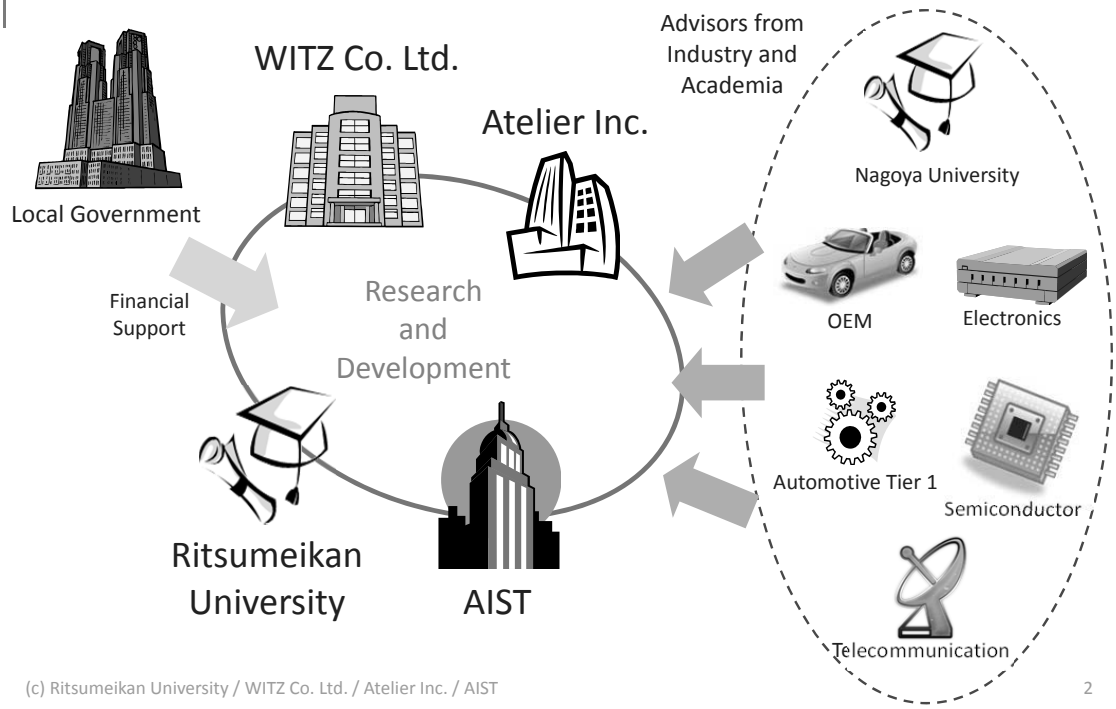
PUF-based Security Enhancement for Automotive Software Update

