U.S. ARMY COMBAT CAPABILITIES DEVELOPMENT COMMAND
AVIATION & MISSILE CENTER

Secure Architecture for Mission Critical Systems

Presented to:

2022 Trusted Computing Center of Excellence (TCCoE) Summit

Tom Barnett

Cyber Integration Lead

Cyber Technologies Division, S3I

1 February 2022

Background

- U.S. Army DEVCOM Aviation and Missile Center (AvMC): a leader in developing technologies to increase weapon system survivability through cyber resilience
- Gap: Today's Aviation systems are based on computer and computational architectures that were not designed with security as a requirement
- Key challenge: Rethink fundamental computational **architectures** and **business models** used to build our mission critical systems
- Need: Agile, secure and **highly assured mission computer** with adaptive defense techniques to minimize the system’s exposure to battlefield cyber threats and accomplish mission objectives

Army Approach

- The AvMC utilizes RDT&E funds from program offices, the Army S&T portfolio and the Army SBIR program to drive development and integration of these technologies
- Utilize an open source architecture, such as RISC-V, to develop a secure, trusted computing base and core operating systems
- Integrate with a formally verified open source microkernel
- Provide a path to leverage substantial DARPA investment in high assurance and secure computing
  - High-Assurance Cyber Military Systems (HACMS)
  - System Security Integrated Through Hardware and Firmware (SSITH)
- Collaborate between AvMC and ARL (Dr. Patrick Jungwirth)
- Sponsor a SBIR topic Army 19-103

Trusted Science and Technology, Inc.

- Project title: High Assurance Just-In-Time Secure System (HAJITSS) Architecture for Army Mission Critical Systems
- Relevant experience:
  - Design and implement a secure and resilient system architecture for UAS/UAV with successful flight demonstrations
  - Develop a secure, resilient system prototype to transition hardware/software integrated security solution to DoD stakeholders
  - Develop and mature the seL4-based **AFRL ARES Secure Kernel** (with security services) for U.S. specific interest
  - Lead US seL4-based solution development and support U.S. governing entities, e.g., DARPA, AFRL, and TCCoE

