BIRD - ICRDE Project Extension:

Program Execution Monitoring & Control

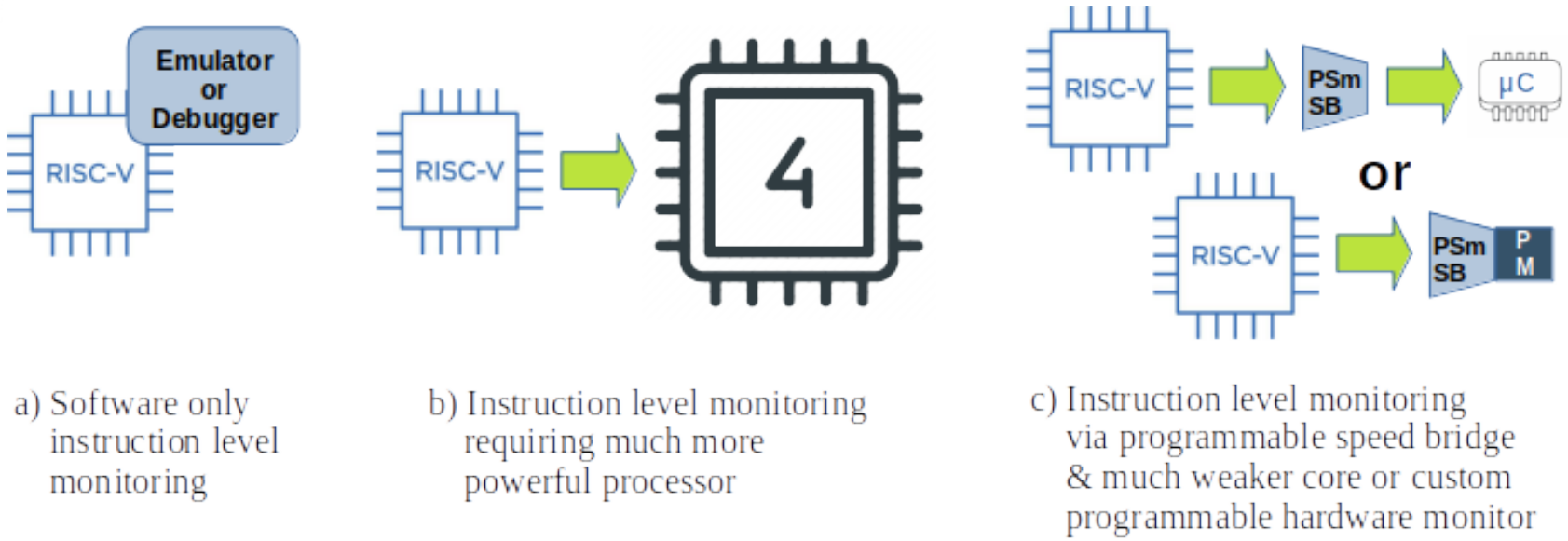
J. Sukarno Mertoguno

Georgia Tech College of Computing
School of Cybersecurity and Privacy

Problem Statement (motivation)

- Monitoring at Instruction Level Granularity enable real-time detection and mitigation of exploits in progress, before they can achieve foothold and compromise the system.
- Software only approach requires execution the software under analysis (SUA) in either emulator or debugger -- not practical for deployed/production software/applications.
- Practical software only monitoring:
  - operates at higher level abstraction/granularity or
  - requires insertion of software instrumentation and security checks and
  - often loses the opportunity to stop exploits in their track.
  - Memory forensic is often required for analyzing and discovering the root cause and entry point of exploits, postmortem.

Problem Statement (motivation)



a) Software only instruction level monitoring

b) Instruction level monitoring requiring much more powerful processor

c) Instruction level monitoring via programmable speed bridge & much weaker core or custom programmable hardware monitor

