An Inherently Secure Computer

This material is based upon work supported by the Assistant Secretary of Defense for Research and Engineering under Air Force Contract No. FA8721-05-C-0002 and/or FA8702-15-D-0001. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Assistant Secretary of Defense for Research and Engineering.

Dr. Hamed Okhravi

MIT Lincoln Laboratory

5 March 2018

Motivation

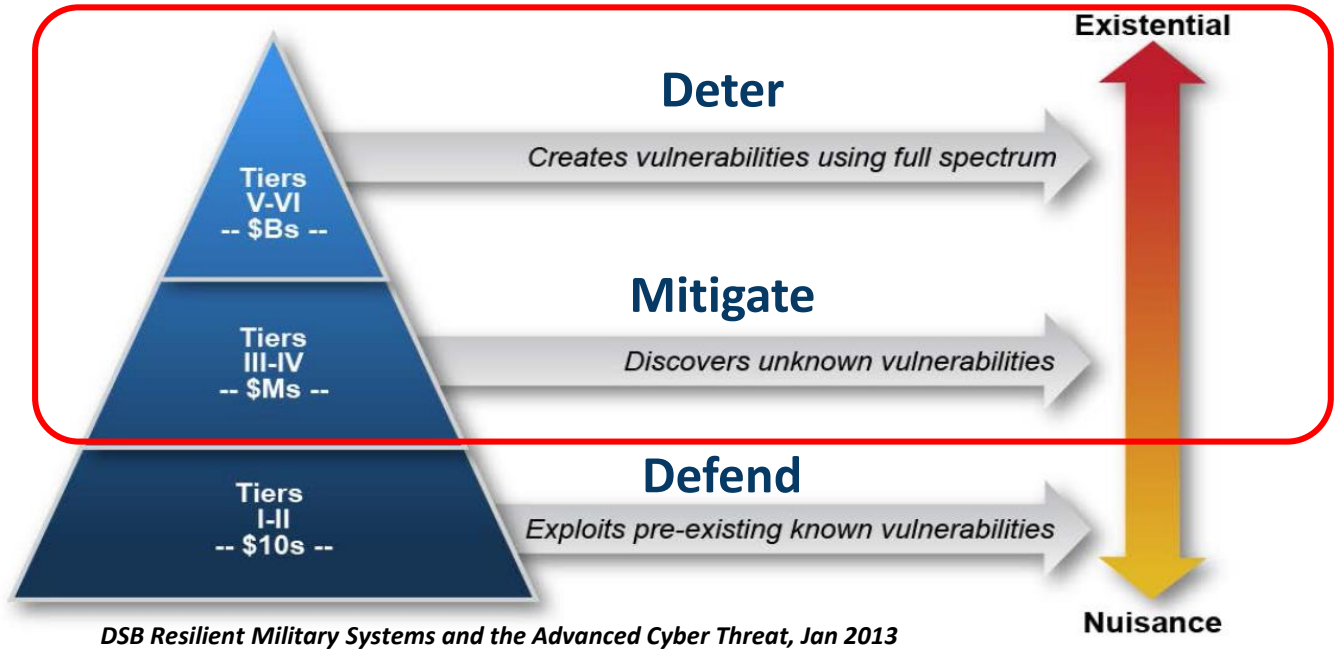
Cyber requirements for mission success:

- Reliable, timely response
- Secure communications over untrusted networks
- Secure backend computing infrastructure

Example problems: Industrial control systems, military, voting systems, etc.

Cybersecurity Challenges

- Prevalent use of COTS/GOTS components in DoD systems
- None of these components has security built in
- Cyber defenses are often patchwork with intangible benefits
- Protections are often limited to best practices (i.e., hygiene)
- Little protection against Tier III-VI adversaries

