



How to Build & Secure a RISC-V Embedded System

HARDWEAR.IO, September 2019

Cesare Garlati, Sandro Pinto



RISC-V Foundation: 200+ Members



RISC-V ISA Security Building Blocks

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)

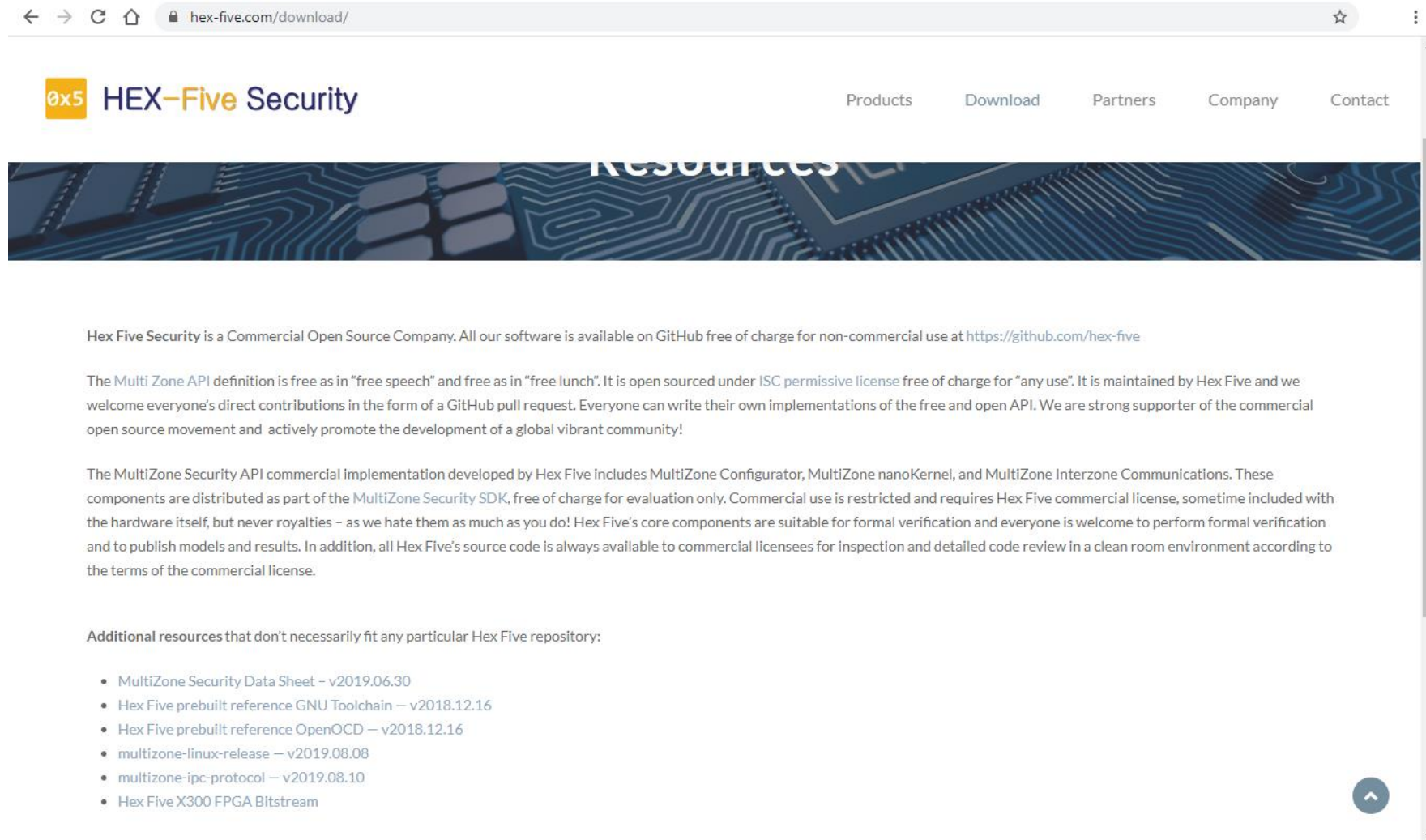
Rings	Modes	Intended Usage
1	M	Unsecured embedded
2	M,U	Secure embedded
3	M,S,U	Linux

