predict
prioritise
prevent

TRESPASS

Technology-supported Risk Estimation by Predictive Assessment of Socio-technical Security

Deliverable D.5.1.2

Final requirements for process integration

Project:	TRE$_S$PASS
Project Number:	ICT-318003
Deliverable:	D.5.1.2
Title:	Final requirements for process integration
Version:	1.0
Confidentiality:	Public
Editor:	J. Willemson
Cont. Authors:	W. Pieters, J. Willemson, M. Seeba, M. Martins, A. Lenin, H. Tark, C. Probst, A. Tanner, B. Kordy, R. Trujillo, R. R. Hansen, M. Ford
Date:	2015-10-30

Part of the Seventh Framework Programme
Funded by the EC-DG CONNECT

Members of the TRESPASS Consortium

1.	University of Twente	UT	The Netherlands
2.	Technical University of Denmark	DTU	Denmark
3.	Cybernetica	CYB	Estonia
4.	GMV Portugal	GMVP	Portugal
5.	GMV Spain	GMVS	Spain
6.	Royal Holloway University of London	RHUL	United Kingdom
7.	itrust consulting	ITR	Luxembourg
8.	Goethe University Frankfurt	GUF	Germany
9.	IBM Research	IBM	Switzerland
10.	Delft University of Technology	TUD	The Netherlands
11.	Hamburg University of Technology	TUHH	Germany
12.	University of Luxembourg	UL	Luxembourg
13.	Aalborg University	AAU	Denmark
14.	Consult Hyperion	CHYP	United Kingdom
15.	BizzDesign	BD	The Netherlands
16.	Deloitte	DELO	The Netherlands
17.	Lust	LUST	The Netherlands

Document History

Authors		
Partner	Name	Chapters, responsibilities
TUD	Wolter Pieters	5, 7, quality assurance
CYB	Jan Willemson	1, 2, 3, 4
CYB	Mari Seeba	2, 3
CYB	Aleksandr Lenin	1, 3, 4
CYB	Heigo Tark	4
ITR	Miguel Martins	5
UT	Dan Ionita	architecture task force
DTU	Christian Probst	6, requirements task force
IBM	Axel Tanner	4
TUHH	Dieter Gollmann	Quality assurance
UL	Barbara Kordy	4
UL	Rolando Trujillo	4
AAU	René Rydhof Hansen	Quality assurance
CHYP	Margaret Ford	4, editing the document
BD	Henk Jonkers	Quality assurance

Quality assurance		
Role	Name	Date
Editors	Margaret Ford, Henk Jonkers	2015-06-29
Reviewer	Sergio Saraiva	2015-08-31
Reviewer	Jan-Willem Bullée	2015-08-31
Reviewer	Dieter Gollmann	2015-10-15
WP leader	Jan Willemson	2015-06-30
Coordinator	Pieter Hartel	2015-08-16

Circulation	
Recipient	Date of submission
Project Partners	2015-09-30
European Commission	2015-10-30

Acknowledgement: *The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 318003 (TRESPASS). This publication reflects only the author's views and the Union is not liable for any use that may be made of the information contained herein.*

Contents

List of Figures

Management Summary

Key takeaways:

- Interviews with security practitioners and hacker community representatives support the project's directions, and provide additional requirements for the TRE$_S$PASS process;
- TRE$_S$PASS will provide risk assessment support for multi-step, socio-technical attacks, based on the Risk Taxonomy of The Open Group;
- The TRE$_S$PASS workflow integrates attack navigators in existing risk management processes, enabling reuse of models and data;
- The processes support selection of cost-effective countermeasures.

Within the TRE$_S$PASS project, WP5 deals with the integration of the risk estimation processes, making use of the TRE$_S$PASS tools. The integration includes the creation of socio-technical security models, the interpretation of analysis results, and the interface to existing risk assessment frameworks. This enables organisations to use the TRE$_S$PASS attack navigators effectively and efficiently.

This document presents the requirements for the integrated TRE$_S$PASS process, based on analysis of existing practices and the limitations of those practices. To this end, this document relies both on literature and interviews with security practitioners.

First, existing risk assessment and audit practices are analysed, based on literature and interviews. From this study, requirements for the TRE$_S$PASS workflow are identified, including integration of social, physical and digital security aspects, and automation of data gathering to the extent possible. Second, the TRE$_S$PASS risk assessment framework is discussed, which is based on the Risk Taxonomy of The Open Group. Third, the workflow for using the TRE$_S$PASS tools in risk assessment and audit practices is outlined. Finally, these elements are used to define concrete requirements for the tasks in the work package, as well as related work packages.

1. Introduction

Appendix A provides the context for this deliverable in the TRE$_S$PASS project. It describes the overall summary of the project and the TRE$_S$PASS workflow.

1.1. Goals

This document provides the final requirements for WP5 of the TRE$_S$PASS project. The objective of this work package is to develop the integrated, systematic TRE$_S$PASS process around the use of the models and tools provided. This integration includes the whole risk assessment workflow: creation of socio-technical security models, transformation and analysis of these models, interpretation and visualisation of the analysis results, and interfaces to existing risk assessment frameworks.

The tasks in this work package have been motivated by the following development goals:

- Methodology to enable the identification of the key actors and key components in the system, their interactions and roles within the organisation;
- Methodology to support the identification and description of the vulnerabilities;
- Methodology to determine the abstraction level of the models acceptable for sharing among different players within the process;
- Process of iteration between the qualitative interpretative data gathering and visualisation and the quantitative predictive socio-technical security model;
- Rules for updating the socio-technical security model based on the feedback from practical security assessments;
- Use the existing development potential and fill the gap in the market of data analysis tools and tools for infrastructure description with the TRE$_S$PASS tools and processes;
- Design TRE$_S$PASS to be able to adapt to changes in the risk environment, which will be one of the key success factors of TRE$_S$PASS.

This document presents the requirements for the work package based on these goals. The report builds upon a previous deliverable D5.1.1 “Initial requirements for process integration” (The TRE$_S$PASS Project, D5.1.1, 2013). Compared to D5.1.1, the current document contains updated descriptions of TRE$_S$PASS workflows as well as a new study of attacker interviews.

1.2. Structure of the document

Firstly, in chapter 2 we describe one widely used model framework (CISA) used by practitioners for security audits. In chapter 3, we provide results of interviews with security practitioners, identifying industry needs and opportunities for improvement. Chapter 4 presents the results of the interviews with the representatives of the white-hat attacker community (penetration testers, security analysts, etc.). In chapter 5, we outline the approach to risk management proposed for the TRE$_S$PASS process, in relation to the existing frameworks. Finally, in chapter 6 we summarise the requirements gathered during the cross-project requirements effort and chapter 7 presents our conclusions.

1.3. Background and Foreground

Most of the current deliverable comprises of foreground results. The sections containing background information are chapter 2 (CISA auditing framework) and part of chapter 5 (discussions concerning existing risk assessment frameworks). Chapters 3 and 4 contain original interviews.

2. A model auditing workflow: CISA

In order to guarantee the effectiveness of TRESPASS in addressing real world needs, an analysis of the existing risk assessment frameworks was conducted. Task 5.2 of the TRESPASS project was specifically devoted to the review of standards and its Deliverable 5.2.1 provides a comprehensive overview of the currently established methods and frameworks (The TRESPASS Project, D5.2.1, 2014). Here we will only briefly review the framework described by the Certified Information Systems Auditor (CISA) manual and then discuss the routines used in practice.

The role of audit is to assure the management of the organisation that risk management is handled properly and the risks are treated efficiently. From this perspective audit can be viewed as one of the risk treatment control mechanisms. Risk assessment, as part of the risk management process, may also form part of the audit process, in that risk identification and prioritisation are required in each case. We are interested in the risk assessment practices and methodologies, as there is potential for the TRESPASS processes and tools to contribute in this area and meet current industrial needs.

2.1. CISA auditing workflow

A detailed overview of established risk assessment methods has been released in Deliverable 5.2.1 of the TRESPASS project in month 24 (i.e. October 2014). References to the relevant standards can be found in Deliverable 9.2.1 (The TRESPASS Project, D9.2.1, 2013). Hence, in this deliverable we will only briefly describe one model of workflow, to serve as a comparison for the various workflows which practitioners use in practice. For our model we chose the workflow described in the Certified Information Systems Auditor (CISA) studyguide (Cannon, Bergmann, & Pamplin, 2006) and manual (*CISA Review Manual 2012*, 2011), as CISA a globally accepted standard of achievement among information systems audit, control and security professionals and worldwide recognised certificate for IS auditors.

The general workflow is depicted in Figure 2.1.

The audit charter gives the auditor the authority to perform the audit, delineates the area to be audited and specifies the purpose and scope of the audit.

Audit preplanning is the phase in which the required skills and resources are identified. The auditor decides on the viability of conducting an audit on the basis of the original charter, according to the availability of the necessary skills and resources.

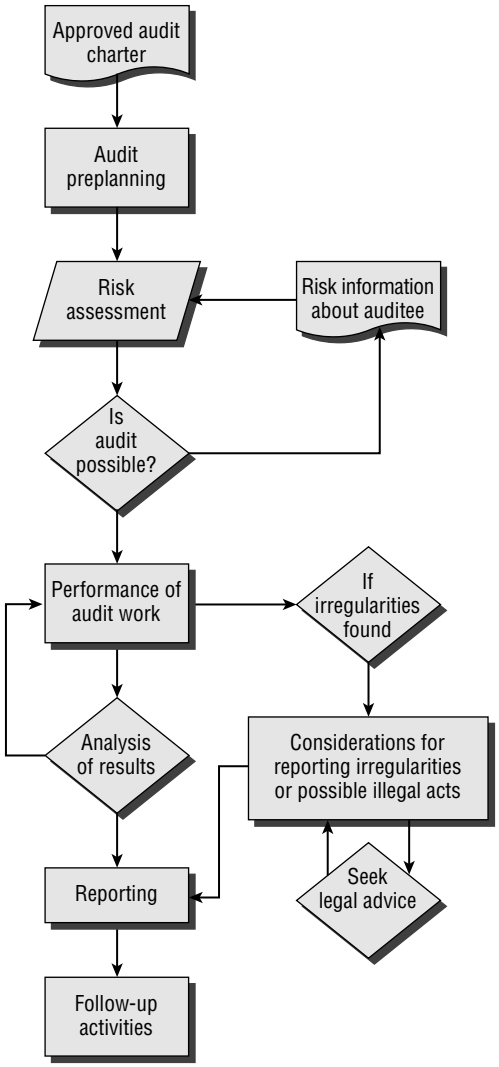

Figure 2.1.: Overview of the audit process (CISA Manual)

The core work of the audit involves audit procedures and steps for data gathering: selection of the audit approach to verify and test the controls, preparing the list of individuals to interview, identifying and obtaining regulations, policies, standards and guidelines to review, developing the auditing tools and methodologies. The auditor can collect information from business records, computer data files, computer assisted audit tools (CAAT), personal interviews, workshops, and surveys. All information and evidence should be recorded and tracked.

The audit report includes findings and recommendations to be submitted to management. The report should include not only negative findings but also give positive feedback about controls which are functioning effectively. When communicating the audit results, there are

usually two presentation techniques: executive summary (easy-to-read concise report) and visual presentation (slides, graphics, etc.).

In the phase following the audit, management will be required to take appropriate corrective actions and the auditor should have a follow-up programme to ensure that the agreed corrective actions have been implemented. External auditors do not necessarily have to be involved in this process.

As mentioned above, risk management is both subject to audit and an essential component of the auditing process, which includes its own risks that need to be assessed, mitigated and treated. The general risk management process is described in Figure 2.2. As the Figure shows, it is a continuous process, requiring the organisation to continually identify and evaluate risks as they arise and evolve.

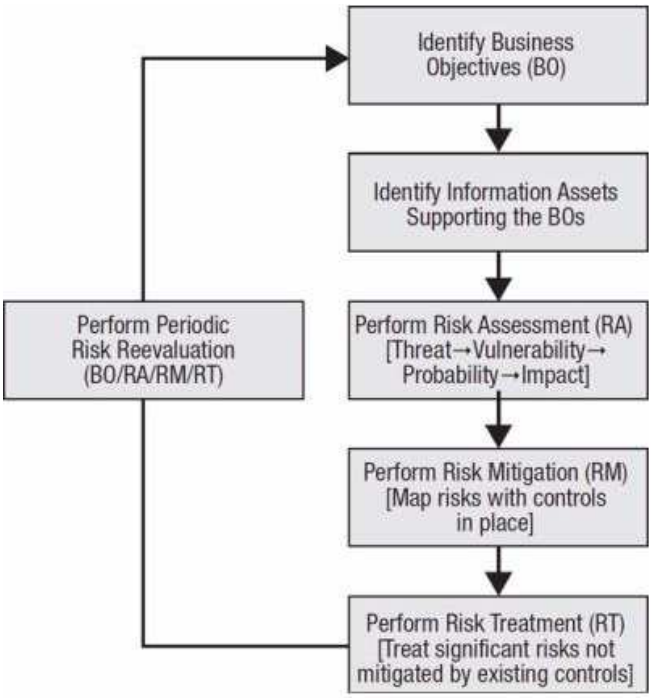

Figure 2.2.: Risk management process (CISA Manual)

One possible approach to handling risks in the audit process can be seen in Figure 2.3.

