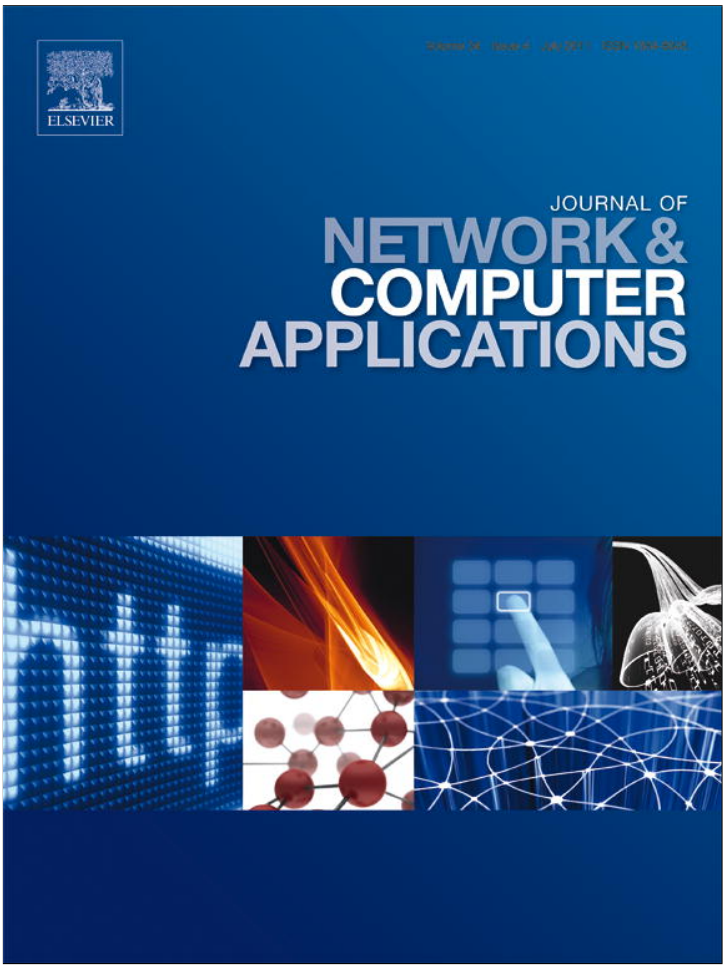

Journal of Network and Computer Applications 34 (2011) 1168–1183

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Systematic design of secure Mobile Grid systems

David G. Rosado [a,*], Eduardo Fernández-Medina [a], Javier López [b], Mario Piattini [a]

[a] University of Castilla-La Mancha, Information Systems and Technologies Department, Ciudad Real, Spain
[b] University of Málaga, Computer Science Department, Malaga, Spain

ARTICLE INFO

Article history:
Received 2 March 2010
Received in revised form
3 December 2010
Accepted 2 January 2011
Available online 19 January 2011

Keywords:
Mobile Grid computing
Security
Design
Development process
Security architecture

ABSTRACT

Grid computing has arisen as an evolution of distributed systems mainly focused on the sharing of and remote access to resources in a uniform, transparent, secure, efficient and reliable manner. It is possible to join Grid technology and mobile technology in order to create one of the most promising technologies and developments to appear in recent years, in that they enrich one another and provide new solutions that solve many of the limitations and problems found in different technologies. Security is a very important factor in Mobile Grid Computing and is also difficult to achieve owing to the open nature of wireless networks and heterogeneous and distributed environments. Success in obtaining a secure system originates in incorporating security from the first stages of the development process. It has therefore been necessary to define a development process for this kind of systems in which security is incorporated in all stages of the development and the features and particularities of the Mobile Grid systems are taken into consideration. This paper presents one of the activities of this development process, the design activity, which consists of defining and designing a security software architecture. This architecture will be built from a security architecture, defined as reference architecture, in which security services, interfaces and operations are defined with the purpose of defining a reference security architecture which covers the majority of security requirements identified in the analysis activity. The design activity will build the system architecture that will be the input artefact for the subsequent activity in the process, which is the construction activity.

1. Introduction

Grid computing has emerged to cater for the needs of computing-on-demand (Jana et al., 2009) resulting from the advent of distributed computing with sophisticated load balancing, distributed data and concurrent computing power using clustered servers. The Grid enables resource sharing and dynamic allocation of computational resources, thus increasing access to distributed data, promoting operational flexibility and collaboration, and allowing service providers to scale efficiently in order to meet variable demands (Foster and Kesselman, 2004). Security is considered to be the most significant challenge for Grid computing (Foster et al., 1998; Humphrey et al., 2005; Nagaratnam et al., 2003; Ramakrishnan, 2004; Welch et al., 2003; Zhou et al., 2005), since the resources that are shared between organizations are expensive and may range from computers and other hardware facilities, to potentially valuable, sensitive and confidential data files.

In recent years the mobile computing community has been successful in utilizing academic and industrial research efforts to bring products to the commercial market. We have seen a proliferation of consumer electronic devices that take advantage of wireless technology and enrich our daily lives with increased productivity thanks to higher connectivity. Mobile computing imposes a degree of complexity inherent to the environment (Giguhre, 2001), such as dynamic environments, mobility, computational resource limitations, latency and instabilities in data transfer, energy supply limitations, and input/output interface limitations. This degree of complexity signifies that security is more difficult to implement on a mobile platform owing to the limitations of resources in these devices (Bradford et al., 2007), and is even more critical owing to the open nature of wireless networks. Therefore, in a mobile computing environment, it is necessary to have a robust security and trust infrastructure (Talukder and Yavagal, 2006).

In the purview of Grid and mobile computing, Mobile Grid is a heir of the Grid, which addresses mobility issues, with the added elements of supporting mobile users and resources in a seamless, transparent, secure and efficient way (Guan et al., 2005; Jameel et al., 2005). It is able to organize underlying ad-hoc networks and offer a self-configuring Grid system of mobile resources (hosts and users) connected by wireless links and forming random and

* Corresponding author. Tel.: +34 926295300; fax: +34 926295354.
E-mail addresses: David.Grosado@uclm.es (D.G. Rosado), Eduardo.FdezMedina@uclm.es (E. Fernández-Medina), jlm@lcc.uma.es (J. López), Mario.Piattini@uclm.es (M. Piattini).

1084-8045/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jnca.2011.01.001

changeable topologies (Litke et al., 2004). Security in a Mobile Grid system should cover all aspects of security, both of Grid computing and of mobile computing, integrating both proposals and providing a security infrastructure for Mobile Grid systems. This infrastructure should be solid, comprehensive, reliable, scalable and interoperable, i.e., able to provide security solutions to the problems, weaknesses, attacks and vulnerabilities (of any magnitude) found in this kind of systems.

On the other hand, the growing need to develop secure systems, mainly as a result of the new vulnerabilities caused by the increase in complex applications such as those applications that are distributed in heterogeneous environments, demand the integration of security into software engineering (Deubler et al., 2004; Fernández-Medina et al., 2009; Jürjens, 2005; Mouratidis, 2004; Osis and Asnina, 2008; Siponen et al., 2007). This demand to integrate security engineering and software engineering has encouraged the scientific community to build robust information systems in which security is not improvised and incorporated once the system has already been completely built. The main reason for this is that security aspects have traditionally only been considered at the implementation stages, thus signifying that security solutions are not perfectly coupled with the design and the other requirements of the system (Artelsmair and Wagner, 2003; Mouratidis and Giorgini, 2006). Ignoring the matter of security is dangerous, since it may be difficult to correct the security of an application once it has been built. Security aspects cannot be inserted "blindly" into a system; they ought to be borne in mind throughout the whole development of a complete process. In those cases in which there is no systematic methodology, security aspects are often updated too late in the design process, or carried out separately from the functional design (Artelsmair and Wagner, 2003). If security is considered only at certain stages of the development process, it is more likely that security needs will come into conflict with the system's functional requirements. Taking security into consideration along with the functional requirements throughout the development stages helps us to limit the cases of conflict by identifying them very early in the system development, and finding ways to overcome them. This is the main reason why our approach incorporates security aspects and requirements from the first stages of the development without forgetting the other requirements which are captured by following the tasks and methods of the generic development processes based on UML as the Unified Process (UP) and its extension of security, secure UP, OPEN, or OpenUP, etc.

Therefore, our goal is to define a systematic development process for Grid systems that supports the participation of mobile nodes and incorporates security aspects into the entire software lifecycle. There are several reasons for defining and building this development systematic process for Grid systems: firstly, the lack of adequate development methods for this kind of systems, since the majority of existing Grid applications have been built without a systematic development process and are based on ad-hoc developments (Dail et al., 2004; Kolonay and Sobolewski, 2004), suggests the need for adapted development methodologies (Giorgini et al., 2007; Graham, 2006; Jacobson et al., 1999; Open Group, 2009). Secondly, since the resources in a Grid are expensive, dynamic, heterogeneous, geographically located and under the control of multiple administrative domains (Bhanwar and Bawa, 2008), and the tasks accomplished and the information exchanged are confidential and sensitive, the security of these systems is difficult to achieve. And thirdly, as a result of the appearance of a new technology in which security is fundamental, together with the advances that mobile computation has undergone in recent years, which have increased the difficulty of incorporating mobile devices into a Grid environment (Guan et al., 2005; Jameel et al., 2005; Kumar and Qureshi, 2008; Kwok-Yan et al., 2004; Sajjad et al., 2005).

Our idea is to elaborate a complete development process in which we define the activities and tasks of which the process is composed, a UML extension for use cases, a security services architecture, transformation rules between models, etc., to improve the quality and security of Mobile Grid computing based systems in the entire software lifecycle (Anderson, 2001; Baskerville, 1993). This development process analyzes and integrates all the requirements and security aspects related to this kind of systems (Foster and Kesselman, 2004; Nagaratnam et al., 2003; Vivas et al., 2007), which results in a secure, robust and scalable Mobile Grid system. A preliminary publication of the process has been presented in Rosado et al. (2008b) in which we describe our general approach, providing an informal presentation of the first steps in Rosado et al. (2008a) which consists of analyzing the security requirements of Mobile Grid systems directed by misuse cases and security use cases. Moreover, the approach has been applied in an actual case study in Rosado et al. (2009b) and Rosado et al. (2010b) from which we obtain the security requirements for a specific application by following the steps described in our process. We have then gone on to elicit some of the common requirements of these kinds of systems, and these have been specified to be reused through a UML extension of use cases (Rosado et al., 2009c, 2011). The first activity in this process was presented in Rosado et al. (2010a) in which we defined a systematic approach of the analysis activity using SPEM 2.0 (OMG, 2008) and in which we also defined the different models generated in the analysis which will be used as input artefacts in the design activity.

In this paper, we advance in our process by defining the complete design activity, indicating all the tasks carried out to integrate and model the newly defined artefacts which are focused on designing and building a security architecture based on both a previously defined reference security architecture and services. This architecture is stored in the process repository and covers the security requirements identified in the analysis activity. In the development of this process, we apply the action-research method (Estay and Pastor, 2000) in order to incrementally improve and refine our approach and the defined models and artefacts. Finally, we are currently applying this activity to a real case study (being developed in a European project) which will be presented in the last section of this paper.

