MICROSAR Cyber Security

4th VECTOR INDIA CONFERENCE 2019
Cybersecurity Realization in Automotive Systems

Basics of Cryptography
Use Cases
Secure Onboard Communication
Cybersecurity Architecture
Secured Communication Configuration
Configuration of Security Manager in CANoe
Vector Company Overview
Goals of Cyber Security

- **Authenticity:**
  Allows to determine whether someone or something is, in fact, who or what it is declared to be.

- **Integrity:**
  Allows to assure the accuracy and reliability of information and allows to prevent or detect unauthorized modification.

- **Confidentiality:**
  Ensures that the necessary level of secrecy is enforced and prevents unauthorized disclosure of information.

- **Availability:**
  Availability protection ensured reliability and timely access to data and resources to authorized individuals.
Cybersecurity Realization in Automotive Systems

Layered Security Concept – The logical view

Associated Security Concepts

- Secure communication to services outside the vehicle
  - Secure Off Board Com.
- Intrusion detection mechanisms
- Firewalls
- Key Infrastructure / Vehicle PKI
- Authenticity of messages
- Integrity and freshness of messages
- Confidentiality of messages
- Key storage
- Secure boot and secure flash
- Crypto library
- HW trust anchor (HTA)
  - E.g.: HSM (Hardware Security Modules)
Cybersecurity Realization in Automotive Systems

Security Mechanisms in Vehicle Architecture

- Firewall
- Key Infrastructure
- Crypto Primitives
- Monitoring / Logging
- Hypervisor
- Intrusion Detection / Prevention
- Secure On Board Com.
- Secure Flash/Boot
- Secure Off Board Com.
- Download Manager

Diagram showing various components and their connections in a vehicle architecture, including Powertrain DC, Chassis DC, Body DC, Central Gateway, Connectivity Gateway, Instrument Cluster, Head Unit, DSRC, 4G LTE, CU, Laptop, Tablet, Smartphone, Smart Charging, ADAS DC, Diagnostic Interface, and Central Gateway.
Cybersecurity Realization in Automotive Systems

Cyber Security Lifecycle

- Asset Definition
- Threat and Risk Assessment
- Derivation of Security Goals
- Security Architecture Design & Analysis
- Security Mechanisms Design & Analysis
- Secure Implementation of Nominal Function and Security Mechanisms
- Incident Management and Response
- Security Validation
- Penetration Testing
- Fuzz Testing
- Functional Security Testing

"Cyber Security does not start or end with cryptography: Similar to Safety, Security needs to be an integrated part of the development process."
Cybersecurity Realization in Automotive Systems

Safety & Security - Interdependency

- **Dependable Automotive Systems**
  - Protection against risks from technical failures

- **Functional Safety**
  - Protection against risks from technical failures

- **Cyber Security**
  - Protection against risks from malicious actions

- **Accident**
- **Attack**
Secure Hardware Extension (SHE)

Hardware Security Module (HSM)
Agenda

Cybersecurity Realization in Automotive Systems

- Basics of Cryptography
  - Use Cases
  - Secure Onboard Communication
  - Cybersecurity Architecture
  - Secured Communication Configuration
  - Configuration of Security Manager in CANoe
  - Vector Company Overview
Hash Functions

Symmetric Cryptography

Message Authentication Code (MAC)

Hashed Message-Authentication-Code (HMAC):
Uses a hash function and a secret (symmetric) key.

Cipher-based Message-Authentication-Code (CMAC):
Use a block cipher (e.g. AES) and the key
Asymmetric cryptosystems are based on a **key pair** owned by a party.

- The key pair consists of a **public key** and a **private key**.
  - Public key can be known by the public.
  - A Private Key shall never be shared.

- Messages can be
  - encrypted with the public key and the cipher function \( M' = E(M, K_{pub}) \).
  - decrypted with the decryption function and the private key \( M = D(M', K_{priv}) \).

- There is no way the private key can be calculated.
## Asymmetric vs. Symmetric Cryptography

<table>
<thead>
<tr>
<th></th>
<th>Symmetric</th>
<th>Asymmetric</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Key</strong></td>
<td>One single secret key</td>
<td>Key pair (One public and one private key)</td>
</tr>
<tr>
<td><strong>Key length</strong></td>
<td>Relatively short</td>
<td>Relatively long</td>
</tr>
<tr>
<td><strong>Data throughput rate</strong></td>
<td>Very fast especially with HW support</td>
<td>Always much slower than symmetric</td>
</tr>
<tr>
<td><strong>Secrecy</strong></td>
<td>Secret has to be shared with each involved communication partner</td>
<td>Secret (private key) is kept to its owner, public key is shared</td>
</tr>
<tr>
<td><strong>Key management</strong></td>
<td>Complexity grows with number of involved communication partners</td>
<td>Complexity is linear with number of number of communication partners</td>
</tr>
</tbody>
</table>
Digital Signature - Principles

Goals: authenticity and integrity of data

- The originator needs to have a generated Public-Private key pair.

