Chapter 1:
Introduction

Raimundas Matulevičius

University of Tartu, Estonia, rma@ut.ee
Goal

• Introduce the bases used in the remaining of the book. This includes
  – Definition of the system and security engineering
  – Overview of few security and security risk management standards, security development approaches and modelling perspectives
Outline

• System and Security Engineering
• Security and Security Risk Management Standards
• Security Development Approaches and Languages
• Model-Driven Security
• Modelling Perspectives for Secure Software Systems
• Running Example
Outline

- System and Security Engineering
  - Security and Security Risk Management Standards
  - Security Development Approaches and Languages
  - Model-Driven Security
  - Modelling Perspectives for Secure Software Systems
  - Running Example
What is System?
What is **System**?

[Anderson, 2008]

- **Component**
  - smartcard, a PC or piece of software
What is **System**?

[Anderson, 2008]

- **Component**
  - smartcard, a PC or piece of software
- **Infrastructure**
  - Operating system, network, etc
What is **System**?

[Anderson, 2008]

- **Component**
  - smartcard, a PC or piece of software

- **Infrastructure**
  - Operating system, network, etc

- **Applications**
What is **System**?

[Anderson, 2008]

- **Component**
  - smartcard, a PC or piece of software
- **Infrastructure**
  - Operating system, network, etc
- **Applications**
- **IT staff**
What is **System**?

[Anderson, 2008]

- **Component**
  - smartcard, a PC or piece of software
- **Infrastructure**
  - Operating system, network, etc
- **Applications**
- **IT staff**
- **Internal users and management**
What is System?

[Anderson, 2008]

- Component
  - smartcard, a PC or piece of software
- Infrastructure
  - Operating system, network, etc
- Applications
- IT staff
- Internal users and management
- Customers and external users
What is **System**?

[Anderson, 2008]

- **Component**
  - smartcard, a PC or piece of software
- **Infrastructure**
  - Operating system, network, etc
- **Applications**
- **IT staff**
- **Internal users and management**
- **Customers and external users**
- **Environment**
What is System? [Anderson, 2008]

- Component
  - smartcard, a PC or piece of software
- Infrastructure
  - Operating system, network, etc.
- Applications
- IT staff
- Internal users and management
- Customers and external users
- Environment

EVERYTHING !!!
How to Crack Encrypted Message?
How to Crack Encrypted Message?

• Acquire massive amount of computing power and brute-force all the possible values of the encryption key?
  – The cryptographer's dream scenario
How to Crack Encrypted Message?

- Although, what about the case where the key is **easily guessable password**?

<table>
<thead>
<tr>
<th>password</th>
<th>master</th>
</tr>
</thead>
<tbody>
<tr>
<td>123456</td>
<td>sunshine</td>
</tr>
<tr>
<td>12345678</td>
<td>ashley</td>
</tr>
<tr>
<td>qwerty</td>
<td>bailey</td>
</tr>
<tr>
<td>abc123</td>
<td>passw0rd</td>
</tr>
<tr>
<td>monkey</td>
<td>shadow</td>
</tr>
<tr>
<td>1234567</td>
<td>123123</td>
</tr>
<tr>
<td>letmein</td>
<td>654321</td>
</tr>
<tr>
<td>trustno1</td>
<td>superman</td>
</tr>
<tr>
<td>dragon</td>
<td>qazwsx</td>
</tr>
<tr>
<td>baseball</td>
<td>michael</td>
</tr>
<tr>
<td>111111</td>
<td>football</td>
</tr>
<tr>
<td>lloveyou</td>
<td></td>
</tr>
</tbody>
</table>

© Springer International Publishing AG 2017
How to Crack Encrypted Message?

• Or, better yet, why not just ask for password?
How to Crack Encrypted Message?

• How about accessing the computer and installing key-logger or trojanised version of the message viewer?
  – Maybe there is already has some remote-controlled malware installed
  – Maybe the decrypted message could be read from computer's memory or hard disk?
What is Security engineering?
What is **Security engineering**?

[Firesmith, 2003]

**Security engineering**

is concerned with lowering the risk of intentional unauthorized harm to valuable assets to level that is acceptable to the system’s stakeholders by preventing and reacting to malicious harm, misuse, threats, and security risks.
What is **Security engineering**?

[Firesmith, 2003]

Security engineering is concerned with lowering the risk of intentional unauthorized harm to valuable assets to a level that is acceptable to the system's stakeholders by preventing and reacting to malicious harm, misuse, threats, and security risks.

