Implement Android Tamper-Resistant Secure Storage and Secure it in Virtualization

Bing Zhu (bing.zhu@intel.com)

Contributors:

Yang Huang, Tomas Winkler, Wei Deng, Yadong Qi, Kai Wang, Luhai Chen, Eddie Dong

Open Source Technology Center / Software and Services Group (SSG)

Linux Security Summit Europe 2018

October 24, 2018

NOTICE & DISCLAIMER

- Intel technologies’ features and benefits depend on system configuration and may require enabled hardware, software or service activation.
- Performance varies depending on system configuration.
- Intel, the Intel logo are trademarks of Intel Corporation in the U.S. and/or other countries.
- *Other names and brands may be claimed as the property of others.

Agenda

Problem Statement

Replay Protected Memory Block (RPMB)

VT-TEE/Trusty* Secure Storage (SS)

Secure Storage Virtualization in ACRN* Hypervisor
(*TEE Isolation, Replay/Integrity Protection and Storage Encryption for Confidentiality*)

Conclusion and Future Considerations

Problem Statement

Data security and privacy:

- Screen-unlock (password/pin/pattern) attempt failure record for defending against brute force attack: https://source.android.com/security/authentication/gatekeeper
- The version of system image for preventing roll-back attack
- Keybox (keypairs), e.g. for content protection and attestation
- The templates of fingerprint or iris sensor images for authentication

Google* Android* CDD requirements since Marshmallow :

- [SR] STRONGLY RECOMMENDED/ SHOULD to use tamper-evident storage

Replay Protected Memory Block (RPMB)

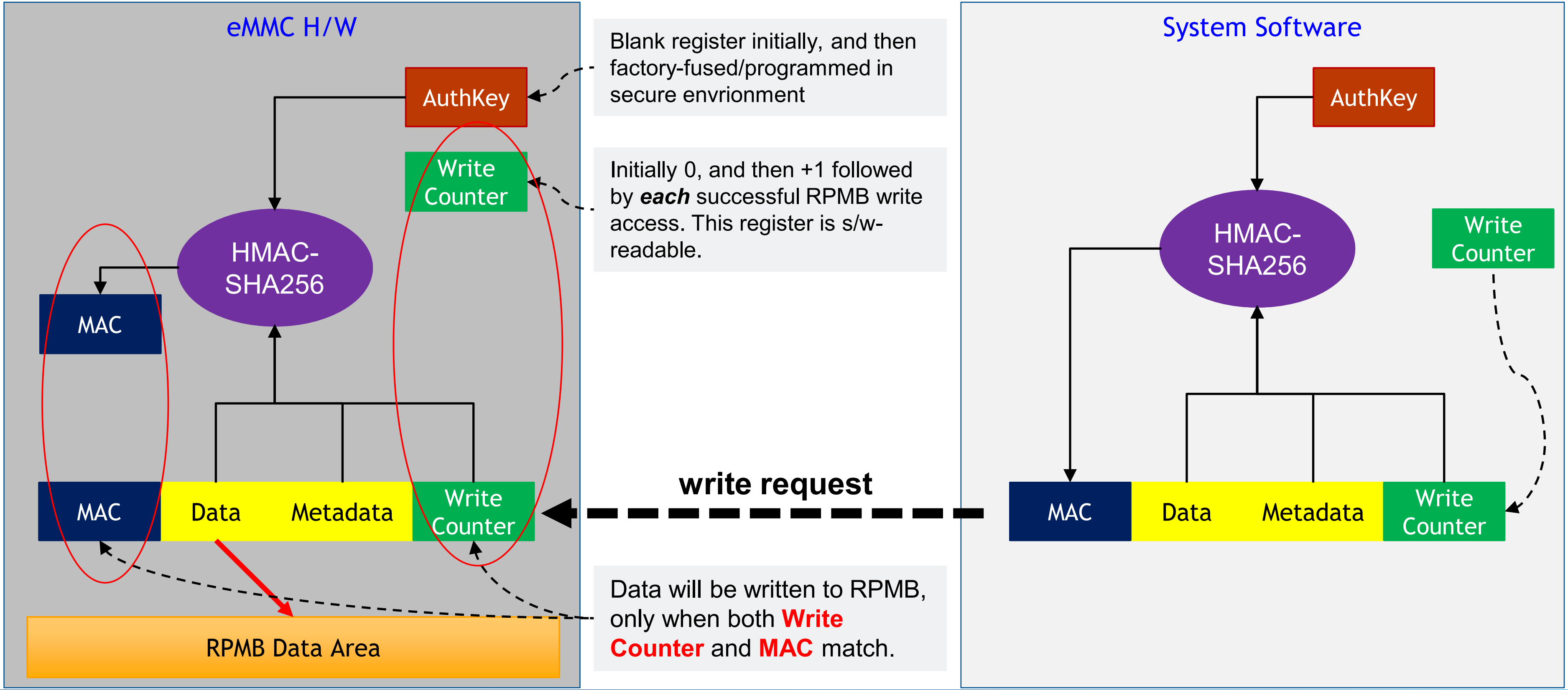
RPMB Partition (e.g. eMMC)

