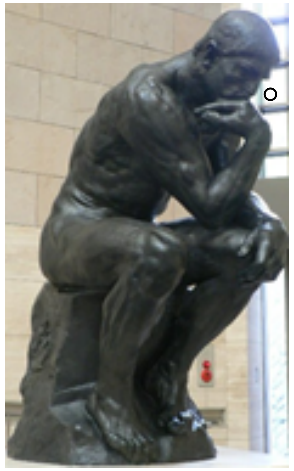
Automotive Security

～Using Secure Element～

KDDI
Keisuke Takemori Ph.d

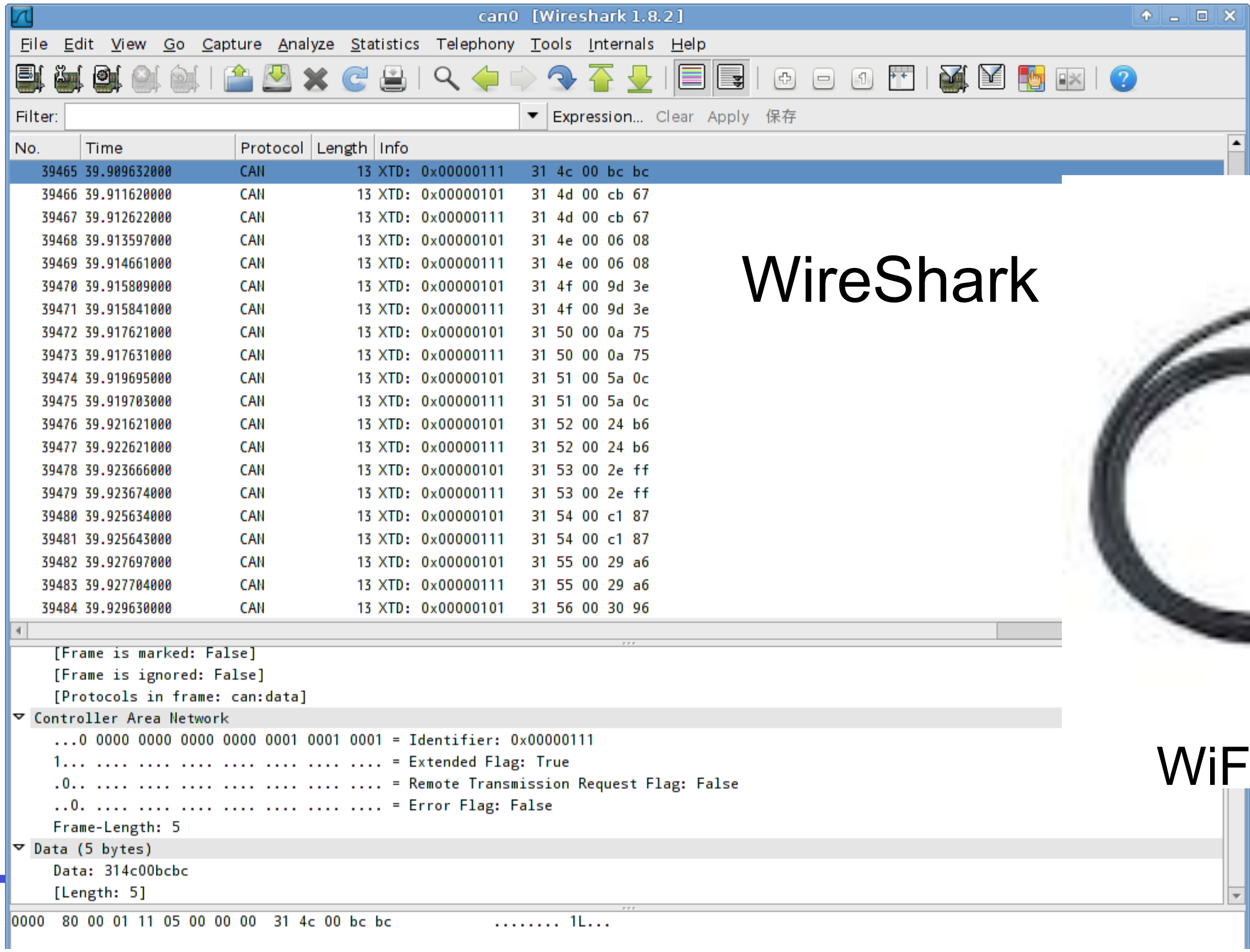
A secure element will be a trust anchor in a vehicle.



- **Vehicle Incidents**
- **(1) Key Management → CAN+MAC**
- **(2) Remote Reprogramming**
- **Basis Security Techniques**
- **Conclusion**

You can capture the CAN packets.

- WireShark + WiFi/OBD-II Connecter
  - The WiFi device is connected to the OBD-II port.
    The PC that installs the WireShark can capture the CAN packets.
  - The WireShark has a replay mode that can send any type of CAN packet.



WireShark

WiFi/OBD-II Connecter

Local Attack） "Fake ECU Injection" 2010-2013

\# Local Insertion Attack

→ The PC is inserted in the controller area network (CAN).

→ Powertrain, steering, and breaking systems were hijacked.

DEFCON 2013 http://drive-love.jp/drivpedia/2013/08/post-19.html

Near Field Attack) "Man-in-the-Middle" July 2015

Attack Scenario

http://samy.pl/defcon2015/2015-defcon.pdf

Countermeasures

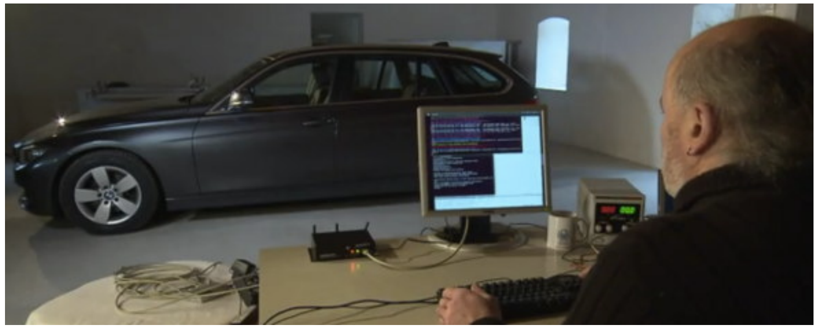
→ The application should verify the server certification when it is received.

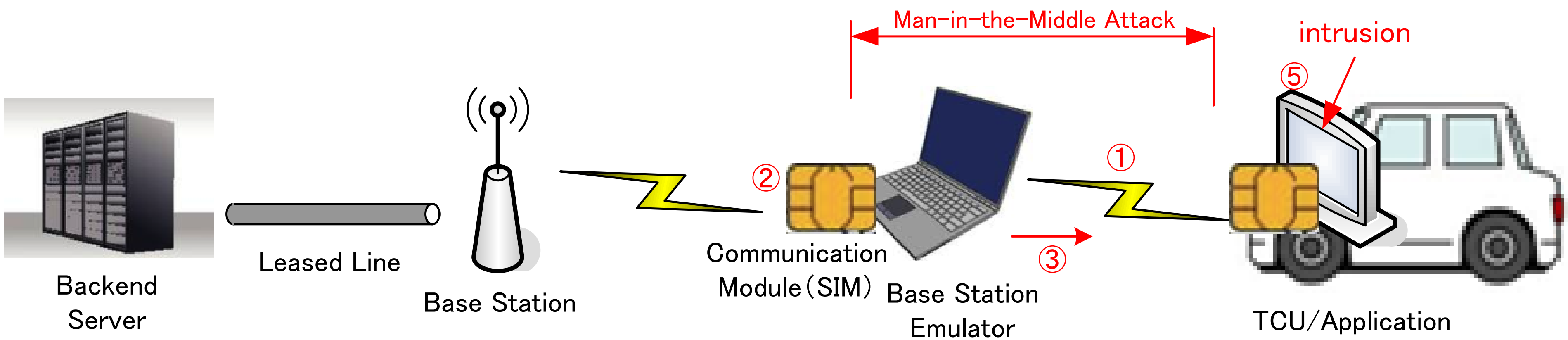
Remote Attack 1） "Man-in-the-Middle" Feb. 2015

Attack Scenario
http://www.sbdjapan.co.jp/bmw_connecteddrive_news/
http://m.heise.de/ct/artikel/Beemer-Open-Thyself-Security-vulnerabilities-in-BMW-s-ConnectedDrive-2540957.html

1) A telecom control unit (TCU) in a vehicle accesses the base station emulator.
2) As the GSM 2G/3G protocol includes vulnerabilities, the attacker cuts in the path.
3) As the commands are sent over the http, the attacker monitors them and injects fake commands.

