CS5231: Systems Security

Lecture 09b: Virtualization and Trusted Computing Environment

Evolution of TEEs

2004
TPM 1.2

2005
TrustZone

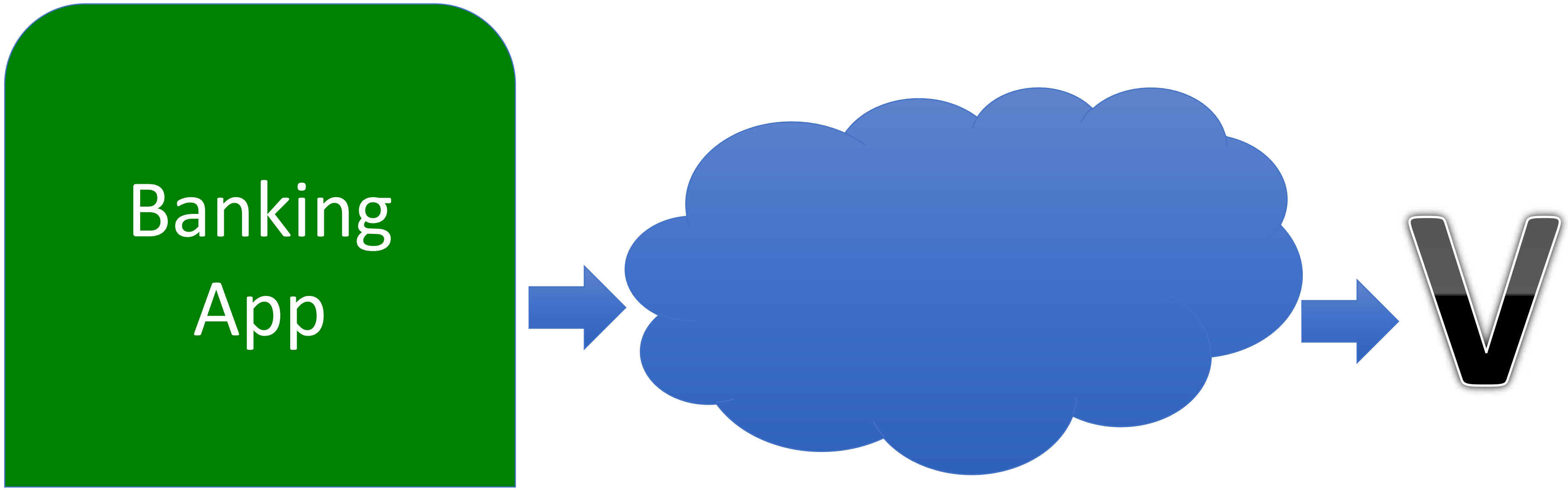
2006
Intel TXT

