



Protection  
of  
Complex Distributed Systems

**Rudolf Schreiner, Ulrich Lang**

[ras@objectsecurity.com](mailto:ras@objectsecurity.com)

[www.objectsecurity.com](http://www.objectsecurity.com)



21 Oct  
2008  
Open Group

[info@objectsecurity.com](mailto:info@objectsecurity.com)  
[www.objectsecurity.com](http://www.objectsecurity.com)

# Agenda

- ▶ Our task: Building secure distributed system
- ▶ Identifying the issue: Complexity
- ▶ SWIM as non trivial test system
- ▶ Reduction of complexity: Model driven policy generation

# Our Task

- ▶ Our task is to build secure complex systems
  - ▶ Air Traffic Management
  - ▶ Defense
  - ▶ Critical infrastructure
  - ▶ Financial
- ▶ We need practical solutions for real world systems: Protection of system as a whole

# Correct Enforcement

- ▶ In the past, a lot of attention was paid to the correct implementation of security functionality
  - ▶ Authentication (Encryption, PKI, Identity Management)
  - ▶ Message protection (Encryption, random number generation)
  - ▶ Domain protection (Firewalls)
  - ▶ Authorization and entitlement management (XACML, OpenPMF)
- ▶ But we really need:
  - ▶ *Correct and consistent enforcement of a correct and appropriate security policy*

# But what about Correct Policies?

- ▶ Security today:
  - ▶ A high level policy document
  - ▶ given to many developers and administrators
  - ▶ enforced on several security systems and mechanisms with little integration
- ▶ This was more or less considered sufficient for simple systems
  - ▶ Simple system architecture
  - ▶ Simple, technology driven security policies
- ▶ In reality, it was never a good approach...
  - ▶ Different interpretations
  - ▶ Policy consistency
- ▶ ...and will completely fail in the future

# System Complexity

- ▶ Service oriented
- ▶ Multiple paradigms
  - ▶ Request/Reply
  - ▶ Publish/Subscribe
- ▶ Heterogeneous
  - ▶ Multiple middleware platforms/operating systems
- ▶ Many stakeholders
- ▶ Size of systems (of systems)
- ▶ Permanent evolution

# Security Requirements

- ▶ Security must be able to support business driven objectives
  - ▶ Compliance
- ▶ We need a fine grained and flexible protection of business assets
  - ▶ Instead of huge trust domains (VPN)
- ▶ We need a high level of assurance:  
Security and safety

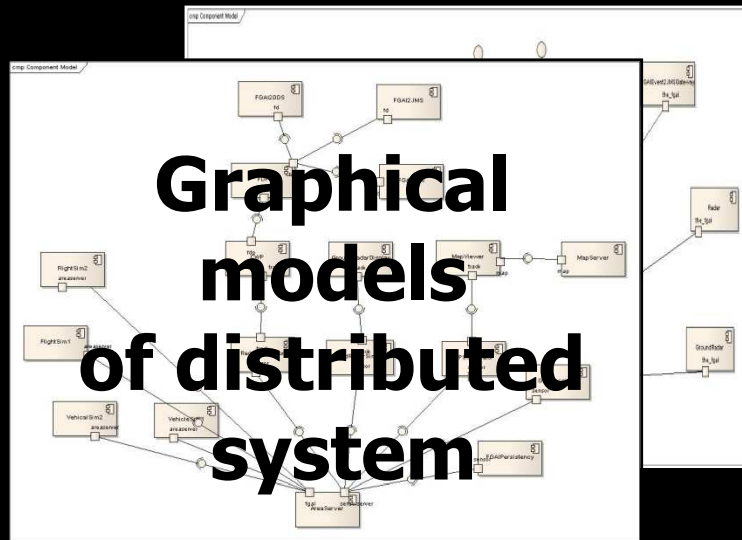
# Complexity of Protection

- ▶ It is still necessary to get all details right
  - ▶ Encryption, random number generators,...
  - ▶ Platform/protocol specific protection
- ▶ But, in general, the single mechanisms are not the main problem anymore
- ▶ The main issue is the complexity of protection and security management
  - ▶ Integrated protection required
  - ▶ Very complex configuration of security mechanisms
  - ▶ Very complex access control policies

