1. Title: Modular Provably Secure High-Assurance Hardware Software Co-design (MPS-HAHSC)
   Presenter: Dr. Amit Vasudevan, Founder and CEO, Nirubi Technologies

Short bio: Dr. Amit Vasudevan is the Founder and CEO of Nirubi Technologies. He was previously a Senior Computer Scientist at the Software Engineering Institute, Carnegie Mellon University. His research areas and expertise spans secure systems, trustworthy computing, operating systems, and applied formal methods. He has over 17 years of R&D expertise at Carnegie Mellon having successfully fundraised and led projects funded by NSA, DoD, Industry, and NSF. He has published two books on trustworthy computing, holds multiple seminal patents in the area of trustworthy computing, and has widely published in the area of cybersecurity. He is also an ardent open-source kernel developer and contributor. Dr. Vasudevan holds a Postdoctoral Fellowship from CyLab/CMU, PhD and MS degrees in Computer Science and Engineering from UT Arlington, and a BS in Computer Science from Bangalore University, India. He was named University Scholar twice in a row during his PhD.

Abstract: Current generation Cyber Physical Systems (CPS), including aviation systems, were not developed with strong computer security requirements. Past, current and future cyber threats need to be countered. Embedded system designs are typically based on commodity hardware and software optimized exclusively for size, weight, and power – leading to critical cyber vulnerabilities that can devastate safety and mission effectiveness. This has also led to the unsustainable “Perimeter, Patch, Pray” Information Assurance strategy that is impractical for fielded critical CPS.

We present Modular and Provably Secure high-assurance hardware/software co-design (MPS-HAHSC) based on the open-source RISC-V instruction set architecture and high-assurance (formally proven) kernels such as seL4. We leverage a mathematical reasoning framework (that encompasses both hardware and software layers) and software verification framework to harden the assurance of the formally verified seL4 kernel to obtain a Future Vertical Lift (FVL) system architecture that can support robust, mathematically proven end-to-end high-assurance. With the advent of RISC-V, this can also be done practically and performant with all the available tooling to support rich CPS mission applications.

2. Title: SWITCHing to CHERI -- Smoothing the Path to Cyber Security
   Presenter: David Musliner

Short bio: Dr. David Musliner is a senior Technical Fellow at Smart Information Flow Technologies (SIFT), specializing in AI for highly autonomous systems ranging from cyber security reasoning to uncrewed ground, air, and space vehicles.

Abstract: Two types of cyber security flaws are responsible for the vast majority of all high-severity software vulnerabilities: memory safety failures and over-privilege. CHERI (Capability Hardware Enhanced RISC Instructions) extends conventional processor Instruction-Set Architectures with architectural capabilities to enable fine-grained memory protection and highly scalable software compartmentalization. When used correctly, the CHERI enhancements allow a computer's hardware to reliably prevent memory safety flaws and enforce least-privilege compartmentalization. However, developing software that uses the CHERI enforcement mechanisms remains a challenge, requiring highly-skilled developers with detailed CHERI experience. SIFT's new SWITCH project will mature the tool chain required to build CHERI-compatible software, so that porting existing software is much easier, and developing new software is no harder than on other common computing platforms. This will lower the bar for adoption of memory-safe, capability-enforcing systems, catalyzing dramatic improvements in software security.

3. Title: Kry10 OS - Trustworthy, Dynamic, and Easy To Use
   Presenter: Ihor Kuz

Bio: Dr Ihor Kuz is a principal operating system engineer at Kry10, helping develop the Kry10 OS and Platform. Ihor has previous experience leading the team developing the seL4 microkernel, and has been involved with seL4 for as long as it's been around. Ihor is a member of the seL4 Foundation's technical steering committee (TSC). In the past he has been an associate professor at UNSW in Australia and has taught operating systems, distributed systems and Erlang there for many years.

