Secure Component Based Applications Through Security Patterns

Rahma Bouaziz, Bernard Coulette

IRIT – MACAO team

University of Toulouse, France

3SL Workshop - 20 /11/2012

Plan

- Context and motivations
- Definitions and State of the Art
- UML Profile: Mechanism for integrating security patterns
- Use case: The GPS system
- Conclusion and Perspectives

Plan

- Context and motivations
- Definitions and State of the Art
- UML Profile: Mechanism for integrating security patterns
- Use case: The GPS system
- Conclusion and Perspectives

Complexity of current Systems

Current software applications:

- are more and more complex
- must satisfy increasing security requirements
- need an expertise to introduce the security

Knowledge about security:

- Notion of security patterns [Yoder et al.:2005]
- A set of Security patterns encapsulate security expert's knowledge.




Problematic

Knowledge gap between Software Developers and Security Specialists

- Several security patterns have been proposed, but:
  - they do not constitute an intuitive solution
  - they do not provide clear directives to guide the developer in their practical application.

For reuse purpose, complex applications are often based on components.

➔ **OUR ISSUE IS : How to guide designers in integrating security patterns into components models?**

Plan

- Context and motivations
- Definitions and State of the Art
- UML Profile: Mechanism for integrating security patterns
- Use case: The GPS system
- Conclusion and Perspectives

COMPONENTS ENGINEERING

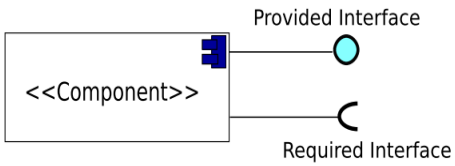
Software Component:

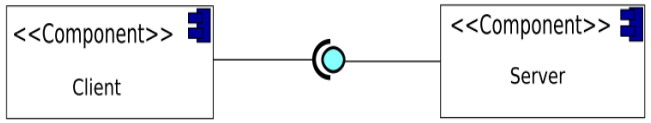
"autonomous and reusable entity interacting with its environment through required and provided interfaces" [Szyperski:02]

Existing components Models :

- Components conceptual models: SOFA, UML 2,0, Fractal, …
- Architecture description languages: Fast, Darwin, Wright, …
- Components Programming languages : ArchJava, Java / A

The component approach is mainly based on the principle of reuse by assembling components.





SECURITY PATTERNS

Pattern :

"Generic solution to a recurring problem in a given context" [Gamma:95],[Alexander:97]

Security Patterns:

- Encapsulate the knowledge of experts in security
- Provide developers with guidelines for reusing human security expertise

Security patterns classification [Schumacher et al. 2006]:

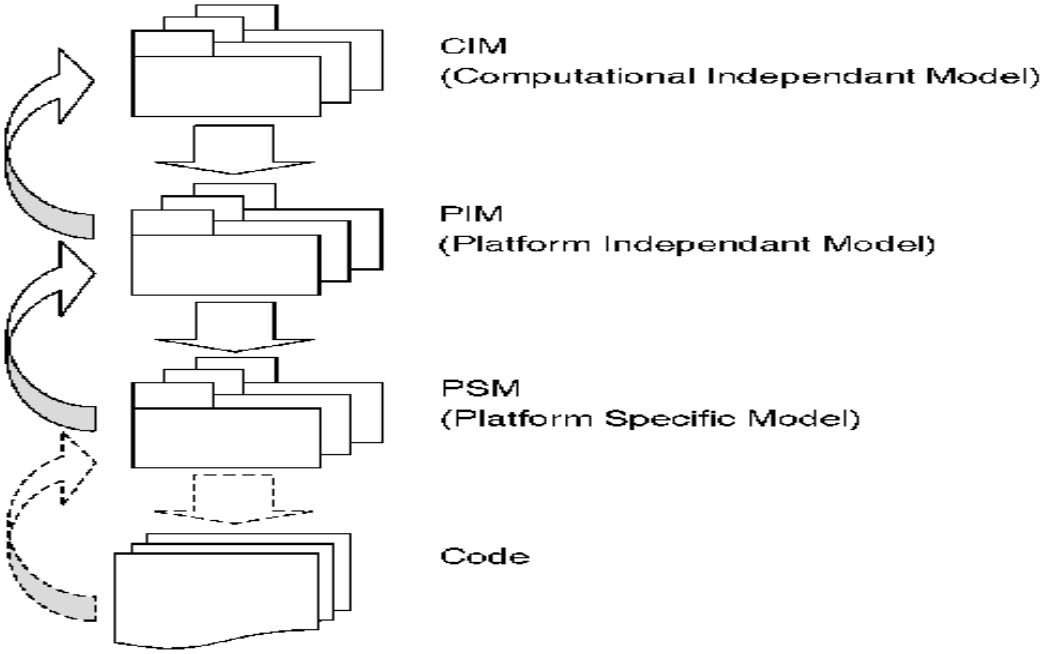
- Identification and authentication
- Access control
- …

