Network of Excellence

Deliverable D13.3

Second report describing the second annual summer school and the annual open competitions.
Abstract

In this deliverable, we report about two Education and Training activities of WP 13, the 12th FOSAD Summer School and the second annual Open Competition (eRISE). The 12th FOSAD summer school (12th International School on Foundations of Security Analysis and Design), involved 40 participants between post-graduate students, PhD students, and young researchers, from 27 universities and industrial laboratories; participants attended 8 lectures given by worldwide known experts in the field of Security Engineering. The eRISE (engineering Risk and Security Requirements Challenge) competition, aimed at evaluating the effectiveness of methods for modeling and analyzing security requirements and risks. The 2012 edition of the competition focused on the comparison of five methods and has involved 27 practitioners in IT Audit for Information Systems and 16 master students in Computer Science. We also report on another competition Web Security War Games which aims at helping people understand the intricacies of web server security.

Keyword List

Doctoral School; Education; Practitioners’ Training; Empirical Evaluation; Security Methods; Risk and Security Requirements.
Document History

<table>
<thead>
<tr>
<th>Version</th>
<th>Type of Change</th>
<th>Author(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>Table of Content</td>
<td>Federica Paci</td>
</tr>
<tr>
<td>0.2</td>
<td>Section about War Games</td>
<td>Pieter Philippaerts</td>
</tr>
<tr>
<td>0.3</td>
<td>Sections on eRISE 2012 and on FOSAD 2012</td>
<td>Federica Paci</td>
</tr>
<tr>
<td>0.4</td>
<td>Introduction and Conclusion sections</td>
<td>Federica Paci</td>
</tr>
<tr>
<td>0.5</td>
<td>Finalisation of document</td>
<td>Federica Paci</td>
</tr>
<tr>
<td>0.6</td>
<td>Revision of document</td>
<td>Federica Paci</td>
</tr>
</tbody>
</table>

Document Review

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Reviewer</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>20-01-2013</td>
<td>0.5</td>
<td>Benoit Baudry (INRIA)</td>
<td>Typos</td>
</tr>
<tr>
<td>20-01-2013</td>
<td>0.5</td>
<td>Marina Egea (ATOS)</td>
<td>Clarify the aim of eRISE and War Games</td>
</tr>
<tr>
<td>20-01-2013</td>
<td>0.5</td>
<td>Leanid Krautsevich (CNR)</td>
<td>Typos</td>
</tr>
</tbody>
</table>
# Table of Contents

**LIST OF FIGURES** ............................................................................. 7

**LIST OF TABLES** ........................................................................... 9

1 **INTRODUCTION** .......................................................................... 11

2 **FOSAD SUMMER SCHOOL** ...................................................... 13
   2.1 **Location** .............................................................................. 13
   2.2 **Steering Committee** ............................................................. 13
   2.3 **Participants** .......................................................................... 13
   2.4 **Lecturers** ............................................................................. 13
   2.5 **Program** ............................................................................... 14

3 **eRISE CHALLENGE 2012** ......................................................... 17
   3.1 **eRISE Design** ...................................................................... 17
   3.2 **The Comparative Evaluation Protocol** ................................ 18
      3.2.1 **Participants** ................................................................. 19
      3.2.2 **Selection of Methods** ................................................... 19
      3.2.3 **Application Scenarios** .................................................. 19
   3.3 **Data Analysis** ...................................................................... 19
      3.3.1 **Interview Transcripts’ Analysis** ..................................... 19
      3.3.2 **Post-it notes’ Analysis** ................................................. 21
      3.3.3 **Questionnaires’ Analysis** ............................................. 23
      3.3.4 **Data Triangulation: Reports’ Evaluation and Participants’ Background** 26
   3.4 **Threats to Validity** .............................................................. 27
   3.5 **Conclusions** ........................................................................ 28

4 **WAR GAMES** ............................................................................ 29

5 **CONCLUSIONS** ......................................................................... 31

**BIBLIOGRAPHY** ............................................................................ 33
List of Figures

Figure 3.1: eRISE editions’ timeline ................................................................. 17
List of Tables

Table 2.1: FOSAD 2012 program .......................................................... 15
Table 3.1: Methods’ positive aspects resulting from Affinity Analysis ........................................ 22
Table 3.2: Methods’ Limitations resulting from Affinity Analysis ........................................ 22
Table 3.3: Answers on Security Requirements’ Completeness ........................................... 23
Table 3.4: Answers on Method’s Positive and Negative Aspects ....................................... 24
Table 3.5: Answers on Method’s Positive and Negative Aspects ....................................... 25
1 Introduction

In this deliverable, we report on two education and training activities carried out in WP 13. The 12th FOSAD Summer School and the second annual Open Competition (eRISE). The 12th FOSAD summer school (13th International School on Foundations of Security Analysis and Design Timetable), took place in Bertinoro (Italy) between 3 September and 8 September 2012 and was co-organized in collaboration with NESSoS and CryptoForma. 40 participants including post-graduate students, PhD students, and young researchers, from 27 universities and industrial laboratories, participated to the summer school and attended 8 seminars given by worldwide known experts in the field of Security Engineering. The other education and training activity is the eRISE 2012 competition which aims at understanding how effective security requirements and risk analysis methods are and why. The 2012 edition of eRISE involved 16 master students with a background in computer science from University of Trento and 27 practitioners attending the MBA in IT Audit for Information Systems at University Paris Dauphine (France). The participants were first instructed about a specific security requirements and risk analysis method. Then, the participants divided in groups had to mimic a real team of security practitioners who analyze the security risks of the NESSoS case studies using one of the security requirements methods under evaluation. To collect data on methods' effectiveness, different data sources have been used. The participants have been audio-video recorded during the application phase. The artifacts generated by each group were collected, and a number of questionnaires were administered during the different phases of eRISE. At the end of application phase, the participants were interviewed and involved in affinity analysis sessions. As a preliminary result of the evaluation from eRISE 2011 and 2012, SECURE TROPOS, SECURITY ARGUMENTATION, SI* and CORAS are methods that effectively support brainstorming and modeling of the system under analysis. Contrary to expectation, they do not support the elicitation of specific security requirements because their identification is left to the analyst's knowledge in security and in the application domain. In contrast, SREP and LINDDUN support elicitation because their process guides the analyst to concrete security and privacy requirements.

Another education and training activity conducted in WP13 is War Games competition, which aims at helping people understand the intricacies of web server security. War games are popular with hackers and professional security experts alike, and offer a valuable training experience. In addition, the interest of different players can be boosted by introducing a competition element in the game and allow the players to play against each other.
2 FOSAD Summer School

Security in computer systems and networks emerged as one of the most challenging research areas. The International School on Foundations of Security Analysis and Design (FOSAD) has been one of the foremost events established with the goal of disseminating knowledge in this critical area. The main aim of the FOSAD school is to offer a good spectrum of current research in foundations of security - ranging from programming languages to analysis of protocols, from cryptographic algorithms to access control policies and trust management - that can be of help for graduate students and young researchers from academia or industry that intend to approach the field. This year the school has been organized in cooperation with the NoE on engineering secure future Internet software services and systems (NESSoS) and with the NoE on the application of formal methods to cryptography (CryptoForma). The school has been held from September 3 to September 8, 2012.

2.1 Location

FOSAD summer school is held annually at the University Residential Centre of Bertinoro (Italy), in the fascinating scenario of a former convent fortress that has been transformed into a modern conference facility.

