RUP-Based Process Model for Security Requirements Engineering in Value-Added Service Development

Hrvoje Belani[1], Željka Car[1], Antun Carić[2]
[1] *University of Zagreb, Faculty of Electrical Engineering and Computing*
Department of Telecommunications, Unska 3, HR-10000 Zagreb, Croatia
[2] *Croatian Post and Electronic Communication Agency*
Jurišićeva 12, HR-10002 Zagreb, Croatia
{hrvoje.belani, zeljka.car}@fer.hr, antun.caric@telekom.hr

Abstract

Due to the spreading of SMS services and appearing of new business models, value-added SMS services have been introduced. According to the research results about wide distribution of security incidents on ICT systems worldwide, in spite of known security solutions, there is a necessity for organizational approach to implement security. This paper presents research and development efforts in building process model SecuRUP for security requirements engineering conformed to RUP framework. The model consists of processes, artifacts, activities and according roles for successful elicitation, analysis and specification of recognized security requirements and is validated on presented case study. The model validation results have shown significant process improvement, especially on roles and activities identification in SecuRUP elaboration process, but only further case studies in industry can be best indicators for usefulness of such models.

1. Introduction

Software development for information and communication technology (ICT) services becomes more interdisciplinary nowadays, having to produce fully functional, secure and usable solutions for users. These services provide added values to users and are referred as value added services (VAS) in telecommunication domain. Besides the standard communication link, data or voice, added values are represented by the content and communication specifics combined from different network and content resources [1].

In a mobile network, short message services (SMS) and multimedia message services (MMS) are considered as value added services related to standard voice call. Recently, due to the spreading of SMS services and appearing of new business models, value added SMS and MMS services have been introduced. Besides the peer-to-peer messaging, premium charged services are offered either by telecom or third party operators, called value added service providers (VASP) [2]. Content providers typically connect to the operator via short message peer-to-peer protocol (SMPP), or directly through the network entity called simple message service center (SMSC), or even through some messaging gateways that (MGs) enable more efficient control and charging of delivered content.

Converging domains of internet and telecom services have new demands for security and protection of systems and users, with greater concern for user privacy. System-user relations between the attributes of security, privacy and protection can be described in a simple system-user model that assigns protection to both entities of system and user, providing them with adequate level of protection from the environment. Ideally, model entities of system (or user) are in full control of leveling their security (or privacy) according to the environment.

According to the research results [3] about wide distribution of security incidents on ICT systems worldwide, in spite of known security solutions, there is a necessity for organizational approach to implement security. Also, there is a recognized need to start taking care of security in early software development phases [4]. This paper presents a contribution in extending standard software process framework called Rational Unified Process (RUP) [5] with security requirements elicitation, analysis and specification activities, roles that perform these activities and artifacts that represent inputs or outputs of these activities.

The structure of this paper is as follows: section 2 covers the topic of value added services development, with security considerations in development lifecycle, and software process modeling approach on tailoring RUP framework. Section 3 presents a model development case study, while section 4 gives a process model for security requirements elicitation, analysis and specification, with all necessary aspects covered by the model to make it suitable for its purpose. Section 5 discusses related work and compares the presented model with existing solutions. The last section concludes the paper, emphasizing the contribution and limitations of the model.

2. Value added service development

Various SMS applications appeared on telecom market in the last decade, providing value added services to their users. Three main types of applications can be recognized [2], with few service examples given:

- Consumer applications, like person-to-person messaging, information services (or machine-to-person), download services and chat applications,
- Corporate applications, like vehicle positioning, remote monitoring, vending machine management, and all other services tailored for the needs of professionals, and
- Operator applications, which employ SMS as a building block for enabling the realization of services such as SIM lock, SIM updates, message waiting indicator and WAP push.

