Reboot-Oriented IoT: Life Cycle Management in Trusted Execution Environment for Disposable IoT devices

Kuniyasu Suzaki [1), Akira Tsukamoto [1),
Andy Green [2), Mohammad Mannan [3)

[1) National Institute of Advanced Industrial Science and Technology
[2) Warmcat
[3) Concordia University

Annual Computer Security Applications Conference (ACSAC) 2020
10 December 2020 13:30-14:45 Session “System and Hardware Security”

Outline

- Background for IoT Security
- Concept of Reboot-Oriented IoT
  - Network boot protected by TEE (Occasional)
  - Live Memory Forensics protected by TEE (Periodical)
  - Life Cycle Management based on PKI and protected by TEE
- Implementation
  - RO-IoT on Linux and OP-TEE with Arm TrustZone
  - Watchdog timer for autonomous reboot protected by TEE
- Performance
- Conclusion

Occasional > Periodical

Background for IoT Security

- IoT devices targeted by RO-IoT
  - Smart cities and smart farming assumes many IoT deices are geologically distributed and managed by M2M (Machine to Machine).
    - IoT devices works as AI Edge of Fog-Computing and use Linux to run intelligent applications.
  - The devices are desired to be disposable when they finish the role. Self-destruction technologies or ITU E-Waste policy are developed, but …
- Concerns
  - General Security issues are not solved.
    - If IoT devices are hijacked by malware (ex., Mirai), it is difficult to recover because no administrator on each device.
  - The supply chain includes some stakeholders which have responsibilities (device, software, and service). These stakeholders want to ruin the device when the responsibilities are terminated because unmanaged IoT devices become Cyber Debris. They don’t want to support the expired devices.

Reboot-Oriented IoT

- Purpose
  - To prevent IoT from unknown attacks
  - To offer suitable life cycle management
- Contributions and challenges
  - 3 special security mechanism protected by TEE (Trusted Execution Environment)

1. Occasional Network Reboot to recover from unknown attacks
   - The IoT runs OS on memory only and reboots (re-installs) OS.
2. Periodical Memory Forensics to detect unknown attacks
   - Assumption: AI-Edge IoT runs a few intended applications only.
   - RO-IoT allows to run the whitelisted application only.
3. Life Cycle Management to prevent becoming cyber debris
   - PKI certificates (CA, Server, and Client) are linked to the lifetimes (Device, Software, and Service).