Fixed in Size, typically 4MB
(128KB ~ 16MB)

Technical Details / Characteristics

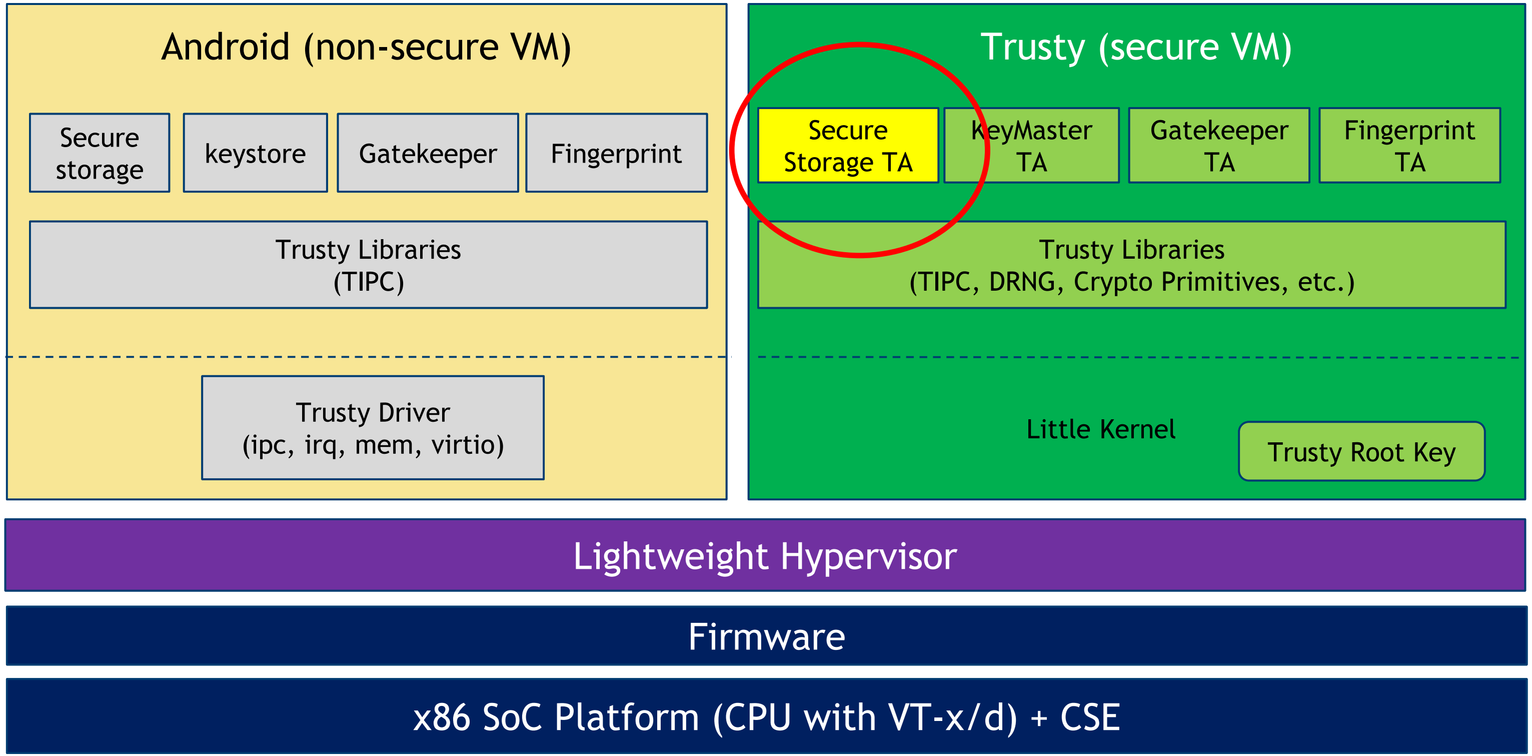
1. Authentication key (RPMB AuthKey) is required.
   - The Key must be programmed before any access, the built-in algorithm is HMAC-SHA256.
   - The key can only be programmed once in device life time, and is invisible to any software after it is programed into h/w device.
   - Key must be required to write data a RPMB partition.
   - Notes: Without RPMB key, read access is still possible, but the data being read may not be authentic (no guarantee of data integrity and replay protection). Hence, RPMB doesn't provide data confidentiality protection (encryption is done by software if necessary)
2. Replay Protection
   - Storage controller H/W built-in monotonic Write Counter is used for replay-protection on **WRITE** access; Software generated Random Number is used for replay-protection on **READ** access.