Hiroyuki Tomiyama

Ritsumeikan University
http://hiroyuki.tomiyama-lab.org/

MPSoC 2015

Our Team

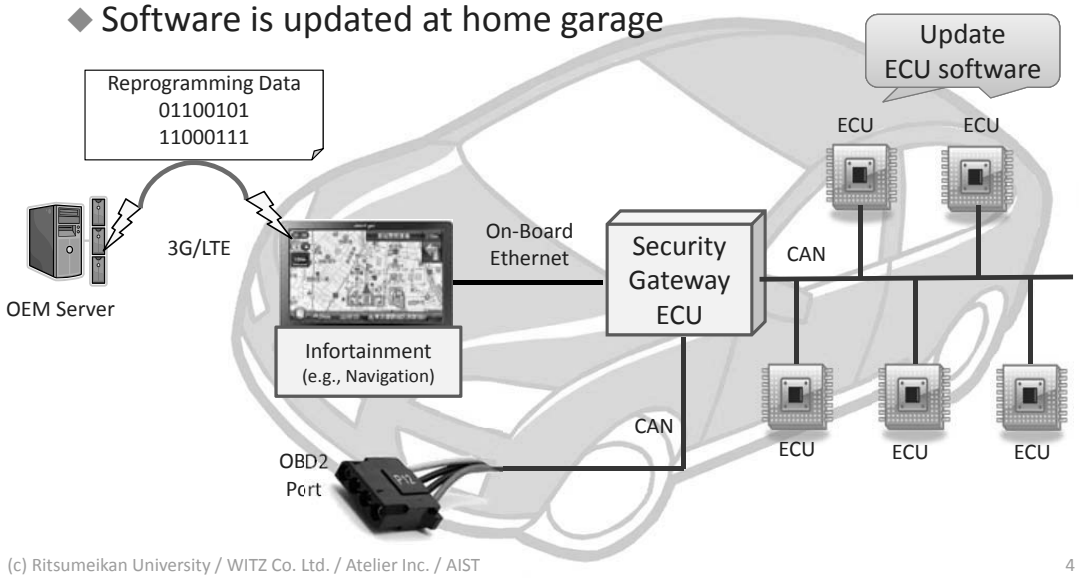


Background

- Recent trends in automotive electronics
  - More and more functionalities are implemented in software
  - Connected to the Internet and other networks
- Automotive software needs to be updated after sales
  - Higher security and safety, lower CO_2 emission, better mileage, better driving comfort, and so on
- At present, automotive software update (a.k.a. reprogramming) is only possible at OEM-authorized garages
  - At the time of recall, repair or periodic inspection
  - Reprogramming takes hours
  - Not as easy as *Microsoft Windows Update*

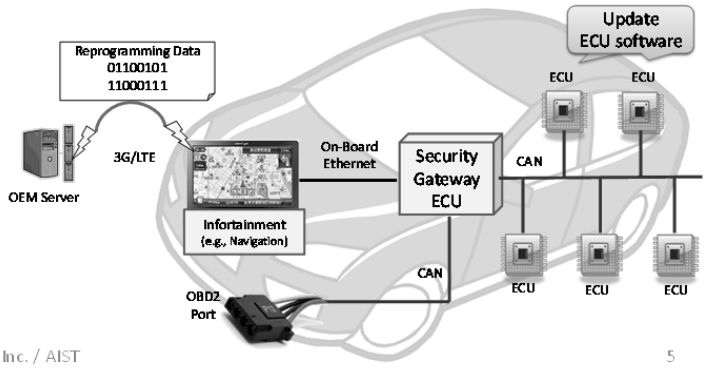
Remote Update of Automotive Software

- Future automotive software needs to be updated more often.
- Remote software update will be necessary
  - Software is updated at home garage



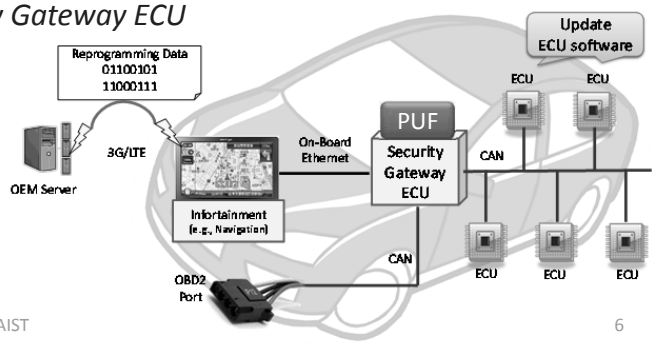
Problems

- Software update process must be
  - Secure
    - If the update process is not secure, the car gets more dangerous
    - Secure update needs *authentication* and *encryption*
  - Fast
    - User cannot drive the car during the update process
  - Inexpensive
    - Automotive manufacturers always worry about production costs
- But, there is a tradeoff