Example

Occasional	>	Periodical
42 hours		15seconds
=15sec *10,000		

Secure Rebooting

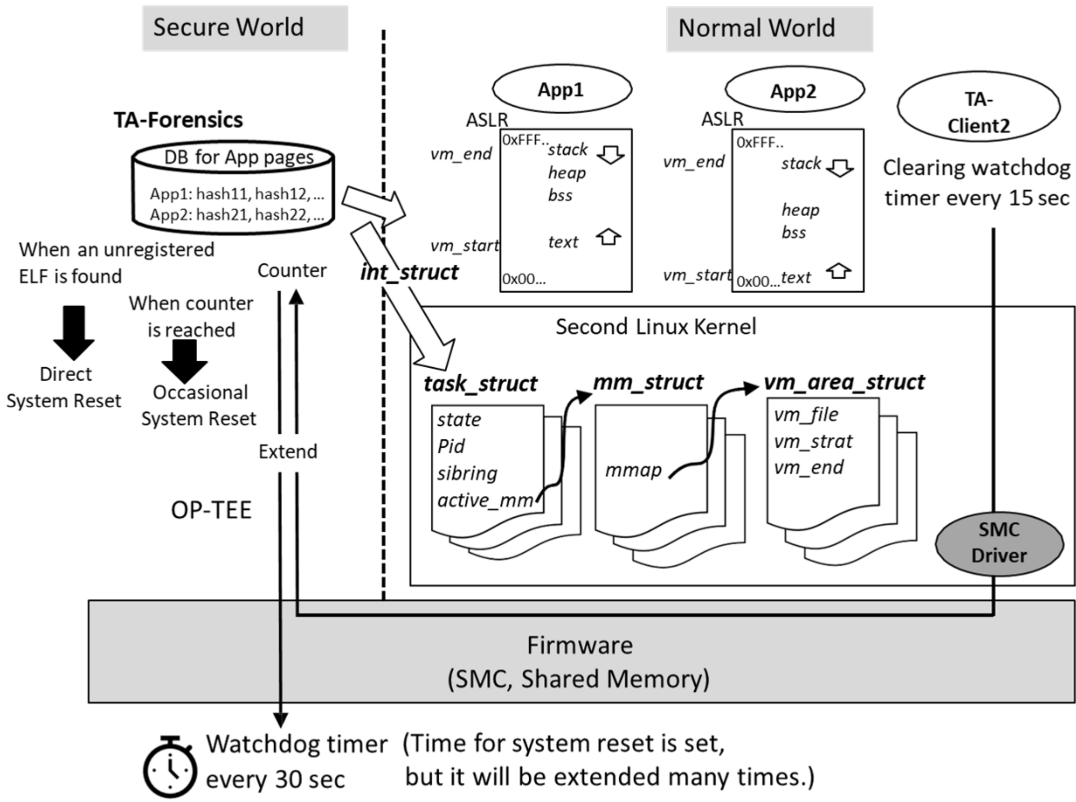
- Reboot (i.e., Re-Installation) is a suitable way to recover from unknown attacks.
  - Related works; CIDER[IEEE SP'19], Misery Graphs[IEEE TIFS'17], YOLO[SPIE'19], TPM2.0 Authenticated Countdown Timer, etc.
- Challenges
  1. Secure network boot
     - The OS image is downloaded by HTTPS and verified by **TEE**.
       - The connection of HTTPS is terminated by TEE and securely downloaded in TEE.
     - TEE has no mechanism to reboot an OS. So, the OS image is transferred to REE and rebooted.
       - The reboot mechanism utilizes the Linux's **kexec**.
     - The download OS runs memory only, i.e., total reinstallation.
  2. Secure autonomous rebooting
     - **watchdog timer** protected by **TEE**.

In order to implement TEE and reboot mechanism easily, small Linux is used as a bootloader(detail in implementation).

