Implementing MultiZone™ Security in RISC-V Applications

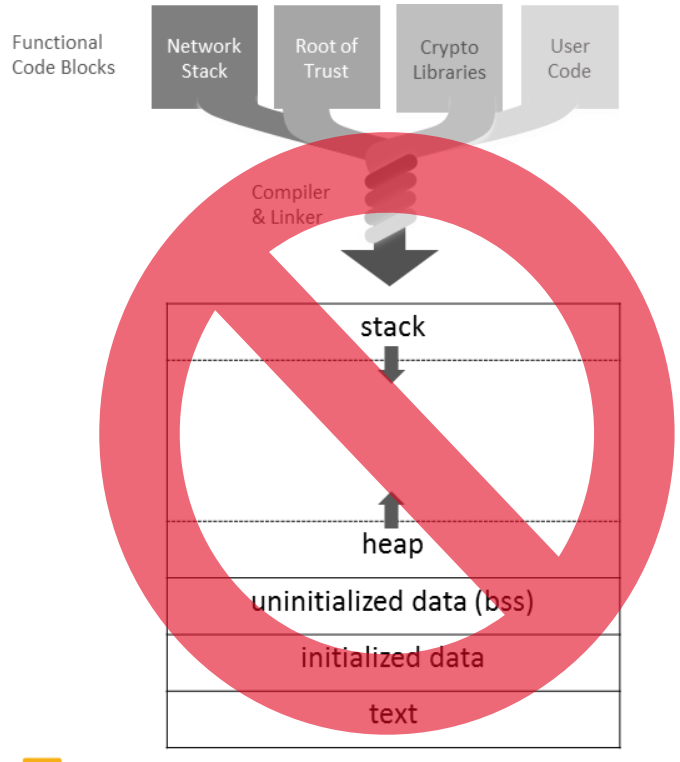
Presenter: Johnny Kim

MultiZone™ Security
Making RISC-V the most secure platform ever

MultiZone™ Security
Making RISC-V the most secure platform ever

PolarFire® SoC
The Worlds First RISC-V SoC FPGA

Embedded Computing Threat Model



- RTOS / MCU based lack basic HW security primitives like MMU & V-MEM
  - ➡ any line of code can break the CIA Confidentiality Integrity Availability
- Linux / MMU based have Virtual Memory but can't be trusted either
  - ➡ 17M+ lines of code attack surface & non-free kernel drivers
- Untrusted software: 3rd party libraries, open source, proprietary binaries
  - ➡ Supply chain security: 100+ libraries in a typical IoT stack
- L0/L1/L2 Caches in mixed-criticality systems:
  - ➡ side-channel attack trivial to exploit

RISC-V Hardware Security Primitives

Privilege Levels & Control and Status Registers

- Machine – always present, highest privilege mode
- Supervisor – Linux, supports MMU / virtual memory
- Reserved (Hypervisor) – work in progress …
- User / Application – unprivileged lowest level
- Trusted Execution Environment runs at highest privilege
- Note: Interrupts always M mode (unless “N” implemented)

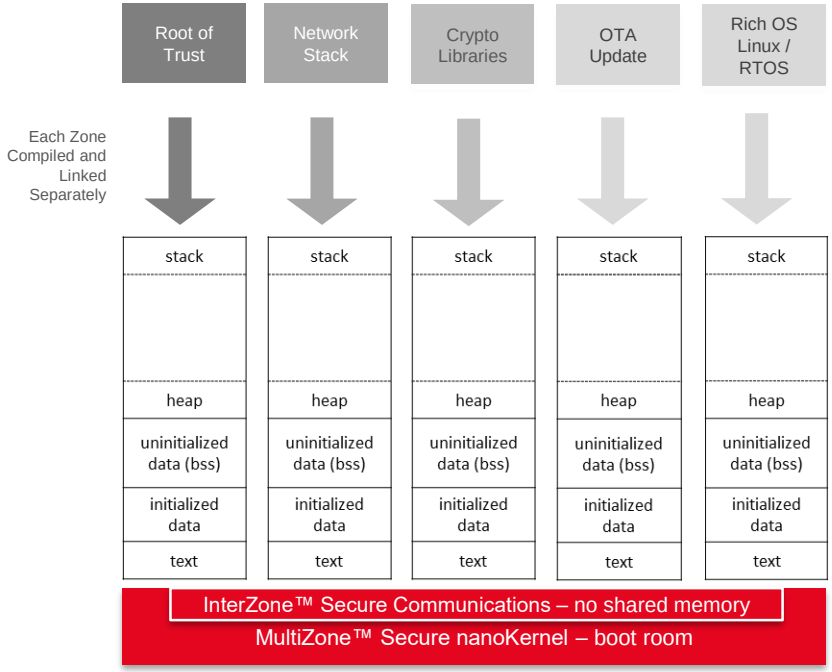
Rings	Modes	Intended Usage
1	M	Unsecured embedded
2	M,U	Secure embedded
3	M,S,U	Linux