The remainder of the paper is organized as follows: in Section 2 we present the related work. In Section 3 we briefly summarize the proposed process, showing all the activities of this development process and presenting the developed prototype tool. In Section 4, we present the main artefacts used in the design activity which have been specifically defined for this activity. In Section 5, we propose the design activity, all of whose tasks are defined by using SPEM 2.0. In Section 6, we apply the design activity to a real case and we describe the learned lessons. Finally, in Section 7, we put forward our conclusions and some research lines for our future work.

2. Related work

Grids appear to provide a promising approach for the future of massive computing, but the greatest obstacle to the widespread adoption of Grids is security, owing to the fact that resources are dynamic, heterogeneous, geographically located and under the control of multiple administrative domains. Moreover, the complexity of these systems is increased with the addition of mobile devices and wireless networks. Mobile Grids require comprehensive security solutions that address the increasingly complex

systems and that cover the security requirements and needs arising from these systems. There are numerous approaches related to Grid computing and Mobile Grid (architectures, middleware, infrastructures, projects, etc.) but here we present some of those that we believe to be most interesting and that consider security as an important factor for success and application in Mobile Grid environments.

This set of approaches is divided into three topics, which are the main topics related to our approach: (1) security approaches in development methodologies; (2) security architectures for Grid environments and (3) the incorporation of mobile devices into the Grid. There are numerous approaches with regard to these topics but here we have described some of those that we believe to be most important and most closely related to our approach.

With regard to those proposals related to secure development processes, there is the Secure Unified Process (Steel et al., 2005) which incorporates security disciplines into the Unified Process, the de facto standard for the software application development process, but this is a very general approach that has to be adapted for each specific application that we wish to develop. The specific aspects of Mobile Grid systems necessitate the definition of new activities, artefacts, roles, techniques and security disciplines which are not considered in Secure UP. Secure Tropos (Giorgini et al., 2007) is an agent oriented methodology with security aspects in which the authors define tools and practice cases to assist in the application of this methodology to the development of a software system. There are many security aspects that cannot be captured as a result of the dynamic behavior and mobile considerations of Mobile Grid systems. UMLsec (Jürjens, 2005) is an extremely interesting approach which incorporates security properties into the UML model. UMLsec has been applied in security-critical systems and in the industrial context of a mobile communication system (Jürjens et al., 2008), and the security aspects of this kind of systems has been analyzed, but it has not been applied in Grid environments with specific security aspects. Our methodological approach therefore considers, at the analysis level, extended use case models with which to specify security requirements for mobile Grid systems, and this use case view can be complimented with other UML diagrams (deployment, activity, classes, collaboration, etc.) using UMLsec to model certain security aspects (generic and mobile) in these diagrams. Model-Driven Security (MDS) (Basin et al., 2003) was conceived as a new approach towards building secure information systems, in which designers specify high-level system models along with their security properties and use tools in order to automatically generate system architectures from the models, including security infrastructures. Finally, AEGIS (Flechais et al., 2003) is the only approach found in which the authors attempt to apply the methodology to Grid systems, although they do not explain how to do this, and do not define guidelines and practices with which to capture specific security aspects in Grid systems. This approach should be adapted to the necessities and features of Grid systems. More details of proposals concerning secure development processes can be found in the editorial (Fernández-Medina et al., 2009) and in the papers published in that special issue.

There are various proposals related to security architectures for Grid environments. OGSA (Open Grid Forum, 2006) represents an evolution towards a Grid system architecture based on Web Service concepts and technologies. The lack of support for mobile devices, including security aspects, makes this incomplete for Mobile Grid systems. GSI (Globus Project, 2005) is an essential middleware component that has been integrated into many tools and offers solutions for Grid systems. These solutions are security solutions for Grid environments in which mobile devices are not considered, and this proposal does not offer solutions for the risk and possible attacks that appear in mobile computing. EGEE Security (EGEE Middleware Design Team, 2004) is a middleware architecture which supports security services for Grid environments. The creators of this architecture define practice cases and tools which assist in the construction of the middleware and the building of applications based on Grid computing, and which use Web service standards. However, as with their predecessors, security aspects for mobile environments are not considered. EGA Security (Enterprise Grid Alliance Security Working Group, 2005) defines a Grid management entity which manages a set of security functions and policies for Enterprise Grid to establish a secure connection with the Grid components that participate in the system. This approach does not consider mobile components in the enterprise Grid and consequently does not consider mobile security aspects. Legion Security (Chapin et al., 1999) defines a set of security mechanisms and policies to enable participants in a Grid system to expose their resources in a manner which is compliant with their local policies. This approach does not consider the security of mobile participants with mobile devices and a wireless network. Globe Security (van Steen et al., 1999) constitutes a middleware level for a wide area distributed system, signifying that the security considerations presented in this approach for Grid systems coincide with the security considerations for distributed systems, but are not exclusive to Grid environments, and mobile computing aspects are not considered. Finally, CRISIS Security (Belani et al., 1998) was developed to support wide area distributed applications and to define a wide set of security features which are present in many Grid systems, although they are not specific to this kind of systems. As with Globe, the mobile security aspects are not considered.

With regard to proposals related to the incorporation of mobile devices into the Grid, we find Leech (Phan et al., 2005) which is a proxy-based clustered infrastructure and creates groups of devices clustered around a nearby proxy. This considers security aspects for wireless network and mobile devices through the proxy which serves as an intermediary between mobile devices and the Grid. This proxy must additionally be protected to safeguard Grid systems with mobile devices. Mobile-To-Grid (Jameel et al., 2005; Kalim et al., 2005) defines a middleware which permits heterogeneous mobile devices access to Grid services. This middleware contains security services, based on GSI, that permit the secure communication between the mobile user, the middleware and the Grid. This approach treats mobile devices like external elements, and security must be implemented outside the Grid environment. Mobile OGSI.net (Chu and Humphrey, 2004) extends an implementation of Grid computing, OGSI.NET, to mobile devices. It is implemented in the Microsoft PocketPC 2003 operating system and uses the security services defined in the operating system itself and in OGSI.NET. This approach does not specify security aspects for these environments, and only defines connection modules between Grid services and mobile services. Grid-M (Franke et al., 2007) is a platform for building Grid Computing applications in embedded and mobile computing devices. It defines new security functionalities for mobile devices and uses an API to connect applications in Grid environments and mobile devices; however, the security of this API is not within the Grid security environment and may cause risks and vulnerabilities in the system.

Finally, after studying and analyzing each of the approaches related to the development and security in Mobile Grid computing, we have concluded that the existing proposals are not sufficiently specific to provide a complete solution to the question of security under a systematic development process for Mobile Grid environments. This is owing to the fact that none of the approaches defines a systematic development process for this specific kind of systems that incorporates security from the earliest stages of the development. Moreover, the existing

security approaches to Mobile Grid computing are more focused on offering a technological solution than on the requirements and design. Neither of them offers solutions that integrate mobile devices as the resources of the Grid themselves in a global security environment.

The approaches studied in the related work therefore suggest that more effort is needed in the incorporation of security into Mobile Grid systems. Specific solutions should be offered and security features for this kind of systems should be considered from the first stages of the development. We propose a development process which offers developers a set of guides, techniques, methods, architectures, services and so on, to establish security aspects in the existing systems or to create new secure systems. It would thus be possible to build secure Mobile Grid systems in a systematic and adaptable manner, having previously carried out a security analysis and integrated the security features into the different approaches (some of which we have shown here) using the methods, techniques or mechanisms defined in our systematic process, and assuring that the resulting Mobile Grid system fulfilled the necessary security requirements and needs.

3. Overview of the development process

The Secure Mobile Grid development process (SecMobGrid) was designed for the construction of software systems based on Mobile Grid computing with security aspects. It is a process which builds a secure software product from the initial requirements and needs of Mobile Grid systems. This process does not only include security in a development process, but is a development process in itself and incorporates security aspects throughout the entire process.

The structure of the process is composed of a planning phase, a development phase including analysis, design and construction, and a maintenance phase. However, it is specially designed for this kind of systems and considers their particular features. Figure 1 shows the definition of the process using SPEM version 2.0 (OMG, 2008).

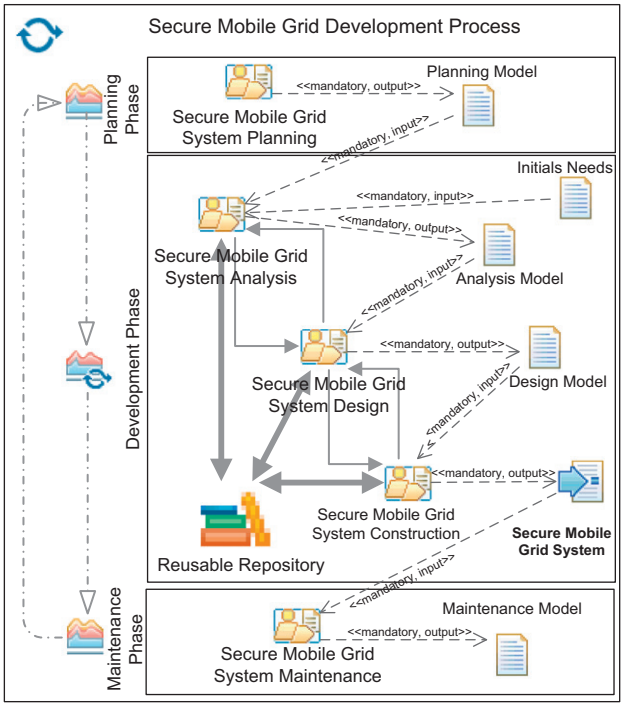

Fig. 1. Development process for secure Mobile Grid systems with SPEM 2.0.

The process is characterized by the fact that it is an iterative and incremental process, that it defines traceability and reusability of the process and product reusability, that it integrates traditional software development techniques and methods, its use of concepts, methods and techniques developed in the field of security software as misuse cases or UMLsec, and that it is supported by a reference security architecture and is developed with current Grid tools and platforms. These characteristics are present in all the development process activities, with emphasis on the analysis, design and construction activities in which we define specific artefacts and tasks for the Mobile Grid systems.

Therefore, in the "*Secure Mobile Grid System Analysis*" activity (Rosado et al., 2010a) we analyze and specify the different requirements, specifically the security requirements that can be found in this kind of systems. This is done by defining a UML profile to capture the security requirements from use case diagrams in which new stereotypes for use cases, actors and associations are defined to capture the security behavior of Mobile Grid systems. The "*Secure Mobile Grid System Design*" activity subsequently focuses on the definition of a secure software architecture which is composed of a software architecture and a security architecture in which the structural elements and security elements of which the system is composed, and the behavior and interfaces between, them are defined and integrated. The security architecture is supported by reference security architecture that can be reused and redefined in particular developments. Finally, the "*Secure Mobile Grid System Construction*" activity is focused on the implementation of the secure software architecture and the other artefacts used in the process, considering the Grid technological platform that will be used. It is possible that the technological environment may have to be expanded to deal with Mobile Grid systems through the addition of new security functions, protocols and mechanisms.