Relevant audit classes

The following audit classes are defined in the CISA review manual:

- **Financial audit** – assesses the correctness of an organisation's financial statements;

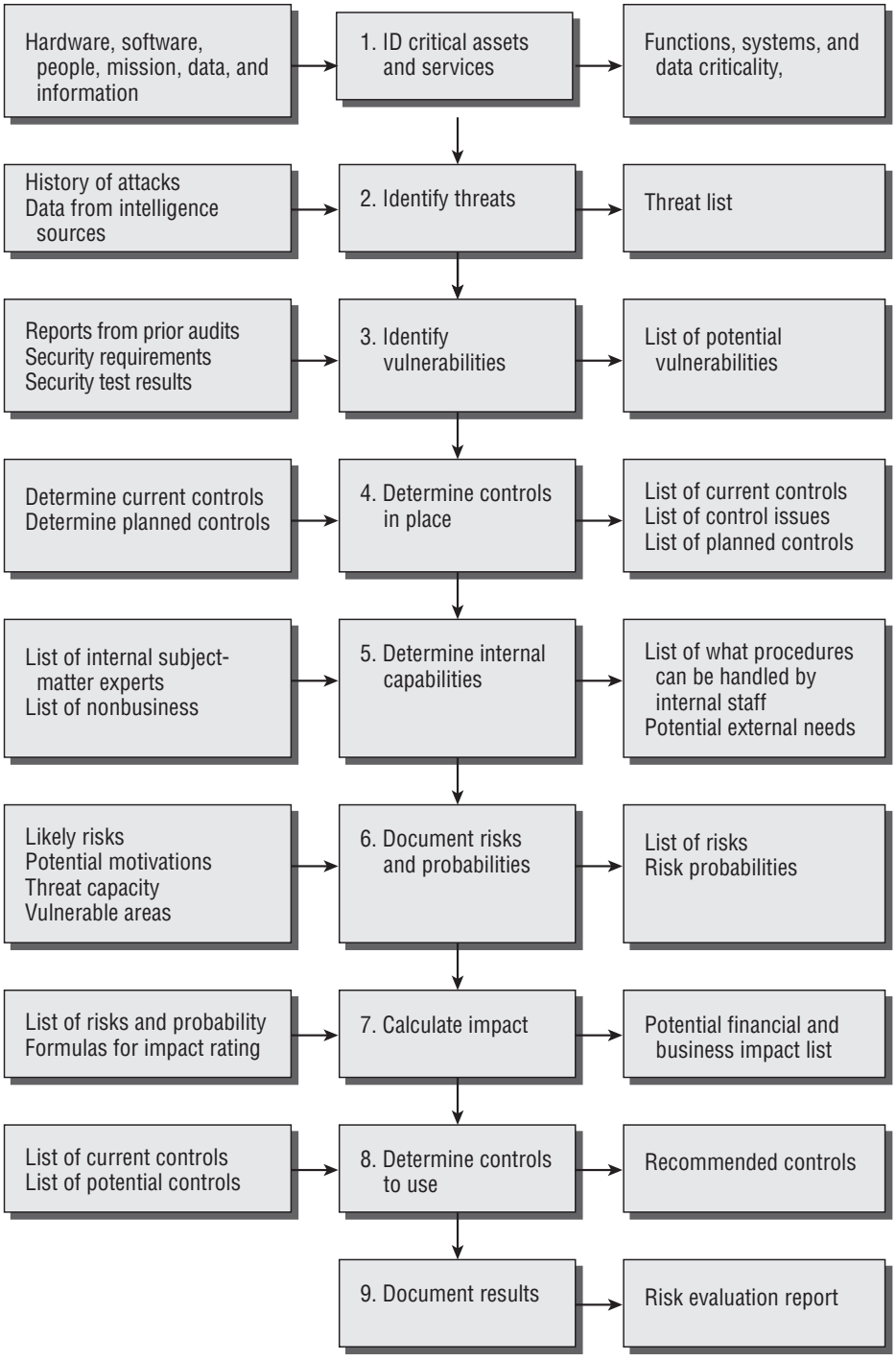

Figure 2.3.: Risk assessment and treatment process flowchart (CISA Manual)

- **Operational audit** – evaluates the internal control structure in a given process or area;
- **Integrated audit** – combines financial and operational audits. It is also performed to assess the overall objectives within the organisation, related to financial information, safeguarding of assets, efficiency and compliance;
- **Administrative audit** – assesses issues related to the efficiency of operational productivity within an organisation, verifies that appropriate policies and procedures exist and have been implemented;
- **Information systems audit** – collects and evaluates evidence to determine whether the information systems and related resources adequately achieve their goals. These goals may include safeguarding assets, maintaining data and system integrity, providing relevant and reliable information, achieving organisational goals effectively, consuming resources efficiently, and having efficient internal controls, with the aim of providing reasonable assurance that business, operational and control objectives will be met and that adverse events will be prevented, or detected and corrected in a timely manner;
- **Specialised audit** – examines areas such as services performed by third parties and forensic auditing, and evaluates internal controls in these environments. The result of this type of audit is to confirm that a service organisation has been through an in-depth audit of their control activities, which generally include controls over information technology and related processes;
- **Forensic audit** – discovers, discloses and follows up on frauds and crimes.

Also, the following audit class is defined in the CISA auditor study guide:

- **Compliance audit** – verifies the implementation of and adherence to a standard or regulation. This usually forms part of other audit classes.

Although the above descriptions relate to more general audit classes, security and risk management form an essential part of any audit. While clients may choose to focus specifically on the security audit, information security features including confidentiality, integrity and availability are relevant to all types of audit.

3. The TRESPASS audit questionnaire

Based on the collective experience of the partners, the TRESPASS consortium designed a questionnaire. The aim of the questionnaire was to gather valuable input on the audit frameworks and processes used by practitioners, allowing the alignment of the TRESPASS processes with industrial needs. Questions relevant to the area of research were picked and included in the questionnaire. The results were collected from practitioners, whom the project partners considered suitable, in different countries across Europe. The questionnaire consisted of the following 12 questions:

1. Please describe a risk management strategy used by your organisation.
2. What is the audit methodology used in this strategy (e.g. compliance audit, IS audit, operational audit)? Please describe different stages of the audit process and/or refer to the appropriate manual/standard.
3. How do you acquire relevant data (e.g. documentation, interviews, system tests)?
4. How is the acquired data managed and analysed (e.g. specialised tools (please name), statistics)?
5. When presenting audit results to your customers, are you using any specific presentation techniques (e.g. visualisation, multimedia)? If yes, please describe some characteristic ones.
6. Do you deviate from your mainstream audit process? If you do, please describe some characteristic cases.
7. How complete do you estimate your audit methodology to be? What kinds of potential problems is it capable of detecting and which ones not?
8. How do you describe client's infrastructure (e.g. network topology, location of data sensitive equipment within a building, physical and digital access control)? Please name the methods and tools used.
9. Are you using any risk assessment tools in your process? Have you developed any yourself? If yes, please list them and describe briefly their benefits and shortcomings.
10. Do you take social engineering attacks into account? If yes, please describe some characteristic cases and tools used.
11. How do you adjust your auditing processes and risk management strategies when the environment changes, new attacks emerge, etc. (e.g. wait for the next version of the standard, update your own best practice manual)?

12. What kind of tools could be useful to you to give socio-technical descriptions of your clients' infrastructures and to take social engineering attacks into account? Please dream freely :)

Based on this questionnaire, the TRE$_S$PASS consortium members conducted the following structured interviews (concrete identities of some of the interviewees have been anonymised due to their clearly expressed preferences).

- An employee responsible for operational integrity within a large global cloud services provider was interviewed by Margaret Ford (CHYP) on January 23rd, 2013;
- An IBM employee working as a team member in Internal Audits, i.e. the specific group in IBM doing audits on internal parts of IBM, was interviewed by Axel Tanner (IBM) on January 17th, 2013;
- An IS auditor who is a member of the ILNAS (Institut Luxembourgeois de la Normalisation de l'Accréditation, de la Sécurité) expert group was interviewed by Guillaume Schaff (ITR) on January 31nd, 2013;
- A manager from Security Audits of Deloitte Netherlands was interviewed by Wolter Pieters (TUD) and Trajce Dimkov (Delo) on January 18th, 2013;
- A person working for an auditing company and dealing with penetration testing was interviewed by Barbara Kordy (UL) in December 2012;
- A member of the national Computer Security Incident Response Team in Luxembourg was interviewed by Barbara Kordy (UL). The interview was performed using email and lasted from December 2012 until January 2013;
- Two specialists from the Estonian Information Sytems' Authority responsible for implementation of the ISKE baseline security framework were interviewed by Mari Seeba and Jan Willemson (CYB) on December 18th, 2012.

Data from the transcriptions of the interviews were grouped by question, and labelled in terms of topic and relevance. For each question, a summary of relevant results was created. The summaries are presented in the next section.

3.1. Analysis of interview results

3.1.1. Risk management strategies used

The organisations described by interviewees have rather a wide variety of practices and frameworks for auditing and risk management. On two occasions, respective ISO standards (31000 (*ISO 3100**, 2013) and 27000 (*ISO 2700**, 2013) families) were mentioned, two interviewees have mentioned the COSO framework (*COSO*, 2013) (targeted mainly towards internal auditors), one mentioned COBIT (*COBIT*, 2013) and one a national baseline security framework. Several organisations have their own, historically developed best

practices involving different technical components like penetration testing and simulated social engineering.

3.1.2. Auditing methodologies

Since most of the interviewees specialise in IS auditing, IS audits were most commonly mentioned in the replies. Also, operational, financial and compliance audits are occasionally performed by the interviewees.

Stages of a typical audit process were nicely laid out in one of the interviews.

1. Before the audit (remote)
    a) Selection of the organisation to be audited, based on risk factors like tasks for which the organisation is responsible, financial impact, time of last audit, substantial recent or future projects.
    b) Notification to organisation 2-4 weeks before on-site audit starts.
    c) Initial data request, e.g. for organisational charts, process overview, data dumps. Has to be responded to within a week by the organisation to be audited.
2. Audit (auditors on site at the organisation)
    a) Week 1
        i. Kick-off meeting with preliminary scope of audited processes (e.g. management of client systems).
        ii. Process presentations by audited organisation.
        iii. Process reviews (in interviews).
        iv. Scope can be adjusted flexibly depending on findings.
    b) Week 2
        i. Further data requests, interviews and walk-throughs according to the process and topic under investigation, digging deeper where needed (e.g. inventory listings, user ID listings, access control lists, access logs, risk acceptance documents).
        ii. Definition of a preliminary list of concerns.
    c) Week 3
        i. Presentation of the list of concerns to the audited organisation.
        ii. Opportunity for the audited organisation to correct factual misunderstandings or to present mitigating data.
        iii. Aggregated data sheets with findings created (short, only a few pages), to be signed by the head of the audited organisation.

d) Week 4

i. Final audit report created (short - ideally one page), with findings and verdict (satisfactory or failed).

ii. Audit report presented to management of audited organisation.

iii. Audit report will be sent to management plus two levels up.

3. After audit

a) Creation of an action plan by the audited organisation within 30 days, to be agreed by the audit team.

b) Action plan will be tracked by the audit team and has to be finished within at most 6 months.

c) If not handled satisfactorily, an escalation process will be started.

There are also standards and sets of guidelines used in the audits, e.g ISO 27001:2005, ISO 27002:2005, ISO 27005:2008, ISO 17002:2004, ISO 15408-1:2009, OWASP (*OWASP*, 2013), RFC2196 (*RFC 2196*, 2013) and EuroPrise (*EuroPrise*, 2013).

3.1.3. Data acquisition

Data acquisition is a key problem for any risk analysis framework, and it is especially relevant in the case of quantitative analysis. Hence, the wide variety of approaches used in practice:

- Interviewing of key personnel is almost always included. It was noted by one interviewee that the results of the interviews are not always reliable, as staff may have an incentive to be optimistic about the state of affairs. They may equally have an inadequate understanding of their environment, without necessarily recognising or admitting this;
- Mostly, the auditors also receive some sort of documentation, and on one occasion source code of some applications to be audited was mentioned;
- Data acquisition from log files was mentioned in 4 structured interviews out of 7. One of the interviewees pointed out the need to gather log files under the supervision of the auditor in order to avoid their alteration (either maliciously or by accident);
- If the aim of the audit requires it, more active methods (like penetration testing or observing operations in action) can be included.