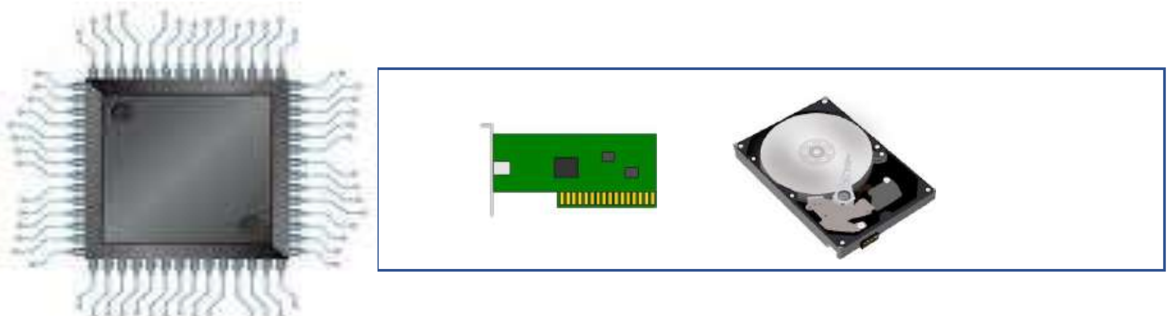
2014
TPM 2.0

2016
Intel SGX

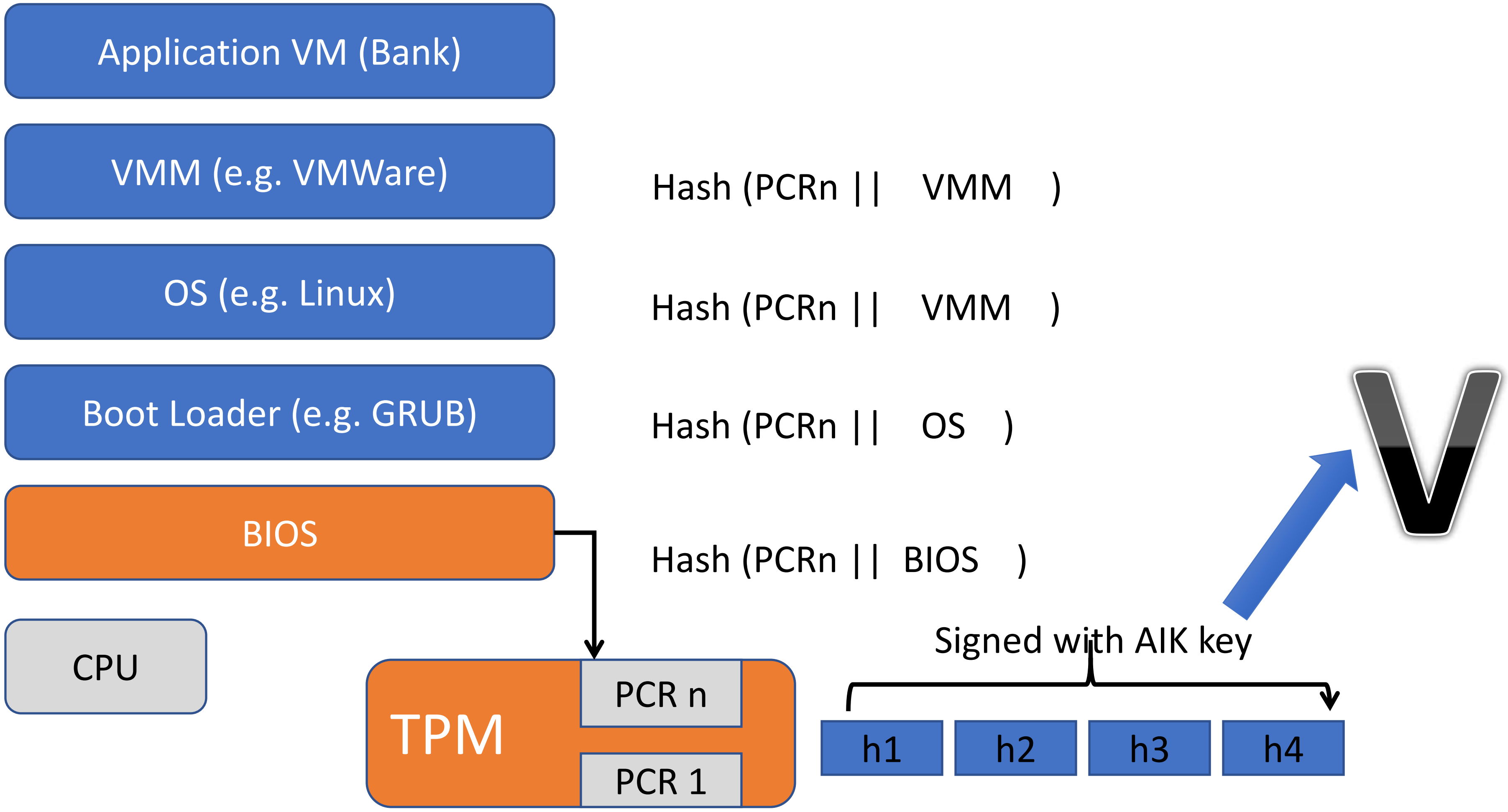
2023
Intel TDX

Trusted Execution Primitives (I): Remote Attestation