How it works (e.g. authenticated write access)

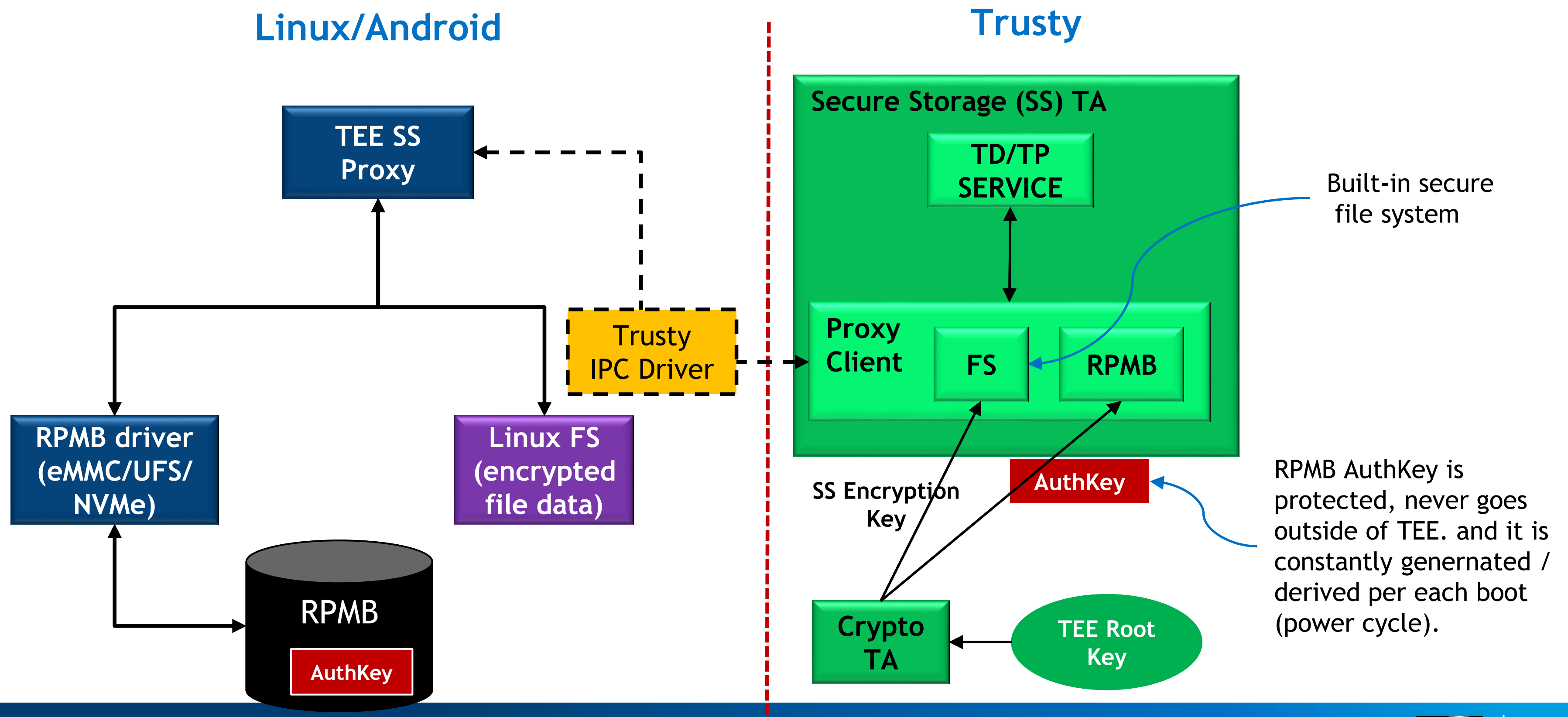


VT-TEE/Trusty Secure Storage (SS)

VT-TEE/Trusty in Android (Two-VM)



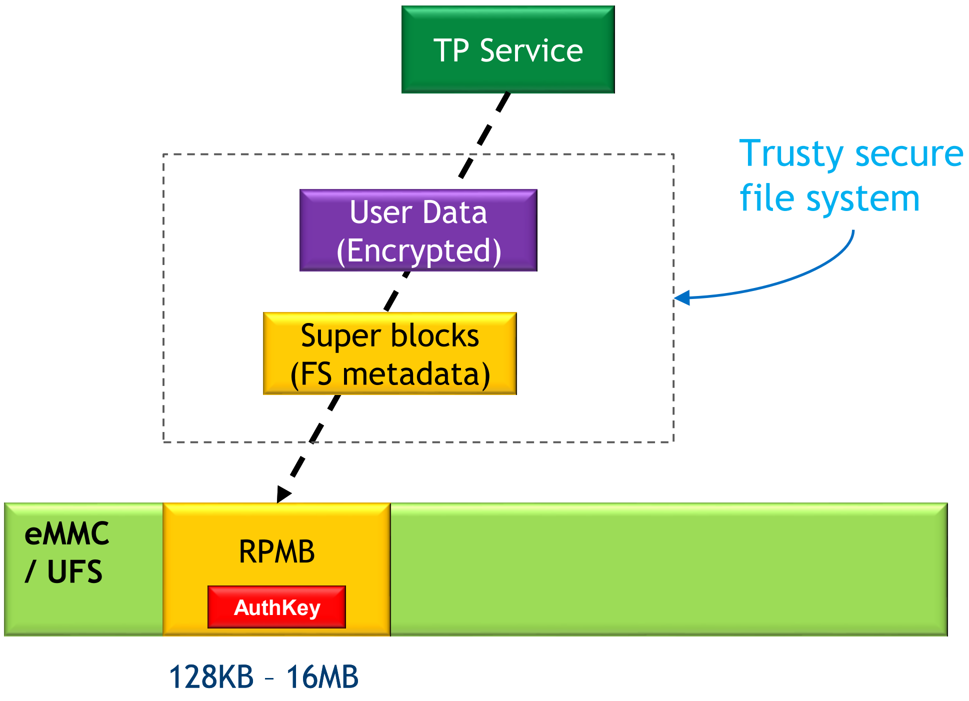
Android Secure Storage (SS)