In general, the questions of data source reliability, completeness and ease of automation were raised over and over again. One of the interviewees explicitly stated the requirement to gather as much data as possible automatically. Manually gathered data often tends to be incomplete, imprecise and often incorrect. Automation of this process would help not

only to reduce the human workload, but improve the correctness and integrity of the collected data and guarantee that all the available data is gathered and nothing was missed in the scope of the automation process. Based on the information gathered from the interviewees we can conclude that **in order to meet industrial needs, the TRESPASS tools and processes need to support automatic data acquisition from corporate networks.** Some existing products which could be used as starting points include business management software (e.g. SAP (*SAP*, 2013)), network scanners for mapping the network topology (e.g. Nmap (*Nmap*, 2013)) and integrated forensics tools (e.g. Maltego (*Maltego*, 2013)). A deeper investigation of potential data sources is included in the WP2 deliverables (D2.1.2 (The TRESPASS Project, D2.1.2, 2015)).

3.1.4. Data analysis

Whereas the auditing processes are generally well-established and even standardised, the results obtained from the interviewees show that the subsequent data analysis step is much less uniform. Mostly, the auditors utilise only the analysis capabilities of the software used to collect the data (e.g. WireShark (*Wireshark*, 2013), itrust's proprietary solution, Nessus (*Nessus*, 2013), OpenVAS (*OpenVAS*, 2013), MetaSploit (*Metasploit*, 2013), Nikto (*Nikto*, 2013), W3af (*W3af*, 2013), Burp (*Burp*, 2013). Alternatively, some generic data management solution may be used (e.g. IBM Lotus Notes (*IBM Lotus Notes*, 2013) or Microsoft SharePoint (*Microsoft SharePoint*, 2013). This means that the link between the acquired data and risk assessment is currently underdeveloped. **Consequently, there is a lot of development potential and a gap in the market, which TRESPASS tools and processes could fill.**

3.1.5. Presentation techniques

The motivation for including the question about presentation techniques was to relate this to the visualisation techniques being developed in WP4. It was interesting to learn the degree to which modern multimedia applications are used to deliver communications between auditors and their customers.

The results show that in this specific area, there is significant room for improvement. All the interviewees responded that they use a standard written report format, occasionally accompanied by a PowerPoint slide presentation. Also, different kinds of graphics and spreadsheets are used, where appropriate. However, these presentations tend to be static, despite the existence of several indicators, such as risk maturity, which can show the evolution of risk.

3.1.6. Deviating from the mainstream auditing process

6 out of 7 interviewees reported rather strict adherence to the prescribed processes. However, the processes are mostly flexible enough to respond to the possible changes in the

environment. Only one interviewee identified deviation as a deliberate feature. It is especially interesting in the context of social engineering, as it helps the system to evolve unpredictably for the attacker.

The TRE$_S$PASS tools should take into account, that strict adherence to standards is not essential, and possible deviations may be present. Thus the TRE$_S$PASS tools should support possibility to deviate from following a standard to some extent, in case a TRE$_S$-PASS user feels a need in such a deviation.

3.1.7. Completeness of the methodology

The aim of this question was to find out how secure the auditors feel about the methodology they use. An unanimous answer was that the methodology is as complete as it can realistically be, but not all of the aspects are or even could be covered. There are challenges in taking the right aspects into account, and the problems are not always where one expects them. As they currently stand, all the auditing methodologies still depend on the expertise of the person running them. **TRE$_S$PASS may improve the current methodologies by automating some aspects of the socio-technical analysis, but most probably it will never be complete in the sense that human intervention would not be needed.** However, the TRE$_S$PASS processes may support security practitioners in focusing on the most important areas for investigation. For example, attacks and countermeasures quantitative attributes estimation cannot be fully automated, but the system can be designed in such a way that possible expected values are presented to the user with the possibility to pick the suitable ones. Even if to assume, that corresponding quantitative metrics can be fully automatically derived from the underlying system, human intervention is still needed to validate the values and estimate if they allow to model the analysed system with the desired precision.

3.1.8. Methods of describing the client's infrastructure

This question was motivated by the aim of developing socio-technical models within the TRE$_S$PASS project. It would appear that currently the infrastructure models are not very well standardised. Several interviewees reported an *ad hoc* approach based on whatever documentation the auditor is able to obtain from the client organisation. Sometimes a slightly more systematic approach is used and some predefined types of data are collected (describing e.g. network topology, physical and digital access control, international locations of data sensitive customer equipment, detection mechanisms, 3rd party activities, outsourced parts, agreements (SLAs), and monitoring). **In general, standard tools for infrastructure description would also fill a gap in the current processes.**

3.1.9. Usage of specific risk assessment tools

4 out of 7 practitioners do not use any specific risk assessment tools. There are some notable exceptions, though.

The TRESPASS consortium partner itrust has developed its own risk assessment tool called TRICK Light. The tool assesses the risk level of an organisation based on a quantitative approach of the risk (annual loss expectancies in EUR) and also assesses the ROSI (Return Of Security Investment) of the implementation of ISO 27002 controls. In addition to the global risk analysis tools, itrust uses a set of open source tools (titled TRICK Tester) to provide penetration tests which can be used to assess the vulnerability of an infrastructure, service, or application.

Another interviewee drew our attention to several useful software tools:

- Computer Based Social Engineering Tools: Social Engineer Toolkit (SET)
  http://www.social-engineer.org/framework/Social_Engineering_Framework
- cve-search (tool for automating local searches of software vulnerabilities)
  http://adulau.github.com/cve-search/, https://github.com/adulau/cve-search/
- Network simulator (for security)
  http://www.hynesim.com/Home-610-0-0-0.html

Out of these references, the SET toolkit seems to be the most relevant one for the TRESPASS project. **In fact, it seems that the SET toolkit could be used as a readily available component in the TRESPASS tool set.**

3.1.10. Taking social engineering attacks into account

In 6 out of 7 cases, social engineering attacks were acknowledged by the interviewees. The means of dealing with such attacks were, however, rather *ad hoc*. End user education was mentioned as the primary prevention mechanism. This is because technical solutions are to great extent deterministic, but social behaviour of people is to some extent stochastic. Thus it cannot be entirely analysed by applying technical solutions for analysis. Some solutions based on AI and machine learning could be some aid, but they would allow to predict and analyse social behaviour to some limited extent only. This area hasn't evolved yet into something that could have practical applicability. Even if technical measures are reliable and the security is kept at a high level, human factor remains the thing that is hard to predict and analyse, and human remains the weakest link in this chain. Thus we cannot eliminate this vulnerability, but we can accept it. Awareness training in this case is one of the ways to reduce the impact of a risk of employees being social-engineered by an attacker, it can be called the best-practice applied nowadays and has proven to be a good mitigation strategy that is acceptable in most cases. There was more uncertainty about detection mechanisms. Different organisations have experimented with simulated social engineering attacks, having ideas of setting up *"human honeypots"*, including fake phone numbers, documents, in order to attract potential attackers and try to detect them as they

make their move. **There is a definite need to supply the auditors with more systematic mechanisms for detecting social engineering attacks. These mechanisms will probably never be fully automatic, but their usage can probably be more effectively packaged for human operators.**

3.1.11. Responding to the changing environment

Mostly, organisations rely on the methodology update features built into the standards development processes (e.g. ISO review process (*ISO Review Process*, 2013)). However, it is also acknowledged that this might not be sufficient to withstand the fast pace of attack development. The case of baseline security (BSI (*BSI*, 2013), ISKE (*ISKE*, 2013)) is very characteristic here – the standards manuals are thousands of pages long and it takes so long to update them that the technology and the corresponding attacks are much faster moving. As a result, such approaches are not viable in the long run and should be replaced by methods that are much easier to update. **One of the key success factors of TRE$_S$PASS is its ability to adapt to changes in the risk environment.**

Best practices on incident response in relation to risk assessment are currently emerging. TRE$_S$PASS processes would need to engage with such practices.

3.1.12. Development ideas for a socio-technical security assessment framework

Several interesting development ideas were proposed by the interviewees.

- A comprehensive system for analysing logs from disparate systems, to assist in identifying incidents and suggesting possible scenarios leading to those incidents (with an appropriate degree of human sensitivity built in) i.e. “a possible action leading to this outcome is XYZ – here are the related log entries”, rather than “PJ’s been repeatedly trying to break into the system and has locked everything up”.
- One could envision a system monitoring all communication channels of employees like email, phone, network, resources etc. and cross-correlating these with resources which require protection, in order to mine for patterns of behaviour of non-compliance (such as separation of duty violations), but besides the technical challenges, this poses fundamental privacy problems which, apart from contravening privacy laws, would also be undesirable.
- To assess the security induced by social behaviour in an organisation using an information system, it would be useful to have:
  - Real-time probes of user activities especially high risk activities such as internet usage behaviour, PC screen session locking behaviour, etc.;
  - Specific honeypot or test processes to assess the risky behaviour of users e.g. by making available an infected appliance (as a USB key);

  - Specific behaviour testing processes on site.

- In particular, an integrated view of people, technology and other organisational factors would be useful, including vulnerabilities in relationships of specific objects, as well as the motivation of individuals. This could be included in GRC (governance, risk, compliance) suites. TRESPASS processes would need to support the alignment of different types of controls, as well as raising awareness about the mutual influences between the potential "silos" of physical, digital, and social aspects of information security.

3.2. Conclusions

The main aim of this chapter is to provide an overview of the data collection and discovery practices used in the analysis of live information systems. Having neither the resources nor the need to conduct studies involving large groups of people at this stage, our aim was rather to collect practical requirements for the development of TRESPASS tools and processes.

In order to gather such requirements, we conducted structured interviews with seven security practitioners, working in a range of environments across Europe. As a comparison baseline, we also included a short overview of one standardised workflow, namely the CISA auditing workflow. (A more thorough analysis of standardised frameworks will be conducted within Task 5.2 of TRESPASS.)

We see that 4 out of 7 interviewees declared their compliance with particular standards (e.g. ISO 27000 family). We also received some other interesting items to serve as input to TRESPASS development process. In 2 out of 7 interviews, usability aspects of the prospective tool set were emphasised. For example, it was required that collecting numerical data for the quantitative analysis should be as automatic as possible, i.e. not requiring people to fill in long forms asking for costs, probabilities, etc. At the same time it was noted that since we are dealing with inherently human and evolving threats, full automation will probably never be possible. One approach to collecting numerical data under such contradictory requirements might be to develop some kind of software for remotely assessing human behaviour (along the lines of the Social Engineering Toolkit).

The preliminary results of this analysis were presented and validated during the TRESPASS meeting in Lisbon (March 2013) and were well received by the project partners. Collection of real world requirements has since continued, with the interviewing of active social engineering attackers and gathering of their views on the analysis of such attacks.

4. The TRESPASS attacker questionnaire

Based on the collective experience of our partners, the TRESPASS consortium designed a questionnaire with a view to interviewing experts, whose practical experience gives them an insight into the motivations of attackers. It was decided that there were a number of practical considerations that prohibited engagement with black hat hackers. However, by interviewing seasoned professionals such as penetration testers, CERT specialists and individuals who have approached hacking as a hobby, we have gained valuable insights into the mind of the attacker. For brevity, we refer to the questionnaire as the 'attacker questionnaire'. It should be noted though that we have not contacted any representatives of the real black-hat hacker community. This is mostly due to none of the project partners having direct connections to this community. Real black-hat hackers are know to be difficult to approach without personal acquaintance; also, their willingness to cooperate with a project like TRESPASS would have been questionable.

Extending the approach taken to the security practitioners' interviews described in Deliverable 5.1.1 (The TRESPASS Project, D5.1.1, 2013), the following steps were undertaken to prepare the survey document:

1. Potential areas of interest were identified through a brainstorming session;
2. An initial version of the questionnaire was developed;
3. This initial version was validated by presenting it to Computer Security Masters students at Tallinn Technical University.
4. Based on the feedback from this validation, the questionnaire was improved and made more concise.
5. Finally, a set of guidelines and additional questions was designed to support the interviewers.

The aim of the questionnaire was to gather expert input on attacker profiles, in order to support the alignment of the TRESPASS processes with the needs of information security practitioners. Questions relevant to the area of research were selected for inclusion in the questionnaire. The responses to these questions and additional comments were gathered from practitioners active in countries across Europe, whom project partners had identified as having the necessary expertise. The questionnaire consisted of 12 questions that can be found in Appendix B.

Based on this questionnaire the TRESPASS consortium members conducted the following structured interviews (concrete identities of some of the interviewees have been anonymised due to their clearly expressed preferences).

- A principal security consultant in Madison Gurkha, working for a company that supports organizations with high quality services to efficiently identify, mitigate and prevent IT security risks. Interviewed by Barbara Kordy (UL) June 20, 2013;
- A penetration testing expert having an extensive knowledge of social engineering attacks interviewed by Barbara Kordy and Rolando Trujillo(UL) on May 28, 2013;
- An ethical hacker and information security enthusiast, having an extensive knowledge of technical vulnerabilities and related attacks interviewed by Aleksandr Lenin (CYB) on August 16, 2013;
- A penetration tester, interviewed by Jan Willemson (CYB) on March 6th, 2014;
- A Security Consultant of IBM Security Services who leads a group of 15 people doing penetration testing as contracted by customers was interviewed by Axel Tanner (IBM) on March 15th, 2014;
- A CEO of a UK based commercial company specialising in penetration testing, interviewed by Margaret Ford (CHYP) on March 20, 2014.