Secure Memory Forensics

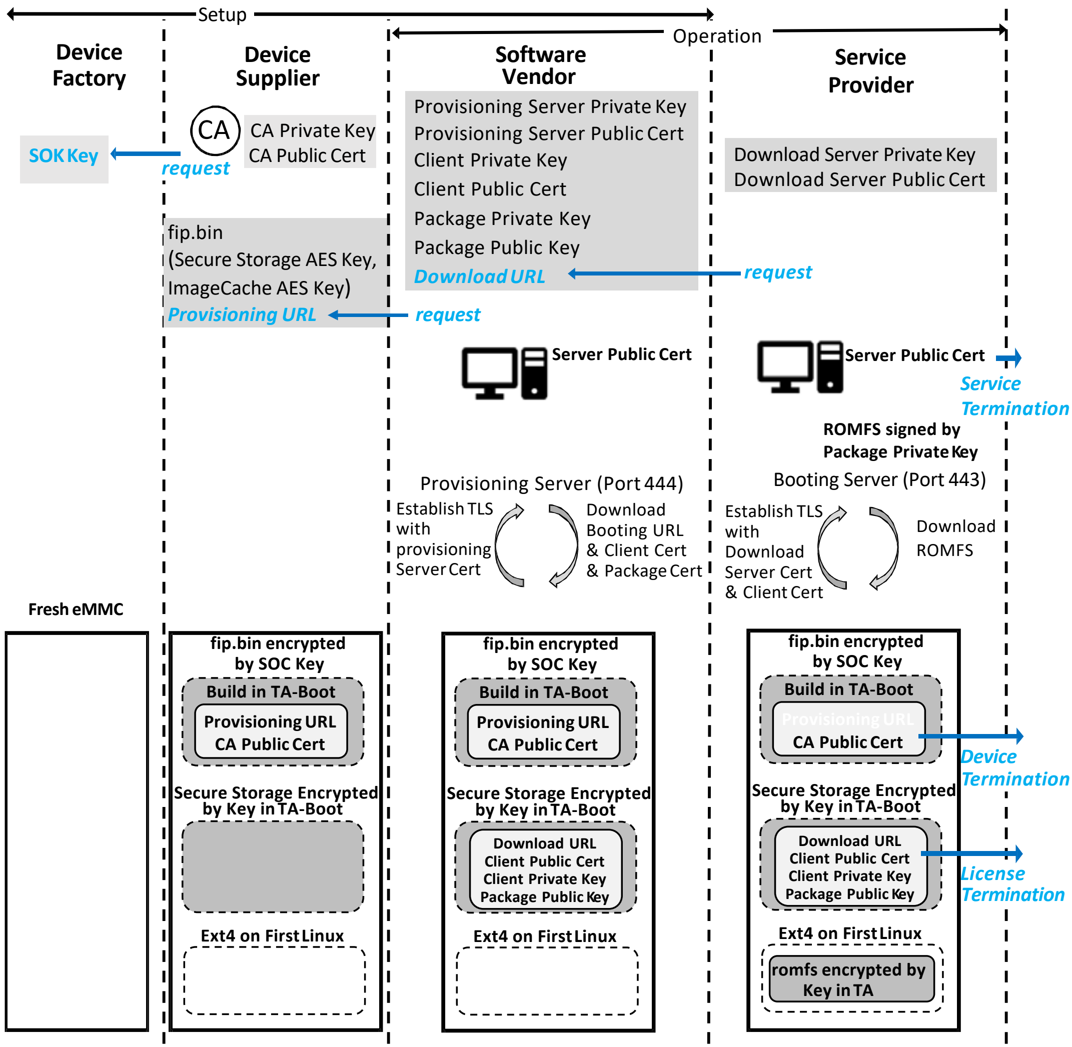
- Assumption: IoT runs a few applications only.
- RO-IoT applies whitelisting security on memory forensics protected by TEE.
- Memory forensics in TEE (TA-Forensics) has DB for whitelisting apps and retrieves the *task_struct* of Linux kernel.
  - If unknown application is found, TA-Forensics causes system rest.
  - TA-Forensics sets the watchdog timer and must be activated periodical to set again to prevent system rest.
  - If the TA-Forensics runs more than thresh hold, it causes system rest occasionally.

System rest causes **secure reboot**.