A case study in this article represents primarily SMS application for consumers, and is integral part of the School notification service (SMS SNS). SMS SNS enables two-way, accurate and personalized notification exchange between school, teachers, students and parents. It contributes to the increased efficiency of the schools' processes, to the better parents' participation in the education of children and to the improved overall quality of education system. Although SMS is the main user interface for this application, there are also other interfaces available to users, such as Web access for parents and school administrators and e-mail notification options. Because of the variety of available user interfaces, and the certain complexity of system architecture that needs to implement all desired VAS functionality, security needs to be taken into account in early phases of application development.

2.1. Security in SDLC

Software security aspects must be considered from the application vision and design, to its implementation and full operation. It is necessary to define and analyze security needs, specify requirements and implement mechanisms in order to make value added service more secure, but also usable to users.

A secure software process is the set of activities performed to develop, deliver and maintain secure software solution [6]. As there are already some software processes widely in professional use, like earlier mentioned RUP, it is justifiable to combine existing process activities with those security-related that are maybe missing. Fig. 1 shows some examples of security-related best practices that are concurrent to typical software development lifecycle (SDLC) phases.

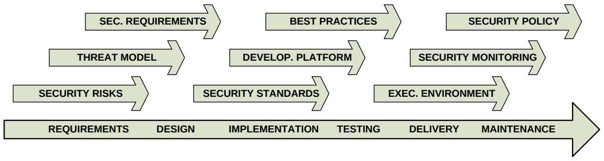

Figure 1. Security-related activities and typical software development lifecycle phases

2.2. RUP tailoring

As we have already mentioned, process framework widely used in software development industry today is RUP. This iterative model consists of four main phases: inception (I), elaboration (E), construction (C) and transition (T). They can practically be conducted iteratively, e.g. in particular VAS development the process phases can be defined in the following order: I1-E1-E2-C1-C2-T1, with two iterations of elaboration and construction.

For each phase, RUP proposes various intensities of activities like business modeling, implementation and testing, grouped in disciplines that need and produce various artifacts, and are conducted by different roles. While RUP relies on best practices of software development, like iterative development, requirements management, component-based architecture, visual modeling, quality verification and change control, it should also be tailored based on the specific project needs [7]. RUP tailoring is the procedure of selecting adequate subset of activities, artifacts and roles RUP framework offers for implementing VAS.

Although RUP covers broad set of activities and suggests many roles and artifacts, those security-related ones aren't soundly pointed out. It is important to also be aware that software security issues cannot be solved during one activity, but spread through all RUP phases, with different levels of involvement needed.

3. Model development case study

Case study presented in this section was used for developing a process model for software security requirements engineering named SecuRUP [1]. Justification for developing such a model lays in a fact that new-founded research and development (R&D) company unit needed new processes to be introduced and used for software development and research activities of their employees. Basically the first real-life project of the unit served as a ground for developing this model.

Plan for SecuRUP process model development is shown on Fig. 2. In order to develop such a security-related process model it is strongly advised to firstly learn software security concepts and have adequate domain knowledge in the field.

Afterwards, RUP framework should be tailored based on the recognized organizational capacities and project needs, in parallel with analysis of best practices and recommendations of development and, specifically, security domain. If the affirmative decision is made to develop particular value added service, it is necessary to continuously monitor and analyze eventual process shortfalls. According to the conducted analysis, the process should be upgraded and the process model evolved and verified.

Iterative nature of presented plan for developing SecuRUP gives the opportunity to company development unit to evolve the process model after conducting every new VAS development project, based on lessons learned and experiences gained. Process evolution and gradual implementation is also necessary for developers and other team members to get used to the process activities and roles, and to embrace the importance of defined artifacts for overall project success.

3.1. Security requirements for VAS development

It is relevant to point out that a model development case study was the real-life VAS development project that has included ten developers in new working environment, and RUP-based roles and tasks have been assigned dynamically to them during the project. It is the common characteristic of new and small project teams [8] to have agile approach in adjusting to the changes, e.g. while accepting new development methods, introducing new tools, etc.

