Dramatically Reducing Attack Surface
Using Integrity MAC Security Kernel

Dr. Roger R. Schell, PhD
President and founder of Aesec Corporation
roger.schell@aesec.com
(831) 657-0899

CERIAS 2020 Seminar
Purdue University
West Lafayette, IN
Streamed live on the Web
September 2, 2020
4:30pm EDT
Presentation Outline

• Problem: national existential risk

• Towards a Reusable Trusted Device (RTD)

• Control Systems: PLC Commercialization
Presentation Outline

• **Problem: national existential risk**
  – Poor Cyber Physical Systems (CPS) resilience
  – Vulnerable critical cyber-physical components

• Towards a Reusable Trusted Device (RTD)

• Control Systems: PLC Commercialization
Leon Panetta, former SecDef & CIA Director
- “Biggest nightmare is of a computer virus that attacks and disables US infrastructure”
- “Could result in millions of lost lives” [Mar 2019]

EO 13920 – US Bulk Power: National Emergency

National Commission on Grid Resilience (NCGR)
- “OEMs are targets for malware that can lie in wait”
- Cyberthreat electric sector investment [Aug 2020]

Washington Post – “Power Grid Collapse”
- “Russia cause[d] physical damage from afar”
- “China has already implanted malware” [Aug 2020]
Computer systems all use operating system (OS)
- Programmable Logic Controllers (PLC) have an OS
Science: secure system requires trustworthy OS
- Must withstand witted adversary cyber attacks
Current commercial PLCs use untrustworthy OSs
- One of a few common OSs – none trustworthy
- Evident by stream of regular “security patches”
Cyberattacks inflict permanent physical damage
- STUXNET destroyed Iranian enrichment centrifuges
- Crash Override for physical Ukraine grid destruction
- Triton aimed for Saudi refinery destruction
• Problem: national existential risk

• **Towards a Reusable Trusted Device (RTD)**
  - Security kernel technology
  - Verifiable Integrity Mandatory Access Control (MAC)
  - OpenPLC on GEMSOS demonstration
  - Mature subversion mitigation

• Control Systems: PLC Commercialization
Security Kernel Technology
Solution Concept Introduction

• Seminal (1972) concept description
  “a compact security 'kernel' of the operating system and supporting hardware – such that an antagonist could provide the remainder of the system without compromising the protection provided.”

• Early (1983) IEEE article characterization
  “the security kernel approach provides controls that are effective against most internal attacks – including some that many designers never consider.”

• Consistent history of mitigating attacks
  “half dozen security kernel-based operating systems ran for years (even decades) in the face of nation-state adversaries without a single reported security patch”
“The only way we know . . . to build highly secure software systems of any practical interest is the kernel approach.”

-- ARPA Review Group, 1970s (Butler Lampson, Draper Prize recipient)

Still true today. Codified in TCSEC Class A1

TCSEC Glossary: “Security Kernel - The hardware, firmware, and software elements of a Trusted Computing Base that implement the reference monitor concept.”
“The only way we know . . . to build highly secure software systems of any practical interest is the kernel approach.”
-- ARPA Review Group, 1970s (Butler Lampson, Draper Prize recipient)

Still true today. Codified in TCSEC Class A1

Reusable Trusted Device: “The hardware, firmware, and software elements implement the reference monitor concept.”
“The only way we know . . . to build highly secure software systems of any practical interest is the kernel approach.”

-- ARPA Review Group, 1970s (Butler Lampson, Draper Prize recipient)

Truly a paradigm shift: no Class A1 security patches for kernel in years of use
Security Kernel Technology
Strategic Approach to Protection

- Controlled sharing between integrity domains
  - Enforce Mandatory Access Controls (MAC) policies
- Verifiable Design required for MAC enforcement
  - **Add on** security by test and analysis has failed
    - Threat/vulnerability detection & response never finish
  - **Build in** security by Construction is successful
    - Reference Monitor basis of the TCSEC Class A1 approach
- Mitigate subversion, e.g., malware (STUXNET)
  - To protect distribution of software & commands
    - Protect installed code, configuration settings & data

All required for Secure Operating System
• MAC policies required
  – To secure information flows
• Reference Monitor
  – Only known verifiable protection technology
• Deal with Subversion
  – Tool of choice for witted adversaries
Verifiable Design for MAC
NIST: Reference Monitor Concept

• NIST highlights in flagship SP-800-160v1
  – “Trustworthy Secure System Development”

• Reference Monitor Concept
  “provides an abstract security model of the necessary and sufficient properties that must be achieved by any system mechanism claiming to securely enforce access controls.”

