RISC-V Enclaves: A Clean Slate Approach To Linux Security

Johnny Kim – Senior Embedded Solutions Engineer

HEX-Five Security

MICROCHIP

Securing Mixed Criticality Systems

- Multiple trusted workloads on untrusted platforms – i.e. rich OS
- Linux / RTOS / Legacy OS
- Sharing the same core(s) is not secure: M-mode SBI & L1/L2
- Zero-trust: assume the supply chain is compromised – i.e. drivers
- Zero-tolerance for thick hypervisors and large codebase sys software

Linux Uncomfortable Truth

```
~/linux-4.18.6$ cloc --exclude-lang=DTD,Lua,make .
   60965 text files.
   60546 unique files.
   14391 files ignored.

-------------------------------------------------------------------------------
Language                     files          blank        comment           code
-------------------------------------------------------------------------------
C                            25782        2554166        2248398       12965944
C/C++ Header                 18693         484773         892818        3629746
Assembly                      1318          47155         105960         232515
JSON                           189              0              0         102201
Perl                            55           5414           3994          27294
Bourne Shell                   346           5633           4983          24450
Python                         108           3055           3337          17427
HTML                             5            669              0           5492
yacc                             9            701            375           4648
lex                              8            326            314           2007
C++                              7            285             77           1844
Bourne Again Shell              51            351            318           1711
awk                             11            170            155           1384
Markdown                         1            220              0           1077
TeX                              1            108              3            915
NAnt script                      2            156              0            599
Windows Module Definition        2             14              0            102
m4                               1             15              1             95
XSLT                             5             13             26             61
CSS                              1             18             27             44
vim script                       1              3             12             27
Ruby                             1              4              0             25
INI                              1              1              0              6
sed                              1              2              5              5
-------------------------------------------------------------------------------
SUM:                         46599        3103252                     17019619
-------------------------------------------------------------------------------
```

(a) Industry Average: "about 15 - 50 errors per 1000 lines of delivered code." He further says this is usually representative of code that has some level of structured programming behind it, but probably includes a mix of coding techniques.

(b) Microsoft Applications: "about 10 - 20 defects per 1000 lines of code during in-house testing, and 0.5 defect per KLOC (KLOC IS CALLED AS 1000 lines of code) in released product (Moore 1992)." He attributes this to a combination of code-reading techniques and independent testing (discussed further in another chapter of his book).

(c) "Harlan Mills pioneered 'cleanroom development', a technique that has been able to achieve rates as low as 3 defects per 1000 lines of code during in-house testing and 0.1 defect per 1000 lines of code in released product (Cobb and Mills 1990).

$17{,}019{,}619 * 10^{-4} =$ **1,701** disasters waiting to happen

RISC-V Security Primitives

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

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

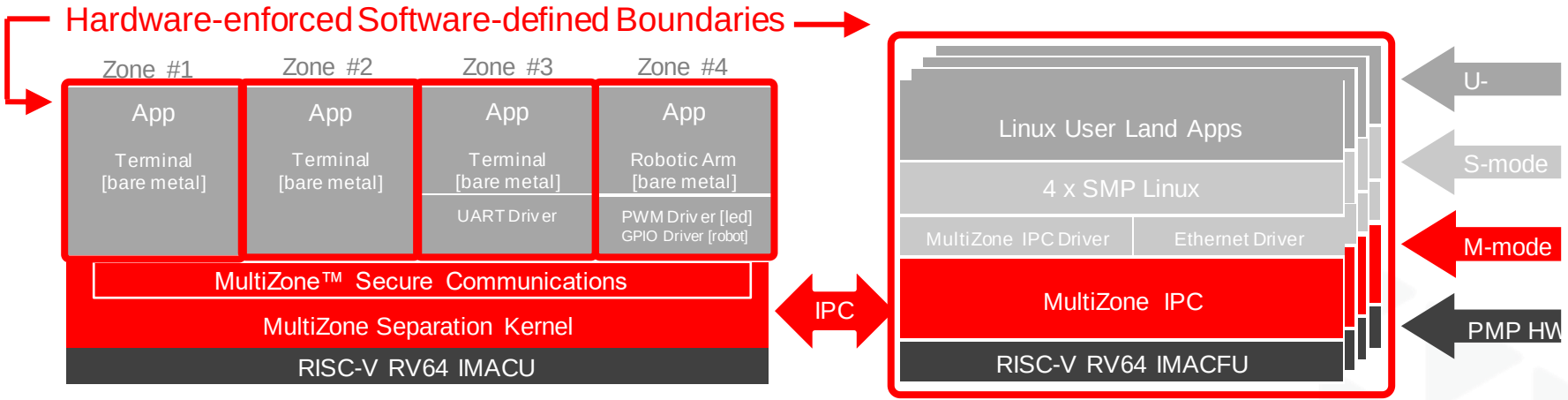
Privilege Levels & Control and Status Registers

