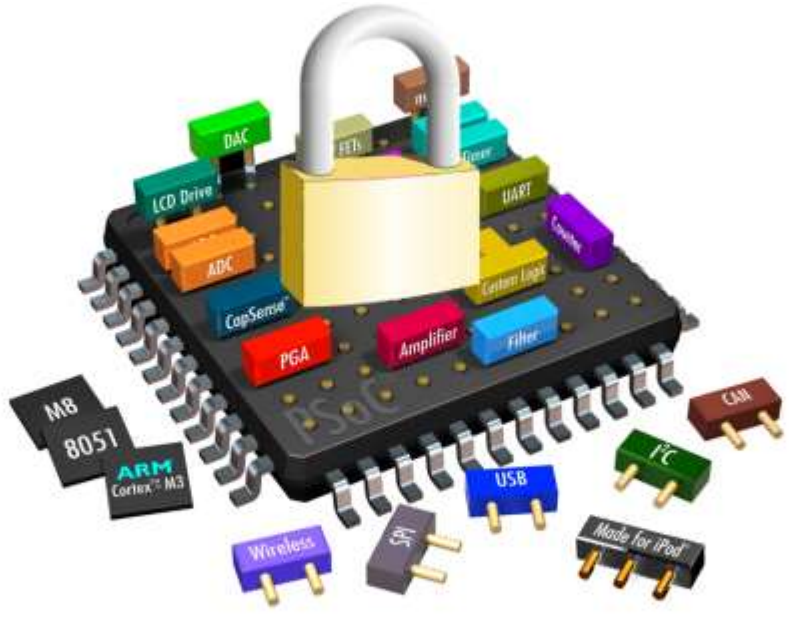
Hardware Security

Trusted Execution Environment



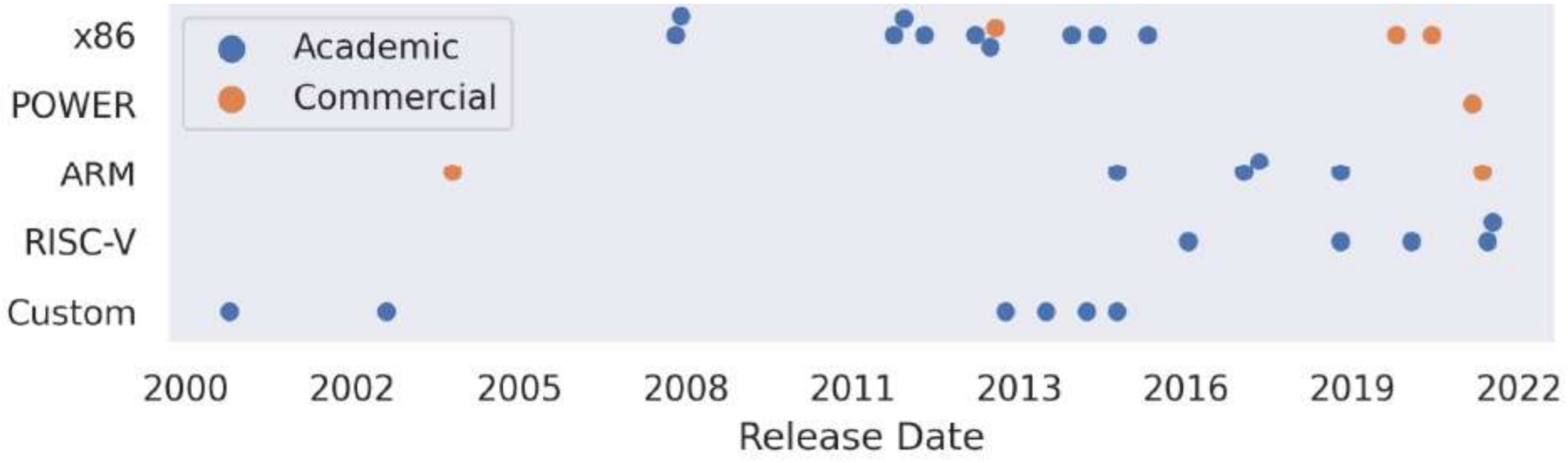
Trusted Execution Environment

- A Trusted Execution Environment (TEE) is a secure area within a processor that ensures confidentiality and integrity of code and data loaded into it. TEEs provide isolated execution of trusted software components, protecting them from the rest of the system.
  - **Verifiable launch** of the execution environment for the sensitive code and data so that a remote entity can ensure that it was set up correctly.
  - **Run-time isolation** to protect the confidentiality and integrity of sensitive code and data.
  - **Trusted IO** to enable secure access to peripherals and accelerators.
  - **Secure storage** for TEE data that must be stored persistently and made available only to authorized entities at a later point in time.