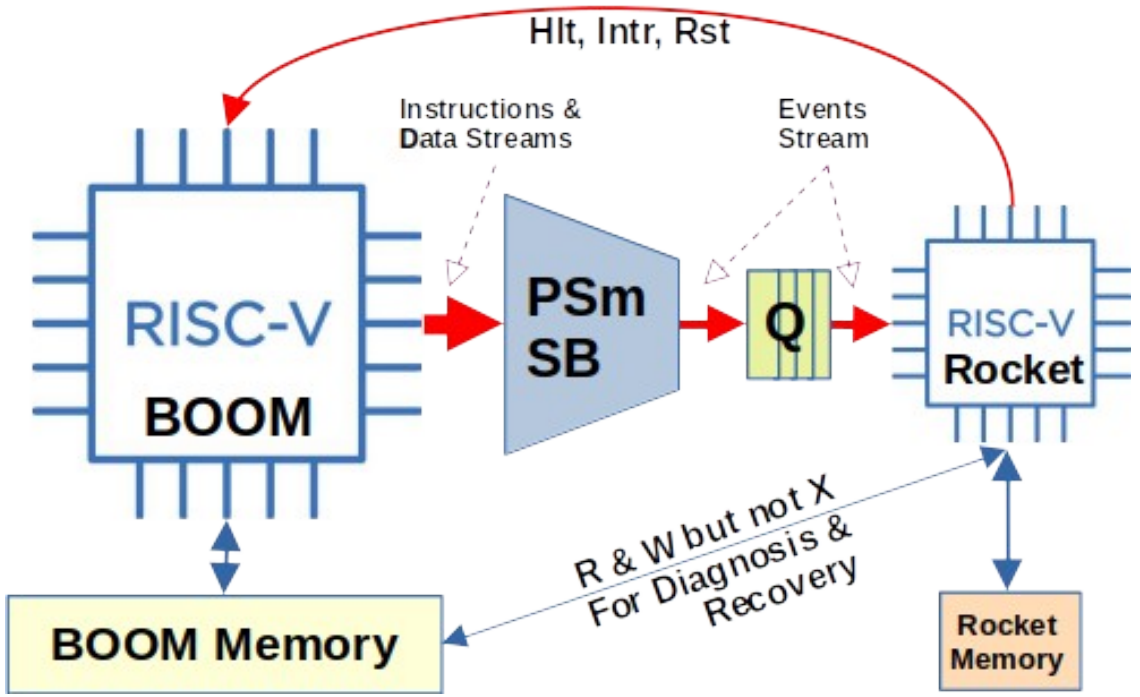
- A hardware solution, using a processor to directly monitor instruction streams of target processor requires the monitoring processor to be significantly (an order) more powerful than the target/monitored processor – highly inefficient

Needs monitoring process with transition rate lower than monitored process (processor clock)

Project Proposal (Solution to the problem)

PEMon: a hardware accelerator for realtime instruction level monitoring:

- Programmable state machine provides first level detection, and
- Enabling weaker monitoring processor to monitor much more powerful target/monitored processor – Efficient
- Minimal (No) impact on monitored processor throughput and latency, applicable for realtime application
- No modification & instrumentation to monitored software



PEMon monitors and controls program execution efficiently with minimal performance impact

Potential Application Areas

General computing security enforcement:

- CFI enforcement
- Input sanitation
- Memory boundary enforcement
- Noisy side-channel-attack detection & mitigation
- Etc.

