SEVENTH FRAMEWORK PROGRAMME

RASEN

Compositional Risk
Assessment and Security
Testing of Networked Systems

Deliverable D5.3.2

Methodologies for Legal, Compositional, and Continuous Risk Assessment and Security Testing v.2

Project title:	RASEN
Project number:	316853
Call identifier:	FP7-ICT-2011-8
Objective:	ICT-8-1.4 Trustworthy ICT
Funding scheme:	STREP – Small or medium scale focused research project

Work package:	WP 5
Deliverable number:	D5.3.2
Nature of deliverable:	Report
Dissemination level:	PU
Internal version number:	1.0
Contractual delivery date:	2014-09-30
Actual delivery date:	2014-09-30
Responsible partner:	SINTEF

Contributors

Editor(s)	Fredrik Seehusen (SINTEF)
Contributor(s)	Jürgen Groβmann (Fraunhofer),Tobias Mahler (UiO), Fredrik Seehusen (SINTEF), Bjørnar Solhaug (SINTEF), Ketil Stølen (SINTEF)
Quality assuror(s)	Arthur Molnar (Info World), Fabien Peuraux (Smartesting)

Version history

Version	Date	Description
0.1	23.07.2014	Table of contents with partner roles
0.2	24.07.2014	Initial input from SINTEF
0.3	29.08.2014	Initial input from FOKUS
0.4	08.09.2014	Input SINTEF/FOKUS
0.5	09.09.2014	Input from UiO
0.6	09.09.2014	Document ready for internal review
1.0	29.09.2014	Final quality check

Abstract

This deliverable documents the second version of the methodologies related to tasks T5.1 and T5.2.

Keywords

Methodology; risk management; information security; risk-based security testing; test-based risk assessment; compositional risk assessment; legal risk management;

Executive Summary

This document constitutes the second version of the RASEN methodologies related to task T5.1 and T5.2 in work package 5. The first version can be found in the RASEN deliverable D5.3.1.

The methodologies address three distinct domains: security risk assessment, security testing, and legal compliance. What it new w.r.t. the previous version of the RASEN methodologies, is that the methodologies in the different domains have been unified into an overall picture. In addition to this, the specific RASEN methodologies have been further developed, and examples of their usage are given.

Table of contents

1 Introduction

The overall objectives of WP5 are to (1) develop a methodology that integrates the techniques developed in WP3 and WP4, (2) develop a methodology which takes into account risk assessment in legal contexts, and (3) develop a toolbox that integrates the tools developed in WP3 and WP4.

This document addresses objectives (1) and (2), and it constitutes the second version of the RASEN methodologies. The deliverable addresses methodologies that combine three distinct areas: security risk assessment, security testing, and legal compliance.

In Section 2 of this deliverable, we describe a generic RASEN process which unifies the three domains addressed (risk assessment, testing, compliance). We then describe how this unified process may in general be instantiated to support specific combinations of the three domains. In particular, we consider combining compliance and security risk assessment (Section 2.1) and combining security testing and security risk assessment (Section2.2). The latter combination is addressed in two directions: first we consider how risk assessment can be used to improve the testing process (referred to as risk-based testing), then we consider how testing can be used to improve the risk assessment process (referred to as test-based risk assessment).

In Section3, we describe specific RASEN methodologies that address specific combinations of the three addressed domains as described in Section2. These specific methodologies may thus be considered instantiations of the general overall RASEN process. For each specific methodology, we describe each of its process steps in detail, and provide examples of usage.

In Section 4, we provide a summary of this document.

2 Overview of the RASEN Methodology

This deliverable describes the RASEN process that combines security risk assessment, compliance assessment and security testing. The process interweaves the identification, estimation, and evaluation of security risks with a set of tests or checks, which either verify conformance or compliance to technical security specifications (i.e. security testing) or compliance with regulatory rules and regulations (compliance assessment). Integrating and interweaving the activities from both sides, thus a systematic integration and completion of risk assessment results with compliance assessment and testing results allow for a more precise, focused and dynamic assessment of systems, processes and other targets. We generally distinguish the following two directions for integration that are depicted in Figure 1.

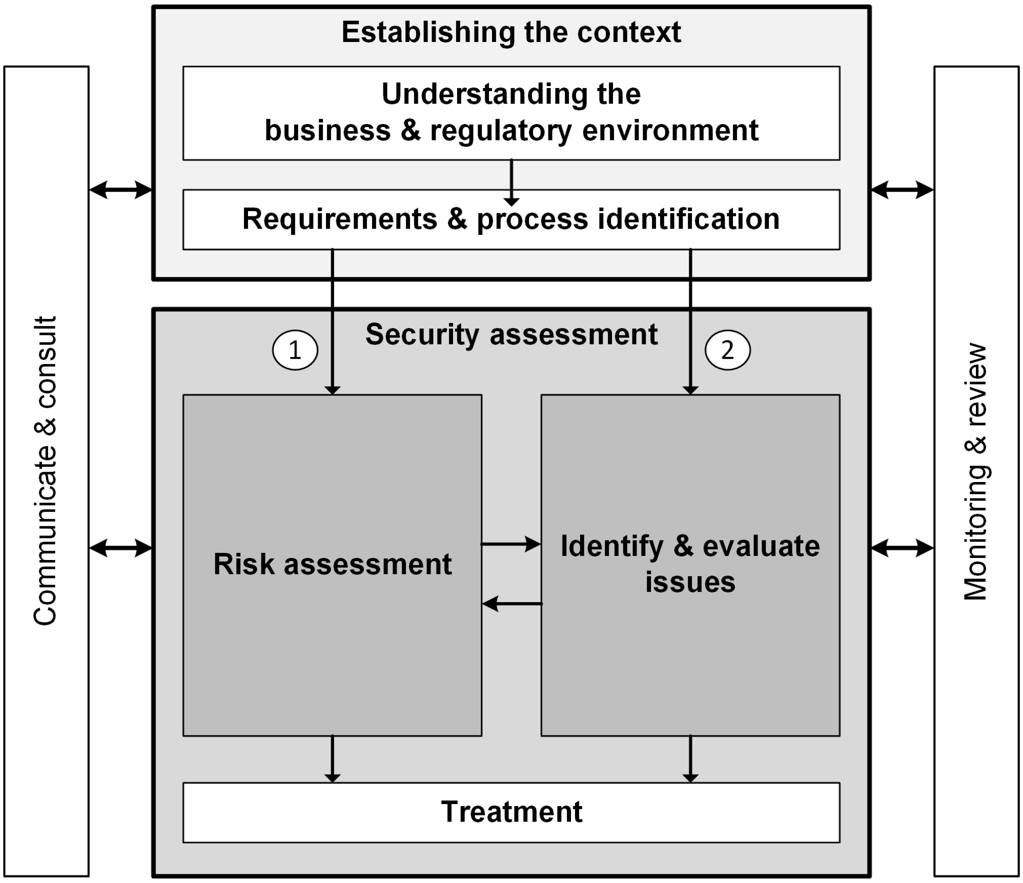
- A test-based or compliance based security risk assessment process (1) will start with the risk assessment and is used to optimize security risk assessment with empirical data coming from test results or compliance issues.
- A risk-based method to compliance or security testing (2), on the other hand side, will start with the identification of issues by security testing or compliance assessment and focus the compliance and security testing resources on the areas that are most likely to cause concern. Such a process involves identifying the areas of risk within the target's business processes or systems and building and prioritizing the compliance measures or testing program around these risks.
- 
Establishing the context
Understanding the business & regulatory environment
Requirements & process identification
Security assessment
1
2
Risk assessment
Identify & evaluate issues
Treatment
Communicate & consult
Monitoring & review

Figure 1 – Overall risk, compliance and quality assessment process

The overall RASEN process of security risk and compliance assessment is derived from ISO 31000 [11] and slightly extended to highlight the identification and evaluation of compliance or security issues as one of the major tasks that need to be carefully aligned with typical risk assessment activities. It is defined independently from any application domain and independently from the level, target or depth of the assessment. It could be applied for legal risk and compliance assessment as well as for any kind of technical assessment. It is integrated with a set preparation and support activities. It starts with establishing the context. This process splits up in two steps:

- "Understanding the business and regulatory environment" is meant to analyze the context of the target under assessment from a business or regulatory perspective.
- "Requirements & process identification" is meant to analyze the technical context of the target under assessment.

Additionally, support activities like "Communication & consult" and "Monitoring and review" are meant to set up the management perspective, thus to continuously control, react, and improve all relevant information and results of the process.

The actual "Risk and compliance assessment" process consists of typical risk assessment activities i.e. "Identify risks", "Estimate risks" and "Evaluate risks" and corresponding activities to "Identify and evaluate issues". Depending on the target of evaluation and the overall goal and perspective of the assessment the checks that are carried out in "Identify and evaluate issues" are either typical testing activities or compliance assessment activities.

In the following sections we describe specific integration of risk assessment and compliance assessment activities as the combination of risk assessment and security testing activities.

All the processes are documented in a similar manner. That is, each step of the methods are documented using the template shown in Table 1.

Name	**The name of the activity**
Actors	The actors that are referred to in the activity
Tools	The tools that are involved in the activity
Precondition	The precondition that needs to be enabled when the activity is initiated.
Postcondition	The postcondition that describes the result of the activity.
Scenario	The scenario that describes the individual actions taken by the actors
Data exchanged/ processed	The data that are exchanged during the integration use case **In:** *The data that go into the activity. Terms from the conceptual model are used to describe the data.* **Out:** *The data that are the outcome of the activity. Terms from the conceptual model are used to describe the data.*

Table 1 – Template for documenting process activities

The possible actors and tools that can be referred to are described below.

Actors:

- **Customer (C):** The person/organization on whose behalf a security assessment is conducted.
- **Risk analyst (RA):** The person responsible for doing the security risk assessment.
- **Security test manager (TM):** The person responsible for doing the security test management
- **Security tester (ST):** The person responsible for doing the security testing.
- **Compliance manager (CM):** The person responsible for ensuring compliance.
- **Auditor (A):** The person responsible for auditing a system.

Tools:

- **Security risk assessment tool (SRAT):** The tool that supports the security risk assessment.
- **Security test management tool (STMT):** The tool that supports the security test specification.
- **Security test specification Tool (STST):** The tool that supports the security test specification.
- **Security test derivation tool (STDT):** The tool that supports the derivation of test procedures and test cases from the SRAT tool to the STT tool.
- **Security Test Execution Tool (STET):** The tool that supports the derivation of test procedures and test cases from the SRAT tool to the STT tool.
- **Security test aggregation tool (STAT):** The tool that supports the aggregation of test results from the STT tool to the SRAT tool.

2.1 Combining Compliance and Security Risk Assessment

The prominence of information technology in day-to-day life means that businesses' ICT infrastructures attract great interest from both cyber-criminals and legislators. Businesses have to deal not only with the increased cyber-attacks, but also with an array of increasingly complex laws dealing with information security. Cyber-attacks clearly represent risks that businesses and organizations need to assess. The need to deal with these risks is based not only on the self-interest of the involved stakeholders, but is also reflected in legal and regulatory requirements. At the same time, some decisions based on legal requirements may also represent legal risks, for example in the form of possible sanctions. According to a Harvard Business Review survey, security and privacy have become significant areas of concern over the past few years. The research underlines that failure to deal with information security risks is not only costly in terms of finances and damage to the company and brand, but the regulatory penalties can also be quite large [8]. This signifies the need for businesses to account for legal issues when addressing their information security risks and to ensure that their day-to-day business operations do not violate legal norms of relevance to information security, such as data privacy laws. For this reason, technical and legal risks often need to be understood in combination. The RASEN project proposes an approach to integrate compliance and security risk assessment.

Moreover, several recent EU policy initiatives require risk management and have an impact on how risk assessment should be carried out. The proposed Network Information Services (NIS) Directive [4] provides key requirements, and the proposed revisions to the Payment Services Directive (PSD2) [5] are of particular relevance. In addition, the proposal for a General Data Protection Regulation (GDPR) explicitly requires a controller, or where applicable the processor, to carry out a risk analysis on the potential impact of the intended data processing on the rights and freedoms of the data subjects, assessing whether the processing operations are likely to present specific risks. Similarly, in the opinion of the Article 29 Working Party,[1] cloud users should perform comprehensive, thorough risk analysis. They need to pay special attention to the legal risks regarding data protection, primarily security obligations and international transfers, before opting to go to the cloud [1]. These rules underline the paramount importance of conducting a risk analysis both from legal and security perspectives. However, the European Data Protection Supervisor has criticized the lack of specific guidelines for how to conduct such legal risk analysis and has recommended that the European Commission develop templates for evaluating and managing risks in cloud computing [6].

The RASEN project contributes to such need by putting forth a systematic and risk-driven approach to risk and compliance assessments. By systematic we mean that relevant risks and control measures are mapped, to the extent possible, to relevant compliance requirements. By risk-driven we mean compliance requirements are prioritized based on their risk levels. The RASEN method enables its users to prioritize compliance requirements based on their level of risks and to take account of legal consequences in making decision about security risks. The RASEN project also provides a technique to help structure and simplify the identification of legal and compliance risks from compliance requirements and the business environment.

[1] This is a working group composed of national data protection authorities.

In the context addressed by RASEN in particular, the legal risk and compliance assessments will be integrated to the overall risk management framework, and will be carried out in conjunction with a security risk assessment. The main objective of the integration is to enable the following:

- The security risk assessment takes account of the legal and compliance issues where the legal risk analysis might help to prioritize the treatment of security risks.
- The legal and compliance assessment benefits from the security risk assessment. For example, the security risks can be used as an input for legal risk assessment and support a systematic approach to legal compliance.
- The security risk assessment provides information relevant for compliance with breach notification requirements.

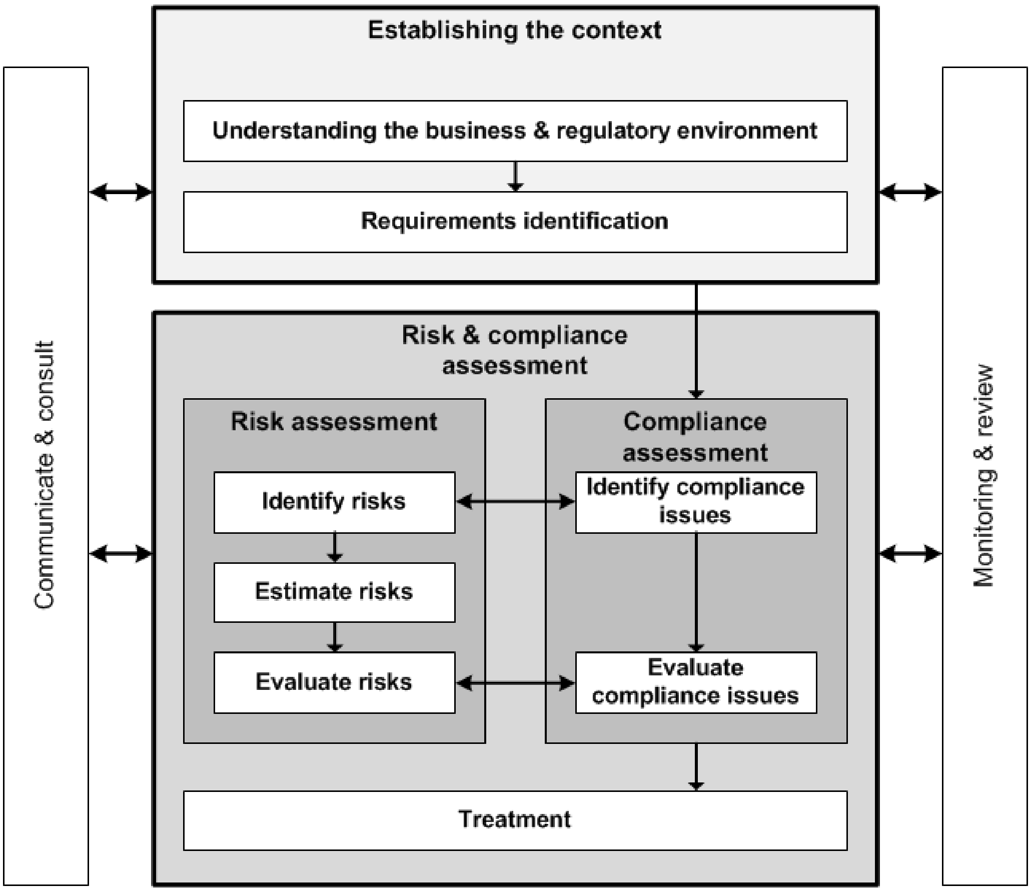

Establishing the context
Understanding the business & regulatory environment
Requirements identification
Communicate & consult
Monitoring & review
Risk & compliance assessment
Risk assessment
Identify risks
Estimate risks
Evaluate risks
Compliance assessment
Identify compliance issues
Evaluate compliance issues
Treatment

Figure 2 – Integrated risk and compliance assessment

Figure 2 shows the overall risk and compliance assessment process. It consists of the risk assessment process as specified in ISO 31000 [11] and a generic compliance assessment process derived from the Australian Standard for Compliance Programs (AS 3806-2006) [1]. Figure 3 and the following paragraphs describe the main interactions between compliance and risk assessment.

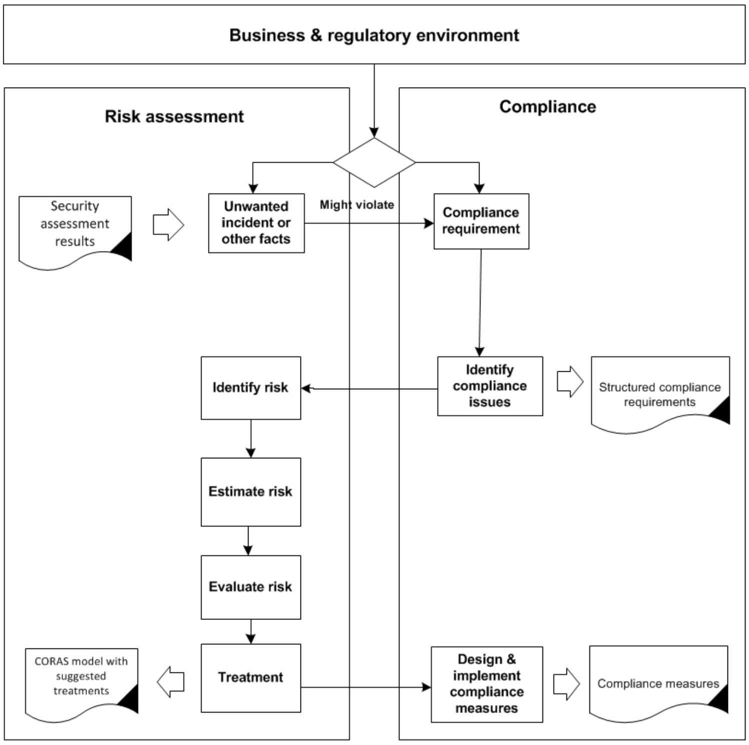

Business & regulatory environment
Risk assessment
Compliance
Security assessment results
Unwanted incident or other facts
Might violate
Compliance requirement
Identify risk
Identify compliance issues
Structured compliance requirements
Estimate risk
Evaluate risk
CORAS model with suggested treatments
Treatment
Design & implement compliance measures
Compliance measures

Figure 3 – Overall interaction of compliance and risk assessment

1. **Compliance risk identification:** Once the context is established, risks can be identified. The main goal of the compliance risk identification is to deal with compliance requirements that imply risk. This should also include identifying legal consequences of security risks. The RASEN approach provides a structured method for identifying risks from compliance requirements or from the business environment.
2. **Compliance risk estimation:** Risk with a large potential loss and a low probability of occurrence is often treated differently from one with a low potential loss and a high likelihood of occurrence. However, in order to estimate the risk, one needs to understand the underlying uncertainty. That uncertainty can originate from a number of sources, including from the compliance requirements themselves. For example, compliance requirements may be unclear, or there may be uncertainty about the consequences of non-compliance.
3. **Compliance risk evaluation:** The risk evaluation step is used to prioritize compliance requirements based on their level of risk and to prioritize security risks based on their legal consequences. Prioritization may be relevant, for example, due to resource limitations.
4. **Treatment:** The goal of this step is to allocate compliance resources efficiently based on their risk level as well as any relevant ethical issues. Once implemented, the measures are intended to contribute to achieving compliance with legal norms, including those relevant to security. In order to avoid unethical business conduct, the risk-based compliance measures should also take consideration of ethical issues. Checking compliance (auditing) also benefits from the risk-driven approach where only high risks areas are audited or checked. In addition, decisions regarding security risks would take account of the legal consequences of security risks.