https://source.android.com/security/trusty/
https://android.googlesource.com/trusty/app/storage/

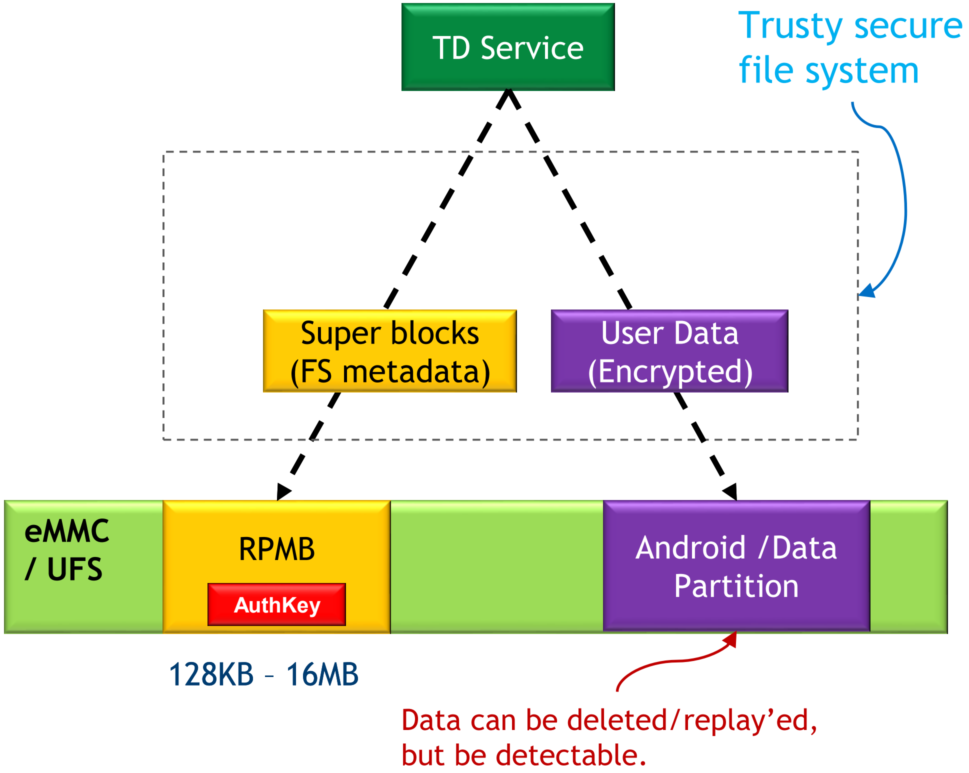
SS/TP : Tamper-Proof Secure storage

1. Secure File System meta-data and user data are all stored in RPMB.
2. Much higher security level of protection – Tamper Resistant!
3. Data survives in Android factory reset (pretty good for storing factory-provisioned key materials)
4. Size constrained; Typically 2MB, depending on eMMC/UFS/NVMe RPMB size in manufacturing.



SS/TD : Tamper-Detection Secure storage

1. Secure File System metadata is stored in RPMB.
2. Support large amount of data.
3. However, the user data (encrypted with hardware-backed encryption key), is stored in Android/Linux-backed file system in ordinary /Data partition.
4. Tamper-Detection (or Tamper-Evident) protection.