AFRL: Air Force Research Laboratory
ARES: Agile Resilient Embedded Systems

Trusted ST Relevant Experience

AFRL RI Flight Testing and Capstone Demonstration

AFRL RY/RI/RH UAV Testbed Integration and Demonstration

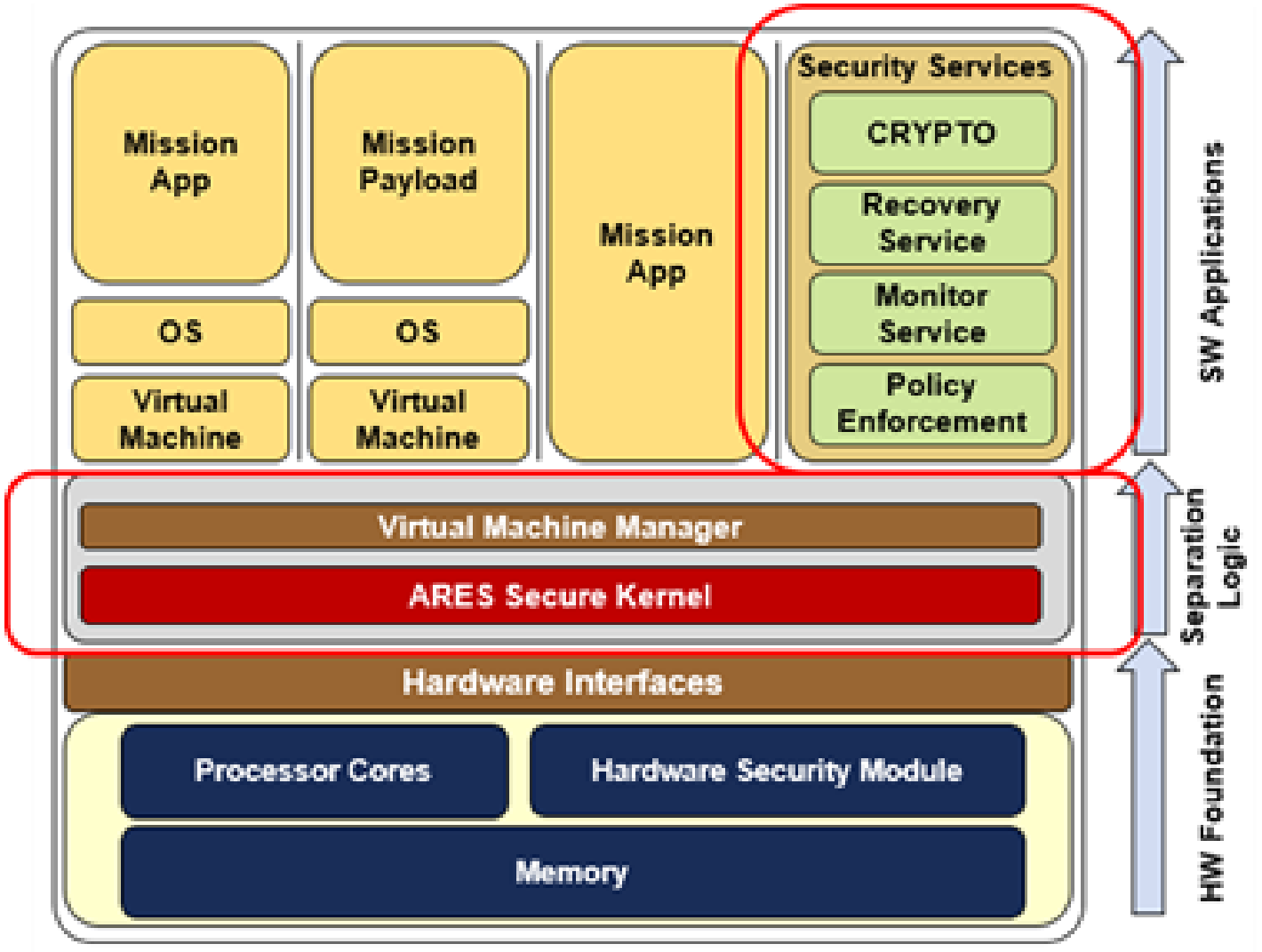
HAJITSS Need Statement

- A cyber-attack that exploits the mission computer allows the adversary unfettered access to system resources that jeopardize the mission or worse, turn the system against friendly forces
- Critical Need: Create proactive defense approaches, along with reactive techniques, for embedded systems on mission critical systems and platforms
- Goal: Develop an agile, secure and highly assured **mission computer** prototype with adaptive defense techniques
  - **Minimize** exposure to battlefield cyber threats
  - **Protect** against relevant attack categories
  - **Recover** from compromised execution or data corruption
  - **Maintain** and ensure mission success

HAJITSS Design Approaches

- HAJITSS architecture embodies core **security precepts**, modularity, and decoupled interfaces
- The component-based architecture (as opposed to being monolithic) allows for **modularity** and decoupled layers
- Componentization enables **function and data isolation**, testability, interface control, security, and damage limitation
- Loosely-coupled interfaces enables low-impact **rapid development and experimentation** of new hardware, operating systems, virtualization, and applications
- Recent open RISC-V & security extensions provide technical playground to develop a secure and resilient system ground up

HAJITSS Architecture



AFRL ARES Secure Kernel with security services

ARES: Agile Resilient Embedded Systems

HAJITSS Features & Benefits

- Highly assured, secure, and resilient computing infrastructure
- Open architecture and Open Source technology basis
- Use formally verified seL4 and RISC-V as SW & HW foundation
- Open standard-based interface design
- Just-In-Time (JIT) mission state / capability loading / execution
  - Mission function is loaded and executed only when necessary
  - Reduces the temporal attack surface for the mission capability
- HAJITSS significantly increases system security and resilience against cyber threats and system failures

HAJITSS Research Plan

Phase I	Phase II	Phase III
• Army system requirement analysis • Analysis of Alternative on design • HAJITSS architecture design update • Proof-of-concept implementation • Initial Testing and Evaluation (T&E) • Certification requirement analysis •Prepare Phase II R&D plan	• HAJITSS design update • Reference implementation at TRL5/6 • System engineering document generation • Thorough T&E with diverse cyber threats • Open source HAJITSS implementation • Certification roadmap and process creation • Look for transition partners	• Reference implementation at TRL 6/7 • Comprehensive T&E support • Capstone demonstration to stakeholders • Government support to create a transition program • Extended use cases development • Certification process support