RUP framework, along with its FURPS+ requirements categorization model [5], sees security requirements as functional ones, although many other classifications treat security as non-functional attribute of the system. There are just a few guidelines given in RUP for recognizing and specifying security requirements:

- Recognizing system resources that need protection,
- Recognizing persons and entities as possible threats to system operation, and
- Recognizing and implementation of security requirements.

Although security requirements ought to be collected, analyzed, specified and implemented, within RUP there are no specified activities, roles and artifacts to be produced in order to successfully fulfill these actions.

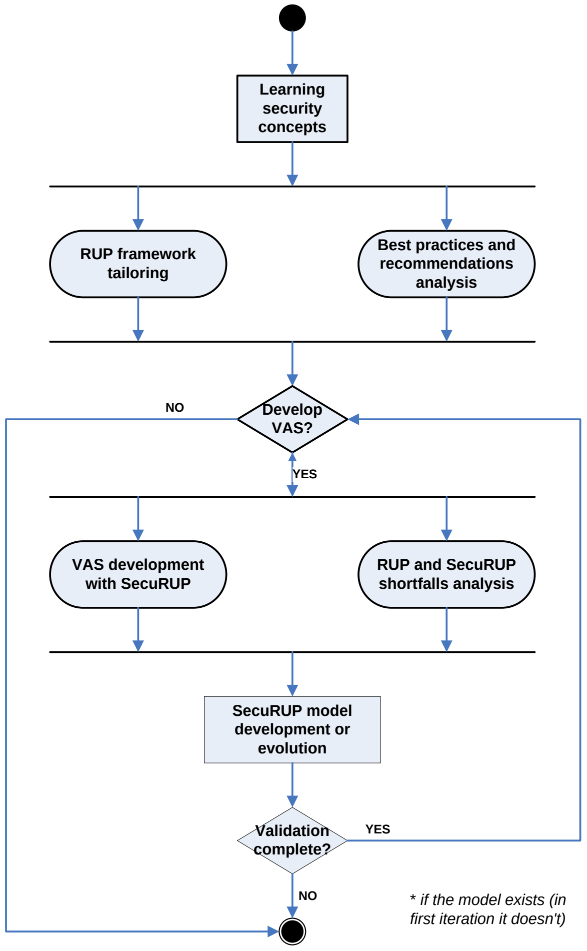
Learning security concepts
RUP framework tailoring
Best practices and recommendations analysis
NO
Develop VAS?
YES
VAS development with SecuRUP
RUP and SecuRUP shortfalls analysis
SecuRUP model development or evolution
Validation complete?
YES
NO
* if the model exists (in first iteration it doesn't)

Figure 2. Plan for iterative SecuRUP process model development and/or evolution

3.2. Case study implementation and testing

During the case study, potential security risks for the particular VAS were considered and analyzed on several different levels [9]:

- Web application security,
- Mobile access security,
- Data access security, and
- Secure communication.

Security considerations of physical deployment of the VAS in telecom network were also given, with the telematic (compound word of telecommunications and informatics) service model analyzed on three layers:

- Communication service provider - relates to communication protocols that provide semantically transparent information exchange, like X.25, TCP/IP or Parlay X protocols;
- Application protocol - relates to protocols that serve application layer, like SMPP (Short Message Peer-to-peer Protocol), HTTP (Hypertext Transfer Protocol) and SMTP (Simple Mail Transfer Protocol) protocols; and
- Application - related to application protocol services for end user on his communication interface, like SMS, WWW or e-mail.

When approaching security-preserving architecture design, according to the recommended security patterns, the security architecture for the system has been developed. With two types of roles for application-level authentication and authorization defined, the access to the system features was role-delegated, supporting the user's actions inside the sessions. Authentication mechanism used authentication cookies, and user credentials were securely stored in the database.

In order to protect communication between client and server, Web application was made accessible only through HTTPS protocol, using SSL 3.0 (Secure Socket Layer). Many other security mechanisms were implemented, including brute force attack detection, IP banning and system misusage detection.

Security attacks in complex systems, like VAS systems, are always the result of combination of faults. Ideally, to avoid all security issues, it is needed to anticipate all combinations of problems. This is practically impossible, so making a system perfectly secure is therefore impossible.