Countermeasures
→ Communication path between the app and the backend server should be encrypted, e.g., https.





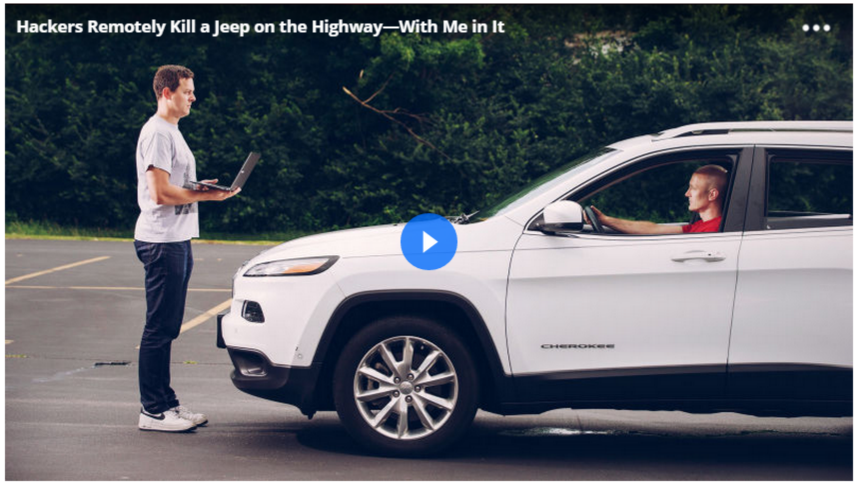
Remote Attack 2）“Intrusion” July 2015

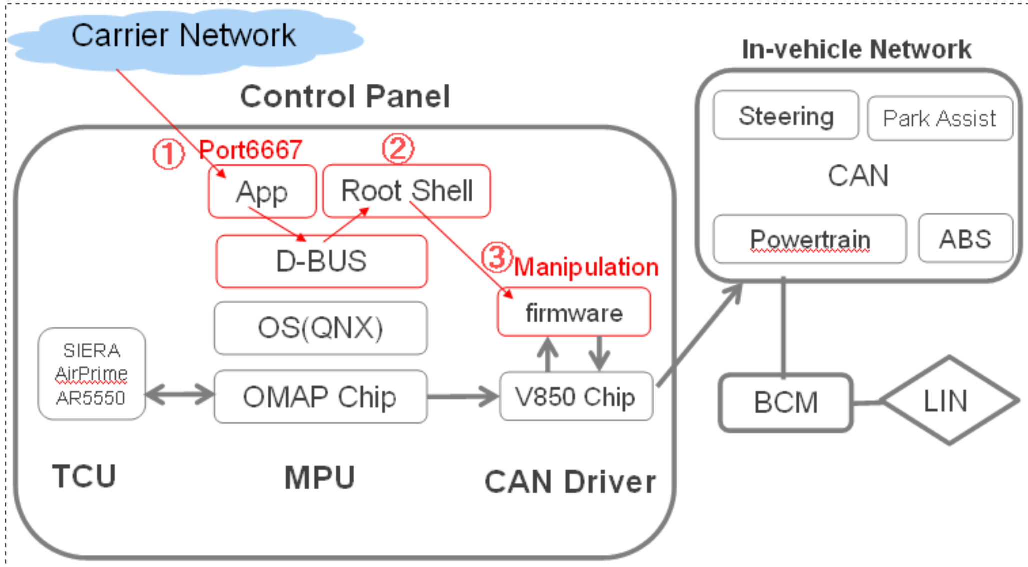
Attack Scenario
http://illmatics.com/Remote%20Car%20Hacking.pdf

1) A PC in the carrier network can access the control panel of a vehicle.
2) The root shell is cracked by the brute-force password attack.
3) The CAN driver is manipulated to read/write access permission.
→ The attacker sends malicious CAN packets from the remote site to the vehicle.

Countermeasures
→ The control panel should verify the sign of CAN F/W, when the F/W is updated.

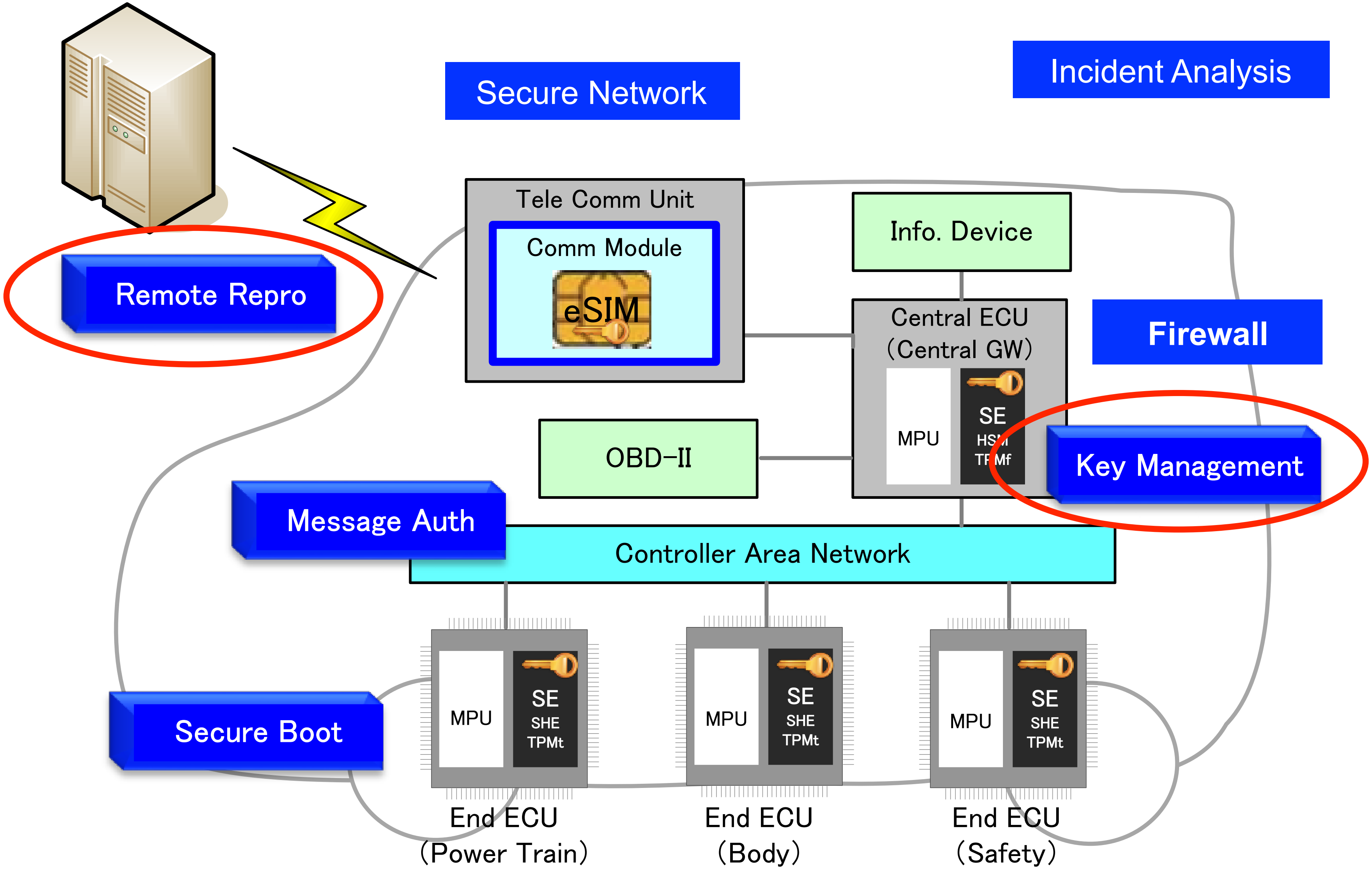
HACKERS REMOTELY KILL A JEEP ON THE HIGHWAY—WITH ME IN IT