There are dependencies (e.g. sequencing) and relationships between security patterns

Model Driven Engineering

Principles:

- Models are considered as first class citizens
- The design process can be seen as a set of model transformations
- Each transformation takes models as input and produces models as output until obtaining the code.

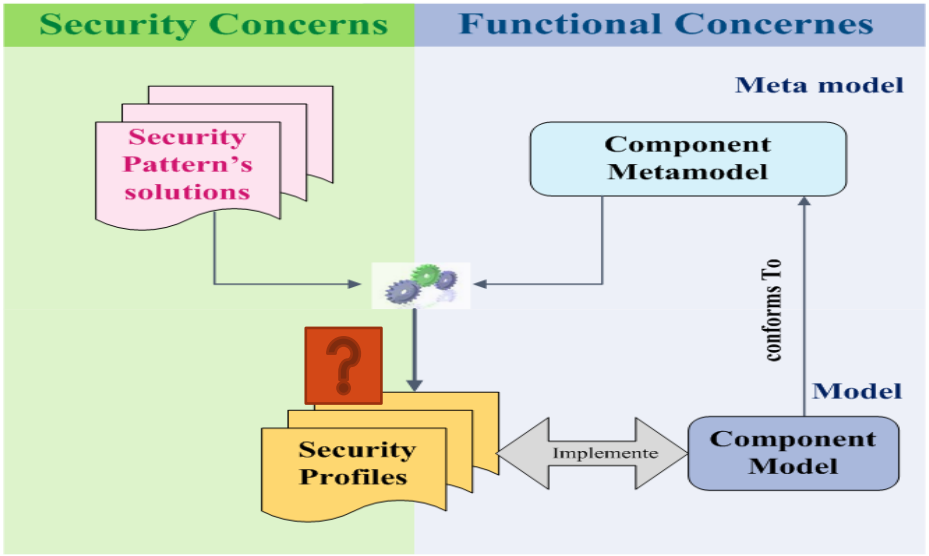


Plan

- Context and motivations
- Definitions and State of the Art
- UML Profile: Mechanism for integrating security patterns
- Use case: The GPS system
- Conclusion and Perspectives