2.2 Steering Committee

Most of the members of the school steering committee are related to NESSoS or are actively involved in NESSoS: Fabio Martinelli (NESSoS Coordinator), G. Barthe (NESSoS partner), Javier Lopez (NESSoS Partner), Sandro Etalle (NESSoS Associate Partner), Martin Abadi (member of the NESSoS NaLAB).

2.3 Participants

40 post-graduate students, PhD students, and young researchers from universities and industrial laboratories participated in FOSAD 2012 summer school.

2.4 Lecturers

- **Martin Abadi** is currently working at the University of California, Santa Cruz and Microsoft Research. He earned his Ph.D. from Stanford University in 1987 as a student of Zohar Manna. He is well known for his work on computer security and on programming languages, including his paper (with Michael Burrows and Roger Needham) on the Burrows-Abadi-Needham logic for analyzing authentication protocols, and his book (with Luca Cardelli) A Theory of Objects, laying out formal calculi for the semantics of object-oriented programming languages. (He is member of the NESSoS NaLAB.)
- **Karthik Bhargavan** is a researcher at INRIA, where he leads the PROSECCO project at INRIA Paris-Rocquencourt. His research concerns the theory, design, and implementation of modern programming language features and formal verification techniques. His recent work focuses on using formal methods to investigate the (in)security of distributed applications such as cryptographic protocols.
- **Fabrice Bouquet** is Professor of Software Engineering at the University of Franche-Comte (UFC) member of INRIA Cassis project since 2002 and Header of Model-Based Testing groups. His main research interests are semantics of modelling languages for test generation and test generation strategies. (He is member of NESSoS.)
- **Jan Jurjens** is professor for Software Engineering at Technical University Dortmund (Germany), Scientific Coordinator "Enterprise Engineering" at Fraunhofer Institute for Software and Systems Engineering ISST (Dortmund), and Senior Member of Robinson College (Univ. Cambridge, UK).
Jan holds a Doctor of Philosophy in Computing from University of Oxford and is author of “Secure Systems Development with UML” (Springer, 2005; Chinese translation 2009) and other publications mostly on software engineering and IT security, totalling over 2500 citations. (He is a NESSoS associate partner.)

- **Bart Preneel** is a full professor in the research group COSIC of the Electrical Engineering Department of the Katholieke Universiteit Leuven in Belgium. His research focuses on cryptographic algorithms and protocols as well as their applications to computer and network security and mobile communications. His favourite research topics are hash functions, MAC algorithms, stream ciphers and block ciphers. (He is member of NESSoS.)

- **Phillip Rogaway** is a professor of computer science at the University of California, Davis. He graduated with a BA in computer science from UC Berkeley and completed his PhD in cryptography at MIT, in the Theory of Computation group. His research has focused on obtaining provably-good solutions to protocol problems of genuine utility. He is also interested in social and ethical issues connected to technology.

- **Mark Ryan** is a professor in Computer Security at the School of Computer Science of University of Birmingham. His research interests include electronic voting, verification of protocols, verification of pervasive systems, trusted computing, and access control systems.

- **Pierangela Samarati** is a Professor at the Computer Science Department of the Universita degli Studi di Milano. Her main research interests are access control policies, models and systems, data security and privacy, information system security, and information protection in general. She has participated in several projects involving different aspects of information protection. On these topics she has published more than 200 peer-reviewed articles in international journals, conference proceedings, and book chapters. She is co-author of the book “Database Security,” Addison-Wesley, 1995. (She is member of the NESSoS NaLAB.)

### 2.5 Program

The detailed program of FOSAD 2012 is reported in table 2.1.
<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Speaker</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mon 3 Sep</td>
<td>09:00 - 11:00</td>
<td>Martin Abadi (University of California at Santa Cruz and Microsoft Research, US)</td>
<td>Software Security</td>
</tr>
<tr>
<td></td>
<td>11:30 - 13:30</td>
<td>Fabrice Bouquet (FEMTO-ST, FR)</td>
<td>Model Based Testing for Functional and Security Test</td>
</tr>
<tr>
<td></td>
<td>13:30</td>
<td>Lunch</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15:00 - 17:00</td>
<td>Mark Ryan (University of Birmingham, UK)</td>
<td>Verifying security protocols with ProVerif and StatVerif</td>
</tr>
<tr>
<td>Tue 4 Sep</td>
<td>09:00 - 11:00</td>
<td>Martin Abadi (University of California at Santa Cruz and Microsoft Research, US)</td>
<td>Software Security</td>
</tr>
<tr>
<td></td>
<td>11:30 - 13:30</td>
<td>Mark Ryan (University of Birmingham, UK)</td>
<td>Verifying security protocols with ProVerif and StatVerif</td>
</tr>
<tr>
<td></td>
<td>13:30</td>
<td>Lunch</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15:00 - 17:00</td>
<td>Jan Jurjens (Fraunhofer ISST Dortmund, GE)</td>
<td>Model-based Security Engineering</td>
</tr>
<tr>
<td>Wed 5 Sep</td>
<td>09:00 - 11:00</td>
<td>Jan Jurjens (Fraunhofer ISST Dortmund, GE)</td>
<td>Model-based Security Engineering</td>
</tr>
<tr>
<td></td>
<td>11:30 - 13:30</td>
<td>Fabrice Bouquet (FEMTO-ST, FR)</td>
<td>Model Based Testing for Functional and Security Test</td>
</tr>
<tr>
<td></td>
<td>13:30</td>
<td>Lunch</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15:00 - 17:00</td>
<td>Karthik Bhargavan (INRIA-Microsoft Research Joint Centre, FR)</td>
<td>Cryptography for the Web: Attacks and Proofs</td>
</tr>
<tr>
<td>Thu 6 Sep</td>
<td>09:00 - 11:00</td>
<td>Phillip Rogaway (University of California at Davis, US)</td>
<td>Provably Secure Shared-Key Encryption</td>
</tr>
<tr>
<td></td>
<td>13:30</td>
<td>Lunch</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15:00 - 17:00</td>
<td>open session</td>
<td></td>
</tr>
<tr>
<td>Fri 7 Sep</td>
<td>09:00 - 11:00</td>
<td>Phillip Rogaway (University of California at Davis, US)</td>
<td>Provably Secure Shared-Key Encryption</td>
</tr>
<tr>
<td></td>
<td>13:30</td>
<td>Lunch</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15:00 - 17:00</td>
<td>Pierangela Samarati (University of Milan, IT)</td>
<td>Privacy in Data Publication and Outsourcing Scenarios</td>
</tr>
<tr>
<td>Sat 8 Sep</td>
<td>09:00 - 11:00</td>
<td>Bart Preneel (KU Leuven, BE)</td>
<td>Hash Functions and MAC Algorithms: Theory and Practice</td>
</tr>
<tr>
<td></td>
<td>11:30 - 13:30</td>
<td>Pierangela Samarati (University of Milan, IT)</td>
<td>Privacy in Data Publication and Outsourcing Scenarios</td>
</tr>
<tr>
<td></td>
<td>13:30</td>
<td>closing and lunch</td>
<td></td>
</tr>
</tbody>
</table>

Table 2.1: FOSAD 2012 program
3 eRISE challenge 2012

eRISE 2012 had been conducted in May and June 2012. The Training phase took place at the University of Trento from May 7th to 9th. The Training was followed by two face-to-face Application phases: one was held at the University of Trento on May 10th and 11th. The second one took place at Dauphine University from June 14th to June 15th 2012. With respect to eRISE 2011, five academic methods were evaluated and compared during eRISE 2012: CORAS [5], LINDDUN [1] SECURE TROPOS [7], SECURITY ARGUMENTATION [3], and SREP [6]. In addition, Marina Egea from Atos Research and Jorge Cuellar from Siemens played the role of the customers and introduced the NESSoS case studies to be analyzed by the participants.