Secure Storage Virtualization in ACRN* Hypervisor

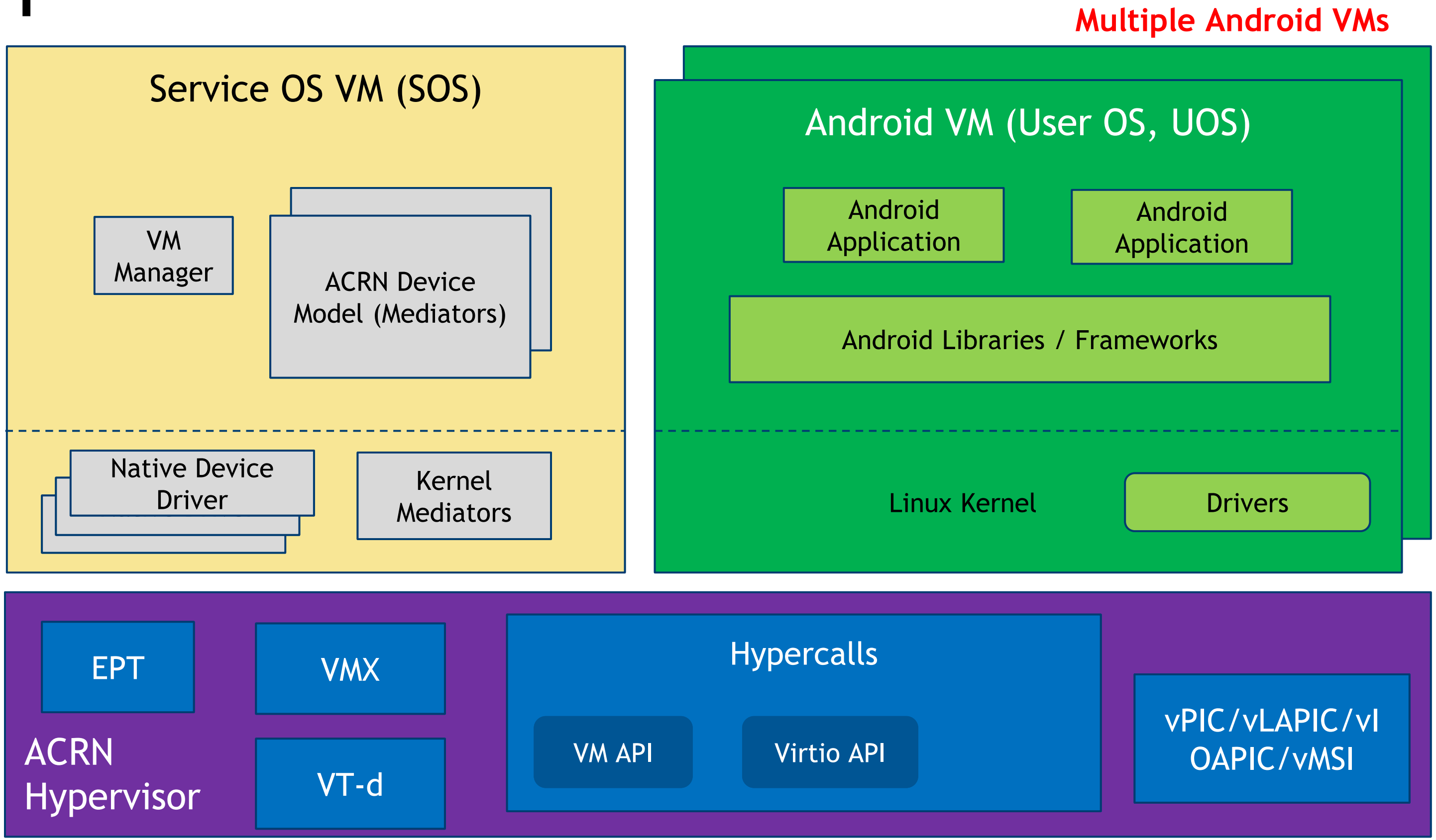
ACORN:
Picture source: https://en.wikipedia.org/wiki/Acorn