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

Download Resources



← → ↻ 🏠 hex-five.com/download/ ☆ ⋮

0x5 HEX-Five Security Products Download Partners Company Contact

Resources

Hex Five Security is a Commercial Open Source Company. All our software is available on GitHub free of charge for non-commercial use at <https://github.com/hex-five>

The Multi Zone API definition is free as in “free speech” and free as in “free lunch”. It is open sourced under ISC permissive license free of charge for “any use”. It is maintained by Hex Five and we welcome everyone’s direct contributions in the form of a GitHub pull request. Everyone can write their own implementations of the free and open API. We are strong supporter of the commercial open source movement and actively promote the development of a global vibrant community!

The MultiZone Security API commercial implementation developed by Hex Five includes MultiZone Configurator, MultiZone nanoKernel, and MultiZone Interzone Communications. These components are distributed as part of the MultiZone Security SDK, free of charge for evaluation only. Commercial use is restricted and requires Hex Five commercial license, sometime included with the hardware itself, but never royalties – as we hate them as much as you do! Hex Five’s core components are suitable for formal verification and everyone is welcome to perform formal verification and to publish models and results. In addition, all Hex Five’s source code is always available to commercial licensees for inspection and detailed code review in a clean room environment according to the terms of the commercial license.

Additional resources that don’t necessarily fit any particular Hex Five repository:

- MultiZone Security Data Sheet – v2019.06.30
- Hex Five prebuilt reference GNU Toolchain – v2018.12.16
- Hex Five prebuilt reference OpenOCD – v2018.12.16
- multizone-linux-release – v2019.08.08
- multizone-ipc-protocol – v2019.08.10
- Hex Five X300 FPGA Bitstream

<https://hex-five.com/download/>

Download Resources – GitHub

hex-five / multizone-sdk

Unwatch 13 Unstar 21 Fork 13

Code Issues 11 Pull requests 0 Security Insights

MultiZone™ Security SDK for any RISC-V platform

risc-v security tee multizone multizone-security trusted-computing trusted-execution-environment hex-five secure-element firmware

secure-boot root-of-trust container microkernel

70 commits 2 branches 3 releases 3 contributors View license

Branch: master New pull request Create new file Upload files Find File Clone or download

cgarlati Kernel-Version: v1.0.1-24-g8816002 Latest commit #63eb56 21 days ago

bsp	Kernel-Version: v1.0.1-21-g0c4b9cf	2 months ago
libhexfive	Rollup patch release 26-AUG-2019	last month
zone1	Kernel-Version: v1.0.1-21-g0c4b9cf	2 months ago
zone2	Kernel-Version: v1.0.1-21-g0c4b9cf	2 months ago
zone3	Rollup patch release 26-AUG-2019	last month
.gitignore	.	8 months ago
LICENSE	Add files via upload	9 months ago
Makefile	Add E902 (rv32e) - WIP	3 months ago
README.md	Update README.md	2 months ago
manual.pdf	Update manual.pdf (#44)	6 months ago
multizone.jar	Kernel-Version: v1.0.1-24-g8816002	21 days ago

<https://github.com/hex-five/multizone-sdk>

HEX-Five X300 SoC – ARTY7 FPGA

E300	X300
RV32ACIM	RV32ACIMU
32.5 MHz clock	65 MHz clock
2 HW breakpoints	8 HW breakpoints
no Ethernet core	Xilinx EthernetLite Ethernet core
1-way icache	4-way icache
no ITIM	ITIM at 0x0800_0000
16 kB DTIM	64 kB DTIM
no perf counters	2 perf counters, hpmcounter3 and hpmcounter4
no CLICs	3 CLICs (BTN0, BTN1 and BTN2)

The **X300** is developed and maintained by Hex Five to support MultiZone IoT applications.

The **X300** SoC is an enhanced version of the Freedom E300 Platform based on the original Rocket Chip developed at U.C. Berkeley and now maintained by SiFive.