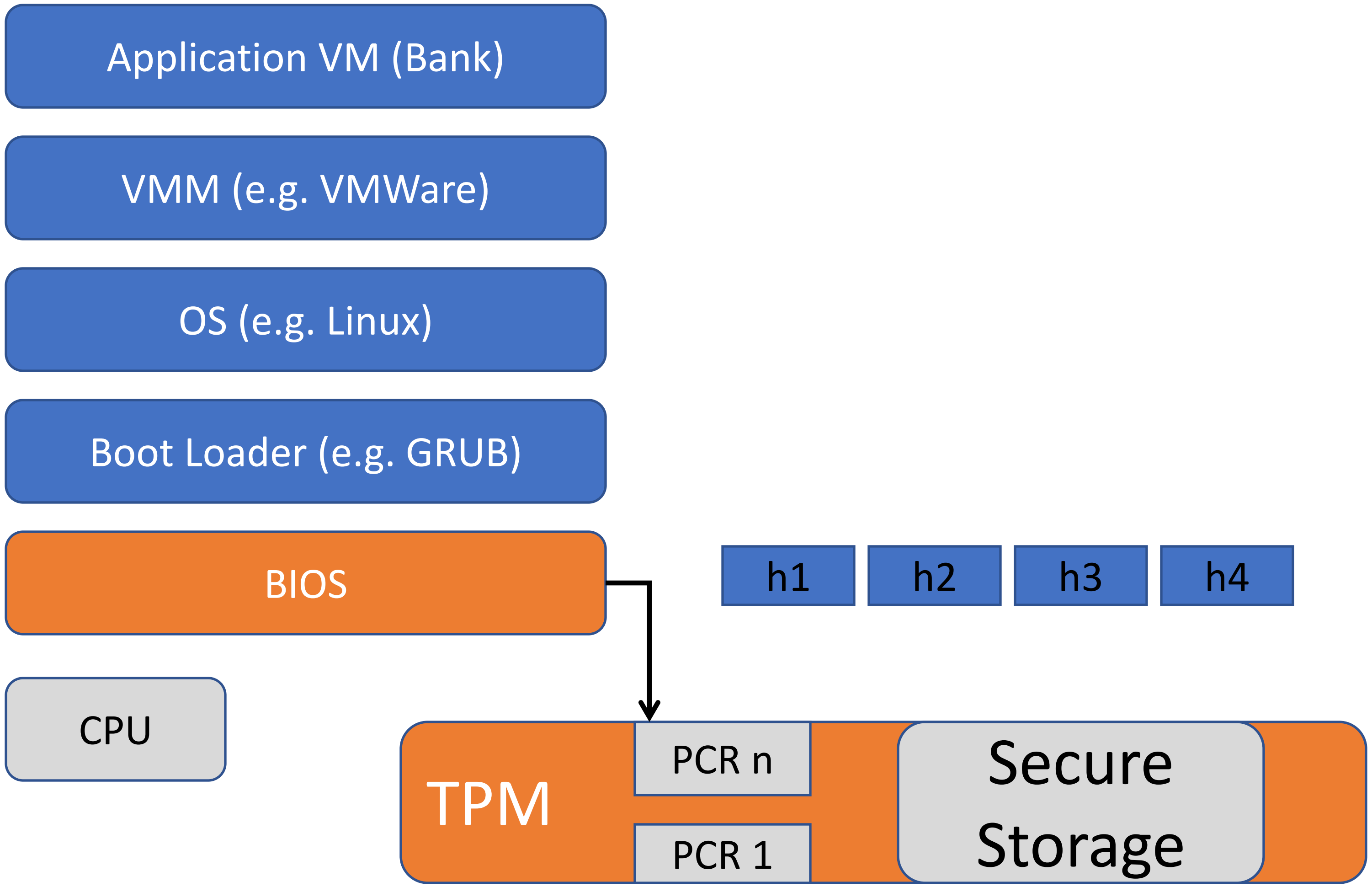


Trusted Execution Primitives (I): Remote Attestation



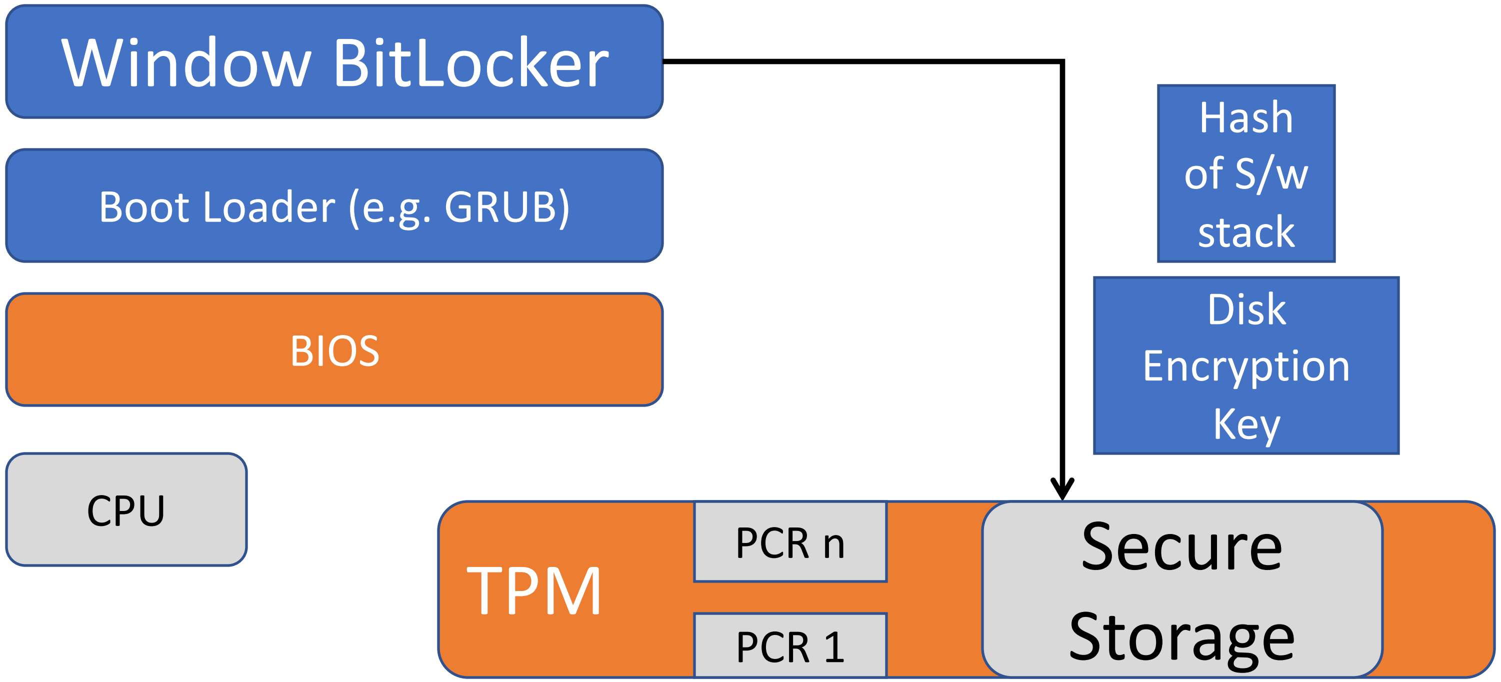
Each TPM has a AIK signing key