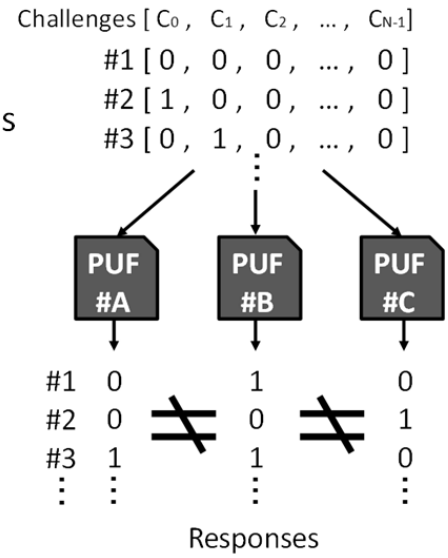
Our Approach

- We employ AES to encrypt reprogramming data between OEM server and vehicles
  - Faster and less expensive than public key cryptosystems (e.g., RSA)
  - But, we need to protect secret keys
    - In many systems, secret keys are stored in *secure non-volatile memory*
    - Secure NVM is expensive
- We encrypt secret keys and use *PUF* as an AES key
  - The encrypted keys can be stored in normal NVM
  - Other secure data can be stored in NVM or RAM with PUF-based encryption
- PUF is implemented in *Security Gateway ECU*



PUF: Physical(ly) Unclonable Functions

- PUF exploits physical variation of individual devices
  - Unclonable
  - Similar to fingerprint, but functions with inputs and outputs
- PUF generates unique ID numbers
- Various PUF implementations
  - Optical PUF
  - Magnetic PUF
  - SRAM PUF
  - Arbiter PUF
  - Ring Oscillator PUF
  - and more

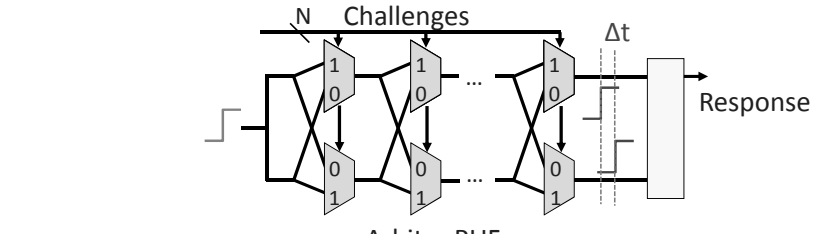


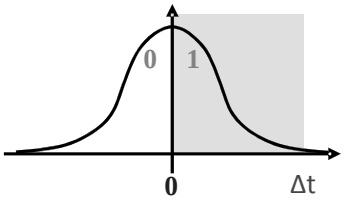
PUF: Physical(ly) Unclonable Functions

- PUF needs to be unique and robust
- Uniqueness
  - PUF individuals should produce different responses (outputs) from the same challenges (inputs)
  - Professor Fujino, a member of our team, proposed DPM-based arbiter PUF for better uniqueness
- Robustness
  - A PUF should produce same responses from same challenges in any condition over years
  - Robustness against aging, temperature, voltage variation, and so on
  - Error correction is necessary

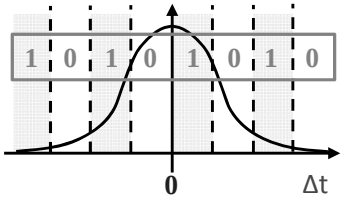
DPM-based Arbiter PUF

- Developed by Professor Takeshi Fujino (our team member) [ISCAS 2011]
- Based on arbiter PUF
  - multiplexer chain
- Finer-granularity delay time measurement
  - Higher uniqueness