The **X300** is completely open source and free of charge for commercial and non-commercial use.

[GitHub hex-five/multizone-fpga](https://github.com/hex-five/multizone-fpga)

MultiZone™ Security – How It Works

```
multizone.cfg
~/eclipse-cdt-ws/hexfive-conf

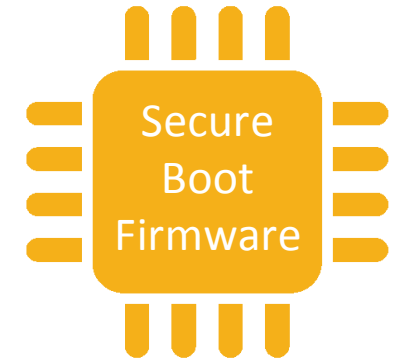
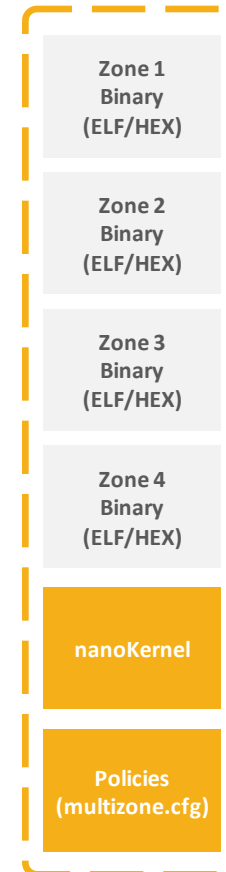
Tick = 10 # ms

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
```