More precisely, the RASEN methodology will offer organizations the following important capabilities. First, the integration between risk assessment and compliance in general opens for a potential integration where compliance (legal) requirements will be accounted for in the general risk analysis including security risk analysis. This is particularly relevant in RASEN because the identification, assessment, and treatment of legal risks related to information security relies on an understanding of the security risks and measures. Similarly, legal norms of relevance to information security often prescribe security requirements that security risk analysts ought to heed. However, lawyers often lack the technical expertise needed to assess technical risks, and technical experts may lack detailed information about the legal security requirements and the legal consequences of technical problems [12]. Therefore, integrating the compliance aspect into the ISO 31000 risk management process will be essential in achieving such objective. Furthermore, such an approach may also contribute to the identification of interdisciplinary solutions to security risks and legal risks. In other words, security risk analysis could benefit from the legal perspective in the sense that legal treatments could be applied in treating security risks such as through a contract (limiting liability), insurance, and persecuting offenders that interfere with the security system. Similarly, it may be possible to reduce the likelihood of non-compliance through non-legal remedies, such as an improved IT system [16].

Such integration also opens the possibility where the security risk analysis could support the legal risk analysis and vice versa. This is particularly important in complying with breach notification requirements. Across EU, there are mandatory breach notification requirements in some sectors such as the telecom business. In addition, many member states have extended such obligations to other sectors domestically. Furthermore, currently in the US, 46 States have notification requirements for breaches of personal information. And more importantly, the draft General Data Protection Regulation [3], which will be uniformly applicable to all member states, has a mandatory provision obligating the notification of data breaches. Similarly, the new proposed Directive on Network and Information Systems under its Article 14 (2) requires member states to ensure that "… market operators notify to the competent authority incidents having *a significant impact* on the security of the core services they provide." Articles 31 and 32 of the draft Data Protection Regulation [3] also require a notification of any data breach to the authorities. Such breach should be notified both to the authorities and data subjects when the *data breach is likely to adversely affect the protection of the personal data, or the privacy, the rights or the legitimate interests of the data subject***.** Determining whether a breach has is 'likely to adversely affect the protection of personal data or privacy' would require taking consideration of the details of the security breach at hand. The assessment of whether a certain security incident has 'a significant impact on the security of core services' under the NIS Directive would also require security risk analysis.

A survey by ENISA [7] shows that a risk-based approach to information breach notifications as essential means to balance the interest of breach notification fatigue for data controllers and the interest survey by the breach. Therefore, an integrated approach for dealing with security and legal matters in conjunction will enable for assessing which of the identified security incidents, if materialized, would need notification to the authorities or both to the authorities and data subjects. In this regard, the security risk analysis is essential in providing essential inputs such as the nature of the data that has been breached (financial, health, etc.), nature of the breach (widespread, or an isolated incident; technical, human error, or theft), and security level (has the data been encrypted). The security risk analysis will also provide information regarding whether the incident has 'a significant impact on the security of core services' so that the breach notification requirement under the NIS Directive need to be complied. Furthermore, the security risk analysis becomes essential when we look at the content of the notification that the regulations require. For example, the General Data Protection Regulation, under its Article 31, states that the content of the notification should at least include the nature of the personal data breach, the categories and number of data subjects concerned and the categories and number of data records concerned. Attaching the data breach notification requirement to security risk analysis would enable organizations to import such content easily from the latter.

Another motivation for bringing the security and legal risk together pertains to the criteria for measuring the consequence of a security breach in case of information security. Often the criterion for measuring the consequences of information security breach is through the number of records affected by the breach. However, from a legal stand point, although the number of records affected are also important, other factors could become even more relevant such as the type of data affected (ordinary personal data, sensitive personal data and child data) and how the breach might affect the data subjects' rights. The latter implies that where the data ends once the breach occurs and the consequent danger it poses might also need to be considered for legal purposes. Therefore, from a risk management perspective, it is important that organizations are able to understand, from their legal standing, what it would entail if a certain information security risk were to materialize. One way of doing this is to perform an assessment of what the information security risks mean from the legal perspective of the organization after such risks are identified through a security risk analysis.

Considering both the security and legal risk together would help organizations determine where to focus their resources. In turn, taking consideration of the legal implications, organizations might be able to prioritize some security risks over others. For further details and a more elaborate presentation of how to assess legal implications of security risks, the reader is referred to [17]. On a similar vein, the law might, directly or indirectly, prescribe certain criteria below which some security incidents might not be acceptable depending on different factors such as sensitivity of the data. For example, Section 2-2 of the Norwegian Personal Data Regulation [15] stipulates that the 'Data Protection Authority may issue orders regarding the protection of personal data, including the establishment of *criteria for acceptable risk* associated with the processing of personal data.' The alignment of the legal risk analysis with security risk analysis will enable to account such legal requirements in the security risk analysis. This would ensure that a certain risk which is acceptable according to the criteria used by the organization is not prohibited by law or is also acceptable from the legal standing of that organization.

2.2 Combining Security Risk Assessment and Security Testing

Almost all the approaches that combine testing and risk assessment fit best into the category of risk-based testing, i.e. risk assessment is primarily used to optimize the testing by means of one of the following activities:

- Risk-based test or feature prioritization: This activity supports testing by using risk assessment artifacts to prioritize artifacts during test design, implementation and/or execution.
- Risk-based test or test technique identification: This activity supports testing by using risk assessment artifacts (typically through fault/threat modeling) to identify test purposes, test techniques and test condition.
- Risk based test scenario generation: This activity supports testing by using risk assessment artifacts (together with a test model) to manually derive or automatically generate test scenarios or test cases.

In addition testing or other measurement approaches can additionally be used to affect or optimize the results of the risk assessment. Risk assessments, similar to other development activities that start in the early phases of a development project, are mainly based on a set of assumptions that have been made on the system to be developed. Testing is one of the most relevant means to do real experiments on real systems and thus helps to gain empirical arguments on the existence or absence of vulnerabilities, the applicability and consequences of threat scenarios and the quality of countermeasures. Considering this, test-based risk assessment uses test results to gain arguments or evidence for the assumptions that have been made during the initial risk assessment phases.

- Test-based risk assessment: This activity supports risk assessment by using security testing results to evaluate risk factors, e.g., the existence of vulnerabilities, their probabilities and the quality and effectiveness of counter measures.

2.2.1 Risk-based Security Testing

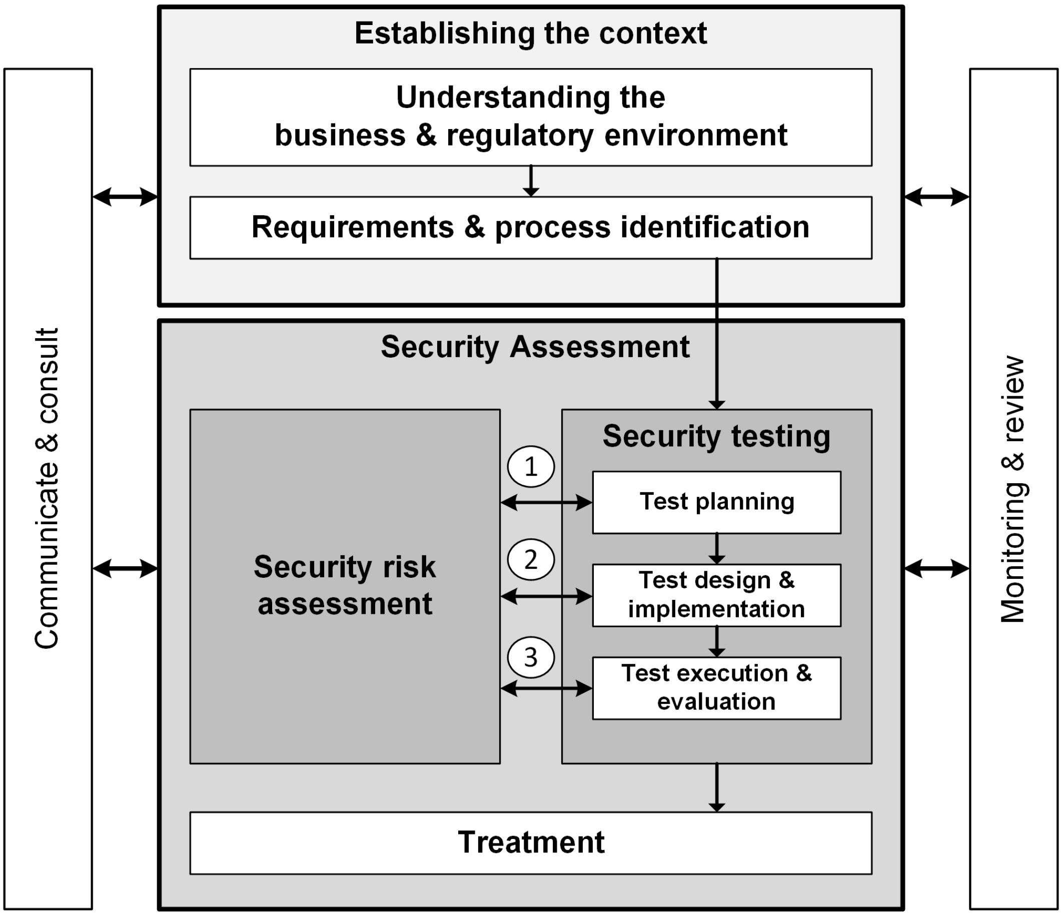


Figure 4 – Process model for risk-based security testing

In Figure 4, we have illustrated three phases of a testing process that are affected and supported by risk-based security testing. In the following, we describe these in more detail.

1. Risk-based security test planning deals with the integration of security risk assessment in the test planning process. For that, security risk assessment is used to roughly identify high-risk areas or features of the system under test (SUT) and thus determine and optimize the respective test effort that is needed to verify the related security functionality or to address the related vulnerabilities. Moreover, a first assessment of the identified vulnerabilities and threat scenarios my help to select test strategies and techniques that are dedicated to deal with the most critical security risks.

2. Risk-based security test design and implementation deals with the integration of security risk assessment in the test design and implementation process. During the test design and implementation phase, test cases are derived, implemented and assembled to test procedures. Security-risk assessment in general provides two different kinds of information that are useful within this process. On the one hand side it provides detailed information on expected threats and potential vulnerabilities. This information can be used to systematically determine and identify test conditions (testable aspects of a system), test purposes or high-level test scenarios that are dedicated to address the identified threats and vulnerabilities. On the other hand side the security risk assessment provides quantitative estimations on the risk, i.e. the product of frequencies or probabilities and estimated consequences. This information can be used to select and prioritize either the test conditions or the actual tests when they are assembled to test sets. Risk-based security test selection criteria can be used to control the selection or the selected generation of test cases. The criteria are designed by taking the risks as well as their probabilities and consequence values to set priorities for the test selections, test case generation as well as for the order of test execution.

3. Risk-based security evaluation and control: The decision on how extensive testing should be is always a question of the remaining test budget, the remaining time and the probability to discover even more critical errors, vulnerabilities or design flaws. Risk-based security test monitoring and control aims for improving the monitoring and control activities by introducing the notion of risk coverage and remaining risks on basis of the intermediate test results as well as on basis of the errors, vulnerabilities or flaws that have been found so far.

While security test planning as well as security test monitoring and control belong to the test management process, security test design and implementation belong to the dynamic test process that is controlled by the test management process.

2.2.1.1 Risk-based Security Test Planning

Test planning is the activity of developing the test plan. According to ISO 29119 [10], it determines the test objective, the test scope, and the risks associated to the overall testing process. The main outcome of these activities is the test strategy to be used and a plan that depicts the staffing, the required resources and a schedule for the individual testing activities. Figure 5 shows the integration of security risk assessment results in the overall test planning process. We have identified three integration activities that all serve different purposes:

a. Integrate risk analysis

b. Risk-based test strategy design

c. Risk-based security resource planning and test scheduling

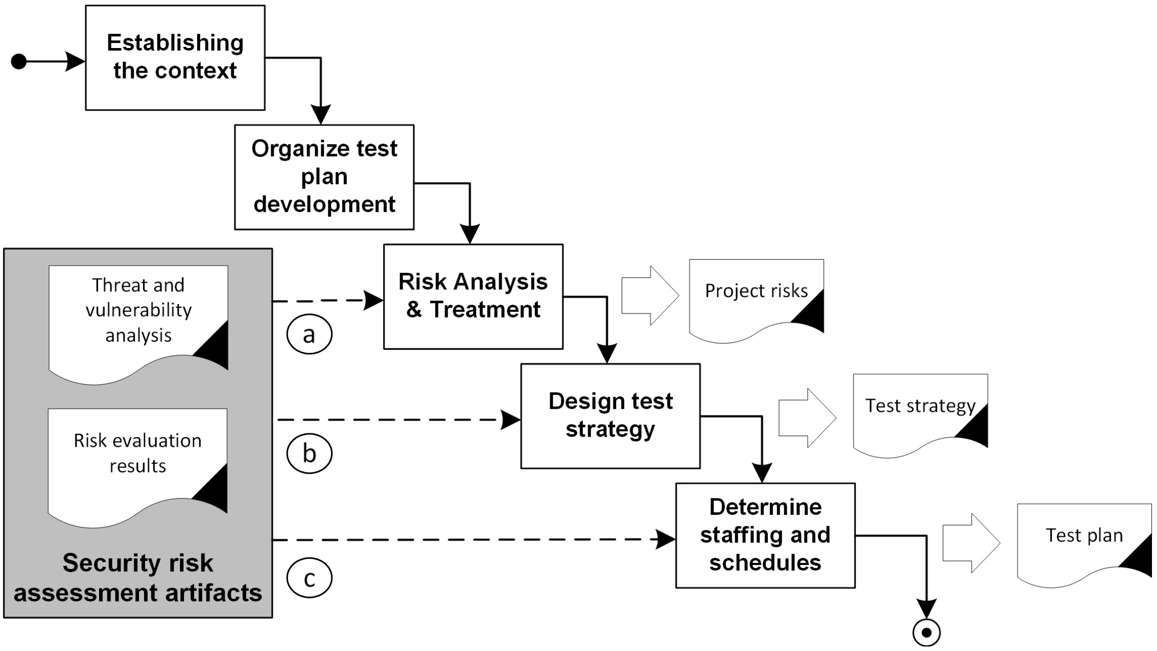

Establishing the context
Organize test plan development
Risk Analysis & Treatment
Project risks
Design test strategy
Test strategy
Determine staffing and schedules
Test plan
Threat and vulnerability analysis
Risk evaluation results
Security risk assessment artifacts
a
b
c

Figure 5 – Process model for risk-based security test planning

Typically, risk analysis is a substantial part of the test planning process. The risk analysis is done to get an estimate on the specific project risks, considering the availability of test resources, considering specific product risks and other project related issues.

Name	**Integrate risk analysis (a)**
Actors	Security Test Manager (TM), Security risk analyst (SRA)
Tools	Risk Assessment Tool (SRAT), Security Test Management Tool (STMT)

Precondition	Contextual information like legal or regulatory requirements, organizational test and security policies, organizational or higher-level test strategies, and technical limitations as well as resource limitations are known. Security risk assessment results (threat, vulnerability and risk estimations) that capture the technical, business, regulatory and legal requirements are available.
Postcondition	A project risk assessment that provides an overall risk picture for the test project, considering project risk that reflect risks that come from the security risk analysis.
Scenario	1. The Test Manager should review the relevant security risks to identify those, which have a special role for security testing. 2. The Test Manager should try to identify additional risks like other product risks or project related risks like missing resources, technical issues related to the test infrastructure etc. 3. The Test Manager should develop an overall risk picture for the test project and communicate the risk picture to the Stakeholders.
Artifacts exchanged/ processed	**In:** *Vulnerabilities, threat scenarios, unwanted incident, likelihoods, consequences, risk level* **Out:** *Project risks*

Table 2 – Risk-based security test planning: Integrate risk analysis (a)

One of the major activities during test planning is the design of a test strategy. A test strategy defines the test phases, the types of testing, the test techniques and the test completion criteria. For security testing especially the identification of test techniques is a challenge that should be optimized by directly considering the potential threats and vulnerabilities, which have been identified during a security risk assessment.

Name	**Risk-based security test strategy design (b)**
Actors	Security Test Manager (TM), Security Risk Analyst (SRA)
Tools	Risk Assessment Tool (SRAT), Security Test Management Tool (STMT)
Precondition	Contextual information like legal or regulatory requirements, organizational test and security policies, organizational or higher-level test strategies, and technical limitations as well as resource limitations are known. Security risk assessment results (threat, vulnerability and risk estimations) that capture the technical, business, regulatory and legal requirements are available. Security risks that are relevant for testing have been identified, see integrated risk analysis (a)
Postcondition	A test strategy comprising test phases, test types, features to be tested, test techniques and test completion criteria that directly address the identified threats and vulnerabilities.

Scenario	1. The Test Manager should assign vulnerabilities and threat scenarios to test items (interfaces, operations, components) and/or test conditions. 2. The Test Manager should try to identify the potential vulnerabilities that have the highest impact on the overall security risks when they are detected. 3. The Test Manager should assign test techniques that are capable to detect the identified vulnerabilities to each test item and/or tor each test condition. 4. The Test Manager should assign test completion criteria to each test item and/or each test condition. 5. The Test Manager should prioritize test item and/or for each test condition by considering the required test efforts to match the completion criteria and the impact testing may have on the overall security risks (i.e. when vulnerabilities are detected or test suites pass without detecting anything)
Artifacts exchanged/ processed	**In:** *Vulnerabilities*, *threat scenarios*, *unwanted incident*, *likelihoods, consequences, risk level* **Out:** List of applicable *test techniques, test completion criteria, prioritized list of test items and/or test conditions*

Table 3 – Risk-based security test planning: Risk-based security test strategy design (b)

The second major activity during test planning is the planning of resources and the schedule for the testing activities. Since the main task of security testing is finding vulnerabilities, resource planning and test schedules should be aligned with the major security risks so that resources and the order of testing allows for a focused testing of the test items or test condition where the detection of vulnerabilities shows the largest impact.

Name	**Risk-based security resource planning and test scheduling (c)**
Actors	Security Test Manager (TM)
Tools	Risk Assessment Tool (SRAT), Security Test Management Tool (STMT)
Precondition	Contextual information like legal or regulatory requirements, organizational test and security policies, organizational or higher-level test strategy, technical and resource limitation are known. Security risk assessment results (threat, vulnerability and risk estimations) are available that capture the technical, business, regulatory and legal requirements. Test strategy depicting the test items, test conditions, test techniques etc.
Postcondition	A test plan that depicts resources, staffing and test schedules respecting certain threats and vulnerabilities and their associated risk scores.
Scenario	1. The Test Manager should allocate resources considering the required test efforts for that test items or test conditions where testing may have the largest impact in terms of treating or minimizing the identified security risks. 2. The Test Manager should plan the test schedules so that test items or test conditions where testing might have the largest impact in terms of treating or minimizing the identified security risks are tested first.

	In: *Vulnerabilities, threat scenarios, unwanted incident, likelihoods, consequences, risk level* **Out:** *Resource allocation and test schedules that respect the identified security risks.*

Table 4 – Risk-based security test planning: Risk-based security resource planning and test scheduling (c)

In summary, the integration of security testing and security risk assessment is addressed during the test planning phase by three activities, that each contribute with the notion of security risks, threat scenarios and vulnerabilities to the testing activities.