Physical Memory Protection

- Hardware enforced – 4 ranges * 4 config reg (if implemented)
- Policy R/W/X => synchronous exception mechanism (trap)
- Overlapping OK, ranges can be locked down
- Top of range (TOR) or naturally aligned power of two (NAPOT)
- Trusted Execution Environment manages PMP context at runtime
- Note: enforced per core – no ISA spec for multi-core / platform

A	Name	Description
1	TOR	Top of range
2	NA4	Naturally aligned 4-byte
3	NAPOT	Naturally aligned power of 2

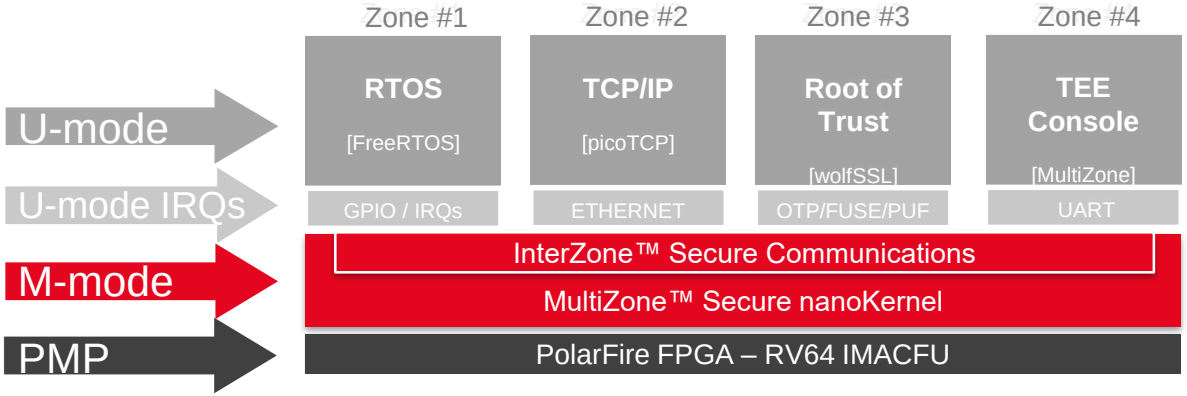
MultiZone™ Trusted Execution Environment



- ✓ Multiple equally secure zones – ram, rom, i/o, irq handlers
- ✓ Hardware-enforced, Software-defined, Policy-driven RWX
- ✓ Extremely lightweight: codebase < 2 KB, formally verifiable

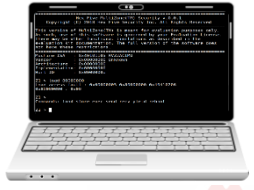
- ➡ The only commercial TEE available for RISC-V processors
- ➡ Designed for mixed-criticality, purpose-built, secure systems - L1/L2
- ➡ Scales from single-core 32-bit to multi-core 64-bit SMP Linux
- ➡ Developer friendly: easy to configure and deploy – toolchain extension
- ➡ No need to rewrite software. Runs unmodified binaries. Open source.
- ➡ Commercial software license. No royalties. Available today.

