How to Elevate the Security Level of RISC-V Based SoC Designs with a RISC-V Based Root-of-Trust

July 18, 2025

Samuel Chiang

Rambus

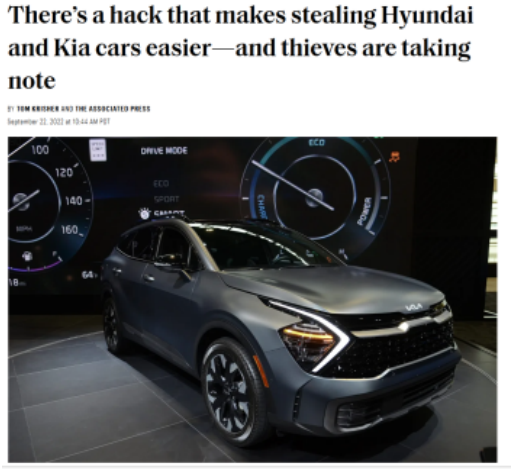
Connected Devices + Vehicles Today are at Ever Greater Risk

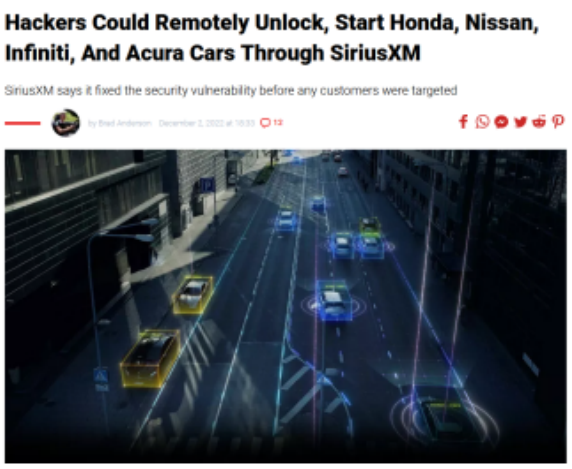
Hangzhou Xiongmai Technology
4.3M Cameras recalled

San Francisco Muni
All rides free for two days!

Dynamic Network Services (Dyn)
100,000 infected devices
Blockage of >1,200 websites

