HEX-Five MultiZone™ Security

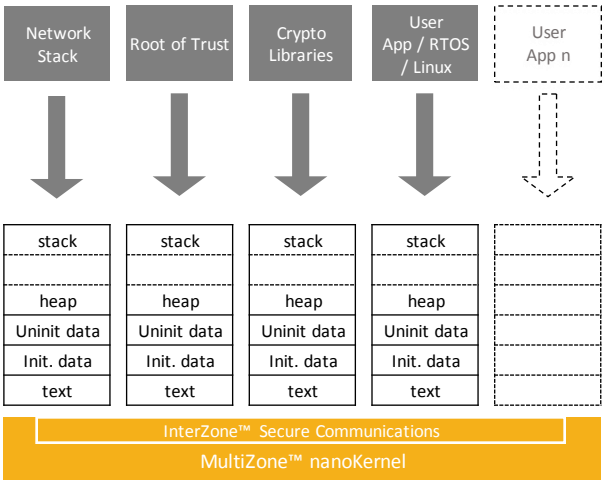
It's like Docker at the chip level. It provides hardware-enforced, software-enabled separation for an unlimited number of security domains, with full control over data, code and peripherals. With MultiZone™ Security open source libraries, third party binaries and legacy code can be configured in minutes to achieve unprecedented levels of safety and security.

What is MultiZone™ Security?

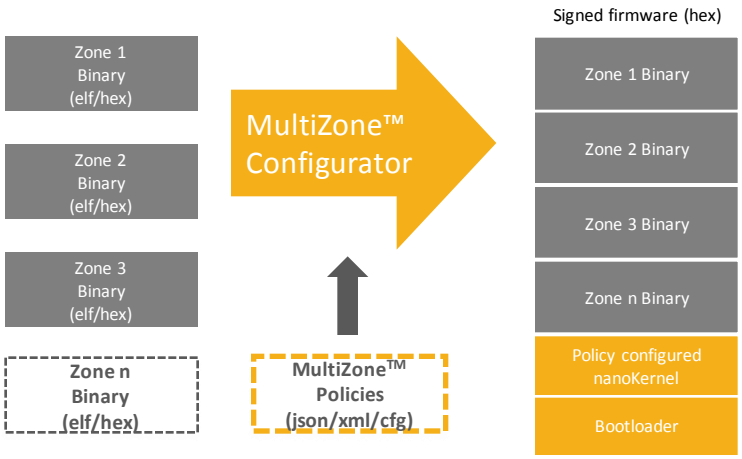
MultiZone™ Security is the first Trusted Execution Environment for RISC-V. It's a thin layer of software that orchestrates RISC-V built-in hardware security blocks to provide robust security through separation within the hardware itself. MultiZone is provided as a self-contained toolchain extension, so no coding or security expertise is required.

MultiZone™ Security consists of the following components

- ***MultiZone™ nanoKernel*** – lightweight, formally verifiable, bare metal kernel providing hardware-enforced separation of ram, rom, and i/o.
- ***InterZone™ Messenger*** – communications infrastructure to exchange secure messages across zones on a no- shared memory basis.
- **MultiZone™ Configurator** – combines fully linked zone executables with policies and kernel to generate the signed firmware image.
- **MultiZone™ Secure Boot** – 2-stage secure boot loader to verify integrity and authenticity of the firmware image (sha-256 / sha-512)



How does MultiZone™ Security work?



MultiZone™ Security integrates seamlessly into your existing IDE such as Eclipse or command line based toolset.

Application blocks are written, compiled and linked separately for each zone producing a set of elf or hex file.

MultiZone™ Policies are set to achieve the desired ram, rom, i/o and interrupt isolation for each zone – RWX, with granularity down to 4 bytes.

Finally the MultiZone™ configurator is invoked to combine zone elf/hex files with the nanoKernel and bootloader into a signed firmware image.

The full system can be written, compiled and debugged with your existing GNU or Eclipse toolset.

Patent pending US 16/450,826 PCT US19/38774

www.hex-five.com

info@hex-five.com

HEX-Five MultiZone™ Security

It's like Docker at the chip level. It provides hardware-enforced, software-enabled separation for an unlimited number of security domains, with full control over data, code and peripherals.
With MultiZone™ Security open source libraries, third party binaries and legacy code can be configured in minutes to achieve unprecedented levels of safety and security.

Features

- Unlimited number of isolated Trusted Execution Environments (zones) - hardware-enforced, policy-defined
- Unlimited number of memory-mapped resources per zone – i.e. flash, ram, i/o, uart, gpio, timers, etc.
- Preemptive real time scheduler: round robin / cooperative, configurable time tick, cpu overhead < 0.1%
- Any combination of top-of-range and naturally aligned configuration – minimum granularity 4 bytes
- Any combination of read, write, execute policy – resource overlapping allowed although not recommended
- Full support for PLIC and CLIC Interrupts – fully configurable zone / interrupt mapping
- Full support for secure user-mode interrupt handlers – doesn't require 'N' extensions
- Full support for low-latency vectored interrupts, preemptable interrupts, and Wait For Interrupt - suspend mode
- Built in trap & emulate for most privileged instructions – i.e. CSR read
- Secure inter-zone communications infrastructure based on messaging - no shared memory or buffers
- Secure boot suitable for 2-stage boot room and/or public key / root of trust / PUF – SHA-256 / SHA-512
- Formally verifiable, completely written in assembly, self-contained - no 3rd party libraries dependencies

Development Environments

- Eclipse IDE including MCU and GNU Toolchain plugins and OpenOCD / JTAG / GDB live debugging
- Linux and Windows command line tools (make, gcc, gdb, etc.) – native Linux, Java 1.8 required for Windows
- Command line configurator utility compatible with any operating system capable of running Java 1.8
- C library wrapper for protected mode execution – via ECALL exception handling mechanism
- SMP Linux and AMP FreeRTOS high-speed drivers available for multicore implementations
- Open source SDK and Secure IoT Stack (freeRTOS, picoTCP, wolfSSL) freely available on GitHub
- Open source Board Support Packages available for SiFive, Andes, Microsemi, Gowin, Codasip

System Requirements

- 32 bit or 64 bit RISC-V ISA with 'U' extension – compatible with 'S' and 'H' if present
- Physical Memory Protection compliant with Ver. 1.10 or greater – at least 4 PMP registers, 8 recommended
- 4KB FLASH and 1KB RAM for 32 bit or 2KB RAM for 16 bit version (4 Zones)

Patent pending US 16/450,826 PCT US19/38774

Hex Five Security is a proud member of the RISC-V Foundation

HEX-Five Security

RISC-V

www.hex-five.com

info@hex-five.com