Automated Assurance of Security Policy Enforcement (AASPE)

Principal Investigator: Peter Feiler, SEI Fellow
Presenter: Sam Procter, Architecture Researcher
Executive Summary

1. **Relevance for the DoD warfighter**
   Safety-critical systems are no longer closed but connected, thus exposed to security threats

2. **Relevance to state-of-the-art in software engineering or cybersecurity**
   Avionics industry embraces model-based architecture-centric virtual integration for safety

3. **Expected DoD practice improvements**
   Continuous assurance through automated architecture security analysis to complement code level security analysis
   - Analysis of security policy specifications for vulnerabilities
   - Analysis of security enforcement runtime architecture for vulnerabilities
   - Generation of runtime system configurations from verified security models
Security Challenges as Safety-Critical Systems Become Connected

Good Example: Aviation

Bad Example: Automotive

Lack of physical and logical isolation within system leads to costly rework and recalls

Safety-critical avionics systems use partitioning to achieve fault isolation

More than secure code and external firewalls

Security policy in form of acceptable command and information flows and isolation requirements
A Model-Based Analysis and Generation Approach

Modeling Tool: AADL
Architecture Analysis & Design Language Widely used standard (SAE International)
- Designed for embedded software systems
- Includes support for timing, performance, safety analysis

Extensions:
Model security policy and enforcement
- Security levels/domains
- Trust/verification
- Encryption
- Authentication
- Physical and logical exposure
- Concurrency
Vulnerabilities in Security Policies and Their Enforcement

Security policy vulnerabilities: Analyze Information Flows
Examples: Verify secrets stay secret, and Sensors can't send commands

Security enforcement vulnerabilities: Analyze Deployment Mechanisms
Example: Hi and low-security channels shouldn't coexist on unpartitioned hardware

Research Connection:
Apply Multiple Independent Levels of Security (MILS) framework (confidentiality) to system security (integrity)
Security Analysis Techniques and Tools

0. Consistency in security policy specification and enforcement
1. Model-Based Attack Impact Analysis (AIA) tool
2. Model-Based Attack Tree Analysis (ATA) tool
3. Generation of security configuration files
   - Model-based auto-configuration of certified kernel (seL4/CAmkES) security policy
Using Security Assurance Techniques and Tools

1. Specify security policy as verifiable requirements
2. Formalize verification activities
3. Automate execution of verification plans

MILS-R0: Components sharing a bus should have the same security level.
MILS-R1: Inter-communicating components should have the same security level.
MILS-R2: Processes with different security levels use isolated memory regions.
MILS-R3: Components associated with identical processing resources share the same security level.
MILS-R4: Threads inside the same process share the same security levels.
CWE-131: Incorrect calculation of buffer size.
CWE-311: Missing encryption of sensitive data.
CWE-805: Buffer Access with Incorrect Length Value.
AASPE Results

Code and Examples on GitHub (https://github.com/cmu-sei/AASPE)

- Tools
  - Security policy and enforcement verification on AADL models
  - Graphical attack impact and attack tree analysis tools
  - Generation of attack impact graphs and attack trees from AADL models with security annotations
  - Generation of seL4 configuration files from AADL based specifications
- Example models
  - Automotive: Jeep, Prius
  - Aircraft model
  - Drone case study

Proposal for AADL Security Annex standard

Papers/report on security analysis, security assurance workflow

Proposal for an integrated safety and security engineering approach
Summary and Future Work

Where We Started: DARPA High-Assurance Cyber Military Systems (HACMS) program successfully demonstrated AADL-based verification and generation for reducing vulnerabilities in unmanned drones.

What We Did: We demonstrated the feasibility of improving security assurance through architecture modeling and analysis of vulnerabilities in security policy specification and enforcement.

What’s Next for the Community: Use of SAE International AADL standard offers transition path through the Open Source AADL Tool Environment (OSATE). We will advance the proposed Security Annex to AADL towards approval.

What’s Next for the SEI: Develop an integrated approach to safety and security engineering approach to mission critical systems funded as new three-year SEI line project.
Contact Information

Principal Investigator
Peter Feiler, SEI Fellow
phf@sei.cmu.edu
412.268.7790

Presenter
Sam Procter
sprocter@sei.cmu.edu