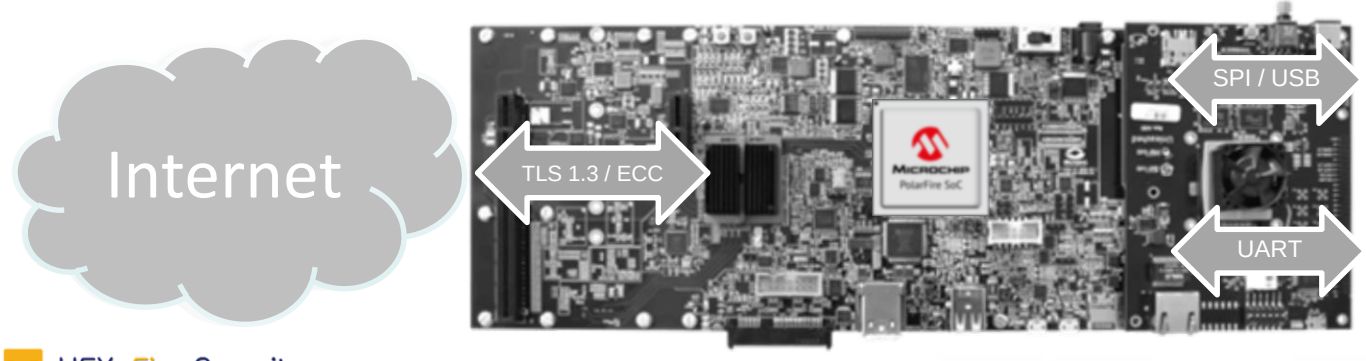
Reference Application - Secure IoT Stack



Crypto

TLSv1.3, Cipher TLS_AES_128_GCM_SHA256
Peer signing digest: SHA256
Peer signature type: ECDSA
Server Temp Key: ECDH, P-256, 256 bits
Server public key is 256 bit
Private Key ASN1 OID: prime256v1
Private Key NIST CURVE: P-256



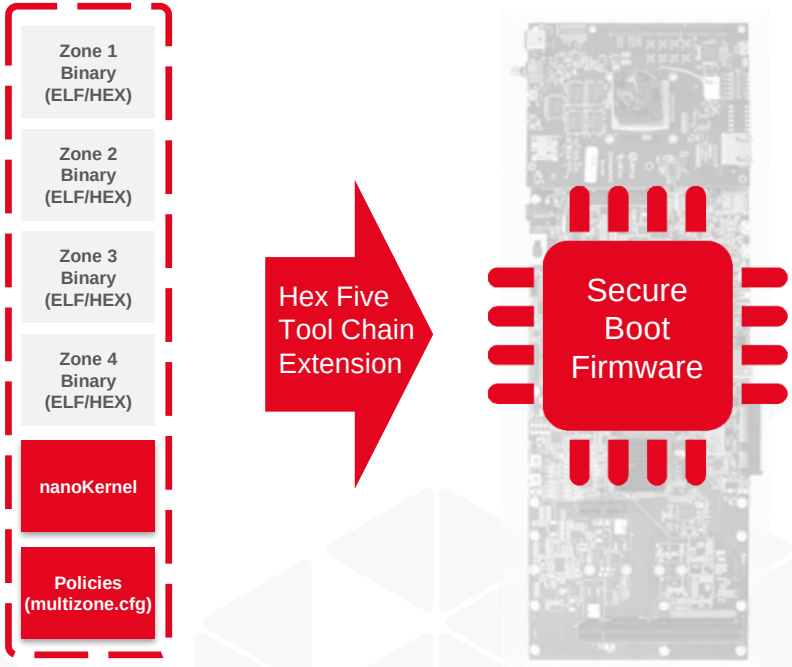


MultiZone™ Security – How It Works

Open | multizone.cfg ~/eclipse-cdt-ws/hexfive-conf | Save

```
Tick = 10 # ms

Zone = 1
                irq  = 16 # BTN0
                base = 0x20410000; size =      64K; rwx = rx # FLASH
                base = 0x80001000; size =      16K; rwx = rw # RAM
                base = 0x10025000; size =    0x100; rwx = rw # PWM
                base = 0x10012000; size =    0x100; rwx = rw # GPIO
                base = 0x0C000000; size = 0x400000; rwx = rw # PLIC

Zone = 2
                irq  = 17, 18 # BTN1, BTN2
                base = 0x20420000; size =   64K; rwx = rx # FLASH
                base = 0x80005000; size =   16K; rwx = rw # RAM
                base = 0x60000000; size =    8K; rwx = rw # XEMACLITE

Zone = 3
                base = 0x20430000; size =   64K; rwx = rx # FLASH
                base = 0x80009000; size =    4K; rwx = rw # RAM

Zone = 4
                base = 0x20440000; size =   64K; rwx = rx # FLASH
                base = 0x8000A000; size =    4K; rwx = rw # RAM
                base = 0x10013000; size = 0x100; rwx = rw # UART
```

Plain Text | Tab Width: 3 | Ln 10, Col 1 | INS



Patent pending US 16450826, PCT US1938774 – Configuring, Enforcing, And Monitoring Separation Of Trusted Execution Environments.