DARPA SSITH seedling has proven the efficacy of PSmSb

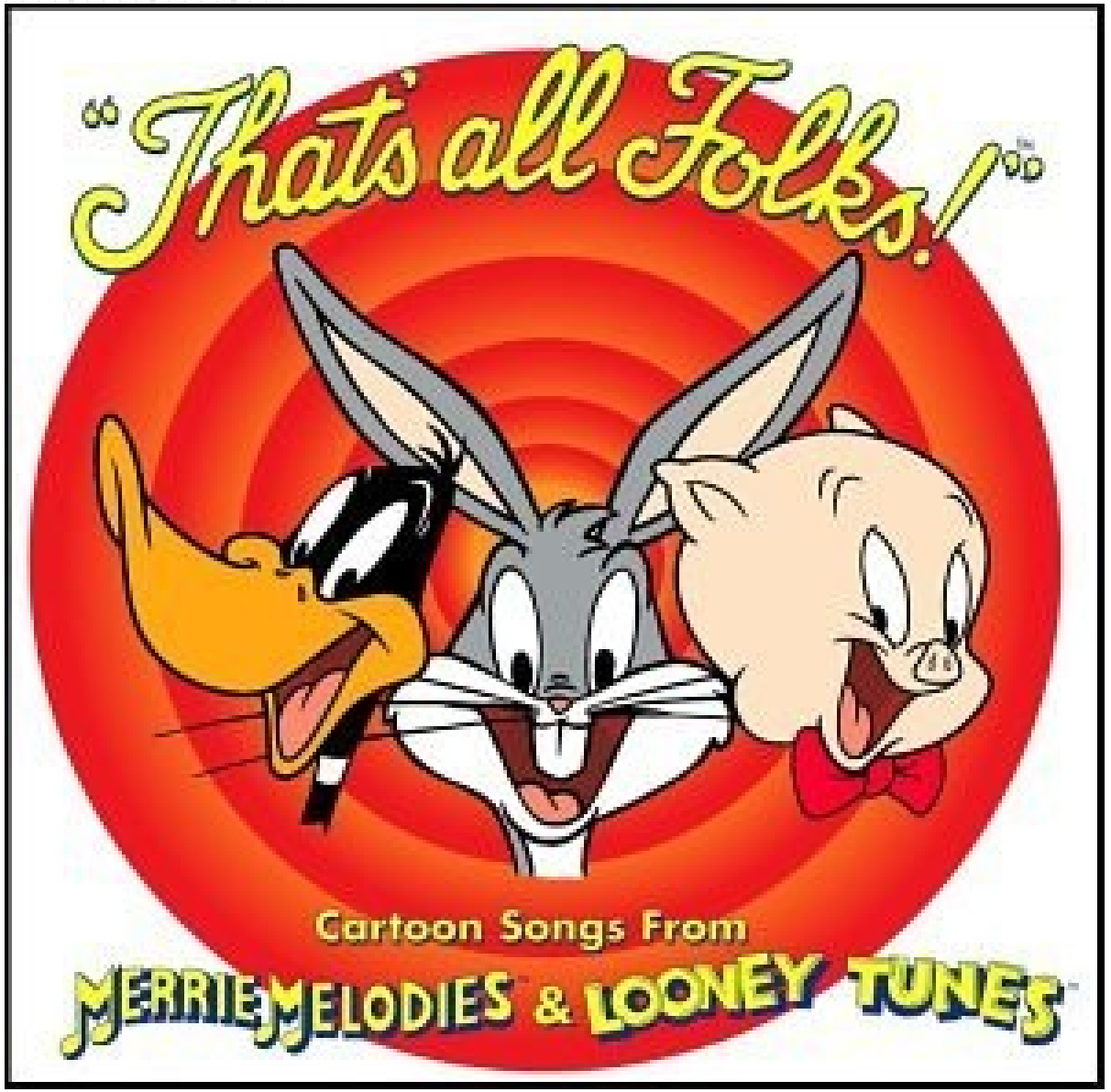
Realtime controller (CPS) specific security and resilience:

- Realtime Model checking
  - Formal model
  - Detailed digital twin model
  - Critical skeleton only simplex architecture
- On-the-fly checkpointing; progressive capture of monitored systems state
- Cyber attack & errors recovery/repair (resilience)
- Reconstitution from known good state (resilience)

