Lecture 4:

**SECURITY MODELLING:**

Alignment between different modelling perspectives

---

**What we have studied so far?**

- Security Risk-oriented BPMN
- Security Risk-aware Secure Tropos
- Security Risk-oriented Misuse Cases
- Mal-activities for Managing Security Risks
Outline

• Modelling perspectives
• Language limitations
• Transformation principles
  – From Functional (organisational) to Goal and rule/Actor and role perspective
  – From Goal and rule/Actor and role to Behavioural perspective
  – From Behavioural to Goal and rule/Actor and role perspective
  – From Behavioural to Functional (software system) perspective
• What’s next?
Modelling Perspectives

[Krogstie, 2012]

- Behavioural
- Functional
- Structural
- Goal and Rule

- Object
- Communication
- Actor and role
- Topological

- **Major phenomenon**
  - Transformation as 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
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

- Major phenomenon
  - States and transitions between them

- Use case diagrams
  - Use case – declaration of behaviour, characterised by
    - Activities
    - interactions
    - States
    - Preconditions
    - Port conditions

- Secure Tropos
  - Actors and their dependencies
  - Goal elicitation, refinement, performance of plans, usage of available resources
Modelling Perspectives

[Krogstie, 2012]

- Behavioural
- Functional
- Structural
- Goal and Rule

- Object
- Communication
- Actor and role
  - Topological

- Ordering between different perspectives

Outline

- Modelling perspectives

  - Language limitations

  - Transformation principles
    - From Functional (organisational) to Goal and rule/Actor and role perspective
    - From Goal and rule/Actor and role to Behavioural perspective
    - From Behavioural to Goal and rule/Actor and role perspective
    - From Behavioural to Functional (software system) perspective

- What’s next?
<table>
<thead>
<tr>
<th>ISSRM</th>
<th>BPMN</th>
<th>Secure Tropos</th>
<th>Misuse cases</th>
<th>Mal-activity diagrams</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assets</td>
<td>Event Gateway combined using Sequence flows</td>
<td>Combined using dependency, contribution, means-ends, and decomposition links</td>
<td>Combined using communication, extends, includes links</td>
<td></td>
</tr>
<tr>
<td>Business asset</td>
<td>Data Object</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IS asset</td>
<td>Data Store</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Security criterion</td>
<td>CIA</td>
<td>Combined using contribution and security constraint decomposition links</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Added to the Business asset constructs, such as Task or Data Object.

**Mal-activity diagrams**

- Combined using control flow links.

---

<table>
<thead>
<tr>
<th>ISSRM</th>
<th>BPMN</th>
<th>Secure Tropos</th>
<th>Misuse cases</th>
<th>Mal-activity diagrams</th>
</tr>
</thead>
<tbody>
<tr>
<td>Risk</td>
<td></td>
<td>Combination of Event and Impact</td>
<td>Combination of Event and Impact</td>
<td>Combination of Event and Impact</td>
</tr>
<tr>
<td>Impact</td>
<td></td>
<td>Impacts</td>
<td>Impact</td>
<td>Mal-activity contained in the mal-swimlane that expresses attack method</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Event</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Combination of Vulnerability, and Threat</td>
<td>Combination of Vulnerability and Threat</td>
<td>Combination of Threat and Vulnerability; if it is implicitly defined</td>
</tr>
<tr>
<td>Vulnerability</td>
<td></td>
<td>added to the IS asset constructs, such as Task or Data Store</td>
<td>Vulnerability</td>
<td>--</td>
</tr>
<tr>
<td>Threat</td>
<td></td>
<td>Combination of Attack method and Threat agent</td>
<td>Combination of Attack method and Threat agent</td>
<td>Combination of Attack method and Threat agent</td>
</tr>
<tr>
<td>Attack method</td>
<td></td>
<td>Task</td>
<td>Plan potentially combined with other Tasks using decomposition links</td>
<td>Missuse case potentially combined with other misuse cases using includes and extends links</td>
</tr>
<tr>
<td>Threat agent</td>
<td></td>
<td>Actor</td>
<td>Misuse case</td>
<td>Misuse case</td>
</tr>
</tbody>
</table>
Limitations of Modelling Languages

- **Redundancy** – two language constructs have the same or overlapping semantics
- **Overload** – same language construct has several meanings
- **Incompleteness** - language does not convey information on a certain phenomenon
- **Under-definition** (or excess) arises when a language construct has no semantics
Outline

• Modelling perspectives
• Language limitations

• **Transformation principles**
  – From **Functional** (organisational) to **Goal and rule/Actor and role** perspective
  – From **Goal and rule/Actor and role** to **Behavioural** perspective
  – From **Behavioural** to **Goal and rule/Actor and role** perspective
  – From **Behavioural** to **Functional** (software system) perspective

• What’s next?