PROPOSITION

A FRAMEWORK FOR PATTERN INTEGRATION

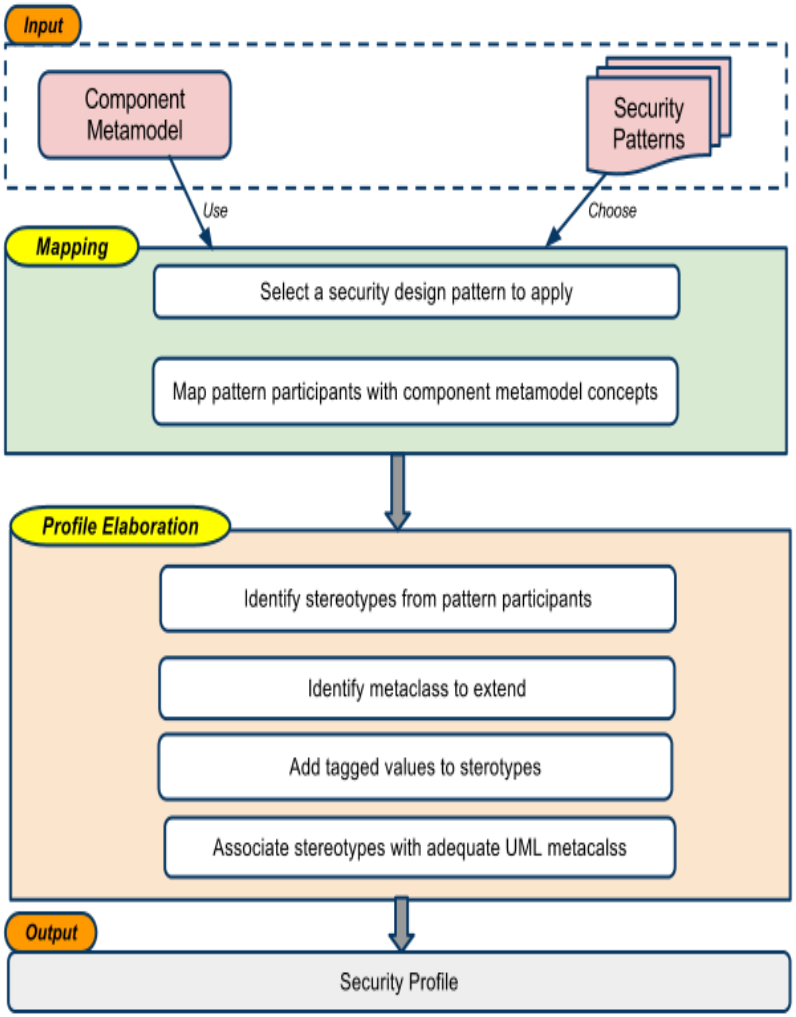


The proposed framework:

- allows using the security knowledge encapsulated in security patterns to produce a secure component-based model.
- presents two dimensions: functional concepts and security concerns.

The security UML profile is applied to elements of a Component application model which is an instance of the component metamodel.

PROFILE CONSTRUCTION PROCESS



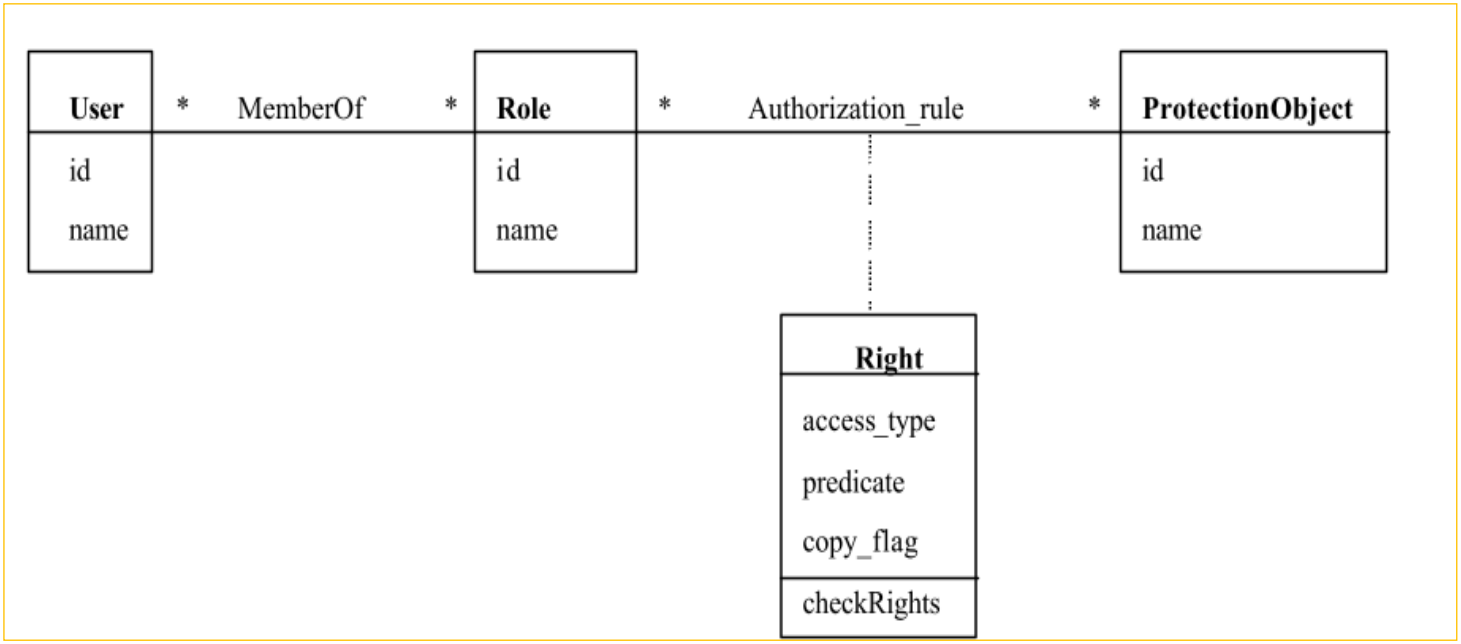
- This process allows converting the abstract solution contained in security patterns to concrete solutions adapted to Component domain concepts.
- Concrete solutions are represented as UML profiles which are used to apply expert design experience described in security patterns.

