Use of Formal Methods in Assessment of IA Properties

44^{th} Meeting of IFIP Working Group 10.4

George W. Dinolt
gwdinolt@nps.navy.mil

Computer Science Department
Naval Postgraduate School
833 Dyer Road
Monterey, CA 93943
USA

Outline

- What is the Question
- Some Previous Work
- A Proposal on Measuring Assessment
- CISR Exemplar Project

The Question

Who is the enemy?

Can one use Formal Methods to measure how well IA properties have been implemented in a system to defend against the enemy?

If so how?

What Properties?

- Describe the Desired Functionaliy

- Provide Assurance that the Functionality Makes Sense

- Provide Assurance that the Functionality is “correctly implemented”

“Formal Methods” in IA

By Formal Methods we mean application of mathematics and mathematical models to:

- Describe the (Security) Properties of the System
- Describe the (Security) Functionality of the System
- Prove that the Functionality is Consistent with the Policy
- Prove that the Implementation is an Instance of the Functionality

Previous Work

Previous IA Assessment Schemes Include:

- **Trusted Computer System Evaluation Criteria** (TCSEC)
- **Common Criteria**
- Other Risk Management/Mitigation Schemes

TCSEC Criteria

C1 - Simple Testing and Audit (Unix/Linux)

C2 - ACLS, Testing, Protected Audit, No Object Reuse,

B1 - MLS Security Policy, Documentation, Testing

B2 - Stronger MAC Policy, Trusted Path, Audit

B3 - Reference Monitor, Highly Structured, More . . .

A1 - Mathematical Model of Security Policy,

Assurance Measures in the TCSEC

- Documentation
- Test Plan Structures
- Certain Functionality
- System Structure
- Documentation of Design, Implementation, Use
- System Security Policy
- Security Model
- Formal Top Level Specifications
- Verified Implementation

Formalisms for Security Policy

- Security Policy (textual description)
- Mathematical Model of Security Policy
- Informal mapping between Security Policy and Mathematical Model
- Proof that Mathematical Model is “Consistent”

Measures of Effectiveness

- Is there a Security Policy?

- Is there a Mathematical Model of the Policy?

- How Transparent is the Mapping between the Textual Policy and the Mathematical Model

- How was the "Consistency" of the Mathematical Model Shown

System Architecture Goals

- Partition System into “Trusted” and “Untrusted” Parts
- Trusted Part Required to Enforce the Security Policy, the Trusted Computing base
- No Behavior of the Untrusted Part can Affect the Security of the System

Formal Top Level Specification

- Choose Specification Language

- Translate Mathematical Model to Specification Language and redo proofs in the terms of the Language

- Describes the Security Properties of the Trusted Portion of the System in the Terms of the Specification Language (FTLS)

- Verify that the FTLS is Consistent with the Security Model

Measures of Effectiveness

- Does the FTLS exist?

- Does it Adequately Describe the Security Model - Is the mapping between the Security Model and the FTLS complete in some sense

- Is the FTLS small enough to be analyzable

Measuring Assurance Using Formal Methods

The measure is “how deep” the process has been carried.

- Security Policy Articulated
- Model Constructed of Security Policy (including mapping)
- FTLS Constructed and Mapped
- Detailed Specification Constructed and Mapped
- Implementation Constructed and Verified
- Hardware Specified and Verified

CISR MicroKernel Project

Goal is to Provide an “Open” Demonstration of how to build a very high assurance component. We are:

- Developing an CC EAL 7 evaluatable Micro Kernel
- All the Documentation, Processes, etc. will be available over the Web
- We expect to provide examples of appropriate Life Cycle Management, Configuration Management, Application of Formal Methods, Implementation Strategies, Documentation, Testing, etc.