Language Transformation

• Capture the security concerns using different modelling languages
  – Representations familiar to the various stakeholders
  – No need to learn new notation

• Ensure systematic and consistent security engineering process through different development stages
  – Maintain traceability

• Enhance the model of the considered system
Language Transformation Concepts

Capturing Multiple Modelling Perspectives through Transformation
Outline

- Modelling perspectives
- Language limitations
- Transformation principles
  - From Functional (organisational) to Goal and rule/Actor and role perspective
  - From Goal and rule/Actor and role to Behavioural perspective
  - From Behavioural to Goal and rule/Actor and role perspective
  - From Behavioural to Functional (software system) perspective
- What’s next?
From
Security Risk-oriented BPMN
to
Security Risk-aware Secure Tropos

• To understand what organisation’s goals can be achieve by performing or another scenario
• Enhance the model with other organisational goals and illustration on how they could be security achieved

TR.BPMN-ST.1: BPMN *pool* is translated to the Secure Tropos *actor*. 
**TR.BPMN-ST.2**: In Secure Tropos dependencies between actors are defined following the BPMN messages flows.

**TR.BPMN-ST.3**: BPMN task is defined as plan in Secure Tropos model.

**TR.BPMN-ST.4**: BPMN event is defined as hardgoal in Secure Tropos model.
TR.BPMN-ST.5: BPMN *data object* and *data store* are translated to *resource* in Secure Tropos model.

TR.BPMN-ST.6: BPMN *lock* construct and its associated comment are translated to the Secure Tropos *security constraint*. The *restrict* relationship is defined from the *security constraint* to the appropriate construct.
TR.BPMN-ST.7: BPMN pool that presents the threat agent is translated to the Secure Tropos (malicious) actor.

TR.BPMN-ST.8: BPMN plans and message flows, used to describe the attack method are transformed to plans and situated within the malicious actor boundary in Secure Tropos model.
TR.BPMN-ST.9: BPMN vulnerability construct is translated to the vulnerability point and placed to the exploited constructs in Secure Tropos model.

TR.BPMN-ST.10: BPMN task that present the security requirement is transformed to the Secure Tropos secure plan.
Needs to be defined:

- **decomposition** and **means-ends** relationships to illustrate how **plans** are decomposed and **goals** are achieved;
- **security criteria** to restrict dependencies among the actors;
- **malicious goal(s)** to complete definition of the **security threats**;
- **exploits** and **targets** relationships to show how **vulnerabilities** (i.e., **vulnerable points**) are exploited and system assets are attacked.
- aggregated attack view to illustrate how the security events could be mitigated.
Let’s continue analysis!
Outline

• Modelling perspectives
• Language limitations
• Transformation principles
  – From **Functional** (organisational) to **Goal and rule/Actor and role** perspective
  – From **Goal and rule/Actor and role** to **Behavioural** perspective
  – From **Behavioural** to **Goal and rule/Actor and role** perspective
  – From **Behavioural** to **Functional** (software system) perspective
• What’s next?
From

Security Risk-aware Secure Tropos
to

Security Risk-oriented Misuse Cases

• Understand the major secure software system functions
• Advance the security model with the links between the functional and security requirements

Transformation Rules

TR.ST-MUC.1: An actor who presents the (software) system in Secure Tropos is translated to a misuse case software system boundary.
TR.ST-MUC.2: Secure Tropos goal and plan, which belong to the (software) system Tropos actor, are translated to use case in the misuse case diagram.
TR.ST-MUC.3: Secure Tropos means-ends and decomposition links are translated to misuse case includes relationship.
TR.ST-MUC.4: Secure Tropos actor that presents the organisational actor (or the actor from the system environment) is translated to a misuse case actor.
TR.ST-MUC.5: Secure Dependency links from Secure Tropos are translated to communication association in the misuse case diagram.
TR.ST-MUC.7: Secure Tropos restricts link is translated to misuse case constraint of relationship