MultiZone™ Open Standard API – C Library

```
/* Copyright(C) 2019 Hex Five Security, Inc.

Permission to use, copy, modify, and/or distribute this software for
any purpose with or without fee is hereby granted, provided that the
above copyright notice and this permission notice appear in all copies.

*/

#ifndef LIBHEXFIVE_H_
#define LIBHEXFIVE_H_

void ECALL_YIELD();
void ECALL_WFI();

int ECALL_SEND(int, void *);
int ECALL_RECV(int, void *);

void ECALL_TRP_VECT(int, void *);
void ECALL_IRQ_VECT(int, void *);

void ECALL_CSRS_MIE();
void ECALL_CSRC_MIE();

void ECALL_CSRW_MTIMECMP(uint64_t);

uint64_t ECALL_CSRR_MTIME();
uint64_t ECALL_CSRR_MCYCLE();
uint64_t ECALL_CSRR_MINSTR();
uint64_t ECALL_CSRR_MHPMC3();
uint64_t ECALL_CSRR_MHPMC4();

uint64_t ECALL_CSRR_MISA();
uint64_t ECALL_CSRR_MVENDID();
uint64_t ECALL_CSRR_MARCHID();
uint64_t ECALL_CSRR_MIMPID();
uint64_t ECALL_CSRR_MHARTID();

#endif /* LIBHEXFIVE_H_ */
```

Permissive Licensing – “any purpose”

Hardware threads (zones) management

Inter zone messaging – zone0 SMP Linux

Traps & IRQs handlers registration (U-mode)

Traps & IRQs enable / disable – per zone

Hardware thread timer – per zone

Trap & emulation helpers
Read-only, selected CSRs
Completely optional – just for speed / latency

Enabling Purpose-built, Real-time, Low Power systems

POLARFIRE® SOC

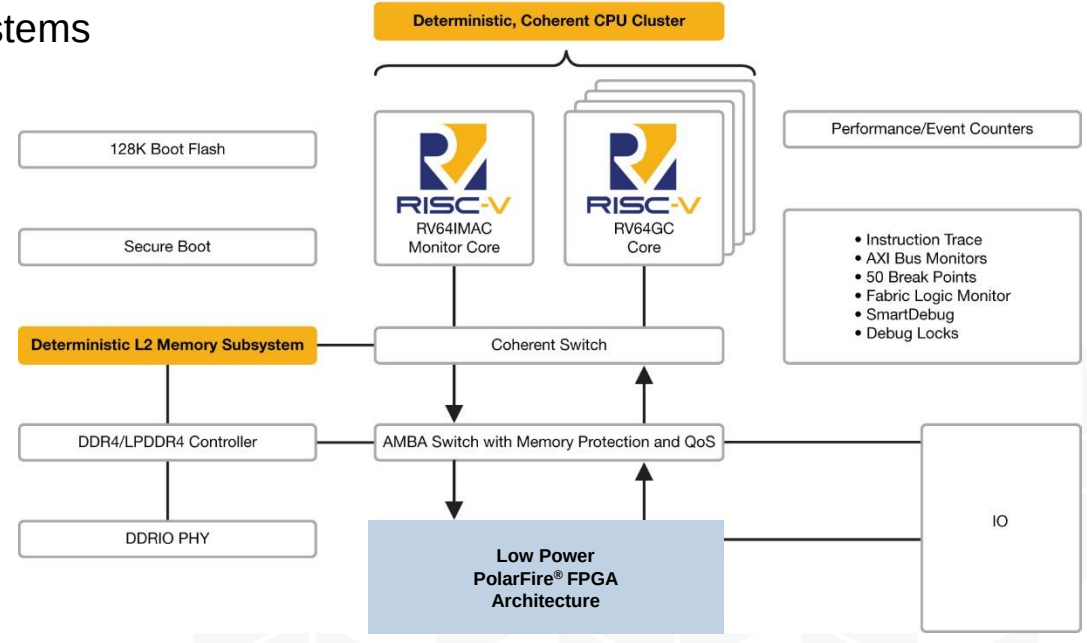
MICROCHIP

PolarFire® SoC - RISC-V enabled innovation platform

Freedom to Innovate in

- Linux and Real-Time
- Thermal and Power Constrained Systems
- Securely Connected IoT systems
- High-Rel Safety Critical Systems