2.2.1.2 Risk-based Security Test Design and Implementation

The test design and implementation process is mainly dedicated to derive the test cases and test procedures that are later on applied to the system under test. To achieve this in a systematic way the overall process should start with a concise definition of the features and test conditions that are the main subjects to test. On basis of that, the relevant test coverage items should be identified, the test cases should be derived and they finally should be assembled to adequate test sets and test procedures. Considering especially security testing, security risks, potential threat scenarios and potential vulnerabilities provide a good guidance which of the features and test conditions require testing, which coverage items should be covered in which depth and how individual test cases and test procedures should look like. We have identified three integration activities for risk-based security test design and implementation:

a. Risk-based identification and prioritization of features sets

b. Risk-based derivation of test conditions and test coverage items

c. Risk based derivation of test cases

d. Risk-based assembly of test procedures

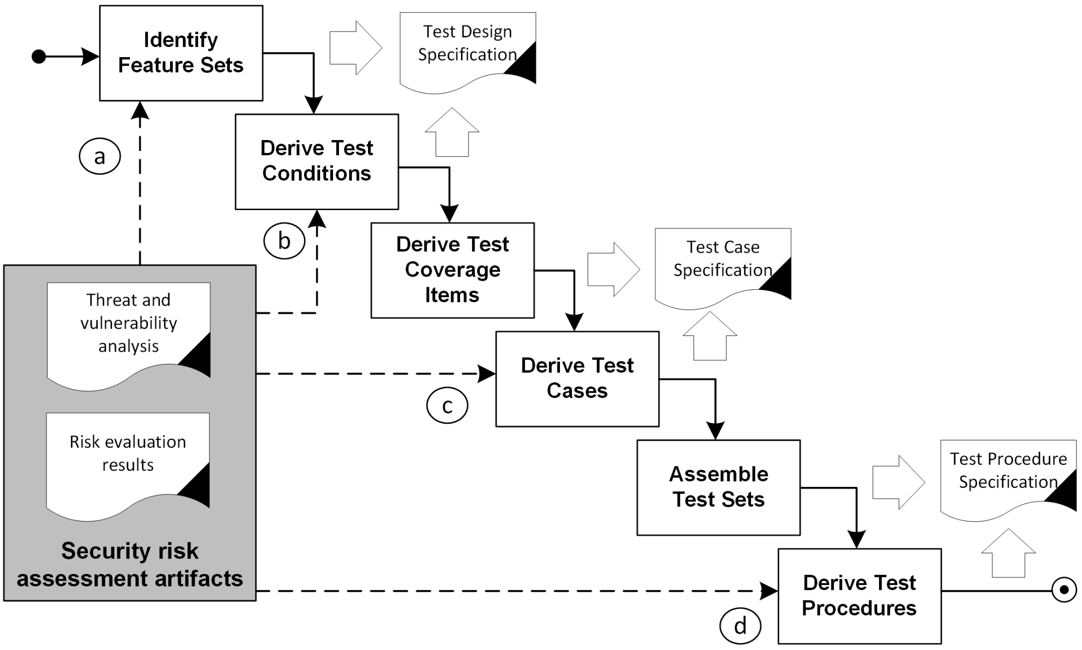


Figure 6 – Process model for risk-based security test design

A first step during the test design phase is the identification and categorization of the security features that will be tested. Since security features describe functional security measures this approach especially allows for testing the correctness of the feature implementation. Security risk assessment can be used to determine the most critical security features so that these features are tested more intensively and in more detail.

Name	**Risk-based identification and prioritization of features sets (a)**
Actors	Security Tester (ST), Security Risk Analyst (SRA)
Tools	Test Specification Tool (STST), Security Risk Assessment Tool (SRAT)
Precondition	Security relevant features are documented and the security risk assessment is available
Postcondition	Security relevant features to be tested are grouped with respect to potential vulnerabilities and threat scenarios.
Scenario	1. The Security Tester should identify testable security relevant features that need to be covered by security testing. The security tester classifies the security relevant features by grouping them to form feature sets that each addresses exactly one threat scenario and/or one vulnerability. 2. The Security Tester should prioritize the security relevant feature sets using the risk levels that are associated with the threat scenario and/or vulnerabilities. 3. The Security Tester should document the relations between security relevant feature sets and their associated threat scenarios and/or vulnerabilities (maintain traceability).
Data exchanged/ processed	**In:** *Vulnerabilities*, *threat scenarios*, *unwanted incident*, *likelihoods, consequences, risk level* **Out:** *Prioritized list of testable security relevant features (security feature sets).*

Table 5 – Risk-based security test design: Risk-based identification and prioritization of features sets (a)

After a set of testable security relevant features have been identified the security tester should derive the test conditions and test coverage items. This could be done on basis of the identified features (see Risk-based identification and prioritization of features sets (a)) but needs to consider that especially security is a non-functional property and that a correct implementation of all security features may not ensure a secure system. Thus, additional test conditions and coverage items need to be derived that especially address the detection of currently unknown vulnerabilities (vulnerability and robustness testing). Security risk assessment should be used to provide guidance for the derivation of test conditions and test coverage items for vulnerability and robustness testing.

Name	**Risk-based derivation of test conditions and test coverage items (b)**
Actors	Security Tester (ST)
Tools	Security Tester (ST), Security Risk Analyst (SRA)
Precondition	Test Specification Tool (STST), Security Risk Assessment Tool (SRAT)
Postcondition	Test conditions and test coverage items weighted according to the impact testing may have on the overall associated security risks

Scenario	1. The security tester should identify test conditions on basis of the security features, threat scenarios and/or vulnerabilities that have been identified during security risk assessment and/or during a **risk-based identification and prioritization of features sets (a**). Please note, testing security features is one approach to security testing that is often not sufficient to cover all major threat scenarios and vulnerabilities. Thus a Security Tester should check whether all relevant threat scenarios already have been covered by **risk-based identification and prioritization of features sets (a)** or if there are remaining risks from potential threat scenarios and vulnerabilities exist that need to be covered by adequate test conditions. 2. The Test Designer should identify test coverage items corresponding to the test conditions identified in 1). Test coverage items and the respective test depth should be chosen according to the impact testing may have on the overall associated security risks.
Data exchanged/ processed	**In:** *Security feature sets, vulnerabilities*, *threat scenarios*, *unwanted incident*, *likelihoods, consequences, risk level, testable sets of security features* **Out:** *Test conditions and test coverage items weighted according to the impact testing may have on the overall associated security risks.*

Table 6 – Risk-based security test design: Risk-based derivation of test conditions and test coverage items (b)

In the next step, the security tester should derive test cases on basis of test conditions and test coverage items. The security tester determines the pre-conditions for the individual test, he selects adequate input values, the actions to exercise the selected test coverage items, and determines the expected results. Since security risk assessment has been used to identify the test conditions and the test coverage items it is already considered through the activities before. However, threat scenarios and potential vulnerabilities that have been identified during risk assessment might still help by identifying the preconditions, input values, actions and expected results.

Name	**Risk based derivation of test cases (c)**
Actors	Security Tester (ST)
Tools	Test Specification Tool (STST), Security Risk Assessment Tool (SRAT)
Precondition	Testable security features, test conditions and test coverage items are known
Postcondition	Security test cases that address threat scenarios and potential vulnerabilities
Scenario	1. The ST should identify the preconditions for the tests, the test data, the test actions and the expected results by examining the test conditions, test coverage items, threat scenarios and potential vulnerabilities. 2. The ST should document the relations between test cases, security feature sets and threat scenarios and/or vulnerabilities (maintain traceability). 3. The Security Tester and a Security Risk Analyst should review the test case specification and their coverage of threats and potential vulnerabilities identified by the security risk assessment.
Data exchanged/ processed	**In:** *Test conditions, test coverage items, vulnerabilities*, *threat scenarios*, *unwanted incident*, *likelihoods, consequences, risk level, testable sets of security features* **Out:** *Security test cases.*

Table 7 – Risk-based security test design: Risk based derivation of test cases (c)

Finally, the test cases should be assembled to test sets and test procedures. While test sets group test cases with common constraints on test environment or test items, test procedures define the order of test execution and thus have to respect the pre- and postconditions. Security risk assessment should be used to prioritize the order of test cases and thus the order of testing with respect to the associated risks.

Name	**Risk-based assembly of test procedures (d)**
Actors	Security Tester (ST)
Tools	Test Specification Tool (STST), Security Risk Assessment Tool (SRAT)
Precondition	Test cases are available and associated with treat scenarios and potential vulnerabilities
Postcondition	Test procedures that prioritize the execution of the most relevant test cases.
Scenario	1. The Security Tester should assemble test sets and test procedures so that the most relevant tests are executed first.
Data exchanged/ processed	**In:** *Test cases, vulnerabilities*, *threat scenarios*, *unwanted incident*, *likelihoods, consequences, risk level, testable sets of security features* **Out:** *Security test procedures.*

Table 8 – Risk-based security test design: Risk-based assembly of test procedures (d)

2.2.1.3 Risk-based Test Analysis and Summary

The decision on how extensive testing should be is always a question of the remaining test budget, the remaining time and the probability to discover even more critical errors, vulnerabilities or design flaws. Risk-based security test analysis and summary aims for improving the evaluation of the test progress by introducing the notion of risk coverage and remaining risks on basis of the intermediate test results as well as on basis of the errors, vulnerabilities or flaws that have been found so far. This process is meant to support the test management process with risk related information that can be used to depict the test results in terms of their relation to the overall security risks. We have identified two integration activities namely:

a. Risk-based test log analysis

b. Risk-based test summary creation

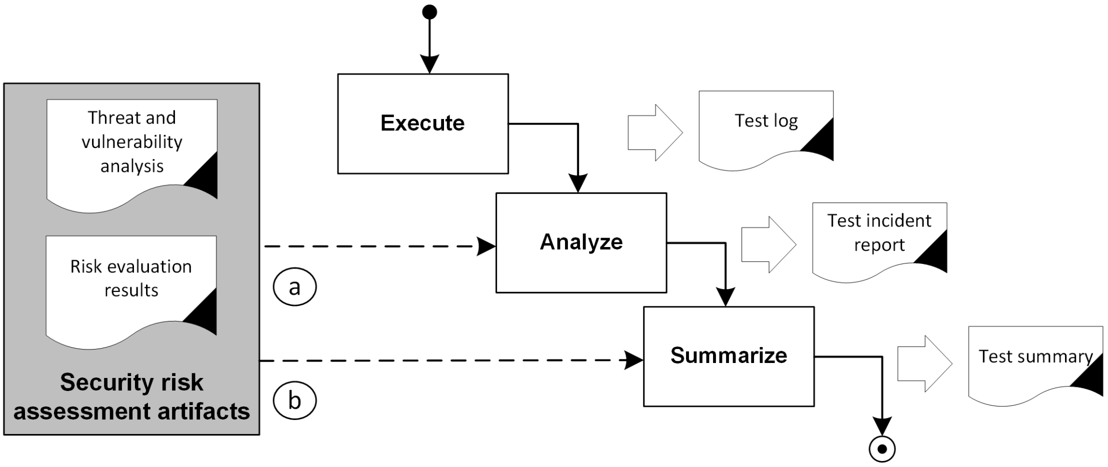


Figure 7 – Process model for risk-based test analysis and summary

The test analysis process is used for the evaluation of the test results and the reporting of test incidents. This process will be entered after the test execution and it mainly covers the analysis and evaluation of test failures and issues where something unusual or unexpected occurred during test execution. Its main purpose is to categorize the issues that occurred during testing and put them into context so that the test manager can rate them. Categorization and context provision can be simply done by referring to risks, threats and vulnerabilities from the security risk assessment.

Name	**Risked-based test result analysis (a)**
Actors	Security Tester (ST)
Tools	Test Execution Tool (STET), Security Risk Assessment Tool (SRAT)
Precondition	Test cases have been executed. Test cases already have a traceable relation to security risk assessment artifacts.
Postcondition	New and/or updated incidents are reported and assigned to either already detected vulnerabilities or to new vulnerabilities. Incidents that probably constitute new actual vulnerabilities are communicated so that they could be considered in the security risk assessment and/or the development.
Scenario	1. The Security Tester should analyze the test results (e.g., the test logs) and identify new incidents. 2. The Security Tester should classify newly identified incidents by means of their relation to artifacts from the security risk assessment (e.g., risks, threat scenarios, vulnerabilities). 3. The Security Tester should prioritize the newly identified incidents by means of associated artifacts from the security risk assessment. Issues related to critical risks should be rated higher than the ones that are associated with minor risks. 4. New and/or updated incidents are communicated to the relevant stakeholders.

Data exchanged/ processed	**In:** *Test logs, security risk assessment artifacts (vulnerabilities*, *threat scenarios*, *unwanted incident*, *likelihoods, consequences, risk level)* **Out:** *Incident report*

Table 9 – Risk-based test analysis and summary: Risked-based test result analysis (a)

Finally, the overall test results, i.e. the test verdicts, the issues and their categorization are summarized in a way, that the stakeholder could understand the outcome of the tests.

Name	**Risked-based test summary creation (b)**
Actors	Security Tester (ST)
Tools	Test Execution Tool (STET), Security Risk Assessment Tool (SRAT)
Precondition	Test cases have been executed Test cases already have a traceable relation to security risk assessment artifacts.
Postcondition	The test results are summarized respecting their relation to the a-priori identified security risks. The test report contains coverage of security risks
Scenario	1. The Security Tester should analyze the test logs and separate security risks that have been tested successfully (all tests are passed) and those that have not been tested successfully (issues have been found). 2. The Security Tester should (re-) characterize the security risks by interpreting the test results. To do so, the security tester should make use of dedicate test metrics to determine the quality of test procedures and thus the significance and validity of the test results.
Data exchanged/ processed	**In:** *Test logs, security risk assessment artifacts (vulnerabilities*, *threat scenarios*, *unwanted incident*, *likelihoods, consequences, risk level)* **Out:** *Test summary*

Table 10 – Risk-based test analysis and summary: Risked-based test summary creation (b)

2.2.2 Test based Security Risk Assessment

The main purpose of integrating the testing process into the risk assessment process is to use testing to enhance some of the activities of the risk assessment process. This is achieved by ensuring that test results are used as explicit input to the risk assessment.

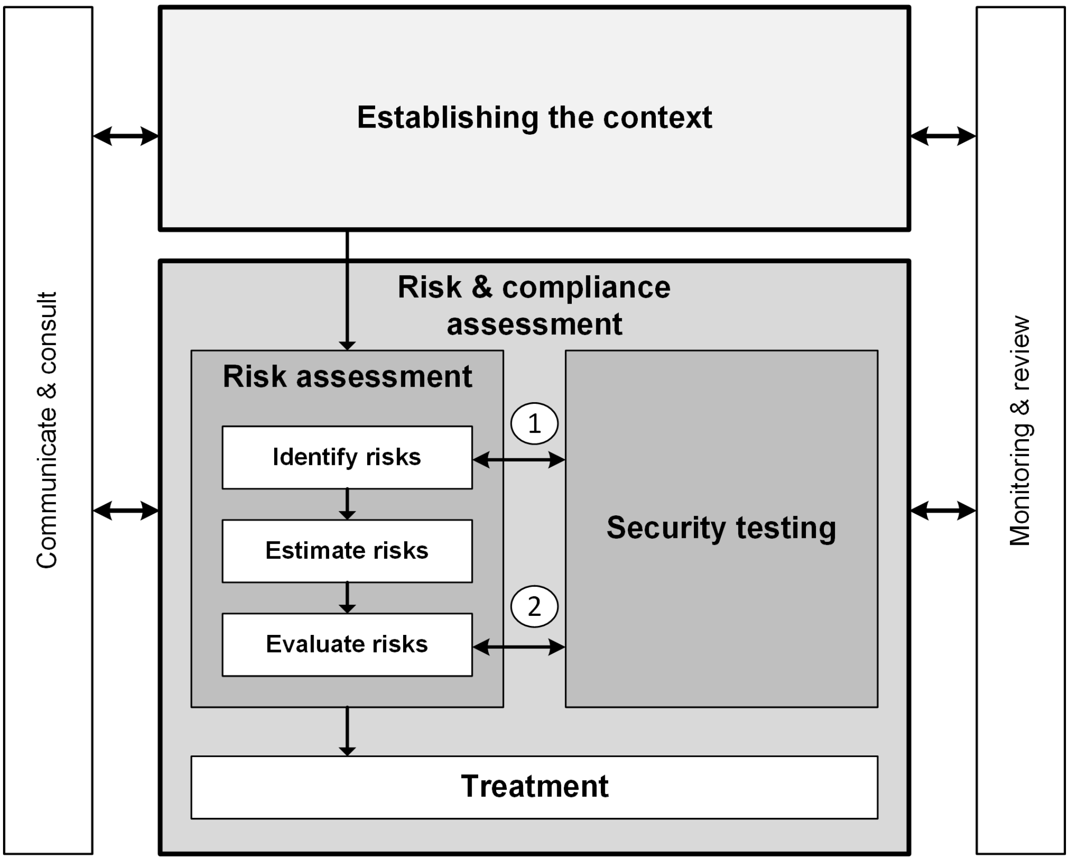


Figure 8 – Generic process for test-based risk assessment

Figure 8 shows how the unified RASEN process (initially introduced in Figure 1) is refined into a process for test-based risk assessment. Here the risk assessment activity has been decomposed into the three activities “identify risks”, “estimate risks” and “evaluate risks”. These three activities, together with the "establishing the context" and "treatment" activities form the core of the ISO 31000 risk management process [11].

As indicated in Figure 8, there are in particular two places where testing can in principle enhance the risk assessment process. This is explained in the following.

2.2.2.1 Test-based Risk Identification

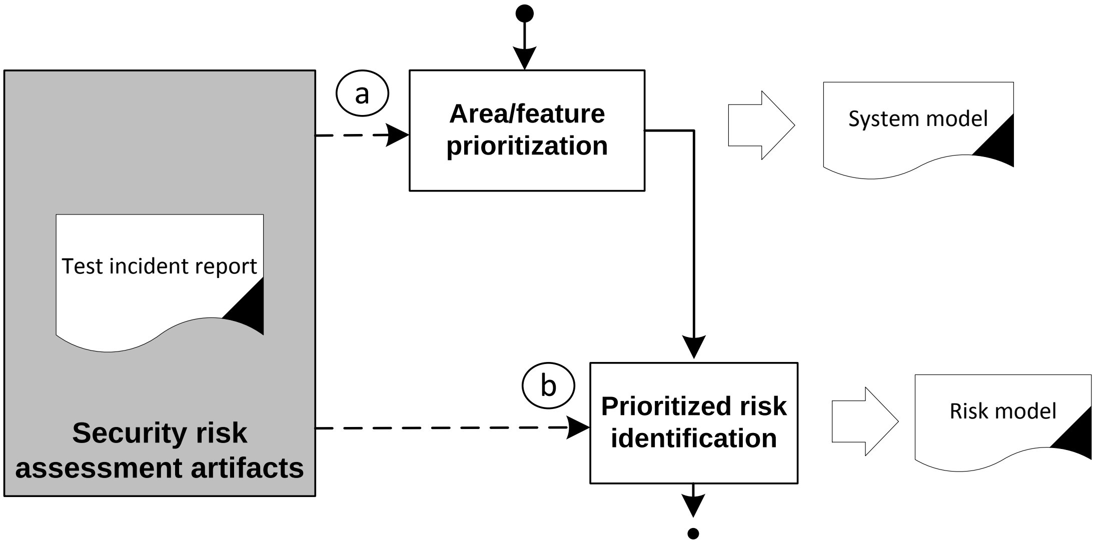

a
Area/feature prioritization
System model
Test incident report
Security risk assessment artifacts
b
Prioritized risk identification
Risk model

Figure 9 – Test-based risk identification

Risk identification is the process of finding, recognizing and describing risks. This involves identifying sources of risk, areas of impacts, events (including changes in circumstances), their causes and their potential consequences. Risk identification may involve historical data, theoretical analysis, informed and expert opinions, and stakeholder's needs.

The process of *test-based* risk identification uses test results to guide the risk identification process. As illustrated in Figure 9, the process is decomposed into two steps. In the first step, test results are used to prioritize areas or features of the target of evaluation for the purpose of risk identification. In the second step, the risk identification is performed on the basis of this prioritization. Particularly relevant in this setting is testing using automated testing tools such as vulnerability scanners or network discovery tools, or results from passing scanning/ monitoring.