http://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/

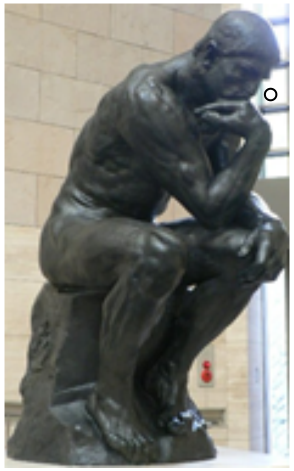
Security Countermeasures



Automotive Security

～Using Secure Element～

KDDI
Keisuke Takemori Ph.d



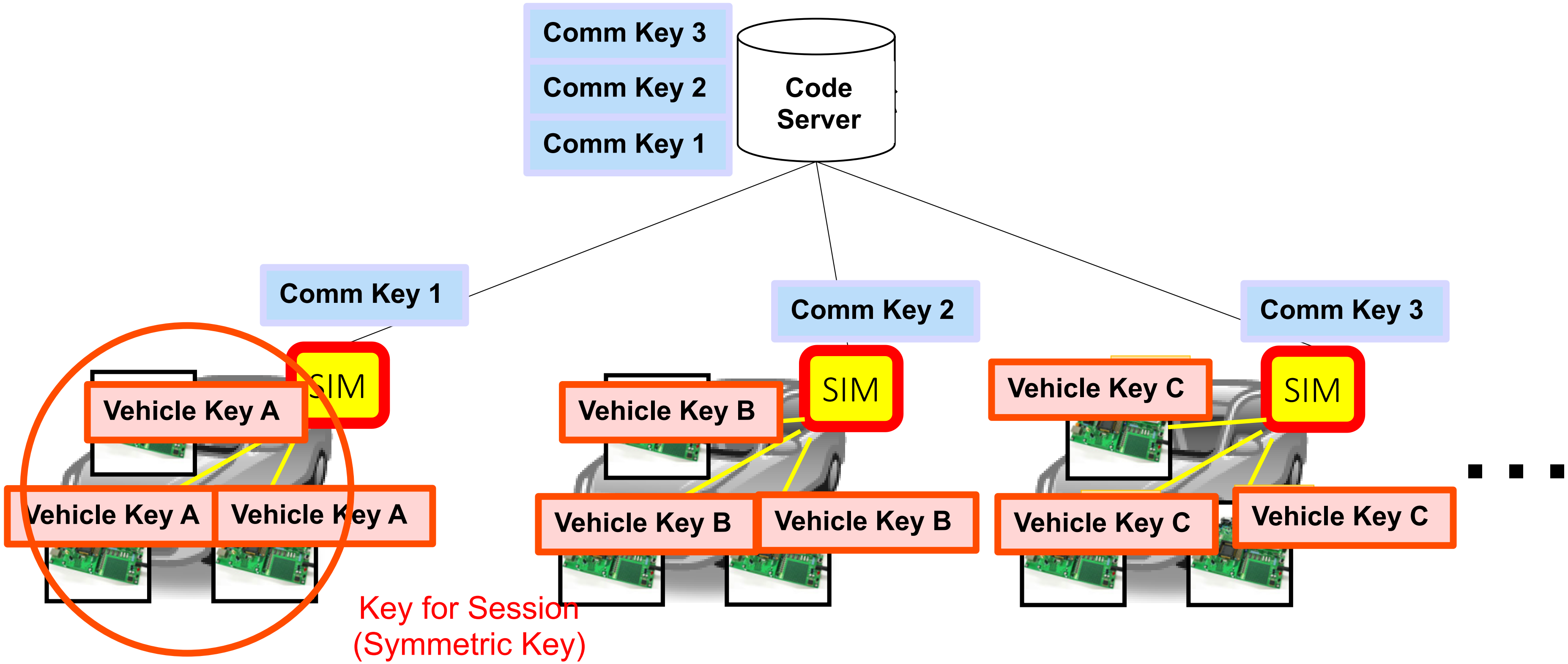
A secure element will be a trust anchor in a vehicle.

- Vehicle Incidents
- **(1) Key Management → CAN+MAC**
- (2) Remote Reprogramming
- Basis Security Techniques
- Conclusion

Cipher Key Management Policy

Cipher Key Management with Privacy Protection
→ From the viewpoint of privacy protection, third party should not manage the cipher keys of the in-vehicle network.

A cipher key of Vehicle A is different from cipher keys of the other vehicles.
→ Even if the cipher key in one of the vehicles is leaked, recall for the other vehicle is not needed.



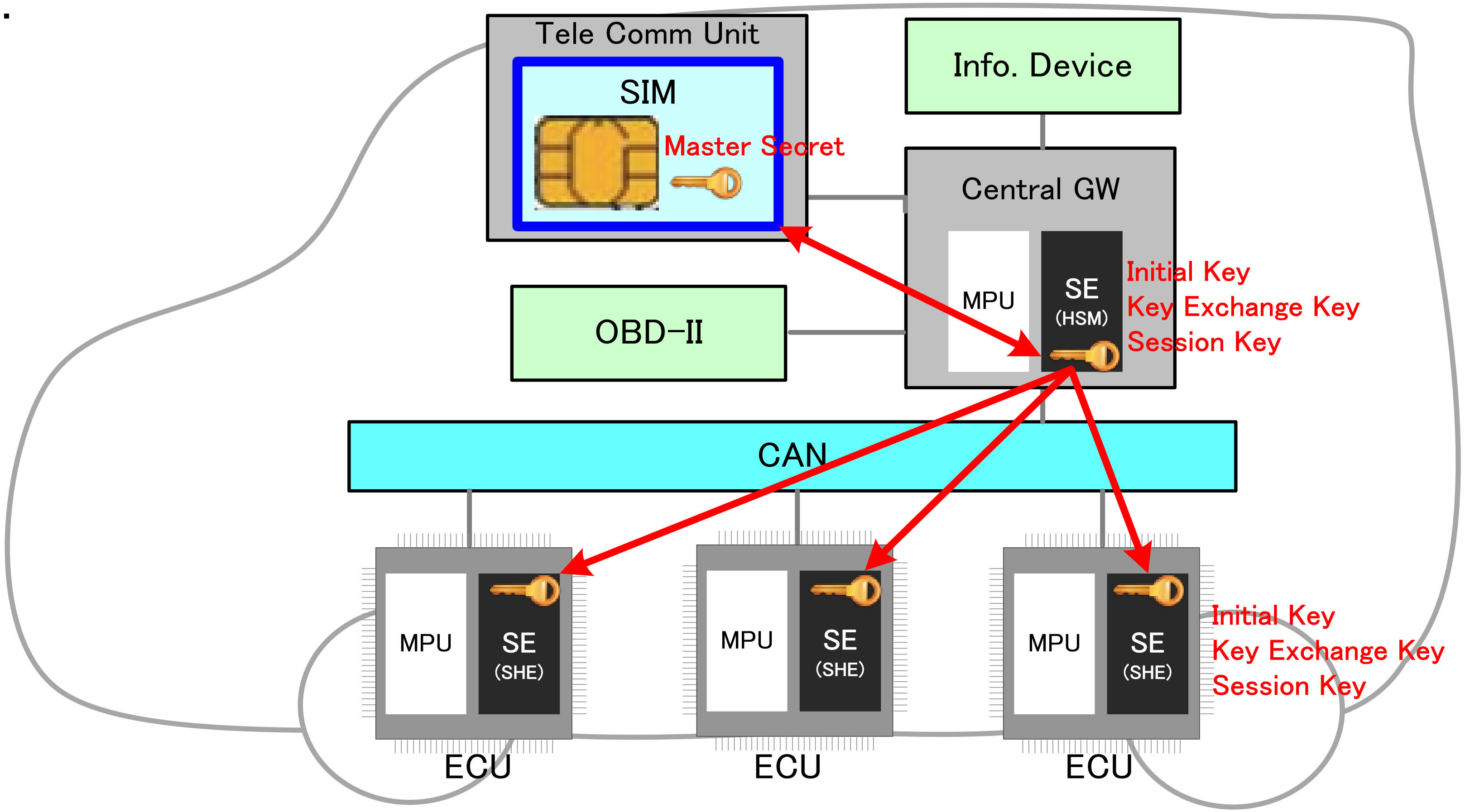
Example of the Key List

	Details	Symmetric or Asymmetric	Mark	Applied for
Master Secret	Seed of Initial Key	–	–	Inside Outside
Initial Key	Authentication Key of ECU	Symmetric	Ki	Inside Outside
Key Exchange Key	Exchange Key of Session Key	Symmetric	Kx	Inside
Session Key	MAC Generation Key	Symmetric	k	Inside
Root Certificate	Authentication Keys of Server and Client Certificates	Asymmetric	KRpub KRsec	Inside Outside
Server Certificate	Authentication Keys of ECU Code	Asymmetric	KSpub KSsec	Inside Outside
Client Certificate	Authentication Keys of ECU Code Update Status	Asymmetric	KCpub KCsec	Inside Outside
Boot MAC Key	CMAC Generation Key for Secure Boot	Symmetric	KB	Inside

How to Manage the In-vehicle Keys

→ A central GW generates both a key exchange key and a session key, and sends them to ECUs.