Trusted Execution Primitives (II): Trusted Boot

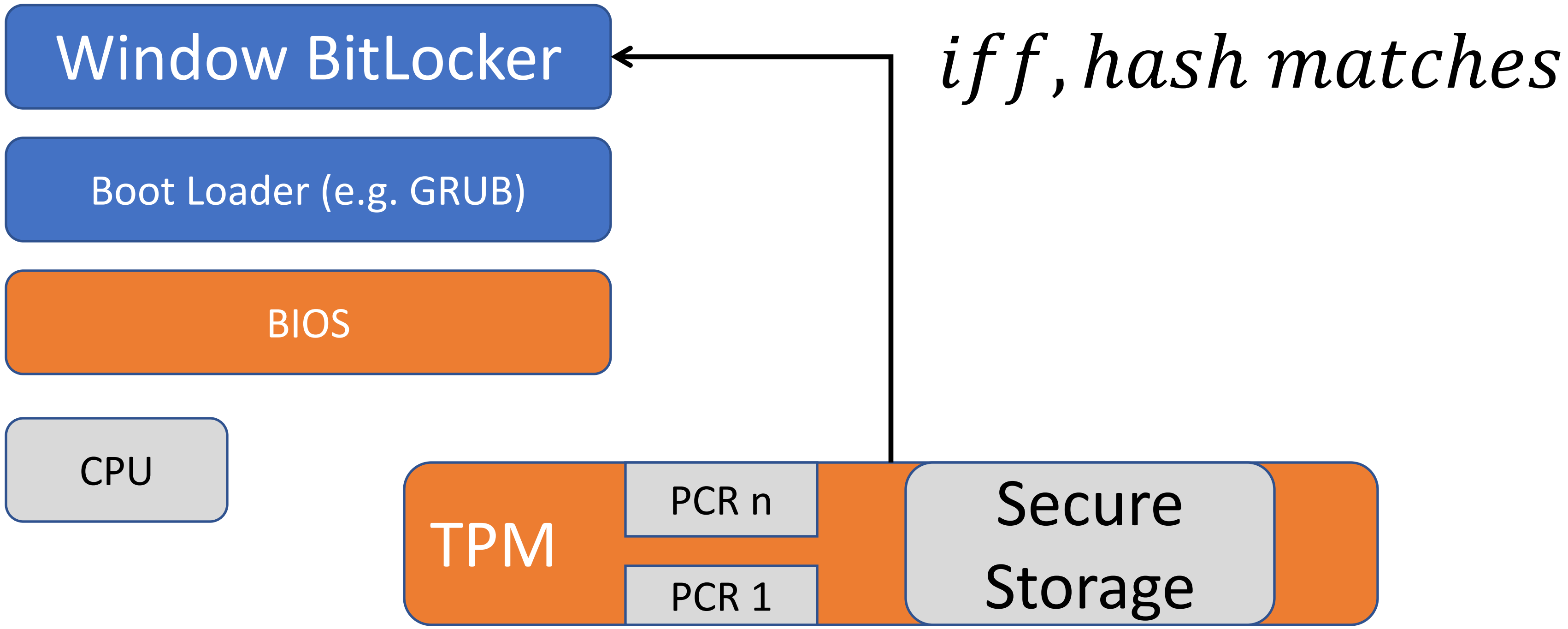


Trusted Execution Primitives(III): Sealing Data

Use TPM Measurements & Secure Storage for Disk Encryption Systems?