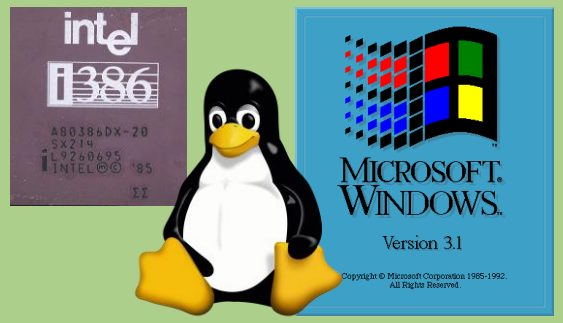


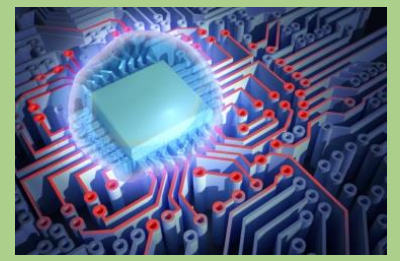
DSB Resilient Military Systems and the Advanced Cyber Threat, Jan 2013

Tiers based on dollars invested by attackers

Secure Systems Lay of the Land

1960s	1970s	1980s	1990s	2000s	2010s	2020s
Inherently Insecure Systems Add-on Defenses				Inherently Insecure Systems Built-in Defenses		Inherently Secure Systems
MULTICS, Project MAC	PDP-11	x86, Windows, Linux		Intel Skylake, DARPA CRASH, and MRC		Inherently Secure Computer
Time-sharing operating system	Predecessor to modern processors and OSes; popularized C	Most systems we use today		Trend toward built-in defenses		Fundamental redesign for security