*ACRN Hypervisor: https://projectacrn.org/

ACRN Hypervisor Architecture

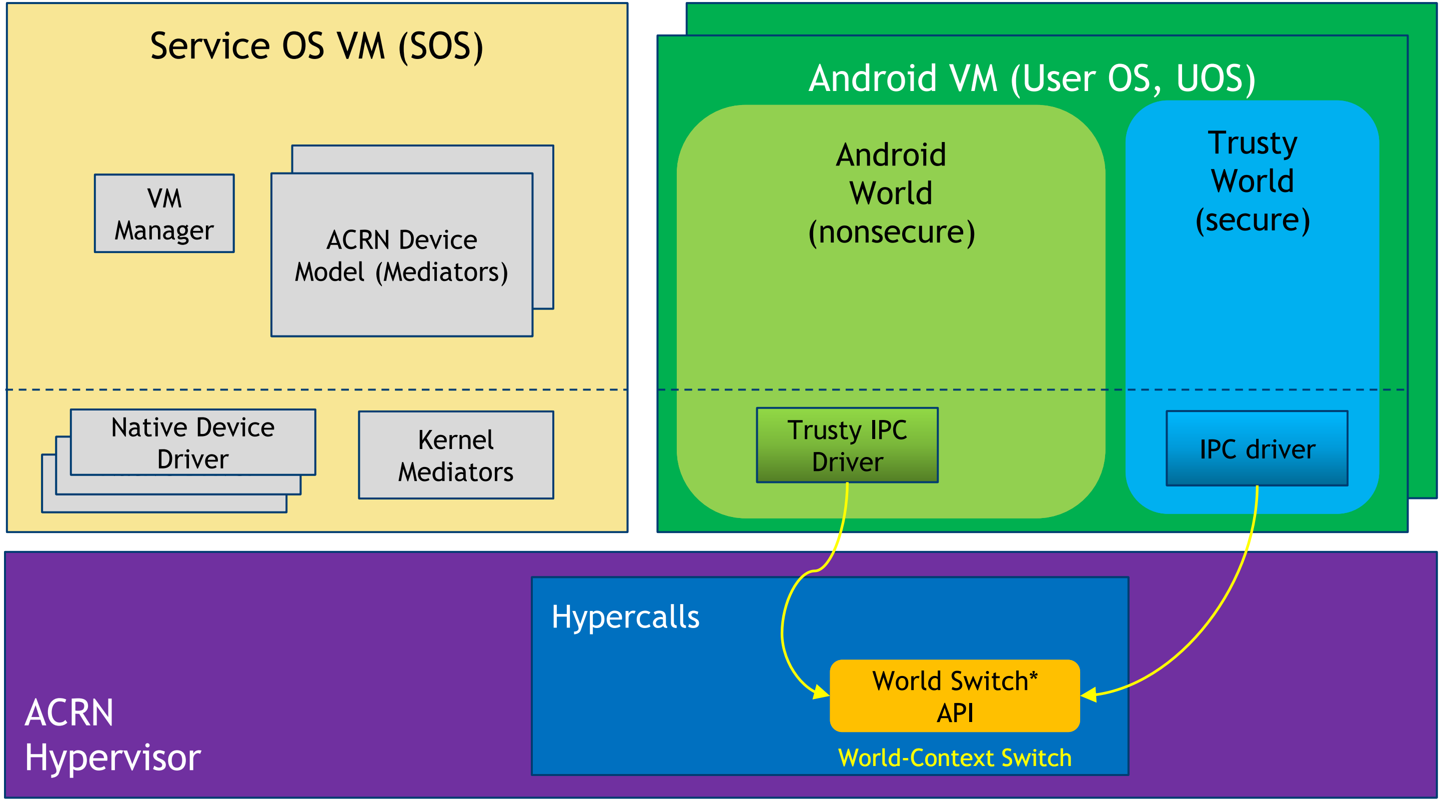
Example Usage: Automotive in-vehicle infotainment or/and entertainment system, can support multiple Android UOS VMs in a single SoC platform.

Note that Service OS is a privileged VM, typically it is a closed system.



Trusty/TEE Isolation in ACRN (One-VM / Two-World)

*ACRN creates only one VM structure per each UOS, but creates two different vCPU context areas to save/restore two worlds' virtual CPU states as per world-switch request from either world.

