Vehicle Key Management – Status of Standardization
Agenda

- **Importance of cryptographic material**
  - Vehicle key management ≠ key storage
  - Challenges for standardization
  - Example: Initial keying at OEM for SecOC
  - Status of standardization
  - Summary
Cryptographic keys are the foundation for technical security mechanisms

- For security reasons different keys are used for different security-related use cases, e.g.
  - Secure flashing of ECUs (a.k.a. code signing, secure reprogramming)
  - Secure boot of ECUs
  - Diagnostic access control
  - Secured communication between the ECUs of a vehicle (e.g. via SECOC)
  - Secure communication from the ECU to external services (e.g. via TLS)
  - SW update over the air (SOTA)
  - Remote feature activation
  - Component theft protection
  - Immobilizer
  - Mobile online services
  - ...

- The affected ECUs require a considerable number of cryptographic keys
Agenda

Importance of cryptographic material

- **Vehicle key management != key storage**
  - Challenges for standardization
  - Example: Initial keying at OEM for SecOC
  - Status of standardization
  - Summary
Vehicle key management != key storage

Vehicle key management in a layered security concept

Security concepts

- Secure communication to services outside the vehicle
- Intrusion detection mechanisms
- Diagnostic policy manager
- **Vehicle key management**
- Security event memory
- Authentic synchronized time
- Authenticity of messages
- Integrity and freshness of messages
- Confidentiality of messages
- **Key storage**
- Secure boot and secure flash
- Crypto library
- HW trust anchor (HTA)
Key storage

- **Goal:**
  - Securely store cryptographic keys

- **Basic functions and key aspects:**
  - Take a cryptographic key from the application
  - Securely store it in NVM or hardware trust anchor of ECU

- Supported by the **crypto stack** (CSM, CRYIF, CRYPTO)
  - Configuration of key structures via key elements
Vehicle key management in the AUTOSAR architecture

Goal:
- Simplifies typical and common key lifecycle management tasks

Basic functions:
- Receives new cryptographic material (keys, certificates) via diagnostic routines
- Verifies authenticity, integrity and freshness of cryptographic material
- Provides callouts to integrate with business logic for different typical key lifecycle phases (production, initialization, update, repair, replacement)
- Supports on board derivation of new keys
- Supports secure distribution of shared secret keys
- Logs security events to security event memory (SEM)
Agenda

- Importance of cryptographic material
- Vehicle key management != key storage

- **Challenges for standardization**
  - Example: Initial keying at OEM for SecOC
  - Status of standardization
  - Summary
Challenges for standardization

Key lifecycle phases

Production of the ECU
- Insertion of initial keys

End of line programming
- Replacement of initial keys by OEM specific master keys
- Insertion of additional keys
- On board derivation of further keys
- Secure distribution of keys in the vehicle network

Aftersales
- Keys can be replaced if they have become compromised
- Keys can be renewed after a certain time to improve security
- Additional keys can be inserted for new use cases
- Replaced ECUs can get appropriate keys to participate in secure vehicle communication

Keys can be replaced if they have become compromised
 Keys can be renewed after a certain time to improve security
 Additional keys can be inserted for new use cases
 Replaced ECUs can get appropriate keys to participate in secure vehicle communication
Challenges for standardization

Variation points for technical solution

- Development-, production-, after sales **processes @ Tier1 & OEM**
- Existing **backend** key management processes and IT infrastructure (e.g. PKI)
- Security goals (based on assumptions about the security of the development / production / service environment)
- Performance goals (based on end of line programming requirements)
- Vehicle **security architecture** / vehicle key management paradigm (centralized / decentralized)

- Current situation: Vector provides **proprietary vehicle key management solutions** to support a large number of different OEMs

- Goal for standardization: find right level of abstraction
  - to provide added value compared to proprietary solutions
  - Support known OEM specifics via configuration and extension interfaces
Agenda

Importance of cryptographic material
Vehicle key management != key storage
Challenges for standardization