- 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)

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 / I/O

MultiZone™ Security PolarFire® Edition



- ✓ 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 Linux enclave for RISC-V
- ✓ Easy to configure and deploy – toolchain extension
- ✓ No need to rewrite software: runs unmodified binaries

0x5 HEX-Five Security

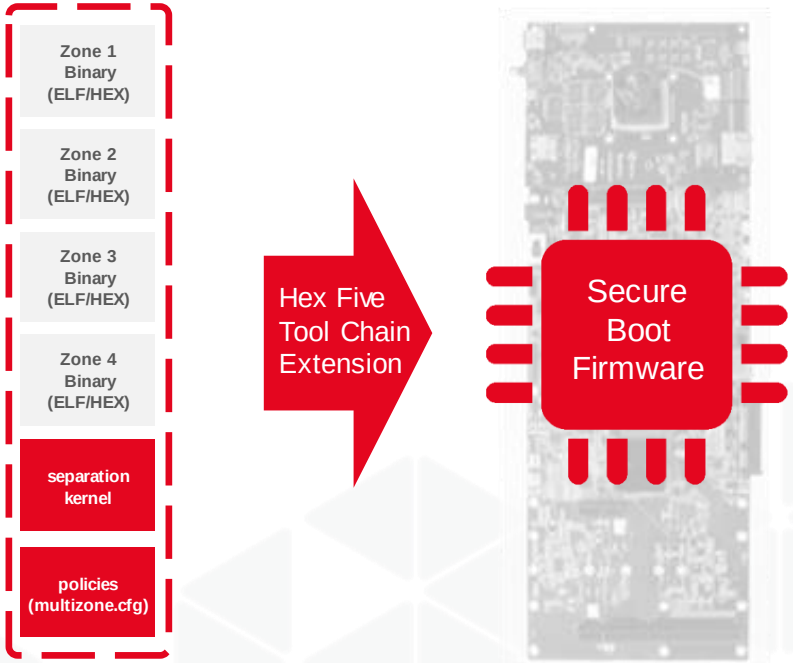
MICROCHIP

MultiZone™ Security - How it works

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

```
Tick = 1 # ms

Zone = 1    base = 0x0008000000; size =   28K; rwx = rx # L2LIM .text
            base = 0x0008007000; size =    4K; rwx = rw # L2LIM .data

Zone = 2    base = 0x0008008000; size =   28K; rwx = rx # L2LIM .text
            base = 0x000800F000; size =    4K; rwx = rw # L2LIM .data

Zone = 3    base = 0x0008010000; size =   28K; rwx = rx # L2LIM .text
            base = 0x0008017000; size =    4K; rwx = rw # L2LIM .data
            base = 0x2000104000; size = 0x100; rwx = rw # UART0 EXP

Zone = 4    base = 0x0008018000; size =   28K; rwx = rx # L2LIM .text
            base = 0x000801F000; size =    4K; rwx = rw # L2LIM .data
            base = 0x000200BFF8 size =   0x8; rwx = r  # RTC
            base = 0x0010020000; size = 0x100; rwx = rw # PWM LED
            base = 0x2000103000; size = 0x100; rwx = rw # GPIO EXP

Hart = 1,2,3,4 # SMP Linux
            base = 0x01000000; size =   256; rwx = rw  # DTIM0
            base = 0x01000000; size =    8K; rwx = --- # DTIM0
            base = 0x01800000; size =    8K; rwx = --- # ITIM0
            base = 0x08000000; size =  128K; rwx = --- # L2LIM
            base = 0x20000000; size =    8M; rwx = --- # FLASH
            base = 0x02004000; size =     8; rwx = --- # MTIMECMP0
            base = 0x10020000; size = 0x100; rwx = --- # PWM
            base = 0x00000000; size =  256G; rwx = rwx # ACCESS ALL
```

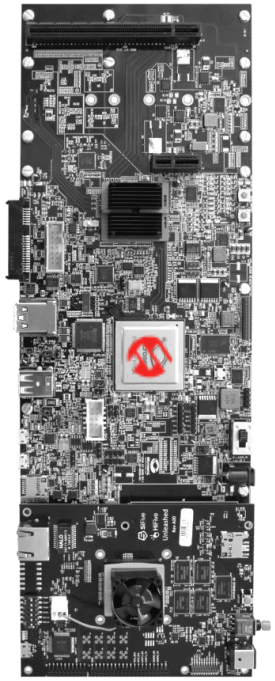
Plain Text ▾ Tab Width: 3 ▾ Ln 10, Col 1 ▾ INS