• Security Kernel defined as its implementation

• Integrity-MAC is access control policy
Verifiable Design for MAC
Secure by Construction

E.g., Hardware Segmentation

Mappings

Security Policy
- Philosophy and Design of Protection

Formal Security Policy Model [linchpin]
- (FSPM) For Reference Monitor.
  i.e., security kernel API

Formal Top Level
- Level Spec (FTLS)

Descriptive Top
- Level Spec

With hardware properties
- visible at interface

Implementation Design Documents
- Source Code
- Layering & info hiding
- Code correspondence

Product End Item
- Trusted Distribution

System Specifications

Covert channel analysis

Development Specifications

Security Features
- Users Guide

Product Specifications

Trusted Facility Manuals

Deliverable Product
Verifiable Design for MAC
Ineffective Shortcuts

• Reference Monitor & **FSPM** are long, hard work
  – Omitted by unwary/lazy for “plausible” shortcuts

• “Verified OS” – for functionality, not policy FSPM
  – Example: seL4 – need to verify info flow outside OS

• “Partition Kernel” lacks FSPM for kernel API
  – Example: MILS – explicitly excludes from kernel

• “Verified capability hardware” – missing a FSPM
  – Examples: DARPA-sponsored CRASH and CHERI

• Static code analysis – lacks FSPM for API of OS
  – Example: LDRA Testbed

• **Shortcuts cannot enforce** Integrity MAC for PLC
Verifiable Integrity MAC
MAC Reduces CPS Attack Surface

Physical Device
Verifiable Integrity MAC
MAC Reduces CPS Attack Surface

- **Public Networks**
  access of any kind gives adversaries a huge attack surface

Public Networks

Physical Device
Verifiable Integrity MAC
MAC Reduces CPS Attack Surface

- **Public Networks** access of any kind gives adversaries a huge attack surface
- **Distributed control** is vulnerable to insider attack
Verifiable Integrity MAC
MAC Reduces CPS Attack Surface

- **Public Networks**
  access of any kind
gives adversaries a huge attack surface

- **Distributed control**
is vulnerable to insider attack

- **SCADA** and other adaptable control systems can be sabotaged
Verifiable Integrity MAC
MAC Reduces CPS Attack Surface

- **Public Networks** access of any kind gives adversaries a huge attack surface
- **Distributed control** is vulnerable to insider attack
- **SCADA** and other adaptable control systems can be sabotaged
- **Cyber Physical Control** requires protection of Safe Regions (e.g., Power System Settings) only Mandatory Access Controls provide
NIST calls out “kernel” in flagship SP-800-160v1

- “Electric Grid – Industrial/process control systems”

- PLC typically controls critical physical component

  “Trustworthy components within ICS, including for example, highly assured, kernel-based operating systems in Programmable Logic Controllers” [PLC]

- Kernel MAC controls integrity security domains

  “can help achieve a high degree of system integrity and availability through domain separation with control over cross-domain flows and use of shared resources.”
OpenPLC on GEMSOS
Reproducible Research Setup

Intel x86 Architecture
Hardware Protection Levels (Protection Rings)

- Essential Hardware Properties
  - Hardware Rings – more than 2
  - Memory Segmentation vs Paging
  - Strong Process Model
- NSA TCSEC/TNI Class A1 – Verified Design

Four distinct **hierarchical** integrity domains

1. Cyber physical system (CPS) control
   - **Only** domain with I/O access to physical hardware
   - Enforces “Pierson Safe Region” for physical device
2. Supervisory Control and Data Acquisition
   - SCADA domain – main PLC “Logic Loop”
3. Distributed control
   - Integrity-protected network interfaces
4. Untrusted public networks (e.g., Internet)
OpenPLC on GEMSOS
Open Source Research PLC

- Originated with Thiago Alves, Brazil
  - Developed at University of Alabama in Huntsville
    - Prof. Tommy Morris-led team
    - https://www.openplcproject.com

- Highly functional PLC for Windows, Linux, etc.

- Some commercial PLC vendors using

- Installed in Matt Bishop’s UCDavis Security Lab
  - On GEMSOS Developer’s Kit
OpenPLC on GEMSOS
4-Domains and CDS Transfers

Cyber Physical Control Domain

- Buffer Management
- Pierson Safe Region
- Physical Device

SCADA Domain

- Logic Loop
- Update Buffers In
- Program Logic
- Update Buffers Out
- mb_db Modbus Database

Distributed Control Domain

- Cross Domain Service #n
- Cross Domain Service #n
- Cross Domain Service #n
- tcp_buf #n
tcp_request
tcp_response

Key:
Information Flows ↔ Processes Databases
## OpenPLC on GEMSOS
Reduced Attack Surface

<table>
<thead>
<tr>
<th>Domain</th>
<th>Processes</th>
<th>Stripped Size (bytes)</th>
<th>Percentage of Whole</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cyber Physical Control</td>
<td>Phys Device Ctrl</td>
<td>14,556</td>
<td>0.7%</td>
</tr>
<tr>
<td></td>
<td>CDS #1</td>
<td>31,100</td>
<td></td>
</tr>
<tr>
<td>SCADA</td>
<td>Logic Loop</td>
<td>1,022,900</td>
<td>20.2%</td>
</tr>
<tr>
<td></td>
<td>Modbus RPC</td>
<td>275,732</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CDS #2</td>
<td>31,396</td>
<td></td>
</tr>
<tr>
<td>Distributed Ctrl</td>
<td>TCP/IP Stack</td>
<td>144,960</td>
<td>2.7%</td>
</tr>
<tr>
<td></td>
<td>CDS #3</td>
<td>31,100</td>
<td></td>
</tr>
<tr>
<td>Public Networks</td>
<td>RTU Handler</td>
<td>25,032</td>
<td>76.4%</td>
</tr>
<tr>
<td></td>
<td>External Network</td>
<td>&gt; 5,000,000</td>
<td></td>
</tr>
</tbody>
</table>
Mature Subversion Mitigation
NIST: Class A1 for Subversion