The model chosen for our process is iterative and incremental, and therefore facilitates the development and gradual integration of security into the systems based on Mobile Grid computing. It also allows reusability through the use of artefacts and models stored in the repository that manages the process. This process shows how to capture functional and security requirements for Mobile Grid systems with reusable use cases, security use cases and misuse cases, aided by a UML profile defined specifically for the analysis activity. The process defines a reference security architecture that ensures the fulfilment of security requirements and provides guidelines and steps through which to obtain, from the design, an implementation of the system based on Grid technologies.

The Software Engineering community is beginning to realize that security is an important requirement for software systems, and that it should be considered from the first stages of their development (Alam et al., 2007; Fernández-Medina et al., 2009; Fernández et al., 2007; Gutiérrez et al., 2005; Jürjens, 2005; Mouratidis and Giorgini, 2007; Steel et al., 2005). If security is considered early in the development and brought into each of the software engineering activities from the first stages, we achieve more robust and efficient and less improvised solutions, and this helps to reduce conflicts between security requirements and other requirements (Kim, 2005; Mouratidis and Giorgini, 2007). It is for this reason that we propose a development process for secure Mobile Grid systems in which security is incorporated from the beginning of the development to assure that the final system is robust and efficient and that it incorporates security aspects into the entire systematic process.

One of the problems faced in everyday practice is the desire to secure an application without spending excessive time and effort; we therefore use certain known solutions, such as security patterns, combined experience and good practices in the design

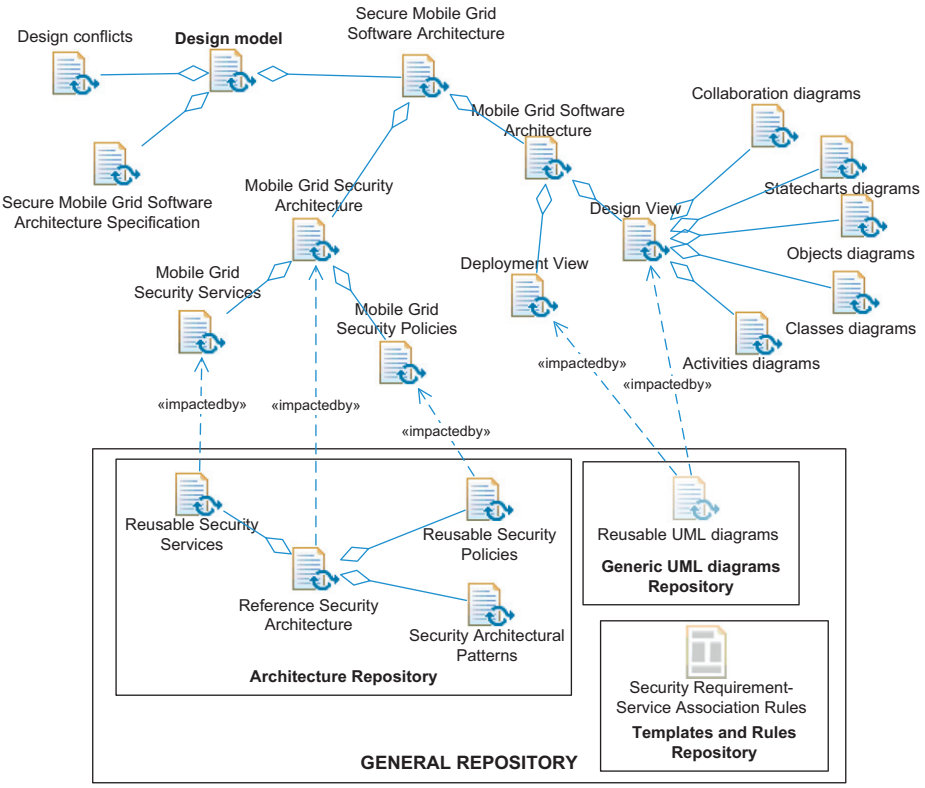

Fig. 2. Artefacts of the analysis activity.

of Systems Information (Fernandez, 2007), thus making it more efficient, precise and of high quality (Fernandez et al., 2009; Maña et al., 2009). We have therefore defined an architectural pattern, which is the reference security architecture, to obtain reliable and efficient solutions related to the development of Mobile Grid systems. This architecture defines a wide set of security services and elements in an attempt to fulfill the majority of the security requirements of any Mobile Grid system. The security architecture therefore saves the developers time and effort, and is a good mechanism with which to optimize the decision process when solving a recurring security problem in these systems.

One possible improvement is to follow Model-Driven engineering (MDE) (Mens and Van Gorp, 2006) and Model-Driven Security (MDS) (Basin et al., 2006) approaches, and we have therefore defined models and artefacts in each of the stages, which are the principal elements for their development, maintenance and evolution, through the implementation of transformations of models (Mens and Van Gorp, 2006). These strategies help to obtain applications in a more efficient manner with regard to time, effort and cost, and to obtain more robust and precise solutions. This strategy allows us to improve and reinforce the integration of security aspects in the entire development cycle and to facilitate the future transformations between different models in the process.

Finally, one of the main features of our approach is the reuse of elements, artefacts and models defined and developed in the process which is a technique by which to save time and effort in the development. The reused elements are improved and tested in each development, thus making the final system more robust and reliable. Moreover, we have developed an automatic prototype tool (called SMGridTool) to assist in the management and construction of models, thus improving the process and making it more efficient.

All of the aforementioned features and particularities of the process that are proposed herein, such as the incorporation of security aspects from the first stages of the software lifecycle, the use of a reference security architecture as a pattern, the adaptation to MDE or MDS and the reuse of elements, therefore help to improve the precision, thus making both the process and the final product efficient and robust.

4. Main artefacts of the design activity

The main artefact generated in this activity is the design model which is composed of other artefacts produced and used in the different tasks of this activity (see Fig. 2). We have the *Secure Mobile Grid Software Architecture* artefact which is formed of the security architecture and software architecture for Mobile Grid environments, the *Secure Software Architecture Specification* which describes and documents the architecture design, and the artefact of *Design conflicts* that identifies the possible conflicts found during this activity. Each of the artefacts of the design model will be described in greater depth below.

The design model is principally composed of the "*Secure Mobile Grid Software Architecture*" artefact which represents the final software architecture of the system integrating the security aspects considered in the analysis activity. The final architecture shows both the software architecture of the system ("*Mobile Grid Software Architecture*" artefact defined by different views and UML diagrams) and security architecture ("*Mobile Grid Security Architecture*" artefact defining security services and policies) built from a reference security architecture ("*Reference Security Architecture*" artefact) which is available in the repository of our process. The software architecture is defined by following the typical methods and tools for the design of systems such as the Unified Process (Jacobson et al., 1999), OPEN,[1] OpenUP,[2] etc., using different

[1] http://www.opfro.org/.
[2] http://epf.eclipse.org/wikis/openup/.

views and UML diagrams. Another artefact of the design model is "*Secure Mobile Grid Software Architecture Specification*" which formally specifies and documents the final architecture built in this activity by following the IEEE 1471-2000 standard (IEEE, 2000) for the architectural description of software-intensive systems, or by following an Architecture Description Language (ADL). Finally, the "D*esign conflicts*" artefact defines the changes and errors found during this activity that should be taken into account in the future iterations of this activity in order to refine aspects, elements, etc., thus improving the design model and, therefore, the system design.

The most significant and innovative artefact of the design model is "*Mobile Grid Security Architecture*" which is based on a reference security architecture built in the repository in which generic security services and policies are defined.

4.1. Reference security architecture

One of the principal and most important artefacts of the design activity is the reference security architecture (Rosado et al., in press), a service-oriented security architecture in which we have defined a complete set of security services and interfaces that cover the majority of security requirements that may appear in any Mobile Grid environment and which should be specified in the analysis activity of the development process. This security architecture protects the Grid system and offers the necessary support to ensure that the security requirements and needs are fulfilled. This reference security architecture defines a wide set of security services and for each Grid application to be developed and only a subset of these, which will cover the specified security requirements for that application, will be necessary. The services and interfaces of the security architecture are defined in an abstract manner for their use in the design activity, but should be implemented in the construction activity.

The security architecture has been defined in 5 levels from the most basic to the most advanced levels, and all the security services can be related to each other, independently of the level to which they belong. Level 1 contains confidentiality, integrity, authentication, authorization and non-repudiation, which are basic and priority services. Level 2 contains the delegation and anonymity services which are composed of the lowest level services. Level 3 shows the privacy and Trust Management services which are higher level services and must use the lower level services to carry out their functions. Level 4 contains the Credential and Identity Management and Mobile Policy services which are services that collaborate with and relate to the other services in the architecture. These are not priorities but are important in these systems. Finally, the highest level services are audit and Grid Security Policy which are closer to the application level, and the Grid Security Policy service is related to all services in the architecture and is therefore located in the highest level of the architecture. More details of the levels, relationships between services and of the architecture can be found in (Rosado et al., in press).

The set of security services, which we have briefly defined in Appendix 1, together with the interfaces and operations which are involved in the security architecture can be seen in Table 1. The interfaces of the security services are defined in a specific and sufficiently generic manner in order to focus on and identify the objective that we wish to attain, and do not depend on any technological platform.

4.2. Security requirement–security service association rules

The security requirement–security service association rules indicate the security services that must be defined in the security architecture depending on the security requirements captured and specified in the analysis activity through use cases. Thus, once the security requirements of the application have been identified and defined (as tagged values) in the Grid use cases, in the design activity we can continue selecting the security services related to the security requirements identified and identifying a set of security services that define the security architecture for the application studied.

We consider the greatest set of security requirements that it is possible to find in Mobile Grid systems gathered from different works (Bellavista and Corradi, 2006; Foster and Kesselman, 2004; Nagaratnam et al., 2003; Open Grid Forum, 2006; Vivas et al., 2007) and which we have briefly defined in Appendix 2. These requirements are indicated in the use case diagrams generated in the analysis activity of the SecMobGrid process following the GridUCSec-profile (Rosado et al., 2009a) in which a set of tagged values is defined for each use case that is related to security requirements.

Table 1
Security services and operations for each interface.

Services	Operations
Integrity	Data shield(parameters); Boolean validate(data); Data unshield(parameters);
Confidentiality	Data hide(parameters); Data reveal(parameters)
Authorization	collectAttributes(parameters); Boolean isPermitted(parameters); Policy getPolicy(identifier)
Authentication	Credential authenticate(credential); Boolean isAuthenticated(context)
Non-repudiation	generateEvidence(parameters); Boolean validateEvidence(parameters); retrieveEvidence(parameters)
Delegation	Boolean delegate(parameters); Boolean revocation(credential); Boolean restriction(parameters)
Anonymity	Boolean setAnonymity(subject, context); Boolean checkAnonymity(subject)
Trust Management	Boolean createTrust(parameters); negotiateTrust(parameters); manageTrust(trustcontext); Boolean revokeTrust(entity, trustcontext)
Privacy	Boolean setPrivacy(context); Boolean checkPrivacy(context)
Credential Management	Credential issueCredential(parameters); Boolean storeCredential(credential); Credential getCredential(parameters); Boolean revokeCredential(credential); Boolean renewalCredential(credential, time); Credential translateCredential(credential, formatO, formatD)
Identity Management	Identity getIdentity(credential); Boolean updateIdentity(identity, data); Boolean checkIdentity(identity); Boolean revokeIdentity(identity); Identity translateIdentity(identity, formatO, formatD)
Mobile Policy	Policy getMobilePolicy(soa, oid, localurl); Boolean updateMobilePolicy(policy, data); Boolean setMobilePolicy(policy, mobServices)
Audit	Boolean recordEvents(parameters); Event retrieveEvents(parameters)
Grid Security Policy	Policy getSecurityPolicy(soa, oid, url); Boolean setSecurityPolicy(policy, services) Boolean updateSecurityPolicy(policy, data)

Table 2 shows the relationships between some of these security requirements and the security services belonging to the reference security architecture.