Secure Life Cycle Management

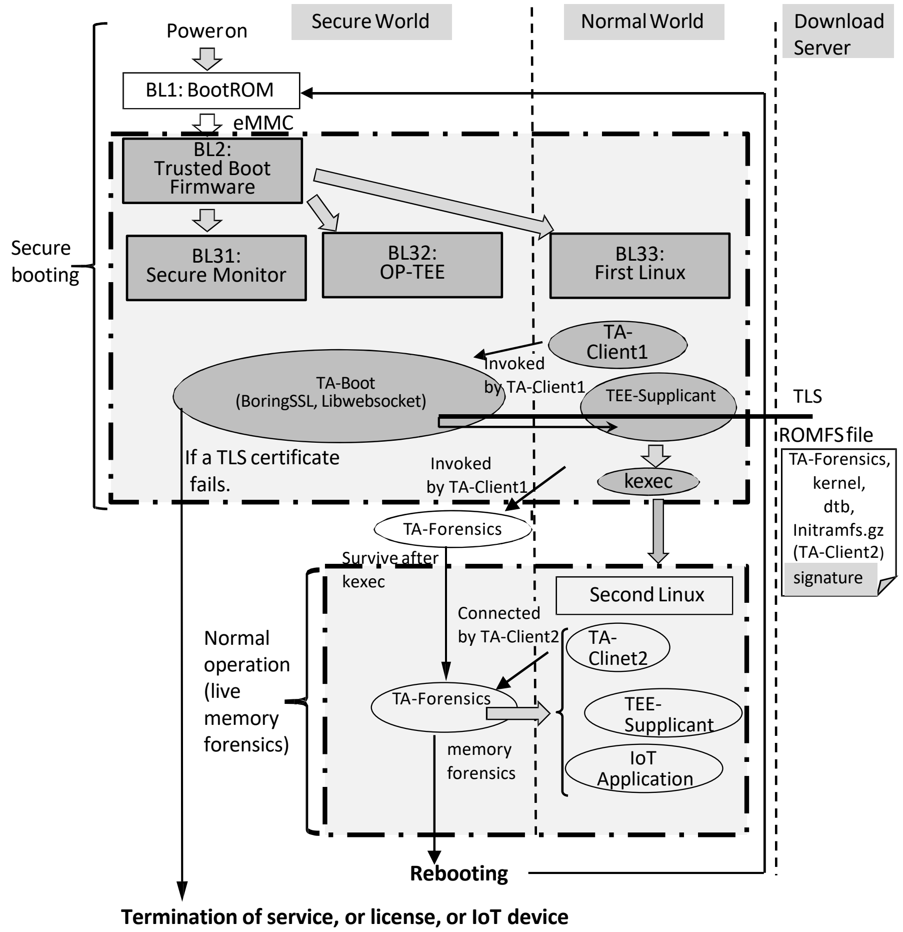
- RO-IoT assumes
  - Life cycle of Device
  - Life cycle of Software
  - Life cycle of Service
- The life cycles are linked to PKI of HTTPS (TLS) certificates (CA, Client, and Server).
  - CA Pub Cert is included in TEE by Device Supplier.
  - Client Pub Cert is included in TEE by Software Vendor.
  - Server Pub Cert is managed by the server of Service Provider.
- The certificates are verified in the TEE when a HTTPS connection is established at **secure reboot**. If a certificate is invalid, RO-IoT does not boot the OS.



Cyber Physical
Security Research Center

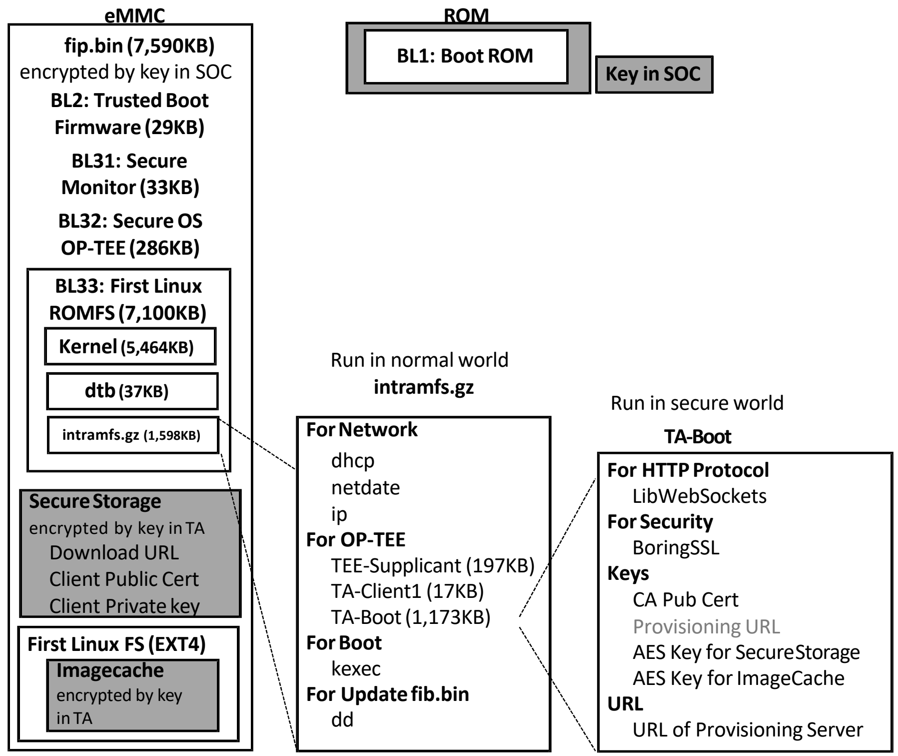
Implementation

- 2 types of Linux
  - First Linux: As a bootloader with kexec
    - The bootloader supports OP-TEE. TA-Boot on OP-TEE downloads the IoT OS image with HTTPS.
    - TA-Forensics is launched on the first Linux because it must be hidden from the second Linux.
    - The downloaded image is moved to REE (Linux) to boot it with kexec.
  - Second Linux: As a IoT OS
    - Applications are monitored by TA-Forensics.
    - TA-Forensics is passive, and the activation must be controlled by an application on the second Linux.
      - Activation Mechanism: TA-Forensics are periodically activated because it causes rebooting with **watchdog timer** if it is not reset.