Trusted Execution Primitives (III): Unsealing Data



Trusted Execution Primitives (III): Sealed Storage

```
# echo 'Secret!!!' | tpm_sealdata -z -i/proc/self/fd/0
-o./mysecret.blob -p17 -p18 -p19


// assuming PCR's are the same
# tpm_unsealdata ./mysecret.blob
Secret!!!

// assuming PCR's are different
# tpm_unsealdata ./mysecret.blob
error 24: Tspi_Data_Unseal: 0x00000018 - layer=tpm,
code=0018 (24), Wrong PCR value
```

Trusted Computing Primitives (I – II): Static Root-of-Trust

- So far, we've seen SRTM systems
  - Checks / Verification at load time

- Many Applications
- Windows BitLocker
- Linux TrustedGrub (because TXT is too slow!)
- Build your own "secure apps"
  - [eXtensible, Modular Hypervisor Framework](#) (from CMU)

Use Case: Full Volume Encryption

- Encryption at the block level underneath file system
- Everything in the volume is encrypted.
- BitLocker is used by Microsoft since Windows Vista
- BitLocker takes advantage of TPM (TEEs)
  - Top level root key sealed in hardware
  - Root key encrypts disk encryption key, which encrypts sector data
- CPU protects disk encryption key by encrypting it
- CPU releases key only after comparing hash of early (unencrypted) boot files with previous hash

New Generation of Trusted Hardware

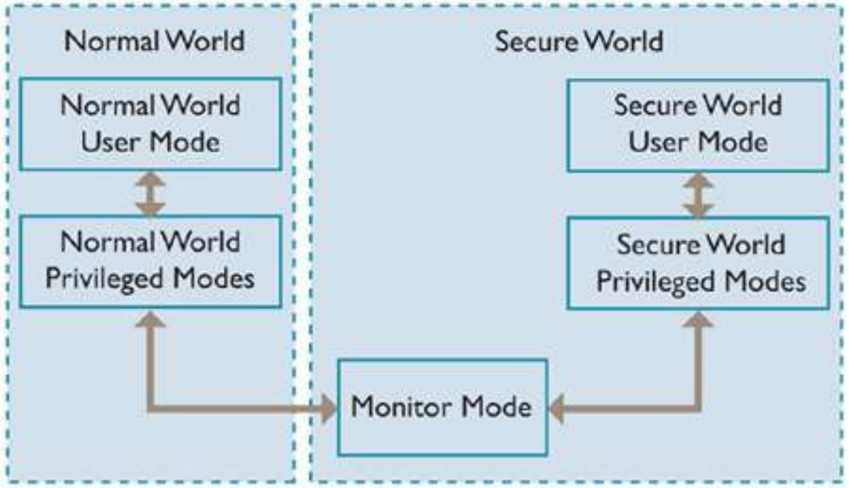
- Arm TrustZone
- Intel SGX
- AMD SEV
- Intel TDX
- Arm Realm

The “root of trust” has to be hardware. You cannot ask a software system to “validate” itself.
Trusted Computing Group

ARM TrustZone

Introduced two basic concepts to the ARM architecture

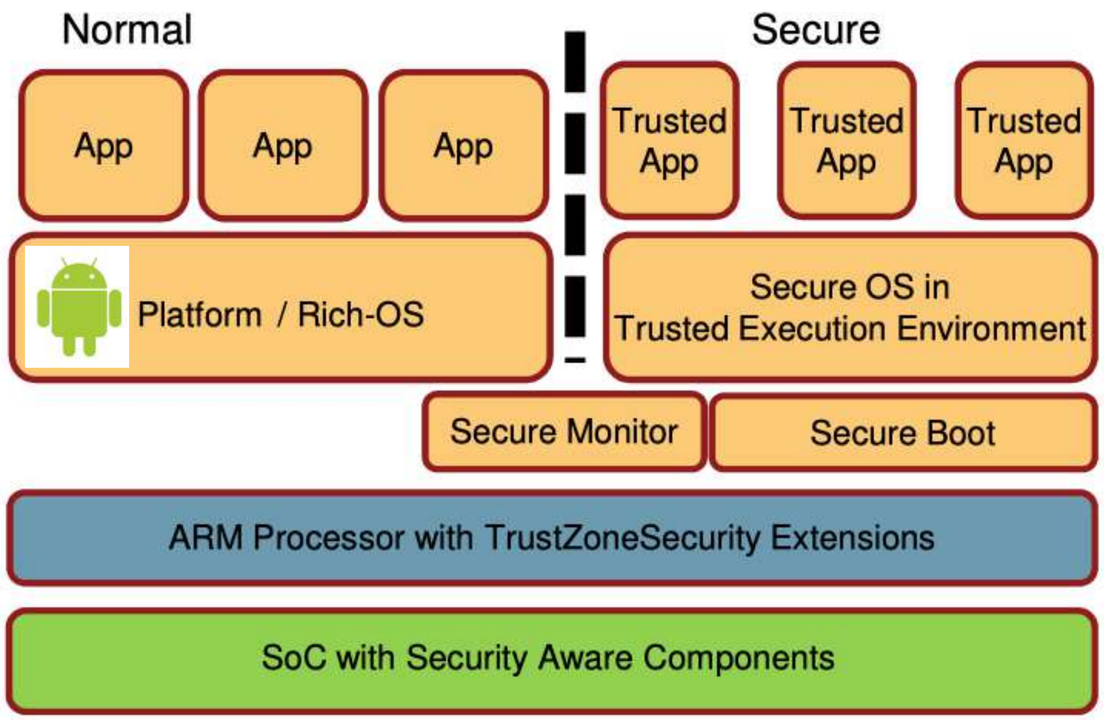
- The ability to tag *system resources* as belonging to either a **Secure** or **Non-Secure** context
- A *virtualization* component which enables a processor to rapidly switch between these contexts



