Model-based Security Analysis for Mobile Communications

Jan Jürjens¹, Jörg Schreck², Peter Bartmann³

¹Computing Department, The Open University, GB
²O2 (Germany)
³Univ. Augsburg, Germany

J.Jurjens@open.ac.uk
http://www.jurjens.de/jan
Mobile Communication Systems

Particularly complex and security-sensitive.
Various access media (WLAN, bluetooth, …), different standards, different security levels, with variations (e.g. “downwards compatible”)
Various enduser devices (laptop, mobile phone, …) with different OSs etc. Varying kinds of protection mechanisms.
Variations of security requirements (different architectural levels, strengths of protection, …).
Several thousand combinations of the above. Need automated tools to analyze for security.
Considered part of corporate security architecture and security policies for mobile communication systems at O₂ (Germany).

Modelled and analyzed security-critical parts using UMLsec and related tool-support.

Scientific goal: investigate use of UMLsec in industrial telecommunications context, show benefits and limitations.
Evaluation Criteria (from Partner)

Goal: Evaluate security analysis approach under following criteria:

• Reproducability
• Delegatability
• Efficiency
• Parallelization
• Traceability
• Expressiveness
Model-based Security Engineering

Requirements

Weave in

Analyze against

(UML) Models

Idea: Extract models from artefacts in development and use of software.

Verify.

Gener.

Reverse Engin.

Source Code

Configure

Tool-supported, theoretically sound, efficient automated security design & analysis.
Secure System Lifecycle

Model-based Security Engineering

Design: Encapsulate prudent security engineering rules.
Analysis: Formally based, automated, efficient tools.
Note: emphasis on high-level requirements.
Model-based Security with UMLsec

Extension of the Unified Modeling Language (UML) for secure systems development.

- evaluate UML models for security
- encapsulate established rules of prudent secure engineering
- make available to developers not specialized in secure systems
- consider security requirements from early design phases, in system context
- can use in certification
**UMLsec**

Insert recurring security requirements, adversary scenarios, security mechanisms as predefined markers. Use associated logical constraints to verify specifications using model checkers and ATPs based on formal semantics. Ensures that UML specification enforces the relevant security requirements wrt Dolev-Yao type adversaries.  

[FASE01,UML02,ICSE05]
Tool Support

For example:

- consistency checks
- mechanical analysis of complicated requirements on model level (bindings to model-checkers, constraint solvers, automated theorem provers, …)
- code generation
- test-sequence generation
- configuration data analysis against UML.
Jan Jürjens (OU): Model-based Security Analysis for Mobile Communications

Tool Support
Cf demo this afternoon

Java editor

UML editor

Java code

UMLsec model

Code with Assert's; Tests

Text Report

Attack Trace

Assertion/Test Generator

Analyzer

Local Code Checker

Automated Theorem Prover

Attack generator

Security Analyzer

FOL fmla

Prolog prog.

[UML04, FASE05, ICSE06]
Model Analysis Process

Security requirements

Abstraktion & Formalization

Security model

Analysis

architecture model
(projection of the technical specification on the security model)

Abstraktion & Formalization

technical specification
(operating system, software components, etc)

Design

usage model
(needed functionality)
Analysis

Extracted 62 security requirements from security policy documents.

- 21 process-related requ. captured in 8 activity diagrams using stereotypes <<fair exchange>> and <<provable>>
- 10 requ. regarding secrecy and integrity of data on physical layer, in 1 deployment diag.
- 3 requ. regarding RBAC
- 15 requ. regarding security of network services / dataflow wrt. use of firewalls / anti-virus software (extension to UMLsec)
- 13 requ.: no appropriate representation in UMLsec
Example: Network Architecture
input_formula(network_architecture_model, axiom, (is_component_of(internal_workstation, iw-application) & is_component_of(intranet, i-server-application) & type_of_component(iw-application, others) & type_of_component(i-server-application, critical) & access_media_availability(internal_workstation, lan) & access_media_availability(intranet, lan) & service_on_access_media(http, lan) & connection(service, iw-application, i-server-application, http, plaindata))).