Connect it to the Internet, someone will try to hack it

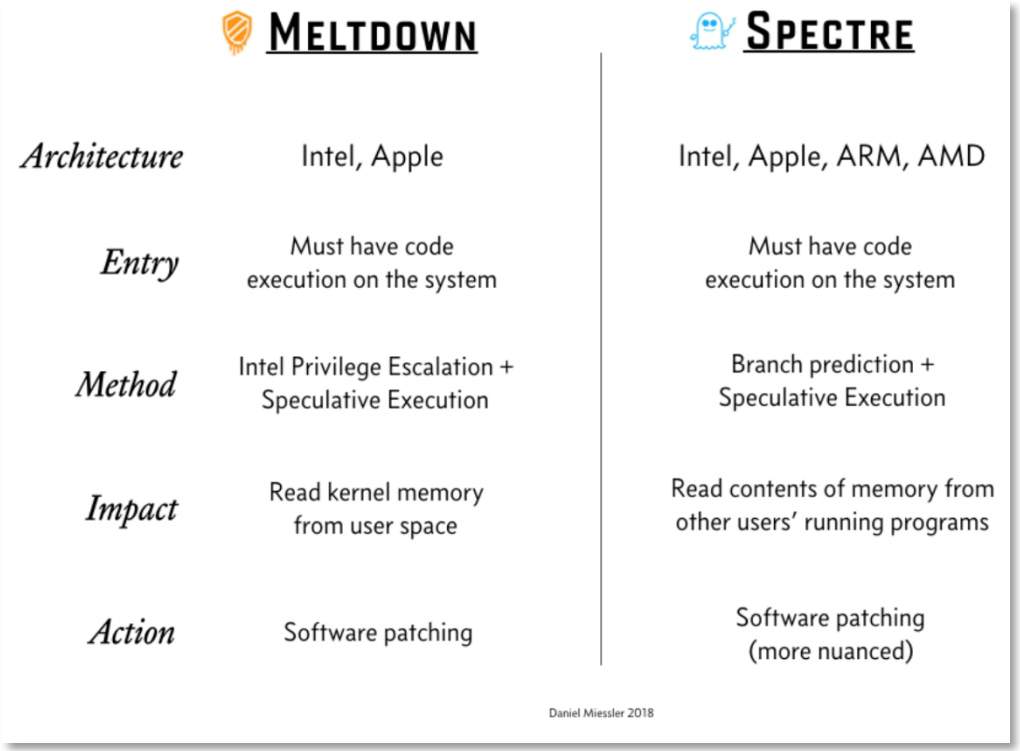




Meltdown and Spectre: Running to Keep Up

Meltdown and Spectre provide a dramatic example of something security experts have known for years:

Complex systems, like modern CPUs, that are designed for performance and not security are inherently weak against attackers

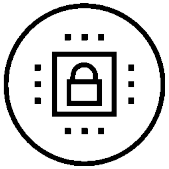


Security First: Implementing Trust by Design in Silicon

Design Freedom

- Root of trust designed from the bottom up for security
- Control all implementation starting with open RISC-V Instruction Set Architecture

Siloed



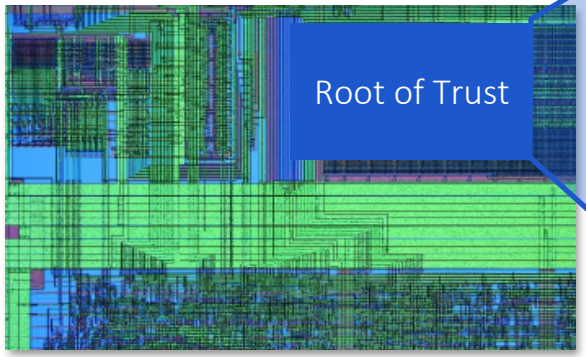
- Separate general and secure processing
- Optimize independently for performance and security

Layered Security

- Strongest security enforced in hardware at inner layer
- Outer layers are more flexible, but less trusted

Why a Root of Trust: Providing Security and Crypto Services

- A Root of Trust provides a trusted foundation that the SoC & applications can use to build their own protection
- Root of Trust products are expected to provide robust Security and Crypto services to the SoC and applications