Data from the transcriptions of the interviews were grouped by question, and labelled in terms of topic and relevance. For each question, a summary of relevant results was created. The summaries are presented in the next section.

4.1. Analysis of interview results

This chapter will give an overview of the responses received from interviewees by analysing the answers given to specific questions. We will use the results of this analysis to identify 'best practice' and further to describe different attacker profiles, to be used in subsequent TRE$_S$PASS profiling.

4.1.1. Profiles of interviewees

The interviewees came from different backgrounds. While none of them was a genuine hacker, they all work for security-related companies. Four of the six are working as pentesters and therefore have skills in planning and launching attacks. The other two are security experts.

All the interviewees are experts in their fields: five of them have worked in the security field for over ten years and one for around three years. Of these, four interviewees have very strong technical experience, with one considering himself expert in social engineering and one being expert in both social and technical aspects.

4.1.2. Motivation, nature and resources when attacking

Most of the interviewees support the idea of combining the motivation, nature and resources of the attacker into the attacker profile. The goal the attacker wants to achieve can also be considered as part of this profile. Great variations in attacker motivation, resources and capabilities may be observed between different attacker profiles.

Some emerging profiles are suggested from the responses received:

1. **Attacker is just an individual:** who tries some automatic scripts, scans to discover known vulnerabilities with a mostly exploratory motivation;
2. **Attacker is a hacktivist:** who has some specific goal to achieve recognition or to mine data for confidential information;
3. **Attacker is a person/group of people with higher skills:** who are able to write code and have specific security related knowledge, and may want to gain and maintain access for very specific purposes.
4. **Attacker is an organization:** who are able to buy/write code and hire security expert related knowledge, for very specific purposes (e.g. NSA, private security companies)

These profiles determine the level of resources put into an attack, the nature of the goal (money, reputation, espionage etc) and how well the attack is planned.

4.1.3. Exploratory activities

Exploratory activities are frequently used, even if they are not part of a specific attack plan. The knowledge gained about vulnerabilities can be used in subsequent steps. Also, hackers may just want to explore options (make non-targeted attacks) regarding security risks. When they find an opportunity, they may build a future attack around it. This approach can be used to support social engineering by trawling information about person's private life from social networks.

4.1.4. Discovering vulnerabilities

The main feature of the responses is the confirmation that discovering and identifying social, technical and physical vulnerabilities is based first on the attacker's previous experience and/or knowledge. Where this experience is lacking, the attacker will try to act based on instinct.

From this point onwards, a capable hacker combines different vulnerabilities, identifying social and/or technical attacks to achieve the goal. As human factors are frequently the weakest link in security, social engineering may provide the easier option.

Depending on the goal, the attacker chooses a preferred pattern and approach to using it.

4.1.5. Criteria to compare weak spots

A common approach to identifying criteria in finding/comparing weak spots and deciding if the attack is worthwhile is related to the attacker's experience and motivation. The attacker's strategy in finding weak spots will be based on personal experience so that the impact/profit/gain is maximised. If the goal is reputation, the attacker will aim for a quick and visible exploit. If the goal is making money, the attack may be planned more thoroughly.

The first choice in comparing weak spots is likely to be based on social engineering, as this is often an area of weakness and may offer an effective area of attack (if first targeted person does not fall for the attack, there is every chance that others will). Usually, to achieve the goal, the attacker will combine different factors (social, technical).

4.1.6. Typical social engineering cases

All interviewees mention that a typical social engineering attack would include phishing, phone calls to get inside information, USB flash usage inside the company, getting access to the company's premises through an employee's willingness to help someone posing as a coworker.

It is frequently mentioned that every attacker has limited resources. One cannot estimate these parameters precisely, but still people are often the weakest point in the security chain. There is a good chance that out of every 100 employees at least some will fall victim at some point (e.g. phishing email, opening door, offering physical access, etc.).

Education about social engineering to improve security awareness within the company is advisable to protect against this threat. The willingness of employees to provide security related information, is illustrated by the response to a survey sent to a group of employees. Although actually part of a penetration exercise, this questionnaire was ostensibly from a university student interested in security practices. The employees, trained to be helpful, replied with extensive security details, including password length and change interval. One enthusiastic manager even enabled further data collection by forwarding this survey to colleagues elsewhere within the organisation.

4.1.7. Definition for attack

There is no single "correct" definition of the word attack across social, technical and physical contexts.

It is suggested that: "An attack is the same in a social, technical or physical context. The idea is to achieve a malicious goal, gain knowledge or get access to confidential data.

The social, technical or physical context only provides the means to achieve such a goal. Different methods are used depending on the context in which the attack is executed, but the objective is always the same."

Estimation of success usually depends on personal experience and the data gathering steps when planning the attack. In planning, the attacker will also estimate the difficulty and likelihood of getting caught. One opinion is that a skilled attacker may be more likely to care about being detected, so may plan countermeasures to avoid this.

4.1.8. Influence factors

There are many factors that influence difficulty and the likelihood of a successful attack. Also a "bit of luck" may sometimes be needed by an attacker. One can see from the interviewees' answers that each points out different factors, although time in relation to social engineering side is a common theme across the interviews.

Also, maturity of the system under attack and defensive measures are mentioned as common factors.

Some other factors mentioned in the responses as playing an important role include: security awareness of employees, security policies, the aspect of time, impact of a vulnerability, likelihood of success (estimated by the attacker), and especially the skillset of the attacker.

4.1.9. Attack strategy

On most occasions strategy changes on the run, based on the results from current/previous steps and findings — this is related mainly to the technical attacks. There are many factors influencing the change — discoveries on the way, attack detection, unexpected set-backs, etc.

On the social engineering side, it is thought that the strategy may be quite static — the attack scenario is put in place and the attacker tries to stick with it. When the victim becomes suspicious the attacker might change the approach to achieving the goal, but the underlying strategy stays the same.

4.1.10. Attack detection

Attackers want to stay undetected, which helps in reaching the goal faster, more efficiently and getting want they want. Low-level attackers might not think and plan in this way. Upon being detected, the attacker may still continue because the chance of getting what they want is within reach, since applying countermeasures will still take the organisation some time. When planning the attack, a higher level attacker also plans "getting out" of the system, wipes the logs and installs specific toolkits to hide any traces (deleting specific information from the logs).

4.1.11. Sequence of steps

When talking about how to construct a sequence of steps leading to your goal in the target enterprise, four out of six interviewees said that the attacker chooses the path or way of attack based on personal experience, preferences, confidence and best effort. This may be true even if this path seems more “expensive” compared to other possible paths.

There are some tools (for example i2, used by intelligence agencies) that can be used, but from the answers we can see that this is probably not the preferred approach for most attackers.

4.1.12. Common patterns

It is clear that different weak spot patterns and configuration/topology flavours exist in different organizations. The level of weakness depends on the maturity and/or size of the company, as well as its function (bank, hospital), etc.

Where patterns of weak spots exist and are also known to the attackers, they can be exploited. The patterns selected largely depend on the attacker’s profile and goal. From the social perspective - people are still frequently the weakest link.

4.2. Conclusions

The main aim of this chapter is to provide an overview of the profiles of attackers, their motivation, technical capabilities and social engineering skills. Our aim was to collect practical information for the development of the TRESPASS tools and processes.

In order to gather such requirements, we conducted structured interviews with six security experts from different locations and companies across Europe.

Profiling attackers is a very complex challenge, analysing the way an attacker thinks and acts to accomplish a particular goal. Based on the responses received, as well as from practical experience and knowledge, the attacker may not have a particular goal in mind, but rather determine this ’on-the-fly’.

If the attacker has a goal, a planning phase is likely, although these plans are likely to remain flexible. The majority of interviewees agree that the attacker makes many decisions based on the situation at hand, their own experience and instinct. Beyond this, it is admitted that sometimes a bit of luck is also needed.

4.2.1. Proposed attacker profiles

It is possible to suggest three separate profiles for attackers based on the interview results. These profiles are defined based on the answers received, where properties like motivation, strategy, resources and capability are described differently for different attacker profiles by interviewees. These conclusions are described in the previous chapter and have been analysed to propose as attacker profiles.

All properties mentioned above are in one way or another connected to the money and time available to the attacker. The goal the attacker wants to achieve and the means of achieving it are also very important.

If the attacker has greater resources, better and specific knowledge or a well designed strategy, the probability of succeeding in an attack is higher and depends on the profile used.

The profiles described should show how the attacker acts and how he or she chooses an attack path. By knowing this we are able to predict what resources the attacker will need in order to succeed. This should give us information about the probability of success.

Based on those descriptions we are able to propose the following profiles:

1. **Profile A: individual** who tries some automatic scripts, scans to discover known vulnerabilities with motivation just to test the system. Most likely uses manuals found online;

    - **Resources:** The attacker doesn't need specific resources at this level. As stated by an interviewee: "To follow a strategy, an attacker will spend a certain amount of resources." Since here the attacker does not try to do new things, but only to run prewritten automatic scripts, resources are not vital. The social-engineering element is minimal or absent altogether.
    - **Motivation:** Motivation comes from the attacker's personal agenda – mainly from the learning perspective, to see how scripts work and what they do. Motivation is therefore largely scattered -– gaining access to systems for botnets or harvesting personal information. This is a low skill process.
    - **Capabilities:** In this profile the attacker has no specific capabilities. The attacker wants to learn new things, techniques and gain more experience for the future.
    - **Strategy:** No specific strategy exists, because the attacker just tries basic approaches to see what can be achieved.

2. **Profile B: hacktivist** who has a specific goal to achieve glory, fame or just to mine data to find confidential information;

- **Resources:** The hacktivist's main resources are time and knowledge. In addition, the motivation to try new things drives the attacker to spend more time in working on attacks. Data gathered for a particular goal may also be regarded as a resource for future activities.
- **Motivation:** The first motivating factor for this profile is definitely fame, glory, new knowledge and the opportunity to find some confidential information. This is connected to the possibility of making money. It is not specific to an individual organisation. The attacker finds random targets to fulfill the goal of succeeding in a specific action, rather than a specific attack. Also a hacker's reputation — a strong motivation might also be to become famous as a hacker or as a hacking group.[1]
- **Capabilities:** This attacker has some knowledge about coding, is able to discover different vulnerabilities even if they are not specific to the current attack. This knowledge may be used in the future. The attacker has gained previous experience and knowledge from hacking.

  Attacker is also able to use social-engineering to gain information about an organisation's security policies. Any information gained is considered useful.
- **Strategy:** In this profile, the attacker may use exploratory activities which are not specific to the attack. The attacker just explores options and when something is found, an attack may be built in the future based on those findings. This approach can also be used for social-engineering.

  There might be a certain strategy built up for an attack, but most likely the attacker will not stick to it and when necessary will make decisions on the fly based on previous experience and knowledge.

3. **Profile C: a person or a group with higher skills** who are able to write code and have specific security related knowledge, who may want to gain and maintain access with a very specific purpose in mind. This profile can also include state driven attacks.

   - **Resources:** In the case of black hat hackers, the resources, especially those related to time, are unlimited. Hackers can spend as much time as they like to gather all the necessary information in order to thoroughly prepare their attacks.
   - **Motivation:** Usually the goal is to get possession of some confidential information. Hence the main driving force of the attackers is financial gain.

     Another possible goal is to disrupt the reputation of a victim and activism — breaking a company/organisation down, in order to cause as much damage to the victim as possible. This usually implies loss of reputation for the company, due to publication of sensitive confidential information.

     Also (industrial) espionage is an important driving force of the attackers.

[1] See, e.g. http://en.wikipedia.org/wiki/LulzSec

Similar to "Profile B" reputation is a strong motivation – just to show their capabilities.

- **Capabilities:** Higher skills and hacking frameworks are used to support the process of gaining and maintaining access, coding skills and security knowledge are required. Attacks tend to be more targeted, driven by a specific aim. Attackers can write exploits. They can very effectively use social-engineering skills to get information needed for attack and connect this data with weak spots in a company to build up a strategy for attack.
- **Strategy:** Attacks are mostly targeted. Strategy is thorough and very often used. The strategies are definitely adaptive and may change on the fly depending on findings on the way, for instance if something goes wrong or the victim becomes aware of the attack. Attackers need creativity to overcome these obstacles.

5. The TRESPASS risk assessment framework

This chapter outlines the basics of a risk assessment framework that meets the requirements for use of the tools in a practitioner's workflow. The risk assessment framework should be suitable for comparing social, digital, and physical aspects of cyber security, and it should enable the (automated) integration of data from the associated variety of sources.

Risk is often conceived as a combination of likelihood of an event, and the negative impact of such an event. Estimating likelihood requires a great deal of experience and professional judgement in the current state-of-the-art. TRESPASS processes will improve such assessments by automated tools as well as decision support, using information from the security models. As discussed below, this is the main area of innovation in the project, and the key goal of the "attack navigator" concept. Estimating impact may also be challenging, and should include assessment of asset value as well as business impact of asset compromise. In this area, TRESPASS will leverage existing knowledge, rather than providing fundamentally new concepts.