In the next sections the design of eRISE 2012 (§3.1) is presented and the protocol to run it with novice users (§3.2). Then the results of the analysis of the transcribed focus group interviews (§3.3.1), the results of affinity analysis on post-it notes (§3.3.2) and the quantitative and qualitative questionnaires’ analysis (§3.3.3) are reported. Then, the cross validation of reports’ assessment by method designers and customers with participants’ background (§3.3.4) and the threats to validity (§3.4) are discussed. Finally, some (meta)-lessons learned (§3.5) are reported.

3.1 eRISE Design

For conducting eRISE 2012, we use qualitative research inspired by the principles of grounded theory [9] because it is more suitable for investigating research questions of the type what, how, and why. In eRISE we want to investigate how academic security requirements methods actually work when applied by someone different than the method designer, what aspects make them work, and why.

To evaluate the effectiveness of the methods under evaluation, we have gathered different kinds of data:

- **Questionnaires** include questions on subjects’ knowledge of IT security, risk assessment, and requirements engineering and their evaluation of the methods’ aspects;

- **Audio/Video Recordings** capture the application of the methods by subjects and the focus groups interviews;

- **Post-it Notes** list positive and negative aspects about the methods and eRISE itself;
Focus Group Transcripts report the discussion on the methods’application between participants and observers;

Group Presentations by participants summarize the results of method’s application;

Final Reports describe in detail how participants have identified the security requirements following the method.

We have used both qualitative and quantitative analysis techniques. Questionnaires have been analyzed using statistical analysis. For post-it notes we have used affinity analysis [4] in order to group similar feedbacks on positive and negative aspects of the methods. The transcripts of the focus groups discussions have been analyzed using coding, a content analysis technique used in grounded theory [9]. Coding helped us to discover text patterns that are relevant to what makes methods effective in identifying security requirements and why. We have performed a qualitative analysis of the final reports.

3.2 The Comparative Evaluation Protocol

One of the goals of eRISE is to investigate whether the methods under evaluation could be used effectively by novices users to the method and therefore we have designed a protocol to conduct comparative empirical studies in this setting. The protocol consists of three main steps.

Training. First, participants are administered a questionnaire to collect information about their level of expertise in requirement engineering, security and on other methods they may know. Then, they are divided in groups of subjects with diverse backgrounds. Each group is assigned to a security requirements or a risk analysis method and to an application scenario to be analyzed using the method. The participants have to attend lectures about the method and on the industrial application scenario. At the end of the Training phase, the participants are administered a questionnaire to determine their level of understanding of the methods and of the industrial applications scenario.

Application. Participants work in groups and apply the method to analyze the application scenario. Group collaboration takes place both face-to-face and remotely by using multiple communication channels (e.g. mail, chat, video conferencing facilities). At the end of this phase, participants are involved in focus groups interviews, and they are requested to fill in post-it notes and a questionnaire about their impressions on the method. To document the application of the methods, the groups are audio-video recorded. In addition, groups have to deliver a final report.

Evaluation. The goal is to assess the correctness of the methods application and the appropriate-ness of the security requirements identified. Method designers evaluate whether the participants have followed the method. Customers, instead, assess if the groups have identified a set of security requirements or countermeasures that are specific for the application scenario, and if they are able to justify the security requirements based on the method’s application.

Five type of actors are necessary to successfully execute this protocol: Method Designer, Customer, Observer, Researcher, and Participant. Method designers and customers contribute to all steps of the protocol. The method designer is the researcher who has proposed one of the method under evaluation. His main responsibility is to train participants in the method and to answer participants’ questions during the Application phase. S/he also contributes to the assessment of the methods’effectiveness by analyzing groups’ reports. The customer is an industrial partner who introduces the industrial application scenario to the participants. He also has to be available during the Application phase to answers all possible questions that participants may raise during analysis. Observers play an important role during the Application phase because they supplement audio-video recording with information about the behavior of participants e.g (if the Participants work in group vs work alone) and the difficulties that they face during the application of the method. The observer also interviews the groups and leads the post-it notes sessions. The researcher takes care of the organization, sets the research questions, selects the participants, invites the method designers and the customers, and analyzes the data collected during eRISE.
3.2.1 Participants

They work in group and apply a method provided by one of the method designers to analyze the risk and security issues of the scenario provided by the customer. We have recruited two types of participants: master students and professionals. The main reason behind this choice is that having only students is known to be a major threat to external validity [8]. Students were enrolled in the Master in Computer Science at the University of Trento and had a background in Security Engineering and Information Systems. Practitioners were attending a Master Course in Audit for Information System in Enterprises at Dauphine University. This master has an admission requirement of a minimum of five years of working experience in the field of Auditing in Information Systems. Thus, eRISE 2012 has involved twenty-seven professionals and fifteen students.

3.2.2 Selection of Methods

The selection of the security requirements methods to be evaluated was driven by three main factors: the number of citations, the fact that research on the method is still ongoing, and availability of the methods designers.

We had invited a number of research groups to join the activity. Out of the various oral and email invitations being sent, only six groups accepted to be involved in eRISE.

CORAS is a model-driven method for risk analysis proposed by SINTEF [5]. LINDDUN [1] is a methodology to elicit the privacy requirements of software-intensive systems and select privacy enhancing technologies designed by DistriNet Research Group at Katholieke Universiteit Leuven. SECURITY ARGUMENTATION [3] is a framework for security requirements elicitation and analysis developed at Open University. SECURE TROPOS [7] is a methodology designed at University of East London; the methodology supports capturing, analysis and reasoning of security requirements from the early stages of the development process. SREP [6] is an asset-based and risk-driven method developed at University of Castilla-La Mancha for the establishment of security requirements in the development of secure Information Systems. SI* [2] is a formal framework developed at the University of Trento for modeling and analyzing security requirements of an organization.

CORAS, SECURITY ARGUMENTATION and SECURE TROPOS have been evaluated in both editions of eRISE. SI* was evaluated only in eRISE 2011, because it is proposed by the same research group who organized eRISE, and the researchers wanted to avoid bias in the assessment of the methods. LINDDUN and SREP have been only assessed in eRISE 2012.

3.2.3 Application Scenarios

For eRISE 2012 we have used the NESSoS application scenarios on Smart Grid and eHealth. Smart Grid scenario is about an electricity network using information and communication technology (ICT) to optimize the transmission and distribution of electricity from suppliers to consumers. In particular, the scenario was focused on the smart meter which records consumption of electric energy and communicates this information daily back to the utility for monitoring and billing purposes.

The eHealth scenario focused on registering new patients in a clinic including assigning the clinicians (doctors, nurses, etc.), reading and updating a record, retrieving patient information from external sources, and providing the results of examinations and treatment to authorized externals clinical entities.

3.3 Data Analysis

In this section we report on the results of the analysis conducted on the transcripts of focus group interviews, the post-it notes and the questionnaires.

3.3.1 Interview Transcripts’ Analysis

The transcribed focused groups were analyzed using coding, a qualitative analysis method used in grounded theory. The analysis was conducted by three researchers from UNITN according to the following steps:
1. **Open coding.** Each researcher had analyzed sentence by sentence the transcripts to identify codes that represent text patterns related to the methods’ effectiveness.