- NIST cites “Class A1” in flagship SP-800-160v1
  - “Application . . . to Commercial Products”

- Products are worked examples and use cases
  “highly trustworthy components and systems that have been verified to be highly resistant to penetration from determined adversaries”

- TCSEC **Class A1** distinguished
  “by substantially dealing with the problem of subversion of security mechanisms.”
Mature Subversion Mitigation
Trusted Device Protects Itself

- Trusted Boot for software/configuration settings
- Vet Trusted Devices for unauthorized behavior
- Code Correspondence stop “dead code” malware
- Trusted Distribution avoids supply chain attacks
- Media integrity mitigates “parking lot” attacks
Mature Subversion Mitigation
Illustrative STUXNET Mitigation

Class A1 Trusted Distribution rejects installation of untrusted software (malware) by checking for authorized cryptographic seal.

Parking Lot USB, phishing payload, etc.

Trusted Device Supplier

Trusted Device

Trusted Device Software

Crypto-Sealed

Lacks Seal

Crypto-Sealed

Trusted Device

Trusted Supplier Software

Physical Device (e.g., Generator)

= Trusted Hardware Root of Trust (e.g., Intel TPM or custom FPGA)
= Trusted Device Distribution Keys
= OEM Distribution Keys

Class A1 Evaluation included use of a suitable Trusted Distribution Cryptographic Algorithm (e.g., Simmons)

Suppliers AND Customers BOTH get Trusted Device updates via Trusted Distribution;

OEM Supplier’s Keys not exposed
Presentation Outline

• Problem: national existential risk

• Towards a Reusable Trusted Device (RTD)

• **Control Systems: PLC Commercialization**
  – Original Equipment Manufacturer (OEM) model
PLC Technology Transfer
Traditional OEM Model

• Security kernel vendor offers Trusted Device
  – Hardware & software domain-specific platform, e.g., motherboard, SOC
  – Trusted distribution, system security certification

• OEMs & manufacturers build PLC platforms
  – Trusted Device is part of any hardware product configuration

• VARs, ISVs, appliance vendors
  – Add applications and system services software, use OpenPLC source

• Solution providers and system integrators
  – Customization and integration for customers
  – Deliver complete solutions

10-15 yrs

<table>
<thead>
<tr>
<th>Trusted Device From Vendor</th>
<th>OEMS &amp; Manufacturers</th>
<th>VARs, ISVs, Appliance Vendors</th>
<th>Systems Integrator Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-3 yrs</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Previous Evaluations Accelerate

- Former DIRNSA LtGen Linc Faurer note [2007]
- “very high priority problem area”
  - “vulnerability of our network components and
t害怕电子凭证到软件subversion”
  - “convinced that an IC disaster looms”
- “demands that the first set of solutions”
  - “directly leverage the designs, architectures and
    - rating maintenance plans [RAMP] which NSA has
    - previously evaluated at the Class A1 level of assurance”
  - “this is the only practical way to be confident the
    - needed solutions can be operationally deployed in the
    - next couple of years.”
• **Problem: national existential risk**
  – Poor Cyber Physical Systems (CPS) resilience
  – Vulnerable critical cyber-physical components

• **Towards a Reusable Trusted Device (RTD)**
  – Security kernel technology
  – Verifiable Integrity Mandatory Access Control (MAC)
  – OpenPLC on GEMSOS demonstration
  – Mature subversion mitigation

• **Control Systems: PLC Commercialization**
  – Original Equipment Manufacturer (OEM) model
Verifiable CPS Bottom Line

• Critical *physical* components need verifiable PLC
  – Limited system risk from remaining components
• Kernel makes CPS attack surface much smaller
  – Each *integrity MAC* domain protected from lower
  – Security kernel *verified design* for unknown attacks
  – Deals with *subversion* of security mechanisms
• PLC performance & functionality retained
  – OEM host PLC on *trusted device* with secure OS
  – PLC manufacturers can use OpenPLC prototype
• Mature OEM business model & support approach
  – Successful security kernel OEM delivery history
CPS Cybersecurity Conclusion

Clear **NEED** for resilient CPS

Commercial **TECHNOLGY** available

**Need** PLC manufacturer **ADOPTION**
Dramatically Reducing Attack Surface Using Integrity MAC Security Kernel

Dr. Roger R. Schell, PhD
President and founder of Aesec Corporation
roger.schell@aesec.com
(831) 657-0899