Software View

- Delivers two separate domains, **normal** and **secure**

- **Secure** domain enables a TEE, for security-sensitive applications



Use Cases

- **Samsung Pay**
  - Uses TrustZone to handle payment card information securely
- **Samsung Knox** – mobile enterprise solution
  - KNOX security software runs in the secure world, so it's isolated from the rest of the system
- **Trusted Language Runtime**
  - Using ARM TrustZone to build a trusted language runtime for mobile applications, ASPLOS'14
- Many more …

Additional References

- Demystifying Arm TrustZone: A Comprehensive Survey, https://dl.acm.org/doi/10.1145/3291047
- GlobalPlatform, https://globalplatform.org/specs-library/?filter-committee=tee
- OP-TEE, https://www.op-tee.org/
- Trusty TEE, https://source.android.com/docs/security/features/trusty
- Open-TEE: https://arxiv.org/pdf/1506.07367.pdf
- https://www.arm.com/technologies/trustzone-for-cortex-a
- https://www.arm.com/technologies/trustzone-for-cortex-m

Intel SGX

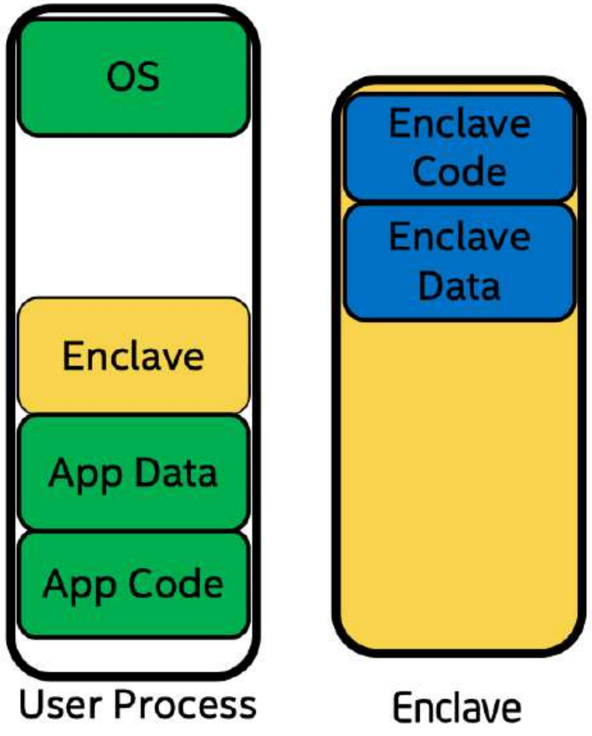
Software Guard Extensions, a set of extensions to the Intel architecture that aims to

- provide **integrity** and **confidentiality** guarantees to **security-sensitive** computation performed
- on an **untrusted** host where all the privileged software (kernel, hypervisor, etc.) is potentially **malicious**

intel works as a Root CA for to verify all the secret keys in the Enclave

Intel SGX

- The key concept behind SGX is an **enclave,** a **TEE** embedded in a **process**
- SGX-enabled processors guarantees two crucial properties:
  - **Isolation**: each enclave's environment is isolated from the *untrusted* software *outside*, as well as other *enclaves*
  - **Attestation**: allowing a *remote* party to *authenticate* the software running inside an enclave



Intel SGX Security Features

Three prominent security features:

- **Confidentiality** for data stored inside an enclave

- **Integrity** of code execution inside an enclave

- **Remote Attestation** of an enclave

Before provisioning sensitive data (e.g., credentials) into the remote enclave, a relying party must assure its **authenticity**

Intel SGX Use Cases

- OSDI’14, Shielding Applications from an Untrusted Cloud with Haven
- OSDI’16, SCONE: Secure Linux Containers with Intel SGX
- SP’18, EnclaveDB: A Secure Database using SGX
- Many more: https://www.intel.com/content/www/us/en/architecture-and-technolog