Why TEEs Matter

- Protect sensitive data against co-located attackers
- Enable secure computation in untrusted environments
- Support for multi-tenant computing platforms
- Reduce the trusted computing base (TCB)

Evolution of TEEs



- The timeline shows the evolution of TEEs across different instruction set architectures.
- We observe a significant increase in TEE proposals in recent years, spanning x86, ARM, POWER, and RISC-V architectures, reflecting growing industry and academic interest in hardware security.

Terminology

1

Enclave

A TEE instance that provides an isolated execution environment (Intel calls these "enclaves," AMD uses "Secure Encrypted VMs," ARM CCA uses "Realms" or "trustlets")

2

TCB

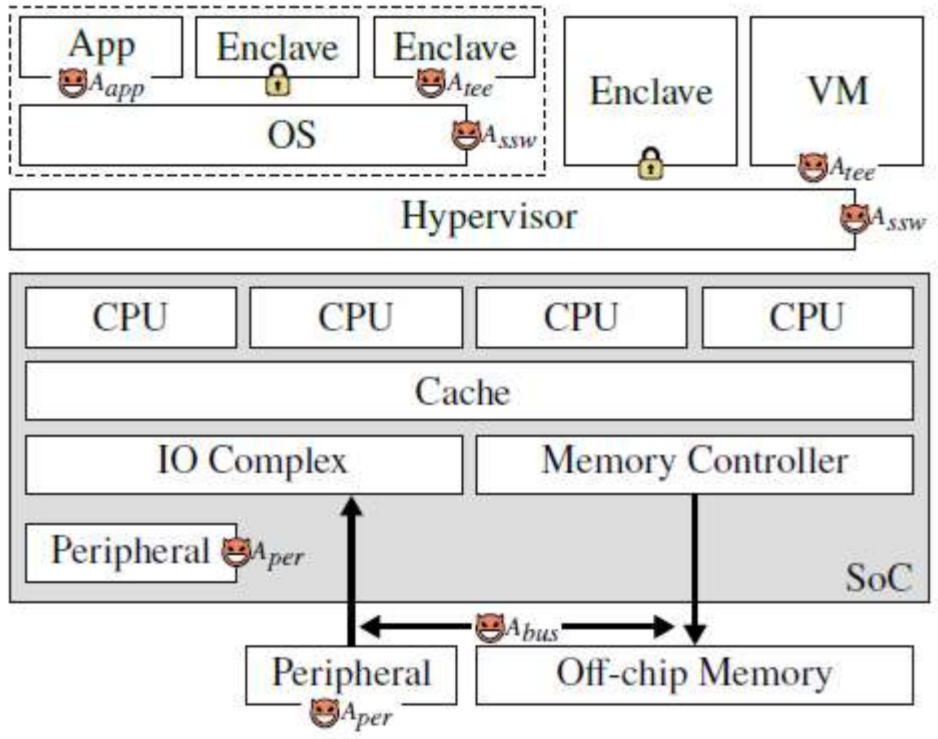
Trusted Computing Base - all hardware and software components that must be trusted for the security of the system

3

TEE

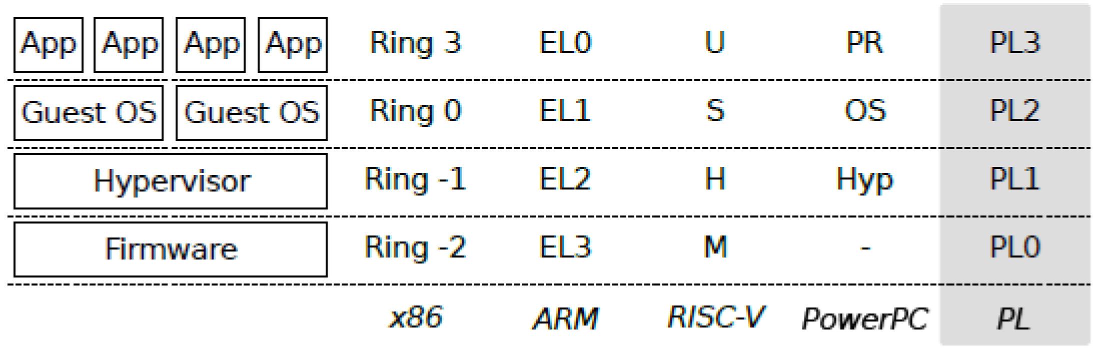
The entire architecture that enables the creation and operation of enclaves