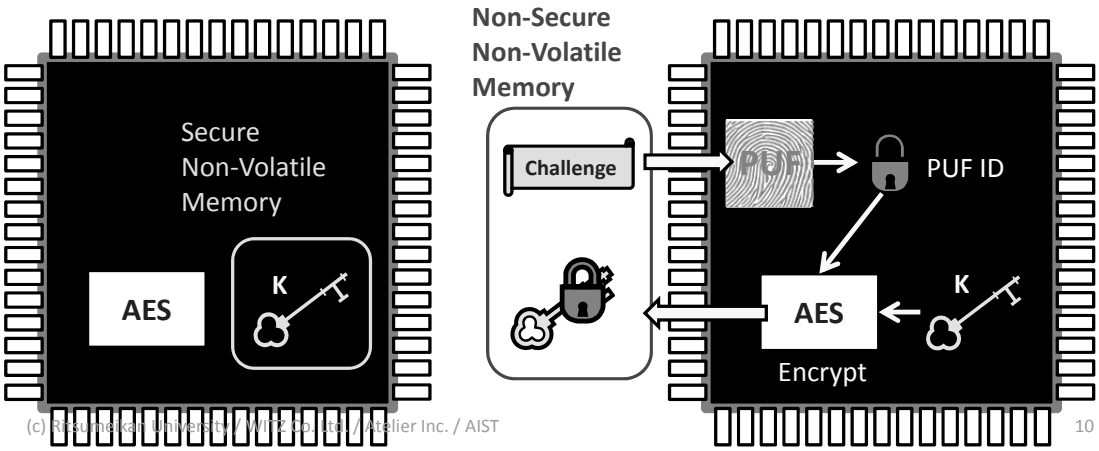
Conventional Arbiter PUF



DTM-based Arbiter PUF

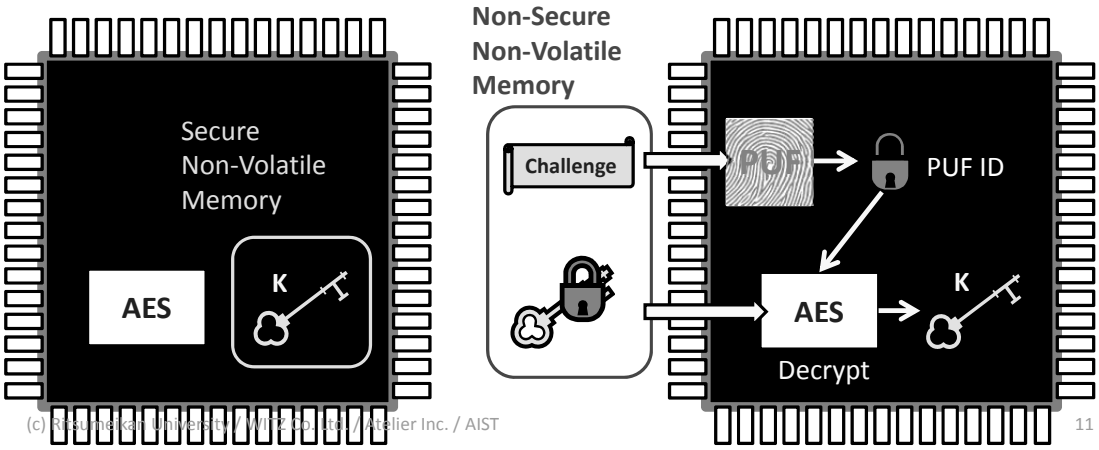
Secure Key Storage with PUF

- Traditional Method
  - Key (K) has to be stored in secure non-volatile memory
- Our Method
  - ➡ Manufacturing time
    - Encrypt the key (K) with PUF-ID and store the PUF-encrypted key in non-secure NVM
  - Reprogramming time
    - Decrypt the key with PUF-ID