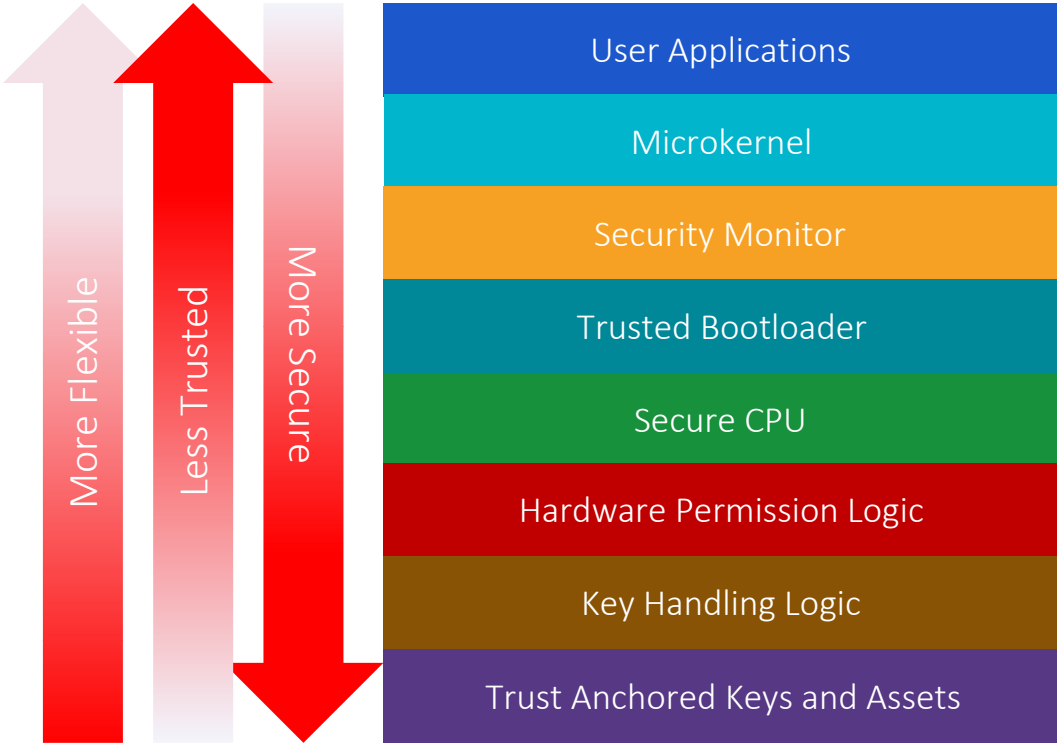


Secure Functionality:

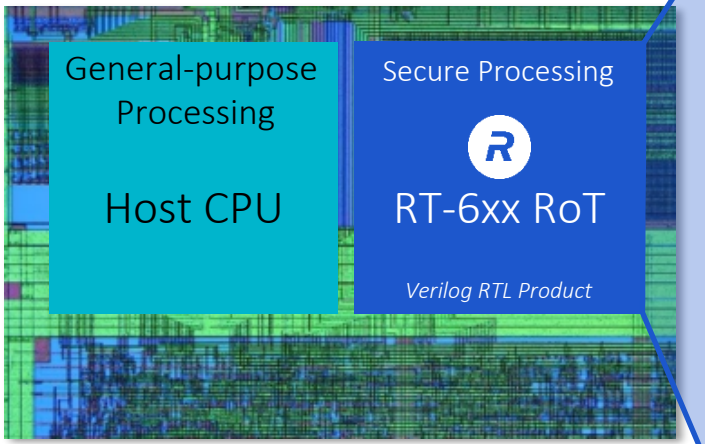
- Secure boot
- Secure firmware update
- Authentication
- Attestation
- Secure data storage
- Secure key storage
- Device personalization
- Key and data provisioning
- User data privacy
- Secure communication
- Runtime integrity checking
- Cryptographic acceleration
- Secure protocol implementation
- Secure debug
- Feature/configuration/SKU management

Defense in Depth Achieved Through Layered Security

- Why a Layered Security Approach?
  - Attack surfaces are large
  - Attack only needs to break weakest link
  - No single point security implementation is resistant to all security attacks
- Markets that adopted separate enclave:
  - Automotive - eHSM, AUTOSAR, EVITA
  - Defense - Root of Trust, Key Management
  - Mobile - eUiCC, eSE, for payment, etc.
  - Video - DRM with isolated video path
  - AI/ML - Protect weights, inference models
  - IoT - Key Management, Authentication
- Solution:
  - Root of Trust firewalled from host CPU
  - Harden security critical operations



RT-6xx: “Secure Island in Silicon”



A secure HW Root of Trust that provides a foundation for security throughout the SoC

Rambus CryptoManager Root of Trust (CMRT)

Custom Secure RISC-V CPU

Private Memory

Crypto Accelerators
(AES, SHA, RSA, ECC, TRNG, others…)