Name	**Area/feature prioritization (a)**
Actors	Security Risk Analyst (SRA)
Tools	Security Risk Assessment Tool (SRAT), Security Testing Tool (STT),
Precondition	A test incident report must be available. A System model may be available.
Postcondition	The step must end with a system model that gives a priority of the system features which can be used for prioritizing the risk identification.
Scenario	1. The SRA analyses the system model and the test incident report. Based on this analysis, the SRA indicateswhich features or areas of the system model which should be prioritized in the risk identification step.
Data exchanged/ processed	**In:** *System model*, *test incident report* **Out:** *System model* (with prioritization of areas/features)

Table 11 – Test-based risk identification - Area/feature prioritization

Name	**Prioritized risk identification (b)**
Actors	Security Risk Analyst (SRA)
Tools	Security Risk Assessment Tool (SRAT), Test Specification Tool (STST),
Precondition	Same as the postcondition for step "area/feature prioritization"
Postcondition	The step must end with a risk model documenting the results of the risk identification.
Scenario	1. The SRA identifies risks on the basis of the system model. The identification process is prioritized according the priorities of the features/areas of the system as specified in the system model.
Data exchanged/ processed	**In:** *System model* (with prioritization of areas/features) **Out:** *Risk model*

Table 12 – Test-based risk identification - Prioritized risk identification

2.2.2.2 Test-based Risk Evaluation

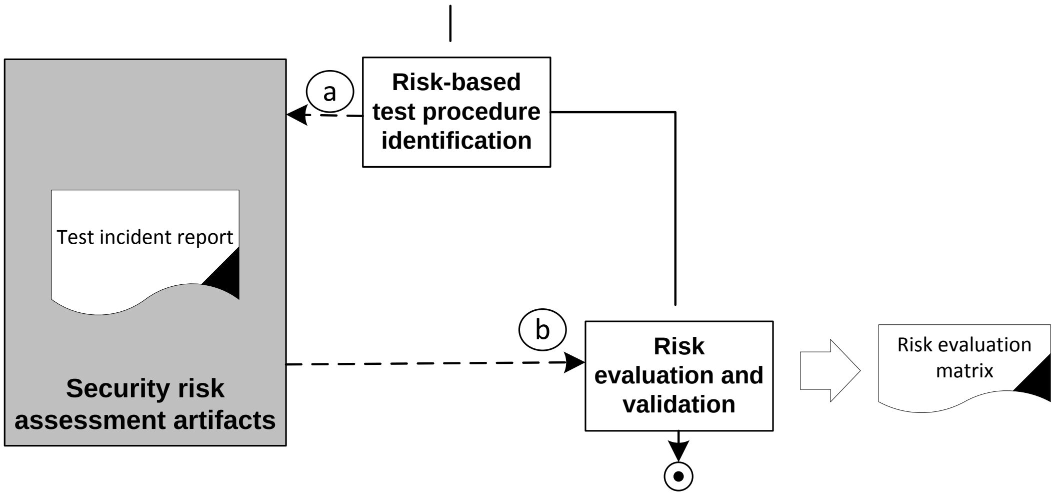

Risk-based test procedure identification
a
Test incident report
Security risk assessment artifacts
b
Risk evaluation and validation
Risk evaluation matrix

Figure 10 – Test-based risk evaluation

Risk evaluation is the process of comparing the results of risk estimation with risk criteria to determine whether the risk and/or its magnitude is acceptable or tolerable. In this context, testing process refers to the process of using testing to validate the correctness of the risk model. In particular, the likelihood estimates of the risk model might have a low confidence if they, e.g., depend on vulnerabilities whose presence in the target of analysis is unknown. By doing testing in this setting, we may investigate whether such vulnerabilities really are present in the target of analysis, and then use the test results to update the confidence level of the risk model.

As illustrated in Figure 10, the test-based risk evaluation process is decomposed into two steps. In the first step, the risk model is used as a basis for identifying high-level test procedures which can be used as a starting point for designing, implementing, and executing tests. In the second step, the test results are used to validate the correctness of the risk model and the resulting risks are evaluated.

Name	**Risk-based test procedure identification (a)**
Actors	Security Risk Analyst (ST)
Tools	Security Risk Assessment Tool (SRAT)
Precondition	A risk evaluation matrix and risk model with identified risks and estimations for likelihood and consequences must be available.
Postcondition	The step must result in a list of prioritized test procedures
Scenario	1. The ST analyzes the risk model and identifies elements that can be tested given the scope of the risk assessment. 2. The identified testable elements are prioritized based on the risk values and estimates of the risk model. 3. Based on the prioritization, a subset of the testable elements are selected and translated into high-level test procedures that have the purpose of checking these elements through testing.
Data exchanged/ processed	**In:** *Risk evaluation matrix*, *Risk model* **Out:**.*Test procedures*

Table 13 – Test-based risk evaluation - Risk-based test procedure identification

Name	**Risk evaluation and validation (b)**
Actors	Security Risk Analyst (ST)
Tools	Security Risk Assessment Tool (SRAT)
Precondition	A risk evaluation matrix, risk model with identified risks and estimations for likelihood and consequences, and test results must be available.
Postcondition	The step must result in a risk evaluation matrix showing the risk values of the risks identified in the risk assessment.
Scenario	1. The ST links test results to elements of the risk model and updates/validates the correctness the estimates of the risk model based on the new information obtained through the testing.
Data exchanged/ processed	**In:** *Risk evaluation matrix*, *Risk model*, *Test incident report* **Out:** *Risk mode*l, *Risk evaluation matrix*

Table 14 – Test-based risk evaluation - Risk evaluation and validation

3 Specific RASEN Methodologies

In this section we describe specific RASEN methodologies. These methods can be seen as refinements of the generic method described in Section 2.

In Sections 3.1 and 3.1.1, we describe two alternative processes for test-based risk assessment. The first one mainly addresses test case derivation by use of patterns, while the latter mainly addresses the use of risk assessment for test procedure identification and selection/prioritization. In the final section, Section 3.3, we describe a method for legal risk assessment.

3.1 Test Pattern supported Risk-based Security Testing

3.1.1 Overview

Security testing and thus, risk-based security testing could additionally gain from reusing existing test knowledge. Security test patterns are a way to formulate a solution for recurring security testing problems in a structured way. The method described here is an instantiation of the risk-based security test design and implementation activities defined in Section 2.2.1.2. It uses the notion of security test patterns as the central element to serve knowledge transfer and reuse within the test specification process.

3.1.2 Process Description

The following description is a slightly modified version of the activity **Risk-based identification and prioritization of features sets (a)** from Section 2.2.1.2**.** Instead of creating and prioritizing dedicated feature sets, this activity aims for prioritizing vulnerabilities that are already linked with dedicated features or artifacts.

Name	**Risk-based security feature identification and prioritization**
Actors	Security Tester (ST)
Tools	Risk Assessment Tool (SRAT), Security Testing Tool (STT)
Precondition	A risk assessment model with likelihood and consequence estimates
Postcondition	A prioritized list of testable features
Scenario	1. ST assigns vulnerabilities and threat scenarios to features (interfaces, operations, components) of a test item. 2. ST identifies and prioritizes potential vulnerabilities and threat scenarios according to their impact on the overall risk picture. 3. ST identifies the vulnerabilities that have the highest impact when they are mitigated.
Data exchanged/ processed	**In (from SRAT):** *Vulnerabilities*, *threat scenarios*, *unwanted incident*, *likelihoods, consequences, risk level* **Out (from ST):** *Vulnerabilities* with *priority score*, *testable features*

Table 15 – Activity: Risk-based security feature identification and prioritization

Table 16 describes an extended version of the activity **Risk-based derivation of test conditions and test coverage items (b)** from Section 2.2.1.2. In addition to the generic approach from above, this version introduces the notion of security test pattern. A security test pattern is a reusable asset that covers an already approved set of test technique, test completion criteria, test coverage item specification etc. for a given security testing problem (e.g., a suspected vulnerability or a threat scenario).

Name	**Security test pattern based derivation of test techniques, test conditions and test coverage items**
Actors	Security Tester (ST)
Tools	Security Testing Tool (STT)
Precondition	*A selected set of Vulnerabilities* with *priority score with associated testable features*
Postcondition	Vulnerabilities with associated test pattern (containing test technique, test completion criteria, test coverage item specification)
Scenario	1. ST assigns vulnerabilities to test pattern (containing test technique, test completion criteria, test coverage item specification)
Data exchanged	**In :** *Vulnerabilities* with *priority score,testable features* **Out :** *Vulnerabilities* with associated *test pattern* and updated *priority score*

Table 16 – Activity: Risk-based derivation of test conditions and test coverage items

Table 17 describes an instantiation of the activity **Risk based derivation of test cases (c)** from Section 2.2.1.2. This activity describes an automated process of test derivation that is guided by test patterns. We distinguish the two variants A. and B. While variant A. relays on an automated test generation on basis of test pattern only, variant B additionally uses behavioral models as well as test model to concisely interact with the behavior of the SUT.

Name	**Security test generation**
Actors	Security Tester (ST)
Tools	Security Testing Tool (STT), Security Testing Derivation Tool (STDT)
Precondition	*Vulnerabilities* with associated *test pattern* and updated *priority score* *Behavioral/Environmental information (model of the SUT, test system specific information)*
Postcondition	*Test procedures* associated to *test pattern* and *vulnerabilities*
Scenario	1. ST generates/realizes *test cases* and *test procedures* according to the information given by the test pattern (test technique, test completion criteria, test coverage item specification). 2. (Alternatively) ST generates/realizes *test cases* and *test procedures* by automatically animating *the behavioral/environmental test model* according to the test purpose and priority score information (test technique, test completion criteria, test coverage item specification).
Data exchanged	**In :** *Test pattern* and updated *priority score* **Out :** *Test procedures* and *test cases*

Table 17 – Activity: Security test generation

Table 18 describes an instantiation of the activity **Risk-based test result analysis (a)** from Section 2.2.1.3. It describes the execution of security tests and the process of test result analysis under special consideration and in relation to the associated security risks.

Name	**Security test execution and result analysis**
Actors	Security Tester (ST)
Tools	Security Testing Tool (STT)
Precondition	*Test procedures* and *test cases*
Postcondition	*Test log* and *test incident report*
Scenario	1. ST executes *test procedures* and creates the *test log* and the *test incident report*.
Data exchanged	**In :** *Test procedures* and *test cases* **Out :** *Test log and test incident report*

Table 18 – Activity: Security test execution and result analysis

3.1.3 Exemplification of Method

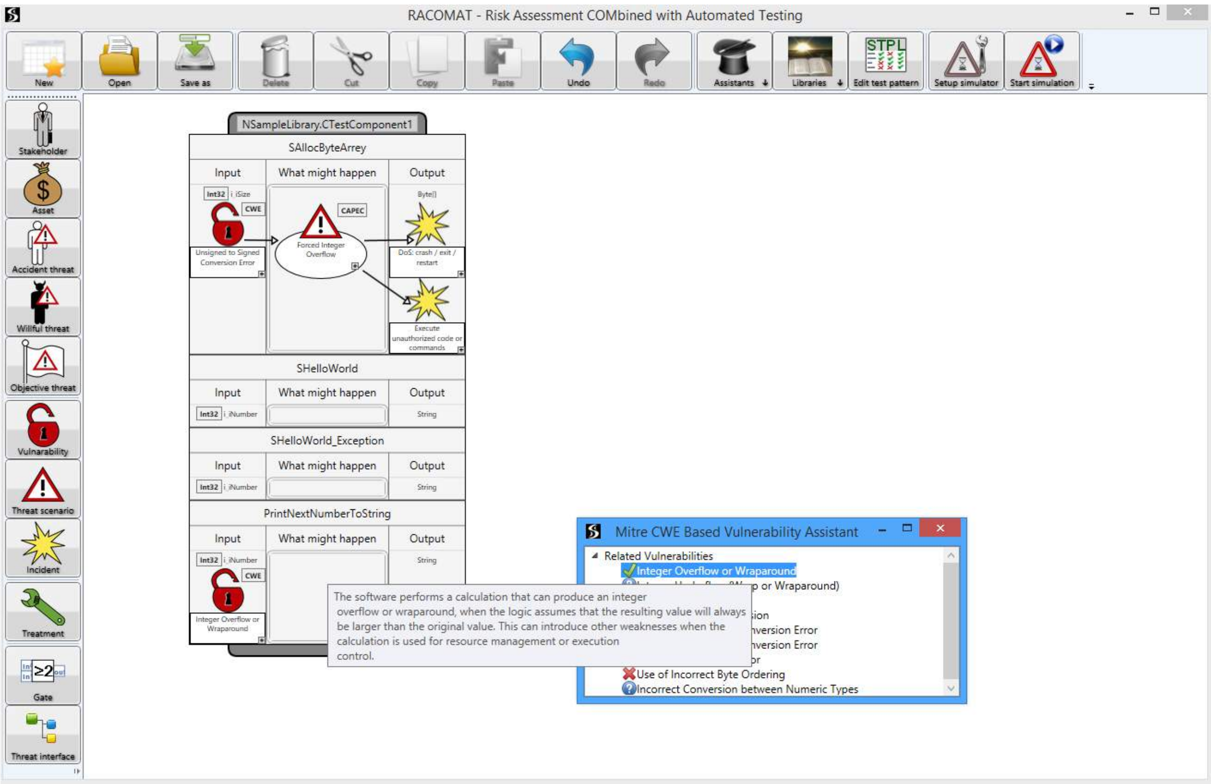
Based on our method for risk-based security testing and test-based risk assessment we have developed the RACOMAT tool, which provides assistance for the whole risk-based security testing and test-based risk assessment process (see also RASEN Deliverables D3.2.2 and 4.3.2). The RACOMAT tool takes the role of a SRAT, STET, STDT and STET. In order to reduce the amount of manual work as far as possible, the tool tries to maximize the reusability of risk analysis artifacts by introducing the notion of reusable risk analysis artifacts, test pattern and testing metrics as a central element of the method. For risk analysis, the tool uses an extended version of CORAS supporting compositionality with the help of reusable *threat interfaces* as described in RASEN Deliverable D.3.2.2. The risk graph that is generated with the RACOMAT tool contains besides the risk related information some information about the target system itself, i.e. interface definitions and other structural information. This information is valuable especially for automated testing since it allows to directly identifying test interfaces and the related features to test.

In the following example, we are generating tests for a static C# function from a sample library called *PrintNextNumberToString*. The function has one input parameter of type 'signed 32 bit integer'.

3.1.3.1 Risk-based Security Feature Identification and Prioritization

To support the identification of testable security features RACOMAT allows the assignment of unwanted incidents, threat scenarios and vulnerabilities to so called threat interfaces. The RACOMAT tool automatically generates partial threat interfaces for components from existing compiled binaries or from the source code of the components. Hence, there is no need to manually create models describing the interfaces if none are available.

The security tester or the risk analysts complete the partial threat interfaces by adding unwanted incidents, threat scenarios and vulnerabilities. While this involves some manual work, the analysts can take advantage of our tool's assistants using existing risk related databases like CWE or CAPEC. For example, CWE based vulnerabilities can be dragged to input ports of threat interface instances, which automatically associates the risk graph element with the system port. After simply clicking on a system port, the RACOMAT tool also provides an optimized list containing only those vulnerabilities which are typically associated with the type of the system port. Hence there is no need to look through the entire vulnerabilities catalogue. The security tester is further supported with suggestions for other nodes like threat scenarios, which might typically also be relevant in conjunction with already inserted nodes. For such suggested elements, even the relations to present nodes are created automatically as soon as they are inserted.



The software performs a calculation that can produce an integer overflow or wraparound, when the logic assumes that the resulting value will always be larger than the original value. This can introduce other weaknesses when the calculation is used for resource management or execution control.

Figure 11 – RACOMAT vulnerability assistant showing only port type related items

In our example shown in Figure 11, the RACOMAT tool automatically suggests to add a vulnerability called “Integer Overflow or Wraparound”, which was generated from CWE190. The vulnerability contains an initial likelihood value “Medium” because CWE190 says this is the likelihood that such a weakness is exploited. This likelihood information can be used for identifying the priority of testing related threat scenarios. The menu of the CWE based vulnerability “Integer Overflow or Wraparound” contains suggestions for potentially related threat scenarios that correspond to CAPEC attack patterns. In the example, the threat scenario “Forced Integer Overflow” is suggested, which is based on CAPEC92. The analyst can insert the threat scenario by dragging it to the risk graph. The relation from the vulnerability “Integer Overflow or Wraparound” to the “Forced Integer Overflow” threat scenario is automatically added.

3.1.3.2 Security Test Pattern-based Derivation of Test Techniques, Test Conditions and Test Coverage items

To support the identification of applicable test techniques or strategies, the RACOMAT tool allows assigning security test pattern to vulnerabilities. A security test pattern contains, beside the description of the test technique in natural language, additional information for (semi-)automatic test case generation. This information identifies dedicated methods for test case generation and dedicated test strategies, which describe the way how a certain test design technique shall be implemented in order to generate test cases automatically. Additionally, a security test pattern provides the estimated effort for testing as well as an indicator that describes how likely it is to find a vulnerability using the test technique proposed by the pattern.

The RACOMAT tool automatically proposes applicable test patterns from our test pattern library. In fact, there are two test patterns that are applicable to our example. They differ in their strategies, directives and metrics. One test pattern proposes the generation of extreme and special integer values like maximum, minimum and zero. The second test pattern additionally proposes to use a data fuzzing strategy to create a certain number of random test values.

Each test pattern contains a list of unwanted incidents that might be detected when an instance of the test pattern is executed. In our example, the test patterns related to “Forced Integer Overflow” both show an unwanted incident called “Unhandled integer overflow”. For instantiation, the unwanted incidents of a test pattern have to be dragged to the output ports of the system where the incidents might eventually be observed. Figure 12 shows the completed instantiation of the security test pattern with data fuzzing for our sample function.

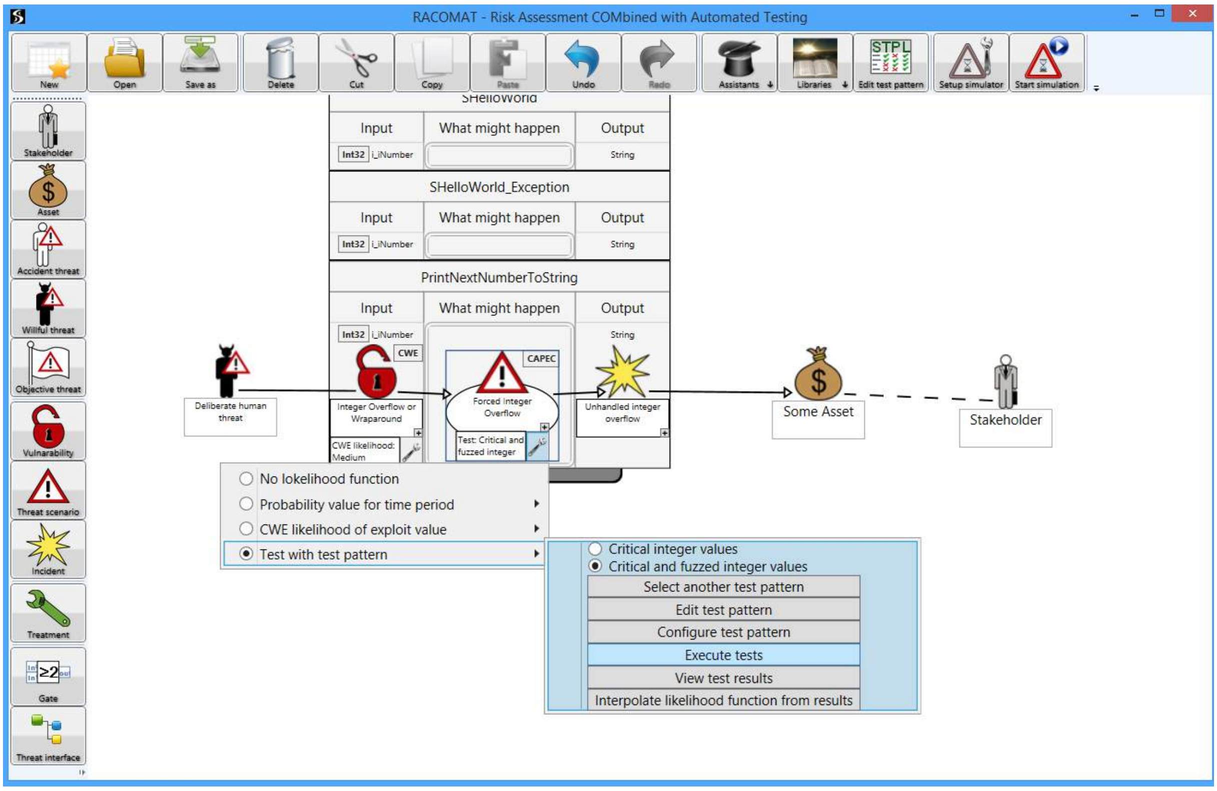


Figure 12 – Test pattern with data fuzzing fully instantiated

3.1.3.3 Security Test Generation

Both security test patterns have associated test generators that can be used for the automated generation of security tests. While the generator for the first test pattern only generates the extreme and special integer values like maximum, minimum and zero. The second test pattern additionally uses a data fuzzing strategy and creates a certain number of random test values. The generator of the second test pattern has multiple optional parameters. One can be used to directly set the number of fuzz test cases that should be generated. Our tool allows for setting this parameter manually or to directly derive it from priority values calculated on basis of the risk graph (see Risk-based security feature identification and prioritization).

3.1.3.4 Security Test execution and Result Analysis

For actually executing the test cases, the threat interfaces are evaluated to identify which functions have to be called with which parameters and what has to be monitored. In the example we present here, this requires no additional manual work at all. The RACOMAT tool compiles the code taken or automatically generated from test patterns. The related vulnerability that is associated with the system input port tells the RACOMAT tool how to pass the generated test values to the system. The related unwanted incident indicates monitor the system output port the incident is related to.