→ A SIM generates the initial key of each ECU using a master secret and ECU ID.

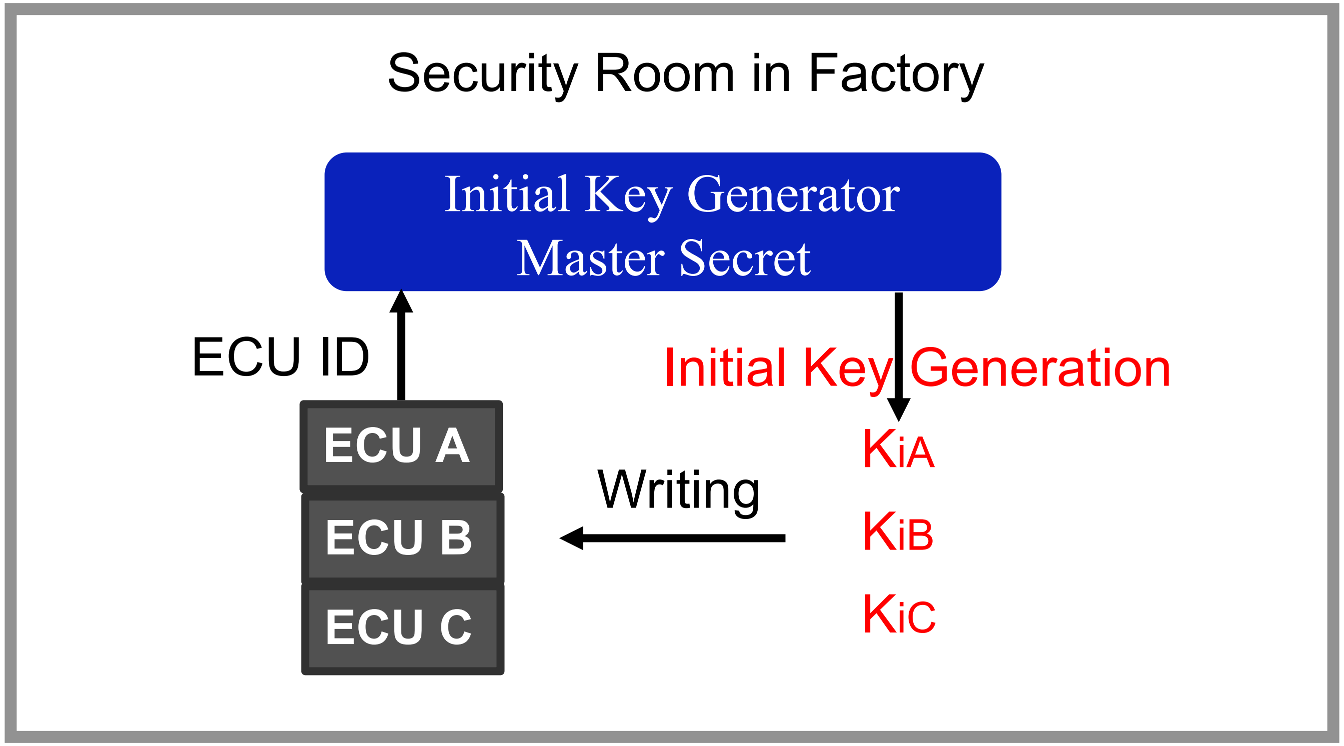


(1-1) Setting the Initial Key into the ECU

\# Generation of Initial Key by the ECU Supplier

→ The initial key is generated, and is written into the ECU.

Initial Key = Digest (ECU ID + Master Secret)



Note: The master key is issued to each ECU supplier.

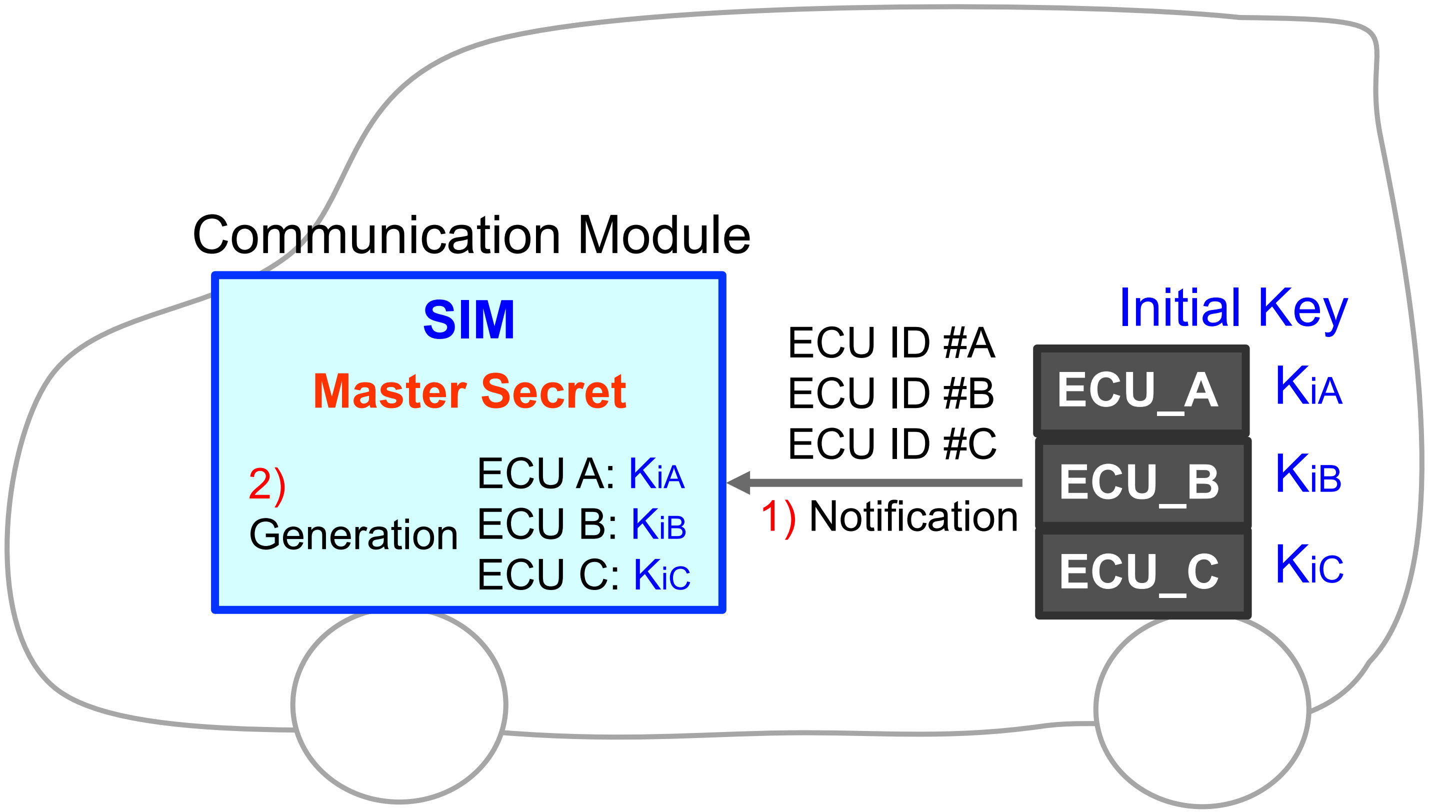
(1-2) Initial Key Sharing

\# At the first time of ignition in the OEM factory,

→ ECUs notify their IDs to the SIM, which manages the master key.

→ The SIM generates the initial keys.