ILLUSTRATION

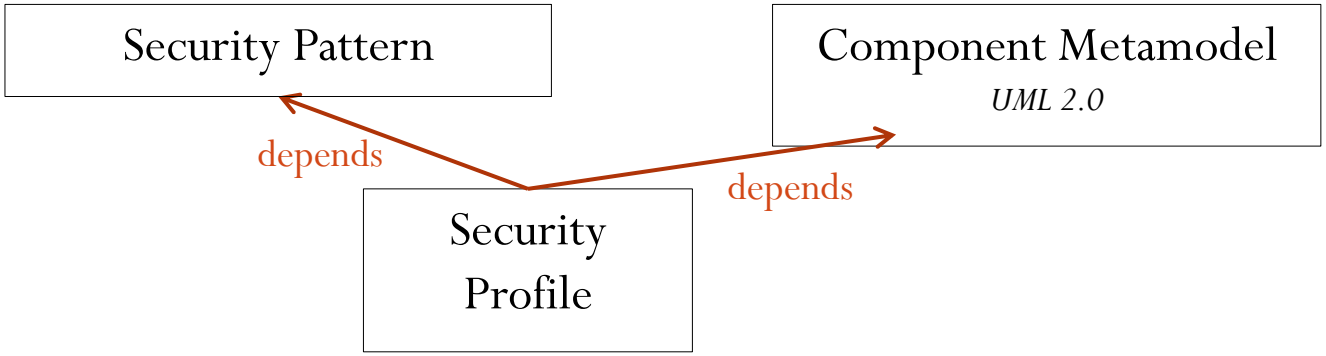
Role Based Access Control (RBAC) pattern

(adapted from [Fernandez, 2006])

- Users have different roles that require different skills and responsibilities, and therefore should have different rights of access to data.



ILLUSTRATION



MAPPING: relation between security pattern participants and component metamodel elements.

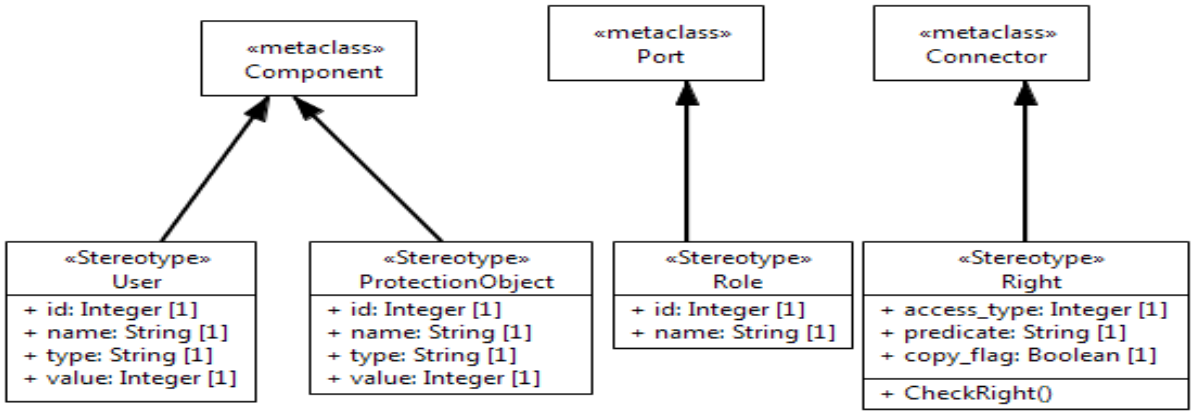
- Example with RBAC pattern:

Access Control Pattern	Access Control Pattern participants	Component metamodel element
RBAC Pattern	Right	Connector
	User	Component
	Role	Port
	ProtectionObject	Component