To allow an interpretation of the observed raw test results, two different versions of the component under test are generated. A *test version* that will throw an exception on any arithmetic integer overflow unless the code explicitly prevents it and an unmodified *release version*. Each test value is first tried with the *test version* of the component that will throw arithmetic overflow exceptions by default. If an exception is thrown, then the same test value is tested against the *release version* that does not throw arithmetic overflow exceptions by default. If again the overflow exception is observed, then the *release version* detects the overflow correctly. Of course, whenever the tested function is called, the overflow exceptions must be treated properly, but throwing the exception itself in the *release version* is not considered to be an error, it is not necessarily an unwanted incident. If treated correctly by the caller, the program might continue without problems. Figure 13 shows how the testing and observation basically works.

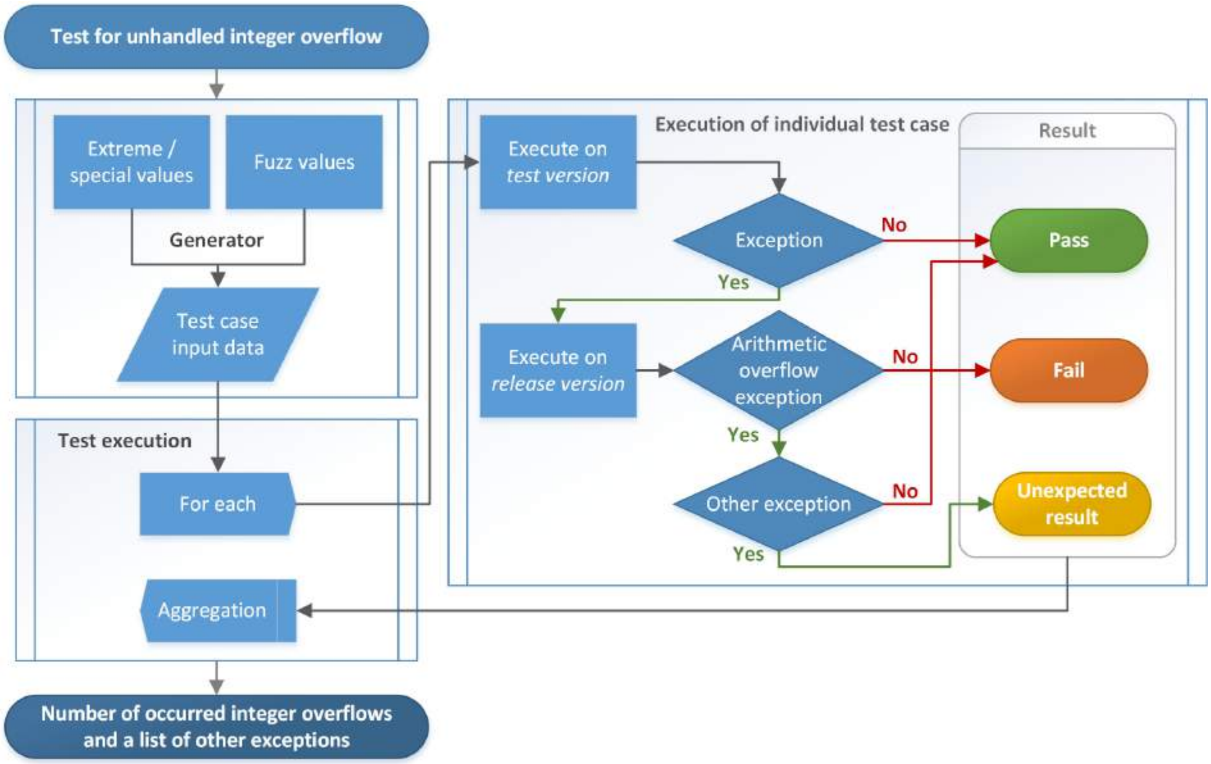

Test for unhandled integer overflow
Extreme / special values
Fuzz values
Generator
Test case input data
Test execution
For each
Aggregation
Number of occurred integer overflows and a list of other exceptions
Execution of individual test case
Execute on test version
Exception
No
Yes
Execute on release version
Arithmetic overflow exception
No
Yes
Other exception
No
Yes
Result
Pass
Fail
Unexpected result

Figure 13 – Testing and observation process

The observation just yields raw results. While a fail result for the exemplary integer overflow testing process with fuzzing indicates, that the related threat scenario could be used for a successful attack and that therefore the vulnerability is exploitable for sure, interpreting the other possible results is more challenging. The RACOMAT tool lets the user choose a security testing metric suggested by the security test pattern for further analysis and interpretation.

If all tests pass, then it makes sense to use one of the coverage or efficiency security testing metrics to calculate the likelihood that an attack could still be possible even though testing failed to trigger the unwanted incident. The RACOMAT tool can update the risk graph automatically with updated likelihood values.

If there are unexpected results, then a list up security testing metric should be used. The RACOMAT tool generates automatically unwanted incidents for the unexpected results that can be added to the risk graph by drag and drop for further analysis.

3.2 CORAS Method for Test-Based Risk Assessment

In this section, we describe a specific method for test-based risk assessment which can be seen as an extension of the CORAS method for risk assessment [14].

3.2.1 Overview

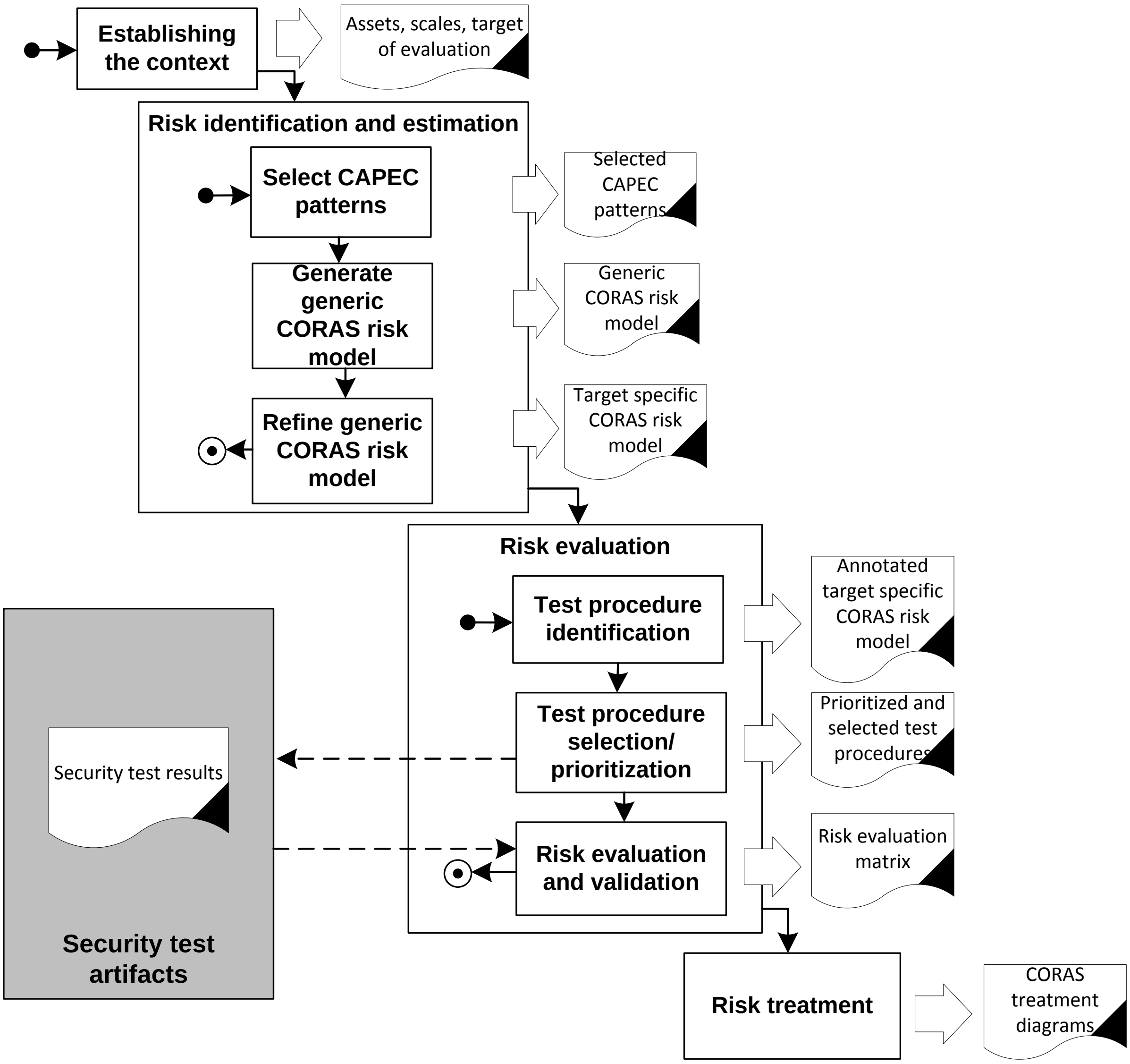

Establishing the context
Assets, scales, target of evaluation
Risk identification and estimation
Select CAPEC patterns
Selected CAPEC patterns
Generate generic CORAS risk model
Generic CORAS risk model
Refine generic CORAS risk model
Target specific CORAS risk model
Risk evaluation
Test procedure identification
Annotated target specific CORAS risk model
Test procedure selection/ prioritization
Prioritized and selected test procedures
Risk evaluation and validation
Risk evaluation matrix
Security test results
Security test artifacts
Risk treatment
CORAS treatment diagrams

Figure 14 – Steps of the CORAS test-based risk assessment method

The main steps of the process of test-based risk assessment are illustrated in Figure 14. The process can both be seen as an extension of the CORAS method for risk assessment and as an instance of the generic RASEN method for test-based risk assessment shown in Figure 8. There are two main differences between the specific and the generic process.

- The first is that the step "Risk identification and estimation" of the generic process is instantiated into three steps of the specific process. These steps enable the automated generation of the risk model through translation from the CAPEC catalog of security attack patterns.
- The second is that the risk evaluation step is instantiated into three steps of the specific process that enable validation of the risk model trough testing.

3.2.2 Process Description

In the following, we document each step of the process using the template described in Table 1.

Name	**Establish objective and context**
Actors	Risk Analyst (RA), Customer (C)
Tools	Security Risk Assessment Tool (SRAT)[the CORAS tool]
Precondition	None
Postcondition	The activity must end with the following output: • A description of the target of analysis, • A description of the assumptions, focus and scope of the analysis, • CORAS asset diagrams defining assets and parties, • Tables defining consequence and likelihood scales, and • Risk matrix tables defining risk evaluation criteria.
Scenario	1. The Risk Analyst describes the target of analysis (for instance using UML) based on documentation that is already available and discussion with the Customer. 2. The Risk Analyst documents assumptions, focus and scope of the analysis in natural language in addition to the system documentation. 3. Based on discussion with the Customer, the Risk Analyst documents • assets and parties using CORAS asset diagrams using the Security Risk Assessment Tool; • at least one likelihood scale which will later be used when estimating the likelihood of risks; • one consequence scale for each identified asset which will later be used when estimating the consequences of risks; • risk evaluation criteria for each asset using a risk matrix.
Data exchanged/ processed	**Out (fromCORAS tool):** Risk model with identified Assets

Table 19 – Activity: Establish objective and context

Name	**Select CAPEC patterns**
Actors	Risk Analyst (RA), Customer (C)
Tools	Security Risk Assessment Tool (SRAT) [the CORAS tool]
Precondition	The CAPEC catalog of security attack patterns must be available

Postcondition	The activity must end with the following output: • A selection of CAPEC patterns
Scenario	1. The Risk analyst, optionally together with the Customer, surveys each CAPEC security attack pattern in the CAPEC catalog to determine which patterns will be used as the basis for the risk assessment. 2. The decision regarding which pattern will be selected is specified by use of the SRAT tool
Data exchanged/ processed	**In (from CORAS tool):** CAPEC catalog **Out (from CORAS tool):** A selection of CAPEC patterns.

Table 20 – Activity: Select CAPEC patterns

Name	**Generate generic CORAS risk model**
Actors	Risk Analyst (RA)
Tools	Security Risk Assessment Tool (SRAT) [the CORAS tool]
Precondition	The precondition of this activity is the same as the postcondition of the activity "Select CAPEC patterns "
Postcondition	The activity must end with the following output: • A generic CORAS risk model
Scenario	1. The risk analysis uses the SRAT tool to automatically generate a risk model from the selected CAPEC patterns. The risk model is generic because it is not yet specific to the target of evaluation.
Data exchanged/ processed	**In (from CORAS tool):** Risk model with identified Assets **Out (from CORAS tool):** A generic CORAS risk model.

Table 21 – Activity: Generate generic CORAS risk model

Name	**Refine generic CORAS risk model**
Actors	Risk Analyst (RA), Customer (C)
Tools	Security risk assessment tool (SRAT) [the CORAS tool]
Precondition	The precondition of this activity is the same as the postcondition of the activity "Generate generic CORAS risk model"
Postcondition	The activity must end with the following output: • A CORAS risk model with identified risks and likelihood and consequence estimates which are specific to the target of evaluation.

Scenario	1. The Risk Analyst and the Customer uses the SRAT tool to refine the generic CORAS risk model into a model which is specific to the target of evaluation. This activity may involve: • adjustment/verification of all likelihood estimates • identification of new threat scenarios and unwanted incidents where appropriate • splitting or merging of threat scenarios/unwanted incidents where appropriate.
Data exchanged/ processed	**In (from CORAS tool):**A generic CORAS risk model **Out (fromCORAS tool):** A target specific CORAS risk model

Table 22 – Activity: Refine generic CORAS risk model

Name	**Test procedure identification**
Actors	Risk Analyst (RA), Security Tester (ST)
Tools	Security risk assessment tool (SRAT) [the CORAS tool]
Precondition	The precondition for this activity is the same as the postcondition of the activity "Refine generic CORAS risk model".
Postcondition	The activity must end with the following output: • A CORAS risk model annotated with effort estimates.
Scenario	1. The risk analyst and the security tester walkthrough the CORAS risk model and annotate each element of the risk model that can be potentially be tested with an estimate indicating the effort/time it will take to test the element. By "test the element" we mean the activity of designing, implementing, and executing tests which can be used for verifying the correctness of a statement derived from the risk model element.
Data exchanged/ processed	**In (fromCORAS tool):** A CORAS risk model **Out (fromCORAS tool):** A CORAS risk model annotated with effort estimates.

Table 23 – Activity: Test procedure identification

Name	**Test selection/prioritization**
Actors	Security Tester (ST)
Tools	Test Derivation Tool (TDT) [the CORAS tool]
Precondition	The precondition for this activity is the same as the postcondition of the activity "Test procedure identification ".
Postcondition	The activity must end with the following output: • A list of prioritized test procedures for deriving test cases
Scenario	1. The security tester estimates the maximum time/effort available for testing. 2. The security tester used the TDT tool to automatically generate an optimal prioritized list of test procedures whose total effort is equal to or below the maximum available effort. 3. The security tester exports the selected test procedures from the TDT tool to the RASEN generic data format, allowing for import into the security testing tools.
Data exchanged/ processed	**In (from CORAS tool):** A CORAS risk model annotated with effort estimates. **Out (from CORAS tool):** A prioritized list of test procedures documented in the RASEN generic format.

Table 24 – Activity: Test selection/prioritization

Name	**Risk evaluation and validation**
Actors	Risk Analyst (RA), Customer (C)
Tools	Security Risk Assessment Tool (SRAT) [the CORAS tool],Security Test Aggregation Tool (STAT)
Precondition	A test incident report linking the test procedures identified in the previous step to test measurements.
Postcondition	The activity must end with the following output: • An updated risk model (based on the test results) • A risk evaluation matrix showing the risk levels of the identified risks
Scenario	1. The risk analyst imports the test procedures with associated test results into the STAT tool and aggregates the test results into risk assessment measurements.. 2. The risk analyst imports the test procedures with the risk assessment measurements in to the SRAT tool which automatically calculates that impact that these results have on the risk model. 3. The risk analyst updates the risk model based on the impact assessment. 4. The risk analyst produces a risk evaluation matrix with the identified risks.

Data exchanged/ processed	**In (from STT to STAT):** Test log, test incident report. **Out (from STAT to CORAS tool):** Test procedures with risk assessmentmeasurements. **Out (from CORAS tool):** An updated risk model (based on the test results) and a risk matrix with identified risks.

Table 25 – Activity: Risk evaluation and validation

Name	**Risk treatment**
Actors	Risk Analyst (RA), Customer (C)
Tools	Security Risk Assessment Tool (SRAT) [the CORAS tool],
Precondition	The precondition for this activity is the same as the postcondition of the activity "Risk evaluation and validation".
Postcondition	The activity must end with the following output: • A CORAS risk model documenting identified treatments, i.e. a set of CORAS treatment diagrams.
Scenario	1. The risk analyst identifies treatments on the basis of the risk model and the risk matrix and documents these using CORAS treatment diagrams. This activity is identical to the treatment activity of the CORAS risk assessment method.
Data exchanged/ processed	**In (from CORAS tool)**: Risk matrix, risk model. **Out (from CORAS tool):** A CORAS risk model with treatments.

Table 26 – Activity: Risk treatment

3.2.3 Exemplification of Method

In this section, we demonstrate the process with an example. In particular, we give examples of outputs for each step of the method.

3.2.3.1 Establishing the Context

This activity is based on Step 1 – Step 4 of the CORAS risk assessment methodology. The output of the activity is:

- A description of the target of analysis,
- A description of the assumptions, focus and scope of the analysis,
- CORAS asset diagrams defining assets and parties,
- Tables defining consequence and likelihood scales, and
- Risk matrix tables defining risk evaluation criteria.

The description of the target of analysis should be based on the documentation that is already available of the system that is analyzed. If this documentation is not sufficient, then a new (high-level) description of the target may have to be specified. A graphical description of the target system (for instance using UML) is preferred as this may make the risk identification easier.

The assumptions, focus and scope of the analysis should be documented in natural language in addition to the system documentation.

Assets and parties should be documented using CORAS asset diagrams. An asset is something to which a party assigns a value and hence for which the party requires protection. A party is an organization, company, person, group or other body on whose behalf a risk assessment is conducted. Typically, there is only one party (the customers on whose behalf the risk assessment is conduced), but there may be more than one.

Identifying and documenting assets is an important part of the risk assessment as every risk will be related to one or more assets. If a party has no assets to speak of, then there is no need to conduct a risk assessment.

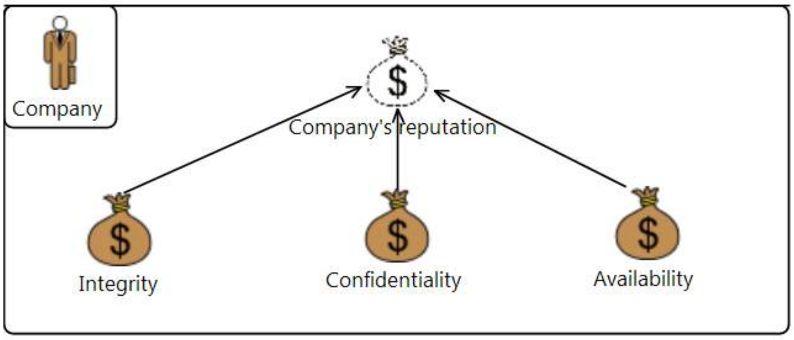


Figure 15 – CORAS asset diagram example

An example of a CORAS asset diagram is illustrated in Figure 15. The party (Company) which assigns values to the assets is specified in the top left corner of the diagram. In the diagram proper, three assets are specified. So-called harms relationships between the assets are also specified using arrows. A harms relation expresses that an asset can be harmed through harm to another asset.

Three likelihood scales should be defined that are suitable for estimating likelihood values related to security attacks as described in CAPEC attack patterns. The three kinds of likelihood scales should be used to estimating the likelihood:

- of attack initiation (i.e. of often is a security attack initiated regardless of whether or not it is successful),
- that an attack will succeed if it is initiated,
- the likelihood that an successful attack will lead to unwanted incidents.

Examples of the three types of likelihood scales are shown below in Table 27- Table 29.