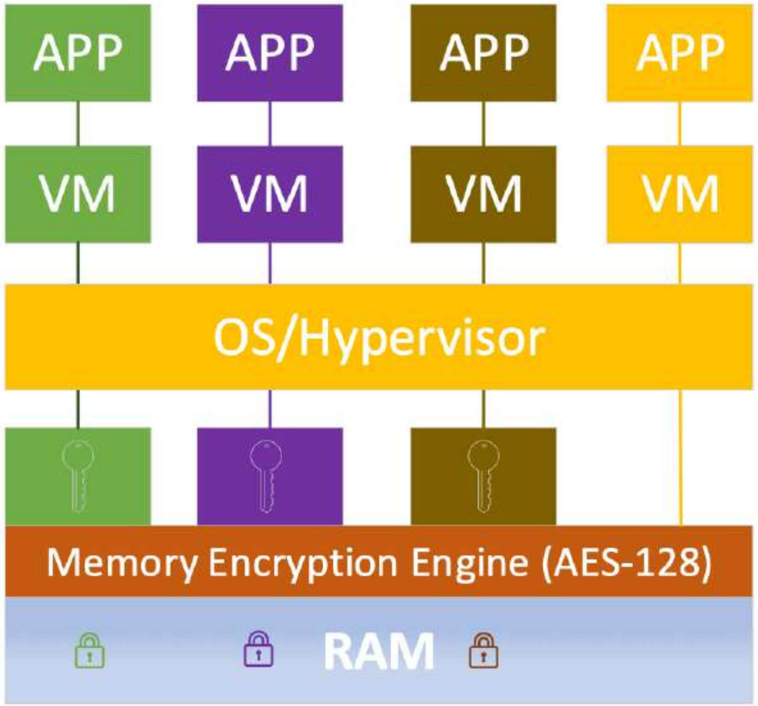
Intel SGX – Additional References

- ISCA’15 Tutorial: https://community.intel.com/legacyfs/online/drupal_files/332680-002.pdf
- Intel SGX Explained: https://eprint.iacr.org/2016/086.pdf
- https://www.intel.sg/content/www/xa/en/architecture-and-technology/software-guard-extensions.html
- https://sgx101.gitbook.io/sgx101/
- Open Enclave SDK: https://openenclave.io/sdk/

AMD SEV and Intel TDX

SEV: Secure Encrypted Virtualization, allows the **memory** of VMs to be **encrypted**

- SEV uses a **unique** memory encryption key for **each** VM
- The encryption of memory pages is completely **transparent** to the **hypervisor** and happens inside **memory controller**.
- Each controller includes a high-performance AES engine that **encrypts** data when it is **written** to **DRAM** and **decrypts** it when **read**.



Credit for Figure: Saeid Mofrad, https://caslab.csl.yale.edu/workshops/hasp2018/HASP18_a9-mofrad_slides.pdf

AMD SEV and Intel TDX

SEV: Remote Attestation of the confidential VM

- SEV calculates a **signature** of the memory contents
- The VM owner uses this signature to **attest** whether the memory was encrypted correctly by the firmware

Intel TDX: Trusted Domain Extensions

- Intel's similar solution to provide a **confidential VM** on untrusted hosts

Credit for Figure: Saeid Mofrad, https://caslab.csl.yale.edu/workshops/hasp2018/HASP18_a9-mofrad_slides.pdf

Use Cases of SEV and TDX

- Shared computing infrastructure, e.g., public clouds
  - VMs are hosted on remote servers which are not under the control of the VMs' owners
- Secure Multi-Party Computation
  - Each party has sensitive data
  - They want to do an analysis on their combined data
  - Such analysis can be perform inside the **confidential** VM
- Many more …