5.1. State of the art – baseline

There are a number of existing approaches to formalising risk assessment methodologies, and all of them having their merits and shortcomings. International standardisation organisations like ISO and IEC have published several standards that refer to risk assessment, e.g., ISO/IEC 13335 and 27005, ISO 15804 (The Common Criteria), and IEC 61025. Being standards, all of them are stated in very general terms and are not well suited to building socio-technical models meant for quantitative analysis and predictive risk assessment.

Various industrial entities have developed their own risk assessment methodologies suitable for their own domain. The fault tree approach, for example, was developed within the aviation community (US Air Force, Boeing) already in the 1960s, and later accepted in other fields as well (nuclear energy sector, NASA). A more general approach was initiated by Weiss in the late 1980s (Weiss, 1991) and later introduced to the IT sector by Bruce Schneier (Schneier, 1999, 2004) under the name attack trees. Both of these approaches concentrate on building the actual failure/attack model and not describing the underlying socio-technical state. From the viewpoint of these approaches, TREsPASS starts one layer lower, concentrating on the socio-technical model, from which the attack model

can later be generated. However methods of quantitative analysis developed for attack trees (Mauw & Oostdijk, 2005; Jürgenson & Willemson, 2007, 2008, 2010; Bagnato, Kordy, Meland, & Schweitzer, 2012; Kordy, Mauw, Radomirović, & Schweitzer, 2012; Kordy, Mauw, & Schweitzer, 2012) can be used on these models as well.

Several research groups have developed their own tool-supported risk assessment methodology, e.g., CORAS (Lund, Solhaug, & Stølen, 2011; Dahl, Hogganvik, & Stølen, 2007; Brændeland, Dahl, Engan, & Stølen, 2007) and SHIELDS (Byers & Shahmehri, 2010). The SHIELDS project concentrated on the model sharing aspect with its Security Vulnerabilities Repository Service (SVRS). However, this service has now been shelved, and even when operational it did not address the issue of finding an acceptable abstraction level of the models suitable for public sharing. This shortcoming will be addressed by the TRESPASS processes. The CORAS project received European funding and reached further in its results, producing a complete methodology for building risk models and taking heuristic assessments into account.

The CORAS security risk analysis consists of eight different steps (Lund et al., 2011), where the first four steps focus on context establishment and the last four steps are about risk identification, estimation, evaluation and possible risk treatments:

1. **Preparations for the risk analysis:** In order to prepare the risk analysis, the main objectives of this step are to define the scope and to estimate the size of the project.
2. **Customer presentation of the target:** This step consists of an introductory meeting with the customer. The main item on the agenda is a presentation of the responsible persons of the customer, revealing their general objectives and expectations and the exact scope of the risk analysis. This has the aim to give a common understanding of the scope and to identify what the targeted organisation is worried about.
3. **Refining the target description using asset diagrams:** The goal of step 3 is to ensure a common understanding of the focus, the scope and the main assets. For this, the analysis team recapitulates the main results of the first meeting and from the readings of the company documents. For modelling the target of the analysis, CORAS uses the Unified Modelling Language (UML). Additionally the main assets to be protected are identified based on the interaction with the customer and a rough high-level analysis is conducted to identify major threat scenarios, vulnerabilities and enterprise risk levels.
4. **Approval of the target description:** Step 4 concludes the context establishment and includes as task the detailed description of the scope of the risk analysis by using a formal or semi-formal notation such as the UML. The description should be approved by the customer before moving on to the next step. Besides, the definition of the risk evaluation criteria for each asset is also done during this step.
5. **Risk identification using threat diagrams:** Step 5 includes the identification of possible risks by organising a brainstorm meeting with participants which have different competences in order to identify as much risks as possible. The risk identification includes the identification of threats, unwanted incidents, threat scenarios and

vulnerabilities with reference to the identified assets. The results will be documented with the help of CORAS threat diagrams, part of the CORAS language.

6. **Risk estimation using threat diagrams:** Step 6 takes the results form Step 5 in order to define the level of the risks. Step 6 is, similarly to step 5, also conducted as a brainstorming with participants having different competences in order to estimate the likelihoods and consequences of unwanted incidents.
7. **Risk evaluation using risk diagrams:** Step 7 consists in evaluating if the identified risks are acceptable or not. The evaluation is done by using the risk evaluation criteria, defined during the context establishment and the results of the risk estimation of step 6.
8. **Risk treatment using treatment diagrams:** The aim of step 8 is the identification of risk treatments for risks which are classified as not acceptable. The different risk treatments are chosen with respect to a cost-benefit analysis.

Further, the CORAS method provides a computerised tool designed to support documenting, maintaining and reporting analysis and results through risk modelling. In summary, the CORAS tool is a diagram editor that is available for free which can be used to draw the different CORAS diagrams (asset diagrams, threat diagrams, risk diagrams and treatment diagrams). It is a diagram editor based on Eclipse (platform independent) which is easy to install and use. The CORAS tool differs from other risk assessment tools because it is only a supporting tool for the CORAS method.

Still, the models of CORAS share general shortcomings of attack models (including attack and fault trees):

1. They are subjective and onerous to maintain.
2. They are not well-suited for discovering previously unknown complex attacks.
3. They lead to repetition of the same efforts in different organisations.
4. More extensive assessments than checklists are often too costly for smaller organisations.

The TRE$_S$PASS processes will aim at improving upon these limitations based on the attack navigator concept.

5.2. Fundamentals

As discussed above, TRE$_S$PASS aims at enabling security risk management in a socio-technical context, moving beyond expert judgement by integrating data from different sources in a navigator map. We therefore need a conceptual framework that can address social, digital, and physical aspects of risk in the same terms and quantities. We consider both risk assessment, which involves defining and evaluating the risks, and risk treatment, which involves proposing and implementing countermeasures. Both risk assessment and risk treatment are considered parts of risk management.

In order to enable quantitative risk calculations, necessary for prioritising attack scenarios and countermeasures, the first choice made by the TRE$_S$PASS approach is to separate clearly between system properties and outside events. The notion of likelihood often incorporates both system properties and attacker behaviour, and therefore needs to be disentangled. For example, the same electronic voting system may have a different likelihood of attack in a stable democracy than in a country at war. This is not due to the system itself, but to the environment, or the attacker model. Without clearly separating the two factors, likelihood cannot be defined precisely.

Additionally, it should be recognised that likelihood needs to be bound to time. One cannot say "the likelihood of this system being attacked is 0.5", because this may refer to today, this week, or to an infinite future. To avoid this problem, we preferably speak about the expected frequency rather than the likelihood of events. One can then, for example, say that the expected frequency of a certain attack is once per year. The experiments on data acquisition (see Section 3.5 of D2.1.1), conducted by TRE$_S$PASS consortium members Cybernetica and University of Luxembourg, also provide reasons for choosing frequency over likelihood. In these experiments, it appears to be hard for a human to quantitatively estimate likelihood with the required precision. Frequency of an event, on the contrary, seems to be closer to human comprehension and thus can be estimated by people in a more native way than event likelihood. Therefore we derive the hypothesis that event frequency estimations collected from human input will be more precise than data obtained for event likelihood. Both theoretical and practical arguments thus exist for using frequencies.

Given these constraints, we have chosen to base the TRE$_S$PASS risk assessment framework on the Risk Taxonomy published by The Open Group (The Open Group, 2009). This has the additional advantage that a link can be established with another Open Group standard, ArchiMate, used in WP1. In the taxonomy, a clear distinction is made between the following:

- Threat Event Frequency, the frequency with which certain attack scenarios occur;
- Vulnerability, the likelihood that an initiated scenario succeeds;
- Probable Loss Magnitude (also called impact), the damage that occurs when an attack scenario succeeds.

These concepts can be applied to any kind of threats and systems, making it possible to integrate the different domains in a single framework. From the threat event frequency and the vulnerability, the *loss event frequency* can be calculated by multiplication, indicating the expected number of loss events per unit of time. For example, if certain attacks are expected to happen once per year, and the likelihood of success of the attack (and thus loss for the organisation) is 50%, the loss event frequency is once every two years. According to the Open Group, risk comprises both loss event frequency and probable loss magnitude, although they do not provide a formula. We use the common approach here, and define risk as the *product* of loss event frequency and probably loss magnitude. This yields the risk expressed in terms of euros per year, or Annual Loss Expectancy (ALE).

In malicious attacks, the first item in the list (threat event frequency) is actually the last to be calculated. Only when vulnerability and impact have been assessed can one estimate frequencies, as an attacker will typically estimate vulnerability and impact himself, and base the decision which scenario to attempt on this. The calculation of frequencies therefore needs to take the attacker model (WP2, Section 5.6 of D2.1.1) into account.

The Open Group rightly expresses that vulnerability depends on the strength of both the threat (attacker) and the defences. This paves the way towards quantitative assessment of this relation. However, the precise calculations remain unclear in the taxonomy. A possible solution has been proposed based on Item Reponse Theory (Pieters, Van der Ven, & Probst, 2012), and the TRE$_S$PASS risk management processes will need to build further on such insights, in close cooperation with system modelling (WP1), data management (WP2), and model analysis (WP3).

5.3. Multi-step attacks

The risk definitions of The Open Group do not specifically address multi-step attacks, in which the attacker takes several steps to achieve his or her goal. Whereas natural or accidental threats typically happen either independently or in a sequence of cascading failures, malicious attacks are different. The attacker may intentionally induce several problems simultaneously or in sequence, thereby increasing the chance to be successful. Such steps may involve remote access, physical access, and social engineering. In order to improve the completeness of the risk assessment methods, including multi-step attacks is therefore essential.

From the data management processes, we will typically acquire data for individual attack steps. For example, risk variables may be defined for phishing for usernames and passwords, for misuse of usernames and passwords to get access to secret data, and for shipping secret data to a competitor, but it is not immediately clear what the risk would be of an attack in which all these steps are combined.

Therefore, in order to make the TRE$_S$PASS tools and processes suitable for risk management, they need to relate quantitative properties of individual steps to quantitative properties of complete attack scenarios, which may be represented as attack trees. This includes in particular notions of defence strength (difficulty), and vulnerability (likelihood of successful attack). Existing methods for performing calculations on attack trees may provide the basis for risk assessment for multi-step attacks, but they would need to be extended to support navigator maps and attacker models. This will be mostly a task of WP1 and WP3, but the processes will have to take this distinction into account, while keeping the benefits of the Open Group Risk Taxonomy. This is for example the case in identifying countermeasures that would be effective against more than one multi-step attack, because the step against which they prevent is present in all of them.

5.4. Socio-technical risks

As indicated by our interviews, integration of the various security domains (social, physical, digital) is one of the important issues, and therefore an essential part of the project. Not only do these domains need to be integrated in terms of defining security architectures, but to make investments cost-effective, they also need to be comparable in terms of the risk that they prevent.

Therefore, the TRE$_S$PASS processes would need to provide an integrated view on social and technical aspects on attacks. This should enable risk assessment of multi-step, cross-domain attacks, as well as comparisons between investments in the domains. To this end, notions like defence strength or difficulty need to be defined in such a way that they apply to both hacking and social engineering steps, and they enable calculations on scenarios that include both. The data management framework (WP2) will need to support this. This enables the TRE$_S$PASS audit processes to transcend the “silos” of the different domains.

5.5. Countermeasures and prevented risk

The TRE$_S$PASS tools will be able to suggest weak spots – entities that contribute risk to many scenarios in the socio-technical system – but they will not be able to propose concrete countermeasures, as this requires much more domain knowledge. Therefore, the TRE$_S$PASS processes will require the user to identify the weaknesses using visualisations presented by the TRE$_S$PASS system. As the next step, the user will need to introduce countermeasures as changes to the navigator map.

The user can also select meaningful combinations of countermeasures, to be evaluated by the tools. The system will then calculate the required parameters, for instance, the prevented risk and cost-effectiveness, for each of the selected sets of countermeasures. These results then need to be taken into account in the reporting phase.

5.6. Conclusions

The TRE$_S$PASS risk assessment framework builds upon the Risk Taxonomy of The Open Group, as it clearly separates external events from internal ones, and recognises that likelihood of an attack is always bound to time, and should rather be called frequency. Moreover, the taxonomy is general enough to cover physical, social and digital aspects of cyber security risks.

In TRE$_S$PASS, the framework needs to be extended to support:

1. quantitative assessment of vulnerability in relation to threat capability and control strength.

2. support for multi-step attacks.
3. precise definition of the quantitative concepts for the different domains.
4. identification and evaluation of countermeasures.

6. Central requirements

Chapters 3 and 4 presented the requirements gathered from the sources external to the TRESPASS project. However, there are also a number of requirements coming to WP5 from other work packages, and vice versa.

In order to coordinate the requirements gathering process between the work packages, a special task force was established within the TRESPASS project. The task force collected and synchronised all the dependencies between the work packages, and a big central requirements table was created. The whole table is released as a part of Deliverable 6.2.2 (The TRESPASS Project, D6.2.2, 2015).