Abstract: We present and demonstrate the Kry10 Operating System (KOS). KOS is a new operation system platform that we are developing at Kry10. It is based on seL4 and builds up a trustworthy foundation, utilising seL4's verified properties to provide a powerful, dynamic, and easy to use OS platform for securing critical systems. Besides providing secure isolation and communication between OS components, KOS also provides: dynamic software update, virtualisation and containerisation support, support for various runtime environments (including freestanding and hosted (semi-POSIX) C, Elixir/Erlang, and Rust), a management subsystem, remote graphical interface, fleet management server, and more. Furthermore, we are in the process of adding multicore support and formally verifying the core KOS runtime system.
> In this presentation we will introduce Kry10 and its mission, present the Kry10 OS architecture, and provide demonstrations of some of its key features, incuding its development environment, dynamic updates, visualisation, and remote management as well as some of the systems we have developed using it.

4. Title: MonT: Toward Real-time Model Checking
   Presenter: Sukarno Mertoguno

Short Bio: Dr. J. Sukarno Mertoguno is a research professor at the SCP, GIT. His research covers broad area of computing systems (hardware & software), cybersecurity and machine learning. He previously served as Chief Innovation Officer for the ICSD of GTRI. Before joining GTRI, he managed basic and applied science research in cyber security and complex software for ONR where he developed several novel concepts. Prior to ONR, he was a system & chip architect and an entrepreneur in Silicon Valley. He received his Ph.D. in electrical engineering from SUNY-Binghamton.

Abstract: Formal methods is rigorous mathematical representation of an abstracted behavior of a program, usually derived from the program formal specification. Formal methods can verify & guarantee software behavior in a manner that testing cannot. Generally formal verification is performed offline before a program is deployed.

Formal model can also be used to enforce program behavior, dynamically, in the form of dynamic (formal) model checking. Dynamic formal model checking can potentially be used to enforce the behavior of COTS executable. Dynamic formal model checking requires mechanism for executing the formal model synchronously with the execution of the COTS program.

Georgia Institute of Technology's CSAFA labs developed MonT, a hardware assisted real-time/online program execution monitoring at instruction level granularity at the nominal speed of the main (observed/monitored) processor. MonT has been demonstrated to be capable of stopping 11 CVEs of various types (buffer overflow, integer overflow, use after free), before the exploit can complete.

MonT demonstrates that online monitoring of program execution at instruction level granularity and at speed is indeed feasible and practical.

As formal model can generally be transformed into state machine, MonT can support formal model checking, in real-time without requiring any software instrumentation.

5. Title: Towards Comprehensive Memory Safety Using Memory Safety Validation
   Presenter: Trent Jaeger, UC Riverside

Bio: Trent Jaeger is a Professor in the Computer Science and Engineering Department at the University of California, Riverside. Trent's primary research interests are in improving trust in operating systems and software security. He has published over 180 refereed research papers and the book, "Operating Systems Security," which has been taught in universities worldwide. Trent has made significant security contributions to the open-source security community, particularly for the Linux kernel. His research has been recognized with the ACM SIGSAC Outstanding Contributions Award in 2020. He is an ACM and IEEE Fellow.

Abstract: Despite a variety of research advances in memory safety defenses and safe languages, we still see critical challenges in achieving comprehensive memory safety within a reasonable cost. Our research proposes to address this challenge by making memory safety explicit in C programs by identifying the objects whose memory operations must satisfy all classes of memory safety (i.e., provably "safe" objects). Our experience has been that a large fraction of heap (over 77%) and stack objects (over 85%) can be validated as "safe," and can be protected from memory errors efficiently by coarse-grained isolation (NDSS 2022 and ACM CCS 2024).

In this talk, we discuss how to use memory safety validation to address the remaining unsafe operations. First, we will discuss how memory safety validation can help use reduce enforcement overheads by identifying what defenses are necessary and enabling selection of the most efficient defense for each case (both USENIX 2024). Second, we will examine how memory safety validation can help memory safety enforcement in Rust, such as removing threats caused by possible memory errors in unsafe Rust code and (in progress). We will discuss how memory safety validation plus intelligent combination of defenses could produce efficient, comprehensive memory safety protection.

6. Title: LITESHIELD: A Lightweight Userspace μKernel Architecture for Secure Container Isolation
   Presenter: Hui Lu

Short bio: Dr. Hui Lu is an Assistant Professor at the University of Texas at Arlington (UTA). His research focuses on operating systems, virtualization, cloud computing, and network systems. He has collaborated with top labs like Intel Labs, HPE, and AFRL, and holds a Ph.D. from Purdue University. Prior to his doctorate, he was a performance engineer at Intel's Asia-Pacific R&D Center and earned his bachelor's and master's degrees from Shanghai Jiao Tong University.

Abstract: We introduce **LITESHIELD**, a lightweight and secure isolation framework for containerized applications for commodity kernels (e.g., Linux). By decoupling traditional guest kernel functions into modular userspace microkernel (μkernel) services, LITESHIELD reduces the reliance on host kernel syscalls from 300+ to just 28, achieving a thin user-to-host interface comparable to virtual machines (VMs). Leveraging fast, shared-memory-based inter-process communication (IPC), LITESHIELD eliminates the need for hypervisors and minimizes costly kernel context switches, reducing the virtualization stack's complexity and attack surface. Its modular design enables seamless integration of specialized userspace services, such as networking and filesystems, while maintaining compatibility with legacy Linux applications through dynamic syscall interception. Performance evaluations show that LITESHIELD matches or exceeds traditional container and VM-based isolation mechanisms in syscall latency, I/O efficiency, and networking throughput. This architecture is particularly suited for cloud-native environments, enabling secure and efficient multi-tenancy while supporting diverse application needs with composable μkernel services. By reimagining container security and performance, LITESHIELD offers a promising alternative to traditional hypervisor-dependent approaches.

7. Title: Unikernels: A DevSecOps alternative to containers
   Presenter: Robbie VanVossen

Short Bio: Robbie is an employee of DornerWorks, has 12 years of experience in embedded hypervisors, and 10 years of experience with the seL4 microkernel. His work and leadership led to the development of aarch64 virtualization support across the seL4 ecosystem, both in the microkernel and the user-space libraries. He has also developed multiple projects using the Rust language, including embedded implementations of device drivers and applications. He has strong interests in embedded security, virtualization, seL4, and Rust.

Abstract: A Unikernel is a specialized image that contains a single application, a minimal kernel, and the drivers, stacks, and libraries needed for that application. They are mostly used on hypervisors and trade general purpose operation for very small images which results in a smaller attack surface, faster boot times, and lower memory utilization. Since many Unikernels can run on both hypervisors and Unix it has similar DevSecOps benefits as containers, such as being able to easily test applications before deploying them as virtual machines on a target, but with the added benefit of requiring less resources and shorter boot times.

In this talk, I will discuss the benefits and drawbacks of using Unikernels in high-assurance security systems and contrast their use against containers for DevSecOps. I will discuss some modern Unikernels which have their own build/orchestration tools and package repositories and how these tools can even integrate with containers. I will also demo some Unikernels running on a host machine and running on top of the secure seL4 hypervisor on embedded target platforms and talk about the porting effort. The talk will conclude with a discussion on next steps for Unikernels in high-assurance systems.

8. Title: Secure High-Assurance Aberdeen Architecture RISC-V Complier and Softcore (SHAAARCS)
   Presenter: Michael Doran

Bio: Michael Doran brings over a decade of experience as an embedded software engineer, specializing in high-assurance designs across ARM, x86, and RISC-V platforms. Beyond his professional work, he is pursuing a Ph.D. in Computer Engineering at Georgia Tech, focusing on cybersecurity for hardware devices for critical infrastructure. As a Cyber Warfare Officer in the Army Reserve, Michael has firsthand experience executing missions in this domain, providing him with unique insight into real-world cyber threats. His combined expertise in engineering, research, and military operations drives his commitment to advancing secure computing for mission-critical systems.

