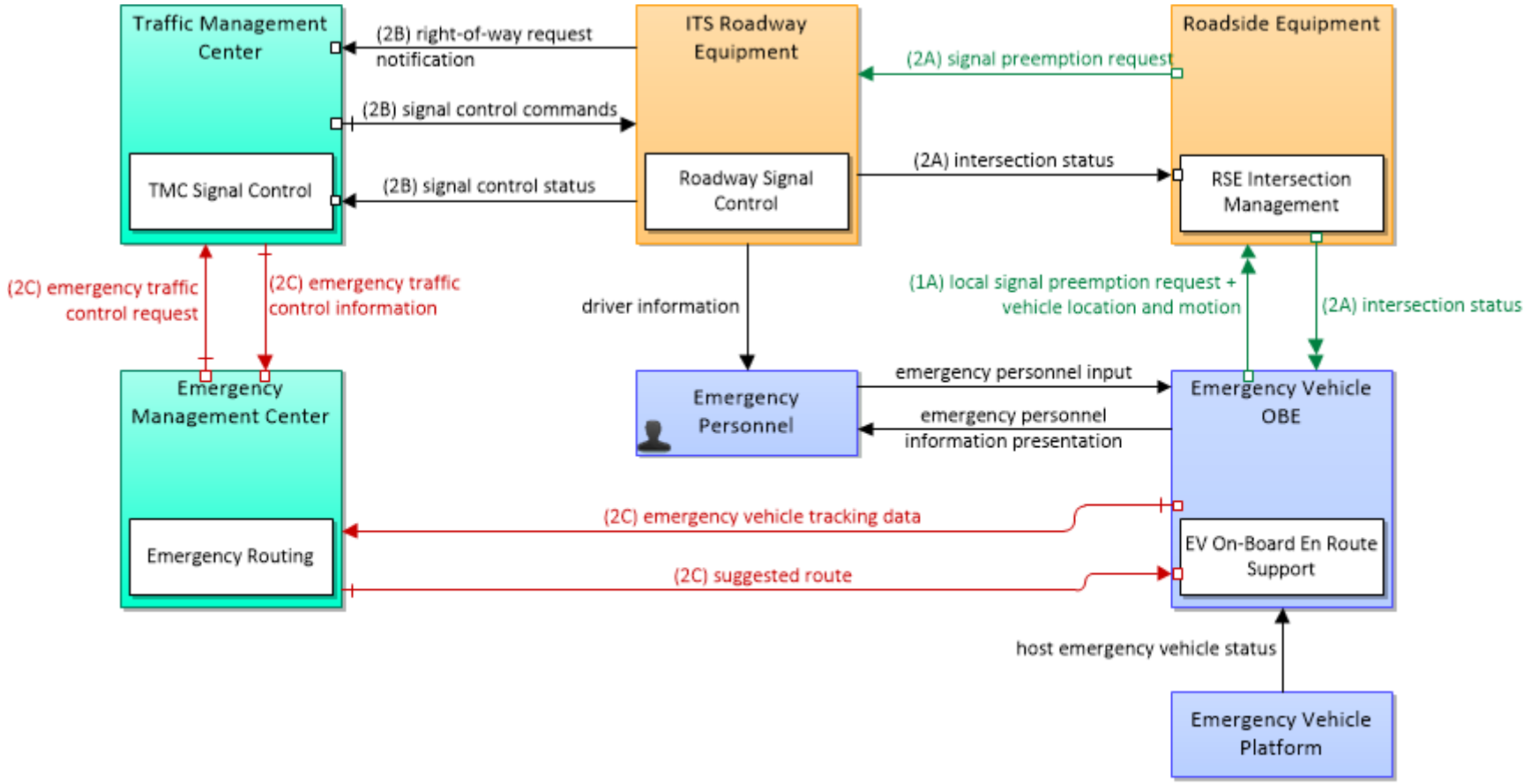
- False positives
  - Unlikely to cause physical harm
  - “Something bad round the corner! swerve now!”
  - But invalid alerts reduce driver faith in system
  - Appropriate security approach: Authentication + misbehavior detection