System Model



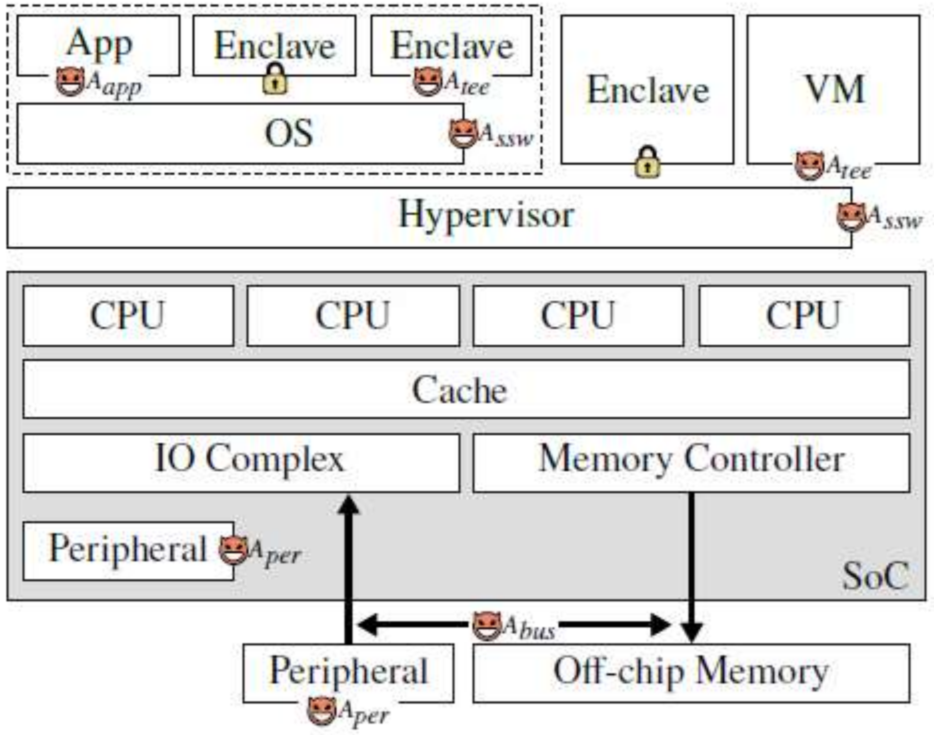
Most TEE solutions target a modern computing platform consisting of a System-on-Chip (SoC) with off-chip memory and peripherals. The SoC contains cores, caches, and fabric connecting to memory controllers and IO complex.

The software stack typically includes an OS and userspace applications, and in virtualized environments, a hypervisor with multiple VMs.



App / App / App / App	Ring 3	EL0	U	PR	PL3
Guest OS / Guest OS	Ring 0	EL1	S	OS	PL2
Hypervisor	Ring -1	EL2	H	Hyp	PL1
Firmware	Ring -2	EL3	M	-	PL0
	x86	*ARM*	*RISC-V*	*PowerPC*	*PL*

Adversary Model



- Co-located Enclave (A_{tee})
  - Adversary controlling other enclaves on the same platform, relevant in multi-tenant environments
- Unprivileged Software (A_{app})
  - Adversary running unprivileged software at the same privilege level as the victim enclave
- System Software (A_{ssw})
  - Adversary controlling system management software like OS or hypervisor
- Peripheral (A_{per})
  - Adversary controlling untrusted peripherals that can launch nefarious IO transactions
- Fabric (A_{bus})
  - Adversary with physical access to intercept off-chip communications
- Startup (A_{boot})
  - Adversary controlling system boot process and configuration