Those security requirements defined for Mobile Grid systems that have not been considered in this table, such as exportability, firewall transversal, integration, mobility, multiple implementation, scalability and self-organization, are implicitly defined by designing and implementing the security architecture, the services, the interfaces, the mechanisms, etc. in the various tasks and activities of the SecMobGrid process.

In addition to the security services identified in Table 2, it is also necessary to add those services that are related to the security services identified to the security architecture that we are defining, because there may be calls to operations of the interfaces of the services to which they are related. It is therefore necessary to define the actual set of security services of the reference security architecture that is needed to fulfill the security requirements.

5. Design activity

The design activity is composed of tasks which build the software architecture, the security architecture and their specifications with different views, and use architectural elements from the repository to obtain the design model in which the architecture is defined. This activity produces internal artefacts which are the output of some tasks and the input of others. All these internal artefacts are included in the design model to be used in the following activities if necessary. Figure 3 shows a graphical view of the design activity tasks using SPEM 2.0 diagrams.

The "*Designing Mobile Grid Software Architecture*" task initially designs a software architecture by following the steps, methods and techniques of the typical development processes (such as the Unified Process, OPEN, OpenUP, etc.) using UML diagrams, views and realizations of use cases from the use case model and analysis model defined in the analysis activity for Mobile Grid systems. The "*Designing Mobile Grid Security Architecture*" task defines the

Table 2
Association rules between security requirements and security services.

Security requirements	Security services
Accounting	Audit, non-repudiation
Anonymity	Anonymity
Assurance	Audit
Authentication and MutualAuthentication	Authentication, Credential Management, Identity Management
Authorization&AC	Authorization, Authentication, Credential Management, Identity Credential, Trust Management
Confidentiality	Confidentiality, Integrity
Credential	Credential Management, Identity Management
Delegation	Delegation, Trust Management, Authentication, Authorization, non-repudiation
Freshness	Authentication, Authorization, Delegation, Credential Management
Integrity	Integrity
Interoperability	Credential Management, Identity Management, Trust Management
Manageability	Credential Management, Identity Management, Trust Management
MappingIdentity	Identity Management
Non-repudiation	Non-repudiation, Integrity
Policy Exchange	Grid Security Policy, Mobile Policy
Privacy	Privacy, Confidentiality, Anonymity, Authentication, Authorization
Revocation	Identity Management, Credential Management, Delegation, Trust Management
Secure group communication	Authentication, Credential Management, Identity Management
Secure logging	Privacy, Confidentiality, Integrity
Single sign-on (SSO)	Delegation, Authorization, Authentication, Trust Management
Trust	Trust Management, Authentication, Authorization, Delgation
Uniform Credential	Credential Management

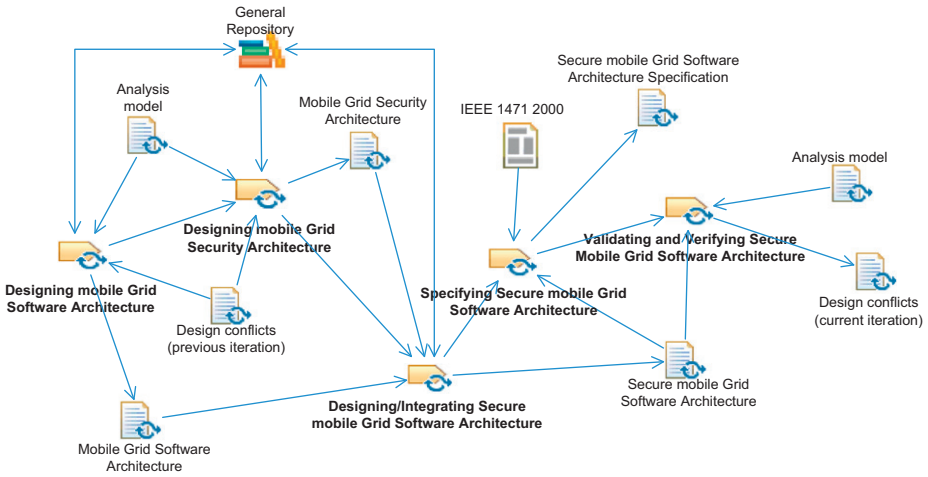

Fig. 3. Tasks of the secure Mobile Grid system design activity.

security architecture by using the reusable elements of the repository and following the security requirement–service association rules of use cases to security services from the use cases of the analysis model. The reusable elements are generic elements which should be instantiated and integrated into the security architecture. Once both the software architecture and the security architecture have been built, it is then necessary to build and define the final architecture which is the security architecture that is integrated into the software architecture to define the relationships between elements (classes, collaboration, sequence, objects, diagrams, etc.), thus obtaining the secure Mobile Grid software architecture. This is executed in the "*Designing/Integrating Secure Mobile Grid Software Architecture*" task. In the "*Specifying Secure Mobile Grid Software Architecture*" task, the final architecture built is then specified by using natural language or the IEEE 1471-2000standard. Finally, in the "*Validating and Verifying Secure Mobile Grid Software Architecture*" task, the architecture obtained in the previous task has to be validated with the requirements specified in the analysis activity, the traceability between artefacts has to be verified, and the possible conflicts or errors in the design have to be identified and analyzed for their subsequent refinement in the next iterations of this activity.

We shall now briefly define each task in this activity, indicating the steps that should be followed to successfully execute these tasks, either with the help of well-known development processes for the common development tasks, or by defining the new techniques and steps that make our process specific to Mobile Grid environments.

5.1. Task D1: designing Mobile Grid software architecture

The input artefact is the analysis model defined in the previous activity, with the identification and analysis of use cases of this kind of systems, which guide the design of the architecture. Use cases can be used to design the software architecture using the techniques and methods of any suitable development process based on UML in which it is possible to make transformations from use cases to classes and objects that describe the design view. This design view should not only be supported by class and object diagrams, but also by activities, statecharts, collaboration diagrams, and other diagrams for the design view used in a typical development process. Generic UML diagrams related to reusable use cases are defined in the repository, and can be used in the different views to complete the design of the software architecture.

As a result of this task, we will obtain the Mobile Grid software architecture that does not consider security aspects. The roles that will take part in this task are: Requirements Engineer, Designer, Mobile Grid Specialist and System Architect. With regard to the techniques and practices for the realization of this task, we can use UML and UMLSec, and methods and guides from the Unified Process.

5.2. Task D2: designing Mobile Grid security architecture

This task carries out the integration of security into the system through a security architecture built from the security use cases identified and specified in the analysis activity. We have defined security requirement–service association rules in which the security use cases, which represent security requirements, can be translated into security services. These security services are integrated into the *reference security architecture* as instances of the abstract security services in order to obtain the final security architecture with all the specific security services defined with the security requirement–service relation rules from the security use cases. It is important to define the security policies of the security architecture (for each security service, virtual organization, communications, resources, mobile resources, user, etc.); some of these are defined as abstract security policies in the repository and should be instantiated as security policies in the final security architecture. Finally, once we have defined the security architecture from the reference security architecture and security policies from the repository, it is necessary to design the final security architecture with UML diagrams, relationships between elements, protocols, the standards used and so on.

Figure 4 shows the steps in this task using SPEM 2.0 diagrams.

As a result of this task, we will obtain the following artefacts: Mobile Grid security architecture, which is the services oriented architecture instantiated from the reference security architecture previously defined. The roles that will take part in this task are: Security Designer, Security Requirements Engineer, Mobile Grid Specialist and Security architect.

With regard to the techniques and practices for the realization of this task, we can use the reusable elements defined in the architecture repository such as the Reference Security Architecture, Reusable Security Policies and Security Requirement–Service Association Rules.

5.3. Task D3: designing/integrating secure Mobile Grid software architecture

The Mobile Grid software architecture and Mobile Grid security architecture have been designed in the previous tasks, and are the input artefacts for this task which carries out the integration of the two architectures (software and security) as a secure software architecture for Mobile Grid environments. The architecture designed should define relationships between software and security elements in all the views (of design and deployment) in which the relations and interactions between software elements and security elements are shown, thus obtaining the secure software architecture. The views model can be defined by using the reusable UML diagrams previously defined in applications with similar features. It is also necessary to define global policies of the application, defining the rules and the behavior that the organizations, institutions, domains and resources should

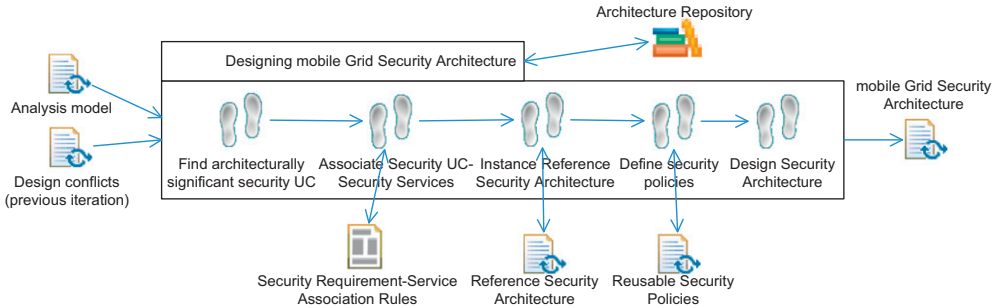

Fig. 4. Task D2: designing Mobile Grid Security architecture.

fulfill and carry out for the exchange of information, messages, data, and so on, around the Grid.

Figure 5 shows the steps in this task using SPEM 2.0 diagrams.

As a result of this task, we will obtain the following artefacts: secure Mobile Grid software architecture which is the integration of the software architecture and the security architecture designed in previous tasks. The roles that will take part in this task are: Designer, Security Architect, Mobile Grid Specialist and System Architect.

With regard to the techniques and practices for the realization of this task, the UML and UMLSec can be used for the specification of diagrams and models, and previously built UML diagrams can be reused.

5.4. Task D4: specifying secure Mobile Grid software architecture

This task is responsible for documenting the secure software architecture, principally through the use of a template in accordance with the standard for documenting software architecture views using IEEE 1471-2000based on a set of views. It is necessary to define a template of security viewpoints in order to document the specific security aspects of the architecture. Figure 6 shows the steps in this task using SPEM 2.0 diagrams.

As a result of this task, we will obtain the following artefacts: specification of secure Mobile Grid software architecture which is the specification of the secure software architecture designed in this activity. The roles that will take part in this task are: Security Architect, Mobile Grid Specialist and System Architect.