Likelihood	Definition	Interval
Seldom	Less than 1 times per 10 years	[0, 0.1>:1y
Unlikely	1-10 times per 10 years	[0.1,1]>:1y
Possible	2-12 times per year	[1,13>:1y
Probable	13-60 times per year	[13,61>:1y
Certain	Over 60 times per year	[61,Infinity>:1y

Table 27 – Likelihood scale for estimating risk and attack initiation

Likelihood	Definition	Probability
Very little	1 out of 100000 attacks is successful	0.00001
Little	1 out of 10000 attacks is successful	0.0001
Small	1 out of 1000 attacks is successful	0.001
Medium	1 out of 100 attacks is successful	0.01
High	2 out of 10 attacks is successful	0.2

Table 28 – Conditional likelihood scale for estimating probability of successful attacks

Likelihood	Definition	Probability
iLow	1 out of 10 successful attacks will cause an incident	0.10
iMedium	1 out of 10 successful attacks will cause an incident	0.25
iHigh	1 out of 2 attacks will cause an incident	0.5

Table 29 – Conditional likelihood scale for estimating the probability that a successful attack will cause an unwanted incident

In addition to defining likelihood scales, we recommend defining a scale for expressing the confidence in the correctness of the estimated conditional likelihood values. An example of such a confidence scale is shown in Table 30. The scale is defined in terms of a buffer which can be used to translatea given conditional likelihood into a likelihood interval. For instance, given a conditional likelihood of say, 0.3 and the confidence value Medium, we can create the interval [0.3 – *b*, 0.3 + *b*], where *b* is the buffer for confidence Medium. In this case, the buffer of Medium is 0.01, so the interval will be [0.29, 0.31].

Confidence	Definition	Buffer
Low	Low confidence	0.001
Medium	Medium confidence	0.01
High	High confidence	0.1

Table 30 – Confidence scale for conditional likelihoods

One consequence scale for each asset should be defined. An example of the definition of consequence scales for the assets "Availability" and "Confidentiality" are shown in Table 31 and Table 32, respectively.

Consequence	Definition
Insignificant	Service unavailable for [0 min, 15 min>
Small	Service unavailable for [15 min, 1 h>
Medium	Service unavailable for [1 h, 8 h>
High	Service unavailable for [8 h, 24*7 h>
Critical	Service unavailable for [24*7 h, ….>

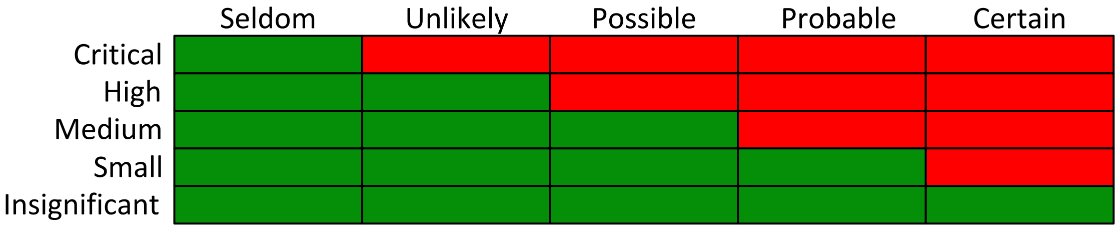
Table 31 – Consequence scale for Availability asset

Consequence	Definition
Insignificant	[0, 1> customers affected by confidentiality breach
Small	[1, 2> customers affected by confidentiality breach
Medium	[2, 10> customers affected by confidentiality breach
High	[10, 50> customers affected by confidentiality breach
Critical	[50, ...> customers affected by confidentiality breach

Table 32 – Consequence scale for Confidentiality asset

Having defined likelihood and consequence scales, risk evaluation criteria should be defined using risk matrices. It is easiest to define only one risk evaluation matrix. However, sometimes it makes more sense to define one risk matrix per asset.

An example of a risk evaluation matrix is given in Figure 16. Here risk values are denoted by greenand red. It's up to the risk analysis to define what is meant by these, but typically risks that have a red risk value must be considered for treatment and green risks can be accepted without being considered for treatment.



	Seldom	Unlikely	Possible	Probable	Certain
Critical	🟩	🟥	🟥	🟥	🟥
High	🟩	🟩	🟥	🟥	🟥
Medium	🟩	🟩	🟩	🟥	🟥
Small	🟩	🟩	🟩	🟩	🟥
Insignificant	🟩	🟩	🟩	🟩	🟩

Figure 16 – Risk evaluation criteria example

3.2.3.2 Select CAPEC Patterns

The purpose of this activity is to select a set of CAPEC attack patterns which will be used as a basis for generating a CORAS risk model. The outcome, then, is

- a set of selected CAPEC attack patterns.

A CAPEC attack pattern contains a lot of information which is not needed for translation into a risk model. For a more detailed description of this, the reader is referred to the RASEN deliverable D4.2.2.

For the purpose of the current example, we assume that two CAPEC attack patterns are selected: CAPEC-66 and CAPEC-88. These are shown in Table 33 and Table 34, respectively.

Attribute	Description
Name	(CAPEC-66, SQL Injection)
Typical likelihood of exploit	Very high
Attack motivation-consequences	(Modify application data, {Integrity}), (Read application data, {Confidentiality}), (Execute unauthorized code or commands, {Confidentiality, Integrity, Availability}), (Gain privileges / assume identity, {Confidentiality})
CIA Impact	(High, High, High)
CWE ID (Related weaknesses)	CWE-89 Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')

Table 33 – Example of CAPEC attack pattern 66

Attribute	Description
Name	(CAPEC-88, OS Command Injection)
Typical likelihood of exploit	High
Attack motivation-consequences	(Read application data, {Confidentiality}), (Execute unauthorized code or commands, {Confidentiality,

	Integrity, Availability}), (Gain privileges / assume identity, {Confidentiality}) (Bypass protection mechanism, {Confidentiality})
CIA Impact	(High, High, High)
CWE ID (Related weaknesses)	CWE-78 Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection') CWE-697 Insufficient Comparison CWE-713 OWASP Top Ten 2007 Category A2 - Injection Flaws

Table 34 – Example of CAPEC attack pattern 88

3.2.3.3 Generate Generic CORAS Risk Model

The purpose of this activity is to generate a CORAS risk model from the selected CAPEC attack patterns. The outcome of the activity is

- a CORAS risk model.

This risk model will be used as a starting point for risk identification and estimation.

The risk model can be automatically generated from the selected CAPEC patterns (as explained in deliverable D4.2.2). Figure 17 and Figure 18 show the CORAS risk models, which have been translated from CAPEC-66 and CAPEC-88, respectively.

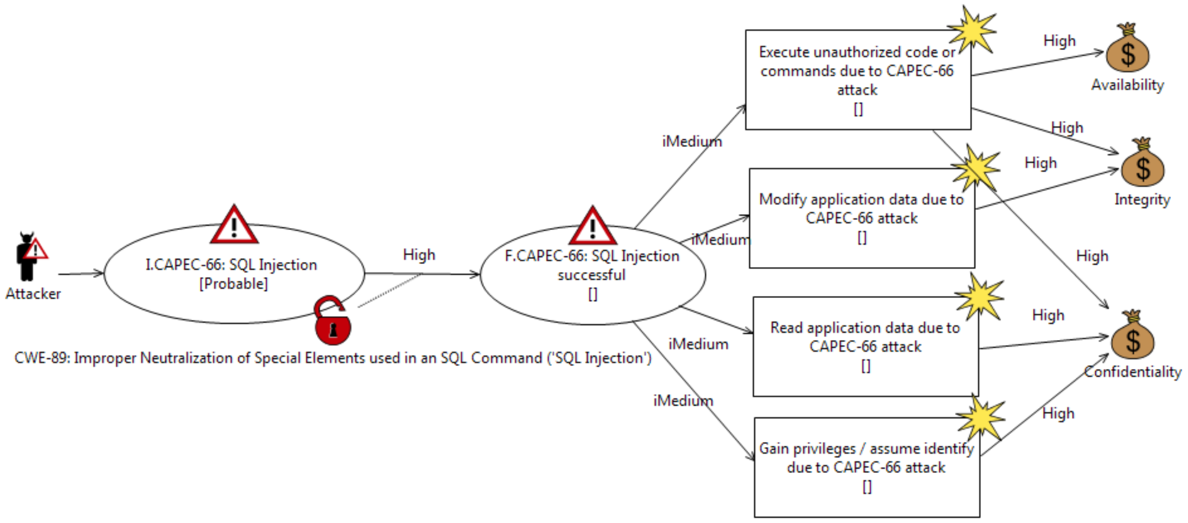

Attacker
I.CAPEC-66: SQL Injection [Probable]
High
F.CAPEC-66: SQL Injection successful []
CWE-89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')
iMedium
iMedium
iMedium
iMedium
Execute unauthorized code or commands due to CAPEC-66 attack []
Modify application data due to CAPEC-66 attack []
Read application data due to CAPEC-66 attack []
Gain privileges / assume identify due to CAPEC-66 attack []
High
High
High
High
High
High
Availability
Integrity
Confidentiality

Figure 17 – CORAS risk model describing CAPEC-66

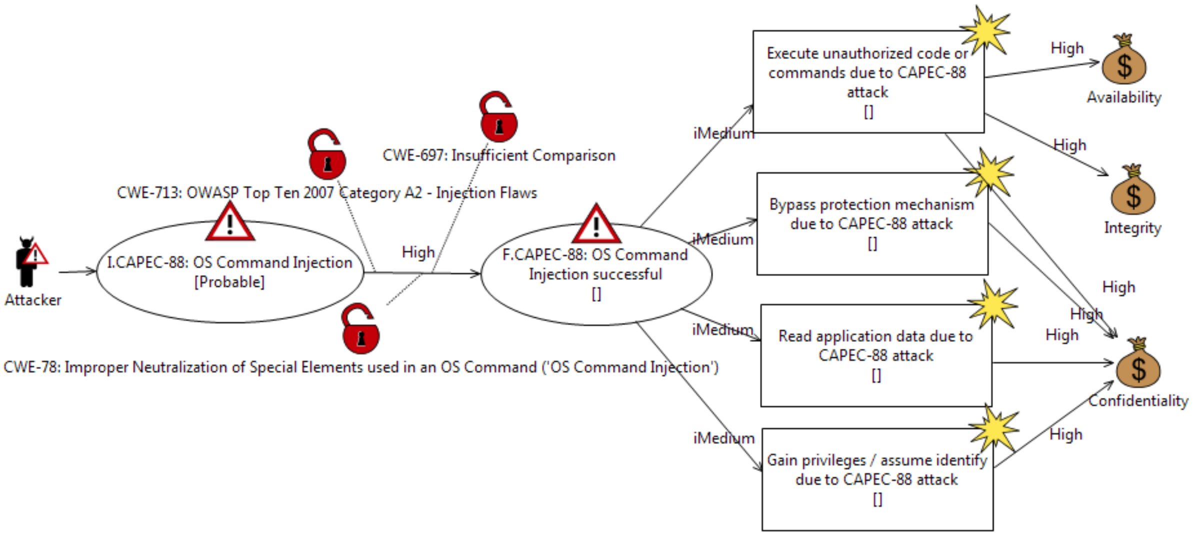

Execute unauthorized code or commands due to CAPEC-88 attack []
High
Availability
CWE-697: Insufficient Comparison
iMedium
High
Integrity
CWE-713: OWASP Top Ten 2007 Category A2 - Injection Flaws
Bypass protection mechanism due to CAPEC-88 attack []
iMedium
Attacker
I.CAPEC-88: OS Command Injection [Probable]
High
F.CAPEC-88: OS Command Injection successful []
High
High
High
iMedium
Read application data due to CAPEC-88 attack []
Confidentiality
CWE-78: Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')
iMedium
High
Gain privileges / assume identify due to CAPEC-88 attack []

Figure 18 – CORAS risk model describing CAPEC-88

3.2.3.4 Refine Generic CORAS Risk Model

The purpose of this activity is to make the generic risk model generated from the CAPEC patterns specific to the target of evaluation. The outcome of the activity is:

- A refined CORAS risk model.

This activity is manual, and should be performed by the risk analyst in collaboration with the Customer. A description of different ways in which the risk model can be refined is given in deliverable D4.2.2.

Figure 19 and Figure 20 show examples of refinements of the risk models of Figure 17 and Figure 18, respectively. In both figures, three new risks (shown on the right hand side of the diagrams) have been introduced. These are intended to be specific to the target of evaluation. In addition to this, the likelihood estimate related to the possibility of successful attacks has been adjusted in both diagrams. For instance, Figure 19, the likelihood estimate High has been adjusted to the value "Small ; Medium" where Small is a conditional likelihood (taken from the likelihood scale of Table 28) and Medium is a confidence estimate (taken from the confidence scale of Table 30).

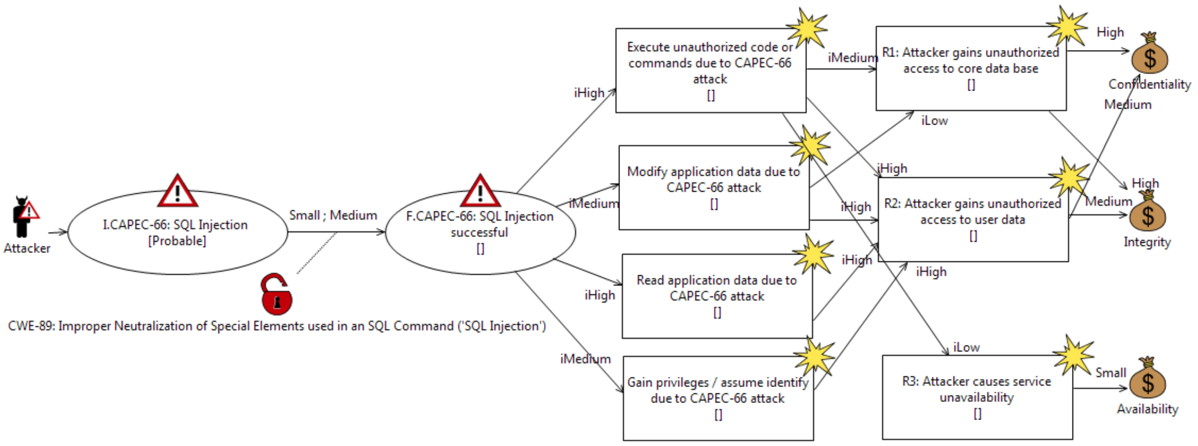

Execute unauthorized code or commands due to CAPEC-66 attack []
iMedium
R1: Attacker gains unauthorized access to core data base []
High
Confidentiality
Medium
iHigh
iLow
Modify application data due to CAPEC-66 attack []
iHigh
High
Attacker
I.CAPEC-66: SQL Injection [Probable]
Small ; Medium
F.CAPEC-66: SQL Injection successful []
iMedium
iHigh
R2: Attacker gains unauthorized access to user data []
Medium
Integrity
Read application data due to CAPEC-66 attack []
iHigh
iHigh
iHigh
CWE-89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')
iMedium
iLow
Gain privileges / assume identify due to CAPEC-66 attack []
R3: Attacker causes service unavailability []
Small
Availability

Figure 19 – Refined CORAS risk model (based on CAPEC-66)

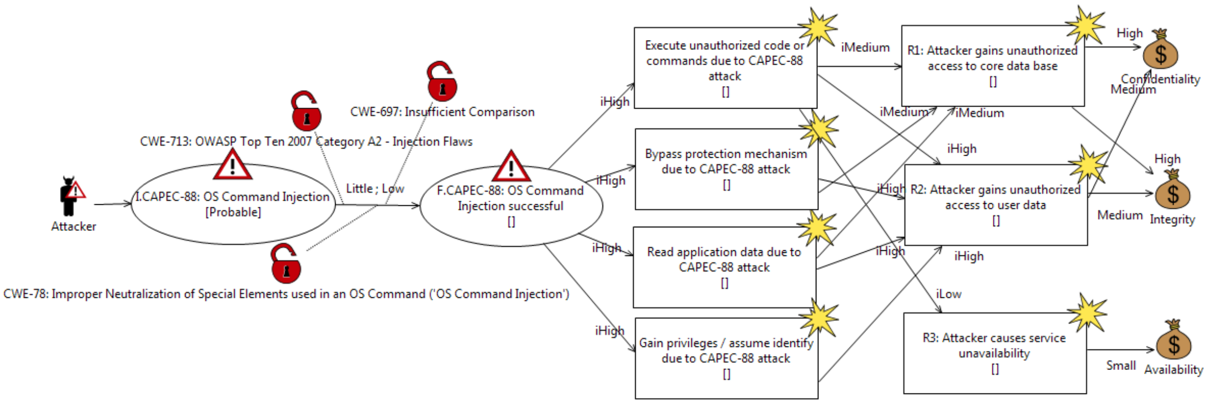

CWE-697: Insufficient Comparison
CWE-713: OWASP Top Ten 2007 Category A2 - Injection Flaws
Attacker
I.CAPEC-88: OS Command Injection [Probable]
Little ; Low
F.CAPEC-88: OS Command Injection successful []
CWE-78: Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')
iHigh
iHigh
iHigh
iHigh
Execute unauthorized code or commands due to CAPEC-88 attack []
Bypass protection mechanism due to CAPEC-88 attack []
Read application data due to CAPEC-88 attack []
Gain privileges / assume identify due to CAPEC-88 attack []
iMedium
iMedium
iMedium
iHigh
iHigh
iHigh
iHigh
iLow
R1: Attacker gains unauthorized access to core data base []
R2: Attacker gains unauthorized access to user data []
R3: Attacker causes service unavailability []
High
Medium
Confidentiality
High
Medium
Integrity
Small
Availability

Figure 20 – Refined CORAS risk model (based on CAPEC-88)

3.2.3.5 Test Procedure Identification

The purpose of this step is to identify which elements of the risk model that can be tested and to estimate the effort/time it will require to implement and execute these tests. The outcome of the step is:

- A CORAS risk model annotated with effort estimates.

At this point in the process, it can be useful to generate a risk matrix showing the risk values of the risks that are described by the risk model.An example of this is given in Figure 21, which shows the risk values of the risks R1 – R3 depicted in Figure 19 and Figure 20. The likelihood intervals of the risks have been automatically calculated based on the likelihood estimates of the risk model. In Figure 21, the left and right hand side of the white boxes representing risks indicate the minimum and maximum likelihood values of the risks, respectively. In this way, the confidence, expressed as the width of a likelihood interval is visualized. Furthermore, we see that some of the risks span across both green and red risk values, meaning that we do not know whether the risks are acceptable or not due to the uncertainty in the likelihood value. This uncertainty is an indication that new information/knowledge is needed in order to obtain a more accurate likelihood estimate. Testing is one of the methods that can be used for this purpose.

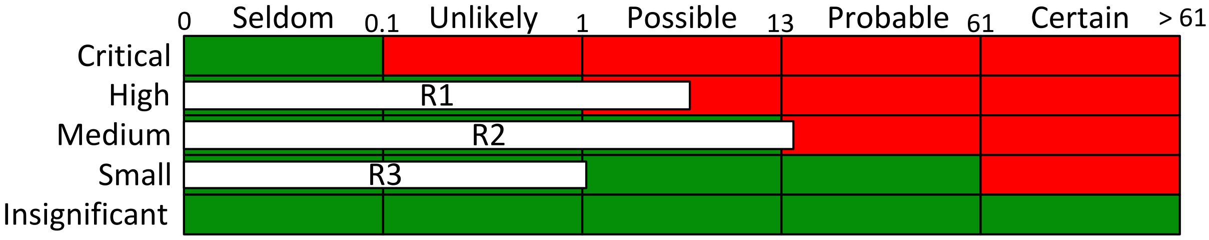


Figure 21 – Risk matrix showing the risk values of risks R1 - R3

In our approach, the elements of the CORAS risk model that we are interested in testing are the transitions/arrows. These transitions specify that one event may lead to another event with a conditional likelihood. Testing such a relation corresponds to checking whether this is correct or not.

For each transition of the risk model, the risk analyst together with the security tester must ask whether it is possible to test it given the scope of the analysis and knowledge/tools available to the testing team. If yes, then the effort required to test the transition must be estimated and documented in the risk model by annotating it with effort estimates.

Assume in this example, that only the two transitions going from attack initiation to attack success in Figure 19 and Figure 20 can be tested, and that it is estimated that 2 days will be needed to test each of these. In order to indicate this, the risk analyst annotated the two transitions with the number 2. Transitions that are not annotated are assumed to be out of scope for testing.

3.2.3.6 Test Procedure Selection/prioritization

The purpose of this step is to select and prioritize the tests identified on the risk model in the previous activity. The outcome is:

- A list of selected and prioritized test procedures.

After having indicated which transition of the risk model that should be tested, we can automatically calculate a sensitivity score for each transition as described in deliverables RASEN D4.2.2 and D4.2.1 (note we use the term priority instead of sensitivity in those deliverables). In addition, we can automatically translate each transition into English text which can be seen as a high level test procedure.

In Table 35, we show the test procedures corresponding to the two transitions of the risk models of Figure 19 and Figure 20 that were identified in Section 3.2.3.5. These have been prioritized in order of sensitivity score (which has been automatically calculated). If 4 days or more are available for testing, then both test procedures can be tested. If there are less than 4 days available, we would select the test procedure with priority 1.