**Different from safety where focus is on unintentional harm**

It is **impossible** to make 100% secure systems

Stakeholders' **values** must be protected

Risk can be of different form

[Firesmith, 2003]

© Springer International Publishing AG 2017

21
Outline

- System and Security Engineering
- **Security and Security Risk Management Standards**
- Security Development Approaches and Languages
- Model-Driven Security
- Modelling Perspectives for Secure Software Systems
- Running Example
Standards

• **Security standards**
  – Define guidelines for information security management
    • ISO/IEC 13335-1:2004
      – security concepts and models fundamental for a basic understanding of information and communication technology security
    • Common Criteria
      for information technology security evaluation
      – Security requirements and desired security goals

• **Security risk management standards**
  – Security risk management activities
    • ISO/IEC 2700x series
    • NIST special publication
    • BSI standards 100 series for information technology
Outline

• System and Security Engineering
• Security and Security Risk Management Standards
• **Security Development Approaches and Languages**
• Model-Driven Security
• Modelling Perspectives for Secure Software Systems
• Running Example

© Springer International Publishing AG 2017
Security Development Approaches

• Security requirements engineering methods
  • Fabian et al., 2010
  • Mellado et al., 2010

• Domain-specific languages for security modelling
  – Security scenarios with domain specific language
  – CORAS modelling language for security and safety risk analysis
  – CySeMol – cyber security modelling language
  – KAOS extensions to security

Brooke et al., 2012
Lund et al., 2011
Sommestad et al., 2013
van Lamsweerde, 2004
Modelling Languages

Early requirements | Late requirements | Architectural design | Detailed design | Implementation and testing

BPMN

i* (actor and goal modelling)

KAOS (goals for software spec.)

Use cases

Activity diagrams

Class diagrams

Component diagrams

© Springer International Publishing AG 2017
Security Modelling Languages

- Early requirements
- Late requirements
- Architectural design
- Detailed design
- Implementation and testing

- Security Risk-oriented BPMN
- Secure TROPOS
- KAOS extension to security
- Misuse cases
- Mal-activity diagrams
- UMLsec
- SecureUML

© Springer International Publishing AG 2017
Outline

• System and Security Engineering
• Security and Security Risk Management Standards
• Security Development Approaches and Languages
  • Model-Driven Security
• Modelling Perspectives for Secure Software Systems
• Running Example
Model-Driven Development
[Staron, 2006]
Model-Driven Security

• Application of the model-driven technologies to develop secure software systems
  – Transformation-oriented approach
  – Generate security constraints from security models

[Yue et al., 2011]
Outline

• System and Security Engineering
• Security and Security Risk Management Standards
• Security Development Approaches and Languages
• Model-Driven Security
  • Modelling Perspectives for Secure Software Systems
• Running Example
Modelling Perspectives

[Krogstie, 2012]

- Behavioural
- Functional
- Structural
- Goal and Rule
- Object
- Communication
- Actor and role
- Topological
Modelling Perspectives

[Krogstie, 2012]

- Behavioural
- **Functional**
- Structural
- Goal and Rule

- **Major phenomenon**
  - Transformation is activity to transform state of phenomenon to a different set of phenomenon

- **BPMN and Activity diagrams**
  - Flow of objects
  - Swimlanes
  - Artefacts
  - Connecting objects

- **BPMN** for organisation’s workflows
- **Activity diagrams** for dynamic aspects of considered software system

© Springer International Publishing AG 2017
Modelling Perspectives

• Behavioural
• Functional
• Structural
• Goal and Rule

• Major phenomenon
  – States and transitions between them

• Object
• Communication
• Actor and role
• Topological

• Use case diagrams
  – Use case – declaration of behaviour, characterised by
    • Activities
    • Interactions
    • States
    • Preconditions
    • Post conditions
Modelling Perspectives

[Krogstie, 2012]

• Behavioural
• Functional
• Structural
• Goal and Rule

• Major phenomenon
  – Goal and Rule
    • Something that needs to be achieved
    • Something that needs to be satisfied
  – Actor and Role
    • Actors, roles and their dependencies

• Object
• Communication
• Actor and role
• Topological

• Secure Tropos
  – Actors and their dependencies
  – Goal elicitation, refinement, performance of plans, usage of available resources

© Springer International Publishing AG 2017
Modelling Perspectives

[Krogstie, 2012]

- Behavioural
- Functional
- Structural
- Goal and Rule

- Object
- Communication
- Actor and role
- Topological

- Ordering between different perspectives
Capturing Various Perspectives
Capturing Various Perspectives

- Security concerns in different modelling languages
- Systematic security engineering process
- Model enhancement
- Model-driven security
Outline

• System and Security Engineering
• Security and Security Risk Management Standards
• Security Development Approaches and Languages
• Model-Driven Security
• Modelling Perspectives for Secure Software Systems

• Running Example
Football Federation

ENVIRONMENT

Team Representative

Umpire

Other system in environment

Football Federation Employee

Administrator

USER INTERFACE

INFORMATION PROCESSING SYSTEM

DATABASE

ERIS

Football Federation

FOOTBALL FEDERATION
Football Federation
Football Federation
Football Federation
Summary

• System and Security Engineering
• Security and Security Risk Management Standards
• Security Development Approaches and Languages
• Model-Driven Security
• Modelling Perspectives for Secure Software Systems
• Running Example