HAJITSS Phase I Summary

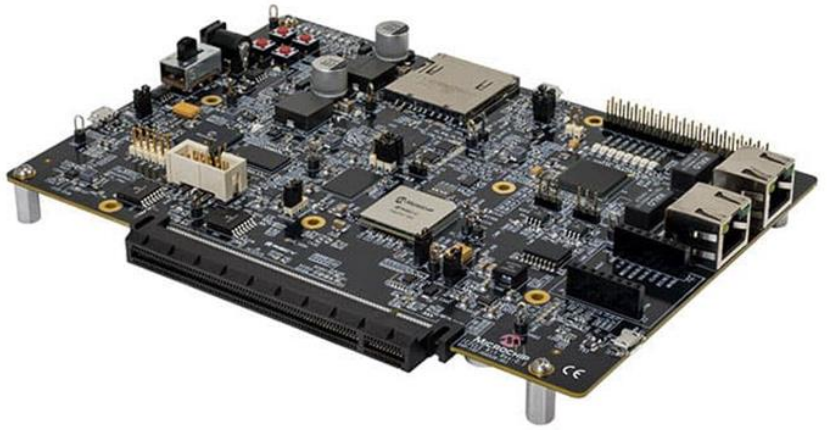
- Evaluation of RISC-V and available technologies
- Porting AFRL ARES Secure Kernel onto RISC-V
- Initial JIT design and development
- Phase I demonstration

RISC-V Evaluation

- Reviewed RISC-V based S&T programs and technology
  - DARPA SSITH program, Particular, CHERI on RISC-V
  - Interacted with key performers and assessed technology and roadmap
- Assessed RISC-V hardware and development schedule
  - SiFive, PolarFire system-on-a-chip (SOC)
  - Draper Lab Inherently Secure Processor (ISP), HENSOLDT Cyber GmbH, etc.
- Engaged stakeholders and potential transition partners
  - GE Aviation, Northrop Grumman
  - Draper Lab, Lockheed Martin

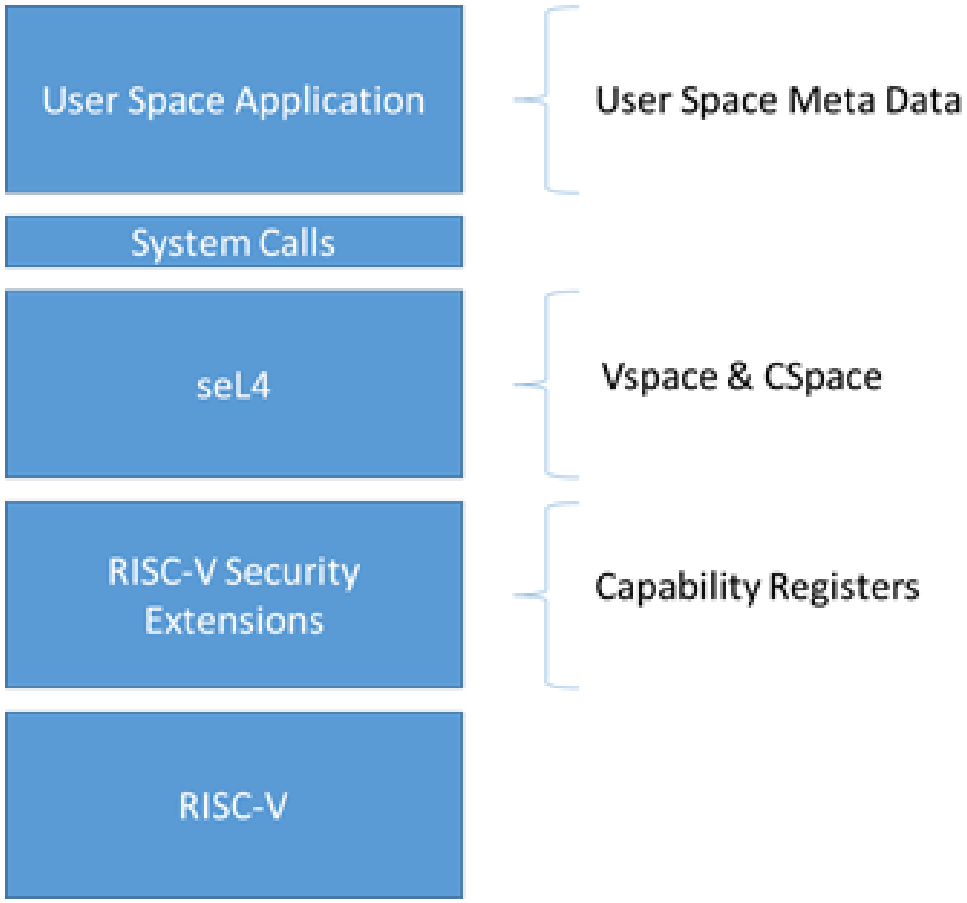
Porting Secure Kernel (SK)