Abstract: Modern cybersecurity relies on a layered defense approach, incorporating least privilege and continuous monitoring. However, microprocessors prioritize performance over security, lacking hardware-level enforcement of these principles, leaving them vulnerable to attacks like Spectre and Meltdown. The Aberdeen Architecture addresses these gaps by introducing hardware state machines that enforce four security policies: (1) instruction execution, (2) page memory access, (3) control flow integrity, and (4) data flow integrity. These state machines operate independently of the processor pipeline, employing local policies for fine-grained instruction-level security and global policies to limit fault propagation, balancing security and performance. This work explores integrating instruction mediation—an element of the Aberdeen Architecture—into an open-source RISC-V processor (e.g., RocketChip). By evaluating RISC-V toolchains for implementing a two-level tagged architecture, necessary modifications were identified for user-defined tags and security policies. Embedding a hardware state machine into the processor and mapping out toolchain adaptations provides a blueprint for realizing the Aberdeen Architecture. This approach strengthens security at the hardware level, mitigating vulnerabilities and laying the foundation for more resilient computing systems.

9. Title: Secure Robotic Operating System (seROS)
   Presenter: Nathan Studer

Bio: Nathan Studer has worked on high assurance software and custom logic applications for 20 years with a focus on embedded virtualization. He obtained his undergraduate degree in Electrical Engineering from Calvin College and has a master's degree from Michigan State University in Computer Science. Nathan is a Technology Strategy Lead for DornerWorks leading the team in designing systems that meet customer's security requirements using an appropriate blend of software, hardware, and/or software that emulates hardware. His combined expertise in embedded software, custom logic, and cybersecurity help him deliver practical solutions to the Department of Defense's difficult cybersecurity challenges.

Abstract: ROS (Robotic Operating System), a modular componentized architecture for robot applications, has made it possible to quickly develop and deploy systems utilizing autonomous or human guided robots. Recognizing the benefits of this approach, the Army has created a collection of military specific ROS components to enable autonomy features now called the Army Robotic Common Software (ARCS). While ROS components developed by the Army themselves can be thoroughly vetted, visibility into third party components may be lacking. Furthermore, ROS depends on many services included in a full Linux distribution to function properly leaving a large attack surface.

To address these challenges, the Cybersecurity for Robotic & Autonomous Systems Hardening (CRASH) Joint Capabilities Technology Demonstration was created to develop a comprehensive cybersecurity software solution tailored for robotic and autonomous systems. This presentation will provide an overview of this program, how a trusted microkernel/hypervisor such as seL4 can be used to retrofit security features, and suggest a roadmap for future work.

10. Title: Fine-Grained Security Control through Combined Memory Access Protection and Isolation
    Presenter: Thomas Wahl

Bio: Dr. Thomas Wahl is a Senior Principal Scientist at Trusted ST. Dr. Wahl received a PhD degree from the University of Texas at Austin in 2007. His expertise lies in model-based formal approaches, specifically in foundational and practical aspects of formally modeling program or system artifacts, and in the scalability of model-based analyses. Particularly relevant to the proposed effort, Dr. Wahl is currently working as the PI for the DARPA INGOTS BAA on vulnerability and exploit modeling, and leading Trusted ST's effort for the DARPA V-SPELLS BAA on code identification and software analysis for security hardening.

Abstract: This presentation reports on our experience with implementing fine-grained software security control, by combining intra-process memory access protection through Capability Hardware Enhanced RISC Instructions (CHERI) architectures like ARM Morello with inter-process protection through memory isolation. The experience stems from work supported by DARPA's STO and MTO offices on securing a Swarm Unmanned Aerial Vehicle (SUAV) application, but it has wide-ranging implications beyond the Swarm effort. We will briefly introduce forms of fine-grained security control and provide some background on CHERI, ARM Morello, and seL4 to set the stage. We will share some key lessons learned from work related to porting applications to the Morello system and building security and resilience on top of it, which may benefit communities working along similar lines. We will conclude with some future visions and steps, such as automated Cherification and improved fault handling, which, in our experience, are essential to improve the adoption prospects of combined Capability hardware and memory isolation. The ARM Morello architecture on top of which this work was performed is by now a fairly mature technology; it is time to undertake efforts to make it more widely accessible and practically viable.