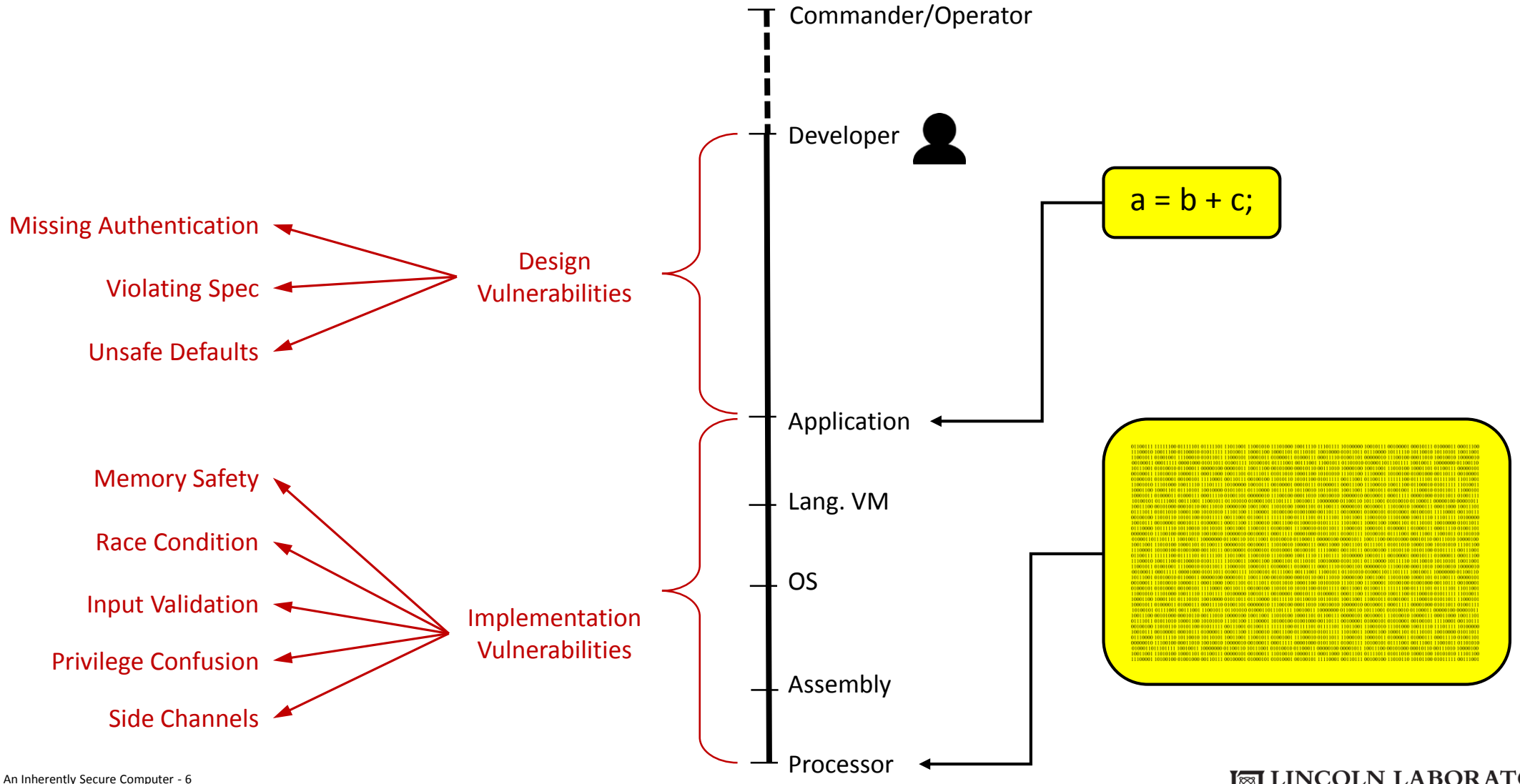


Resilient mission computer seeks to redesign hardware and software for security, fundamentally changing the paradigm of adding defenses to legacy systems

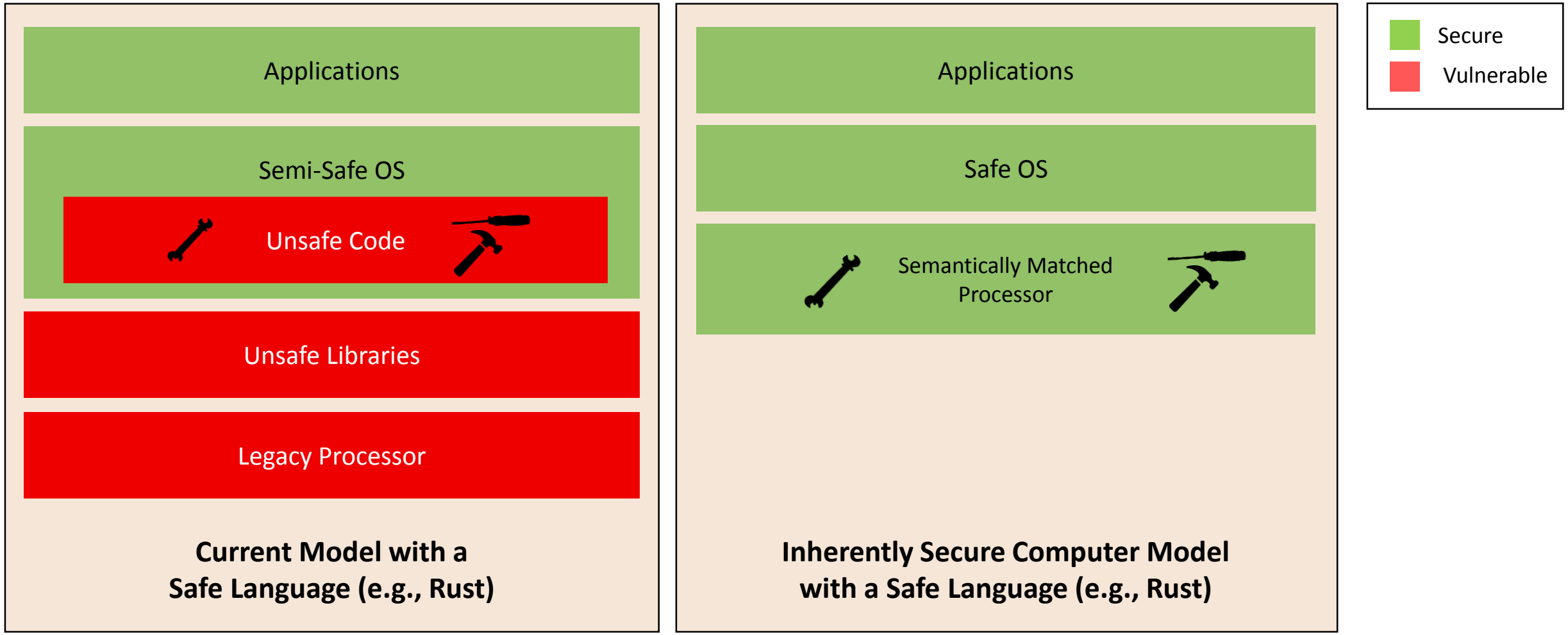
Vision

- Vision
  Create the inherently secure computer system of the future
- Goals
  - Create an inherently secure processor design
  - Implement the processor and integrate with proper software stack
  - Demonstrate it for a use case
  - Expand to other use cases
- Impact
  - Novel rethinking of computer architecture with security as a central goal
  - Demonstration on a mission use case
  - Reshape the cybersecurity landscape