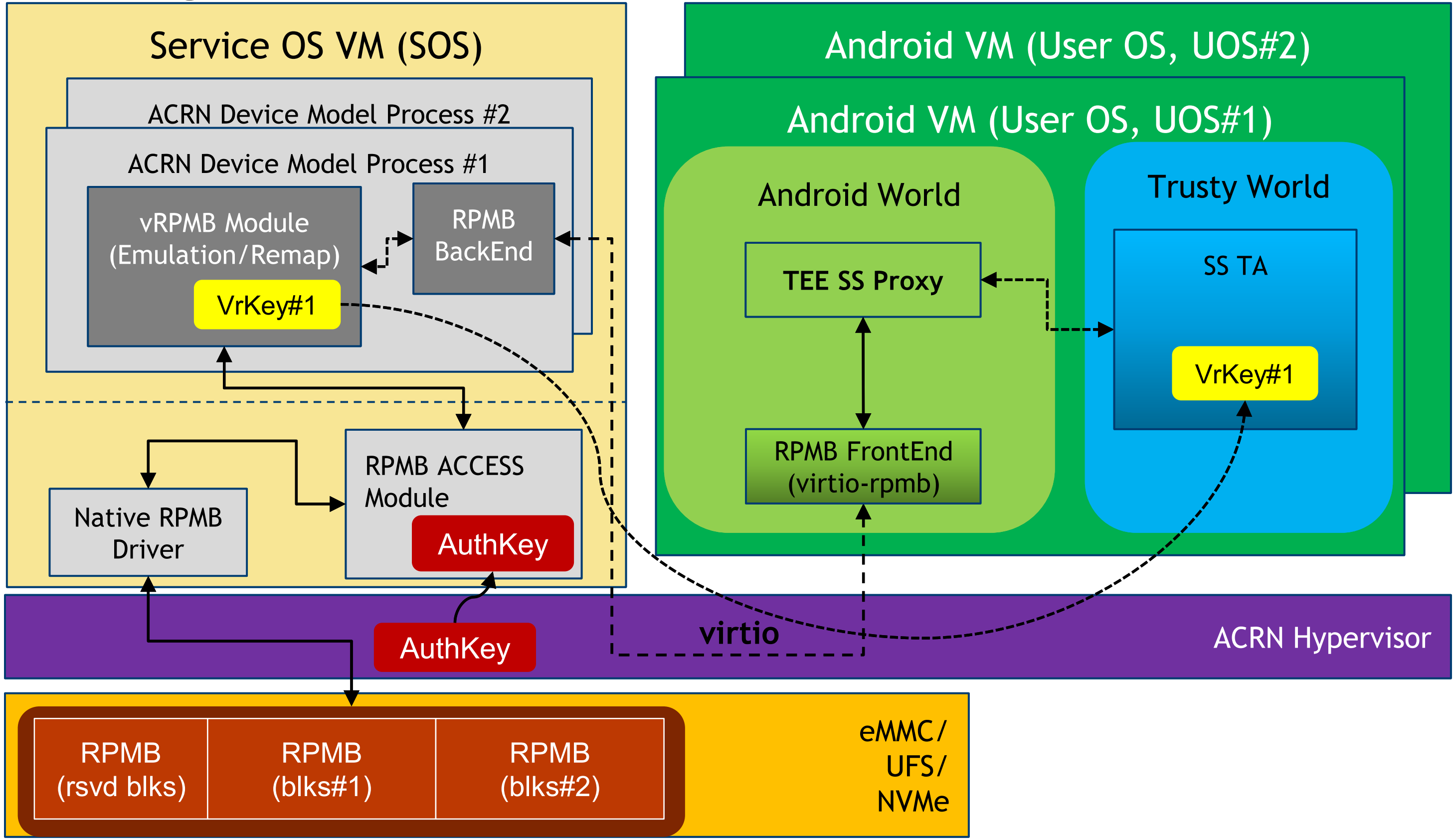


Secure Storage Virtualization

SOS (Service OS) is a closed system and privileged VM.

The VrKey (virtual RPMB Authkey) is generated randomly per UOS boot, and securely distributed it to TEE/Trusty SS TA.

vRPM Module in SOS forwards/remaps vRPMB data/frame to physical RPMB partition.

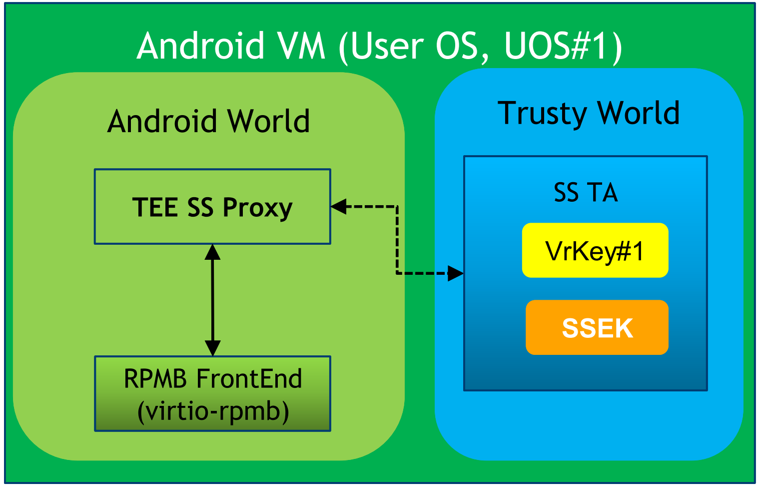


Single RPMB Partition

Secure Storage Virtualization – Confidentiality

Problem:

- How to ensure secure storage data confidentiality for each TEE/Trusty instance per UOS?