With regard to the techniques and practices for the realization of this task, the IEEE 1471-2000 specification and the security viewpoints specification defined for this purpose can be used.

5.5. Task D5: validating and verifying the secure Mobile Grid software architecture

This task validates whether the designed software architecture covers, fulfils and considers all the requirements (contained in the analysis model artefact) specified in the analysis activity. It also validates that the security and attack scenarios considered in the current iteration are solved and identifies potential conflicts in the solution (designed architecture) with regard to functional or quality requirements. Moreover, it verifies the traceability of

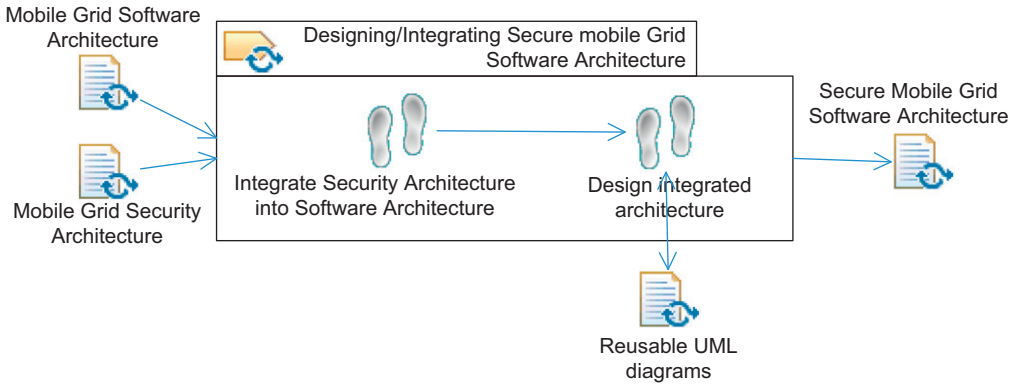

Fig. 5. Task D3: Designing/integrating secure Mobile Grid software architecture.

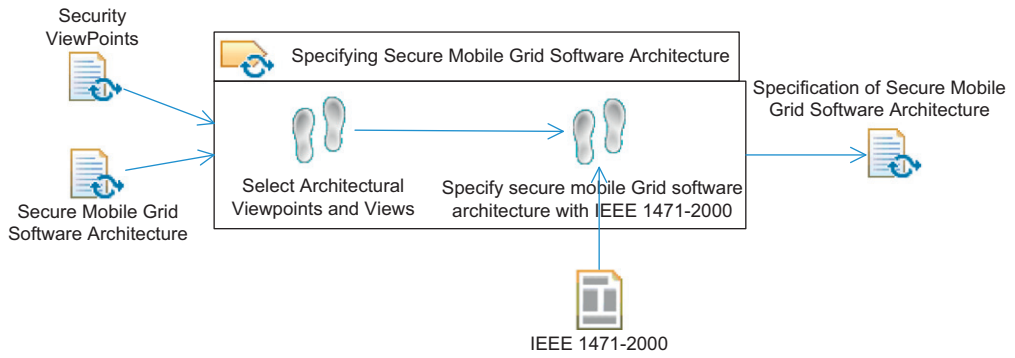

Fig. 6. Task D4: Specifying secure Mobile Grid software architecture.

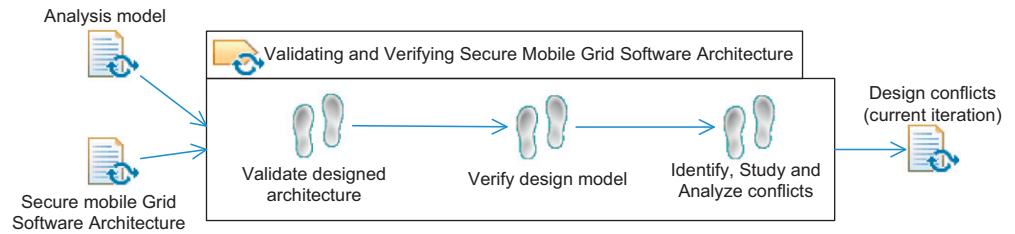

Fig. 7. Task D5: Validating and verifying the secure Mobile Grid software architecture.

artefacts by studying whether the analysis artefacts have been considered and correctly generated in the design artefacts. It also verifies that the remaining scenarios are fulfilled with the architecture design by identifying possible conflicts, studying their scope and analyzing the opportunities to correct and deal with them.

Figure 7 shows the steps in this task using SPEM 2.0 diagrams.

As a result of this task, we will obtain the following artefacts: design conflicts of the current iteration found in the realization of this activity. The roles that will take part in this task are: Security Requirements Engineer, Requirements Engineer, Security Architect, Mobile Grid Specialist and System Architect.

With regard to the techniques and practices for the realization of this task, meetings can take place between those involved in the development.

6. Case study

GREDIA[3] is a system which aims to enable commercial users (such as those represented by banking and media application pilots) to manipulate data and use services in a Grid computing environment, thus leveraging the potential of computing Grids for business purposes along with providing nontrivial business functionality solutions for end users in a controlled, secure environment. GREDIA will work on the specifications of a Security Framework to provide protection for data and transactions at all levels through a dedicated security framework that will be specifically developed for Grid based applications. The framework will address security issues in Grid P2P architecture, such as the authentication of entities, confidentiality and integrity to enable the secure accessing of rich multimedia content.

Our development process is being applied to one of the pilot applications: the media (news) sector. The process is assisting us in the construction of a Mobile Grid application, which will allow journalists and photographers (media domain actors) to make their work available to a trusted network of peers at the same moment as it is produced, either from desktop or mobile devices.

With the explosion of ultra portable photo/video capture media (i.e. based on mobile phones, PDAs or solid state camcorders) everyone can capture reasonably good quality audiovisual material while on the move. We wish to build a system that will cater for the reporter who is on the move with lightweight equipment and wishes to capture and transmit news content. This user needs to safely and quickly upload the media to a secure server to make it easier for others to access, and to avoid situations in which the device's battery dies or another malfunction destroys or makes his/her media unavailable.

This pilot application was developed by using an iterative and incremental approach, signifying that at the end of the iteration cycle, we have developed the final product. Therefore, in a first iteration, which is shown here for the design activity, we must select a part of the aims and goals of the system, of an acceptable size, so that we can apply the different activities and tasks to this part of the system and thus obtain reasonable artefacts, including a reduced version of the product. In the next iterations, which have been omitted here, this part of the system has been refined and extended with new elements and the process has again been applied to obtain a new version of the product. In the last iteration, the process has been applied to the whole system to obtain the final product.

The design activity begins with the artefact input which is the analysis model obtained in the previous activity (Rosado et al., 2010a). The aim of this first iteration in the analysis activity is to define a set of requirements of the system through the use case diagrams built with the help of the SMGridTool tool and the repository of reusable elements of the process. Once these requirements have been identified and specified, the process continues with the design activity.

We apply the design activity in this real case but consider only a reduced set of use cases (shown in Fig. 8) owing to space constraints. The tasks in the design activity are shown below.

6.1. Task D1: designing Mobile Grid software architecture

This task defines the software architecture of the system using traditional software engineering methods and mechanisms such as the Unified Process, OPEN, OpenUP, etc. The software architecture is designed from functional requirements and use cases of the application that have been analyzed by following the techniques and guides offered by certain development processes, but its input artefact is the analysis model elaborated in the previous activity of this process.

The output artefact is a software architecture oriented towards Mobile Grid systems (which has been omitted here).

6.2. Task D2: designing Mobile Grid security architecture

This task defines the security architecture in which all security aspects of the application are considered. The aim of this task is to design a security architecture whose services and components cover the security requirements and needs for this case study and can be integrated into the software architecture designed in the previous task.

The input artefact is the analysis model, with all use cases and requirements specified in the analysis activity.

6.2.1. Step D2.1: find architecturally significant security UC

In this first iteration of the process, we are working with a reduced set of security use cases for simplicity, so that the same security use cases identified in the analysis activity, extracted from the use cases sub-diagram built in the analysis activity of the development process (see Fig. 8) using a new UML profile (Rosado et al., 2009a), will be selected to develop the various tasks in the process design activity. The security use cases selected are: Authenticate, Authorize access, Ensure Confidentiality and Ensure Integrity.

Each of these security use cases are associated with some or many security services from the reference security architecture which will be identified in the next step.

6.2.2. Step D2.2: relate security UC–security services

By following the association rules defined between security requirements and security services in Section 4.2, we obtain that the security services to be considered in the security architecture of this application from the security use cases identified in the previous activity (shown in Table 3) are: Authorization, Authentication, Credential Management, Identity Management, Trust Management, Confidentiality and Integrity. We should not omit the Grid Security Policy service and the Mobile Policy service which are needed to manage the different policies of the system. These security services are obtained from the relationship with the security requirements which are defined as tagged values in the GridUCSec-profile.

6.2.3. Step D2.3: instantiate reference security architecture

Each security use case is associated with one or more security requirements, and the security services associated with these

[3] www.gredia.eu.

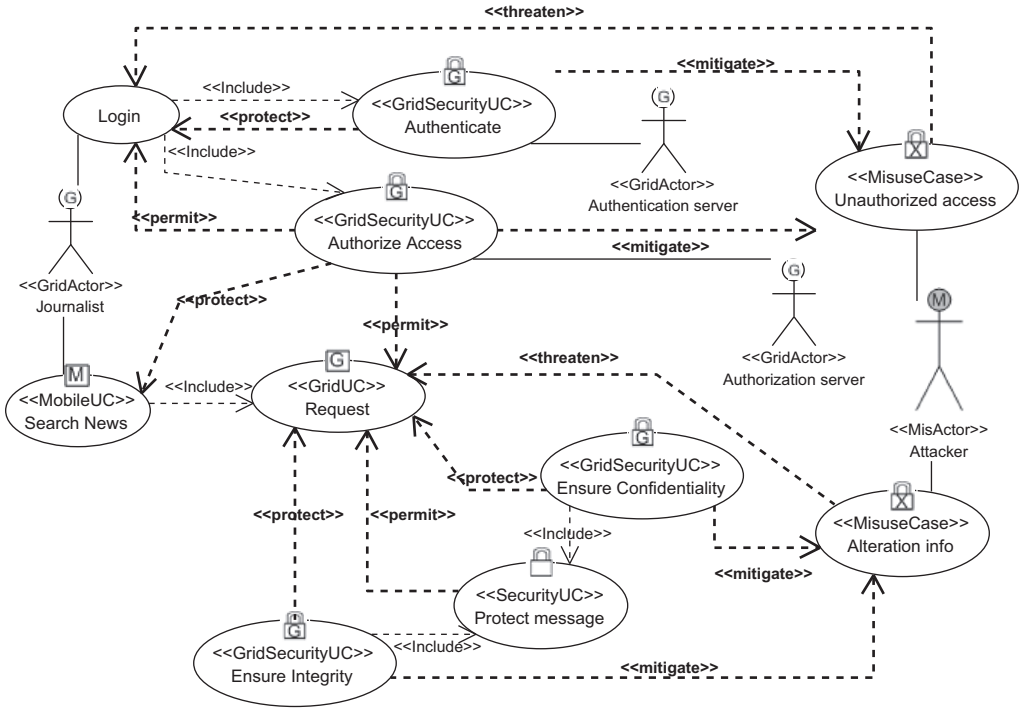

Fig. 8. Use case sub-diagram built in the analysis activity.