- The originator uses the following process to generate a signature
  - Calculate Hash Value for the Data
  - Resulting hash value is encrypted using the private key of the originator
  - Signature is appended to the data and sent over to the user

- The user uses the following signature verification process
  - Decrypts the hashed value with the originator's public key
  - Calculates the hash of the data
  - Compares the hashed data and decrypted hashed value
  - If they are the same, then the authenticity and integrity of data can be assured

- Standards for Digital Signatures are set forth in FIPS 186\(^1\)
Certificates

- ... used to identify communication partners
- ... contains signed personal characteristics of the owner (name, place, ...)
- ... can be restricted to a limited period of time, service and location.
- ... can be provided by a "Trusted Authority" (TA) or "Certificate Authority" (CA)), which is the trust anchor and has built the signature within the certificate.
Basics of Cryptography

Certificates - Example for Automotive PKI

OEM Root Certificate,
Self signed

\[ \text{Sign(Root CA)} = \text{SignAlgo( tbsCert (Root CA), K(Priv RootCA) )} \]

Sign(PlatformCA) = SignAlgo( tbsCert (PlatformCA), K(Priv RootCA) )

Sign(CarCA) = SignAlgo( tbsCert (CarCA), K(Priv PlatformCA) )

Sign(TesterCA) = SignAlgo( tbsCert (TesterCA), K(Priv RootCA) )

Sign(CarCA) = SignAlgo( tbsCert (CarCA), K(Priv TesterCA) )
Application data can be reliably exchanged on an IP based Network by the Transport Control Protocol (TCP), but, TCP does neither ensure privacy, nor integrity of the exchanged data.

To protect a TCP connection, the Transport Layer Security Protocol (TLS 1.2, RFC5246) can be used.

- Privacy is ensured by Symmetric Cryptography (e.g. AES).
- Data integrity is ensured by a Hash-based Message Authentication Code (H-MAC).
- The encryption and H-MAC computations are using temporary secret keys bound to TLS 1.2 session.
- Authenticity of the server is always ensured by a Digital Certificate (X.509v3).
- Optionally the client can be authenticated, too.

```
TLS_ECHDE_ECDSA_WITH_AES_128_CBC_SHA256
```

- Key Exchange Algorithm
- Digital Signature Algorithm
- Symmetric Encryption Algorithm
- Hash Algorithm
Agenda

Cybersecurity Realization in Automotive Systems
Basics of Cryptography

► **Use Cases**

Secure Onboard Communication
Cybersecurity Architecture
Secured Communication Configuration
Configuration of Security Manager in CANoe
Vector Company Overview
Example: HMAC for flash programming

Hash function

$k_{secret}$: Secret key

HMAC

Keyed-Hash Message Authentication Code (FIPS PUB 198)

Flashfile

Hash

HMAC

HMAC

Flashfile

Calculated

HMAC

Verify

Flash data

Boot

$k_{secret}$

Transferred

Transferred

Flash-download

© 2019. Vector Informatik India Private Limited. All rights reserved. Any distribution or copying is subject to prior written approval by Vector. V0.2 | 2019-07-25
Example: Digital Signature for flash programming
Use Cases

Usage Example: Communication between tester and vehicle

- Certificates are used for an authentic communication
- They can be used for the following purposes:
  - Tester to car:
    - Usage of $\text{Cert}_{\text{Tester}}$ and $\text{Cert}_{\text{Car}}$ for communication.
    - Car requests the revocation list of tester serial number from backend.
    - Option: If revocation list cannot verified at the moment, only restricted operations by the tester are allowed.

Example:
Secure Onboard Communication

Goals

integrity

ECU 1  ECU 2

authenticity

ECU 1  ECU 2
Secure Onboard Communication

AUTOSAR SecOC

MAC = Message Authentication Code
Secure Onboard Communication

AUTOSAR SecOC

ECU 1

- Key
- Data
- Freshness value
- MAC

BUS

ECU 2

- Data
- Freshness value
- MAC

Authenticated message
Secure Onboard Communication

AUTOSAR SecOC

ECU 1

ECU 2

BUS

Authenticated message

Data

Freshness value

MAC generator

MAC
Secure Onboard Communication

AUTOSAR SecOC

ECU 1

BUS

authenticated message

MAC

ECU 2

MAC

© 2019. Vector Informatik India Private Limited. All rights reserved. Any distribution or copying is subject to prior written approval by Vector. V0.2 | 2019-07-25
Freshness Value Manager (FvM)

- AUTOSAR does not specify the calculation/synchronization of the freshness value.
- Instead a generic callout to a Freshness Value Manager (FVM) component is provided
- FVM specification is left to the OEM
Secure Onboard Communication