- Example: Initial keying at OEM for SecOC

Status of standardization
Summary
Scenario 1: Off-board (backend) key generation

Diagnostic Tester

- Diagnostic Tester provides backend generated keys to each node
- Key managers are limited to validating backend generated SECOC keys via
  - SHE1.1 key update protocol or
  - OEM specific key update containers
Scenario 2: On-board key derivation with coordinator

Example: Initial keying at OEM for SecOC

Diagnostic Tester

- Diagnostic Tester triggers SecOC keying

KEYM (Server)

- On-board KEYM server creates and stores vehicle specific secret
- On-board KEYM server coordinates secure distribution of secret to clients (e.g. via Diffie-Hellman)

KEYM (Clients)

- KEYM clients use secret and key derivation function to securely derive SecOC keys
Scenario 3: On-board key generation without coordinator

- Diagnostic Tester triggers SecOC keying
- No dedicated KEYM server which coordinates key negotiation (completely decentralized)
- Group of ECUs participates in negotiation of shared secret (e.g. via Burmester-Desmedt)
- Participating nodes derive SecOC keys from shared secret

Example: Initial keying at OEM for SecOC
Agenda

- Importance of cryptographic material
- Vehicle key management != key storage
- Challenges for standardization
- Example: Initial keying at OEM for SecOC

- Status of standardization

Summary
Vehicle key management in a layered security concept

Security Concepts

- Secure communication to services outside the vehicle (TLS)
- Intrusion detection mechanisms
- Diagnostic policy manager
- **Vehicle key management**
- Security event memory
- Authentic synchronized time
- Authenticity of messages
- Integrity and freshness of messages
- Confidentiality of messages
- **Key storage**
- Secure boot and secure flash
- Crypto library
- HW trust anchor (HTA)

Standard

- AUTOSAR4.4
- Security Extensions AUTOSAR4.4
- SecOC
- CSM / CRYIF / CRYPTO
- SHE, HSM, TPM, TEE,...
AUTOSAR 4.4 Security Extensions

- C1: Security Event Memory
- **C2: Vehicle Key Management / Key Distribution**
  - C3: Secure Boot Status (dropped)
  - C4: Authentic Synchronized Time
  - C5: Dynamic Rights Management for Diagnostic Access
  - C6: Improved Certificate Handling (integrated in C2)
  - C7: Abstract pre-definition of Crypto Items in System Template (improves AUTOSAR tooling support for security)
Status of standardization

Timeline 2018

**Timeline Diagram**

- **Conc. Review**
  - MS2
  - Call for Review MS3a
  - MS3a

- **Validation**
  - MS3b

- **Incorporation**
  - MS4
  - Release 4.4

**MS2 Criteria**
- Features, use cases and technical approach are agreed among all stakeholders
- Solution is described on requirement level
- Impact analysis completed

**MS3a Criteria**
- Technical solution is detailed out on specification level (C&P readiness)
- Validator, doc owner and WP review findings are considered in the concept

**MS3b Criteria**
- Technical solution is validated
- Validation results are considered in Concept
- Concept is ready for incorporation

**MS4 Criteria**
- Concept is incorporated
- Successful incorporation is confirmed by Lead-WPs and concept owner
Importance of cryptographic material
Vehicle key management != key storage
Challenges for standardization
Example: Initial keying at OEM for SecOC
Status of standardization

Summary
Summary

Important points

- **Vehicle key management != key storage**
- Secure management of cryptographic keys in all lifecycle phases adds an important layer of security
- **Standardization** has a lot of potential for cost saving but is challenging due to OEM specifics
- Vector provides OEM specific key management implementations for a number of OEMs
- **AUTOSAR 4.4 Security Extensions** provide KEYM module as a framework for vehicle key management

Outlook:
- Security Extensions will be continued in **AUTOSAR 4.5**
For more information about Vector and our products please visit

www.vector.com

Author:

Dr. Eduard Metzker

Vector Informatik GmbH