Expressing Common Criteria Security Requirements in Domain Models in Model-based Architecture

Ringo Ling, Hugo Latapie and Vu Tran
Focus of the Presentation

- Expressing security requirements as constraints in a declarative domain model.

- In particular, Common Criteria security requirements are expressed as constraints using Object Constraint Language in a UML domain model.
Motivation

Problems:

- Embedded Software often implemented in low level languages.
- Embedded Software often lack formal requirements and models.
- Security requirements impose additional constraints on embedded software.

Motivation: To improve our embedded software development.
Approach for Improvement

- Linking requirements to implementation
  - Adopt the OMG’s Model-based Architecture approach, i.e. express software and its models at multiple levels of abstraction.

- Improving requirement specification, especially security requirements.
  - Use Common Criteria Standard
OMG’s Model-based Architecture

- Computation Independent Domain Model
- Platform Independent Computational Model
- Platform Specific Implementation Model
A declarative model expresses domain knowledge and implements business requirements of a domain.
But a key is more than just a string, it needs to meet some security requirements.
We adopt the security requirements of Common Criteria Standard to support our domain modeling.

- Comprehensive requirements,
- A methodology to support the development of requirements.
- Supported by organizations in many countries including NIST and NSA in US.
Common Criteria Derivation of Requirements and Specifications

Establish Security Environment

Assumptions
Threats
Security Policies

Security Objectives

Functional Requirements
Assurance Requirements
Requirements for Environment
Smart Card Protection Profile

- Smart Card Security User Group (SCSUG), version 3.0.
- It has 42 security requirements in following categories:
  - Audit
  - Cryptographic key
  - Data Access
  - File Structure Control
  - Identification
  - Recovery
Issue:

- We would like to express the Common Criteria security requirements for some domain attributes and operations,
- On the other hand, we do not want to introduce many security attributes and operations in the domain model. We want to avoid obscuring the declarative nature of the model.
The Solution

- Use Object Constraint Language to express security requirements as constraints for attributes and operations in a domain model.
Approach

Model-based Architecture

Common Criteria Security Requirements expressed as constraints in OCL

Declarative Domain Models
Examples

- FCS_CKM.1: Cryptographic Key depends on algorithm and key size

```
Smart card

Key: String
```

--Key Constraints
def self.key = KeyGeneration(a: algorithm, s: keySize)
Examples

- FAU_LST.1.1: Shall generate auditable events for level of audit,

| Smart card |
| HistoryList: List |

--History List to produce an audit list of certain level
HistoryList->Collect(LevelOfAudit)
Choice of security requirements

- Only the security requirements that are directly related to attributes and operations of a domain are expressed as constraints at the domain level.

- Other security requirements, primarily dealing with management and storage, are addressed at the computational level.
Examples of Choice

- Domain level:
  - FAU_LST.1 (Audit List generation)

- Computational level:
  - FAU_SAA.1 (Potential violation analysis)
  - FAU_STG.1 (Protected Audit trail storage)
  - FAU_STG.3 (Audit Data loss)
Examples of Choice

- Domain level:
  - FCS_CKM.1.1 (Cryptographic key generation)

- Computational level:
  - FCS_CKM.3.1 (Cryptographic key access)
  - FCS_COP.1.1 (Cryptographic Operation)
Contribution

- Use the existing framework of UML to express the Common Criteria security requirements as constraints for attributes and operations in declarative domain models.
Conclusion:

Model-based Architecture

Common Criteria Security Requirements expressed as constraints in OCL

Declarative Domain Models