In order to discover and remove recognizable security flaws, there are different security testing techniques, which vary in methodology, scope and purpose of testing. So, some of security testing in the case study were conducted manually (e.g. blind penetration tests), while others had very useful automatic support (e.g. stress tests).

3.3. Experiences in VAS deployment

Value added service from the building model case study was successfully implemented and deployed on several operation sites for trial and commercial usage. During the case study VAS operation data were collected from two deployment sites, statistics for site “A“ were months-based and for site “B” were weeks-based.

Common security-related activities to both deployment sites were regular system upgrades, installation of operating system patches and antivirus and firewall software, in order to provide secure baseline for VAS operation and high availability.

System administrator monitored daily reports from the VAS system operation logs, in order to timely react, if necessary, with locking out user account. For both sites these actions weren't needed so far. Some of the IP addresses were black-listed, but the number of addresses on both sites didn't exceed the limit of four in three weeks, or one black-listed address in 5.25 days.

Due to the fact that end users were not adept to use Web and SMS technologies, system logs on both sites have recorded user requests for new passwords for Web access, and user errors on typing-in PIN (Personal Identification Number) while using SMS interface. Nevertheless, user knowledge grew with time, and these numbers were lowering.

Other VAS operation data can be valuably interpreted when all of statistics is put into context of service usability, e.g. statistics for successful login attempts, login failures and number of both SMS per requests while identifying end user with PIN. Security usability is a brand new and developing area of research [10], furthermore value added services have just the user in focus, his perception of service and its user-friendliness, but not detrimentally on security.

4. Process model SecuRUP

RUP lacks of activities, roles and information flow needed for successful implementation of security. As said before, SecuRUP represents new process model for implementation of security requirements for value added services. SecuRUP is based on RUP process framework and built upon four RUP phases, but only the first two are dealing significantly with requirements engineering.

Therefore, in this paper, SecuRUP is presented with models for inception and elaboration, along with model usage experiment results. SecuRUP also contains detailed models [1] for construction and transition phases, but these exceed the scope of this paper.

4.1. SecuRUP for inception

SecuRUP for inception introduces two new roles, Security Analyst and Domain Expert, and two new artifacts, Security Vision and Software Security Body of Knowledge, explained in further text. Other roles, artifacts and activities presented in this model (shown in Fig. 3) already exist in RUP and are tightly connected with new security-related activities.

4.1.1. Security Analyst. This role evaluates security needs for the system and service, defines security requirements and manages their implementation into the service. Actor of this role has to have outstanding communication abilities for permanent and continuous information exchange with other roles in development team and with end users as well.

4.1.2. Domain Expert. This role has specific knowledge from domains of implementation of rather sophisticated security mechanisms into the service. This role is valuable player with state of the art knowledge in software security.

4.1.3. Software Security Body of Knowledge. This artifact represents an existing, available and upgradeable library of known security concepts, methods and mechanisms, with best practice advices for their implementation into VAS. It includes the following artifacts into its content: **Known Attacks Patterns Catalog** - suite of known ways of jeopardizing VAS security, recognized from various security threat models; **Security Design Patterns Catalog** - suite of known design patterns for architecting secure solution for VAS. Whole artifact is built once the SecuRUP has been introduced into the organization, and its size (document pages, model size, etc.) can vary from organization to organization and from one domain expert to another.

4.1.4. Security Vision. This artifact is similar to the software Vision artifact and contains descriptions of all security aspects for particular VAS, along with the project scope and other issues of security implementation. Security Vision includes the following artifacts into its content: **User Requests** - manageable list of requests from the VAS users collected with one or few requirements elicitation techniques (such as prototyping, questionnaires and interviewing); **Security Risks** - manageable list of security risks, recognized as possibilities for manifesting security threats. It is advisable to contain some applicable attack patterns as well.