2. **Focused coding.** The researchers worked in groups to compare and cluster the codes for identification of categories.

3. **Theoretical coding.** The researchers identified the core categories and the relationship between them.

A first hypothesis which emerges from the analysis is that the evaluated methods help brainstorming because they provide a good way to model the domain but do not provide guidance to identify security requirements or risks. Only if the user who applies the method has a background in security, he/she can identify the security requirements for the application domain. From the analysis, it also emerged a second hypothesis related to how an effective method in eliciting security requirements and risks should be: a good method is a method which requires an analyst only to acquire knowledge in the domain and follow the process to obtain specific/valuable security requirements.

The hypotheses are grounded on three main categories that are related to what the methods provide, what the methods do not provide, and what is required to apply them: **Mindmapping**, **Identifying Security Requirements/Risks**, and **Knowledge**.

**Mindmapping**

Participants found that the methods help them in brainstorming on the main risks threatening the application scenario:

"for me it helps to put all the brainstorming in the diagrams. CORAS helps to organize the ideas in the mind, by using the diagrams. it helps to realize what you are thinking", CORAS, Professional.

"its just a tool to represent what I know. its a tool to represent my knowledge and my experience. all I need is the target from client, and rest of the stuff is me. the tool doesn’t help me in that. the method gives me a representation of what I think", CORAS, Student.

"I think the method is good for mind mapping. as long as you have the nice ideas, you can keep them for the solutions", SECURE TROPOS, Professional.

"the fact that you have to identify the actors, the objectives, the templates help us to have a clear idea of what is the problem of our use case... its a good way to mindmap the use case", SECURE TROPOS, Professional.

"but its just a tool like UML. to understand the system, you need to modelize using UML, its exactly the same comparison. For me UML is not a method", SECURE TROPOS, Professional.

**Identifying Security Requirements/Risks**

One of the major concerns reported by participants is that methods do not provide any guidelines or support to identify security requirements or risks. Thus, there is a contradiction between what the participants reported and the goal that the methods state to accomplish.

So a natural question that arise is: **Do these methods for eliciting risk and security requirements really deliver what they promise?** Based on participants’ feedbacks, it seems CORAS, SECURE TROPOS, SECURITY ARGUMENTATION and SI* do not deliver what they promise.

"I don’t think the method helped me to identify. I read the case study and thought about the risks", CORAS, Professional.

"the method doesn’t provide details on which risks to focus on", CORAS, Professional.

"it doesn’t tell me this is a risk, I decide this is a risk.", CORAS, Student.

"the objective is to identify security recommendations, the tool doesn’t help in this directly", SECURE TROPOS, Professional.

"but it is not a method to find security recommendations...it helps us to represent the model but does not help in finding solutions", SECURE TROPOS, Professional.

"I was expecting the tool to help us find the security recommendations, but after using the tool, I came to know that the tool just helps to model the use case. but the tool did not directly help us to find some security recommendations. ", SECURE TROPOS, Professional.

"method does not give good recommendations, and it’s not clear how to get there", SECURITY ARGUMENTATION, Student.

"difficult to identify the requirements and if they are missing it will be difficult to identify it", SECURITY ARGUMENTATION, Student.
“the method dint help, it was them finding out the requirements”, SI*,Student.

“[Observer: following the methodology, where you able to find the requirements?] No, it was based on our experience. Also based on other methods like COBIT”, SI*, Professional.

“[Observer: How much did the method help you in this?] As I said we didn’t focus much on SI*, we did it the expert way”, SI*, Student.

In contrast, participants acknowledged that SREP and LINDDUN help to identify security and privacy requirements respectively.

“It helps also in making security requirement that tell how to resolve, solve the problems. It helps to know what are the problems, and conclude how to solve them”, SREP, Professional.

“it helps to find out specific security requirement. at the beginning when you have to define the different assets, you imagine what kind of threats”, SREP, Professional.

“because we start with the general objectives, and then we need to go into every single detail of threat and security and the risks. so I think the method allows you to come to the specific security requirement”, SREP, Student.

“About the threat tree pattern, it is useful because it makes you think about the threats”, LINDDUN, Professional.

“if the company needs to work on the privacy of their information, they should use this method because the method steps helps to ensure safety of their data”, LINDDUN, Professional.

“the method that I use officially, it never helped to identify assets like anonymity, privacy affects security, so it would be good if other methods think about anonymity”, LINDDUN, Professional.

Knowledge

Since some of the methods do not provide support to elicit risks and security requirements, the analysts have to rely on their knowledge in security and in the application domain to identify security requirements.

“I think that either the direct representative of the customer or the one who has good subject knowledge should directly interact with us. We can clear our confusion”, CORAS, Student.

“yes, so if you get the right person with the right experience then maybe you get the right diagram”, CORAS, Student.

“The language is quite easy to understand but again we need an expert. The tool is also easy to understand but we need more knowledge. The application of the method and tool was quite complicated because of the lack of experience”, CORAS, Professional.

“if I have deep knowledge about the risks and treatment, I can put them into the diagram”, CORAS, Student.

“identification of the constraints, it also depends on the experience of the user”, SECURE TROPOS, Student.

“You already need to have the knowledge to apply the tool”, SECURE TROPOS, Professional.

“yeah because you have to add the knowledge to add security recommendations, but if you don’t have the knowledge the method is not really helpful”, SECURE TROPOS, Professional.

“I used to manage some requirements like this for microsoft. If I compare the two interface, I think its easier to modelize using visio or UML than in this, because here you need to have some technical experience”, SECURITY ARGUMENTATION, Professional.

“we did not have information about security. not like in COBIT where you just need to read the material and follow the steps. But here we need to know about security and what we are looking for”, SECURITY ARGUMENTATION, Professional.

“No, it was based on our experience. Also based on other methods like COBIT, SI*, Professional.

“yes, we prioritize our recommendations, maybe from our experience. Some points seemed more critical than others, SI*, Professional.

“outcome based on experience, SI*, All Participants.

The main finding resulting from coding is that CORAS, SECURE TROPOS, SECURITY ARGUMENTATION and SI* support brainstorming but they do not help to identify security requirements. The identification of security requirements depends only on the analyst’s knowledge in security and in the application domain. On the contrary, SREP and LINDDUN do not require the analyst to have a previous knowledge because the analyst is guided through the identification of security/privacy requirements.

3.3.2 Post-it notes’ Analysis

Affinity analysis had allowed us to identify the aspects, which influence the methods’ effectiveness. The analysis was conducted by the participants during a dedicated session of the duration of an hour on the last day of the Application phase. For each method, the participants were divided in two groups. The composition of each group was such that the participants who had worked together during the Application phase were not in the same group.