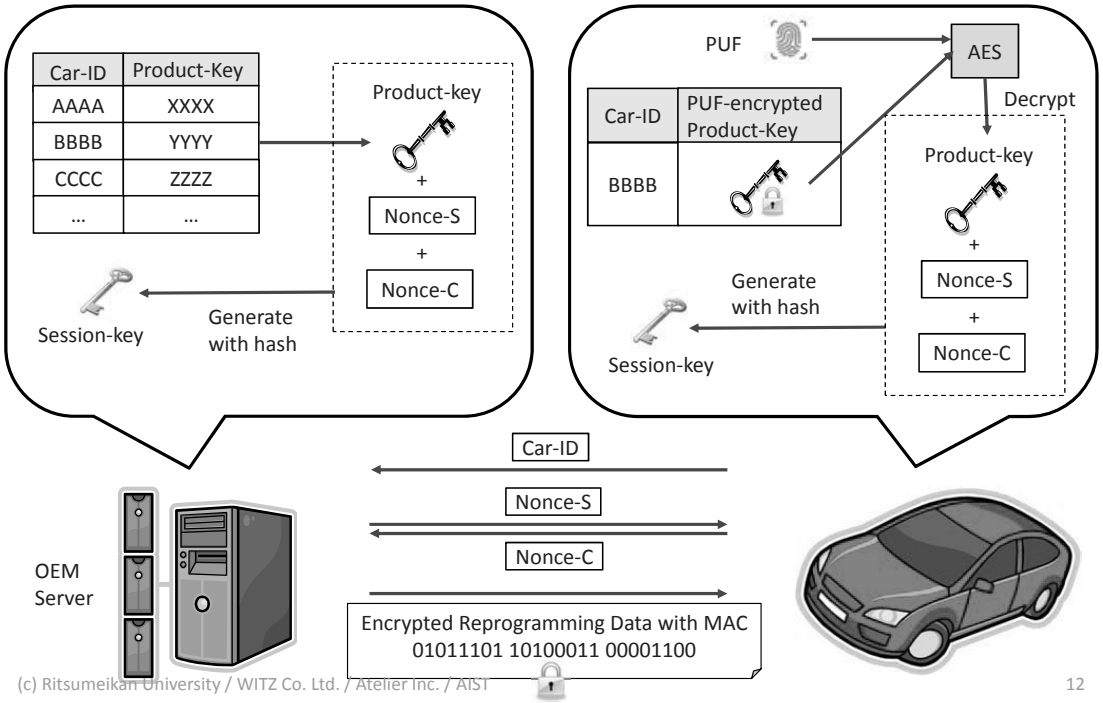


Secure Key Storage with PUF

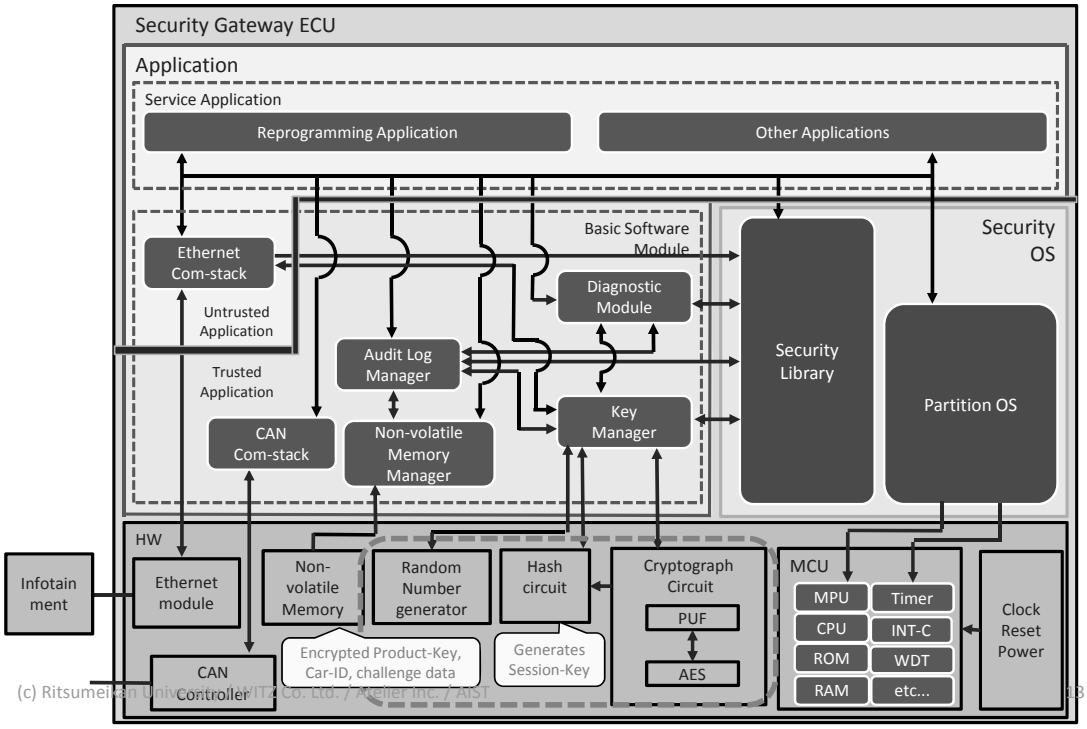
- Traditional Method
  - Key (K) has to be stored in secure non-volatile memory
- Our Method
  - Manufacturing time
    - Encrypt the key (K) with PUF-ID and store the PUF-encrypted key in non-secure NVM
  - ➡ Reprogramming time
    - Decrypt the key with PUF-ID