# Example: SWIM

- ▶ Protecting HelloWorld is simple
  - ▶ We need a more realistic test
- ▶ We use a *System Wide Information Management* simulation as non-trivial test platform
- ▶ Based on Model Driven Development
  - ▶ Eclipse EMF, GMF, DSL
  - ▶ OpenArchitectureWare for transformations
  - ▶ Intalio BPMN
- ▶ Heterogeneous runtime  
Web Services (Java AppServer), CCM, JMS, DDS, XML

# SWIM



- ▶ Simulation of
  - ▶ Airports&Aircrafts
  - ▶ Air traffic
- ▶ Information exchange
- ▶ Various Displays
  - ▶ CWP/Radar
  - ▶ Flight Data
  - ▶ Time table

# Protecting the SWIM System

- ▶ Use an authorization management system
  - ▶ ObjectSecurity OpenPMF Policy Management Framework
  - ▶ XACML could be used as well
- ▶ Define authorization rules in a formal language: PDL
- ▶ Distribute rules to Policy Enforcement Points
- ▶ Enforce rules at runtime
- ▶ **We failed miserably**

# Complexity of Protection

- ▶ We were not able to define the access control rules
  - ▶ Had to use a trial and error method
  - ▶ Ca. 1000 rules in this quite simple system
  - ▶ Very low assurance of protection
  - ▶ Very low safety: Correct operation of the system?
  - ▶ Very brittle system, low agility
- ▶ We did not sufficiently understand the system to define the rules, esp. the interactions

# The Way to a Solution

- ▶ Humans are bad at dealing with many details: Fine grained rules
- ▶ But why do they have to deal with all these rules?
- ▶ Because security is separated from the rest of the system life cycle!
- ▶ Solution: Full integration of security into Model Driven Engineering over the full life cycle

# Model Driven Development

- ▶ For the development of our ATC system we had used Model Driven Engineering
- ▶ Modeling of the functional aspects of the system:
  - ▶ Service interface definitions
  - ▶ Service interaction definitions
  - ▶ System and deployment definitions
- ▶ We added a security model: Security Policy Language

# Model Driven Security

- ▶ We now have:
  - ▶ A model of the entities and their interactions
  - ▶ A high level security policy
- ▶ We generated:
  - ▶ Configuration of encryption (command line arguments)
  - ▶ Authorization rules (OpenPMF PDL, XACML) for all middleware platforms and also for business processes
  - ▶ A human readable evidence

# Evaluation

- ▶ Same SWIM test system
  - ▶ Security configurations and authorization now automatically generated from models
- ▶ The generated policy was correct from the beginning
  - ▶ No false denies anymore
  - ▶ Simulated violations were detected

# MDS Advantages 1

- ▶ High level security requirements comparatively easy to express
  - ▶ Small number of high level rules
- ▶ Traceability from compliance rules to low level enforcement
  - ▶ Security: Fine grained protection, high assurance
  - ▶ Safety: Rules always in line with functional behavior of the system

# MDS Advantages 2

## ▶ Agility

- ▶ In agile SOA systems the security configuration is always automatically adapted to functional modifications

## ▶ Reduced effort

- ▶ Policy is automatically generated with no human interaction

## ▶ Improved compliance monitoring

- ▶ Deviations from expected behavior can be detected

# Conclusion

- ▶ Security is driven by business and compliance requirements
- ▶ This leads to very complex security policies, which cannot be manually defined anymore
- ▶ Security has to be fully integrated into system life cycle
- ▶ Model Driven Security makes appropriate security manageable



▶ Resources:

- ▶ OpenPMF 2.0 website: [www.openpmf.com](http://www.openpmf.com)
- ▶ Model Driven Security blog: [www.modeldrivensecurity.org](http://www.modeldrivensecurity.org)
- ▶ SOA Security Concerns Analysis wiki: [www.secure-soa.info](http://www.secure-soa.info)
- ▶ ObjectSecurity website [www.objectsecurity.com](http://www.objectsecurity.com)



info@objectsecurity.com  
www.objectsecurity.com



[www.objectsecurity.com](http://www.objectsecurity.com)

[info@objectsecurity.com](mailto:info@objectsecurity.com)