Then, each group performed the following activities:
Table 3.1: Methods’ positive aspects resulting from Affinity Analysis

<table>
<thead>
<tr>
<th>Method</th>
<th>Overall</th>
<th>Modeling Language</th>
<th>Process</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>CORAS</td>
<td>Brainstorming method</td>
<td>Easy to understand</td>
<td>Detailed Process</td>
<td>Good Interface</td>
</tr>
<tr>
<td></td>
<td>Structured method with guidelines</td>
<td>Diagrams summarize risks</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LINDDUN</td>
<td>Easy to Apply</td>
<td>DFD is easy to understand</td>
<td>Clear and easy to follow</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Focused on usually neglected privacy aspects</td>
<td>Easy to Use</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SECURE TROPOS</td>
<td>Method to model a scenario</td>
<td>Support Mindmapping</td>
<td>Actors identification</td>
<td>Good Interface</td>
</tr>
<tr>
<td></td>
<td>Intuitive</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SECURITY ARGUMENTATION</td>
<td>Easy to understand</td>
<td>Formal Argumentation</td>
<td>Graphical support to create argumentation diagrams</td>
<td></td>
</tr>
<tr>
<td>SI</td>
<td>Help Brainstorming</td>
<td>Easy to understand</td>
<td>Easy to find functional requirements</td>
<td>Easy to draw diagrams</td>
</tr>
<tr>
<td>SREP</td>
<td>Easy to Understand</td>
<td>Easy to understand</td>
<td>Easy to Understand</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Easy to Apply</td>
<td>Familiar Vocabulary</td>
<td>Easy to Argument</td>
<td></td>
</tr>
</tbody>
</table>

Legend: For SREP and LINDDUN there are no aspects listed for the tool because the methods have no tool support.

Table 3.2: Methods’ Limitations resulting from Affinity Analysis

<table>
<thead>
<tr>
<th>Method</th>
<th>Overall</th>
<th>Modeling Language</th>
<th>Process</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>CORAS</td>
<td>Not applicable in real life</td>
<td>Visual Notation Tend to Be Complex</td>
<td>Risk Estimation</td>
<td>Bugs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Complex systems are not representable</td>
<td>Long Process</td>
<td>Modeling is difficult</td>
</tr>
<tr>
<td>LINDDUN</td>
<td>Some concepts are not clear</td>
<td>No concepts to model risks</td>
<td>Difficult to Assign Priority to Threats</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Missing major concepts like actors</td>
<td>Not expressive language</td>
<td>Long Process</td>
<td></td>
</tr>
<tr>
<td>SECURE TROPOS</td>
<td>No help to find security requirements</td>
<td>Some concepts are not clear</td>
<td>Process Not Well Defined</td>
<td>Not working</td>
</tr>
<tr>
<td></td>
<td>Not a security method</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SECURITY ARGUMENTATION</td>
<td>Hard to Apply</td>
<td>Hard to Understand</td>
<td>Difficult to identify functional requirements</td>
<td>Argumentation diagram creation is buggy</td>
</tr>
<tr>
<td>SI</td>
<td>Hard to Apply</td>
<td>Semantic of concepts not clear</td>
<td>Difficult to identify the risks</td>
<td>Buggy</td>
</tr>
<tr>
<td>SREP</td>
<td>No Tool Support</td>
<td>A Lot of Terminologies</td>
<td>Long Process</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Risk Assessment Procedure Not Clear</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend: For SREP and LINDDUN there are no aspects listed for the tool because the methods have no tool support.

- **Post-it Notes.** Each member of the group had been requested to annotate on post-it notes, a positive and a negative aspect on the method as a whole, the modeling language, the process, and the tool supported by the method.

- **Post-it Notes Grouping.** The participants hanged the post-it notes on a board, and grouped the post-it notes which report similar opinions about the aspects (process, modeling language, tool, overall impression) of the method. Each group of similar opinions represents a category.

- **Post-it Notes Prioritization.** The participants listed the categories in order of importance.

The categories with highest priority resulting from the analysis are illustrated in Table 3.1 and Table 3.2. We summarize the main findings in what follows.

**CORAS**  CORAS is considered a well-structured method because the steps of the process are very well detailed. The modeling language is also easy to understand and the diagrams give an overview of the possible risks. However, participants were concerned about the visual notation which does not scale well for complex scenarios. The participants have also some concerns about the process, which consists of too many steps and suggested that some of the steps should be merged. They also reported lack of guidelines to define the likelihood and consequence scales. The user interface of the tool is user-friendly, but the modeling feature has a lot of bugs.
### Table 3.3: Answers on Security Requirements' Completeness

<table>
<thead>
<tr>
<th>Method</th>
<th>Disagree (1-2)</th>
<th>Neutral (3)</th>
<th>Agree (4-5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CORAS</td>
<td>8</td>
<td>2</td>
<td>7</td>
</tr>
<tr>
<td>LINDDUN, SREP</td>
<td>2</td>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>SECURE TROPOS, SECURITY ARGUMENTATION, SI*</td>
<td>22</td>
<td>11</td>
<td>8</td>
</tr>
</tbody>
</table>

**LINDDUN**  
LINDDUN is a method easy to understand and to apply because the process to identify privacy requirements is clearly defined and easy to follow. However, participants believed the process is too long and found difficult to identify the privacy threats and to prioritize them. Regarding the modeling language, the participants reported that the semantic of some concepts is not clear, and that concepts to model risk are missing. However, they appreciated the data flow diagrams because they allow to define the scope of the analysis.

**SECURE TROPOS**  
The positive aspect of SECURE TROPOS is that the diagrams support mindmapping because they are a good way to model the scenario. The concerns are related to the process to identify security requirements which is not well defined, and to the tool which has several bugs especially related to the creation of diagrams.

**SECURITY ARGUMENTATION**  
The most appreciated aspect of SECURITY ARGUMENTATION is argumentation analysis. About the tool, participants liked the feature to create argumentation diagrams, which automatically creates the textual formalization of the arguments. Participants liked this feature of the tool because they did not have to learn the formal language that was considered too technical. Regarding the process, the participants reported that argumentation analysis step is very detailed compared to the other steps for which no guidelines are provided on how they have to be performed.

**SI**  
About SI*, participants liked the models because they give a good visual representation of all the dependencies between actors in an organization. They also find it very easy to identify the functional requirements. However, the participants are concerned about the semantics of some concepts that is not clearly defined. There are also several concerns on the risk analysis step of the method and on the tool that is full of bugs.

**SREP**  
SREP has been greatly appreciated by the participants because it is a method easy to understand and apply. The reasons are that the modeling language consists of concepts easy to understand even for non-expert users and that the process' steps are detailed. The participants suggested that more guidelines on how to conduct the risk assessment step are needed, and that the execution of the process would benefit from tool support.

### 3.3.3 Questionnaires' Analysis

The participants were administered five questionnaires during the execution of eRISE. The first one (Q1) was administered at the beginning of the Training phase to collect participants' background. The second questionnaire (Q2) was distributed at the end of the Training phase. The third (Q3) and fourth (Q4) questionnaires were administered during the two Application phases. The last one (Q5) was administered at the end of the Application phase to compare the method applied by the participants with other methods they may already knew.

