FORMALIZATION OF SECURITY REQUIREMENTS

Arnaud Lefray, Eddy Caron, Christian Toinard
Journée Langages - 17 Mars 2016

Équipe Avalon - LIP - ENS Lyon / Qirinus Startup Project - Inria
Growing security breaches

2015 Average cost per breach: $3.79 million
2015 Average cost per stolen record: $154
Growing security breaches

2015 Average cost per breach: **$3.79 million**
2015 Average cost per stolen record: **$154**
Security Issues - Why do we fail?

Current practices: Manual security

An IT managing security “by hand” (configuration, etc.)

Cumbersome Security

- Lack of usability: Complex security mechanisms
- Lack of expertise: Misconfiguration
- Lack of formalism: Requirements in natural language

Security Composability

The sum of locally secure systems does not imply a globally secure system
Virtualized Distributed Systems

- Data
- Processes/Services
- VM
- Network

Trends

- Complexification of distributed systems
- Outsourcing and Cloud adoption
- Mutualization of resources: Multitenancy
Global Problem Statement

Problem

How to provide global security of large scale virtualized distributed systems?

- Usability: ability for a user to specify security requirements.
- Scalability: no centralized system or single point of failure.
- Provability: Guarantee the global enforcement.
- Applicability: security enforcement doable for heterogeneous environments.
Model-based Automatic Security Enforcement

- Usability: high-level security language to specify requirements.
- Scalability: Distribute global properties locally on each node.
- Provability: Proof-based global to local transformations.
- Applicability: local enforcement based on existing security mechanisms.
What is Security?

**Security Policy**

What it means to be secure. Defined by

**Security Properties**

- **Confidentiality**: Absence of unauthorized disclosure
- **Integrity**: Absence of unauthorized alteration
- **Isolation**: Confidentiality + Integrity
- **Availability**: Absence of denial of use

**Identification**

I'm Bob

**Authentication**

ID please

**Authorization**
What is Security?

Security Policy

What it means to be secure. Defined by security properties

Security Properties

- Confidentiality: Absence of unauthorized disclosure
- Integrity: Absence of unauthorized alteration
- Isolation: Confidentiality + Integrity
- Availability: Absence of denial of use
Q. Can I read document File?
Q. Can I read document File?

Access Control
Explicit perms., implicit flows
A. Yes, access is granted.
Q. Can I read document File?

Access Control
Explicit perms., implicit flows
A. Yes, access is granted.

Information Flow Control
Implicit perms., explicit flows
A. Depends on previous flows.

Access Control checks place restrictions on the release of information, but not its propagation.
Existing models – Nguyen et al. [APSEC2013]

▶ Specific isolated security concerns (Not all security properties)
▶ Lack of formality
▶ Incomplete integrated approach (automation, process-integration, etc.)

Problem

No Models for Information Flow Properties on Virtualized Distributed Systems
Solution

A unified security-aware metamodel: Sam4C
Metamodel (Model of models)

Reducing complex programming tasks by:

- abstracting system-specific constraints
- providing automatic transformation
USECASE: AIRPORT MANAGEMENT

- Industrial UseCase (Ikusi Company)
- $n$-tier application (Standard for building enterprise software)
**Integrity Property**

Musik MAD application logs can only be modified by the Musik MAD service.

**Isolation Property**

The whole AirportContentManager framework is isolated from any other tenant in the hosting virtualized infrastructure.
**Security Model – Properties**

**Context**

Identifier referring to a (single or) group of entities represented by a list of attributes

**Integrity Property**
Secured [, Authorized]

**Isolation Property**
Secured [, Authorized]

\[
\text{Integrity} (\text{ctxLogMusikMad}, \text{ctxServiceMusikMad}) \\
\text{Isolation} (\{\text{AirportContentManager}\})
\]
Security Properties Interpretation

Problem

- What is the definition/interpretation of security properties? (e.g., Integrity)
- How to guarantee that $\sum localConfigs \equiv globalProperty$? (e.g., Isolation)
Problem

- What is the definition/interpretation of security properties? (e.g., Integrity)
- How to guarantee that $\sum_{local\text{Configs}} \equiv global\text{Property}$? (e.g., Isolation)

Solution

Use a formal language: Logic

- Unique interpretation
- Proof of equivalences (global/local)
**Information Flows**

**Flows**
- $B$ read $A = (A > B)$
- $B$ write $C = (B > C)$

**Types of Flows**
- Direct (e.g., $A > B$)
- Indirect (e.g., $A \gg C$)
Existing Logics not suitable

- Focus on Access Control (and not Information Flow Control)
- Do not tackle Indirect Flows (Transitive Closure)
Existing Logics not suitable

- Focus on Access Control (and not Information Flow Control)
- Do not tackle Indirect Flows (Transitive Closure)

IF-PLTL: a many-sorted first-order temporal logic

- **first-order**, Exists/Forall
- **temporal**, Condition(s) on past actions
- **many-sorted**, 2 sorts: **Context** *(e.g., Alice, File)* and **Domain** *(e.g., Admins, Madrid)*
Confidentiality (S,A)

\[(\forall_{ctx} x \in S)(\forall_{ctx} y)(x > y) \rightarrow (y \in S \cup A)\]

Integrity (S,A)

\[(\forall_{ctx} x \in S)(\forall_{ctx} y)(x < y) \rightarrow (y \in S \cup A)\]

Isolation (S,A)

\[confidentiality(S, A) \land integrity(S, A)\]
Global/Local Equivalences

Integrity(S, A)

\[(\forall_{ctx} x \in S)(\forall_{ctx} y)(x > y) \rightarrow (y \in S \cup A)\]

Given \( S = S_1 \cup S_2 \)

\[\text{Integrity}(S, A) \equiv \text{Integrity}(S_1, A \cup S_2) \land \text{Integrity}(S_2, A \cup S_1)\]
GLOBAL TO LOCAL TRANSFORMATION

Global Implicit Properties → Implicit To Explicit → Global Explicit to Singleton → Singleton to Typed Properties → Locally Enforceable Properties
Global to Local Transformation

Tenant Input
1 global implicit property: **Isolation**

**Problem**
Public should access Proxy
Solution
Use graph connectivity

Result
2 Explicit Isolation Properties
Input
1 global explicit property: **Isolation**

Problem
No solution for Musik_MAD, ..., Intranet.
Solution
Use global/local equivalences

Result
4 singleton explicit properties
GLOBAL TO LOCAL TRANSFORMATION

Input
1 singleton explicit properties

Problem
No isolation mechanism for mixed VM/VNet e.g., Mysql, Intranet
**Solution**

Use typed equivalences

**Result**

1 Explicit Property = 2 Typed Properties

IsolationVM and IsolationVNET
Conclusion

Problem summary

How to provide global security of large scale virtualized distributed systems?

- Usability
- Scalability
- Provability
- Applicability

Presented solution

Model-based Automatic Security Enforcement

- Focus: IFPLTL, a logic suitable for information flow properties
Open research problems

- Modelization: Elastic apps, reconfiguration at runtime
- Formalization: General equivalences, coherence verification
- Security: distributed reference monitor, autonomic security

Research Agenda at Qirinus

Toward a unified language/model to specify and automate all security aspects

- Network-based security
- Host-based security
- Application Hardening
- Fault-tolerance