Cyber Hardening Weapons Systems

Bill Beckwith, Objective Interface Systems
Dr. Ben Calloni, P.E., Lockheed Martin
Jess Irwin, Raytheon
Ellen Story, EMariaEnterprises
Gordon Uchenick, Coverity
W. Mark Vanfleet, NSA

System and Software Technology Conference (SSTC) 2012

Abstract

- Modern weapons systems
  - Contain hundreds, if not thousands, of processing points
    - Interacting in diverse and complex manners
  - Hardening these interactions is today's topic
  - There are two predominate approaches
    - Increasing the resilience of each processing point and the communications paths that connect them
    - Bounding the ways that attackers can access interfaces
  - The need to contain cost and meet schedule frequently constrain hardening efforts
  - Today we will discuss how trustworthy separation can bound the attackable interfaces while simultaneously reducing cost and schedule risk

Background

- Modern weapons systems
  - Complex multiprocessing systems which are
    - Highly interdependent, inherently brittle
    - Employ heterogeneous technologies
      - COTS, GOTS, and custom-built
      - Multiple tech refreshes driven by obsolescence events
        - Multiple generations of technology (10-20-40 years old)
        - Different levels of readiness (TRL5 – TRL9)
        - Developed, integrated, and maintained by multiple sequential contractors
        - Uncertain quality, completeness, and relevance of documentation
    - Continually evolving, potentially conflicting technology standards
    - Certification and Accreditation Processes constantly changing and overlapping
      - Which have been waived in the past because of urgency
      - Which are being circumvented by clever specmanship

Weapons Systems: Mitigating Threats

- Decompose system into components
  - Standardized Interfaces
  - Respect Interfaces among components
- Separate application from infrastructure
  - Allows zero day vulnerabilities to be addressed affordably
  - Preserves massive investment in mission critical applications
- Benefits
  - Minimizes disruption from obsolescence events
  - Accomplishes gigabuck technology refresh with megabucks
    - Enables “Do more, do it faster, do it with less”

Cyber-Survivability

- Cyber-survivability is the capability of a system to provide essential services even when components are penetrated and/or compromised
- Building affordable survivability demands
  - Realistically understanding the threat profile of the system
  - Performing appropriate risk management upon those threats
  - Architecting so that no vulnerability is a single point of failure
- Vigilance: Don’t allow system operating experience during peacetime to give a false sense of survivability

Risk Management

- Eliminate complexity thereby eliminating fragility
- Metadata produced as close to sensor as possible and unforgeably bound to the data
- Who is communicating through me and why?
- How do I reject unintended communications?
- Who am I willing to collaborate with?
- How can I tell if collaborators have been compromised?
- Can attribution be traced to each invocation of each critical function?

Weapons System Hardening Objectives

1. Irrefutable Identity
2. Trusted Path
3. Avoiding False Confidence in Weapons Systems
4. Integrating Multiple Generations of Technology
5. Bounding Cyber Threats
6. Bounding Life Cycle Costs
7. Safe and Secure Distributed Object Management
8. Trustworthy Separation

1. Irrefutable Identity (Hardware Perspective)

- Trusted Identity is a weaker form of Irrefutable Identity
- Definition of Irrefutable Identity
  - Identity cannot be stolen or masqueraded
- Without Irrefutable Identity
  - A web of trust cannot be created
  - Bad actors cannot be identified nor their actions attributed to them
- With Irrefutable Identity
  - Can utilize a “Calculus of Trust”
    - Everyone is not completely trusted
    - We know how much to trust each via Byzantine Agreement Protocols
  - Can create trust mesh within untrusted networks and domains
  - Can determine paths of maximum trust through the “mine field”

2. Trusted Path

- **T-I-M-E**
  - **T**ype-safe
    - Must fit footprint, proper syntax, allowed values
  - **I**nfiltration
    - Source of message is never mis-attributed
  - **M**ediation
    - Message initiated only from trusted node at its request
    - Messages can't be delayed or accelerated without detection
  - **E**xfiltration
    - Messages can't be delivered to or used by unauthorized party

3. Avoiding False Confidence in Weapons Systems

- Misconceptions (not in any order)
  - Security through obscurity
  - Multiple layers of weak protection automatically provide strong protection
  - Confidence can be derived from steel hulls, air frames at high altitudes, fences, locks, and guys with guns
  - Defense systems are isolated
  - Virtualizing an obsolete system adequately protects it
  - No one will bother to write a virus against an AN/UYK or a Mil-STD1750 Processor
  - As trust in individuals and systems increases, auditing and monitoring can decrease
  - As perceived risk lowers, barriers to attack can be reduced