Priority	Description	Sensitivity	Effort
1	Check that OS Command Injection leads to OS Command Injection successful with conditional likelihood [0.0, 0.1001], due to vulnerabilities Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection'), Insufficient Comparison and OWASP Top Ten 2007 Category A2 - Injection Flaws	9.688E-4	2 days
2	Check that SQL Injection leads to SQL Injection successful with conditional likelihood [0.0, 0.011], due to vulnerability Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection').	5.927E-6	2 days

Table 35 – List of prioritized test procedures

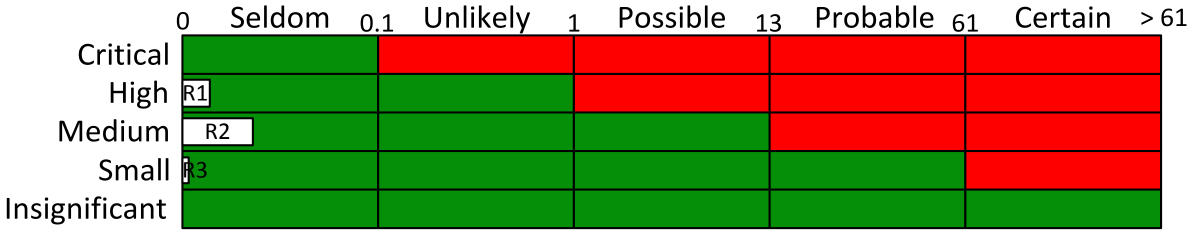
3.2.3.7 Risk Validation and Evaluation

The purpose of this activity is to update the risk model based on the results of the security testing and to evaluate the risk level of the identified risks. The outcome this activity is

- An updated risk model and a risk matrix showing the risk values of the identified risks.

As described in the RASEN deliverable D3.2.2, after the testing has been completed, the test results are translated into security measurements. These measurements are then aggregated into risk assessment measurements which are used to update the likelihood estimates of the risk model.

One possible result of doing the testing is that the confidence in the correctness of the likelihood estimates of the risk model is increased in the sense that the likelihood can be estimated with greater accuracy. For instance, the two conditional likelihood values of the transitions used as the basis for generating the test procedures in Table 35 were "Little ; Low" (i.e. Little likelihood and Low confidence) and "Small; Medium". After testing is completed however, it may be the case the confidence in the correctness of the likelihood estimates have increased, resulting in the likelihood estimates of the transitions to be updated to"Little ; High" and "Small; High”. This adjustment has a big impact on the likelihood of the risks. If we recalculate the risk likelihoods after the adjustment, we obtain the risk matrix shown in Figure 22.


0
0.1
1
13
61
> 61

	Seldom	Unlikely	Possible	Probable	Certain
Critical	🟩	🟥	🟥	🟥	🟥
High	🟩 R1	🟩	🟥	🟥	🟥
Medium	🟩 R2	🟩	🟩	🟥	🟥
Small	🟩 R3	🟩	🟩	🟩	🟥
Insignificant	🟩	🟩	🟩	🟩	🟩

Figure 22 – Updated risk matrix after testing

3.2.3.8 Risk Treatment

The purpose of this activity is to identify treatments for the risks that are not acceptable (if any). The outcome of the activity is:

- A set of CORAS treatment diagrams.

The risk treatment activity is not particular to the RASEN method, and it can therefore be performed according the CORAS risk assessment method w.r.t risk treatment.

3.3 RASEN Methodology for Compliance Risk Assessment

In this section, we describe the RASEN methodology for compliance risk assessment. Before proceeding to describe the RASEN method, first, we discuss how to structure the identification of compliance risks. This is because the structured identification of compliance risks constitutes an integral part of the RASEN methodology. This is followed by an overview of the steps in the RASEN methodology and a more detailed process description of the methodology. This section ends by illustrating the method through a concrete example.

3.3.1 Structured Compliance Risk Identification

The identification of legal and compliance risks involves too much analytical activity, which can sometimes be frustrating. Consequently, the main goal of this proposed approach is to reduce the analytical activity involved in identifying legal and compliance risks by structuring the identification of compliance risks. This is achieved by providing the risk analyst with a starting point for risk analysis by schematically translating the compliance requirements and facts in the business environment into threat scenarios and unwanted incidents. These artifacts can be used for further analysis during brainstorming sessions and meetings with the relevant stakeholders. Doing so also reduces the time spent conducting legal and compliance risk analysis.

According to ISO31000 [11], the key aspects of risk identification are the sources of risk and events. In the context of compliance, risk identification involves examining how a compliance requirement—an obligation or prohibition—can lead to risk. Among the available risk identification techniques, structured brainstorming is considered relatively well suited for the compliance context because it involves an interdisciplinary group of experts [12]. In brainstorming activities, different stakeholders contribute their knowledge and experience to identifying risk events and assessing these events under the law, guided by structured questions. In the context of compliance, risks can be identified through two approaches: *law centered* and *facts centered* [12]. At the core of the law-centered approach is the compliance norm or requirement. In this approach, the brainstorming activity focuses on identifying through guiding questions what triggers this norm. When applied to voluntary compliance requirements, this approach can be described as *requirement centered*. In contrast, the *facts-centered* approach focuses on identifying facts and assessing their legal consequences. The facts-centered approach also reuses already identified risks from other areas, such as technical risk assessments, and assesses their legal consequences. The law and factual assessments need to be understood together in order to get a full picture of the legal or compliance risk. Below, we demonstrate how the identification of compliance risks can be structured in using the requirement-centered and facts-centered approaches.

3.3.1.1 Requirements-Centered Approach

In the requirements-centered approach, the goal is to identify risks by focusing on the compliance requirement, such as an obligation or prohibition. Every compliance norm consists of an antecedent (if A) and a consequent (then B) [13]. The antecedent is the circumstances necessary for the norm to apply. The consequent is the (legal) effects of the application of the norm. The (legal) effect of a particular norm depends on the factual circumstances: an actor, an activity performed by that actor, and the actor's role while performing that activity. This implies that the 'activity' performed by an 'actor' in a certain 'role' should be in-line with the 'activity' prescribed by the compliance requirement. Figure 23 shows the conceptual model for a compliance norm and schematically identify the compliance threat and unwanted incidents.

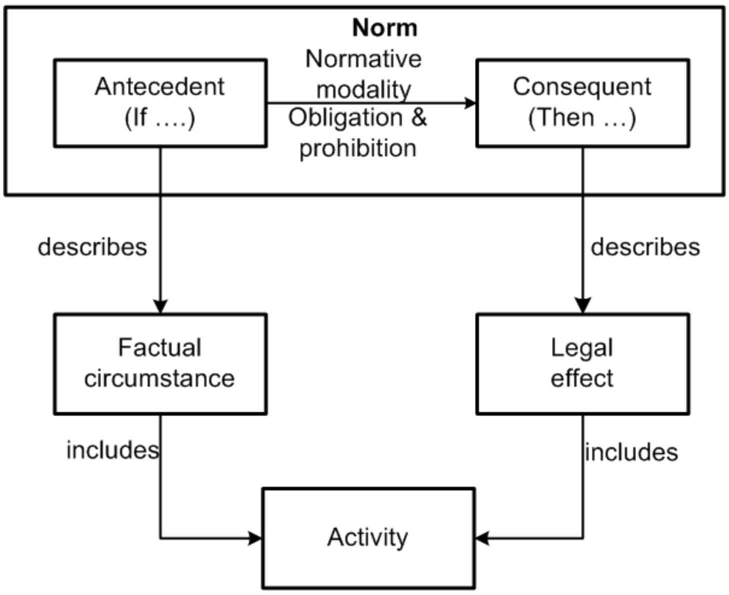


Figure 23 – Describing compliance norm

These concepts of the compliance norm can be mapped to the concepts of risk analysis used in the CORAS approach, as shown in **Figure 24**.

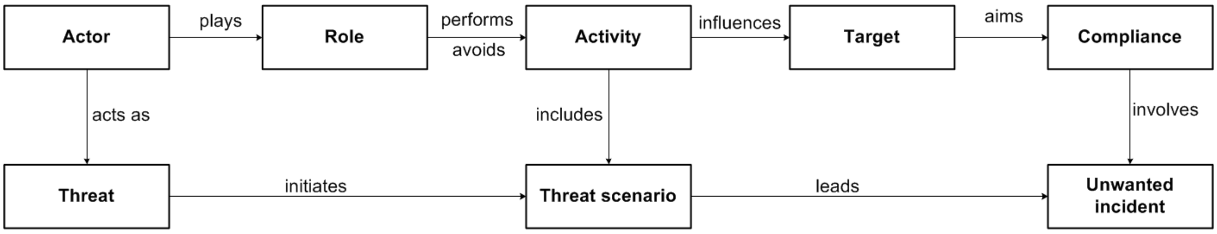


Figure 24 – Mapping compliance norm to CORAS notions.

Once the compliance requirement is identified, the determination of possible non-compliance risks can begin by identifying the notions of the actor, the normative modality, and the activity the actor is obliged to or prohibited from performing. The RASEN template can serve as a starting point for structuring a compliance requirement into these notions.

Legal source	E.g., Norwegian Personal Data Regulation Section 2-4
Normative modality	E.g., Obligations
Actor	E.g., Banks
Role	E.g., Data controller
Activity	E.g., The data controller shall carry out a risk assessment in order to determine the probability and consequences of breaches of security.
Target	E.g., ICT system
Threat scenario	E.g., Failure to carry out a risk assessment in order to determine the probability and consequences of breaches of security.
Unwanted incident	E.g., Non-compliance with Norwegian Personal Data Regulation Section 2-4

Table 36 – Template for structuring compliance requirement

While identifying compliance risks in the requirement-centered approach, the focus is on the activity of the actor. If the activity is an obligation, then the threat scenario is failure to perform that specific activity. If the activity is a prohibition, the threat scenario is the possible performance of that specific activity. In addition, the threat scenario must lead to non-compliance with the specific compliance norm in order to become an unwanted incident, causing deviation from the objective or asset. Not all failures to perform an obligation or performances of a prohibited activity lead to non-compliance. For example, an actor might be generally prohibited from doing a certain activity but may do it under certain circumstances. Considering such exceptions, the unwanted incident can be schematically translated from the legal source as non-compliance with the specific compliance source, as shown in **Figure 25**.

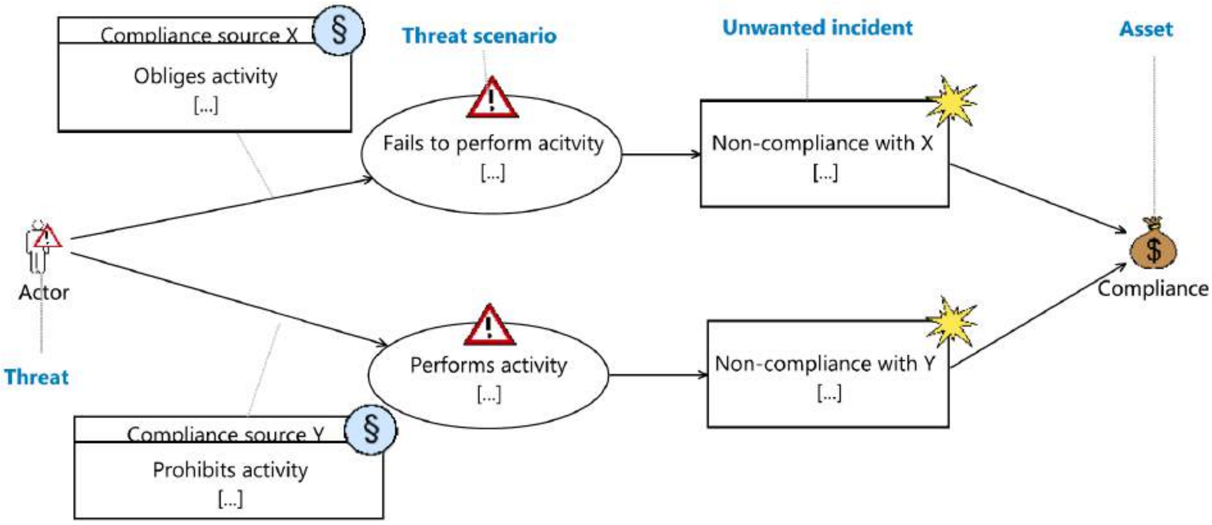


Figure 25 – Modeling compliance threat in CORAS

3.3.1.2 Facts-Centered Approach

The facts-centered approach identifies risks by focusing on the facts or other risks and assesses their legal consequences. This approach is especially relevant in the context of information security because of the capability to assess the legal consequences of security risks. The alignment of legal and security risk analysis enables accounting for legal requirements in risk-decision making. This ensures that a risk considered acceptable by the organization's criteria is not prohibited by law and is acceptable from that organization's legal position. Generally, the facts-centered approach aids in assessing the compliance implications of a planned change, project, or task.

In the facts-centered approach, the stakeholder is aware of certain facts or risks and wishes to consider their compliance implications. Once the facts are known, all compliance requirements which might be triggered are identified through guiding questions. Next, the risks of non-compliance are identified in the same manner as in the requirement-centered approach by focusing on the notion of activity presented in the compliance requirement.

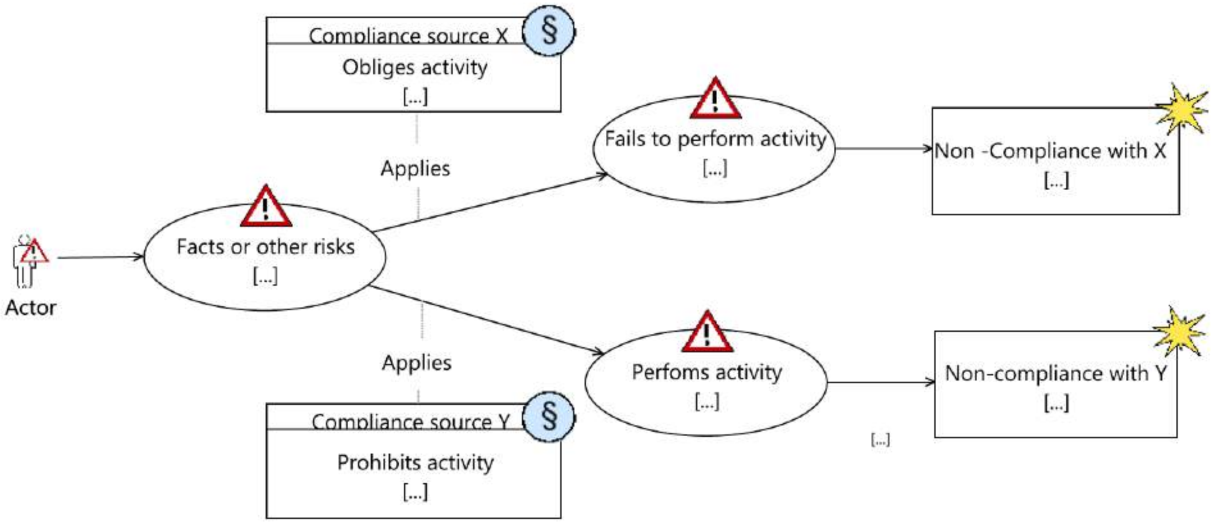


Figure 26 – Facts-centered CORAS threat diagram

In using the facts-centered approach, the template in Table 37 can be used to structure the facts, identify the compliance norm triggered by the facts, and schematically translate the activity in the compliance norm to threat scenarios in the same manner as in the requirement-centered approach. The RASEN template can be utilized to structure the facts, identify the compliance norm triggered by the facts, and schematically translate the activity in the compliance norm to threat scenarios in the same manner as in the requirement-centered approach. A difference of the facts-centered approach from the requirement-centered approach is that a certain fact could trigger the application of many compliance requirements from the same or different sources. This possibility, however, does not create a significant deviation in the RASEN methodology as described above.

Facts	E.g., Cloud provider located outside Norway
Legal source	E.g., Rundskriv 14/2010, para 10
Modality	E.g., Prohibitions
Actor	E.g., Bank
Role	E.g., Owner of ICT systems
Activity	E.g., Banks shall not outsource their critical ICT systems to high risk countries
Target	E.g., ICT systems
Threat scenario	E.g., Outsourcing critical ICT systems to high risk countries
Unwanted incident	E.g., Non-compliance with Rundskriv 14/2010, para 10

Table 37– Template for structuring facts-centred identification of risks

3.3.2 Process Overview

The RASEN methodology for compliance risk assessment is an instance of or extension of the generic RASEN method described in Figure 3.

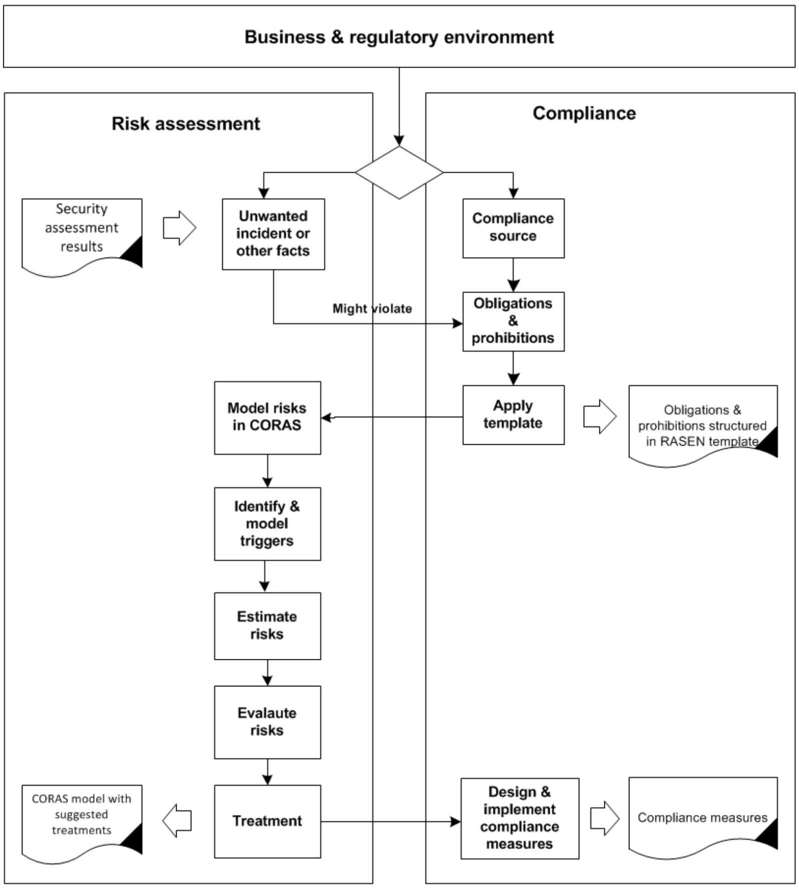

Business & regulatory environment
Risk assessment
Compliance
Security assessment results
Unwanted incident or other facts
Compliance source
Might violate
Obligations & prohibitions
Model risks in CORAS
Apply template
Obligations & prohibitions structured in RASEN template
Identify & model triggers
Estimate risks
Evalaute risks
CORAS model with suggested treatments
Treatment
Design & implement compliance measures
Compliance measures

Figure 27 – Specific RASEN methodology for compliance risk assessment

The main steps of the process of compliance risk assessment are illustrated in Figure 3. The main difference is that some steps are further instantiated into activities as shown below in steps 2, 3 and 4.

Overview of steps:

- Step 1: Understanding the business and regulatory environment
  - Input: Decision to ensure compliance
  - Output: Tables defining consequence and likelihood scales, risk evaluation criteria, parties relevant for the analysis (e.g. relevant regulatory authorities, contracting parties)
- Step 2: Requirement identification
  - Activity: Identify relevant compliance sources (What to comply with?)
    - Input: Business objective and regulatory environment
    - Output: Selected relevant compliance documents, and/or process under analysis,
- Step 3: Identify compliance issues
  - Activity 1: Make a list of compliance requirements based on obligations/prohibitions
  - Activity 2: Structure compliance requirements in RASEN template
    - Input: Relevant compliance documents
    - Output: List of obligations and prohibitions structured in RASEN template
- Step 4**:** Compliance risk identification

- Activity 1 : Model risks in CORAS
    - Input: List of obligations and prohibitions structured in RASEN template
    - Output: Legal CORAS risk model
- Activity 2: Identify and model triggers
    - Input: Legal CORAS risk model
    - Output: Legal CORAS risk model with triggers

- Step 5: Estimate compliance risks
    - Input: Legal CORAS risk model
    - Output: Legal CORAS risk model with likelihood and consequence estimates
- Step 6: Evaluate compliance risks
    - Input: Legal CORAS risk model with likelihood and consequence estimates
    - Output: A risk matrix with risks
- Step 7: Treatment
    - Input: A risk matrix with risks
    - Output: Legal CORAS risk model with suggested treatments
- Step 8: Implement compliance measures
    - Input: Legal CORAS risk model with suggested treatments
    - Output: Compliance measures implemented

The above process describes a compliance risk assessment which employs a *requirement-centered* identification of risk, see Section 3.3.1.1. A compliance risk assessment that employs *facts-centered* approach (see Section 3.3.1.2) for identifying compliance risks is slightly different from the above. In the latter case, step 2 would focus on identifying the relevant facts or consider other risks (e.g. security risk) as an input to the analysis. The rest of the process remains the same. Figure 27 shows the process for compliance risk assessment that employs both *requirement-centered* and *facts-centered* approach for identifying risks.