Table 3
Selected security services from security use cases for the case study.

Security UC	Security requirements	Security service
Authenticate	Authentication	Authentication, Credential Management, Identity Management, Trust Management and Integrity
Authorize access	Authorization	Authorization, Authentication, Credential Management, Identity Management, Trust Management and Integrity
Ensure Confidentiality	(Message) Confidentiality	Confidentiality and Integrity
Ensure Integrity	(Message) Integrity	Integrity

security use cases cover the security requirements represented by these security use cases. Therefore, the aim is to obtain a set of security services which cover and take into account the security requirements represented by the four security use cases. These security services are: Integrity, Confidentiality, Authorization, Authentication, Trust Management, Credential Management, Identity Management, Mobile Policy and Grid Security Policy Services.

Figure 9 shows the class diagram of the services and their interfaces, together with the relationships between them.

6.2.4. Step D2.4: define security policies

Security policies must be defined for the services identified in the previous step and they must indicate the security rules permitted for each service along with the protocols, mechanisms, attributes, etc., admitted by the service. The policies are stored in one or more LDAP repositories distributed throughout the Grid system but always available online.

There are security policies for two different domains, Gredia and NewsGredia domains, indicated in tagged values of GridUCSec-profile for Grid use cases. The Gredia domain defines security policies for the Grid elements and these policies are managed in the Grid Security Policy service, while the NewsGredia domain defines security policies for the mobile resources and users which initiate requests to the Grid, and these policies are managed in the Mobile Policy service.

Security policies must therefore be associated with each Grid security service of the instantiated architecture (for example, the authorization service security policy); we must also define security policies for messages exchanged between services, or inbound to the system (for example, each message must be protected from attacks against the integrity and confidentiality of the message body); and we must define security policies for inter-, intra-organizational partnerships (the sending of messages should be protected from attacks on privacy, integrity, confidentiality and the sender entity must be trusted).

Grid Security Policies govern and control the overall security of the Grid system and are associated with the security services, with inter-, intra- organizational partnerships, with the communication and exchange of information between resource users and Grid entities. Mobile Policies govern the use of mobile devices and wireless communications by establishing rules, standards, norms and advice on how resources must act in the mobile environment, protect their data, communicate via protocols, send data, etc.

The output artefact is a part of the security architecture designed from the set of use cases selected and with the help of the reference security architecture.

6.3. Task D3: designing/integrating secure Mobile Grid software architecture

This task cannot be applied to this case study because we have omitted that part of the software of the architecture and we

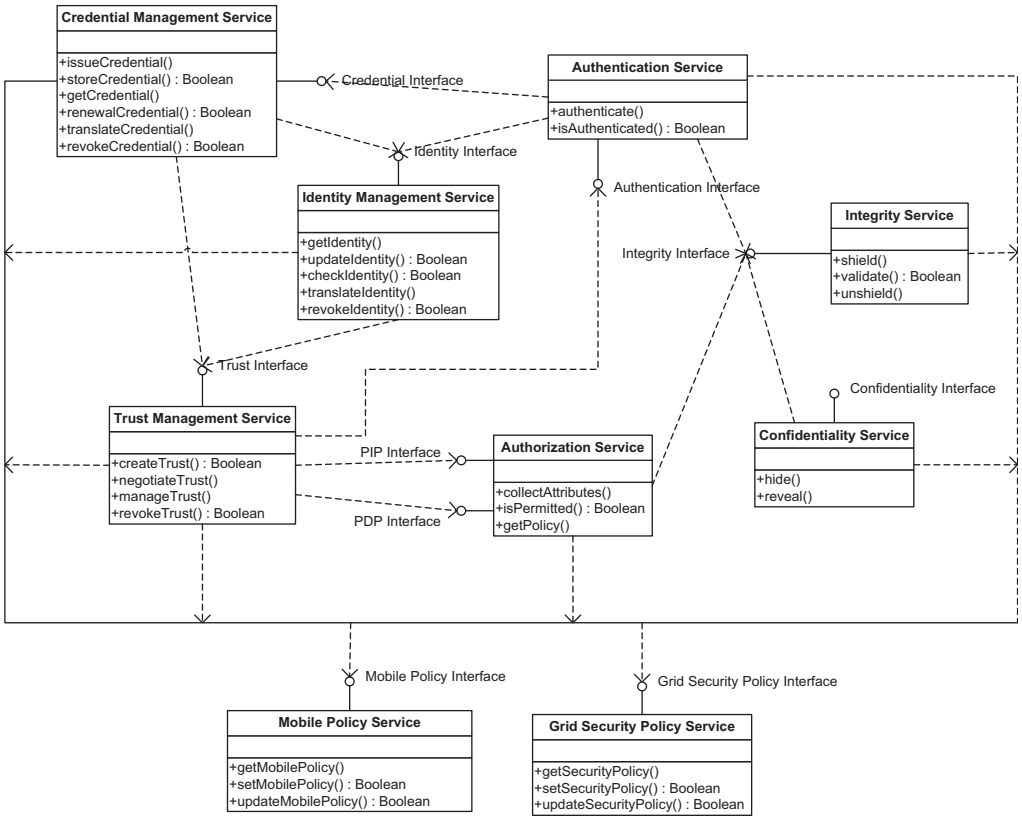

Fig. 9. Services and interfaces of security architecture instantiated for this case study.

cannot therefore show how the integration is carried out. Basically it is necessary to define how the various elements of the system, software architecture and security architecture, potentially constructed at different times, should be brought together into a single architecture: a secure software architecture.

Elements of the security architecture should be integrated into the software architecture since many software elements have to make use of the security elements, and the security elements have to act over the software elements to provide security in the system. It is therefore necessary to define the relations between these elements through the interfaces of the services, defining the protocols and mechanisms of exchange, policies, permissions and constraints.

Moreover, the static, dynamic and deployment views of all these elements of the secure software architecture are very useful to assist in understanding and obtaining a better knowledge of the architecture and the interactions between all its elements when implementation decisions are made in the construction activity. These views are defined by using traditional methods with the secure software architecture as the input artefact.

6.4. Task D4: specifying secure Mobile Grid software architecture

Systems Architect and Security Architect create an initial version of the secure software architecture document in which all the information outlined in the previous points is widespread. This document follows the basic points of the document, specifying the functional architecture consistent with IEEE 1471-2000, and adding new players, views, viewpoints and security view packages.

6.5. Task D5: validating/verifying secure Mobile Grid software architecture

This task validates and verifies the designed architecture. It is only possible to validate and verify the security architecture that is being developed in this case study. The analysis model with the specified use cases and requirements, and the architecture designed in this task, are the input artefacts used to validate and verify the architecture.

6.5.1. Step D5.1: validate designed architecture

By using the secure software architecture specification document as input (although in this case study we only have the security architecture, and the software architecture has not been shown), the process of validating the security architecture was carried out through review meetings with those roles involved in this task. This validation found that the security architecture responds to the needs of the stakeholders involved and, moreover, that it is integrated without making any impact on the software architecture designed.

6.5.2. Step D5.2: verify design model

We verified that the traceability between requirements and services designed through the association rules defined in Section 3 was successfully performed. The activity analysis artefacts are therefore needed as input to the design activity and are the basis for developing the design model artefacts.

6.5.3. Step D5.3: identify, study and analyze conflicts

We also observed that this architecture did not come into conflict with any other non-functional requirements and that all

the elements designed in this iteration were well integrated and there were no irregularities or inconsistencies.

6.6. Lessons learned

The development of this case study has enabled us to apply our process in a real environment and to verify, in practice, that the process permits the development of a secure Mobile Grid system from the initial needs and requirements.

The main lessons learned from the implementation of this development process in the aforementioned case study are:

- The application of this case study has allowed us to improve and refine several tasks of our process, essentially by adding new artefacts that were necessary or by removing others that were not used. A set of steps and the use of certain techniques, along with the repository of reusable elements, have also been updated and refined.
- A reusable and generic security architecture with a set of security services which cover the majority of the security requirements of Mobile Grid systems is fundamental to assist in the design of these systems and save time and effort in the design of a new security architecture for each system to be developed.
- Some of the services in the security architecture have been defined in order to capture the behavior of the mobile devices in these systems. This signifies, for example, that we have had to add the "Anonymity" and "Mobile Policy" services to cover the mobile requirements and particularities that incorporate the mobile devices in the Grid.
- It has been shown that the support of a tool is crucial for the practical application of this process, specifically in the analysis activity, owing to the large number of use cases managed and the numerous security relationships and attributes which have to be defined for a complete analysis of system requirements.
- We were also able to identify aspects which allowed us to make improvements to the prototype tool (SMGridTool), mainly regarding repository management since, as different applications are being developed, the number of use cases and diagrams stored in the repository grows. Moreover, we have realized that it would be interesting to extend the functionality of the SMGridTool to also support the design activity by carrying out the transformation of security use cases into security services belonging to the reference security architecture.
- The reuse of artefacts is an essential feature of the process since the majority of use cases to be used in the construction of the use case diagrams for such applications have been generated: (i) initially with the definition of many generic use cases, security use cases and misuse cases which represent the common behavior in most Mobile Grid systems, and (ii) in the various applications of our process, thus establishing numerous relationships between these common use cases which can be entirely reused.
- We have identified certain elements of our UML profile which are specific to this kind of systems, such as the "GridSecurityUC" use case that captures the security aspects found only in these systems and the "Permit" relationship that establishes security permissions between use cases to describe that a security use case contains other ones or part of other ones.
- We have described a wide set of tagged values in order to capture and store the major set of information related to Mobile Grid systems such as "DomainSecurity", "KindPermission" and "GridRequirement", "SecurityRequirement" and "MobileRequirement" to differentiate the specific features of these requirements that can be found in this kind of systems.
- A wide set of possible values to be assigned to each of the tagged values of our UML profile have been defined. These values have been obtained from the different elements (actors, relationships, functions, domains, requirements, assets, etc.) that have participated in the case study.
- The implementation of some of the services with Globus and PERMIS libraries, such as the Authorization service, has allowed us to define the code of these services in a generic and abstract manner, and to store them in the repository, to be used in any other application. We have also installed and configured libraries, configuration files and code for a Globus environment which can be used to implement other Grid applications.
- The repository is an elemental artefact in the process which contains a set of reusable elements, that originate in executions of the process in other Mobile Grid applications where common aspects are extracted and stored, or that were initially specifically built for this kind of systems and are available to be used by the different activities and tasks in the process.
- The reusability grants our process an efficient mechanism with which to develop a robust and secure Mobile Grid system with solutions that have been specially checked and built for these systems, thus saving time, effort and cost in its development.
- More case studies should be applied to obtain a more exhaustive validation and to check that our process is valid in other contexts and real life scenarios.

7. Conclusions

Grids and Mobile Grids may be the ideal solution for many large scale applications since they are of a dynamic nature and necessitate user transparency. The Grid will increase not only the job throughput and performance of the applications involved but also the utilization rate of resources by applying efficient resource management mechanisms to the vast amount of its resources. A Grid infrastructure that supports the participation of mobile nodes will thus play a significant role in the development of Grid computing. The idea of developing software through systematic development processes to improve software quality is not new. Nevertheless, there are still many information systems, such as those of Grid Computing, that are not developed through methodologies adapted to their most differentiating features.