4. Integrating Multiple Generations of Technology

- Inherently brittle
  - Legacy software will not necessarily run on current generation motherboards
  - Legacy software and protocols will not necessarily interoperate with newer versions
- Untrustworthy separation, or none at all
  - Everything is effectively visible to everything else
  - Everything is effectively connected to everything else
- Diminished Engineering and Manufacturing Sources
  - Original engineering work force reassigned, promoted, or retired
  - AN/UYK, Mil-STD 1750, TTL Logic virtualized to run on microprocessors or FPGA Cores

5. Bounding Cyber Threats

- System uses only data from appropriate sources
  - Data only accepted only from known identifiable sources
  - Trustworthy metadata incorruptibly bound with data at the source
  - Data earns increased level of trust based on metadata and trust level of source
- Mission classification takes into account mission phase, location, sensor resolution, and deployed sensor algorithms
- Releasability policy developed at mission planning

6A. Bounding Life Cycle Costs: The Problem

- Technology refreshes are never completely fresh
  - Minimum Weapon System Technology Refresh is 48 months
  - Maximum COTS Technology Refresh is 18 months
  - Sausage making between solicitation and implementation can consume most of that 18 months!
  - Typical commercial obsolescence is 36 months
    - H/W parts no longer available for purchase at any price
- How does life cycle cost become unbounded?
- Requirements creep!
  - Systems need to do new things in new environments
  - Outpaces budget

6B. Bounding Life Cycle Costs: The Solution

- Affordably upgradeable hardware and software architecture
  - Updating a mother board often requires moving to the next generation of the operating system, device drivers, etc.
    - New hardware often has new functionality that old software does not use
    - Can those unused functions become a penetration point?
  - Updating the operating system should not invalidate application level functions
- Protected design data
  - Must start over from scratch when design data is compromised or stolen
  - Security by Obscurity does not work

7. Safe and Secure Distributed Object Management

- Security with Object Request Brokers (CORBA, RMI)
- Leverages irrefutable identity
  - Requires trustworthy network infrastructure
  - Requires trustworthy name server
- Elements of a trustworthy infrastructure
  - Ensures that endpoints of communication{MAC Address, IP Address, TCP Port} are authorized
  - Attestation
  - Data marshalling/de-marshalling does not put system at risk of exploitation
- Distributed object risk management
  - Holistic look at architecture and infrastructure
  - Access Controls in policy must correspond to architecture

8. Trustworthy Separation

- Protection by Separation
  - Data isolation and control of information flow
- Separation in Transmission
  - IPSEC, TLC/SSL, Partitioning Communications System (PCS)
- Separation in Processing
  - RTOS, Application Whitelisting, SE LINUX, MILS, Storage
- Separation in Data at Rest
  - TPM, Data at Rest Protection

Risk Management

- Vulnerability Classification
  - Administration / Configuration Vulnerabilities
  - Sandboxing Vulnerabilities
  - Network looking Vulnerabilities
  - Infrastructure Vulnerabilities
- Application Vulnerability
  - Down-grading of data and/or metadata injects vulnerabilities
  - Complexity begets vulnerability
- Attack Vulnerability
  - Partner Vulnerabilities
  - Site Vulnerabilities
  - Data Preparation Vulnerabilities
  - Connectivity Vulnerabilities
  - Identity Vulnerabilities

Conclusions

- Architect systems such that, over the entire system life
  - Appropriate risk management can be performed and affordably reassesed
  - Systems do not unknowingly accept and process untrustworthy data
  - Zero-day vulnerabilities can be reasonably countered
  - Technology refreshes are affordable and do not impact mission or security of the mission
  - System capabilities, scope, and interconnections are scalable
- Manage systems such that:
  - Policy and planning are performed before there is a crisis
  - False confidence does not diminish security over time

Web Resources

- Coding Standards and Practices
  - http://www.cert.org/secure-coding/scstandards.html
  - http://cwe.mitre.org/
- National Vulnerability Databases
  - http://web.nvd.nist.gov
  - http://cve.mitre.org/
- DHS Pocket Guides for Security
  - https://buildsecurityin.us-cert.gov/swa/pocket_guide_series.html
- SEI Software Assurance Curriculum
  - http://www.sei.cmu.edu/library/abstracts/reports/10tr005.cfm
- Risk Management Framework
  - http://csrc.nist.gov/publications/nistpubs/800-37-rev1/sp800-37-rev1-final.pdf