- False negatives
  - Need to warn about denial of service once system is widely deployed

# General Approach

- FIPS-199 Confidentiality, Integrity, Availability (C-I-A) low/medium/high (little/significant/catastrophic)
- Focus on the flow of information/data in and out of the device
- Analyze a representative subset of projected applications from the CVRIA
- Use the devices data C-I-A requirements to derive the device's requirements
- See which C-I-A combinations turn up most frequently
- Define a "minimal useful" set of device classes that cover all the identified C-I-A combinations
- Specify security controls for each device class based on NIST SP 800-53

# Emergency Vehicle Preemption




| Emergency Vehicle Preemption | | | |
|---|---|---|---|
| 2 | Physical | Jun 17, 2014 | NAT |

# Example flow

### *3.2.2.6 EV OBE -> RSE: Local Signal Preemption Request*

Direct control signal or message to a signalized intersection that results in preemption of the current control plan and grants right-of-way to the requesting vehicle.

---

- **Confidentiality**: **LOW**. It does not matter if someone is able to eavesdrop on this request. There will be many other more obvious indicators that the request was made, such as sirens and flashing lights on the emergency vehicle.
- **Integrity**: **HIGH**. The system must be able to trust these requests. Emergency Vehicles should be able to send these requests and know that they are being operated on by the receiving system. Additionally, if an unauthorized vehicle is able to send these requests it could bring traffic to a standstill by disrupting signal coordination citywide.
- **Availability**: **MEDIUM**. The alternative to this request is existing mechanisms – such as using the sirens to stop traffic. The difference between the emergency signal preemption application and existing practice is not significant enough to justify a HIGH availability requirement.

# Device Requirements

- Roadside Equipment (3.2.3.6) for EVP

| Information flow name | Inbound / Outbound | C | I | A |
|---|---|---|---|---|
| EV OBE -> RSE: Local Signal Preemption Request | I | L | H | M |
| EV OBE -> RSE: Vehicle Location and Motion | I | L | M | M |
| ITS RE -> RSE: Intersection Status | I | L | M | M |
| RSE -> EV OBE: Intersection Status | O | L | M | M |
| RSE -> ITS RE: Signal Preemption Request | O | L | H | M |
| **Overall** | | **L** | **H** | **M** |

# Deriving device classes

- (L, M, L): 3 entries: Includes many typical uses of OBEs used to exchange information that is publicly available and used for mobility or efficiency applications
- (L, M, M): 1 entry: Includes the RSE used to exchange information that is publicly available and used for safety applications
- (L, H, M): 4 entries: Includes the RSE and field devices when they are used to transfer data used for safety applications
- (M, M, L): 2 entries: Includes the field equipment when it is used to transfer data related to border security
- (M, M, M): 1 entry: Includes the RSE when it is used to transfer data related to border security
- (M, H, M): 3 entries: Includes the RSE and devices in emergency vehicles that transfer time sensitive information that should not be impersonated
- (H, H, M): 1 entry: Includes the field devices that manage personal information that may have national security implications.

| Confidentiality | Integrity | Availability | Number of entries | Level up to |
|---|---|---|---|---|
| L | L | L | 0 | |
| L | L | M | 0 | |
| L | L | H | 0 | |
| L | M | L | 3 | L, M, M |
| L | M | M | 1 | Self |
| L | M | H | 0 | |
| L | H | L | 0 | |
| L | H | M | 4 | M, H, M |
| L | H | H | 0 | |
| M | L | L | 0 | |
| M | L | M | 0 | |
| M | L | H | 0 | |
| M | M | L | 2 | M, M, M |
| M | M | M | 1 | Self |
| M | M | H | 0 | |
| M | H | L | 0 | |
| M | H | M | 3 | Self |
| M | H | H | 0 | |
| ... | | | 0 | |
| H | H | L | 0 | |
| H | H | M | 1 | Self |
| H | H | H | 0 | |