input_formula(connection_of_service_to_connection_of_dataflow, axiom, (! [ComponentX, ComponentY, Service, DataEnc] : ( (connection(service, ComponentX, ComponentY, Service, DataEnc)) => (connection(dataflow, ComponentX, ComponentY, Service, DataEnc) & connection(dataflow, ComponentY, ComponentX, Service, DataEnc))))).

input_formula(requirement_1, conjecture, (? [ComponentX, ComponentY, Service] : (connection_without_firewall_regulation (dataflow, ComponentX, ComponentY, Service, plaindata) & type_of_component(ComponentX, insecure) & type_of_component(ComponentY, critical)))).
Formal Security Analysis

```
...

(!
  knows(ArgC_3)
  & equal(fst(ArgC_3), type_serverkeyexchange)
  & equal(snd(ext(snd(snd(ArgC_3), k_ca)), skey))
  & equal(snd(snd(ArgC_2), k_ca), fst(snd(ArgC_3)))))
=>
  (!
    (knows(ArgC_4_1)
    & equal(ArgC_4_1, type_serverhelldone)))
=>
  (!
    (true & equal(ClientKeyExchange, enc(premasterkey, skey))
    & equal(Certificate, (bl_HDR, cert)))
    & equal(HandshakeDone, (bl_HDR, sha_3)))

%----------------------------- Conjecture --

input_formula(attack, conjecture, (k
  knows(mastersecret) ))).
```

analyzing results ... 

_model found/total failure_

time limit information: 19 total / 18 strategy 
(leaving wrapper).
task myUML_PID1491 on atbroy1 has status SUCCESS 
(model found by strategy 300) consuming 1 seconds 
deleting temporary files.
e-SETHEO done. exiting
Security Analysis Results

The security properties that were considered were found to be enforced (examples with details in paper).

Note: 100% security proof is impossible for principled reasons. Goal is optimal cost effectiveness for finding weaknesses in highly security-critical system parts.
Revisit Evaluation Criteria

Evaluated security analysis approach under following criteria:

- Reproducability
- Delegatability
- Efficiency
- Parallelization
- Traceability
- Expressiveness
Some Insights

- Model-based development with notations such as UML does incur effort.
- That effort seems manageable when applied to core critical parts of the system.
- It seems to be justifiable in case of high assurance needs (e.g. in security).
- We believe it to compare favorably with traditional assurance methods offering a similar degree of trustworthiness.
- UMLsec seems to be well-suited for the domain of mobile communication systems.
Possible Improvements

Experiences indicated potential improvements:

• extend notation further (e.g. to cover remaining 13 requirements)
• provide tool support for these extensions
• improve simplicity of parts of the existing tool support

Link models to code using model-based monitor generation or testing [cf SESS talk tomorrow].
Some Other Applications

Analyzed designs / implementations / configurations for:

- biometry, smart-card or RFID based identification
- authentication (crypto protocols)
- authorization (user permissions, e.g. SAP systems)

Analyzed security policies, e.g. for privacy regulations.
Some Related Work

Other applications of UMLsec:
• Apvrille, Pourzandi (IEEE Security & Privacy, 2005)
• Best, Jurjens, Nuseibeh (ICSE 2007)

Other approaches for UML + security:
• RBAC: Fernandez et al., Basin et al., Breu et al., Koch/Parisi-Presicce, ...
• Aspect-Oriented Modeling (France et al.)
• Model-based Risk Assessment (CORAS project, Stoelen, Houmb et al.)
• Agents: Yoshioka, Honiden, Finkelstein
• Misuse cases: Whittle (earlier this conf.)
• ...

Conclusions

Application of the UMLsec approach in industrial setting.

Model-based security analysis of mobile communications architecture and related security policies. Benefits were:

• consideration of security goals within a standard industrial design technique

• automated security analysis

The approach was found to be applicable with justifiable training and time effort.