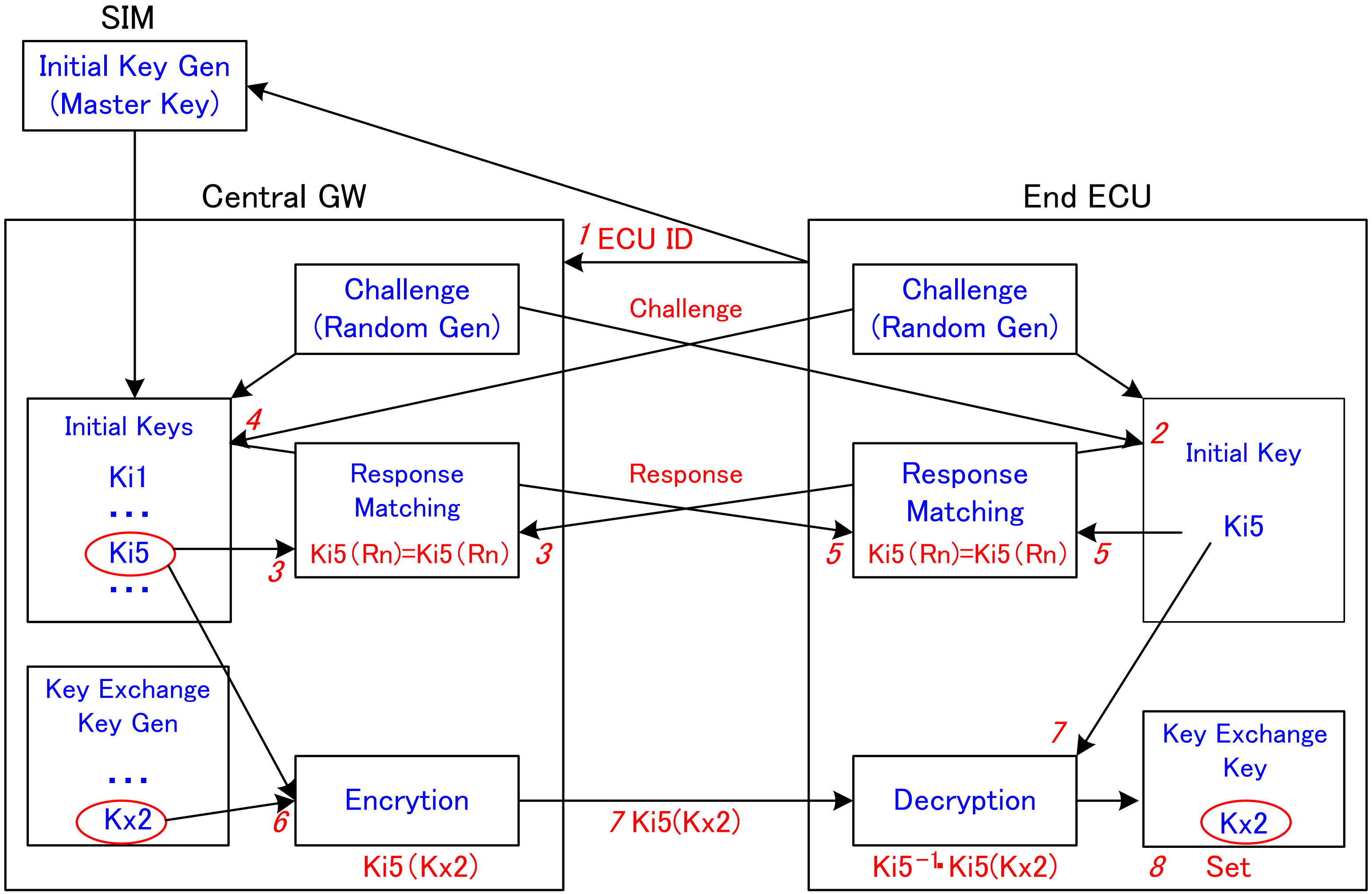
Initial Key = Digest (ECU ID + Master Secret)



Good Design) The initial keys are generated and are managed in the vehicle.

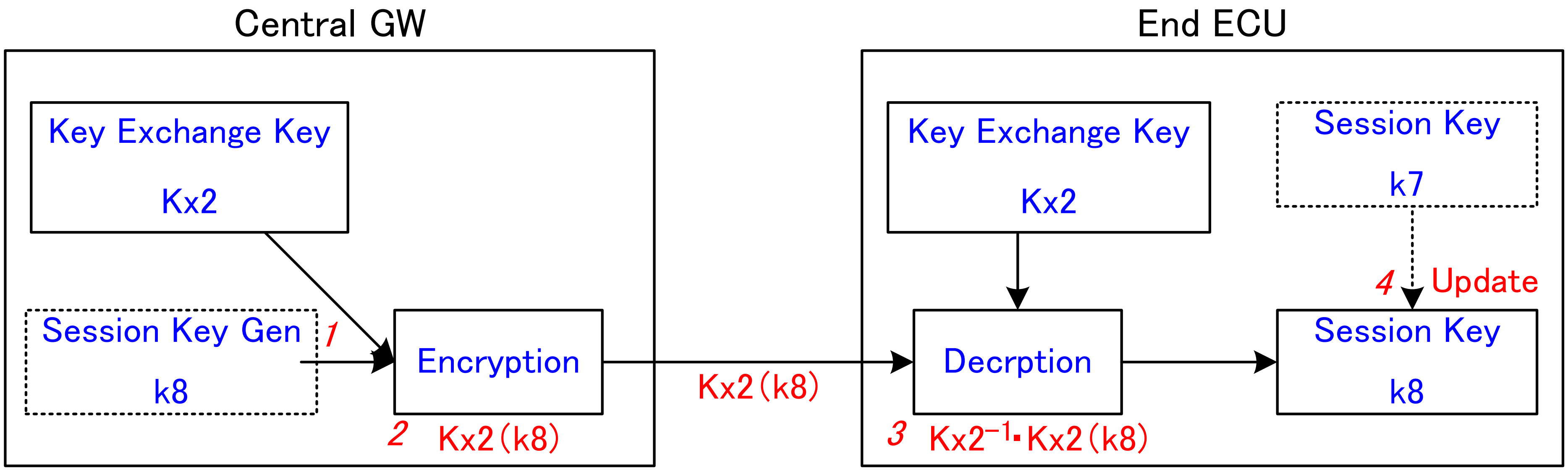
(2) Sharing of Key Exchange Key

2-5) Challenges are encrypted by the initial key. Also, responses are verified by the initial key.
6-8) A central GW generates a key exchange key, and encrypts it by the initial key.



(3) Sharing of Session Key

1) When an engine is started, the new session key is generated at the central GW.
2) The new session key is encrypted by the key exchange key, and is sent to the ECU.
3) The new session key is decrypted by the key exchange key in the ECU.

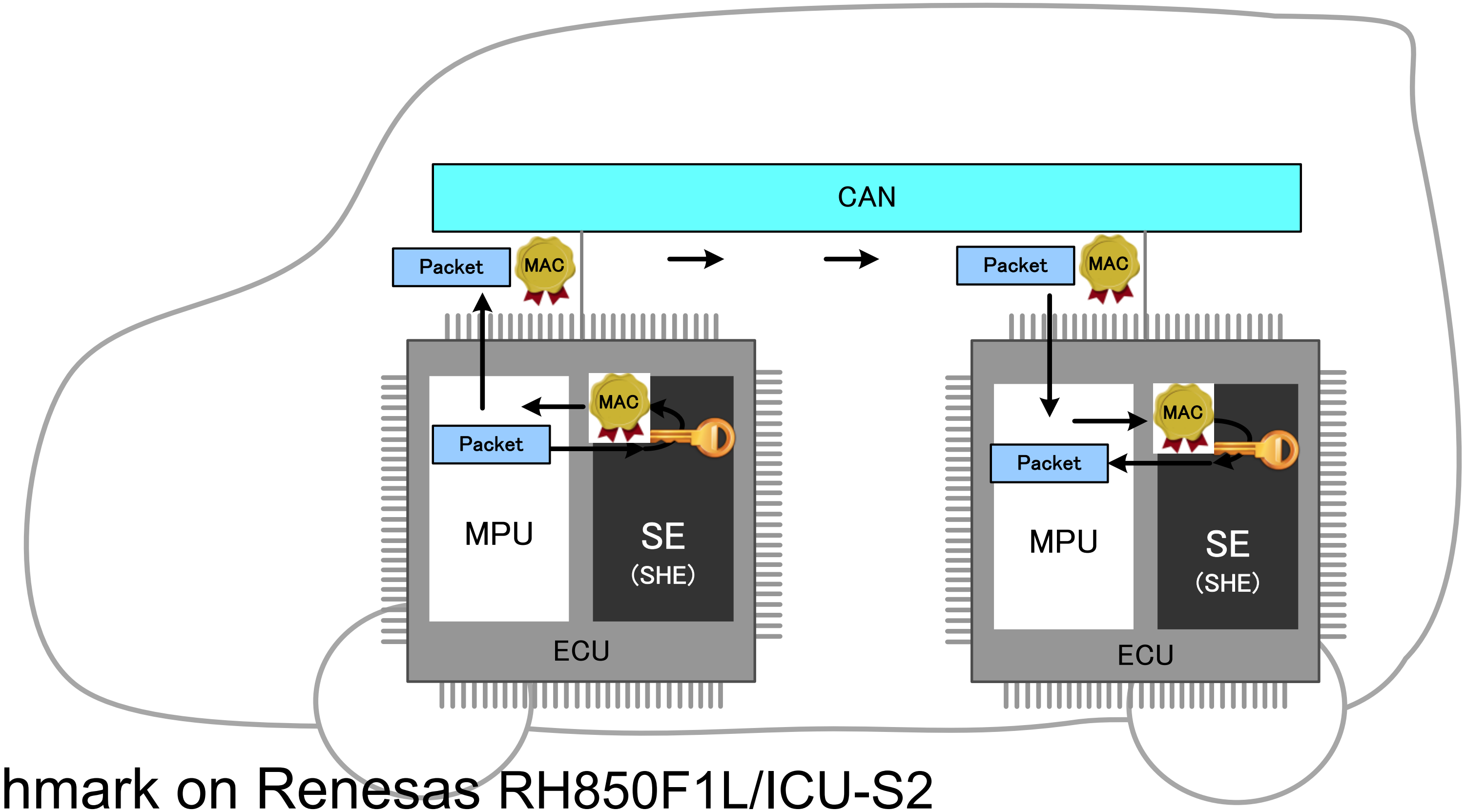


Appendix: MAC Insertion into CAN Packet

Message Authentication Code (MAC)

→ The MAC is inserted and is verified in the SE.

MAC = Digest (Control Data, Session Key, Packet Counter)



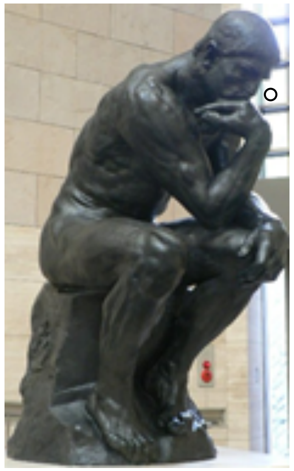
- Benchmark on Renesas RH850F1L/ICU-S2

→ The latency of generation or attestation of MAC is about 40 *us*.

Automotive Security

～Using Secure Element～

KDDI
Keisuke Takemori Ph.d

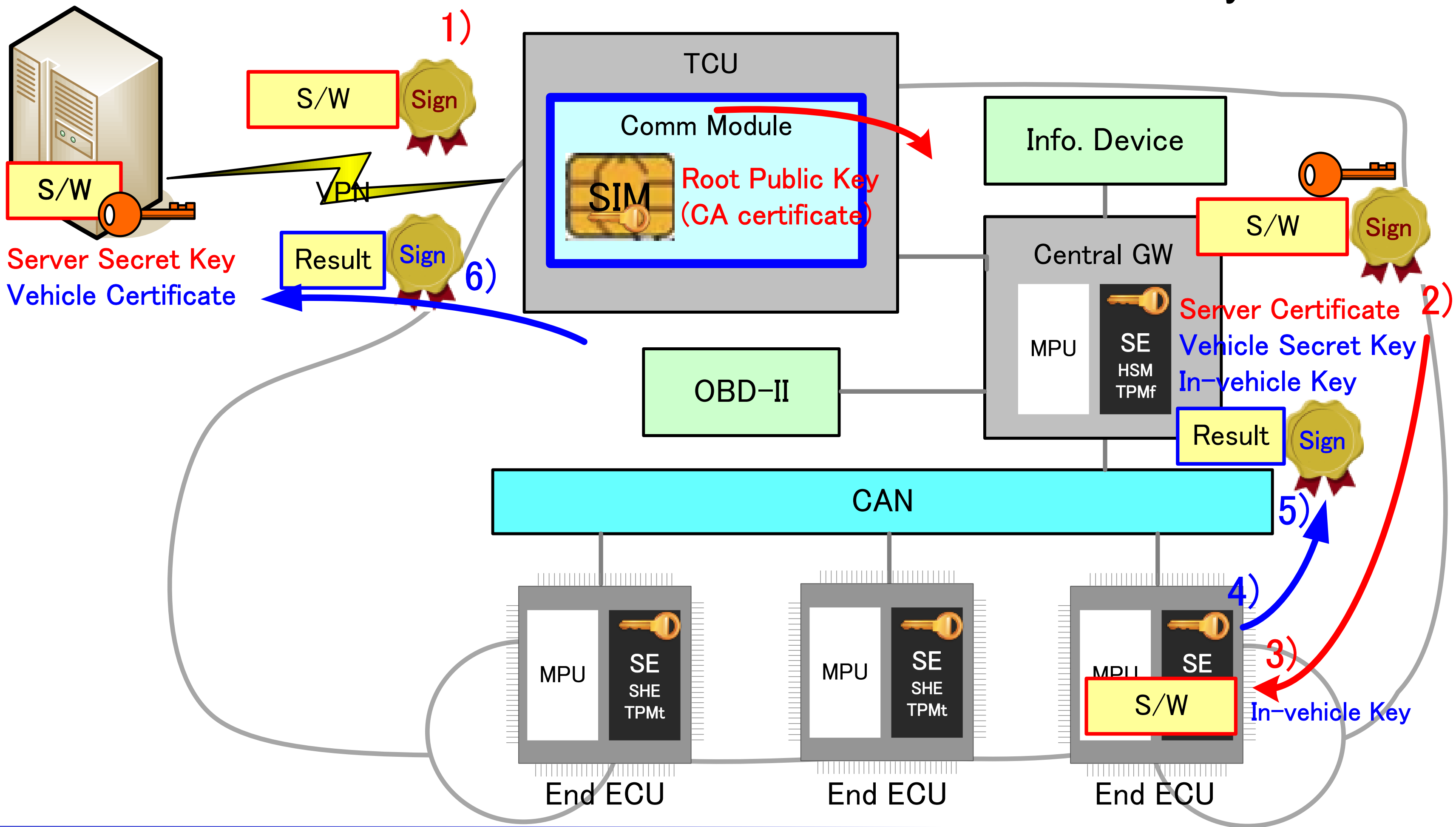


A secure element will be a trust anchor in a vehicle.

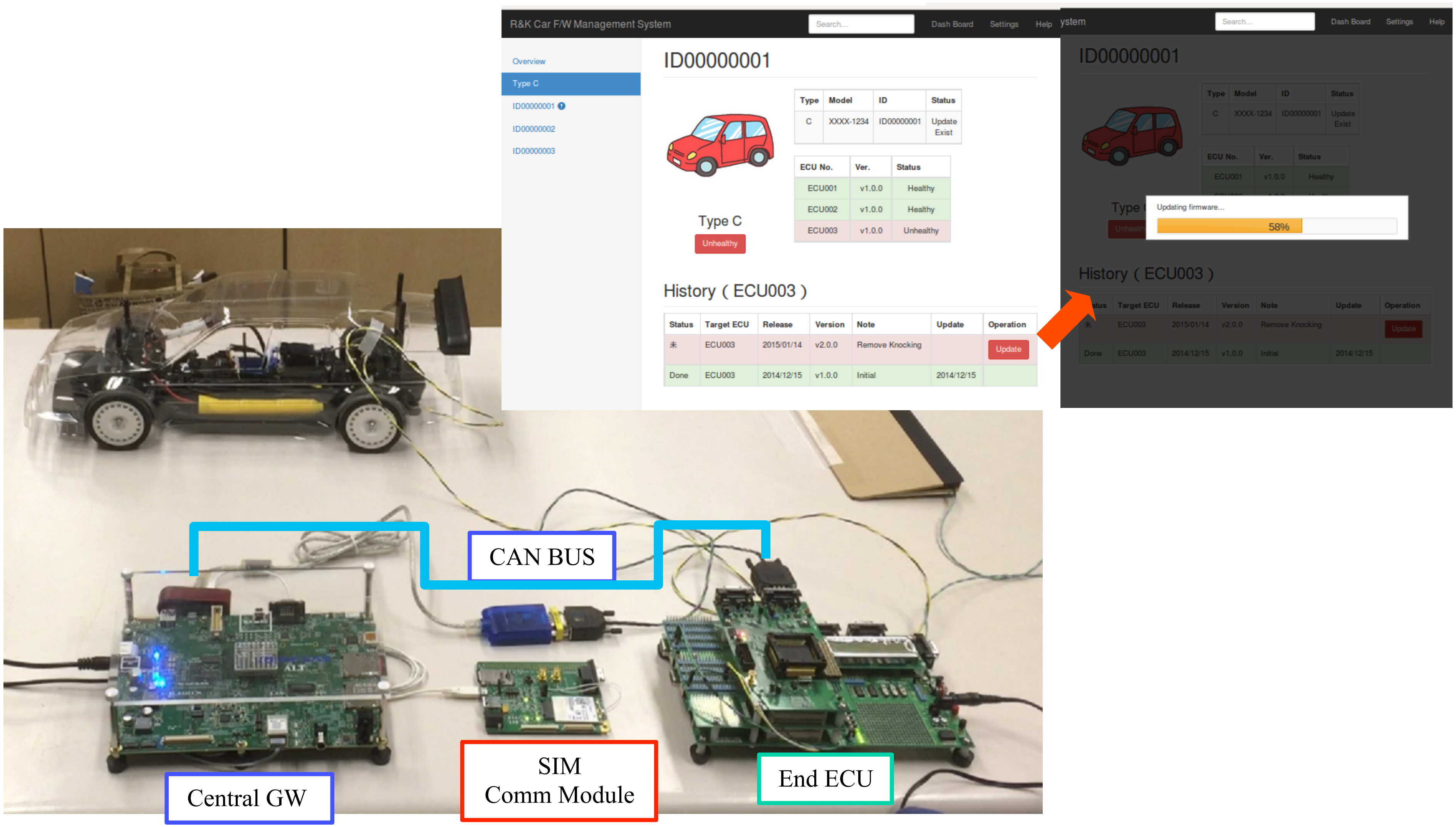
- Vehicle Incidents
- (1) Key Management → CAN+MAC
- **(2) Remote Reprogramming**
- Basis Security Techniques
- Conclusion

Secure Remote Reprogramming

→ The ECU code is verified by the central GW, and is applied it to the ECU.
→ The update status is measured, and is signed by the Central GW.
  → The server and the client certificates are authenticated by the SIM.



Demo: Remote Reprogramming

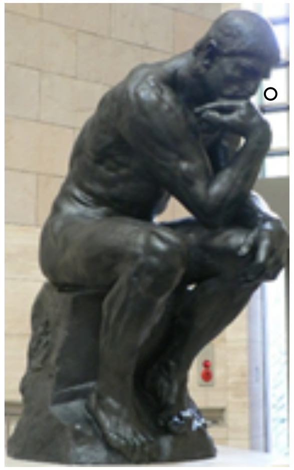


Automotive Security

～Using Secure Element～

KDDI
Keisuke Takemori Ph.d

A secure element will be a trust anchor in a vehicle.



- Vehicle Incidents
- (1) Key Management → CAN+MAC
- (2) Remote Reprogramming
- **Basis Security Techniques**
- Conclusion

Secure Elements for Vehicle

	SHE	HSM	TPM *	SIM
Tamper Resistance	Low	Low	Low	High
Latency	Small	Small	Small	Large
Accelerators	Few	Midium	Many	Many
App Execution	–	Support	–	Support
Device Cost	Low	Midium	Midium	High

Today's Suggestion

→ The combination of the secure elements should be considered.

→ "SHE" is applied to the end ECUs.

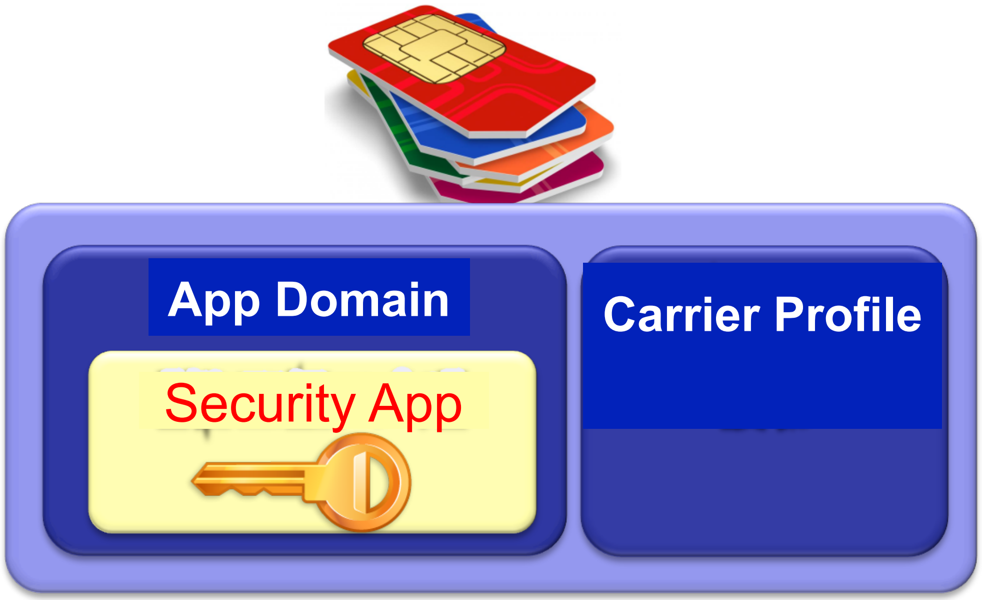
→ "HSM" is applied to the central GW.

→ "SIM" is applied to the in-vehicle CA.

* There are no commercial products for a vehicle in 2015.

Trust Anchor: Java Application in SIM

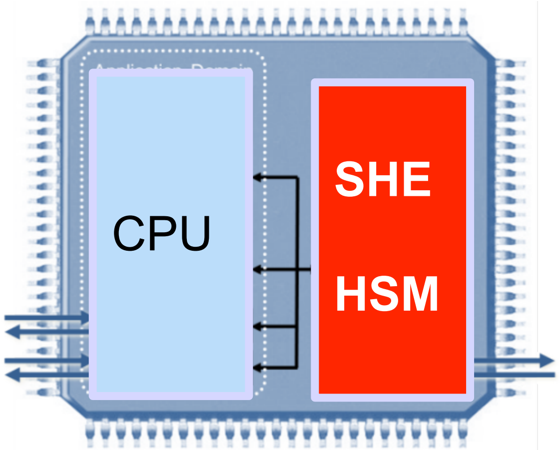
Advantages of SIM
- → Tamper resistant level of SIM is certified as EAL 5+.
- → Applications and/or keys are securely managed in the application domain.
- → When applications and/or keys in the SIM are compromised, it is securely updated over the air (OTA).



H/W Support e.g.,
- → RSA 1024, 2048
- → ECC 256
- → AES 128, 256
- → SHA-1, 256
- → HMAC

SHE, HSM

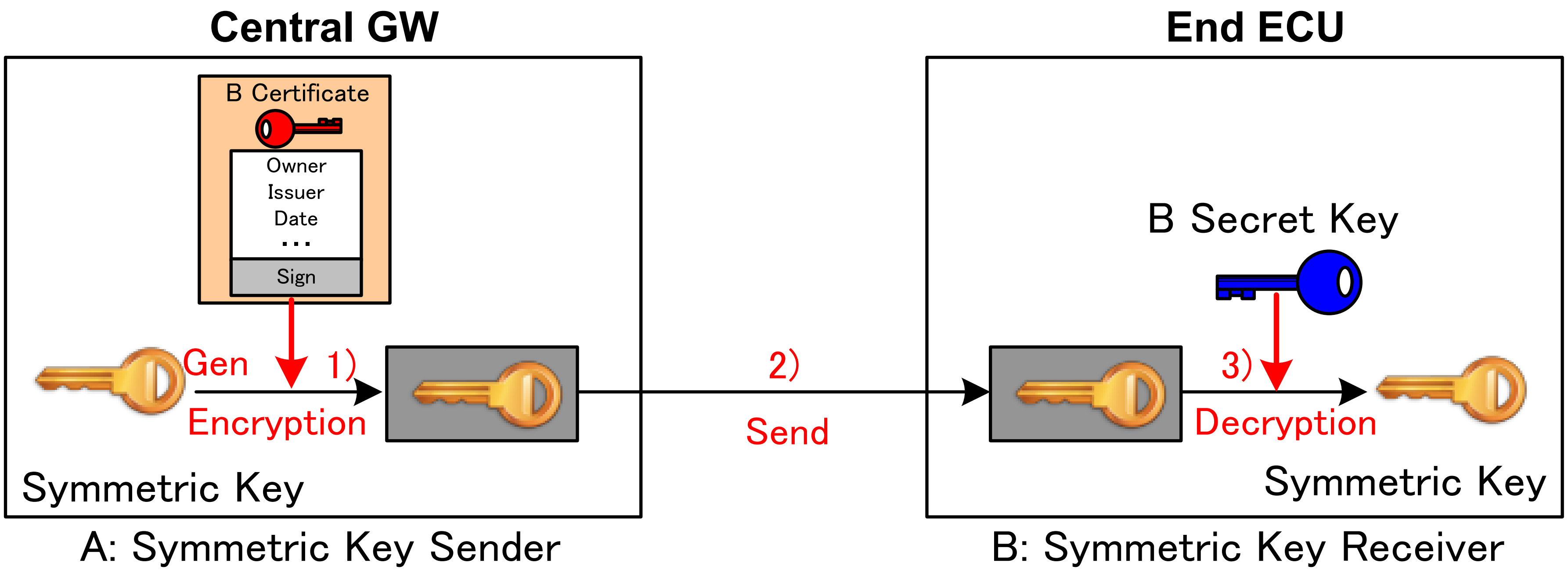
EVITA HSM http://www.evita-project.org/Publications/AEHR10.pdf