# NIST SP 800-53 Security Controls

- Maintenance
- Media Protection
- Physical & Environmental Protection
- Planning
- Personnel Security
- Risk Assessment
- System & Communication Protection
- System & Information Integrity

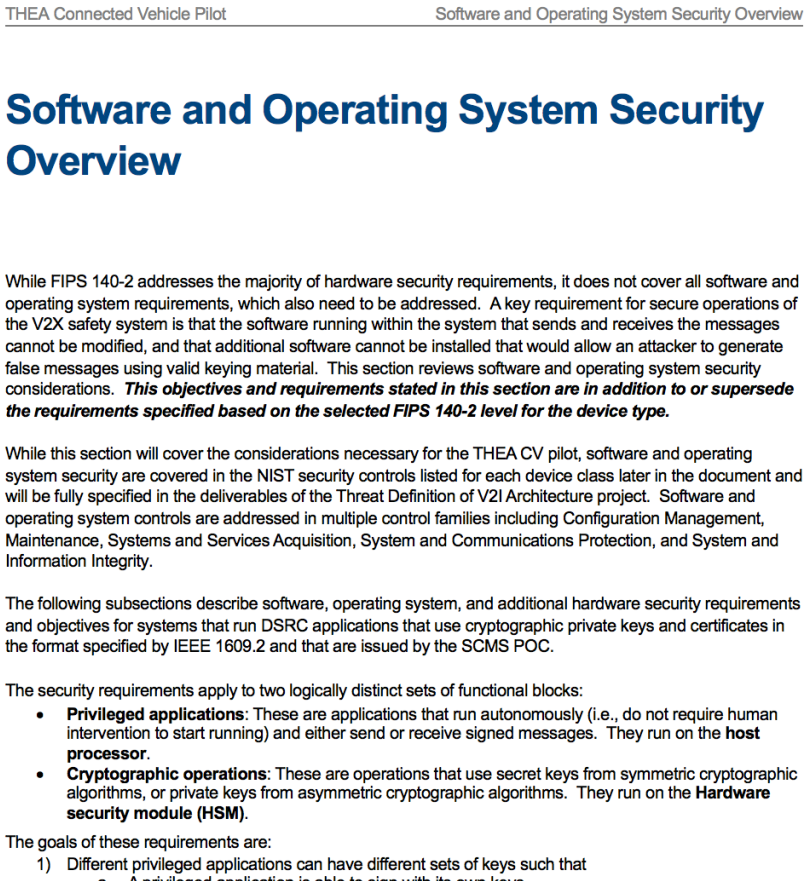
- … work is ongoing to produce definitive list of controls

- Access Control
- Awareness & Training
- Audit and Accountability
- Security Assessment & Authorization
- Configuration Management
- Contingency Planning
- Identification & Authentication
- Incident Response

- Note, this goes considerably beyond standard FIPS 140 cryptographic module considerations

# Currently available documents

- RSU Specification v 4.0
  - SNMP v3 and SSH, no telnet
  - Currently being updated
  - Doesn't map to device classes as it predates the analysis
- Hardware/Software/OS Security Requirements developed within CV Pilot Deployments
  - Not a complete set of controls but a good starting point
  - Security Management Operating Concepts for different Pilot Deployments include the same set of requirements
  - Requires secure boot and signed software updates





## Software and Operating System Security Overview

While FIPS 140-2 addresses the majority of hardware security requirements, it does not cover all software and operating system requirements, which also need to be addressed. A key requirement for secure operations of the V2X safety system is that the software running within the system that sends and receives the messages cannot be modified, and that additional software cannot be installed that would allow an attacker to generate false messages using valid keying material. This section reviews software and operating system security considerations. ***This objectives and requirements stated in this section are in addition to or supersede the requirements specified based on the selected FIPS 140-2 level for the device type.***

While this section will cover the considerations necessary for the THEA CV pilot, software and operating system security are covered in the NIST security controls listed for each device class later in the document and will be fully specified in the deliverables of the Threat Definition of V2I Architecture project. Software and operating system controls are addressed in multiple control families including Configuration Management, Maintenance, Systems and Services Acquisition, System and Communications Protection, and System and Information Integrity.

The following subsections describe software, operating system, and additional hardware security requirements and objectives for systems that run DSRC applications that use cryptographic private keys and certificates in the format specified by IEEE 1609.2 and that are issued by the SCMS POC.

The security requirements apply to two logically distinct sets of functional blocks:

- **Privileged applications**: These are applications that run autonomously (i.e., do not require human intervention to start running) and either send or receive signed messages. They run on the **host processor**.
- **Cryptographic operations**: These are operations that use secret keys from symmetric cryptographic algorithms, or private keys from asymmetric cryptographic algorithms. They run on the **Hardware security module (HSM)**.

The goals of these requirements are:

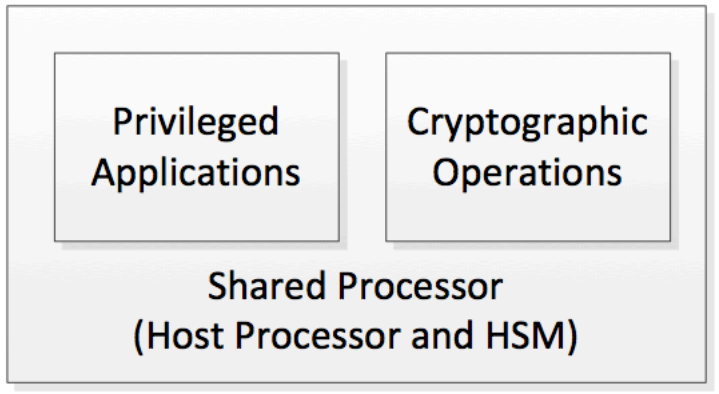
1) Different privileged applications can have different sets of keys such that

# Goals of SCMS Device Requirements for CV Pilots

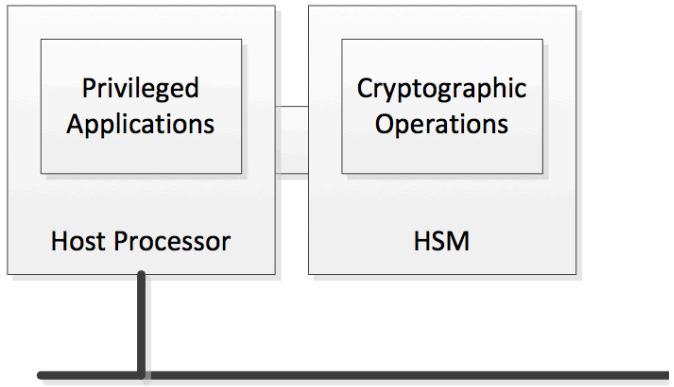
- Enable Privileged applications to securely use different cryptographic keys
  - Privileged vs. unprivileged applications
  - Privileged application crypto key access
- Prohibit read access to key material!
- Public key (signature verification) protected from unauthorized replacement
- Control access to data by application
- Secure software/firmware update
- https://wiki.campllc.org/display/SCP/Hardware%2C+Software+and+OS+Security+Requirements

# Architecture Types

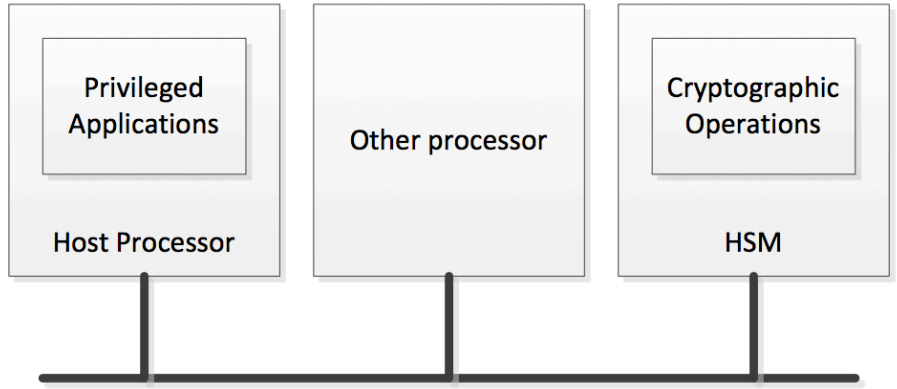
- Integrated




- Connected




- Networked



# Host Processor Requirements

- States
  - Manufacturing State
    - Initialization (fewer protections)
  - Operational States
    - All protections
- Switching from Operational to Mfg. State
  - Transition shall wipe all privileged applications from the host processor and all keys from the HSM
  - Guaranteed local, physical presence allows unauthenticated command to return to Mfg. state

# Host Processor Requirements (Cont.)

- Secure Boot
  - Host Processor power-up integrity test required
  - Must use hardware-protected root of trust
- Host processor's conditional / continuous integrity tests
  - No signing allowed until checks have passed
  - No private key access whatsoever until checks have passed
  - No privileged application may operated with failed test
  - Checks Root CA keys/certs for lack of modification/substitution since last access
    - Any failures? Device shall reject all incoming signed messages that chain back to those root CA certificates as invalid
  - Integrity checks shall be supported by a root of trust protected by hardware of the physical security level appropriate to the device class

# Host Processor Requirements (Cont.)

- OS
  - System security policy (capabilities vs. applications) and mandatory access control
  - System security policy enforcement
    - boot-time configuration
    - Non-modifiability during runtime
  - Privileged vs. unprivileged application separation
  - Guaranteed access/cycles to critical applications and OS itself
  - Mandatory Access Control (MAC)
    - Application mapping to HSM-protected private keys, protected, plaintext memory areas, etc.
    - Data read-access permissions as well as cryptographic key input permissions
  - Unauthenticated vs. ‘optionally authenticated’ services

# Host Processor Requirements (Cont.)

- OS (cont.)
  - Non-modifiability of running code - All running applications must correspond to an unmodified signed executable image
  - Safety vs. non-safety process prioritization
  - Support for testing tools that perform security related analysis
    - static source code analysis, run time error checking, stack overflow checks, and MISRA (Motor Industry Software Reliability Association) rules conformance
- Secure Updates
  - All software signed by manufacturer
  - Host processor successfully verifies signed image before installation
  - The integrity of the verification key protected by local hardware, either by:
    - directly storing the key in local hardware, or
    - by creating a chain of trust from the key to a hardware-protected key

# HSM Requirements

- Cryptographic code developed/installed, protected from unauthorized disclosure & modification
- FIPS-Approved integrity technique applied to cryptographic SW/FW components in HSM
  - MACs can only be used if the MAC key is UNIQUE to the HSM and the HSM MAC key and computation are internal-only
- All cryptographic SW/FW, keys, control/status info that are under OS control -> OS meets functional requirements of FIPS 140-2 Annex B listed CC Protection Profiles (or equivalently-trusted OS)
- Discretionary access controls, roles/services mappings and enforcement
- Crypto logic protection from operators & processes
- Approved random number generators in FIPS 140-2 Annex C (2016 draft)