ObjectSecurity Ltd.  
St John's Innovation Centre  
Cowley Road  
Cambridge CB4 0WS  
United Kingdom



ObjectSecurity LLC  
Plug&Play Tech Center  
530 University Ave  
Palo Alto, CA 94301  
USA



Tel: +44 (0) 1223 420252  
Fax: +44 (0) 1223 420844



Tel: 1-800-898-9148  
Fax: 1-360-933-9591



[www.objectsecurity.com](http://www.objectsecurity.com)  
[info@objectsecurity.com](mailto:info@objectsecurity.com)



[www.objectsecurity.com](http://www.objectsecurity.com)  
[info@objectsecurity.com](mailto:info@objectsecurity.com)



[info@objectsecurity.com](mailto:info@objectsecurity.com)  
[www.objectsecurity.com](http://www.objectsecurity.com)

# Terms & Conditions

- ▶ **© 2000-2008 ObjectSecurity Ltd. All rights reserved.**
- ▶ **This entire document is copyright protected and no changes to the document granted without permission.**
- ▶ **No re-selling permitted without prior explicit permission. No patenting of any of any of the described aspects permitted.**
- ▶ Intellectual property: This document (and the previously delivered outline) describes internals of OpenPMF, which are the intellectual property (background IPR) of ObjectSecurity Ltd., for which patents are pending. ObjectSecurity is the inventor of several of the described concepts, and any exploitation of these without permission will be considered as infringement of ObjectSecurity's legal rights
- ▶ Copyright, author rights, trademarks and other intellectual property rights: Some names are protected by trademarks which are the property of ObjectSecurity Ltd. or other third parties whether a specific mention in that respect is made or not. In particular (but not limited to): The ObjectSecurity logo, ObjectSecurity, the OpenPMF logo, OpenPMF, the ObjectWall logo, ObjectWall, SecureMDA, the SecureMDA logo, TrustedSOA, the TrustedSOA logo, SecureMiddleware, the SecureMiddleware logo, Security Management Ecosystem, SimulateWorld, and the SimulateWorld logo are trademarks or registered trademarks of ObjectSecurity.
- ▶ This document and its contents are protected by copyright, author rights and/or other intellectual property rights which are the property of OBJECTSECURITY or third parties. Reproduction and use of the materials (or any information incorporated thereto such as but not limited to articles, graphical images, pictures, diagrams, video materials...) published in this document are hereby authorized provided that :
  - ▶ (a) reproduction and use are solely for informational and non commercial use within your organisation in support of your better knowledge of Model Driven Security; and
  - ▶ (b) any reproduction retains all original notices including proprietary or copyright notices ; and
  - ▶ (c) materials are not modified, in whole or in part, in any way whatsoever.
- ▶ No other use of the materials and of any information incorporated thereto is hereby authorized.
- ▶ All concepts described may be protected by one or more patents or pending applications.
- ▶ No part of this document may be reproduced in any form by any means without prior written authorisation of ObjectSecurity Ltd.
- ▶ **Disclaimers**
- ▶ This document is provided for general information only and should not be relied upon or used as the basis for making any transactions of any kind whatsoever.
- ▶ All the information and any part thereof provided in this document are provided « AS IS » without warranty of any kind either expressed or implied including, without limitation, warranties of merchantability, fitness for a particular purpose or non infringement of intellectual property rights.
- ▶ OBJECTSECURITY makes no representations or warranties as to the accuracy or completeness of any materials and information incorporated thereto and contained in this document.
- ▶ OBJECTSECURITY makes no representations or warranties that this document will be free of harmful components.
- ▶ The use of the materials (or any information incorporated thereto), in whole or in part, contained in this document is your sole responsibility. OBJECTSECURITY disclaims any liability for any damages whatsoever including without limitation direct, indirect, incidental and/or consequential damages resulting from access to the document and use of the materials provided therein.
- ▶ This document may contain links to third party sites. The links are provided to you only as a convenience and the inclusion of any link does not imply neither an endorsement by OBJECTSECURITY of the linked sites nor any warranty from OBJECTSECURITY on said sites. Access to said linked sites is at your own risk.
- ▶ THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.
- ▶ THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS.