Exercise

Role-based Access Control

Goals

• Learn and apply languages for role-based access control modelling
• Understand the limitations of these languages
Consider the Following Scenario (slide 1)

The football game report is started before the match, continued during the match, and finalised once the match is over. The new report is created by the league secretary, who is employed by the Football Federation. The league secretary fills in initial information such as league name, his own contact data, region, division, and game numbers as well as the names of the competing teams. Once done, the league secretary informs the team representatives, so that they could provide the team composition for the game, including player names and their registration numbers. The team composition should be provided no later than 2 h before the game.

Consider the Following Scenario (slide 2)

During the match the game report needs to be maintained by the umpire, who needs to register the scored goals and “given” cards. The umpire is also responsible for filling in the match results, including the scores (including final, full time, extra time and penalties). After the match, the umpire needs to invite the team representatives to sign the report and (optionally) provide comments. Once signatures from the team representatives are received, the umpire himself comments and signs the report. After the game the report is sent to the league secretary for confirmation.
Task 1

- Define the SecureUML model representing role-based access control policy regarding the gathered and registered data.
Task 1

• Define the SecureUML model representing role-based access control policy regarding the gathered and registered data

To guide your solution, consider
• What are objects and their concerned attributes?
• What operations do change the values of the attributes?
• What are roles?
• What are the security actions?
• What are the permissions of roles towards the object?
• Who are the users?

• The question in blue should explicitly be answered in a separate diagram
• Answers to other questions can be given directly in the defined SecureUML model
Task 1

• Define the **SecureUML** model representing role-based access control policy regarding the gathered and registered data

  • What authorisation constraints should be defined in your model?
    - They can be defined in any *formal*, *semiformal* or *natural* language (in case of natural language, use English 😊)

Task 2

• Define the **UMLsec** model representing the role-based access control policy regarding the gathered and registered data
Task 2

• Define the UMLsec model representing the role-based access control policy regarding the gathered and registered data.

To guide your solution, consider:
• What are the objects?
• What are their operations?
• What are the roles?
• What are the role’s rights?
• What are the associated tags?
• Who are the users?

{\text{protected} = \text{protected\_action}}
{\text{role} = (\text{actor, role})}
{\text{right} = (\text{role, protected\_action})}
Task 3

Explain what RBAC policy concerns are captured in your...

... SecureUML model and *not* in the UMLsec model

... UMLsec model and *not* in the SecureUML model

Submit your solutions

- Deadline: **Monday – 29th April**
  - Do not be late!!
  - Late submissions assessed with 50% penalty

- **PDF file**
  (other format will not be checked)

- Submit using the **Upload** function:
  <https://courses.cs.ut.ee/2019/ssd/spring/Main/Upload>