11. Title: Hardware Attacks and Defenses for High Assurance Systems.
    Presenters: Dr. Sean Zhou, Dr. Leonid Meyerovich, Dr. Jason Li

Bio: Dr. Sean Zhou is a Principal Scientist at Trusted ST specialized in hardware security, computer architecture, and hardware-software codesign. Before Trusted ST, Dr. Zhou has worked at Intelligent Automation Inc. (IAI) and University of Hawaii. He has been PI and technical lead for more than 20 SBIR and BAA programs funded by DARPA, DOD, NASA, DOE, DHS. Particularly relevant to the proposed effort, Dr. Zhou is currently working as the PI for one DARPA SBIR project on hardware side channel attacks, two Army SBIR projects on secure hardware architecture, and one DHS SBIR projects on hardware-assisted malware detection framework.

Abstract: This presentation reports on our experience with evaluating hardware attack and defense techniques for high assurance systems. One the offense side, we evaluated both analog side-channel attacks and microarchitectural side-channel attacks, and their impacts to secure architectures including Intel SGX, ARM Morello, and seL4 microkernel. On the defense side, we are developing hardware-software codesign architecture by extending RISC-V cores and seL4 microkernel for security isolation. Such security enhancement can systematically integrate security policy enforcement and isolation at both the software and hardware levels against a variety of hardware side channel attacks. We will briefly introduce state-of-the-art hardware side channel attack techniques and provide some background on secure hardware architectures. We will discuss our experiences on RISC-V Instruction Set Architecture (ISA) extensions and integration with seL4 microkernel on FPGA platforms. We will also present our works on hardware-assisted malware detection techniques by leveraging hardware performance counter (HPC) traces to detect ransomware attacks in real time. We will conclude with our visions on secure hardware architecture design including hardware Trusted Computing Base (TCB) and hardware level security policy enforcement for high assurance systems.

12. Title: Mobile Software Understanding for Compound Vulnerabilities Using Knowledge Graph (KG) and Graph Neural Network (GNN)
    Presenter: Fei Sun

Short Bio: Fei Sun is a Cyber Systems Engineer and the Principal Investigator for the Cyber Internal Research and Development project at Leidos Innovations Center. Before joining Leidos, Fei was a Robotics Systems Engineer and Principal Investigator in the Artificial Intelligence and Autonomy Innovation Center at MITRE.

Abstract: Mobile applications present unique security challenges due to dynamic execution environments. Traditional vulnerability analysis often struggles with obfuscated or deeply embedded security threats in mobile software.

Our approach leverages AI-based analytics and continuous learning of vulnerability, patch, advisory, and threat data to improve modeling and validation of compound vulnerabilities. Applying our novel Network Path Traversal (NPT) technique to augmented vulnerability analysis and modeling enables accurate, automated exposure and quantification of attack surfaces. Furthermore, we extend existing vulnerability research to refinement, synthesis, and forecasting of mobile exploit sequences through a continuous-learning workflow that feeds validation results back into the GNNs.

To gain a deeper understanding of compound vulnerabilities, we use KG to model complex interdependencies between system components to expose attack surfaces and advance the understanding of effectiveness for individual vulnerability in the sequence. We use GNN foundational model for AI/ML-based analysis, using inference development to gain insights into compound vulnerabilities. Using LLM, we further enrich the process of rapid ingestion and actionable threat reports, alerts, and patch advisories. Finally, a continuous learning stream consumes all feedback from any point in the process and feeds into the GNN, for continuous quantitative improvement in analysis efficiency and accuracy.

13. Title: Cross-Domain Solutions in Safety-Critical Military Ground Vehicle Applications

Presenter: Leonard Elliott

Bio: Mr. Elliott has been an electrical engineer in the Combat Capabilities Development Command (DEVCOM) Ground Vehicle System Center (GVSC) Vehicle Electronics & Architecture (VEA) Division since September 2010. He is currently the Technical Specialist for the Embedded Systems and Software Branch and has led GVSC's effort to mature and develop software and networking aspects of the PEO-GCS Common Infrastructure Architecture (GCIA). He supports research and development in the areas of middleware, open system architectures, and safe/secure embedded systems. Mr. Elliott is a Certified Information System Security Professional (CISSP), holds a B.S. degree in Electrical and Computer Engineering from the University of Massachusetts Amherst, and an M.S. in Electrical and Computer Engineering from Worcester Polytechnic Institute.