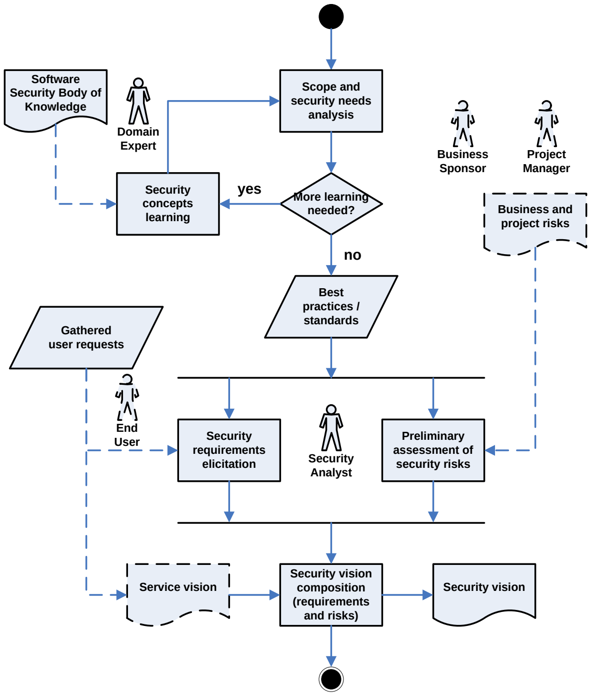
Software Security Body of Knowledge
Domain Expert
Scope and security needs analysis
Business Sponsor
Project Manager
Security concepts learning
yes
More learning needed?
Business and project risks
no
Best practices / standards
Gathered user requests
End User
Security requirements elicitation
Security Analyst
Preliminary assessment of security risks
Service vision
Security vision composition (requirements and risks)
Security vision

Figure 3. SecuRUP model for inception

4.2. SecuRUP for elaboration

SecuRUP for elaboration introduces two new roles, Security Developer and Security Tester, and two new artifacts, Security Requirements Specification and Security Implementation Plan, explained in further text. Other roles, artifacts and activities included in this model (shown in Fig. 4) already exist in RUP and are tightly connected with new security-related activities, as well as other regular process elements of RUP.

4.2.1. Security Developer. This role is responsible for implementing specified security requirements and has good knowledge of security features of targeted development platform. The actor of this role needs to be in frequent communication with Security Analyst and Security Tester roles.

4.2.2. Security Tester. This role conducts various security tests over targeted VAS software system, from Web application vulnerability tests to rather sophisticated penetration tests. The actor of this role logically belongs to the testing team.

4.2.3. Security Requirements Specification. This artifact contains detailed specification of security requirements, along with the following artifacts: **Security Use Cases** - UML specified behavior of VAS users and VAS system according to the recognized security needs; **Security Abuse Cases** - UML-like specified behavior of VAS abusers and attackers according to the recognized threat models, that results of analysis of external and internal security threats. Data used in the system needs to be classified in order to have full control of who can create, read, update and delete (CRUD) data, when and under what conditions.

4.2.4. Security Implementation Plan. This artifact suggests particular solutions for security implementation into VAS, provides applicable models and plan for their implementation into the VAS. It contains also the following artifacts: **Security Architecture** - architectural model for VAS with all security features gained from specified security requirements, depending on development platform, programming language, level of integration with existing, proprietary systems, etc; **Security Check-Lists** - manageable list of particular and concrete security threats and attacks with guided procedure how to do static code checks and eliminate these shortfalls.