ILLUSTRATION

RBAC PROFILE ELABORATION : produce a UML profile according to the previous mapping:

- define stereotypes and tagged values, identify metaclasses to be extended and associate these metaclasses with defined stereotypes.
- all pattern's participants are specified as UML stereotypes.
- RBAC pattern's participant's properties are added respectively as tagged values to the corresponding stereotypes.
- each identified stereotype extends a specific metaclasses

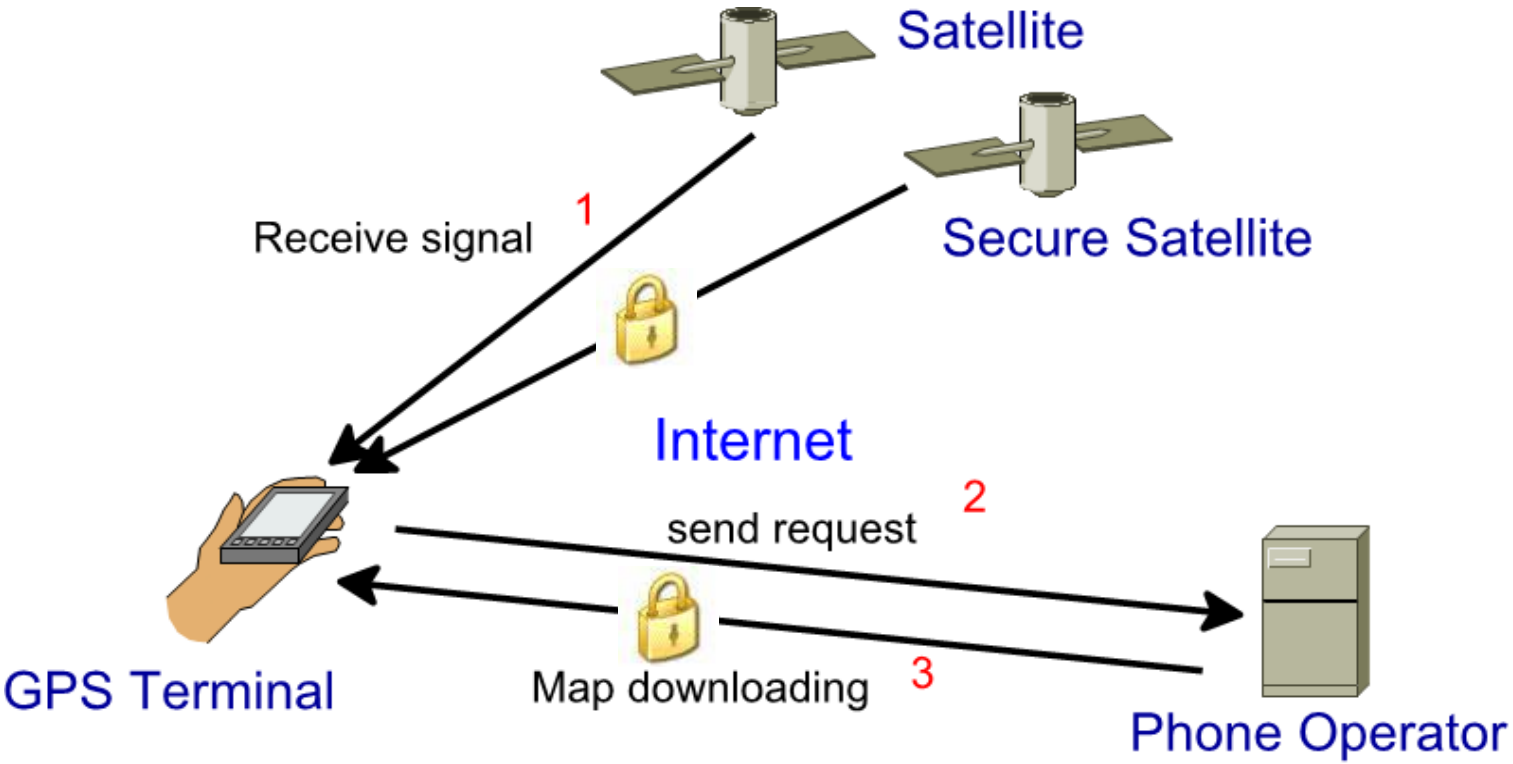


Plan

- Context and motivations
- Definitions and State of the Art
- UML Profile: Mechanism for integrating security patterns
- Use case: The GPS system
- Conclusion and Perspectives