- Chose a Polarfire SOC board as an initial RISC-V hardware platform
- Connected the seL4 (Secure Kernel) capability model with RISC-V security extensions
- Updated Secure Kernel security services to reflect the new ISA security extensions
- Adopted layer abstraction to support different RISC-V hardware platforms
- Will offer to the U.S. government and performers via TCCoE Private repository



Porting Approach & Status

- Connect RISC-V security extensions capability registers with seL4 / SK capabilities
- The register/capability mapping will be reflected in the user-space
- Hence, higher layers become aware of additional capabilities
- Update bootloader and sequence to load SK
- ARES SK runs on multiple cores in RISC-V Machine mode
- Porting SK to run in Supervisor mode is work in progress



Initial JIT Design & Development

- Explored different paths of the JIT capability
- Virtual Machine (VM) failover
  - Created the ability to stop and restart a VM
  - Allows the SW stack to dynamically switch between VMs
  - Serves as the technical foundation of JIT capability
- Process stop/restart
  - Allows a process (or application) to stop and restart
- Docker container support
  - Adding docker container functionality to the SW stack
  - Allows 1 or more VMs to host 0+ docker containers
  - Adds finer grain control over the JIT functions and tasks

Phase I Demonstration

- VM Failover
  - Dynamically configure a VM running on different ARM cores
  - Start/Stop/Restart a VM as necessary
  - Control information flow in/out of a VM
  - Technical foundation for Just-In-Time mission application operation running in VMs
  - VM failover on a RISC-V board is work in progress
- Process Recovery
  - Start/Stop/Restart seL4/ARES SK native processes
  - Add software resilience and configurability to minimize attack surface and maximize survivability

Way Forward

- Complete RISC-V Supervisor mode support
- Update the virtual machine manager (VMM) and mediate the data flow
- Update ARES SK security services with RISC-V security extensions and capability mappings
- Finish the Phase II workflow and system prototype
- Execute rigorous test and evaluation
- Conduct system assessment under Army oversight
- Transition the technology and prototype

Summary

- Leverage DARPA and AFRL demonstrated capabilities
  - HACMS & SSITH: seL4, RISC-V
  - ARES: AFRL Secure Kernel, Security and Resilience services
- Provide a Trusted Execution Environment (TEE) based on RSIC-V and Secure Kernel; needed for TCCoE
- Plan for rigorous T&E and capstone demonstrations using mission-relevant scenarios and platforms
- Share capabilities with other U.S. government and U.S.-based performers through TCCoE

Thoughts and Remarks

- Technology Base and Solutions
  - RISC-V (hardware base) and development on top of RISC-V
  - seL4 (microkernel) and AFRL Secure Kernel & execution environment (as a recommended architecture and solution)
  - Foreign investment, U.S.-based solutions, regulations and export control (Session #12)
- Functional Correctness Proof and Execution Evidence
  - From proof to practice (Dr. Fisher and Dr. Martin)
  - Execution evidence at binary level (Feb 1 Panel)
- Practical Gaps and Needs (Feb 2 Panel)

Questions?