Light EVITA = SHE
Medium EVITA = HSM

		Central GW	End ECU SHE
	Full EVITA HSM	**Medium EVITA HSM**	**Light EVITA HSM**
Internal RAM	✓ (e.g. 64 kByte)	✓ (e.g. 64 kByte)	optional
Internal NVM (Non-volatile memory)	✓ (e.g. 512 kByte)	✓ (e.g. 512 kByte)	optional
Symmetric Cryptographic Engine (e.g. AES-128 CCM, GCM f/AE)	✓	✓	✓
Asymmetric Cryptographic Engine (e.g. ECC-256-GF(p) NIST FIPS 186-2 prime field)	✓		
Hash engine (e.g. Whirlpool)	✓		
Counters	✓ (e.g. 16 × 64-bit monotonic counter)	✓ (e.g. 16 × 64-bit monotonic counter)	optional
Random Number Generator	✓ (e.g. AES-PRNG with TRNG seed)	✓ (e.g. AES-PRNG with TRNG seed)	optional
Secure CPU (e.g. ARM Cortex-M3 32 bit, 50–250 MHz)	✓	✓	
Hardware Interface	✓	✓	✓

Asymmetric key/PKI-based Key Delivery



Key Distribution Steps

→ The central GW manages public keys of end ECUs.

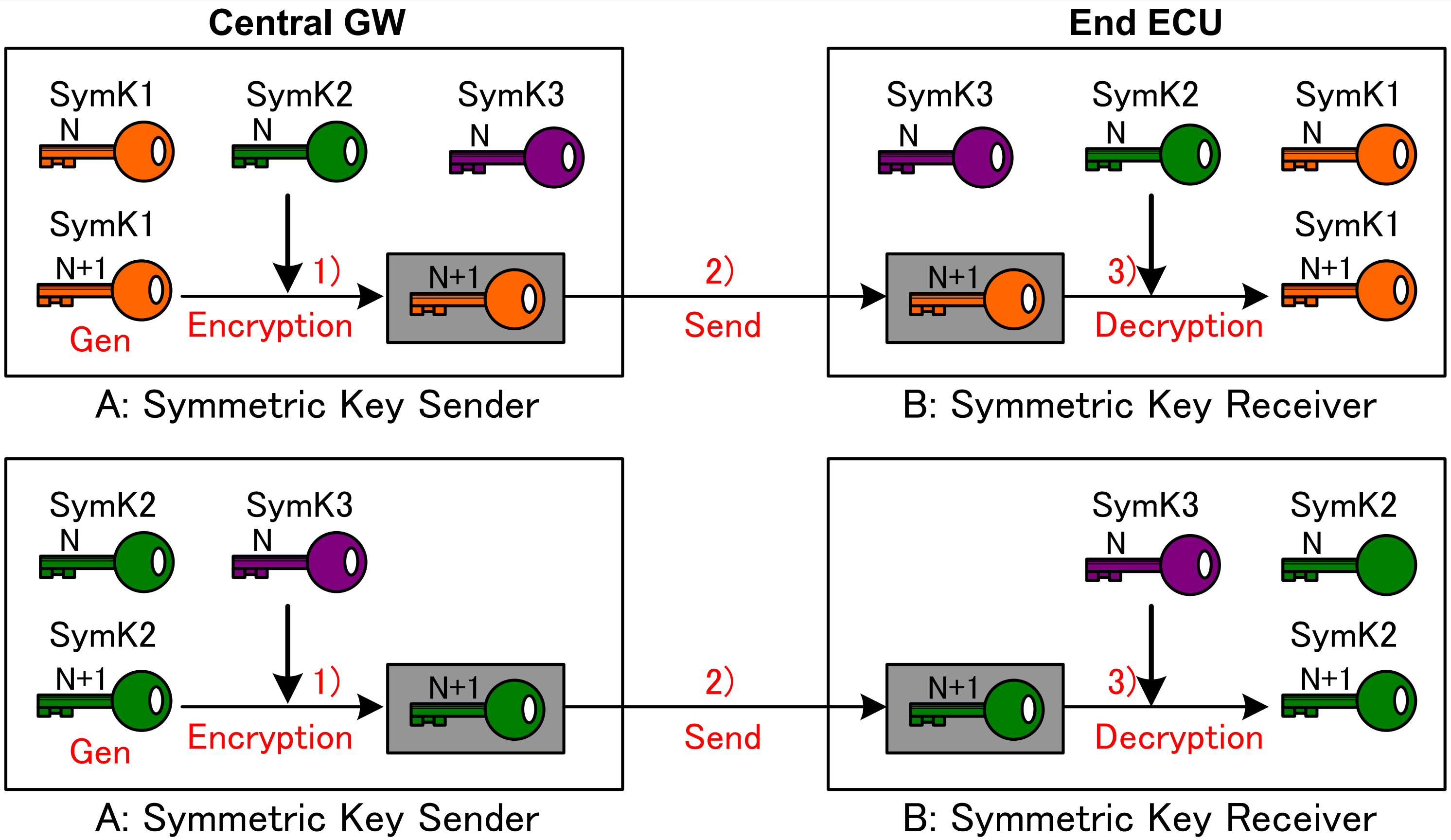
1) The symmetric key is generated and is encrypted by the public key of the end ECU.

2) The encrypted symmetric key is sent to the end ECU.

3) The end ECU decrypts and manages the symmetric key.

- The latency of public key-based processing is large.
- The size of encrypted data is large.

Multi-layer Symmetric Key-based Key Delivery



→ A few symmetric keys are managed in different registers of SE.

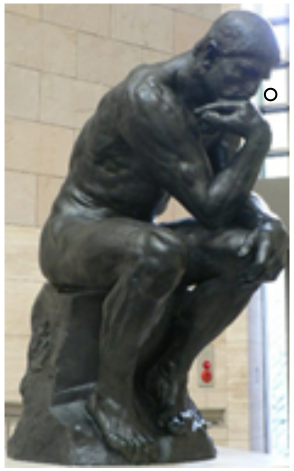
→ Low layer keys are encrypted/decrypted by the high layer keys to deliver the keys.

- The latency of symmetric key-based processing is small.
- The size of encrypted data is small.

Automotive Security

～Using Secure Element～

KDDI
Keisuke Takemori Ph.d



A secure element will be a trust anchor in a vehicle.

- Vehicle Incidents
- (1) Key Management → CAN+MAC
- (2) Remote Reprogramming
- Basis Security Techniques
- **Conclusion**

Conclusion

Authentication in the Internet "PKI"

→ The certificate of the asymmetric key is issued by the certification authority (CA), which is used for the unknown user authentication.

→ It should be applied to the V2X and the ECU code authentication.

Authentication in the Telecom Industry

→ The symmetric key in the SIM is issued by the carrier, which is used for the known user authentication.

→ It should be applied to the ECU key authentication.

Additional Suggestion for ECU Key Authentication

→ A CA on the Internet should not be used after the shipping.

→ Tamper resistance secure element, e.g., SIM, can be used as a CA in the vehicle.