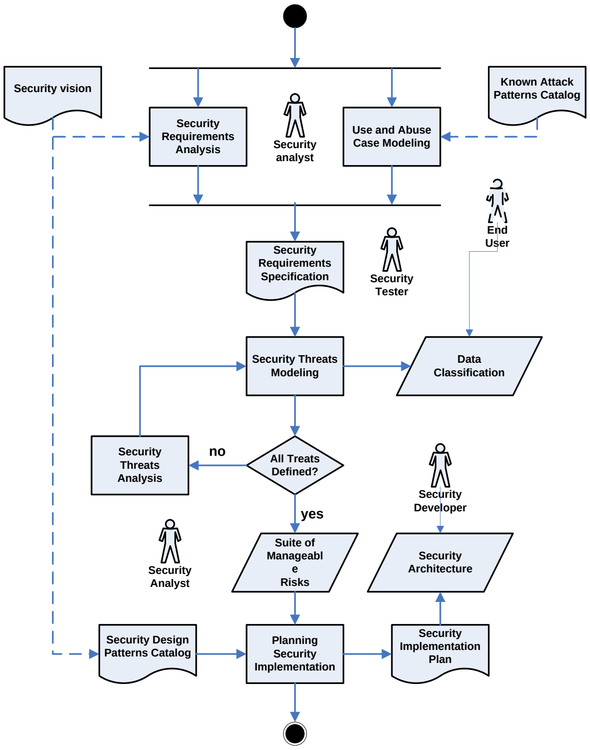
Security vision
Security Requirements Analysis
Security analyst
Use and Abuse Case Modeling
Known Attack Patterns Catalog
End User
Security Requirements Specification
Security Tester
Security Threats Modeling
Data Classification
Security Threats Analysis
no
All Treats Defined?
yes
Security Developer
Security Analyst
Suite of Manageabl e Risks
Security Architecture
Security Design Patterns Catalog
Planning Security Implementation
Security Implementation Plan

Figure 4. SecuRUP model for elaboration

4.3. Model usage experiment

After building the SecuRUP model, its validation was conducted on new case study for RUP phases of inception and elaboration. Preliminary validation was done while analyzing and specifying security and even privacy requirements for new VAS.

First results in the new VAS development project have shown significant improvement in team ability to recognize which security-related roles and artifacts are every of them supposed to act, use and produce.

With process model like SecuRUP, there is a strict way of dealing with documented knowledge when to do which activity and with what inputs, what artifact to use and make of it, and which role needs to be played within.

In the first case study, only one, and sometimes two, team members were engaged in security-related activities, on ad-hoc basis and with no process guidance, while introducing SecuRUP into the second case study changed the picture - five team members have recognized what they need to do and conducted their activities successfully.

Although the number of roles and artifacts was increased, security-related activities early in the SDLC annulled the extra effort of later need for exhausting testing and other negative consequences, like customer unreliability into the service and regarding financial loses. Based on experiences and lessons learned from these projects, conceptual framework for business process engineering was developed [11], as a solid ground for further work in bridging the gap between business and IT domain when developing value added services.

5. Related work

SecuRUP process model is comparable with some existing and alike solutions from the area of software engineering in ICT. There are five more process models - SQUARE [12], MS SDL, [13] CLASP [14], RUPSec [15] and TSP-Sec [6], freely available for wide usage, which can be compared with SecuRUP on two grounds:

- Conformance to RUP framework - full, partial or none, and
- Software development lifecycle phases coverage - full, partial or none.

From analysis of given process models, it can be seen that only RUPSec and SecuRUP are fully conformed to RUP, while CLASP model is just partially conformed, based on some commonalities recognized, but without detailed documentation available. It can be concluded that it is advisable to have process model conformed to widely spread process framework, such as RUP. In this way the efforts of getting the development team to learn and know the process is minimal, with significant positive feedback and results.

Next comparison shows the level of coverage of given process models with soft-ware development lifecycle (SDLC) phases. Even four models (MS SDL, CLASP, TSP-Sec, SecuRUP) are focused on phases from requirements till testing, while other two (SQUARE, RUPSec) are more concentrated on a single phase, early in software development lifecycle. If it is mandatory for development team to work in all SDLC phases, it is advisable to have security process model stretched over all the phases also.

6. Conclusion

Very misleading approach in software development, but very common in practice, is tendency of starting to take care of security yet after implementing a significant part of the software functionality. Thus, systems are made vulnerable by design and risks for potential security breaches are higher. Later implementation efforts usually produce insufficient results and customer looses it reliability into services, but also significant financial loses. Therefore, the need for security process model, like SecuRUP that is proposed in this paper, is imminent.