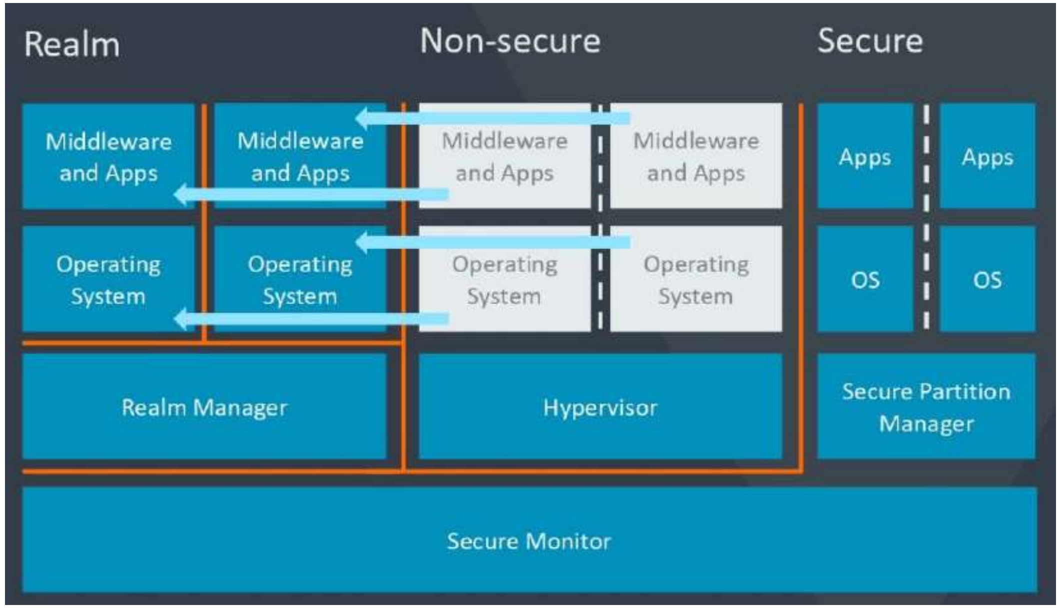
Additional References

- AMD SEV: https://developer.amd.com/sev/
- https://github.com/AMDESE/AMDSEV

- Intel TDX: https://www.intel.com/content/www/us/en/developer/articles/technical/intel-trust-domain-extensions.html
- https://github.com/intel/tdx-tools

ARM Realm

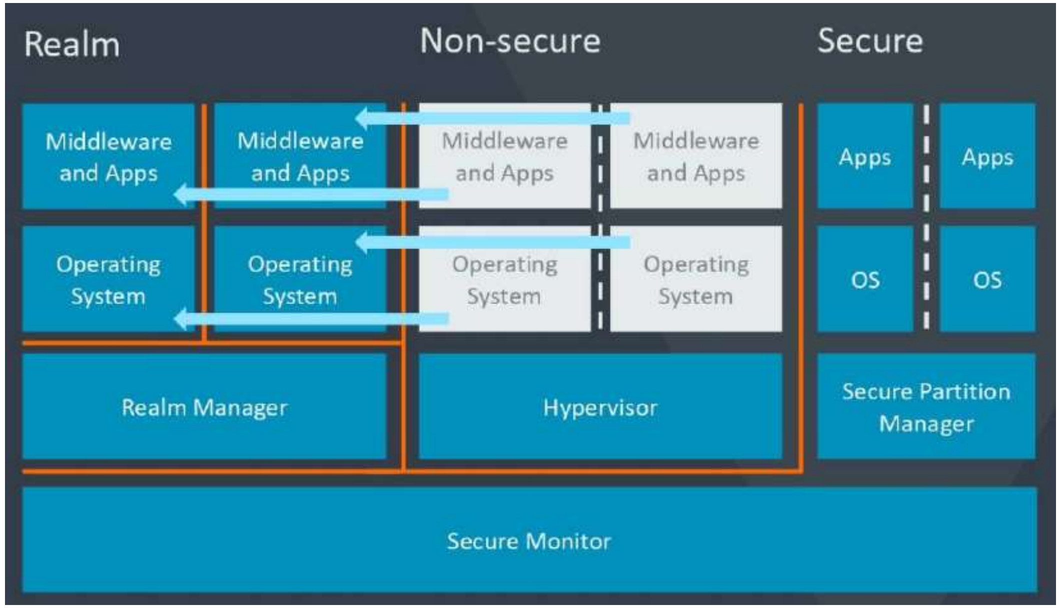
- A new class of **attestable** isolation environment for modern workloads like VMs and containers
- Realms are isolated from the existing Normal and Secure worlds that we have in TrustZone
- Realm excludes **privileged** software (kernel and hypervisor) in the trusted computing base



ARM Realm

ARM's efforts to provide an **isolated environment** for modern workloads like VMs and containers

- Confidentiality
- Integrity
- Attestation
- Minimized TCB (excluding *privileged* software like kernel and hypervisor)



Additional References

- Arm Architecture Security Features: https://www.arm.com/architecture/security-features
- Arm Confidential Computing Architecture: https://www.arm.com/architecture/security-features/arm-confidential-compute-architecture

Confidential Computing

For sensitive and regulated data,

Protecting **data-at-rest**: encryption

Protecting **data-in-transit**: TLS

Protecting **data-in-use:**

- Homomorphic encryption --> impractical
- Secure multi-party computation --> impractical
- Confidential Computing -> a new paradigm of computing

Confidential Computing

- Confidential Computing protects **data in use** by performing computation in a hardware-based, **attested** Trusted Execution Environment.
- These secure and isolated environments prevent **unauthorized** access or **modification** of applications and data while in use, thereby increasing the security assurances for organizations that manage sensitive and regulated data.

Confidential Computing

Enabling trusted hardware:

- Intel SGX
- AMD SEV
- Intel TDX
- Arm Realm

More info can be found at the Confidential Computing Consortium, https://confidentialcomputing.io/