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



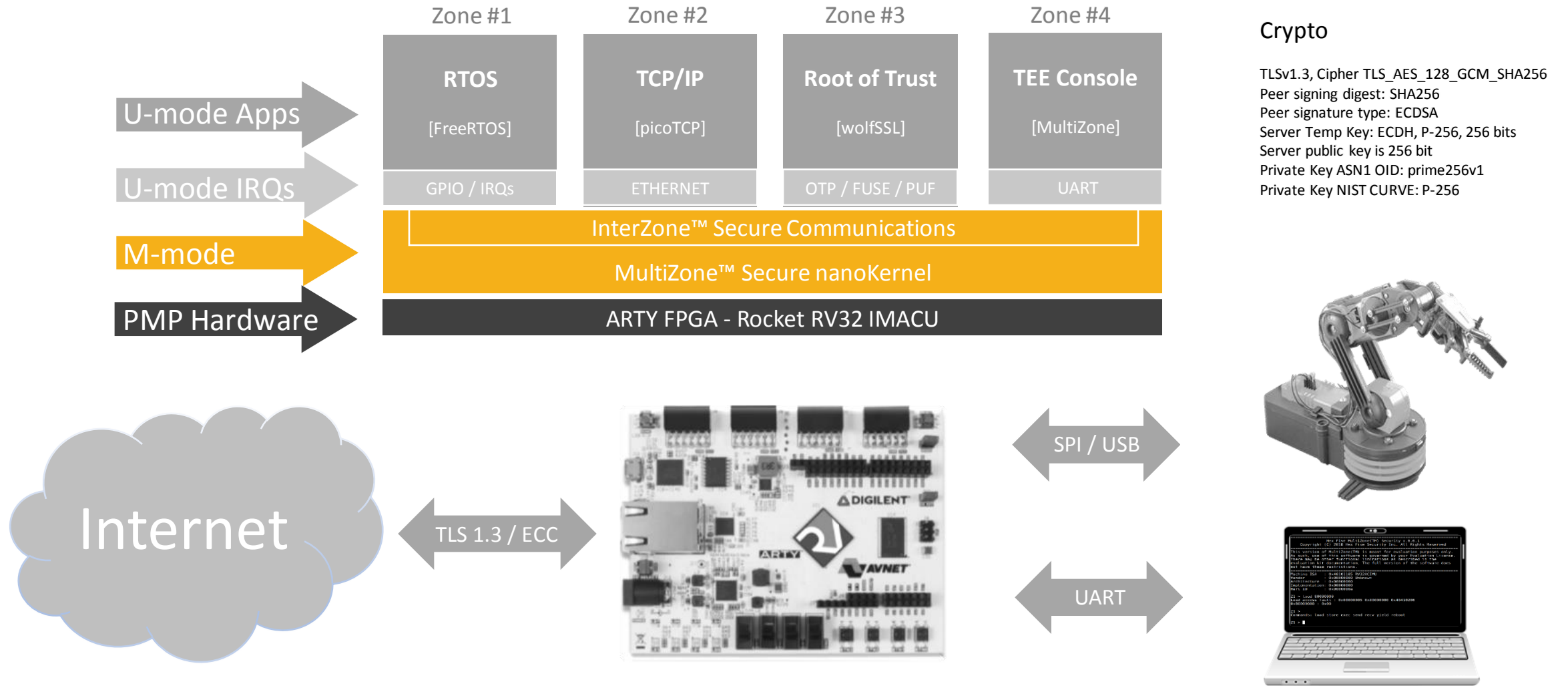
Trap & emulation helpers

Read-only, selected CSRs

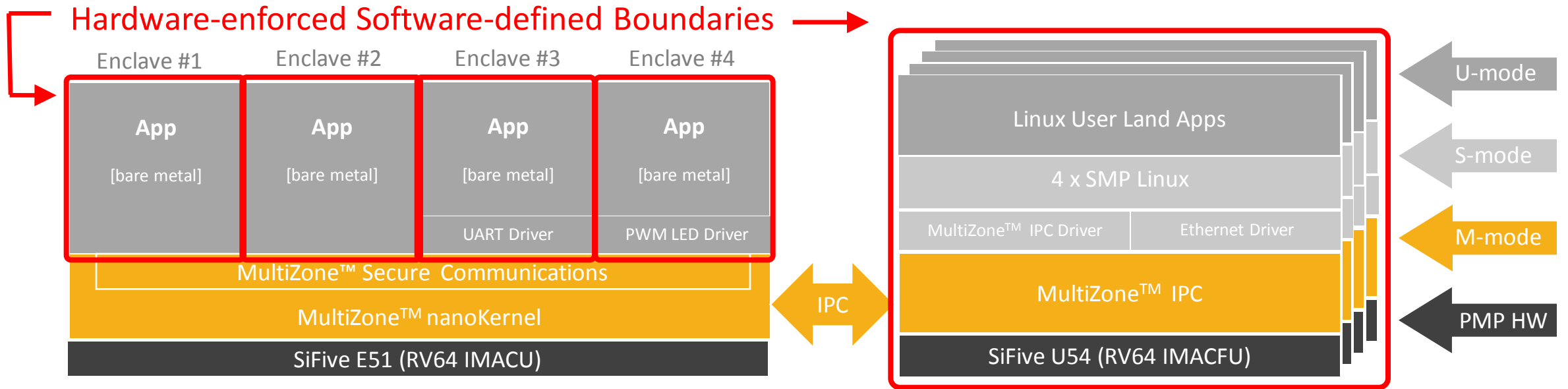
Completely optional – just for speed / latency



Reference Application – Secure IoT Stack



MultiZone™ For Linux – Enclave Concept



- ✓ Multiple statically defined enclaves – ram, rom, i/o, irq
- ✓ Secure messaging with no shared mem – secure buffers for Linux IPC
- ✓ Secure interrupt handlers mapped to enclaves and executed in U-mode
- ✓ Trap & emulation of privileged instructions, Soft-timers, Secure boot



Hex Five MultiZone™ Security

Hex Five Security, Inc. is the creator of MultiZone™ Security, the first Trusted Execution Environment for RISC-V. Hex Five open standard technology provides software-defined hardware-enforced separation for multiple security domains, with full isolation of data, programs and peripherals. Contrary to traditional solutions, MultiZone™ Security requires no additional hardware or changes to existing software: open source libraries, third party binaries and legacy code can be configured in minutes to achieve unprecedented levels of safety and security.