#### Quantitative Analysis

To find evidence of our hypothesis on methods' effectiveness we have analyzed the answers given by the participants to the question "The results of your analysis are complete" from Q3 and Q4. The questions is about the opinion of the participants on the completeness of the requirements identified by applying the method. The participants had to specify a value in a five-item Likert scale: Strongly disagree (1), Disagree (2), Neutral (3), Agree (4), Strongly agree (5). Table 3.3 reports for the methods the frequency of
Table 3.4: Answers on Method’s Positive and Negative Aspects

<table>
<thead>
<tr>
<th>Question</th>
<th>CORAS</th>
<th>LINDDUN</th>
<th>SECURITY ARGUMENTATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q2.1 What's your overall impression on the method? (1-10 scale)</td>
<td>6,53</td>
<td>5</td>
<td>3,2</td>
</tr>
<tr>
<td>Q2.2 What do you like the most about the method?</td>
<td>Easy, Modeling</td>
<td>Easy</td>
<td>Argumentation</td>
</tr>
<tr>
<td>Q2.3 What do you like the least about the method?</td>
<td>Complex Diagrams, Tool support</td>
<td>Use of Assumptions</td>
<td>Hard, Graphical Notation</td>
</tr>
<tr>
<td>Q3.1 What's your overall impression on the method? (1-10 scale)</td>
<td>6,1</td>
<td>6,1</td>
<td>4,3</td>
</tr>
<tr>
<td>Q3.2 What do you like the most about the method?</td>
<td>Detailed Process</td>
<td>Easy, DFD, Detailed Process, Threat tree pattern, Privacy specific</td>
<td></td>
</tr>
<tr>
<td>Q3.3 What do you like the least about the method?</td>
<td>Consequence Scale Definition</td>
<td>Use of Assumptions, Hard to Link Privacy Threats</td>
<td>Technical Application</td>
</tr>
<tr>
<td>Q3.12 Your experience with the tool was (1-5 scale)</td>
<td>3,38</td>
<td></td>
<td>2,88</td>
</tr>
<tr>
<td>Q3.13 Which features of the tool did you like most?</td>
<td>Tool Interface, Modeling</td>
<td></td>
<td>Automatic Generation of Arguments from Graphical Representation</td>
</tr>
<tr>
<td>Q3.14 Which features of the tool did you like least?</td>
<td>Lack of traceability between diagrams</td>
<td></td>
<td>Bugs</td>
</tr>
<tr>
<td>Q4.1 What's your overall impression on the method? (1-10 scale)</td>
<td>5,8</td>
<td>6,5</td>
<td>4,3</td>
</tr>
<tr>
<td>Q4.2 What do you like the most about the method?</td>
<td>Detailed Process, Diagrams</td>
<td>DFD</td>
<td>Modeling, Argumentation</td>
</tr>
<tr>
<td>Q4.3 What do you like the least about the method?</td>
<td>Unnecessary steps</td>
<td>Threat tree pattern identification; Unclear concepts</td>
<td>Hard to identify functional requirements</td>
</tr>
<tr>
<td>Q4.12 Your experience with the tool was (1-5 scale)</td>
<td>3</td>
<td></td>
<td>2,5</td>
</tr>
<tr>
<td>Q4.13 Which features of the tool did you like most?</td>
<td>Visualization</td>
<td></td>
<td>Generation of code by drawing diagrams</td>
</tr>
<tr>
<td>Q4.14 Which features of the tool did you like least?</td>
<td>Bugs</td>
<td></td>
<td>Bugs</td>
</tr>
</tbody>
</table>

Legend: For questions Q2.1, Q3.1, Q3.12, Q4.1, and Q4.4 we report the mean score, for the other questions we report the most frequent answers given by the participants. For LINDDUN there are no answers to questions Q3.12, Q3.13, Q3.14, Q4.12, Q4.13, and Q4.14 because there is no tool support.

The answers confirm our hypothesis on methods' effectiveness. SREP and LINDDUN are the methods which are effective in identifying security/privacy requirements since the number of answers “Strongly agree” is double the number of the answers “Strongly disagree”. For SECURE TROPOS, SECURITY ARGUMENTATION, and SI* the proportion between the “Strongly agree” and the “Strongly disagree” answers is one to three. This confirms the negative opinions of the participants on the effectiveness of these methods expressed during the focus groups interviews. For CORAS the number of answers “Strongly agree” and the “Strongly disagree” is almost equal attesting that CORAS is a well-structured method, but with space for improvements.

We have assessed the statistical significance of the data reported in Table 3.3 using Fisher exact test which is suitable for small data samples. We use 0.05 as the significance level. The resulting p-value of the test is 0.04408 which is lower that 0.05. Thus, we can conclude that our data are statistically significant.

Qualitative Analysis

We have analyzed the answers relevant to understand which aspects influence methods’ effectiveness. In Table 3.5 we report the questions and answers from Q2, Q3, and Q4 related to strengths and limitations of the methods. We summarize the results in what follows.

Overall Appreciation The answers to questions Q2.1, Q3.1, and Q4.1 on the overall appreciation of the methods attest that SREP and LINDDUN have been the methods most appreciated by the participants. For CORAS, SECURITY ARGUMENTATION and SI* the level of appreciation of the method decreases after the participants have applied them. Instead, the more the participants have applied LINDDUN the more, they have appreciated it. The opinion of the participants on SREP and SECURE TROPOS did not
Table 3.5: Answers on Method’s Positive and Negative Aspects

<table>
<thead>
<tr>
<th>Question</th>
<th>SECURE TROPOS</th>
<th>SI*</th>
<th>SREP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q2.1 What’s your overall impression on the method? (1-10 scale)</td>
<td>5.8</td>
<td>6</td>
<td>7.25</td>
</tr>
<tr>
<td>Q2.2 What do you like the most about the method?</td>
<td>Process, Tool Support</td>
<td>Modeling</td>
<td>Easy</td>
</tr>
<tr>
<td>Q2.3 What do you like the least about the method?</td>
<td>Dependencies</td>
<td>Hard</td>
<td>Risk Assessment</td>
</tr>
<tr>
<td>Q3.1 What’s your overall impression on the method? (1-10 scale)</td>
<td>5.8</td>
<td>4.9</td>
<td>7.5</td>
</tr>
<tr>
<td>Q3.2 What do you like the most about the method?</td>
<td>Modeling</td>
<td></td>
<td>Security Requirements</td>
</tr>
<tr>
<td>Q3.3 What do you like the least about the method?</td>
<td>Long Process, Tool Usability</td>
<td>First two steps not well defined</td>
<td></td>
</tr>
<tr>
<td>Q3.12 Your experience with the tool was (1-5 scale)</td>
<td>2.43</td>
<td>4.13</td>
<td></td>
</tr>
<tr>
<td>Q3.13 Which features of the tool did you like most?</td>
<td>User Interface</td>
<td>Modeling</td>
<td></td>
</tr>
<tr>
<td>Q3.14 Which features of the tool did you like least?</td>
<td>Not Stable</td>
<td>Hard to install</td>
<td></td>
</tr>
<tr>
<td>Q4.1 What’s your overall impression on the method? (1-10 scale)</td>
<td>5.3</td>
<td>4.9</td>
<td>7.5</td>
</tr>
<tr>
<td>Q4.2 What do you like the most about the method?</td>
<td>Simple concepts</td>
<td>Easy</td>
<td></td>
</tr>
<tr>
<td>Q4.3 What do you like the least about the method?</td>
<td>Tool support, Lack of guidelines</td>
<td>Risk Assessment</td>
<td></td>
</tr>
<tr>
<td>Q4.12 Your experience with the tool was (1-5 scale)</td>
<td>1.8</td>
<td>2.44</td>
<td></td>
</tr>
<tr>
<td>Q4.13 Which features of the tool did you like most?</td>
<td>User Interface, Modeling</td>
<td>Easy to Draw Diagrams</td>
<td></td>
</tr>
<tr>
<td>Q4.14 Which features of the tool did you like least?</td>
<td>Bugs</td>
<td>Bugs</td>
<td></td>
</tr>
</tbody>
</table>

Legend: For questions Q2.1, Q3.1, Q3.12, Q4.1, and Q4.4 we report the mean score, for the other questions we report the most frequent answers given by the participants. For SREP there are no answers to questions Q3.12, Q3.13, Q3.14, Q4.12, Q4.13, and Q4.14 because there is no tool support. For SI* there are no answers to questions Q3.2, Q3.3, Q4.2, and Q4.3 because SI* was evaluated only during eRISE 2011.

change among the different phases of eRISE.