Problem Statement

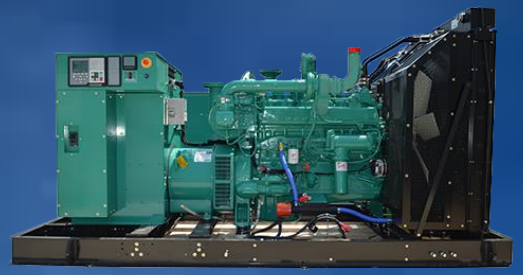


Vision for Processor and OS Components



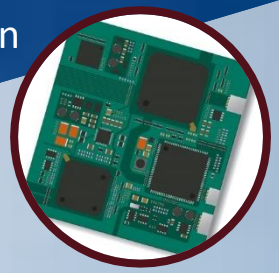
Broader Vision

Vision: Create a secure-by-design system in which the mission can succeed regardless of attempted attacks



Commercial legacy compute stacks and trends drive increased cyber vulnerability

Resilience for duration of the mission



Resilient Mission Computer

Clean-slate "minimalist" stack built to guarantee resilience

1960s	1970s	1980s	1990s	2000s	2010s	2020s & beyond
USER ASSEMBLY CODE (ASM) PROCESSOR	USER LOW-LEVEL LANGUAGES (LLL) LIBRARIES OPERATING SYSTEM (OS) PROCESSOR	USER HIGH-LEVEL LANGUAGES (HLL) LOW-LEVEL LANGUAGES (LLL) LIBRARIES OPERATING SYSTEM (OS) VIRTUALIZATION PROCESSOR		USER HIGH-LEVEL LANGUAGES (HLL) LANGUAGE VIRTUAL MACHINES LOW-LEVEL LANGUAGES (LLL) CONTAINERS LIBRARIES OPERATING SYSTEM (OS) VIRTUALIZATION PROCESSOR		
• Inherently Insecure • Add-on defenses				• Inherently Insecure • Built-in defenses		• Inherently Secure
MULTICS, Project MAC	PDP-11	x86, Windows, Linux		Intel Skylake, DARPA CRASH, and MRC		Inherently Secure Computer

- Authenticated User and Monitored Program
- Inherently Safe HLL to Minimize Attack Surface
- Validated Secure OS Developed in HLL
- Semantically Matched Secure Processor
  - Hardware directly implements HLL semantics to remove vulnerabilities
  - Checking built into the hardware to guarantee secure execution
  - Anti-tamper for CPI protection

New computer architecture provides inherent security and guarantees mission success

Technology Enablers for Inherently Secure Computer

Validated, Secure OS

- Preventing large classes of attacks using safe languages
- Security enforced among all software layers

Semantically Matched Secure Processor

- Semantically rich processor aware of security requirements
- Assured enforcement of security checks in the processor

Monitored Program

- Monitoring mission requirements
- Detecting hard-to-prevent attacks

Authenticated User

- Strong continuous authentication
- Proper attribution of actions

Anti-Tamper

- Cyber seals to detect tampering
- Side channels to detect supply chain attacks

Time Horizons for Technology Enablers

Initial Proof of Concept

- Validated, Secure OS
- Semantically Matched Secure Processor

Extension and Expansion

- Monitored Program

Full Vision

- Authenticated User
- Anti-Tamper

2020 2023 2026 TIME

Current Campus Collaboration and Opportunities

- Current Collaboration:
  - Dr. Howie Shrobe (HW Security)

- Opportunities:
  - Operating Systems
  - Programming Languages
  - Compilers
  - Formal Methods
  - Architectures
  - Secure Enclaves
  - AI
  - Cryptography

Summary

- Large classes of attacks are possible because of legacy design choices
- Inherently Secure Computer envisions a rethinking of computer architecture with ‘security’ as its central goal
- A combination of hardware and software innovations aim to prevent attacks by design
- It aims to provide a more even playing field for defenders
- Focus is on ‘mission success’ rather than general notions of security
- We are looking for new collaboration opportunities