Freshness: Replay of authenticated messages
Secure Onboard Communication

Freshness: Replay of authenticated messages

ECU 1

ECU 2
Secure Onboard Communication

Freshness: Replay of authenticated messages
Secure Onboard Communication

Concepts of Freshness

- **Message counter based freshness (MCBF)**

- **Trip counter based freshness (TCBF)**

- **Time stamps**

- **Hybrid system: time stamp & message counter**
Agenda

Cybersecurity Realization in Automotive Systems
Basics of Cryptography
Use Cases
Secure Onboard Communication
  ▶ **Cybersecurity Architecture**
    Secured Communication Configuration
    Configuration of Security Manager in CANoe
    Vector Company Overview
Cybersecurity Architecture

MICROSAR Cyber Security Solution

- **Crypto Service Manager – CSM**
  - Services can be called by SWCs
  - Configuration of cryptographic key elements
  - Configuration of cryptographic services

- **Crypto Interface – CRYIF**
  - Supports dispatching of security jobs to HW or SW crypto drivers

- **Crypto Driver – Crypto (SW/HW)**
  - Implementation of cryptographic functions
  - Crypto (SW): Usage of SW-libraries
  - Crypto (HW): Usage of resources and capabilities of HW-Trust Anchors (SHE, HSM, TPM,...)
Cybersecurity Architecture

Interaction of the AUTOSAR Application, BootLoader with Vector HSM

Host Core of ECU

HSM Core of ECU

Interaction of the AUTOSAR Application, BootLoader with Vector HSM

Host Core of ECU

- SWC/Application
  - RTE
    - CSM
    - COM
    - CRYIF
    - MCAL
    - Cryptos (HW)

- FBL Application
  - Secure Boot
  - Secure Reprogramming
  - FBL Crypto (HW)

HSM Core of ECU

- HSM Application
  - Job Dispatcher
    - Crypto Primitive Manager
      - Cry KeyM
      - Cry TRNG
      - Cry AES
      - Cry ...
    - Vector Crypto Lib

- FBL Application
  - Secure Boot
  - Secure Reprogramming
  - FBL Crypto (HW)

- HSM Application
  - Mode Manager
  - Secure Boot Manager
  - Custom
Cybersecurity Architecture

CSM Job Handling:

- Priority
- Synchronous job
- Asynchronous job
- CSM keys
- CSM queues
- CSM primitives
Cybersecurity Architecture

Scheduling of Asynchronous Jobs

- Upon calling CSM service function, the job is added into the queue
- Scheduling and execution in CSM mainfunction

- Sort job wrt. their priorities
- Synchronously call Driver Object to process the job with highest priority
- After job finishes, call callback function & remove the job from the queue
Cybersecurity Architecture

Crypto Driver objects & CSM Keys

▶ Crypto Driver:

![Diagram showing cryptographic capabilities and driver objects]

Primitives

Cryptographic capabilities

Driver Obj1

Driver Obj1

Driver Obj1
Agenda

Cybersecurity Realization in Automotive Systems
Basics of Cryptography
Use Cases
Secure Onboard Communication
Cybersecurity Architecture

- **Secured Communication Configuration**
  Configuration of Security Manager in CANoe
  Vector Company Overview
Secured Communication Configuration

Flow of Secured Receive Frame:

- **CAN Bus**
  - data freshness value MAC authenticated message

- **CAN**
  - data FV MAC

- **CANIF**
  - data FV MAC

- **PDUR**
  - data FV MAC

- **COM**
  - data

- **SecOC**
  - MAC

- **CSM**
  - CryIF

- **FVM**
  - FV
Secured Communication Configuration

SecOC Configuration

User can configure Call back function, which will be called based on each individual PDU propagation Mode

User can select the Freshness calculation function
Secured Communication Configuration

AES primitive configuration for MAC calculation:

SecOC ID has been considered from CAN ID

Here user can decide for verification result propagation / Call backs
Secured Communication Configuration

Key Configuration:

This is the Symmetric Key, which should be the same in both sides.
In this example, same key should be configured in the Canoe Security profile.
Configuration of Security Manager in CANoe

**Testing: Enabling Analysis and Test of Secured Networks**

- **CANoe Fuzz Testing**
  - Available: for selected Pilot Customers

- **Security Manager**
  - Available: (OEM specific)
## Application Areas and Product Examples

### Development of Distributed Systems
- PREEvision

### Embedded Software and Systems
- MICROSAR, CANbedded, VC ECU, Customer Projects

### Testing
- CANoe, CANalyzer, vTESTstudio, VT System, Logger, VectorCAST

### Diagnostics
- CANdelaStudio, Indigo, vFlash, CANoe.DiVa

### ECU Calibration
- CANape, VX1000, vCDM, vADASdeveloper, ASAP2 Tool-Set

### Measurement Technology
- vMeasure exp, vSignalyzer, vMDM, MDF4 Lib, Analog Measurement Devices

### Consulting
- Consulting Services, Engineering Services
More information!
Visit our Website for:
> News
> Products
> Demo Software
> Support
> Workshops
> Contact Addresses

www.vector.com
Author:
Subrahmanyam Namdikam
Vector Informatik India Limited