Positive and Negative Aspects In the answers to questions Q2.2, Q2.3, Q3.2, Q3.3, Q4.2, Q4.3 on the methods’ aspects appreciated the least or the most, the participants reported the same aspects they have listed on the post-it notes.

CORAS has two key aspects: a detailed process to conduct risk analysis and the diagrams which provide a good overview of the risks. The process has a problematic step that is the definition of likelihood and consequence scales for assets. The tool requires further development because it has a lot of bugs.

LINDDUN is a method easy to apply because it has a detailed process to identify privacy requirements. Moreover, the process is based on data flow diagrams which give an overview of the scope of the analysis. However, the participants criticize the use of trust assumptions to limit the number of threats. They also reported that it was difficult to map the privacy threats to the data flow diagrams.

The aspects of SECURITY ARGUMENTATION which have been more appreciated by the participants are argumentation analysis. They also liked the feature of the tool which allows to draw argumentation diagrams and automatically generate the textual formalization. The aspect liked the least was the lack of support to identify functional requirements.

Participants’ main concerns on SECURE TROPOS are about tool support, the length of the process and the lack of guidelines on how to proceed. The models were appreciated because they support brainstorming.

SREP is appreciated because it is easy to master and makes easier the identification of security requirements. Guidelines on how to conduct risk assessment step should be given.

Tool support About the tool support, none of the participants reported a positive experience with the tools. For CORAS and SECURITY ARGUMENTATION the opinion of the participants did not change between the application phases, while for SECURE TROPOS and SI* it gets worse. A common concern
for the tools is the presence of a lot of bugs.

3.3.4 Data Triangulation: Reports’ Evaluation and Participants’ Background

In order to validate our hypothesis that to identify security requirements the participants had to use their expertise, we have correlated the assessment of the reports done by the method designers and customers with the background of the members of the groups who have delivered the reports.

In what follows, we first summarize for each method the results of the evaluation and the background of the groups. Then, we draw our conclusions.

CORAS   None of the groups was able to complete all the steps of the CORAS process, but according to the customers all groups came up with specific security requirements for the application scenarios. In all the groups at least one participant had an intermediate knowledge in security.

LINDDUN   One group was able to propose a set of privacy requirements specific for the application scenario even though the members of the group did not go through all the steps of the LINDDUN methodology and they were novices to security and privacy. Another group fully applied the method and came up with a set of privacy requirements specific to the application scenario. The group was composed of two members with low knowledge in privacy, and one member with good knowledge in security and requirements engineering. The third group applied all the steps of the methodology but identified a set of generic privacy requirements. The group was composed by two members: one of the two had good expertise in security, privacy and requirements engineering.

SECURE TROPOS   Only one group was able to fully apply the method and to propose a set of specific security requirements for the scenario even though the members had low expertise in security. Of the other two groups, one produced generic requirements while the other specific requirements without applying the method. In both groups, at least a member had a good knowledge in requirement engineering and intermediate knowledge in security.

SECURITY ARGUMENTATION   All the groups came up with a specific set of security requirements for the application scenario. However, the method designer reported that only one group was able to apply all the steps of the process. The customers pointed out that for all the groups is not clear how the security requirements proposed for the scenario have been identified based on the method application. The group composition was such that at least one member had intermediate or good expertise in security and intermediate expertise in requirements engineering.

SI*   One group had applied the method and have identified only generic requirements. Most of the participants had intermediate level of knowledge in COBIT. Another group came up with specific recommendations but they have used COBIT instead of SI* as attested by the transcripts of the audio-recordings of the Application phase. One member of the group had eleven years of work experience in IT audit and he was an expert in COBIT. The third group came up with specific requirements but from the report it is not clear which method they have followed. The comments given during the focus group interviews revealed that they used their experience to come up with the results and not the SI* method. One group member had seventeen years of work experience in IT audit.

SREP   One group was able to identify a valuable set of security requirements for the scenario which was clearly identified based on the application of the method. In this group, all the members were novices to security, privacy, and requirements engineering. Another group had applied all the steps of the method and had identified only a set of generic security requirements. One of the group members had good knowledge in requirements engineering while another member had intermediate knowledge in security. The third group partially applied the method and came up with a set of security requirements that address specific security problems for the application scenario. In this case, the group had a member with intermediate knowledge in security, privacy and requirements engineering.
What emerges from the triangulation of the evaluation of the reports with the background of the members of the groups is that CORAS, SECURE TROPOS, SECURITY ARGUMENTATION and SI* have been partially applied or not applied at all, but the groups who have applied them have proposed security requirements specific for the application scenario under analysis. If groups were not able to go through all the steps of the process supported by the method, this means that to identify security requirements the groups had to use their knowledge in security or applied other security methods e.g COBIT. This supports our hypothesis that these methods do not help to identify security requirements and that to apply them the analysts need to have at least an intermediate knowledge in security. Instead, SREP and LINDDUN are effective in eliciting security/privacy requirements because even groups with low or no expertise in security or privacy came up with a set of specific requirements for the application scenario. They can thus be considered as a reference model of an effective method for eliciting and analyzing security requirements.

3.4 Threats to Validity

• **Construct Validity.** The main threat to construct validity in eRISE regards the design of the research instruments: our main measurement instruments are interviews and questionnaires. Three researchers have checked the questions included in the interview guide and in the questionnaires; therefore we believe that our research instruments measure what we want to measure. Moreover, to reduce this threat we have gathered data using other data sources (audio-video files, post-it notes, and participants’ reports) and performed data triangulation.

• **Internal Validity.** An important threat to internal validity is that one of the methods under evaluation - SI* - was designed by UNITN who organized eRISE. In this case, the training and the assessment of methods’ effectiveness could have been biased because the researcher may have presented SI* more positively than the other methods or manipulated the data on SI*. To avoid this threat, the tutorials on the other methods were given by the methods’ own inventors. Moreover, in eRISE 2012 we did not include SI* among the methods to be evaluated.

Another threat to internal validity is that the time spent in training participants was too short for them to apply the method and understand the application scenarios. To mitigate this threat, method designers and customers were available to answer questions that participants may raise during the application of the methods.

One additional threat is represented by participants’ previous knowledge of other methods. For example, in one group it was decided to use COBIT to identify the security requirements for the application scenario, rather than the method assigned to the group. In this case, the feedbacks provided by the group had no value because the method was not applied.

• **External Validity.** We have evaluated the effectiveness of the security requirements and risk analysis methods with both master students and professional from different countries with different background, and we have applied the methods in different contexts - Smart Grid and Healthcare. This give us some confidence that our hypothesis and conclusions on methods’ effectiveness have a medium degree of generalizability.

• **Conclusion Validity.** An important threat to the conclusion validity of eRISE is that our sample is relatively small in statistical terms. In fact, for SI*, SREP, and LINDDUN which have been evaluated only during one of the two eRISE edition, the size of the sample is of 10-12 participants. In order to increase the statistical significance of the emerging hypothesis and insights on methods’ effectiveness, we are going to conduct the third edition of eRISE to have a bigger data sample.