The requirements contain the following components:

Identifier : Unique identifier id taken from requirement table;

Requirement : Specification of requirement

Source WP: Source WP is the originator of the requirement which deem the requirement as functionally relevant to fulfil its own tasks.

Target WP: Target WP is the WP that first accepts the requirement, evaluates it and then mutually agree with source WP to fulfil the requirement in a time bound manner.

Goals: They serve as justification of the requirement from the origin WP and the basis for target WP to understand and respond it.

Acceptance criteria: Source WP sets certain criterion over which a raised requirement by it needs to be fulfilled and also serves as benchmark for target WP to complete it.

Status: This refers to the current position of the requirement if it has been already completed or is in process of acted on, in accordance to the acceptance criterion of the source WP

Dependencies: It refers to the cross links that the requirement imposes. To be precise, on its completion, which other requirements or tasks are progressed.

6.1. Requirements targeted towards WP5

Requirement R03

Requirement : Availability of an attack pattern library.

Source WP: WP6

Target WP: WP5

Goals: Reuse of sub-attack-trees

Acceptance criteria: One example per case study.

Status: Agreed

Dependencies: None

Requirement R13

Requirement : Define a visualisation process.

Source WP: WP4

Target WP: WP4,5,6

Goals: Needed for development of interface

Acceptance criteria: Documented process accepted by reviewers in Deliverable 4.1.1 (achieved) Accepted interpretation by practitioner panel and data analysis designers (WP) in Attack Navigator Map (iterative and on-going)

Status: Agreed

Dependencies: None

Requirement R14

Requirement : Develop visual thinking tools.

Source WP: WP4

Target WP: WP2,4,5

Goals: Needed to produce meaningful analysis results

Acceptance criteria: Documented thinking tools accepted by academic publication. peer reviewers (achieved), documented thinking tools accepted by project reviewers in deliverable 2.3.2 (on-going) tools accepted by practitioner panels (iterative and on-going)

Status: Agreed

Dependencies: None

Requirement R26

Requirement : It should be possible to generate attack scenarios from the model.

Source WP: WP7

Target WP: WP3,5

Goals: Needed for case studies

Acceptance criteria: Generation of attack scenarios from the telco models.

Status: Agreed

Dependencies: R26

Requirement R27

Requirement : The TREsPASS tool should provide solutions or mitigation strategies for attack scenarios.

Source WP: WP7

Target WP: WP3,5

Goals: Needed for case studies

Acceptance criteria: Mitigation strategy for the identified attack scenario.

Status: Shelved

Dependencies: Reason for shelving: in line with the discussion that occurred in WP5 (as put forward by Jan Willem during the MT meeting), and the points raised by the advisory board, that the scope of our work is not ?impact analysis?.

Requirement R36

Requirement : Due to a persistent increase in product complexity and diversity on the feature side and margin pressure on mass products, risks cannot be eliminated, but need to be assumed and accepted instead, whilst minimising their possible impact. Hence, the focus should be the question of calculability.

Source WP: WP7

Target WP: WP5

Goals: Needed for case studies

Acceptance criteria: A tool/technique for prioritizing the risks/attack scenarios.

Status: Agreed

Dependencies: None

Requirement R37

Requirement : Identification of required analysis measures and quantities

Source WP: WP3

Target WP: WP5

Goals: In order to develop dedicated analysis methods, we need a clearer idea of the kind of properties to be analysed. This is needed for Task 3.3.

Acceptance criteria: Agreed-upon list of measures/quantities.

Status: Completed

Dependencies: Task 3.3 depends on this.

Requirement R56

Requirement : The system can suggest updates to the TRESPASS model based on scenarios and associated parameters.

Source WP: MT

Target WP: WP3,5,6

Goals: Derived from P1 (TRESPASS tools should assist the user in building a navigator map). Supports Use-case 1 (Security Investment), Use-case 2 (Audit) and Use-case 5 (Quick scan), specifically U1.12, U1.13, U2.4 U5.6.

Acceptance criteria: 80% of the users specified in the Use Cases are able to respond effectively to the suggested updates.

Status: Shelved

Dependencies: Reason for shelving: adaptations to the sociotechnical model will have to be done manually, not automagically.

Requirement R61

Requirement : User can select parts of the model to share with a specified group of other users.

Source WP: MT

Target WP: WP5 (backend).

Goals: Derived from P1 (TRESPASS tools should assist the user in building a navigator map). Supports Use-case 1 (Security Investment), Use-case 2 (Audit) and Use-case 5 (Quick scan), specifically U1.16, U2.15 and U5.17.

Acceptance criteria: Only explicitly selected parts are made available.

Status: Agreed

Dependencies: None

Requirement R62

Requirement : Users can add parts of shared models to their own model.

Source WP: MT

Target WP: WP5

Goals: Derived from P1 (TRESPASS tools should assist the user in building a navigator map). Partly supports Use-case 2 (Audit), Use-case 4 (Product-service system) and Use-case 5 (Quick scan), specifically U2.2, U4.2 and U5.4.

Acceptance criteria: 80perc. of the users specified in the Use Cases are able to use the interface effectively after receiving the specified training.

Status: Shelved

Dependencies: None

Requirement R67

Requirement : The system can retrieve standard attacker profiles by name.

Source WP: MT

Target WP: WP5

Goals: Derived from P1 (TRESPASS tools should assist the user in building a navigator map). Supports Use-case 5 (Quick scan), specifically U5.3

Acceptance criteria: 80% of the users specified in the Use Cases are able to use the interface effectively after receiving the specified training.

Status: Agreed

Dependencies: Assumes the attack pattern library exists.

Requirement R75

Requirement : The tools can calculate the total risk associated with a ranked set of scenarios and a set of attacker profiles.

Source WP: MT

Target WP: WP3,5

Goals: Derived from P4 (TRESPASS tools should be able to rank attack scenarios in a risk-based prioritisation). Supports Use-case 1 (Security Investment), Use-case 2 (Audit), Use-case 4 (Product-service system) and Use-case 5 (Quick scan), specifically U1.11, U2.9, U4.4 and U5.11.

Acceptance criteria: The output is consistent with the mathematical definition.

Status: Completed

Dependencies: None

Requirement R86

Requirement : Users can select base models from a Model Template Library.

Source WP: MT

Target WP: WP4,5,6

Goals: Supports Use-case 4 (Product-service system) and Use-case 5 (Quick scan), specifically U4.2, U5.4

Acceptance criteria: 80perc. specified in the case studies are able to use the interface effectively after receiving the specified training.

Status: Agreed

Dependencies: Assumes the Model Template Library exists.

6.2. Requirements originating from WP5

Requirement R07

Requirement : Ability to generate an attack tree.

Source WP: WP5

Target WP: WP3

Goals: Needed to proceed with Task 5.3

Acceptance criteria: The tool exists and runs correctly.

Status: Agreed

Dependencies: None

Requirement R08

Requirement : Parameter values for the trees in APL.

Source WP: WP5

Target WP: WP2

Goals: Usability of APL depends on it. Needed to produce meaningful analysis results.

Acceptance criteria: Parameter values are provided and look reasonable.

Status: Agreed

Dependencies: None

Requirement R20

Requirement : The model should support macros to ease modelling of, e.g., complex repeated properties and domain-specific extensions.

Source WP: WP5-7

Target WP: WP1

Goals: Needed for case studies

Acceptance criteria: All the reasonable system components that I can think of can be modeled using the model extension framework.

Status: Agreed

Dependencies: None

7. Conclusions

This deliverable focused on gathering and systematising the requirements for TRE$_S$PASS frameworks and workflow.

We have analysed existing industry processes by conducting structured interviews with security practitioners from around Europe, which enabled us to identify current industrial needs and opportunities for improvement. Earlier work in WP5 compared the proposed TRE$_S$PASS risk management process to existing frameworks. Ongoing work will be focused on more detailed design of the TRE$_S$PASS processes, more detailed investigation of the current standardised methodologies, including identifying possible constraints on the representation of socio-technical aspects in existing standards, as well as identifying opportunities for further integration.

References

Bagnato, A., Kordy, B., Meland, P. H., & Schweitzer, P. (2012). Attribute Decoration of Attack–Defense Trees. *International Journal of Secure Software Engineering, Special Issue on Security Modeling*, *3*(2), 1–35. doi: 10.4018/jsse.2012040101

Brændeland, G., Dahl, H. E. I., Engan, I., & Stølen, K. (2007). *Using dependent coras diagrams to analyse mutual dependency.*

Bsi. (2013). Retrieved from https://www.bsi.bund.de/EN/Home/home_node.html

Burp. (2013). Retrieved from http://portswigger.net/burp/

Byers, D., & Shahmehri, N. (2010). Unified modeling of attacks, vulnerabilities and security activities. In *SESS '10: Proceedings of the 2010 ICSE Workshop on Software Engineering for Secure Systems* (pp. 36–42). New York, NY, USA: ACM. doi: http://doi.acm.org/10.1145/1809100.1809106

Cannon, D. L., Bergmann, T. S., & Pamplin, B. (2006). *CISA Certified Information Systems Auditor Study Guide*. Wiley Publishing, Inc.

CISA Review Manual 2012. (2011). (ISACA, http://www.isaca.org)

Cobit. (2013). Retrieved from http://www.isaca.org/cobit

Coso. (2013). Retrieved from http://www.coso.org

Dahl, H. E. I., Hogganvik, I., & Stølen, K. (2007). *Structured semantics for the coras security risk modelling language.*

Europrise. (2013). Retrieved from https://www.european-privacy-seal.eu/

Ibm lotus notes. (2013). Retrieved from http://www-142.ibm.com/software/products/us/en/ibmnotes

Iske. (2013). Retrieved from https://www.ria.ee/iske-en

Iso 2700.* (2013). Retrieved from http://www.27000.org/

Iso 3100.* (2013). Retrieved from http://www.iso.org/iso/home/standards/iso31000.htm

Iso review process. (2013). Retrieved from http://www.finsys.umn.edu/sales/sls_internal_reviewprocess.pdf

Jürgenson, A., & Willemson, J. (2007). Processing Multi-Parameter Attacktrees with Estimated Parameter Values. In A. Miyaji, H. Kikuchi, & K. Rannenberg (Eds.), *Iwsec* (Vol. 4752, pp. 308–319). Springer. doi: 10.1007/978-3-540-75651-4_21

Jürgenson, A., & Willemson, J. (2008). Computing Exact Outcomes of Multi-parameter Attack Trees. In R. Meersman & Z. Tari (Eds.), *Otm conferences (2)* (Vol. 5332, pp. 1036–1051). Springer. doi: 10.1007/978-3-540-88873-4_8

Jürgenson, A., & Willemson, J. (2010). On Fast and Approximate Attack Tree Computations. In *Proceedings of the 6th international conference on information security practice and experience* (pp. 56–66). Berlin, Heidelberg: Springer-Verlag. Retrieved from http://dx.doi.org/10.1007/978-3-642-12827-1_5 doi: 10.1007/978-3-642-12827-1_5

Kordy, B., Mauw, S., Radomirović, S., & Schweitzer, P. (2012). Attack–Defense Trees. *Journal of Logic and Computation*. (Available at http://logcom.oxfordjournals.org/content/early/2012/06/21/logcom.exs029) doi: 10.1093/logcom/exs029

Kordy, B., Mauw, S., & Schweitzer, P. (2012). Quantitative Questions on Attack–Defense Trees. In *Icisc* (Vol. 7839, pp. 49–64). Springer.

Lund, M. S., Solhaug, B., & Stølen, K. (2011). *Model-driven risk analysis: the CORAS approach*. Springer Berlin Heidelberg. Retrieved from http://www.springer.com/computer/swe/book/978-3-642-12322-1

Maltego. (2013). Retrieved from http://www.paterva.com/web6/

Mauw, S., & Oostdijk, M. (2005). Foundations of Attack Trees. In D. Won & S. Kim (Eds.), *Icisc* (Vol. 3935, pp. 186–198). Springer. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.97.1056

Metasploit. (2013). Retrieved from http://www.metasploit.com/

Microsoft sharepoint. (2013). Retrieved from http://office.microsoft.com/en-us/microsoft-sharepoint-collaboration-software-FX103479517.aspx

Nessus. (2013). Retrieved from http://www.tenable.com/products/nessus

Nikto. (2013). Retrieved from http://www.cirt.net/nikto2

Nmap. (2013). Retrieved from http://nmap.org/

Openvas. (2013). Retrieved from http://www.openvas.org/

Owasp. (2013). Retrieved from https://www.owasp.org/index.php/Main_Page

Pieters, W., Van der Ven, S. H. G., & Probst, C. W. (2012). A move in the security measurement stalemate: elo-style ratings to quantify vulnerability. In *Proceedings of the 2012 workshop on new security paradigms* (pp. 1–14). New York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/2413296.2413298 doi: 10.1145/2413296.2413298

Rfc 2196. (2013). Retrieved from http://www.ietf.org/rfc/rfc2196.txt

Sap. (2013). Retrieved from http://www.sap.com/index.epx

Schneier, B. (1999). Attack Trees: Modeling Security Threats. *Dr. Dobb's Journal of Software Tools*, *24*(12), 21–29. Retrieved from http://www.ddj.com/security/184414879

Schneier, B. (2004). *Secrets & Lies: Digital Security in a Networked World*. Indianapolis, Ind.: Wiley.