Abstract: Modern vehicle architectures are becoming increasingly integrated as more and more vehicle functions become sofware-defined. This is especially true in military ground vehicles where the vehicle architecture supports not only mobility but also lethality, survivability, and robotic functionalities. Cross-Domain Solutions (CDS) are highly specialized components which enable strong isolation and segmentation of communications networks and computing resources in enterprise and tactical systems. CDS assurance requirements are almost exclusively expressed in terms of security, however these items are also used as key enablers in cyberphysical system architectures, supporting both safety-critical and security-critical functions. This presentation will examine the subtle yet crucial differences in the way that CDS must support safety-crtical versus security-critical capabilities as well as exploring emerging techniques for deploying CDS in safety-critical weapon systems.

14. Title: Exploring Next-Generation Hardware Platforms to Harness the Benefits of CHERI Experimental Implementations

Presenter: Hui Zeng

Bio: Dr. Hui Zeng is Director at P & J Robinson Corporation (PJR). Dr. Zeng received his Ph.D. in Electrical Engineering from the University of Maryland at College Park. He has extensive experience in networks, security, communications, and applications. His research interests include cyber security, AI/ML, signal processing, networking, wireless network, and tactical cloud. At PJR, Dr. Zeng has played a pivotal role in technology development, team management, business growth, as well as in shaping technical vision and creating strategic roadmaps. Prior to PJR, he was Associate Director (Acting Director) of the Networks & Security division at Intelligent Automation, Inc.

Abstract: Despite rigorous software testing, exploitable software vulnerabilities frequently arise from memory issues. While memory-safe languages like Rust offer greater inherent memory safety, they do not provide absolute protection against such vulnerabilities and cannot soon replace the extensive body of C/C++ code. A hardware approach called CHERI (Capability Hardware Enhanced RISC Instructions) has shown promise in effectively addressing multiple memory safety issues using an ARM Morello prototype. However, several hardware enhancements are still needed to fully harness the benefits of CHERI's experimental implementations.

To address this need, P&J Robinson Corporation (PJR), along with its subcontractors, is exploring the design and development of next-generation hardware platforms to harness the benefits of selected CHERI hardware prototypes like ARM Morello. The goal of our effort is to improve cost efficiency, compactness, modularity, scalability, interoperability and supply chain reliability while maintaining the same "CHERI" features for enhanced security. Towards this goal, we have achieved preliminary results in the testing and evaluation of off-the-shelf PCIe cards integrated with the current Morello board, and the initial design of an extension board to support existing and evolving CHERI boards.

15. Title: Building a Memory Safe Software Ecosystem with CHERI

Presenter: Brooks Davis, SRI

Bio: Brooks Davis is a Principal Computer Scientist at SRI. Since 2012 he has worked on the CHERI project and leads development of the CheriBSD operating system. Prior to joining SRI, he worked on high-performance computing and networking at The Aerospace Corporation. He holds a Bachelor's Degree in Computer Science from Harvey Mudd College.

Brooks has been a member of the open-source FreeBSD operating system project since 2001 and has served on the project's elected core team. He is also a Visiting Research Fellow at the University of Cambridge Department of Computer Science and Technology (Computer Laboratory).

Abstract: CHERI makes it possible to impose strong memory safety and compartmentalization on existing, memory unsafe C and C++ code bases with reasonable effort. Ported software includes hypervisors (seL4), operating systems (Linux, FreeBSD), desktop environments, and even the Chromium web browser. The corpus of code includes some of the most complex open source software in the world consisting of tens of millions of lines of code. While the effort to port it was non-trivial, it pales in comparison to the effort required to write it or (in many cases) even maintain in. With the imminent availability of commercial hardware platforms, CHERI gives us hope for the billions of lines of C and C++ software underpinning critical systems today.