HEX-Five MultiZone® Security

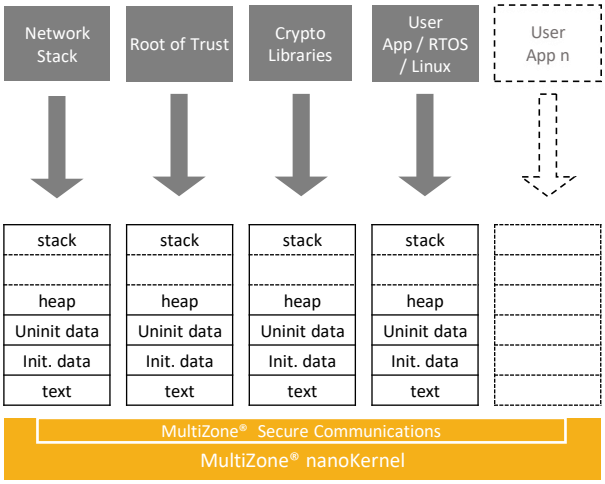
MultiZone® Security is hardware-enforced software-defined separation of multiple application domains, with full control over data, programs, and peripherals. Contrary to traditional solutions, MultiZone® Security is policy-driven and requires no hypervisor software or hardware support for virtualization: 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?

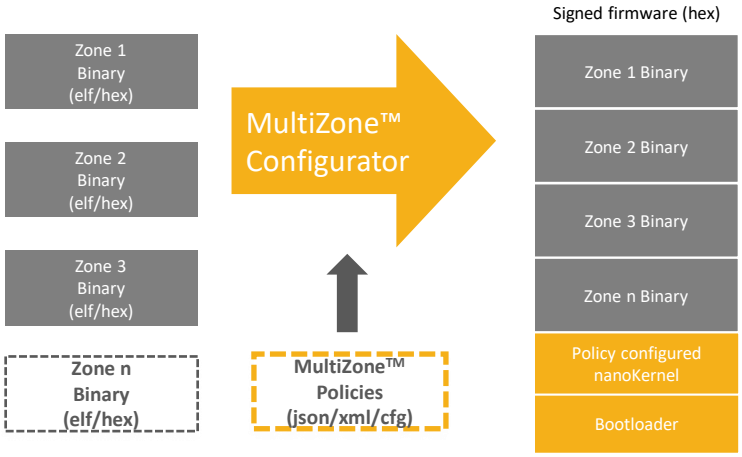
MultiZone® Security is the first Trusted Execution Environment for RISC-V. It's a thin layer of software that orchestrates RISC-V built-in hardware security blocks to provide robust security through separation within the hardware itself. MultiZone is provided as a self-contained toolchain extension, 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.
- ***MultiZone® Messenger*** – communications infrastructure to exchange secure messages across zones on a non-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

MultiZone® Security is hardware-enforced software-defined separation of multiple application domains, with full control over data, programs, and peripherals. Contrary to traditional solutions, MultiZone® Security is policy-driven and requires no hypervisor software or hardware support for virtualization: open source libraries, third party binaries, and legacy code can be configured in minutes to achieve unprecedented levels of safety and security.

Features

- Multiple separated Trusted Execution Environments (zones) – hardware-enforced software-defined
- Multiple memory-mapped resources per zone – i.e. flash, ram, i/o, uart, gpio, timers, irqs, etc.
- Any combination of read, write, execute policy – resource overlapping allowed although not recommended
- Preemptive scheduler for safety-critical applications: round robin, configurable tick, cpu overhead < 0.01%
- Secure interzone communications based on messaging – no shared memory
- Built-in trap & emulation for all privileged instructions – i.e. CSR, MRET, ECALL, WFI, etc.
- 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 Wait For Interrupt and and CPU suspend mode for low power applications
- Formally verifiable runtime <2KB, 100% written in assembly, self-contained, zero 3rd party 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 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 Andes, Microchip, GoWin, SiFive, NXP Vega Board

System Requirements

- rv32i, rv32e, rv64i ISA with U-mode extension – compatible with 'S' and 'H' if present
- Physical Memory Protection compliant with Ver. 1.11 or greater – minimum 4 PMP registers, 8 recommended
- 4KB FLASH and 1KB RAM for 32 bit or 2KB RAM for 16 bit version (4 Zones configuration)

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