PolarFire® SoC Architecture



SiFive

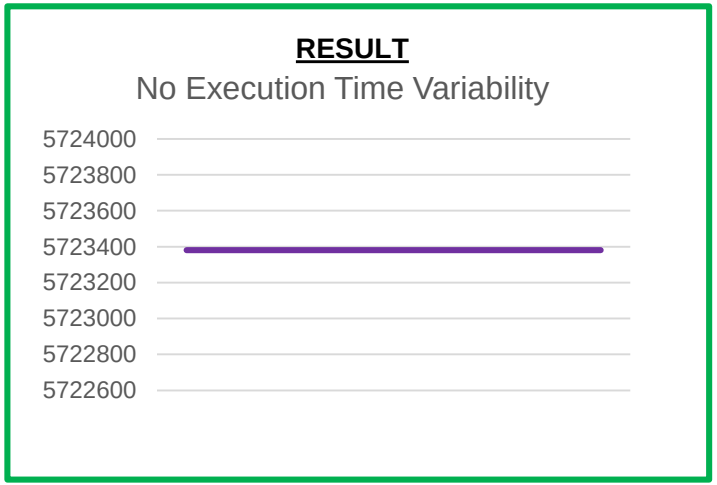
ultraSoC

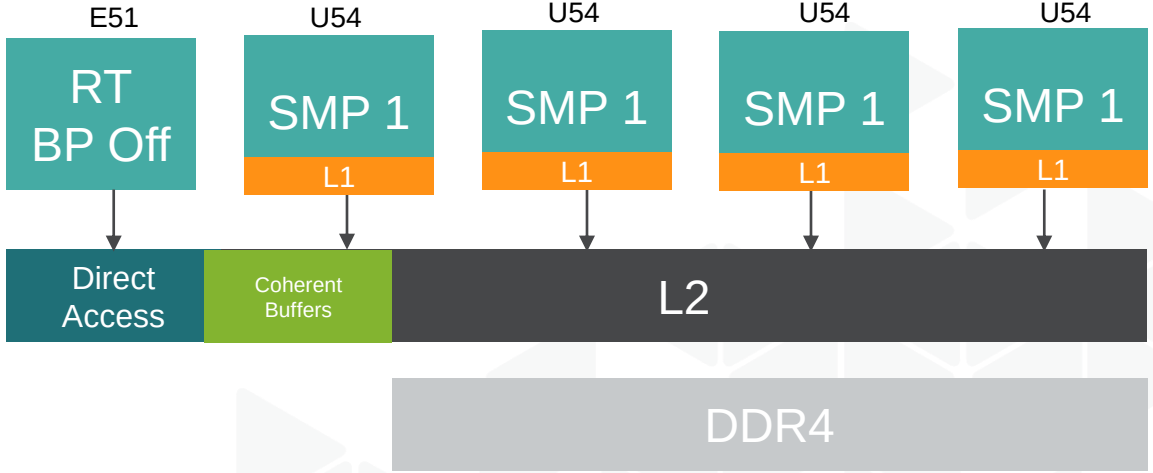
0x5 HEX-Five Security

MICROCHIP

Real-Time and Linux

- **Turn off the CPU branch predictors**
- **Configure L1 to Tightly Integrated Memory**
- **Configure the L2 memory system to provide determinism**
- **Make sure all cores coherent to the memory subsystem**
  - Share coherent memory for message passing





Security Built for Defense, Ready for IoT

PolarFire® SoC inherits defense grade PolarFire FPGA Security

- DPA resistant bitstream programming
- Anti-tamper
- Cryptographical bound supply chain assurance
- Physically unclonable function
- True random number generator
- Side channel resistant crypto coprocessor

PolarFire SoC has:

+ **Secure Boot**
+ **Spectre and Meltdown immunity**
+ **Physical memory protection**
+ **SECDED on all memories**

Thank You

For more about MultiZone™

- www.hex-five.com
- info@hex-five.com

For more about PolarFire® SoC

- PolarFire SoC
- PolarfireSoC@Microchip.com