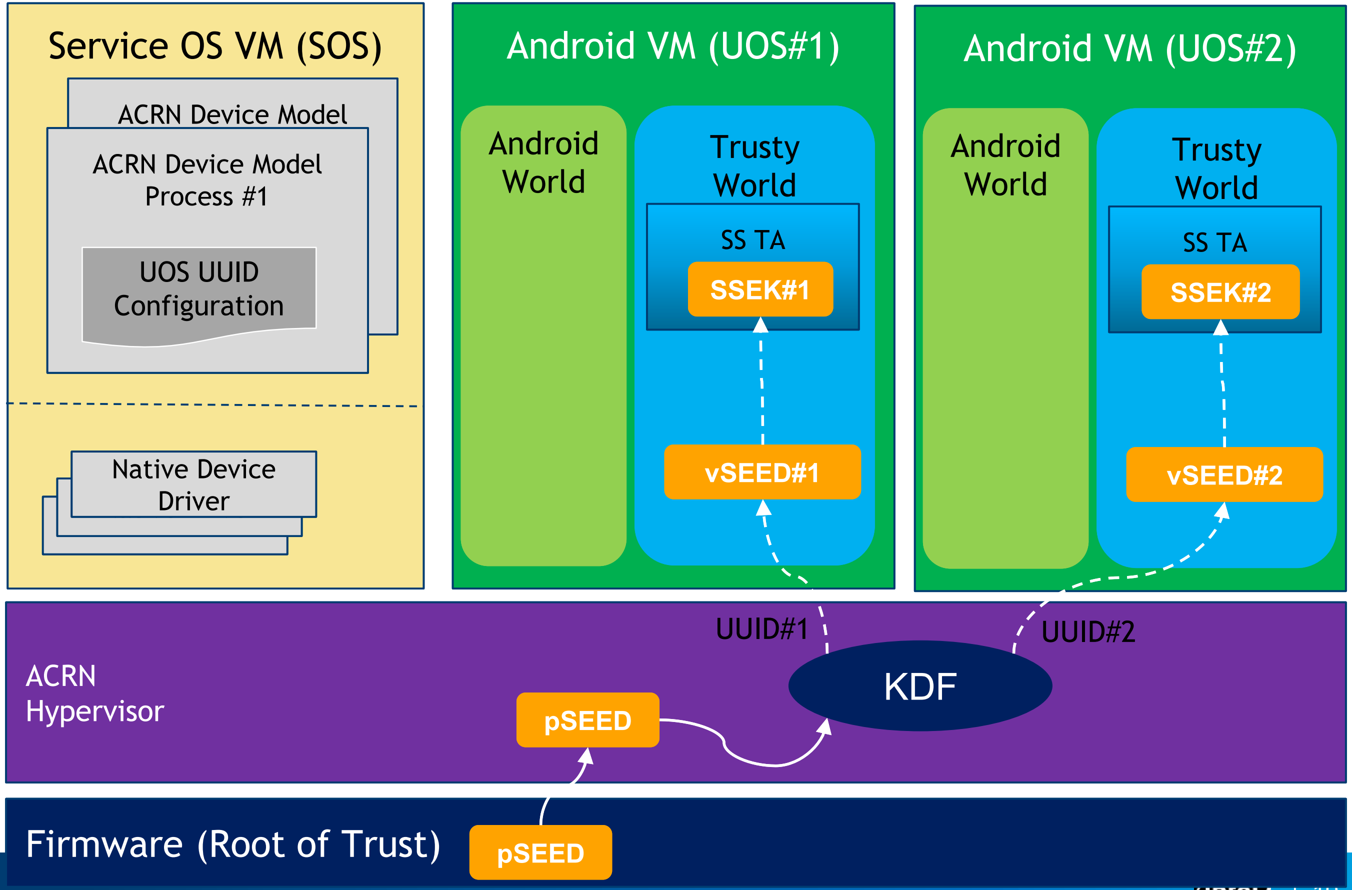


Hence, how to generate ***Secure Storage Encryption Key*** (SSEK) per each TEE/Trusty?

ACRN Hypervisor SEED/SSEK Derivation

RoT firmware generates a Platform SEED (pSEED, unique per platform, 256+ bit)

Hypervisor gets the pSEED, derives VM-SEED (vSEED) for each Trusty/TEE in UOS, and sends it to the associated Trusty/TEE guest instance.



- - - - -▸ Key Derivation

Conclusion and Future Considerations

Conclusion

1. Both Tamper-resistant and Tamper-evident secure storage can be implemented in native Android and multiple virtual Android VMs on ACRN Hypervisor.
2. Both Date Integrity and Confidentiality protection can be achieved.
3. Replay Protection can be achieved for native Android, but for virtual Android on ACRN hypervisor, it relies on the integrity of Service OS (SOS)
    1. SOS is implemented as a closed system, and SOS has no knowledge of secure data encryption key for each virtual Android/Trusty, but,
    2. SOS does have actual physical RPMB key (recording data then replaying it later)
4. The entire solution depends on intact chain of trust (e.g. verified boot)

Future Considerations

1. Enhance security with dedicated RPMB partition per VM/UOS
   - Latest UFS (v3.0) support 4 RPMB paritions with 4 different RPMB Authkeys.
   - NVMe storage supports multiple RPMB partitions as well.
2. Service OS (SOS) application / data integrity protection (e.g. dm-verity)
   - Refer to ACRN security HLD: https://projectacrn.github.io/latest/developer-guides/security-hld.html

Questions?

References

Google/Android Trusty:
https://source.android.com/security/trusty?hl=en-us

Google Trusty Secure Storage:
https://android.googlesource.com/trusty/app/storage/

eMMC Specification (latest: v5.1)
https://www.jedec.org/standards-documents/technology-focus-areas/flash-memory-ssds-ufs-emmc/e-mmc

UFS Specification (latest: v3.0)
https://www.jedec.org/standards-documents/focus/flash/universal-flash-storage-ufs

NVMe Specification:
https://nvmexpress.org/resources/specifications/

ACRN Project:
https://projectacrn.org/
https://projectacrn.github.io/latest/introduction/index.html
https://github.com/projectacrn

Backup Slides

RPMB Key Generation and Programming

RPMB Key generation requirements:

1. Key is tied to hardware unique key (HUK).
2. Key is also bound to eMMC/UFS/NVMe flash storage serial #.

RPMB key programming:

1. Typically firmware is responsible for programing the RPMB Key (in cleartext) into RPMB controller through RPMB key programming interface.
2. Do it once in factory, or just right after eMMC/UFS/NVMe replacement if applicable.
3. Key cannot be changed once it's programmed successfully (OTP FUSED)