# HSM Requirements / Device Classes

- Low/Medium confidentiality & Medium integrity
  - HSM -> tamper-evident hardware equivalent to FIPS 140-2 level 2 physical security
  - Secure boot -> supported by tamper-evident hardware equivalent to FIPS 140-2 level 2 physical security
  - HSM OS -> capable of evaluation at EAL 2 (per FIPS 140-2 level 2 requirements)
- Medium confidentiality & High integrity
  - Same as above, but FIPS 140-2 level 3 security and EAL3 for OS
- High confidentiality and High integrity
  - Same as above, but FIPS 140-2 level 3 security and EAL4 for OS

# Enforcement

- Vendors self-certify for Pilot Deployments
- There are interoperability tests but no platform security tests yet
- Note, FIPS 140-2 level 2 or 3 hardware *equivalent*

# Alternatives, futures, and certification

# Automotive Secure Hardware - Timeline

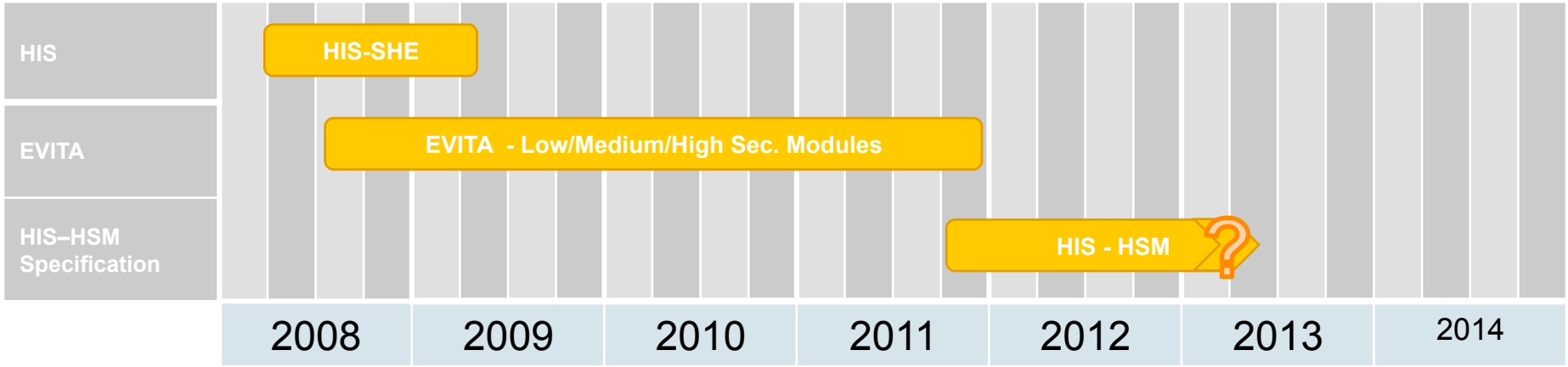



Illustration adapted from Jurgen Franck, Freescale / NXP

# Evita and Oversee

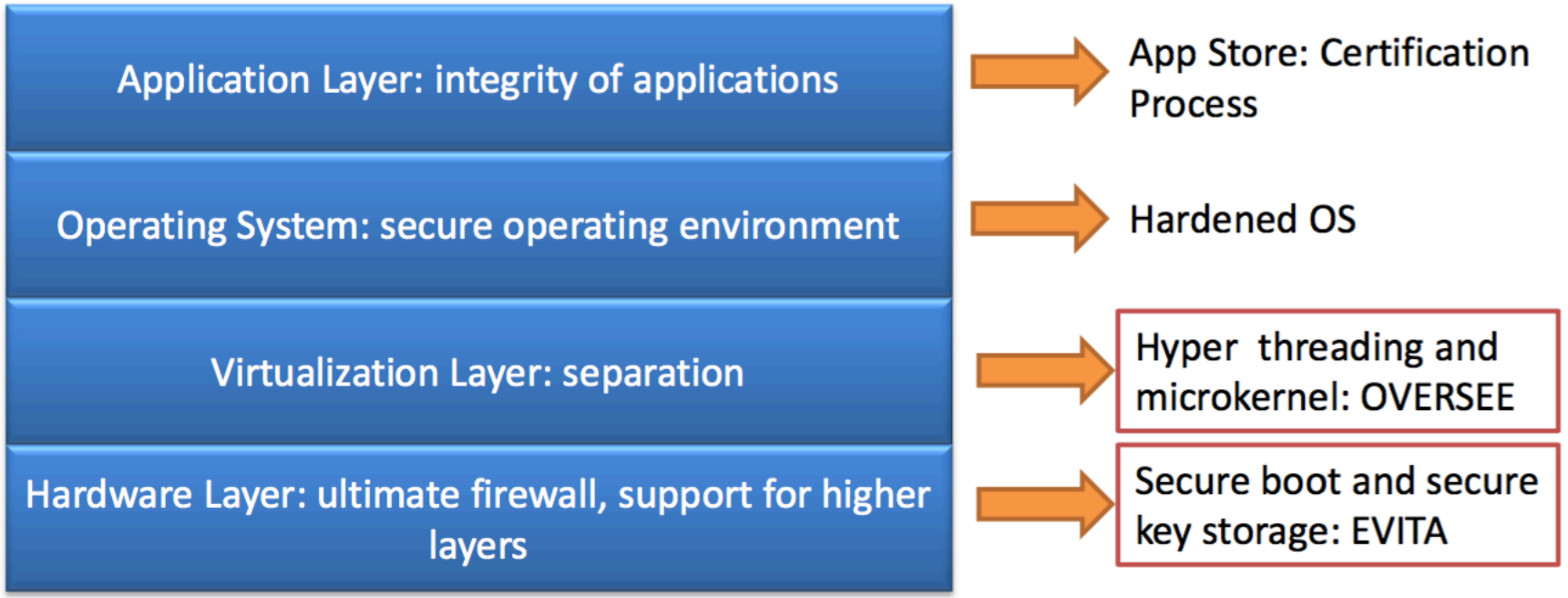



Illustration due to Andre Weimerskirch, Escrypt / ETAS / Bosch

# Evita architecture

- A starting point for hardware security
- Not widely referenced by OEMs
- https://www.evita-project.org/