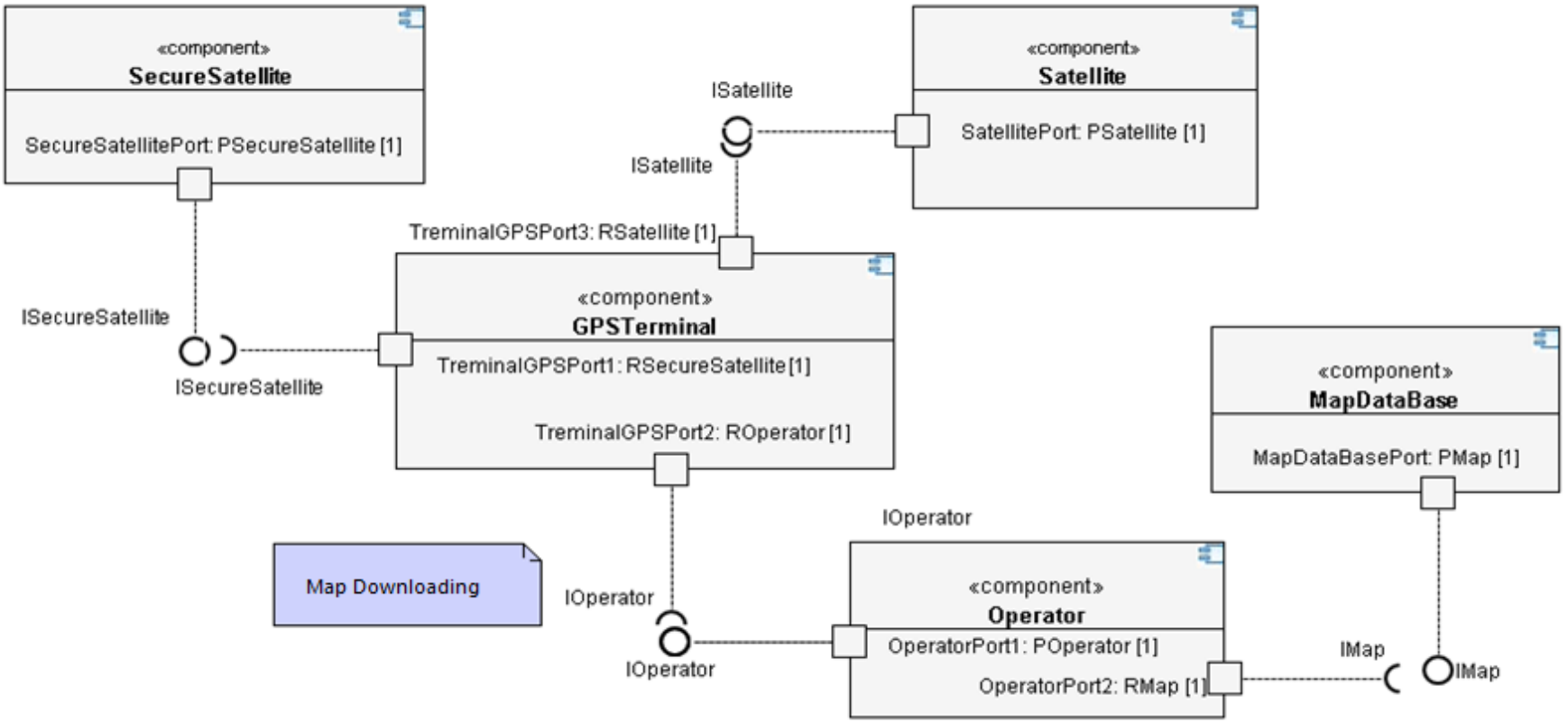
USE CASE: THE GPS SYSTEM

Description of the Basic GPS system and use cases



USE CASE: THE GPS SYSTEM

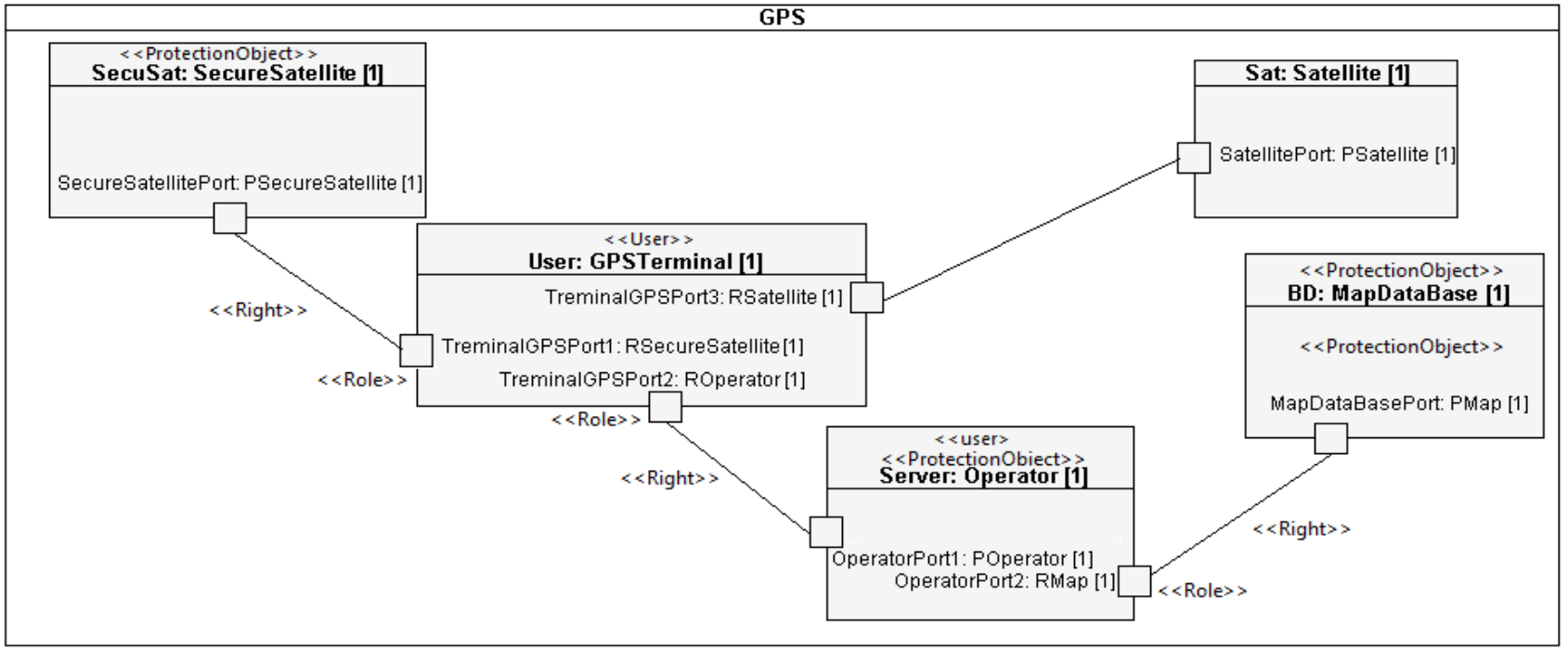
The GPS modeling is made using the UML component diagram



USE CASE: THE GPS SYSTEM

We apply the formalism of composite structure UML diagram:

- to model the Basic GPS system
- to stereotype system elements with the produced RBAC UML profile



Plan

- Context and motivations
- Definitions and State of the Art
- UML Profile: Mechanism for integrating security patterns
- Use case: The GPS system
- Conclusion and Perspectives

CONCLUSION

- We propose a novel approach that utilizes security patterns for injecting security into component based application designs
- Our technique enables security UML profile elaboration from security patterns that embed proven experts security solution
- We propose an UML profile construction process
- Such security profile extends the component metamodel to the security context.
- We have developed a case study (GPS system) to demonstrate the application of UML profile produced via our approach

PERSPECTIVES

- Cover a large number of security patterns proposed in literature.
- Automatic application of the produced profiles by establishing a method based on the description of security pattern transformation rules.
- Consider dependencies among security patterns when describing security pattern transformation rules (like between Authenticator, RBAC and Reference monitor patterns)
- Develop an assistance tool guiding component application's designer in security pattern integration in order to allow their easy.

THANK YOU FOR YOUR ATTENTION

ANY QUESTIONS ?

Rahma BOUAZIZ

http://www.irit.fr/~Rahma.Bouaziz