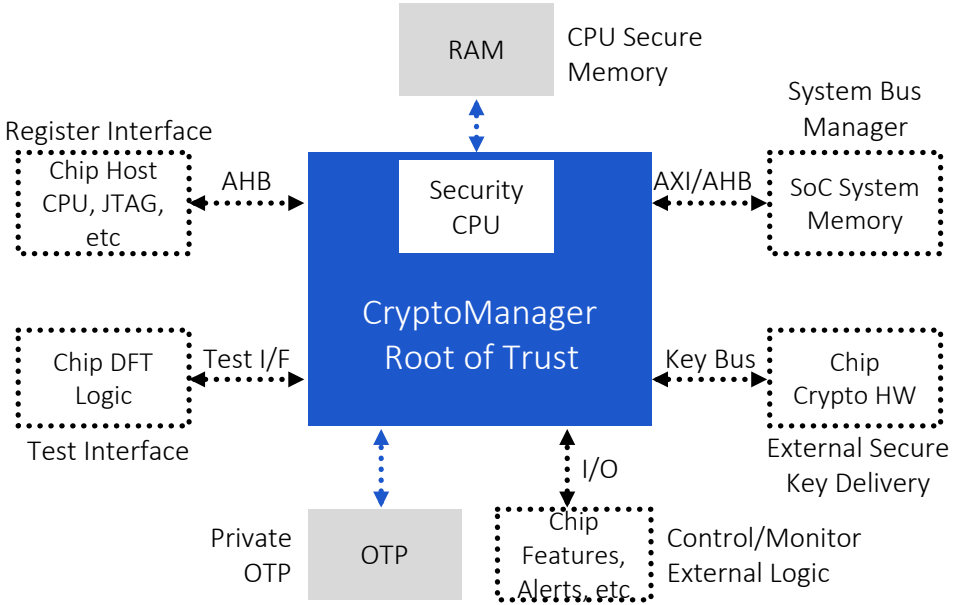
OTP Interface

Secure Functionality:

- Secure boot
- Secure firmware update
- Authentication
- Attestation
- Secure data storage
- Secure key storage
- Device personalization
- Key and data provisioning
- User data privacy
- Secure communication
- Runtime integrity checking
- Cryptographic acceleration
- Secure protocol implementation
- Secure debug
- Feature/configuration/SKU management

CryptoManager Root of Trust (CMRT) RT-6xx series

- A hardware IP core that can be integrated into a wide range of semiconductor devices
- A secure root of trust that provides a foundation for security throughout a system
- A secure location that stores and manages security assets such as keys
- A programmable security enclave based on a custom RISC-V CPU, with a complete secure firmware stack
- A hardware block that provides security and cryptographic services to the rest of the system

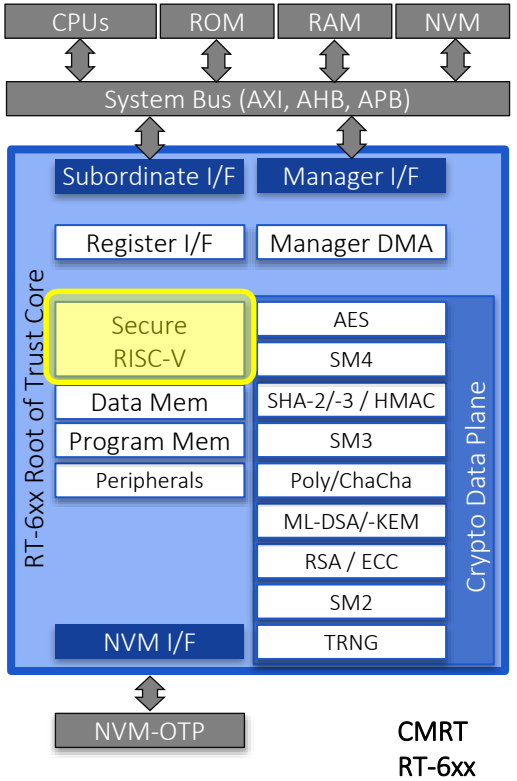


Design Philosophy for CMRT

- **Overall approach**
  - Design for security from the ground up, and avoid compromises inherent in adding security to existing component or system
  - Limit complexity and keep the attack surface small. Avoid optimizations for performance or other goals not related to security
  - Partition security-related processing from general-purpose processing
- **Custom RISC-V security CPU**
  - Start with a clean sheet of paper wherever possible. Build a custom CPU specifically for security applications. This enables adding security functionality like DPA protection in execution stages of the CPU pipeline, shadow call stack logic, glitch protection, and so on
  - Custom developed CPU can be audited in detail
- **Principle of least privilege**
  - Function performed at lowest allowed privilege
  - Entities are only granted minimal set of required privileges
- **Defense in depth**
  - Redundant (layered) protections to eliminate single point of failures
  - Active monitoring for security breaches
- **Future proofing** by quantum safe boot flow