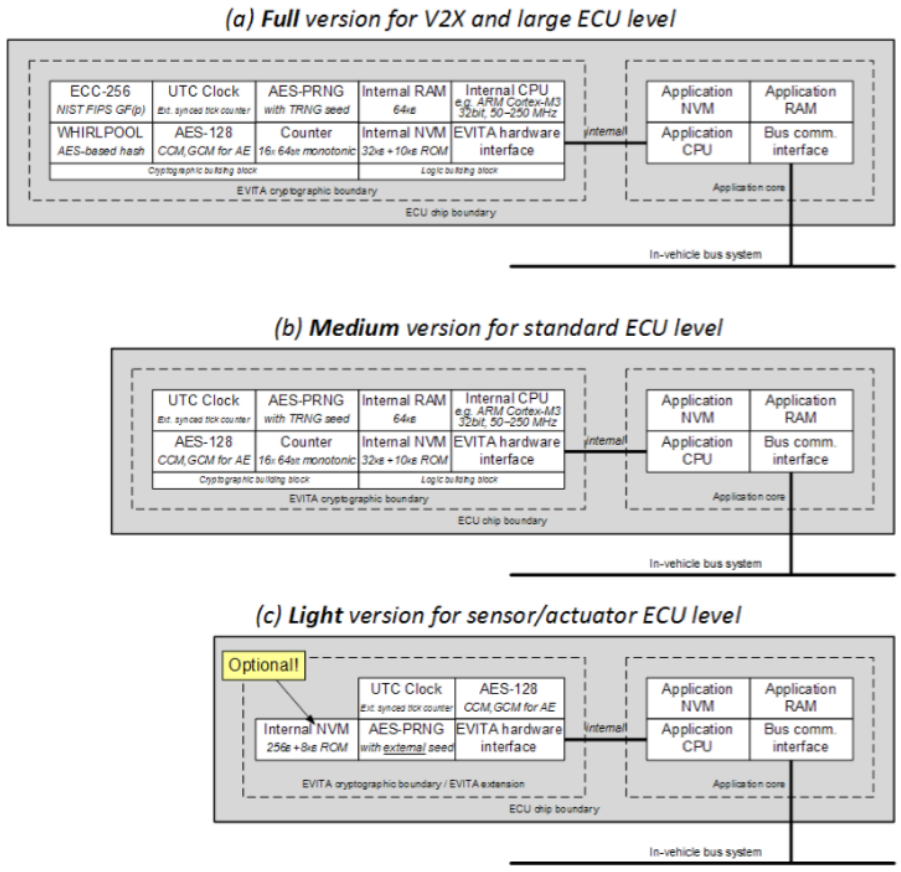



Illustration due to Andre Weimerskirch, Escrypt / ETAS / Bosch

# Other approaches

- Proposed mandate
  - https://www.federalregister.gov/documents/2017/01/12/2016-31059/federal-motor-vehicle-safety-standards-v2v-communications
  - FIPS 140 level 3 hardware protection for keys
  - Does not address other platform security issues
- SAE J2745/2
  - Specifies minimum performance requirements for BSM including security
  - FIPS 140 level 2 hardware protection for keys
- Car-2-Car Communications Consortium: Protection profile
  - Not yet public
  - Similar but not identical to the Pilot Deployments document
  - Separates into HSM and V2X Box
- Autosar
  - https://www.autosar.org/fileadmin/files/presentations/2016_09_06_EUROFORUM_Automotive_Software_Development.pdf
  - Industry-wide initiative originating in Germany to design architecture for application platforms
  - Includes security functionality but no platform requirements yet
- SAE J3101
  - Ongoing standardization work in SAE to develop hardware security requirements
  - Originally developed separately from V2X requirements
    - Informal harmonization process is underway
  - Considering access control, secure boot, etc.

# Certification

- US has self-certification regime
- Significant concerns about cost of formal certification
  - Cost-sensitive industry
  - Many different ECUs
  - Certification process sometimes difficult to reconcile with software updates
- Final outcome unclear at this time

- https://one.nhtsa.gov/cars/testing/comply/Mission/1_ovsc_1.html

- It is the responsibility of a manufacturer of vehicles and/or items of motor vehicle equipment to certify that each motor vehicle and/or equipment item is in full compliance with the minimum performance requirements of all applicable Federal Motor Vehicle Safety Standards (FMVSSs). This is a self-certification process as opposed to the type approval process which is used in some other countries such as Japan. The NHTSA does notissue approval tags, stickers or labels for vehicles or equipment items before or after the first sale. In order to provide certification, the manufacturer takes whatever actions it deems appropriate. This usually means laboratory testing in accordance with the FMVSS or conducting other studies or analyses (due care process) to ensure that its products fully comply.

- A compliance testing program has been in place since 1968. All of the 44 testable FMVSSs are included in a compliance test program over a period of 5 years with vehicle inspections conducted for the remaining 7 non-testable FMVSSs. A FMVSS self-certification program exists in the United States. The NHTSA does not certify that vehicles or items of motor vehicle equipment meet the requirements of various FMVSSs or issue "approval" stickers, labels, certificates, etc. Each year the OVSC randomly selects vehicles and items of motor vehicle equipment for compliance testing by approximately 21 independent testing laboratories under contract with the OVSC to verify that the manufacturer's certification is valid. The OVSC compliance testing program is a strong incentive for manufacturers of vehicles and/or items of motor vehicle equipment to institute and maintain a strong quality control/product surveillance program.

# Thank you!

Questions?

# Questions / Discussion