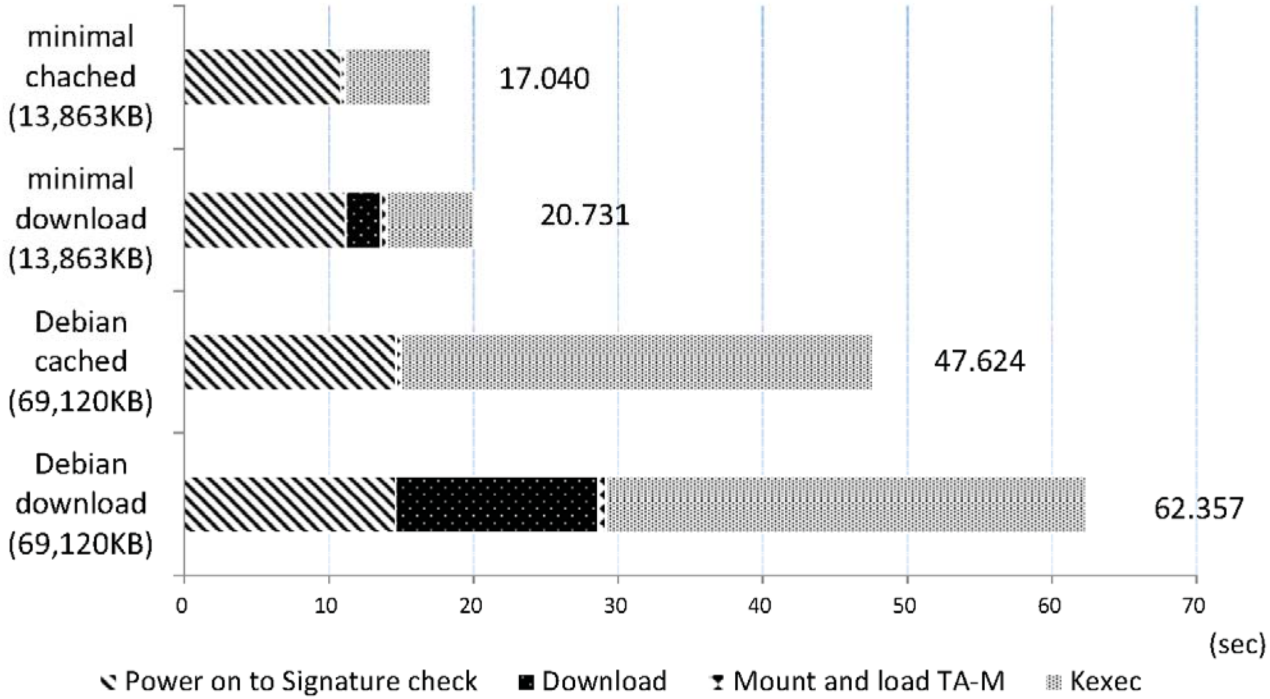
Implementation

- RO-IoT is implanted on HiKey board (Arm Cortex-A, 2GB Memory).
- eMMC includes the bootloader (First Linux) with OP-TEE image (TA-Boot).
  - TA-Boot includes BoringSSL and LibWebSockets for HTTPS.
- The bootloader has a mechanism to cache an OS image. If the OS image is not updated, the bootloader use the saved OS image to eliminate the download time.