In this paper, we have presented a systematic development for secure Mobile Grid environments and we have defined the design activity which is focused on the design of a secure software architecture that is the software architecture of the system with the incorporation of a specific security architecture for these environments. We have also defined the reference security architecture, which is a security service-oriented architecture specially constructed to accommodate all the security requirements and needs that can be found in Mobile Grid systems. This reference security architecture offers a design solution which is able to cover these security requirements and facilitate the design of the security architecture generated in the design activity of the process.

The development of the case study presented here has enabled us to apply the development process in a real environment, and to verify in practice that the process permits the development of a secure Mobile Grid system from the initial needs and requirements. Moreover, this case study has allowed us to improve many aspects of the process, such as for example, a complete definition of the security architecture by identifying operations of the

different interfaces and adding parameters for these operations that capture the main aspects specified in the analysis activity through the Grid use cases diagrams, which were not considered in earlier versions of the security architecture.

As future work, we intend to define the construction activity of this process from the secure software architecture built in the design activity by implementing each service (its interfaces and operations) on a specific implementation platform such as Globus and using different programming languages (JAVA, C#, .NET, etc.) to make the services compatible with the Grid systems currently implemented. We shall also apply new iterations in order to refine and improve the process, the artefacts and the models used and to assure that our process analyses, designs and implements a complete Mobile Grid system from the initial needs and requirements. Although the case study in which we have applied our process has proved to be very interesting, and has allowed us to improve the process by performing a first validation, our purpose is to continue improving the process in order to obtain feedback which can be applied in other real cases.

Acknowledgements

This research is part of the following projects: QUASIMODO (PAC08-0157-0668), SISTEMAS (PII2I09-0150-3135) and SEGMENT (HITO-09-138) financed by the "Viceconsejería de Ciencia y Tecnología de la Junta de Comunidades de Castilla-La Mancha" (Spain) and FEDER, and MEDUSAS (IDI-20090557), BUSINESS (PET2008-0136), PEGASO/MAGO (TIN2009-13718-C02-01) and ORIGIN (IDI-2010043(1-5)) financed by the "Ministerio de Ciencia e Innovación (CDTI)" (Spain). Special acknowledgment to GREDIA (FP6 34363—Grid enabled access to rich media content) funded by the European Commission.

Appendix 1

We present a brief description of each service of the security architecture that we have defined (more information can be found in Rosado et al., in press):

- Integrity: the Integrity service ensures that messages (data) and communications are not tampered with while in transit or in storage (in the device's memory, for example). This service protects the integrity of the data or messages exchanged, and validates and verifies the integrity of data or messages at any given moment.
- Confidentiality: the confidentiality service ensures that information is accessible only to those authorized to have access to it. This service protects data by using encryption techniques and by removing the protection with decryption techniques.
- Authorization: authorization refers to mechanisms that decide when a user is authorized to perform a certain task. This authorization service is based on the XACML authorization architecture which consists of several logical components. A Policy Enforcement Point (PEP) is first used to intercept access requests and transmits them to a Policy Decision Point (PDP) for the retrieval and evaluation of applicable policies.
- Authentication: the authentication service authenticates the user and obtains credentials for it, containing authenticated identity and privileges. This service extracts security attributes and information from the credentials or certificates in order to establish security associations and policies.
- Non-repudiation: the non-repudiation service provides generation of evidence of actions and later verification of this evidence, to prove that the action has occurred. There is often data associated with the action, so the service needs to provide evidence of both the data used and the type of action.
- Delegation: the delegation service makes the delegation of credentials between entities possible, along with the possible revocation of delegated credentials, and the ability to restrict, at any given time of the lifetime of the credential, initially delegated rights.
- Anonymity: this service provides anonymity for mobile users involved in the Grid system, protecting data and any information related to the user, thus avoiding its exposition and publication by external parties. This service establishes anonymity for a subject with regard to fulfilled actions, and verifies that the ownership of anonymity is preserved during the time that the user belongs to the system.
- Trust Management: this service establishes trust relationships between institutions, organizations and security domains that participate in the Grid. It therefore establishes trust between entities and negotiates a possible relationship of trust based on information and attributes of entities.
- Privacy: this service ensures and verifies that messages, data, personal information, attributes, roles, identity, credentials, etc. are not visible to parties other than the sender and authorized recipients. The anonymity of a user is a requisite for privacy, and these services must therefore be related.
- Credential Management: this service manages all those aspects related to the credentials that any entity belonging to the Grid must provide. This service issues credentials for Grid users, storages and renewals of expired credentials, translations between different formats and types of credentials in the Grid to be understood by all services and policies, and it revokes credentials.
- Identity Management: this service obtains the identity from the credentials submitted by users, updates the information related to an identity (roles, permissions, etc.) owing to the continuous mobility between domains, checks the validity of identities and eliminates a certain identity when an entity ceases to belong to the Grid or is no longer valid.
- Mobile Policy: this service is responsible for managing the specific policies on the use of mobile devices in the Grid and should be considered in all functions and operations of security services of the architecture, where mobile devices are involved. This service provides operations to obtain a policy associated with a service, updates information about any particular policy, or generates and associates policies to specific services.
- Audit: this service records each event that has occurred with the necessary information to know in detail what has happened at a particular time in the system. This service therefore logs events of the system and recovers such events when they are requested by an auditor.
- Grid Security Policy: this service manages the security policies that govern the Mobile Grid system, and all the security services of the architecture must have at least one associated policy for the service to be able to carry it out. The function of this service is to obtain a policy stored in the system, to create and associate security policies, and to update existing policies.

Appendix 2

Some of the security requirements that we have considered are briefly described as follows:

- Accounting: the usage of Grid resources by the users and by groups of users must be used to discover abuses and to help

avoid them, to implement submission policies based on user quotas or on resource usage, etc.
- Anonymity: preserving anonymity is of great concern in mobile systems for several reasons.
- Assurance: this provides a means to qualify the security assurance level that can be expected of a hosting environment.
- Authentication: difficult issues with regard to authentication in Grids are scalability, trust across different certification authorities, revocation, key management and delegation.
- Authorization and access control: in Grids, local access mechanisms should be applied whenever possible, and the owner of a resource should be able to enforce local user authorization.
- Confidentiality: the nature of Grids forces data to be stored in accessible online databases. Data may also need to be replicated on multiple sites, and should therefore be stored in an encrypted form and remain consistent throughout.
- Credentials: interdomain access requires a uniform means of expressing the identities of users or resources, and must therefore employ a standard for the encoding of credentials.
- Delegation: privilege delegation for operations executed by a proxy is a basic requirement for Grid environments, among other reasons in order to satisfy the single sign-on requirement.
- Exportability: code is required to be exportable and executable in multinational testbeds. As a result of this, bulk encryption cannot be required.
- Freshness: freshness is related to authentication and authorization and is important in many Grid applications.
- Integration: in order to allow the use of existing services and resources, integration requirements call for the establishment of an extensible architecture with standard interfaces.
- Integrity: many applications have strong code or data integrity concerns. Integrity is also an issue with regard to delegation, since the set of rights that have been delegated must not be modified maliciously.
- Interoperability: in the context of Mobile Grids, interoperability signifies that services within a single virtual organization must be able to communicate across heterogeneous domains.
- Manageability: this explicitly recognizes the need for manageability of security functionality within the OGSA security model. For example, identity management, policy management, key management and so forth.
- Mobility: since mobile devices come with many capabilities, mobile applications must run on a wide variety of devices, including the devices embedded in various environments and devices carried by users.
- Multiple implementations: it should be possible to enforce security requirements with distinct security technologies and mechanisms.
- Non-repudiation: non-repudiation refers to the inability to falsely deny the performance of a particular action. It is especially important in e-commerce which involves money transactions and mobile environments.
- Policy exchange: this allows service requestors and providers to dynamically exchange security (among other) policy information in order to establish a negotiated security context between them.
- Privacy: privacy is the ability to keep information from being disclosed to determined actors. Privacy can be important in many Grid applications, such as medical and health Grids.
- Revocation: revocation is crucial for authentication in the case of a compromised key and for authorization when a VO is terminated or a user or mobile user proves to be untrustworthy.
- Scalability: A grid must be easy to extend and capable of progressive replacement in mobile environments.
- Secure group communication: authenticated communications for dynamic groups is required since the composition of a process group may change dynamically during execution.
- Secure logging: this provides all the services, including the security services themselves, with facilities for time-stamping and securely logging any kind of operational information or event in the course of time.
- Single sign-on: a user should be able to authenticate only once, whereupon s/he may acquire, use and release resources without further authentication in different domains of the Grid.
- Trust: sites in a Grid must be able to enter into trust relationships with Grid users, mobile users and perhaps other Grid sites.

References

Alam M, Breu R, Hafner M. Model-driven security engineering for trust management in SECTET. Journal of Software 2007;2(1):47–60.

Anderson R. Security engineering—a guide to building dependable distributed systems. John Wiley&Sons; 2001.

Artelsmair C, Wagner R. Towards a security engineering process. In: Proceedings of the the seventh world multiconference on systemics, cybernetics and informatics, Orlando, Florida, USA; 2003.

Basin D, Doser J, Lodderstedt T. Model driven security for process-oriented systems. In: Proceedings of the ACM symposium on access control models and technologies, Como, Italy: ACM Press; 2003.

Basin D, Doser J, Lodderstedt T. Model Driven security: from UML models to access control infrastructures. ACM Transactions on Software Engineering and Methodology 2006;15(1):39–91.

Baskerville R. Information systems security design methods: implications for information systems development. ACM Computing Surveys 1993;25(4):375–414.

Belani E, Vahdat.A, Anderson T, Dahlin M. CRISIS: a wide area security architecture. In: Proceedings of the seventh USENIX security symposium; 1998.

Bellavista P, Corradi A. The Handbook of Mobile Middleware. Auerbach Publications; 2006.

Bhanwar S, Bawa S. Securing a grid. World Academy of Science, Engineering and Technology 2008.

Bradford PG, Grizzell BM, Jay GT, Jenkins JT. Cap. 4. Pragmatic security for constrained wireless networks. In: Security in distributed, grid, mobile, and pervasive computing, Tuscaloosa, USA: A. Publications. The University of Alabama; 2007. 440 p.

Chapin S, Wang C, Wulf W, Knabe F, Grimshaw A. A new model of security for metasystems. Future Generation Computer Systems 1999;15(5–6):713–22.

Chu, D, Humphrey M. Mobile OGSI.NET: Grid computing on mobile devices. In: Proceedings of the fifth IEEE/ACM international workshop on grid computing—Grid2004 (at Supercomputing 2004); 2004.

Dail H, Sievert O, Berman F, Casanova H, YarKhan A, Vadhiyar S, et al., 2004. Scheduling in the grid application development software project. In: Grid resource management: state of the art and future trends, p. 73–98.