TR.ST-MUC.8: Secure Tropos actor who exploits and/or targets elements of other actors, is translated to the misuser in the misuse case diagram.
TR.ST-MUC.9: Secure Tropos goal and plan that belong to the actor who exploits and/or targets constructs of other actors, are translated to the misuse case in the misuse case diagram.
TR.ST-MUC.10: Secure Tropos vulnerability point is translated to a vulnerability in the misuse cases diagram. The includes relationships should be defined between the asset use case and this vulnerability.
TR.ST-MUC.11: Secure Tropos exploits link should be represented by two relationships in misuse case diagram: (i) as threatens relationship from the misuse case to the asset use case, and (ii) as exploit relationship from the misuse case to the asset vulnerability.
TR.ST-MUC.12: Secure Tropos goal and plan, which express security requirements (and controls) are translated to the (security) use case in the misuse case diagram.
TR.ST-MUC.13: A Secure Tropos mitigates link is translated to a mitigates relationship in the misuse cases diagram.
Demo

• From Security Risk-aware Secure Tropos to Security Risk-oriented Misuse Cases

• From Security Risk-oriented Misuse Cases to Security Risk-aware Secure Tropos
Outline

• Modelling perspectives
• Language limitations
• Transformation principles
  – From **Functional** (organisational) to **Goal and rule/Actor and role** perspective
  – From **Goal and rule/Actor and role** to **Behavioural** perspective
  – From **Behavioural** to **Goal and rule/Actor and role** perspective
  – From **Behavioural** to **Functional** (software system) perspective
• What’s next?

Demo

• From Security Risk-aware Secure Tropos to Security Risk-oriented Misuse Cases

• From Security Risk-oriented Misuse Cases to Security Risk-aware Secure Tropos
Let’s continue analysis!
Outline

• Modelling perspectives
• Language limitations
• Transformation principles
  – From Functional (organisational) to Goal and rule/Actor and role perspective
  – From Goal and rule/Actor and role to Behavioural perspective
  – From Behavioural to Goal and rule/Actor and role perspective
  – From Behavioural to Functional (software system) perspective

• What’s next?
To understand the how separate secure system functions should be performed by the software system

What are security threats to the software system

What are the security countermeasures, i.e., controls

TR.MUC-MAL.1: A system boundary that presents software system in the misuse case diagram is translated to the swimlane in Mal-activities.

TR.MUC-MAL.2: A (mis)use actor is translated to the swimlane in Mal-activities.
TR.MUC-MAL.3: A use case construct is translated to the activity construct.

TR.MUC-MAL.4: The (mis)use case security constraint is translated to the (mal)activity security constraint.
The misuse case construct is translated to the mal-activity.

The misuser is translated to the mal-swimlane.
TR.MUC-MAL.7: The misuse case vulnerability is translated to the mal-activity vulnerability.

TR.MUC-MAL.8: A security use case is translated to mitigation activity in mal-activities.
Needs to be defined

• *swimlanes* to present *attack method* and *control*;

• under which *swimlanes* transformed *activities* and *mal-activities* need to be situated;

• relationships (i.e., *flows*) to define the sequences among *activities*, *mal-activities*, and *security activities*;

• *security constraint* and *vulnerability* places in the model.
Outline

• Modelling perspectives
• Language limitations
• Transformation principles
  – From **Functional** (organisational) to **Goal and rule/Actor and role** perspective
  – From **Goal and rule/Actor and role** to **Behavioural** perspective
  – From **Behavioural** to **Goal and rule/Actor and role** perspective
  – From **Behavioural** to **Functional** (software system) perspective

• What’s next?

Capturing Multiple Modelling Perspectives through Transformation
Let’s continue analysis!
Message to take home

- Modelling perspectives
- Language limitations
- Transformation principles
  - **Functional** (organisational) \(\rightarrow\) **Goal and rule/Actor and role**
  - **Goal and rule/Actor and role** \(\rightarrow\) **Behavioural**
  - **Behavioural** \(\rightarrow\) **Goal and rule/Actor and role**
  - **Behavioural** \(\rightarrow\) **Functional** (software system) perspective
- Model driven security