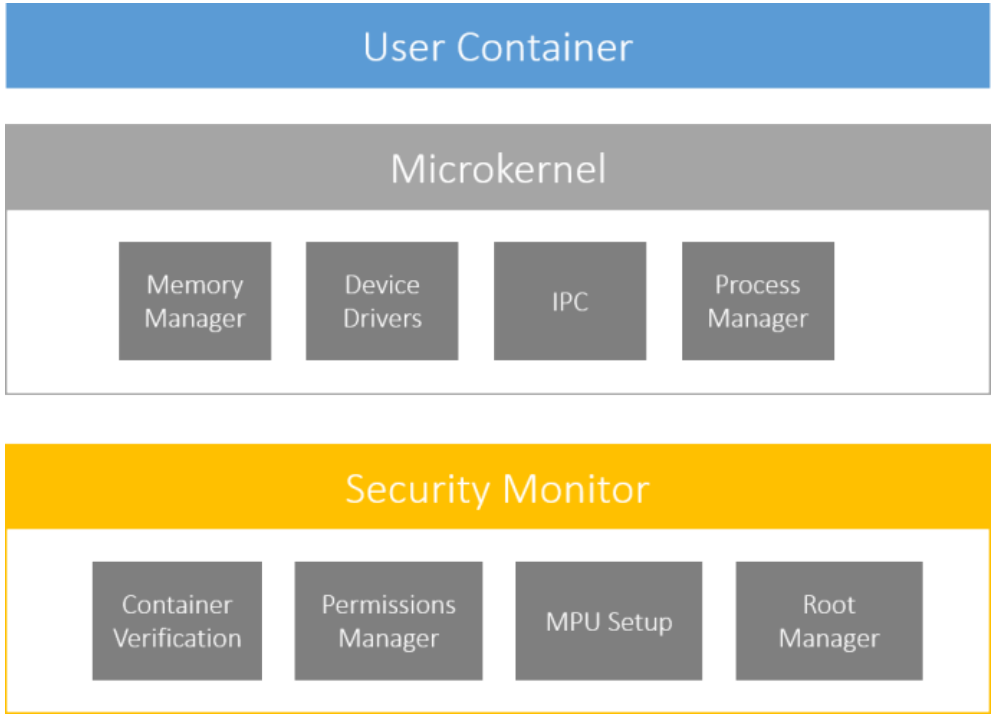
CMRT Hardware Architecture

- Security-optimized multi-stage 32-bit RISC-V RV32I based CPU
  - Custom Rambus design, simple to verify and audit for security
  - Privilege levels: machine, supervisor, user
- MPU
  - Sets regions of memory for access for one or more privilege levels (machine, supervisor or user) and access type (R,W,X)
  - MPU registers can be “locked” until the next CMRT reset
- Security Features (partial list, some optional):
  - Self-contained secure boot, starting with first-stage boot ROM synthesized into gates
  - Glitch protection
  - Memory Protection Unit locked at boot time by secure code
  - Private SRAM
  - CPU bus isolated from secure key bus
  - Call stack protection
  - Side-channel resistant execution pipeline

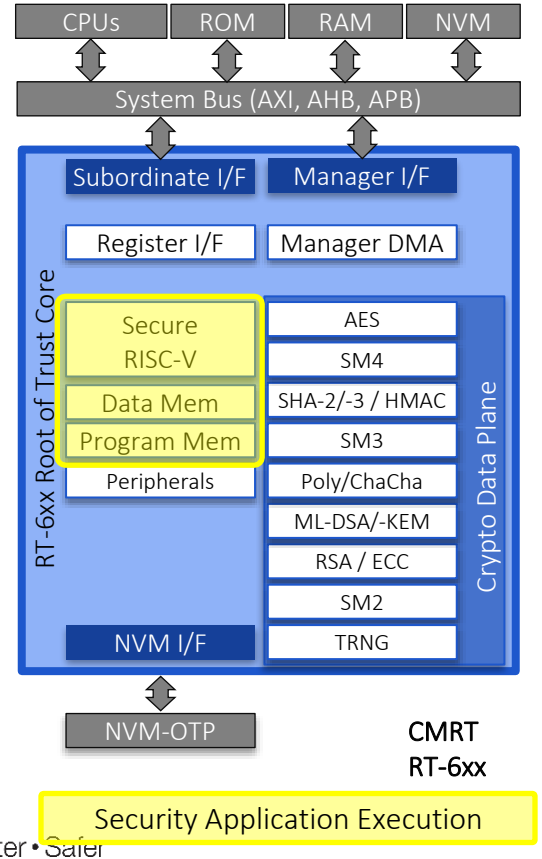


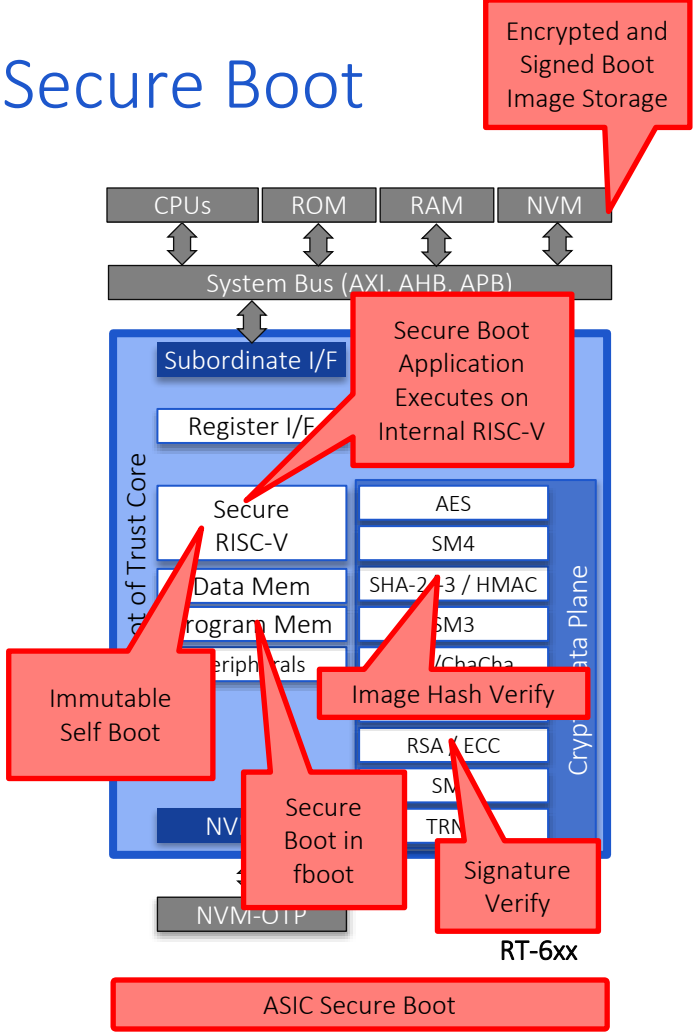
CMRT Software Architecture

- The CMRT uses a layered security model for software. It uses the hardware-enforced privilege levels of the RISC-V ISA for separation of data between layers.
- User level: User-written secure applications (containers)
- Supervisor level: Zephyr-based microkernel customized for security application
- Machine level: Security monitor oversees microkernel and containers and their interaction with hardware
- Future proofing by included quantum safe boot flow

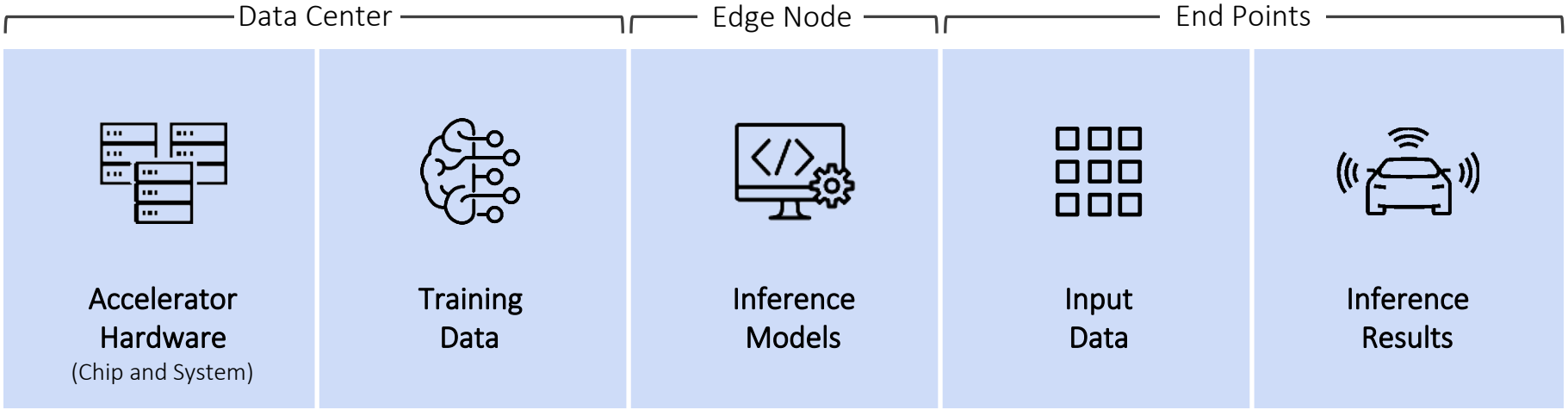


Secure Application Execution and Secure Boot





Vertical Use Case #1: Protecting Machine Learning Assets



Threats to AI Assets:

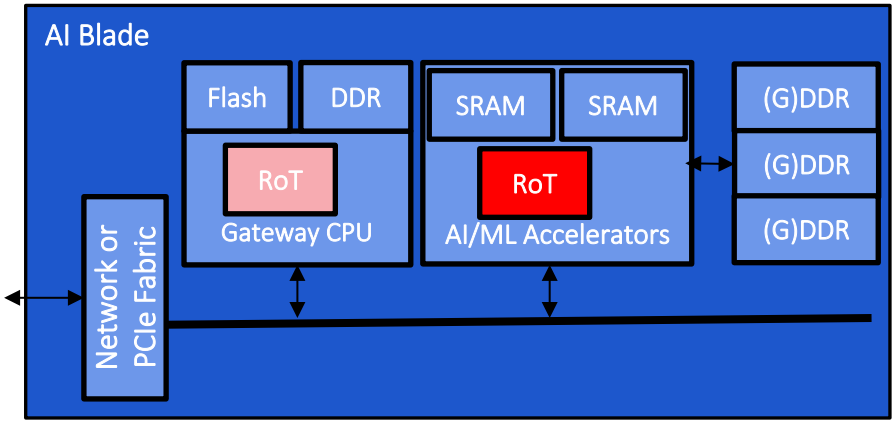
- Denial of use, misuse
- Device and data tampering, bypass security
- Asset theft, inference models, algorithms, training data

Multiple Layers of Attack:

- Noninvasive vs. Invasive
- Software
- Firmware
- Hardware

Adding CMRT to AI Blades for Protection

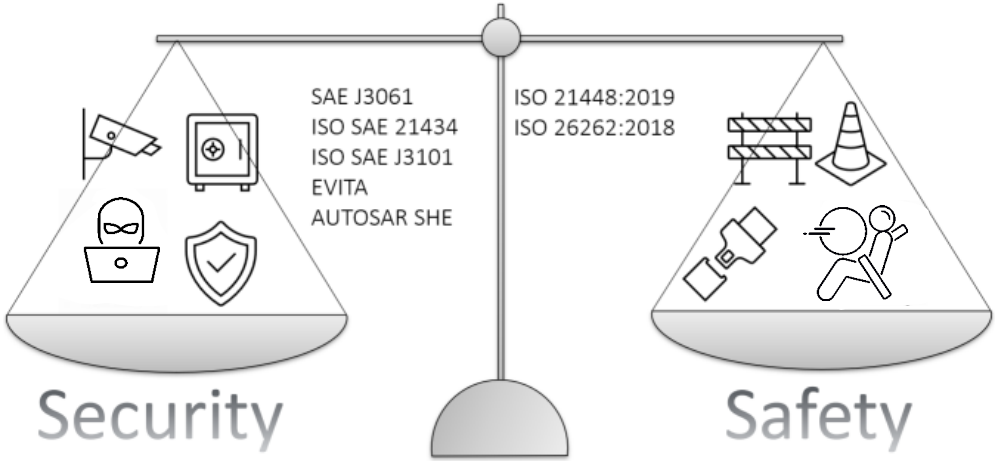
- A hardware Root of Trust (RoT) can be:
  - Provided as a discrete component
  - Embedded in each NPU (CPU optional)
- The RoT will have its own private SRAM and OTP memory (not shown here)
- The RoT will have access to the CPU/NPU, DRAM, Flash, accelerator, & network interfaces over the system bus
- The RoT may have other connections into the SoC hardware, including control and/or monitoring of reset, test and debug, and other system features
- The RoT handles accelerator personalization, provisioning, FW management, authentication.



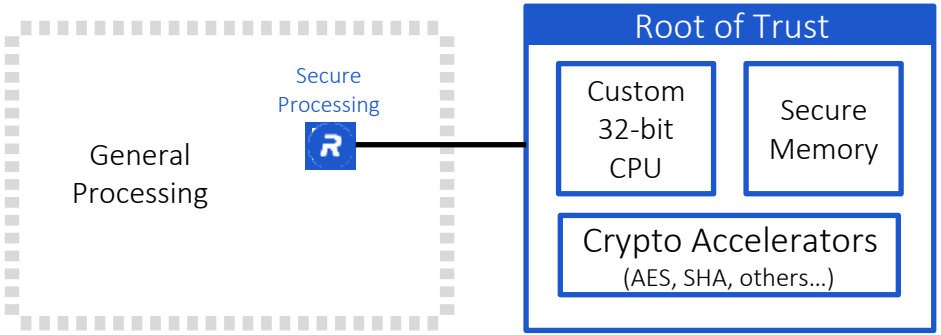
- Prevent extraction (stealing) of
  - Inference algorithms, models and parameters, training algorithms and training sets
- Prevent replacement
  - With malicious algorithms, models and training sets
  - Avoid poisoning to alter the inference results causing misclassification
- Data privacy, adhere to regulations such as
  - HIPAA in the US and GDPR in Europe

Vertical Use Case #2: Ensuring Safe and Secure Vehicles

- A safety vehicle system can be compromised by a security attack if not properly protected
- A security vehicle system can fail and enable unauthorized system actions if not implemented with the proper fault correction mechanisms



Rambus RT-64x eHSM Root of Trust for Automotive



Secure Co-Processor
RT-64x Series Root of Trust

RT-640	ASIL-B RoT	ISO-26262 **ASIL-B** certified, RoT-eHSM, EVITA Full/Medium
RT-641	ASIL-B RoT	ISO-26262 **ASIL-B** compliant, RoT-eHSM, EVITA Full/Medium, China
CH-767D	ASIL-D CMH	ISO-26262 **ASIL-D** compliant, CryptoManager Hub, EVITA Full/Medium, China

RT-64x meets ISO26262 ASIL-B Safety Integrity Levels. RT-640 is certified.
CH-767D meets ISO26262 ASIL-D Safety Integrity Levels. Certification pending.
ISO/SAE DIS 21434 cybersecurity CSMC compliance

Summary

- Connected devices across consumer electronics, datacenters, and automotive face increasing security risk and implementing a high-standard / high-quality root of trust IP in the SoC ensures security

- Rambus Crypto Manager Root of Trust (CMRT) family of IP embedded with a secure, custom RISC-V core is designed with a defense-in-depth philosophy and is ideally suited for advanced datacenter and automotive systems

- CMRT is augmented with comprehensive Security and Safety certifications like FIPS 140-3, ISO-21626, and ISO-21434 to target specific segments, reducing silicon level certification efforts and meeting compliance requirement