Deubler, M, Grünbauer J, Popp G, Wimmel G, Salzmann C. Towards a model-based and incremental development process for service-based systems. In: Proceedings of the international conference on software engineering (IASTED SE 2004), Innsbruck, Austria; 2004.

EGEE Middleware Design Team. EGEE Middleware Architecture. From 〈https://edms.cern.ch/document/476451/〉; 2004.

Enterprise Grid Alliance Security Working Group. Enterprise Grid Security Requirements Version 1.0; 8 July 2005.

Estay C, Pastor J. Towards the project-based action-research for information systems. In: Proceedings of the 10th annual business and information technology conference (BIT'2000), Manchester, United Kingdom; 2000.

Fernández-Medina E, Jurjens J, Trujillo J, Jajodia S. Special issue: model-driven development for secure information systems. Information and Software Technology 2009;51(5):809–14.

Fernandez E. Security patterns and secure systems design. Dependable Computing 2007:233–4.

Fernández EB, Larrondo-Petrie MM, Sorgente T, Vanhilst M. Chapter V. A methodology to develop secure systems using patterns. In: Mouratidis H, Giorgini P, editors. Integrating security and software engineering. Advances and future vision. Idea Group Publishing; 2007. p. 107–26.

Fernandez EB, Wu J, Larrondo-Petrie MM, Shao Y. On building secure SCADA systems using security patterns. In: Proceedings of the fifth annual workshop on cyber security and information intelligence research: cyber security and information intelligence challenges and strategies. Oak Ridge, Tennessee: ACM; 2009.

Flechais I, Sasse MA, Hailes SMV. Bringing security home: a process for developing secure and usable systems. New security paradigms workshop (NSPW'03), Ascona, Switzerland; 2003.

Foster I, Kesselman C. The grid2: blueprint for a future computing infrastructure. 2 edition San Francisco, CA: Morgan Kaufmann Publishers; 2004.

Foster I, Kesselman C, Tsudik G, Tuecke S. A security architecture for computational grids. In: Proceedings of the fifth conference on computer and communications security, San Francisco, USA: ACM Press; 1998.

Franke Hans A., Koch Fernando L., Rolim Carlos O., Westphall Carlos B., Balen Douglas O. Grid-M: middleware to integrate mobile devices, sensors and grid computing. In: Proceedings of the third international conference on wireless and mobile communications (ICWMC′07) Guadeloupe, French Caribbean; 2007.

Giguhre E. Java 2 micro edition: the ultimate guide to programming handheld and embedded devices. John Wiley & Sons, Inc.; 2001.

Giorgini P, Mouratidis H, Zannone N. Modelling security and trust with secure tropos. In: Giorgini HMaP, editor. Integrating security and software engineering: advances and future visions. Idea Group Publishing; 2007. p. 160–89.

Globus Project. Grid security infrastructure (GSI). From 〈www.globus.org/security〉; 2005.

Graham D. Introduction to the CLASP Process. From 〈https://buildsecurityin.us-cert.gov/daisy/bsi/articles/best-practices/requirements/548.html〉; 2006.

Guan T, Zaluska E, Roure DD. A grid service infrastructure for mobile devices. In: Proceedings of the first international conference on semantics, knowledge, an grid (SKG 2005), Beijing, China; 2005.

Gutiérrez C, Fernández-Medina E, Piatini M. PWSSec: process for web services security. In: Proceedings of the IEEE international conference on web services, Orlando, Florida, USA; 2005.

Humphrey, M, Thompson MR, Jackson KR. Security for Grids Lawrence Berkeley National Laboratory. Paper LBNL-54853; 2005.

IEEE. Recommended practice for architectural description of software-intensive systems (IEEE Std 1471-2000). New York, NY: Institute of Electrical and Electronics Engineers; 2000. 29 p.

Jacobson I, Booch G, Rumbaugh J. The unified software development process. Addison-Wesley Professional; 1999.

Jameel H, Kalim U, Sajjad A, Lee S, Jeon T. Mobile-to-grid middleware: bridging the gap between mobile and grid environments. In: Proceedings of the European grid conference EGC 2005, Amsterdam, The Netherlands: Springer; 2005.

Jana D, Chaudhuri A, Bhaumik NB. Privacy and anonymity protection in computational grid services. International Journal of Computer Science and Applications 2009;6(1):98–107.

Jürjens J. Secure systems development with UML. Springer; 2005.

Jürjens J, Schreck J, Bartmann P. Model-based security analysis for mobile communications. In: Proceedings of the international conference on software engineering, Leipzig, Germany: IEEE Computer Society; 2008.

Kalim U, Jameel H, Sajjad A, Lee S. Mobile-to-grid middleware: an approach for breaching the divide between mobile and grid. In: Proceedings of the fourth international conference on networking, Reunion Island, France: Springer; 2005.

Kim, H.-K. Automatic translation form requirements model into use cases modeling on UML. In: Proceedings of the ICCSA 2005, LNCS; 2005. p. 769–77.

Kolonay R, Sobolewski M. Grid interactive service-oriented programming environment. Concurrent engineering: the worldwide engineering grid. Tsinghua, China: Press and Springer Verlag; 2004.

Kumar A, Qureshi SR. Integration of mobile computing with grid computing: a middleware architecture. In: Proceedings of the second national conference on challenges & opportunities in information technology (COIT-2008), Mandi Gobindgarh, India; 2008.

Kwok-Yan L, Xi-Bin Z, Siu-Leung C, Gu M, Jia-Guang S. Enhancing Grid Security Infrastructure to Support mobile computing nodes. Lecture Notes in Computer Science 2004;2908/2003:42–54.

Litke A, Skoutas D, Varvarigou T. Mobile grid computing: changes and challenges of resource management in a mobile grid environment. In: Proceedings of the fifth international conference on practical aspects of knowledge management (PAKM 2004); 2004.

Maña A, Serrano D, Ruiz JF, Armenteros A, Crespo BGN, Muñoz A. Development of applications based on security patterns. In: Proceedings of the Second International Conference on Dependability. DEPEND′09; 2009.

Mens T, Van Gorp P. A taxonomy of model transformation. Electronic Notes in Theoretical Computer Science 2006;152:125–42.

Mouratidis H. A security oriented approach in the development of multiagent systems: applied to the management of the health and social care needs of older people in England. University of Sheffield; 2004.

Mouratidis H, Giorgini P. Integrating security and software engineering: advances and future vision. IGI Global; 2006.

Mouratidis H, Giorgini P. Secure tropos: a security-oriented extension of the tropos methodology. International Journal of Software Engineering and Knowledge Engineering 2007;17(2):285–309.

Nagaratnam N, Janson P, Dayka J, Nadalin A, Siebenlist F, Welch V., et al. The Security Architecture for Open Grid Services; 2003.

OMG. Software & systems process engineering meta-model specification (SPEM) 2.0; 2008.

Open Grid Forum. The open grid services architecture, Version 1.5; 2006.

Open Group. TOGAF™ Version 9—the open group architecture framework. From 〈http://www.opengroup.org/architecture/togaf9-doc/arch/〉; 2009.

Osis J, Asnina E. Enterprise modeling for information system development within MDA. In: Proceedings of the annual Hawaii international conference on system sciences, Waikoloa, Big Island, Hawaii; 2008.

Phan T, Huang L, Ruiz N, Bagrodia R. Chapter 5: integrating mobile wireless devices into the computational grid. In: Ilyas M, Mahgoub I, editors. Mobile computing handbook. Auerbach Publications; 2005.

Ramakrishnan L. Securing next generation grids. IT Professional 2004;6:34–9. IEEE Computer Society.

Rosado DG, Fernández-Medina E, López J. Applying a UML Extension to build Use Cases diagrams in a secure mobile Grid application. In: Proceedings of the fifth international workshop on foundations and practices of UML, in conjunction with the 28th International conference on conceptual modelling, ER 2009, Gramado, Brasil, LNCS 5833; 2009a.

Rosado DG, Fernández-Medina E, López J. Obtaining security requirements for a mobile grid system. International Journal of Grid and High Performance Computing 2009b;1(3):1–17.

Rosado DG, Fernández-Medina E, López J. Reusable security use cases for mobile grid environments. Workshop on software engineering for secure systems, in conjunction with the 31st international conference on software engineering, Vancouver, Canada; 2009c.

Rosado DG, Fernández-Medina E, López J. Security Services Architecture for Secure Mobile Grid Systems. Journal of Systems Architecture. Special Issue on Security and Dependability Assurance of Software Architectures 2010. doi:10.1016/j.sysarc.2010.05.009.

Rosado DG, Fernández-Medina E, López J, Piattini M. Engineering process based on grid use cases for mobile grid systems. In: Proceedings of the the third international conference on software and data technologies—ICSOFT 2008, Porto, Portugal; 2008a.

Rosado DG, Fernández-Medina E, López J, Piattini M. PSecGCM: process for the development of secure grid computing based systems with mobile devices. In: Proceedings of the international conference on availability, reliability and security (ARES 2008), Barcelona, Spain: IEEE Computer Society; 2008b.

Rosado DG, Fernández-Medina E, López J. Towards an UML Extension of Reusable Secure Use Cases for Mobile Grid systems. IEICE TRANSACTIONS on Information and Systems 2011; E94-D(2).

Rosado DG, Fernández-Medina E, López J, Piattini M. Analysis of secure mobile grid systems: a systematic approach. Information and Software Technology 2010a;52:517–36.

Rosado DG, Fernández-Medina E, López J, Piattini M. Developing a secure mobile Grid system through a UML extension. Journal of Universal Computer Science 2010b;16(17):2333–52.

Sajjad A, Jameel H, Kalim U, Han SM, Lee Y-K, Lee S. AutoMAGI—an autonomic middleware for enabling mobile access to grid infrastructure. In: Proceedings of the joint international conference on autonomic and autonomous systems and international conference on networking and services—(icas-icns′05); 2005.

Siponen M, Baskerville R, Kuivalainen T. Extending security in agile software development. Integrating security and software engineering: advances and future visions. Idea Group Publishing; 2007. p. 143–59.

Steel C, Nagappan R, Lai R. Chapter 8. The alchemy of security design methodology, patterns, and reality checks. core security patterns: best practices and strategies for J2EE, web services, and identity management. Prentice Hall; 2005. 1088 p.

Talukder A, Yavagal R. Chapter 18: security issues in mobile computing. mobile computing. McGraw-Hill Professional; 2006.

van Steen M, Homburg P, Tanenbaum AS. Globe: a wide-area distributed system. IEEE Concurrency 1999:70–8.

Vivas JL, López J, Montenegro JA. Chapter 12. Grid security architecture: requirements, fundamentals, standards, and models. Security in distributed, grid, mobile, and pervasive computing. Tuscaloosa, USA: A. Publications; 2007. 440 p.

Welch V, Siebenlist F, Foster I, Bresnahan J, Czajkowski K, Gawor J, et al. Security for Grid services. In: Proceedings of the 12th IEEE international symposium on high performance distributed computing (HPDC-12 ′03), IEEE Computer Society; 2003.

Zhou Q, Yang G, Shen J, Rong C. A scalable security architecture for grid. In: Proceedings of the sixth international conference on parallel and distributed computing, applications and technologies, IEEE Computer Society; 2005. p. 89–93.