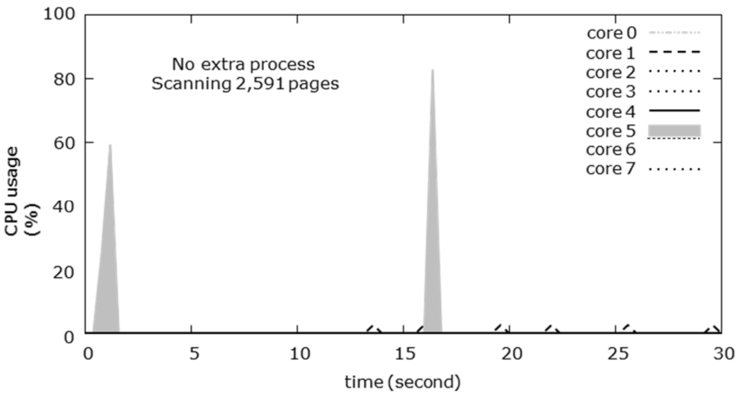
Performance of Reboot (Reinstallation)

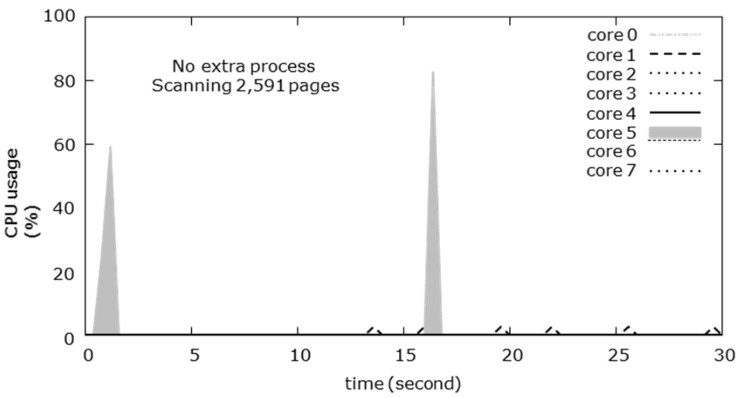
- Downloaded OS image
  - Minimal 13,863KB
    - initramfs.gz 8,637KB
    - TA-Forensics 226KB
  - Debian 69,120KB
    - initramfs.gz 63,340KB
    - TA-Forensics 781KB

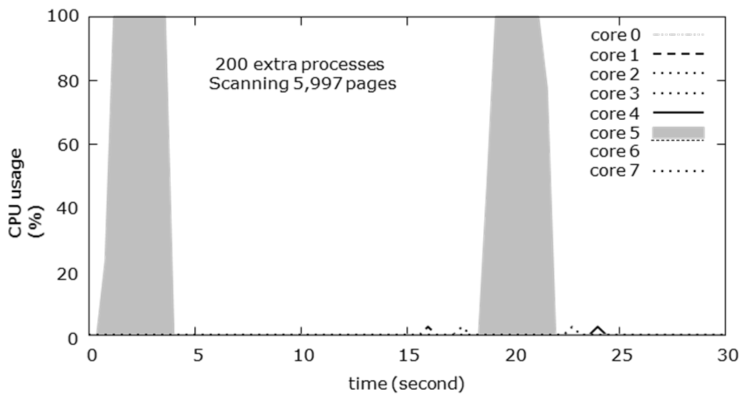


Performance of memory forensics on TEE

- Watchdog timer is set to cause within 30 seconds.
  - The time reset is issued every15 seconds.
  - The memory forensics must finish within 15 seconds (until next time rest is issued).
- We evaluated the memory forensics on TEE with 0, 100, and 200 extra processes.







Future Work

- Target applications of RO-IoT were AI Edge, which allowed short-time suspension.
- Next target is mission critical applications (mobility and life support for smart city).
  - RO-IoT with partial OS update mechanism.
  - RO-IoT with fault tolerant mechanism.

Conclusions

- Return-Oriented IoT makes IoT device disposable with 3 security mechanisms protected by TEE (Trusted Execution Environment).
  1. Occasional Network Reboot replaces whole OS image on memory and recovers from unknown attacks
  2. Periodical Memory Forensics detects unknown attacks
  3. Life Cycle Managements linked to PKI certificates prevents becoming cyber debris