Cyber-protection for hard realtime systems. Stopping attack on its track

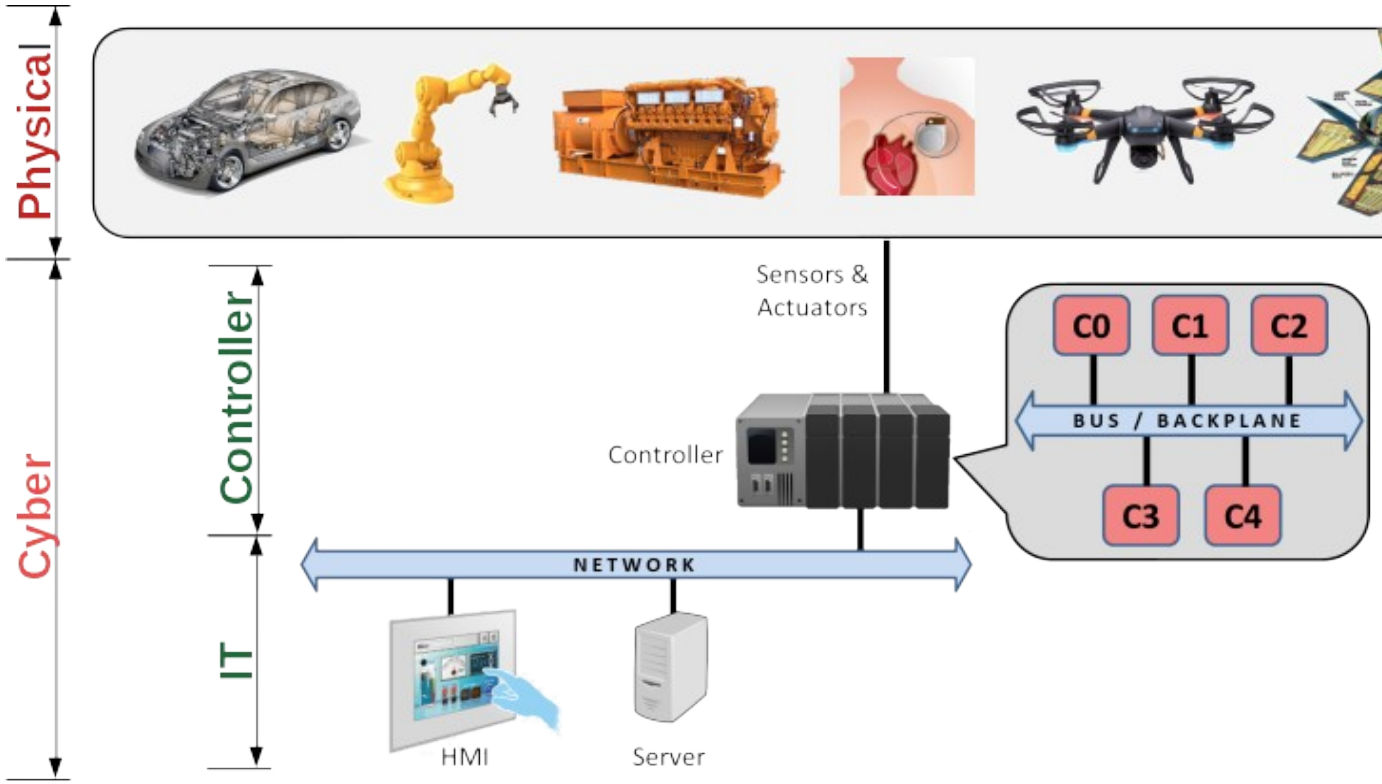
References

- C. Yagemann, M. A. Noureddine, W. U. Hassan, S. Chung, A. Bates, and W. Lee, “Validating the integrity of audit logs against execution repartitioning attacks,” in Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security, ser. CCS ’21
- C. Yagemann, M. Pruett, S. P. Chung, K. Bittick, B. Saltaformaggio, and W. Lee, “ARCUS: Symbolic root cause analysis of exploits in production systems,” in 30th USENIX Security Symposium (USENIX Security 21), USENIX Association, Aug. 2021, pp. 1989–2006
- C. Yagemann, S. P. Chung, B. Saltaformaggio, and W. Lee, “Automated bug hunting with data-driven symbolic root cause analysis,” ser. CCS ’21, Virtual Event, Republic of Korea: Association for Computing Machinery, 2021
- Sukarno Mertoguno, “Cognizant engines: Systems and methods for enabling program observability and controlability at instruction level granularity”, 2008 https://www.freepatentsonline.com/y2010/0107252.html.
- Rocket Chip Generator https://bar.eecs.berkeley.edu/projects/rocket_chip.html
- Welcome to RISCV-BOOM’s documentation https://docs.boom-core.org/en/latest/
- Meng Xu, Kangjie Lu, Taesoo Kim, and Wenke Lee, “Bunshin: Compositing Security Mechanisms through Diversification”, 2017, arXiv preprint arXiv:1705.09165
- J.G. Rivera, A.A. Danylyszyn, C.B. Weinstock, L.R. Sha, M.J. Gagliardi, “An Architectural Description of the Simplex Architecture”, CMU/SEI-96-TR-006, https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=12521



CPS → Physics Rules

Device Level Security: Robustness from the Ground Up



- Effect of Compromised Device:
  - Lie to monitors – doing one thing, reporting another (e.g. Stuxnet)
  - Transport layer (communication) security **irrelevan**t – protecting the attacker

- Cyber Attack Resilience
  - Relying on CPS (controller) properties to tolerate direct cyber attack
  - Agnostic to the specificity of the attack (malware)
  - Requires multi-factor authentication for firmware update

- Side-Channels Monitoring
  - Ensuring firmware/software performing as expected
  - Cannot easily be circumvented by attacker (malware)

Building Resilience System from Resilient Components

Cyber Security Triad – CIA

- **Confidentiality**
  - protection of information from unauthorized access.
  - CPS: no-information leaks
  - Common techniques: Encryption
- **Integrity**
  - information is kept accurate and consistent unless authorized changes are made
  - CPS: provides correct and proper operation/service (as expected)
  - Common technique: Authentication, Hash/integrity checking
- **Availability**
  - information is available when and where it is rightly needed
  - CPS: Service availability
  - Common technique: Robust & Resilience operation

The Importance of C, I & A can be evaluated from the type of data/information, physical dynamics and needs/requirements