The Open Group. (2009). *Risk taxonomy* (Tech. Rep. No. C081). The Open Group. Retrieved from www.opengroup.org/pubs/catalog/c081.htm

The TRESPASS Project, D2.1.2. (2015). *Final requirements for the data management process.* (Deliverable D2.1.2)

The TRESPASS Project, D5.1.1. (2013). *Initial requirements for process integration.* (Deliverable D5.1.1)

The TRESPASS Project, D5.2.1. (2014). *Currently established risk-assessment methods.* (Deliverable D5.2.1)

The TRESPASS Project, D6.2.2. (2015). *Final refinement of functional requirements.* (Deliverable D6.2.2)

The TRESPASS Project, D9.2.1. (2013). *Standardisation plan.* (Deliverable D9.2.1)

W3af. (2013). Retrieved from http://w3af.org/

Weiss, J. D. (1991). A system security engineering process. In *14th nat. comp. sec. conf.* (pp. 572–581).

Wireshark. (2013). Retrieved from http://www.wireshark.org/

A. Project Summary

This chapter gives an overview of the TRE$_S$PASS project and its use cases. The section is shared by the public deliverables to provide the necessary background and to put the current deliverable in context.

Information security threats to organisations have changed completely over the last decade, due to the complexity and dynamic nature of infrastructures and attacks. Attacks like StuxNet involve technical and human factors, and they damage physical infrastructure. The recent attack on a German steel mill [1] was a combination of both targeted phishing emails and social engineering attacks. The phishing helped the hackers extract information they used to gain access to the plant's office network and then its production systems. As a result, the technical infrastructure of the mill suffered severe damage.

The attack on the German steel mill illustrates that we need to integrate the social and technical aspects of systems in assessing their security - and we need to do so today. Socio-technical systems pose new challenges by combining parts for which we often understand the security issues; the combined system is however much more complex due to interactions between these parts.

The main innovation of the TRE$_S$PASS project is the attack navigator, a tool and metaphor that enables defenders to predict and preventing attacks on socio-technical systems. The attack navigator supports current risk-assessment techniques with the TRE$_S$PASS process (developed in Work Package WP5), an analytical approach to identifying attacks and evaluating their impact.

The four main stages in the TRE$_S$PASS process are *data collection*, *modelling*, *analysis*, and *visualisation*. Data collection (WP2) is vital to understanding the nature of a scenario and providing input to subsequent tasks of modelling, analysis and visualisation. Within the project, the focus has been on collection and analysis of social, technical and physical data and the ways in which these relate to one another. Within each of these domains, different approaches have been taken to provide different viewpoints on the nature of the organisation being investigated.

The models (WP1) developed in TRE$_S$PASS can be adapted to the application scenario. We have developed physical modelling techniques in order to understand where further investigation may usefully be targeted. The TRE$_S$PASS model describes relevant aspects of the organisation and their connections. To explore contractual and commercial relationships, the e3value method has been adopted.

[1] BBC News, *Hack attack causes 'massive damage" at steel works*, http://www.bbc.com/news/technology-30575104, last visited October 31, 2015.

The analysis methods (WP3) developed in TRE$_S$PASS identify attacks in models and identify the most effective controls to prohibit these attacks. The analyses are supported by tools and together they provide the defender with a comprehensive understanding of properties attacks, *e.g.*, cost for the attacker, required skills, or required time.

The innovative visualisations (WP4) developed in TRE$_S$PASS focus focus particularly on visualising elements of the analysis, as this is key to the overall project goal of providing "decision support" to practitioners. However, visualisations contribute also to model development and data gathering.

Practitioners can access the TREsPASS toolkit via the attack navigator map interface, which provides an intuitive means of selecting appropriate tools (WP6) for data gathering, modelling, analysis and visualisation. These can be used, individually or in combination, to strengthen operational and strategic decision-making.

A.1. Case Studies

The TREsPASS process and tools are validated by means of case studies (WP7) in the area of cloud infrastructure, telecommunications infrastructure, ATM infrastructure, and an organisation processing privacy sensitive data.

A cloud infrastructure shares infrastructure within or across organisations, giving the cloud services provider and its employees full physical and logical access to all resources across the different consumers. In TRE$_S$PASS we formalise typical components in cloud infrastructures as well as human actors and their interrelationships, to identify their contribution to attacks on the organisation.

In telco infrastructure new products need to be launched under significant time pressure, often opening up loopholes for so-called knowledge insiders who know the market very well, trying to make as much monetary gain from the new products as possible. In TRE$_S$PASS we model both the infrastructure and contractual relationships to identify physical and monetary attacks.

The ATM infrastructure connects machines that are composed of a money safe and a computer that controls the ATM's devices. There are well protected ATMs installed inside bank branches, while others are deployed in the street and some are not even embedded in a wall. ATM attacks are common and include classic physical attacks and emerging digital attacks. In TRE$_S$PASS we model ATM installations, and identify attack likelihoods using geospatial data.

The organisation processing privacy sensitive data develops a system supporting primarily elderly and disabled people in performing online payments and managing their own money from their home. This case study involves from strictly technical security aspects, such as how information is protected while stored or transmitted, to socio-technical security aspects covering security issues arising from the use of and interaction with the technology. In TRE$_S$PASS we identify social-engineering and trust-based attacks on such systems.

A.2. Overview of TRESPASS Integration

The TRESPASS workflow involves several stages with various activities, some of which are optional. Figure A.2 shows the architecture diagram and Figure A.1 shows a visual description of the notation used. In practice, stages may not follow a linear order. For example, depending on the goal of the risk assessment, new data requests may be issued later in the process, or automatic updates of data may be supported.

The **Data collection stage** prepares for analysis and modelling steps, and may require the gathering of one or more of the following kinds of data.

Physical data collection provides knowledge about the physical layout of the organization including locations, buildings, rooms, doors, windows, etc.

Digital data collection gathers information about the organization's IT infrastructure.

Social data collection focuses on organisational and individual data, and results in actor profiles containing, *e.g.*, attributes of employees, stakeholders, or potential attackers.

Commercial data collection gathers information required for *e3fraud* analyses, which focus on potential fraud.

Stakeholder goal collection identifies assets and policies the protection of which is critical to one or more stakeholders.

The **model creation stage** handles the creation of the TRESPASS model and associated actor profiles. The e3value model creation process is complementary to the main TRESPASS model, for cases requiring a more specific financial focus:

TRESPASS model creation is a key activity result in a system model that can be further extended and analysed.

- **Components customization (optional)** takes place before or during the TRESPASS model creation to create specialized custom model components.
- **Attacker profile creation** creates the attacker profile that the TRESPASS model analysis should consider, based on ready-made attacker profiles.
- **Defender/target profile creation** creates similar profiles for the other actors in the model based on the social data gathered in the social data collection activity.

e3value model creation This interactive activity involves using the *e3value toolkit* [2] to create business value models. These models structure the commercial information gathered in the data collection stage in a formal way.

In the **analysis stage** different analyses are possible depending on the model chosen. The analysis of the TRESPASS model involves these steps:

1. In the **attacker profile selection**, the user selects the attacker profile to use in the analysis.

[2] http://e3value.few.vu.nl/tools/

2. The **attacker goals creation** provides the attack generation with the attacker goals. These can be derived by hand from the stakeholder goals or deduced automatically from the selected attacker profiles.
3. The **scenario selection** selects a scenario, consisting of a single pair of attacker and attacker goal, to run the TRESPASS analysis on.
4. To extend attack trees, **attack pattern creation and sharing** provides libraries with known attack steps. The attack tree generation can only reach a certain level of abstraction, which may not be sufficient for quantitative analyses.
5. **Attack generation** transforms the TRESPASS model to an attack tree.
6. **Attack tree annotation & augmentation** then extends the attack tree with attack patterns and decorates leaf nodes with parameter values from the data collection stage for quantitative analysis.
7. The **attack tree analyses** compute quantitative properties of attacks, *e.g.*, utility for the attacker or success probability of the attack.

The analysis of the **e3value model** is complementary to the core TRESPASS analysis and has only one step:

1. For the **fraud model generation**, the user needs select an attacker and an interval of expected occurrence rates of the commercial transactions specified by the e3value model. The e3fraud tool then identifies all possible violations of contracts, the loss for actors, and the delta in profit for the other actors.

The **visualisation stage** can be used continuously to provide practitioners with feedback regarding the results of their activities:

1. **Fraud model visualisation** shows the generated attacks as a ranked list of textual descriptions of the attack steps and displays charts showing the profitability for each actor.
2. **Attack tree visualisation** shows the intermediary attack trees.
3. **Attack tree analysis visualisation** visualises analysis results.

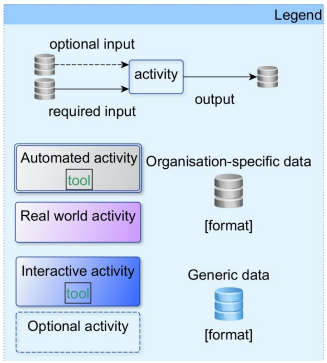

Figure A.1.: Legend for the Integration diagram in Figure A.2.

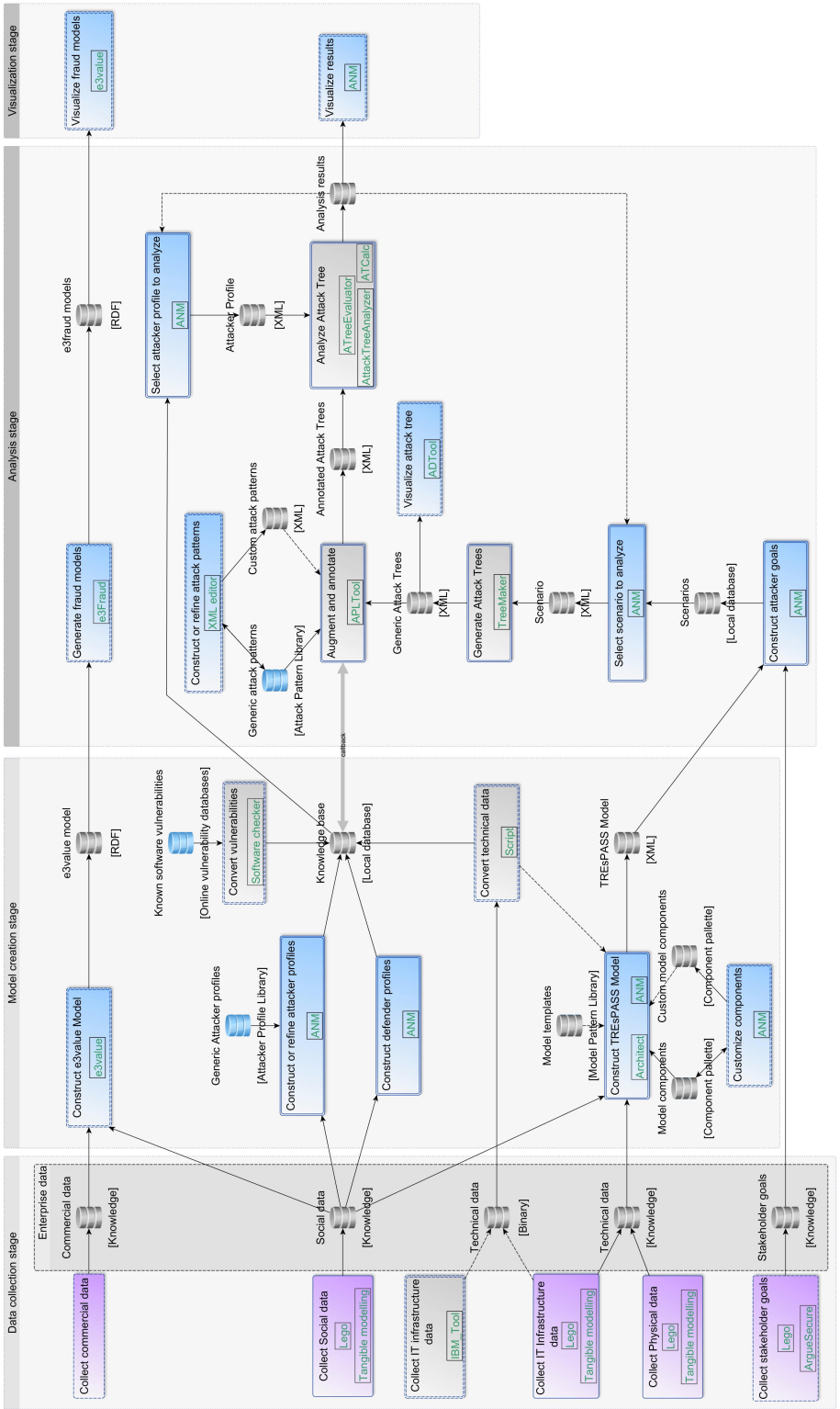

Figure A.2.: Integration diagram for the TRESPASS project.