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

MICROCHIP

MultiZone™ Security – Live Demo



SSH – Enclave #1

SSH – Enclave #2

UART – Enclave #3

```
cesare@cesare-pc: ~
File  Edit  View  Search  Terminal  Help
cesare@cesare-pc: ~
File  Edit  View  Search  Terminal  Help
cesare@cesare-pc: ~
File  Edit  View  Search  Terminal  Help
================================================================================
                    Hex Five MultiZone(TM) Security
    Copyright (C) 2018 Hex Five Security Inc. All Rights Reserved
================================================================================
This version of MultiZone(TM) is meant for evaluation purposes only.
As such, use of this software is governed by your Evaluation License.
There may be other functional limitations as described in the
evaluation kit documentation. The full version of the software does
not have these restrictions.
================================================================================
Machine ISA    : 0x00101105 RV64 ACIMU
Vendor         : 0x00000000
Architecture   : 0x00000000
Implementation: 0x00000000
Hart ID        : 0x00000000
CPU clock      : 1000 MHz

Z1 > pmp
0x08008000 0x0800FFFF r-x NAPOT
0x08001000 0x08001FFF rw- NAPOT

Z1 > load 08010000
Load access fault : 0x00000005 0x08010000 0x08008252
0x08010000 : 0x00

Z1 >
```

0x5 HEX-Five Security

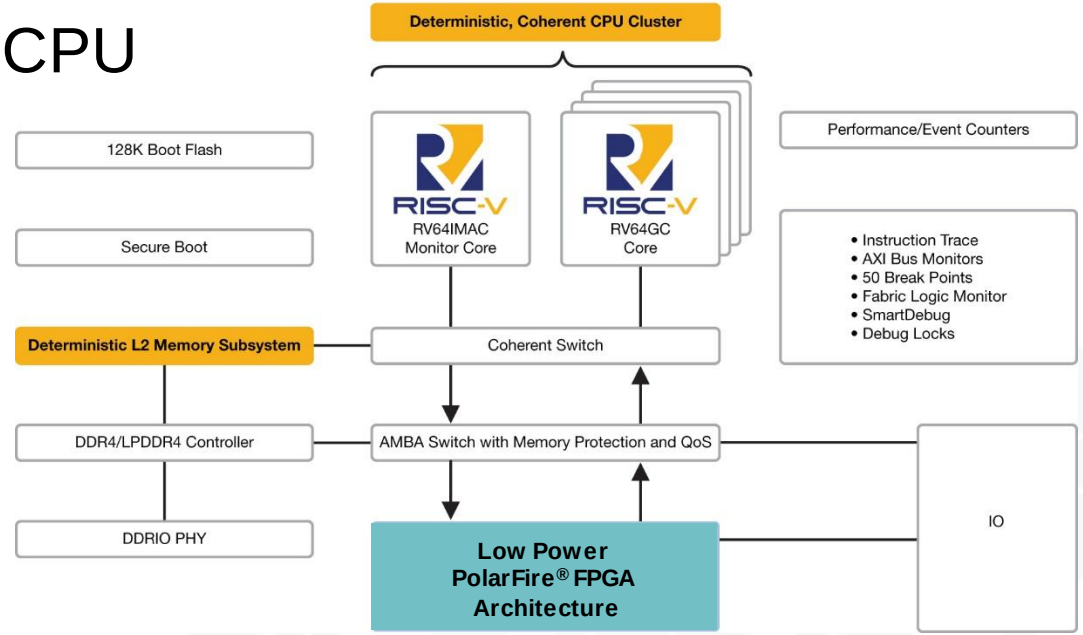
MICROCHIP

PolarFire® SoC and Multizone for Mixed Criticality Systems

PolarFire® SoC

- 64 bit RISC-V 5 Core CPU
- Multi-Zone Capable
- Secure Boot

PolarFire® SoC Architecture



0x5 HEX-Five Security

SiFive

ultraSoC

MICROCHIP

Security Built for Defense, Ready for IoT

PolarFire® SoC inherits defense grade PolarFire FPGA Security

- Cryptographical bound supply chain assurance
- DPA resistant bitstream programming
- Anti-tamper
- 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**

MICROCHIP

Thank You!

More about MultiZone™ Security

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

More about PolarFire® SoC

- PolarFire SoC
- PolarfireSoC@Microchip.com