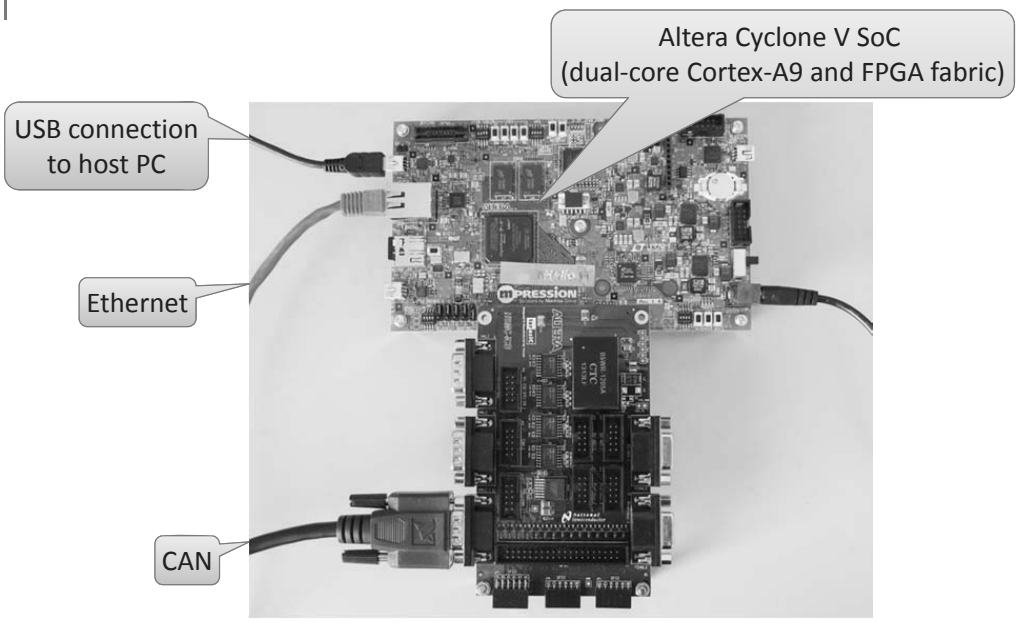
Product-Key and Session-Key



Security Gateway EUC Architecure



FPGA Prototype of Security Gateway ECU



Concluding Remarks

- Our on-going project on remote update of automotive software
  - The key idea is to encrypt secret keys using PUF-ID as a key
  - Secure NVM is not necessary
  - FPGA prototyping of security gateway ECU
- Future work
  - Prototyping a server system
- Special Thanks
  - Takeshi Fujino (Ritsumeikan University)
  - Hideyuki Takeda (WITZ Co. Ltd.)
  - Ayumu Sugiyama (WITZ Co. Ltd.)
  - Hiroaki Hara (WITZ Co. Ltd.)