Except presented SecuRUP models for inception and elaboration, there are also detailed SecuRUP models for construction and transition phases developed [1], but they exceed the scope of this paper. Further SecuRUP evolution and improvement need to introduce software process metrics, in order to gain quantitative information of a process, for evaluating its advantages and limitations. Relevant context factors could be development team size, project complexity, etc. Although this model was built during VAS development, it is also inherently highly applicable for common ICT systems development, because almost every application has some kind of security requirement even if its use is not considered critical.

Still, unavoidable presumption remains that the model has to be verified and evolved further more on case studies from industry, because that is the best indicator for usefulness of such models. Academic environment has been proven right for developing the process model according to the already known solutions in the world. The necessity to have coherence between scientific approach and practical surroundings was once again proved right.

7. Acknowledgement

This work was carried out within the research project "Content Delivery and Mobility of Users and Services in New Generation Networks", supported by the Ministry of Science, Education and Sports of the Republic of Croatia.

8. References

[1] Belani, H. *Process Model for Implementation of Security Requirements for Value Added Services*, MSc thesis, FER, Zagreb, 2007. (In Croatian)

[2] Le Bodic, G. *Mobile Messaging Technologies and Services: SMS, EMS and MMS*, Wiley & Sons, Ltd, 2003.

[3] Gordon, L.A. Loeb, M.P. Lucyshyn, W. Richardson, R., “11th Annual CSI/FBI Computer Crime and Security Survey”, *Computer Security Institute*, San Francisco, 2006.

[4] Devanbu P.T., Stubblebine, S., “Software Engineering for Security: a Roadmap”, *Proceedings of the Conference on The Future of Software Engineering*. ACM Press, 2000.

[5] Kruchten, P., *The Rational Unified Process: An Introduction*, Addison Wesley Longman, 1999.

[6] Davis, N., “Secure Software Development Life Cycle Processes”, *CMU/SEI-2005-TN-024*. Software Engineering Institute. Carnegie Mellon University, 2005.

[7] Car, Ž. Labor, O. Carić, A. Huljenić, D., “Tailoring RUP: E-School Project Case Study”, *Proceedings of the MIPRO Convention*, Opatija, Croatia, 2004, pp. 27-32.

[8] Pripužić, K., Belani, H., Gjenero, L., “Improving Virtual Team Communication”, *Proceedings of the International Conference on Software, Telecommunications and Computer Networks SoftCOM*, Split-Dubrovnik, 2006.

[9] Herceg, S. Protega, G, Belani, H., “Administration, Management and Security Aspects of Panoptes System”, *Proceedings of the MIPRO Convention*, Opatija, Croatia, 2004, pp. 45-50.

[10] Cranor, L.F., Garfinkel, S., *Security and Usability: Designing Secure Systems That People Can Use*, O'Reilly, 2005.

[11] Car, Ž. Carić, A., Belani, H., “Conceptual Framework for Business Process Engineering: A Case Study”, *Proceedings of the ICSEA 2006*, Tahiti, 2006.

[12] Mead, N. R., Hough, E.D., Stehney II, T.R., “Security Quality Requirements Engineering (SQUARE) Methodology,” *Technical Report CMU/SEI-2005-TR-009, ESC-TR-2005-009*, 2005.

[13] Howard, M., “How Do They Do It? A Look Inside the Security Development Lifecycle at Microsoft”, *MSDN Magazine*, May 2005.

[14] Gregoire, J., Buyens, K., De Win, B., Scandariato, R., Joosen, W. “On the Secure Software Development Process: CLASP and SDL Compared”, *Third International Workshop on SESS*, Minneapolis, 2007.

[15] Jaferian, P., Shirazi, M.R.A., Sadeghian, B., “RUPSec: Extending Business Modeling and Requirements Disciplines of RUP for Developing Secure Systems”, *Proceedings of the EUROMICRO-SEAA'05*, 2005, pp. 232-239.