Another threat to conclusion validity is related to the correctness of the method application and of the security requirements elicited by the participants. We have limited this threat by requesting to the method designers and the customers to evaluate the reports documenting the methods' application.
3.5 Conclusions

This chapter reported on eRISE 2012 and the results of the analysis of the data collected during eRISE 2011 and 2012. Six academic methods - CORAS, LINDDUN, SECURE TROPOS, SECURITY ARGUMENTATION, SI* and SREP- have been evaluated and compared during the two editions of eRISE.

eRISE underlines that there is a gap between what the methods really provide and what they claim to accomplish. An hypothesis which emerges from the analysis of the interviews with the groups of participants is that CORAS, SECURE TROPOS, SECURITY ARGUMENTATION and SI* are methods to support brainstorming because they provide a good way to model the system under analysis. They are not methods for eliciting security requirements because the identification of security requirements is left to the analyst's knowledge in security and in the application domain. On the contrary, SREP and LINDDUN do not require the analyst to have a previous knowledge because the process guides the analyst to the identification of valuable security/privacy requirements.

In addition eRISE clearly shows with both qualitative and statistical means that being effective in supporting brainstorming is not enough to elicit specialized security requirements. Concrete steps in the process should be designed to identify concrete security requirements. If the method has to rely on the security expertise of the participant, then it has no advantage over a generic method for requirements elicitation. In the eyes of the practitioners, this generates the impression that “the method does not work” and hinderance its adoption in practice.

The execution of the two editions of eRISE also give us the opportunity to identify several ways to improve the evaluation protocol.

- **Train Observers.** Interviews conducted by the observers with groups of participants are one of our main research instruments. Interviewing groups is a challenging task which requires practice and experience. It is important to organize pilot studies with academic or personal groups so that the observers have the opportunity to practice in advance.

- **Clarify what you expect from participants.** it is important to clarify what the participants are expected to deliver at the beginning of each phase of eRISE. In the first run, a couple of groups thought that identifying a set of security requirements for the scenario was more important than using the method. Thus, when the assigned method failed to deliver, they switched to a practitioner’s method they already knew. We avoided this problem in the second run.

- **Have method designers and customers present during the Application phase.** The presence of the method designer and customer during the Application phase showed to be important because it allows participants to ask for additional information that may have not been provided during training due to the available short time.

- **Have a translator present.** The participants of eRISE were of mixed nationalities and not all of them had good English language skills. Thus, it might be useful to have a translator present during focus groups sessions to avoid misinterpretation of participants’ feedbacks or worse to loose feedbacks because participants feel not confident to speak in English.

Overall, we believe that our work is a first step to build an empirical framework which helps to compare and select security requirements methods, help researchers to increase the effectiveness of their research methods, and motivate practitioners to use new techniques.
4 War Games

One of the ways that NESSoS is promoting itself is by organizing open competitions and have as many people as possible interact in them. These events have proven to be very successful in the past, and seemed to be ideal to build an environment for web security experts to meet and interact with each other. We have opted to build a permanent open competition in the form of a game that security experts can play where they have to hack into a website in order to win. These types of games attract a lot of attention of professional and non-professional security experts, and by branding the game as part of the NESSoS NoE, we manage to attract people with the profiles we are interested in to NESSoS. The war game we built is aimed at helping people understand the intricacies of web server security. The game is composed of different levels. Each player starts at the same level and they have to work their way up towards more difficult levels. A level can be completed by successfully exploiting a vulnerability in the system and retrieving the credentials to go to a next level. The first levels focus on easily exploitable vulnerabilities where more background information about the underlying systems is available. The player gradually gets less information needed to break the system, and the exploits become harder. In the most advanced levels, a number of exploits must be combined to finish a level. These types of games offer valuable training experience, and this is what makes them so popular among security experts and hackers alike. The War Game is available online at the location http://www.overthewire.org/wargames/natas/ on the website overthewire.org, which hosts a number of other war games like this. The added value for us is that this site is already frequented by people with a computer security profile, so by hosting the NESSoS war game there, we can immediately reach out to these people.

The NESSoS War Game, named NATAS, currently consists of 17 levels ranked from easy to difficult. Players start at the easiest level, level 1, and work their way up to the most difficult level. They cannot choose where to start. Hence, in order to reach level 17, all other levels must be completed first. The goal is to complete all the levels of the game. In order to do so, the players must exhibit a strong and diverse technical knowledge.

Players interested in contributing to the war game can do so. The game is built in such a way that it is easy to insert new levels. The following is the list of levels that are present in NATAS, and the vulnerability that players need to exploit to complete the level:

- Level 1: The password is hidden in the webpage as a comment
- Level 2: The password is hidden in the webpage as a comment, but right-clicking has been disabled (JavaScript)
- Level 3: Contains a directory traversal vulnerability; a backup of the password is located in the directory autoindex
- Level 4: Robots.txt contains a hidden directory that contains a backup of the password.
- Level 5: User gets access to the password if the referrer is set to a specific value.
- Level 6: Contains a client-side cookie manipulation
- Level 7: The PHP source is visible; username and password are included from a .inc file
- Level 8: The PHP source is visible; a local file inclusion is possible
- Level 9: The PHP source is visible; plaintext password is not visible, but an encoded version is.
- Level 10: The PHP source is visible; a shell command injection is possible.
- Level 11: The PHP source is visible; a shell command injection (with a blacklist of prohibited characters) is possible.
- Level 12: The PHP source is visible; it contains a cookie that is encrypted with XOR and an unknown key. A plaintext attack is possible.
- Level 13: The PHP source is visible; it contains a file upload vulnerability
• Level 14: The PHP source is visible; it contains a file upload vulnerability but only files with valid EXIF data are accepted.

• Level 15: The PHP source is visible; contains a simple SQL injection vulnerability

• Level 16: The PHP source is visible; contains a SQL injection vulnerability where one bit can be leaked

• Level 17: The PHP source is visible; contains a shell command injection vulnerability with an extensive blacklist of prohibited characters.

The NESSoS war game was well received by the community, having 4500 players in the first week and being picked up by Forbes¹.

¹http://www.forbes.com/sites/andygreenberg/2012/10/29/hurricane-bound-hacker-heres-a-rainy-day-web-hacking-war-game/
5 Conclusions

In this deliverable we have reported the education and training activities conducted in WP13: the FOSAD summer school, eRISE 2012 and War Games competition. FOSAD 2012 has contributed to enhance the background in Security Engineering of 40 PhD students and young researchers by offering lectures on different topics given by recognized international experts in the field. eRISE 2012 has involved 43 master students and practitioners who had the opportunity to learn new security methods and who had contributed with their evaluation of the methods to the realization of an empirical ground on methods’ effectiveness. The empirical ground will play an important role both for researcher and practitioners: for researchers will be a basis to improve their own methods and for practitioners will serve as basis to select new security methods. War Games is paving the way for the realization of new kind of competitions that allow web security experts to meet and learn about web applications’ vulnerabilities.

During the last period of NESSoS, WP13 will continue to promote education and training activities through the organization of two PhD schools and the third edition of eRISE. FOSAD 2013 is co-organized by NESSOS and the EPSRC NoE on the application of formal methods to cryptography (CryptoForma). The school will be held in Bertinoro from September 2-7 2013. Another PhD school co-organized by NESSoS and ANIKETOS project will be held in May 2013 in Malaga and co-located with IFIP TM conference. eRISE 2013 has been planned and it will focus on the comparison of academic and industrial methods. Pete Herzog has already accepted to participate as method designer of OSSTMM methodology.
Bibliography