B. The attacker questionnaire

B.1. The questionnaire

1. Please specify your field(s) of expertise
   - Social
   - Technical
   - Physical
   - Other ..... (please specify)
   - Educational background (self-educated, professional status, general level of education).
   - Country of residence/culture.
   - Respondent's age.
   - Respondent's social preferences: use of social networks, sharing achievements with others, motivation e.g. fame, profit, working preferences e.g. working alone or in a team of attackers.
   - Length of time the respondent has been involved in hacking.
   - Preference for self-education (hacker conferences, private forums).
   - Set of skills respondent feels most familiar and proficient with.
   - Self-assessment of respondent's own hacking skill level.
2. Which events, factors or variables could influence your motivation to attack, the nature of your goal and the resources you dedicate to achieving that goal?
3. Do you sometimes undertake exploratory activities without an initial target or goal in mind? If so,
   - Do you try to make use of everything you find on a target system, or do you have a different strategy?
   - After achieving your goal - do you quit, or perform some other actions? (*e.g. destructive actions, looking for other opportunities beyond your main goal etc.*).

4. How do you discover and identify social, technical and physical vulnerabilities to exploit? How do you identify potential targets for social engineering and the weak spots in technical and physical security? Can you list some typical vulnerabilities that are particularly easy to exploit? Do you use tools to support this discovery process?

    - Is the reconnaissance phase present in every attack or not? Which factors affect this decision?
    - How are technical and physical vulnerabilities discovered?
    - How are social vulnerabilities discovered? Is this process systematic, or not? Are any tools used for this? Ask for examples, e.g. “Can you list some typical vulnerabilities that are particularly easy to exploit? How can those be discovered?”

5. What criteria do you use to compare the weak spots in an organisation’s defences in order to discover the “weakest link”? Which factors do you consider, when evaluating whether a particular enterprise is a realistic target and if the attack is worthwhile?

    - Given the reconnaissance phase results, which factors are compared to make a decision regarding whether a physical, technical or social approach would be the most efficient?
    - Comparing several weak spots in the same domain (social, technical, or physical) – what are the criteria for comparing them and deciding which one will be the most efficient to exploit?
    - Given the reconnaissance phase results for the target enterprise and a goal – what are the subsequent steps that need to be undertaken to achieve the goal? Request examples.
    - Is the choice of those subsequent steps based on the respondent’s skill and experience, or are there some other factors that count? Request examples.
    - Which factors are taken into consideration when comparing several different paths leading to the same goal? How is the most efficient path chosen, or is there no particular preference?

6. Please describe some typical cases involving social engineering. Can you suggest any systematic ways to protect a system against social engineering attacks?

    - How would the attacker estimate the likelihood of success and difficulty to accomplish the chosen set of actions that lead to his goal? Value domain, scale, etc. Request examples.
    - Do attack difficulty and likelihood of success depend solely on defensive measures and the attacker’s skills, or are there other important considerations?
    - Are there any other parameters that we have not mentioned and that could be estimated for an attack?

- Which factors are considered, when evaluating whether a particular enterprise is a realistic target and if the attack is worthwhile? How are the difficulty of attack and likelihood of success estimated in this case? Request examples.

7. How would you define the term *attack* in social, technical and physical contexts? What methods do you use to estimate the likelihood of success and difficulty of an attack? Could you name any other parameters that you estimate when preparing an attack? Which signs/flags indicate this? Can you provide any examples?

   - Does the attacker care about being detected and to what extent?
   - How would the attacker estimate the likelihood of being detected? Ask for examples.
   - What are the preferable actions upon being detected? I.e. stop attacking, continue attacking, cancel current attack, place a backdoor and leave or something else. Ask for examples.
   - Does the attacker undertake any actions to hide evidence of his/her presence? e.g. wipe log entries, remove evidence of the attack, etc. Request examples.
   - How does the nature of the attacker's motivation influence his choice of strategy? (E.g. if the motivation is profit or sabotage, perhaps the attacker will prefer to stay undetected and hide all traces; if the motivation is fame, the attacker may not care much about being detected and may leave traces of the attack; in case of rage or revenge the attacker is not likely to behave rationally, etc.) – these aspects may result in a range of different combinations of motivation and behaviour.

8. Which factors influence attack difficulty (*e.g. does difficulty depend solely on defensive measures and your skills as an attacker or are there other important considerations*)? What influences the likelihood of an attack (*e.g does it depend solely on your actions and decisions, or are there other variables that affect the likelihood of an attack*)?

   - Is the attacker's strategy static or adaptive (adaptive meaning that the subsequent steps or basic actions depend on the results of the previous steps)? Request examples.
   - Which environmental factors or events could drastically influence the strategy and force it to change rapidly? Such factors could possibly be the discovery of defensive measures in place, emotional reasons, discovery of a better goal, etc. Request examples.

9. Does your strategy remain static, or does it change under certain conditions? Please provide examples. Which factors could influence your strategy (*e.g. discovery or knowledge of preventive/protective measures*)? Do you sometimes change strategy "on the fly"? If so, please provide examples.

10. Do you care about being detected (*if your attack is detected by the preventive/protective measures deployed on the target enterprise*)? Your typical action upon being detected is .... (*stop attacking or continue attacking*). Do you take steps to avoid being traced (*wipe log entries, remove evidence of the attack, etc.*)?
11. Given the reconnaissance phase results (*e.g. a list of discovered vulnerabilities, etc.*) for the target enterprise and your chosen goal (*e.g. access confidential information stored on an internal network within that enterprise*) - how do you construct a sequence of steps leading to your goal in the target enterprise? Do you use automated tools to support this process, or rely on your own skill and experience? If you have discovered several paths that all lead to your chosen goal, how do you select your " preferred path"?
12. Do you find common patterns in enterprise topology, system configurations, with predictable weak spots, and corresponding common attack patterns? Or is every case unique, requiring its own unique approach? How would you describe the similarities / differences in the attack and defence approaches you have experienced? Please provide examples.

B.2. Comments on the questionnaire

To assist the persons performing the interviews, the questionnaire development team also prepared some assisting comments concerning the questions to be asked.

1. The aim is to gather an overview of the personal properties of the respondent:
   - General level of education, professional qualifications, informal self-education.
   - Country of residence/culture.
   - Respondent's age, gender, status, affiliations.
   - Living arrangements - family/shared accommodation/living alone.
   - Use of substances (e.g.alcohol/drugs) when "working".
   - Respondent's social preferences (use of social networks, sharing achievements with others, such as exploits he has written or sharing discovered security flaws for fame or other reasons? does the respondent prefer to attack alone or in a team of attackers?). Collaboration with others with similar interests (hacker conferences, private forums,etc).
   - How long has the respondent been involved in hacking.
   - Respondent's methods of self-education (hacker conferences, private forums).
   - Set of skills the respondent feels most familiar and proficient with.
   - If the respondent regards himself/herself as a good (capable) hacker?

Answers to this question may contribute to attacker profiling. For instance, the behavioural patterns of a teenager will differ from a professional's footprints. Country of residence may also influence the attacker profile. Cultural / behavioural influences may be deduced from the responses and may provide valuable information for the attacker model, as well as suggesting the respondent's social preferences. The last question on the list is meant to provide the opportunity for the respondent to showcase their skills and keep focused and motivated with subsequent questions. By asking for the set of skills the respondent is most familiar with, we derive the field of expertise which can later be identified within the social, technical or physical domains. This would allow us to rank the degree of confidence which might be placed in the answers provided to the different questions by each respondent.

2. We want to learn which conditions, circumstances, or changes in environment may cause the attacker goals and/or motivation to change and which factors influence this. Ask the respondent to provide examples of instances in which their approach may have changed during an attack. By processing answers to this question we can derive factors under which attacker behaviour may change. This would allow the accuracy of the attacker model to be increased.

3. We want to discover if non-targeted attacks are popular among attackers. If this is the case, then we should think about modeling attacks as unrooted DAG (Directed Acyclic Graph) structures representing attack-defense scenarios with several "root" nodes, instead of rooted attack-defense trees.

4. We want to learn attacker strategy, more specifically, the reconnaissance phase. We need to get answers to the following questions:

   - Is the reconnaissance phase present in every attack or not? When yes? When no? Why not? etc.
   - How are technical and physical vulnerabilities discovered? (We expect the answers to be quite predictable, but nevertheless we may get some interesting answers here.)
   - How are social vulnerabilities discovered? Is this process systematic, or not? Are any tools used for this? Ask for examples, i.e "Can you list some typical vulnerabilities that are easiest to exploit? How can those be discovered?"

   Answers to these questions contribute to the WP2 data discovery process specification.

5. We want to learn how a complex attack is constructed, as a sequence of basic actions.

   - Given the reconnaissance phase results, which factors are compared to decide whether a physical, technical or social approach would be the most efficient?
   - Comparing several weak spots in the same domain (social, technical, or physical) – what are the criteria for comparing them and deciding which one will be the most efficient to exploit?

- Given the reconnaissance phase results for the target enterprise and a given goal – what are the subsequent steps that need to be undertaken to achieve that goal? Ask for examples.
- Is the choice of those subsequent steps based on the respondent's skill and experience, or are there some other factors that are taken into account? Ask for examples.
- Which factors are taken into consideration when comparing several different paths leading to the same goal? How is the most efficient path chosen, or is there no particular preference regarding paths?

Answers to this question will contribute to attack-defense scenario generation and risk prediction in the models. Attacker strategies can be derived from the answers as well. This contributes to the attacker profiles.

6. Please describe some typical cases involving social engineering. Can you suggest any systematic ways to protect a system against social engineering attacks?
7. We want to get information on possible parameters of attacks from the attackers' point of view and possible domains of their quantification.
   - How would the attacker estimate the likelihood of success and difficulty of achieving the chosen set of actions that lead to his/her goal? Value domain, scale, etc. Ask for examples.
   - Do attack difficulty and likelihood of success depend solely on defensive measures and attacker's skills, or are there other important considerations?
   - Are there any other parameters that have been missed out and that could be estimated for an attack?
   - Which factors are considered, when evaluating whether a particular enterprise is a realistic target and if the attack is worthwhile? How are the difficulty of the attack and the likelihood of success estimated in this case? Ask for examples.

   The above questions should be asked separately for technical, physical and social domains, as estimations of the difficulty or likelihood of technical attacks and social attacks are likely to vary. We need to figure this out. Answers to those questions contribute to WP2 attack quantitative metrics, attack attributes, quantitative estimations, risk calculation, etc.
8. We want to identify possible strategy categories, i.e. aggressive, rational, risk-averse, etc.

   Answers to those questions contribute to the WP2 attacker profile and WP3 attacker strategy.
9. We want to know how stable the strategy chosen by the attacker is, how likely it is to change and according to which factors and conditions.

- Is the attacker's strategy static or adaptive (adaptive meaning that the subsequent steps or basic actions depend on the results of the previous steps)? Ask for examples.
- Which environmental factors or events could drastically influence the strategy and force it to change rapidly? Such factors could possibly be the discovery of the presence of defensive measures, emotional reasons, discovery of a better goal, etc. Ask for examples.

Answers to this question contribute to the WP2 attacker profile, and further specification of the attacker strategy for WP3 calculations.

10. We want to have specific examples of the attack phase in practice. i.e. the description of typical cases involving social engineering, systematic ways to protect against social-engineering attacks, ways to prevent those attacks, ways to detect attacks of that kind.

    - Does the attacker care about being detected and to what extent?
    - How would the attacker estimate the likelihood of being detected? Ask for examples.
    - What are the preferred actions upon being detected? I.e. stop attacking, continue attacking, cancel current attack, place a backdoor and leave or something else. Ask for examples.
    - Does the attacker undertake some actions to hide the evidence of his/her presence? I.e wipe log entries, remove evidence of the attack, etc. Ask for examples.
    - How does the attacker's motivation influence his choice of strategy? (E.g. if the motivation is profit or sabotage, will the attacker prefer to stay undetected and hide all traces; if the motivation is fame, the attacker would not care much about being detected and might leave traces of the attack; in case of rage or revenge the attacker may not behave rationally, etc.) – we need to figure out possible combinations of those.

    This is a possible source of interesting ideas.

11. We want to know if the attacker sticks to the chosen strategy and how likely he/she is to deviate from it. Namely, after having achieved the goal of the attack, does the attacker quit or perform some other actions (e.g. destructive malicious actions, place a backdoor or rootkit, look for other goals beyond the scope of the main goal).

    The answer to this question contributes to the attacker strategy and attacker profiles.

12. We want to know if there are some common patterns in infrastructure, configurations, attacks and defenses. Ask the respondent to describe (based on his/her own experience) the similarities and differences that are present in:

    - enterprise physical infrastructure, technical topologies, system configurations,

- corresponding offensive measures have their own patterns,
- attack and defence approaches that the respondent has seen.

Ask for examples.

Answers to this question are possible inputs for patterns of offensive / defensive measures, as well as model and data sharing guidelines.