3.3.3 Process Description

In the following, we document each step of the process using the template described in Table 1.

Name	**Understanding business and regulatory environment**
Actors	Compliance Manager (CM), Risk analyst (RA), Customer (C)
Tools	Security Risk Management Tool (SRMT),
Precondition	Decision to ensure compliance
Postcondition	The activity must end with the following outputs: • Description of the focus of compliance analysis, assumptions and scope • Tables defining consequence and likelihood scales • Risk matrix tables defining risk evaluation criteria • Relevant parties to the analysis including regulatory authorities and contracting parties are identified

Scenario	1. The CM together with the customer describes the focus of the compliance analysis, i.e. Whether the objective is to ensure the compliance of a specific target with relevant laws or to ensure compliance with a specific legislation. If the focus of the analysis is a specific target, e.g. specific business process, information system, then the target should be described. Whether the analysis includes compliance with voluntary requirements such as industry standards. 2. CM documents the legal framework applicable to the business environment, the relevant regulatory authorities on the area and contracting parties. 3. Based on discussions with the customer, the CM and RA document: • The consequence and likelihood scales, • Risk matrix tables defining risk evaluation criteria • The parties and the target of analysis.
Data exchanged	**In (from stakeholder):** Decision to ensure compliance **Out:** Tables defining consequence and likelihood scales, risk evaluation criteria, parties relevant for the analysis (e.g. relevant regulatory authorities, contracting parties)

Table 38 – Activity: Understanding business and regulatory environment

Name	**Requirement identification**
Actors	Compliance Manager (CM), Risk analyst (RA), Customer (C)
Tools	Security Risk Management Tool (SRMT)
Precondition	The precondition for this activity is the same as the postcondition of the activity “Understanding business and regulatory environment”
Postcondition	The activity must end with the following output: • Relevant sources of compliance are identified.
Scenario	1. CM and customer identify and document the relevant compliance requirements applicable to the business or to a specific target as defined in step 1. The legal sources of relevance might stem from contracts, legal regulations, court decisions and administrative decisions. In addition compliance sources could be voluntary by nature, such as industry and organizational standards and codes, principles of good governance and accepted community and ethical standards. Some relevant guiding questions for this task include: Which law(s) or requirements does the organization want to ensure compliance with? If the objective is to ensure compliance of a specific target, then the relevant question would be what laws might apply to the target at hand?
Data exchanged	**In:** Decision to ensure compliance **Out:** Applicable compliance sources

Table 39 – Activity: Requirement identification

Name	**Identify compliance issues**
Actors	Compliance Manager (CM), Risk Analyst (RA), Customer (C)
Tools	Security Risk Management Tool (SRMT),
Precondition	The precondition for this activity is the same as the postcondition of the activity “Requirement identification”.
Postcondition	The activity must end with the following output: • List of obligations and prohibitions structured in RASEN template (**Table 36** and Table 37)
Scenario	1. CM identifies list of obligations and prohibitions from the relevant compliance sources. Obligations prescribe the specific actions that the organization must undertake in order to comply with the corresponding compliance requirement. Prohibitions specify the actions that the organization must not. If the relevant source specifies the consequences of failing to adhere to the obligations or prohibitions, such norms have to be identified as well. A guiding question for this task could be: What obligations and prohibitions are incumbent on the organization or the target at hand? Or In case of facts-centered approach, the question would be: which obligations and prohibitions might be infringed by the fact or risk at hand? 2. CM structures the obligations and prohibitions in the RASEN template describing the notions of the ‘actor’, the ‘normative modality’, and the ‘activity’ the actor is obliged to or prohibited from performing by the compliance requirement. See more in Section 3.3.1. 3. CM and RA describe non-compliance. This is done by schematically identifying compliance threats using the documentation in the RASEN template. The compliance threat scenario is schematically translated from the notion of ‘activity’ which is obliged or prohibited. If the activity is an obligation, then the threat scenario is failure to perform that specific activity. If the activity is a prohibition, the threat scenario is the possible performance of that specific activity. The unwanted incident is identified by adding non-compliance to the compliance norm at hand. See more in Section 3.3.1 4. The CM documents the results using the RASEN template.
Data exchanged	**In:** Applicable compliance sources &/or process under analysis **Out:** List of obligations and prohibitions structured into RASEN template and schematically identified compliance threats and unwanted incidents

Table 40 – Activity: Identify compliance issue

Name	**Compliance risk identification**
Actors	Compliance Manager (CM), Risk Analyst (RA), Customer (C)
Tools	Risk Assessment Tool (SRAT),

Precondition	The activity starts with the following input: - Obligations and prohibitions structured in RASEN template (Table 36 and Table 37)
Postcondition	The activity must end with the following output: - A set of compliance CORAS risk models
Scenario	1. The RA models the threat scenarios and unwanted incidents in CORAS. Essentially this activity is semi-automatic in the sense that the RASEN template provides all the components to be modeled in CORAS. 2. RA guides the CM and the customer through guiding questions to identify vulnerabilities or triggers for the threat scenarios. Triggers of the threat can be identified by asking a relevant guiding question, such as: What facts could trigger the threat at hand? Or What vulnerabilities are there which could be exploited by the threat? 3. The RA documents the triggers in the CORAS risk diagram as initiating threat scenarios or vulnerabilities depending on their nature.
Data exchanged	**In:** Obligations and prohibitions structured in RASEN template, schematically identified compliance threats & unwanted incidents. **Out:** CORAS compliance threat diagram with triggers

Table 41 – Activity: Legal and compliance risk identification

Name	**Compliance risk estimation**
Actors	Compliance Manager (CM), Risk analyst (RA), Customer (C)
Tools	Security Risk Assessment Tool (SRAT),
Precondition	The activity starts with the following input: - CORAS compliance threat diagram with triggers - Tables defining consequence and likelihood scales
Postcondition	The activity must end with the following output: - CORAS compliance threat diagrams with likelihood and consequence values
Scenario	1. The CM, RA and customer walk through the risk model and estimate the level of non-compliance risk by considering its negative consequences and likelihood according to criteria established in advance by the stakeholders. In estimating the consequences, account should be taken, if any, to the relevant compliance requirements that specify the consequences of failing to adhere to the obligations or prohibitions. This is followed by determining the risk level as, for example, ‘high risk’ ‘low risk’ depending the consequences and likelihood of occurrence. 2. The RA documents the results using CORAS threat diagrams.
Data exchanged	**In:** CORAS compliance risk model with identified threat scenarios, triggers and unwanted incidents **Out:** CORAS compliance threat diagrams with likelihood and consequence values

Table 42 – Activity: Compliance risk estimation

Name	**Compliance risk evaluation**
Actors	Compliance Manager (CM), Risk analyst (RA), Customer (C)
Tools	Security Risk Assessment Tool (SRAT),
Precondition	The precondition for this activity is the same as the postcondition of the activity “Compliance risk estimation”.
Postcondition	The activity must end with the following output: • A set of risk matrix with all identified compliance risks.
Scenario	1. CM, RA and customer make decisions on whether to treat or accept risks. Evaluation and prioritization are decided based on the risk estimation conducted in step 6 and the risk evaluation criteria established in step 1.
Data exchanged	**In:** CORAS compliance threat diagrams with likelihood and consequence values **Out:** Compliance risk models with risk matrix

Table 43 – Activity: Compliance risk evaluation

Name	**Treatment**
Actors	Compliance Manager (CM), Risk analyst (RA), Customer (C)
Tools	Risk Assessment Tool (SRAT),
Precondition	The precondition for this activity is the same as the postcondition of the activity “Compliance risk evaluation”.
Postcondition	The activity must end with the following output: • CORAS risk model with suggested treatments
Scenario	1. CM, RA and customer identify compliance measures that can address the prioritized compliance risks and the most suitable compliance measures are selected based on an assessment of the costs and benefits of each measure. 2. The RA documents the results in CORAS treatment diagrams.
Data exchanged	**In:** Compliance risk models with risk matrix and prioritized triggers **Out:** CORAS risk model with suggested treatments

Table 44 – Activity: Treatment

Name	**Design and implement compliance measures**
Actors	Compliance Manager (CM), Risk analyst (RA), Customer (C)
Tools	Risk Management Tool (SRMT),

Precondition	The precondition for this activity is the same as the postcondition of the activity “Treatment”.
Postcondition	The activity must end with the following output: • Compliance measures implemented
Scenario	1. The customer implements control measures to manage the identified compliance obligations and prohibitions and achieve desired behaviors. This includes, among other things, integrating compliance obligations into existing business practices and procedures including computer systems, forms, reporting systems and contracts. 2. The CM documents the results.
Data exchanged	**In:** CORAS risk model with suggested treatments **Out:** Compliance measures are implemented and documented

Table 45 – Activity: Design and implement compliance measure

3.3.4 Exemplification of the Method

In the following, we give an illustration of the RASEN methodology based on a real business case. The example is based on the Evry case study and a consultation made with the Norwegian Financial Supervisory Authority (Finanstilsynet). The study examines the compliance risks of Evry’s netbank system. However, due to the confidentiality of the information, we do not show any actual incidents or real risk estimates for the Evry case. In addition, having regard to the importance of cloud services highlighted in the first year project review, we have added a cloud scenario to the analysis. This would not only align the methodology with cloud computing paradigms as suggested in the review, but also it would significantly enhance the relevance of the legal and compliance risk assessment. To demonstrate the two approaches of risk identification discussed in 3.3.1, we divide the assessment into two, one assessment that employs the *requirement-centered* identification of risks and another based on the *facts-centered* approach.

3.3.4.1 Exemplification of the Requirement-Centered Compliance Risk Assessment

Step 1: Understanding the business and regulatory environment

The business environment for the use case constitutes a netbank software system which is used by about one million users all throughout Norway to manage electronic payment transactions. The security of this system is crucial as security breaches could potentially have a negative financial impact on the customers (both private persons and organizations) as well as damaging the reputation of the Customer. Also, a security breach can potentially cause the net bank to be become unavailable to the users. The customer’s netbank is regarded as an important part of Norway’s infrastructure. As such it’s put under strict demands by the financial supervisory authority of Norway and the Data Inspectorate of Norway through laws and regulations to ensure that availability of the system and to protect sensitive information of the users. Figure 28 below describes the parties involved in the netbank system and their basis for their relationship.

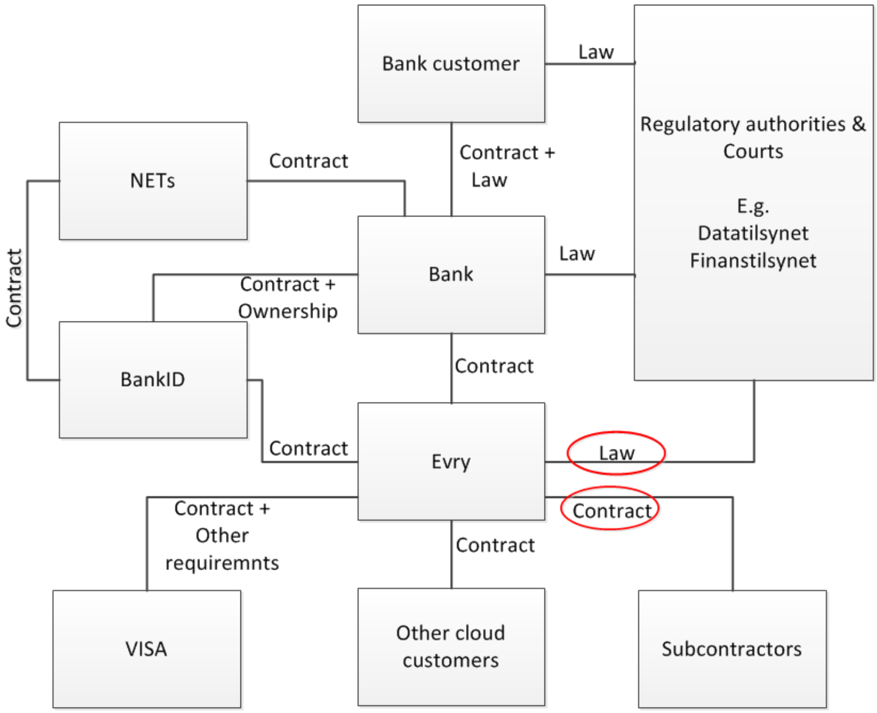


Figure 28 – Actors and legal basis for their relationships

Scope of analysis: Ensuring compliance of the netbank system with the applicable compliance requirements. In particular, the analysis will focus on the legal requirements that tie the Customer with the regulatory authorities together with the contractual aspects with the cloud providers (See the focus points in Figure above). In achieving this objective, the Customer together with CM and RA document:

- *Likelihood & consequence scale*
- *Risk matrix tables defining risk evaluation criteria*

Cloud scenario: The Customer is also considering the potential use of cloud services and wants the analysis to take account of the associated compliance risks where some of the components of the netbank are moved to the cloud. The figure below depicts the current netbank architecture and the potential target for the cloud scenario.

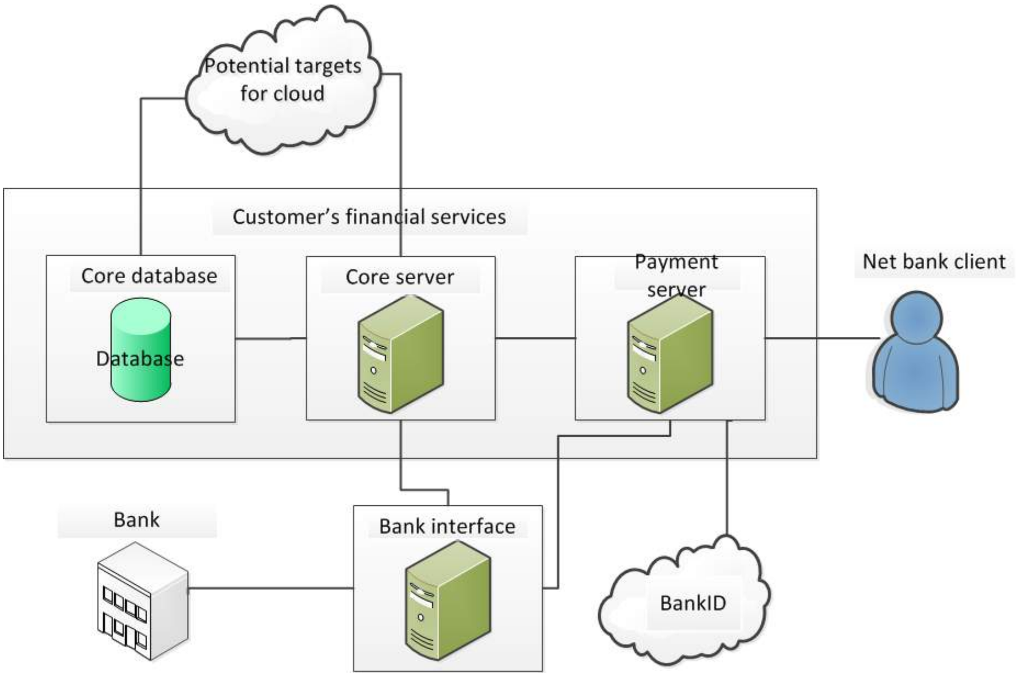


Figure 29 – Customer's current netbank architecture and potential cloud scenario

Step 2: Requirement identification

Once the objective and the risk evaluation criteria are defined, the next step is to identify relevant legal sources for compliance. This can vary depending on whether the objective is to ensure the compliance of a specific target with relevant laws or to ensure compliance with specific legislation. In the case at hand, the goal is to ensure the compliance of the netbank system with the relevant laws and contractual obligations. The following compliance source is relevant to the netbank system and is extracted from Lovdata, a database containing legal information:

- The Norwegian ICT Regulation (hereinafter NORICTR)[2]

Step 3: Identify compliance issues

Once the compliance requirements are identified, the determination of possible non-compliance risks begins by making a list of obligations and prohibitions. This is followed by structuring of obligations and prohibitions using the RASEN template.

Activity 1: Make a list of obligations and prohibitions

As part of this task, the following provision is identified from the above compliance source as imposing an obligation on the Customer in relation to the netbank.

List of prohibitions and obligations from NORICTR

- Section 13 of NORICTR: Documentation

Activity 2: Structure compliance requirements into RASEN template

Section 13 of the NORICTR can be structured as follows:

Legal source	NORICTR Section 13
Modality	Obligation
Actor	Bank

[2] Forskrift om bruk av informasjons- og kommunikasjonsteknologi (IKT), FOR-2003-05-21-630.

Role	Owner of ICT systems
Activity	An assembled up-to-date overview shall exist of the organization, equipment, systems and significant factors related to ICT activities. An up-to-date documentation shall exist of each ICT system important to the institution which document the compliance with the demands in this regulation.
Target	ICT system
Threat scenario	Failure to document an up-to-date overview of the organization, equipment, systems and significant factors related to ICT activities Failure to keep an up-to-date documentation of important ICT systems and their compliance with the regulation
Unwanted incident	Non-compliance with NORICTR Section 13

Table 46 – NORICTR Section 13

Step 4: Compliance risk identification

Once the obligations and prohibitions are structured in the template, the next step is to model the risk and identify triggers.

Activity 1: Model risk in CORAS

Essentially this activity is semi-automatic in the sense that the RASEN template provides all the components to be modeled in CORAS. The results of step 3 are documented in CORAS as follows.

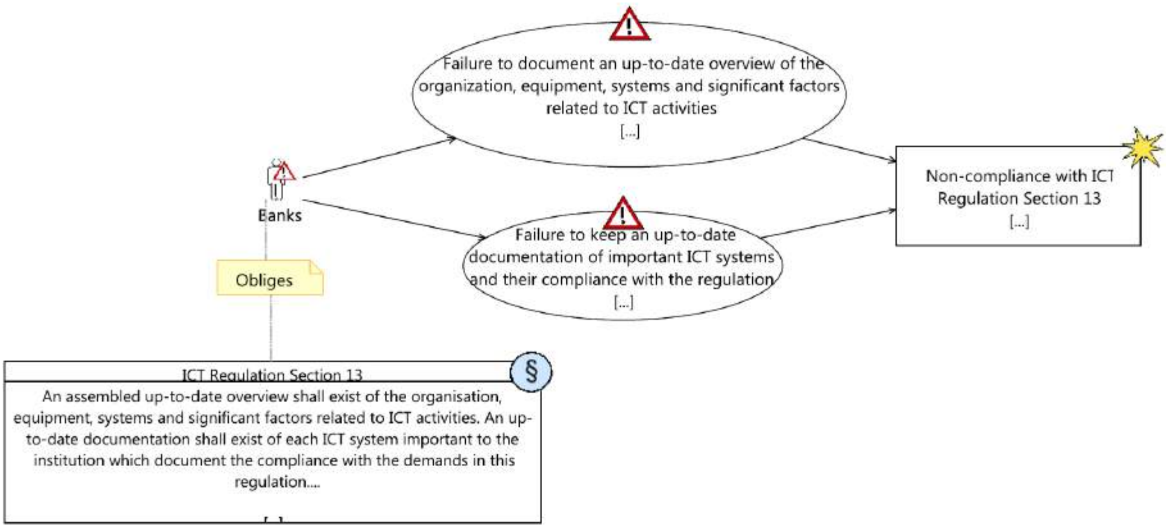

Failure to document an up-to-date overview of the organization, equipment, systems and significant factors related to ICT activities
[...]
Banks
Non-compliance with ICT Regulation Section 13
[...]
Failure to keep an up-to-date documentation of important ICT systems and their compliance with the regulation
[...]
Obliges
ICT Regulation Section 13
§
An assembled up-to-date overview shall exist of the organisation, equipment, systems and significant factors related to ICT activities. An up-to-date documentation shall exist of each ICT system important to the institution which document the compliance with the demands in this regulation....
[...]

Figure 30 – CORAS NORICTR Section 13

Activity 2: Identify triggers and model them in CORAS

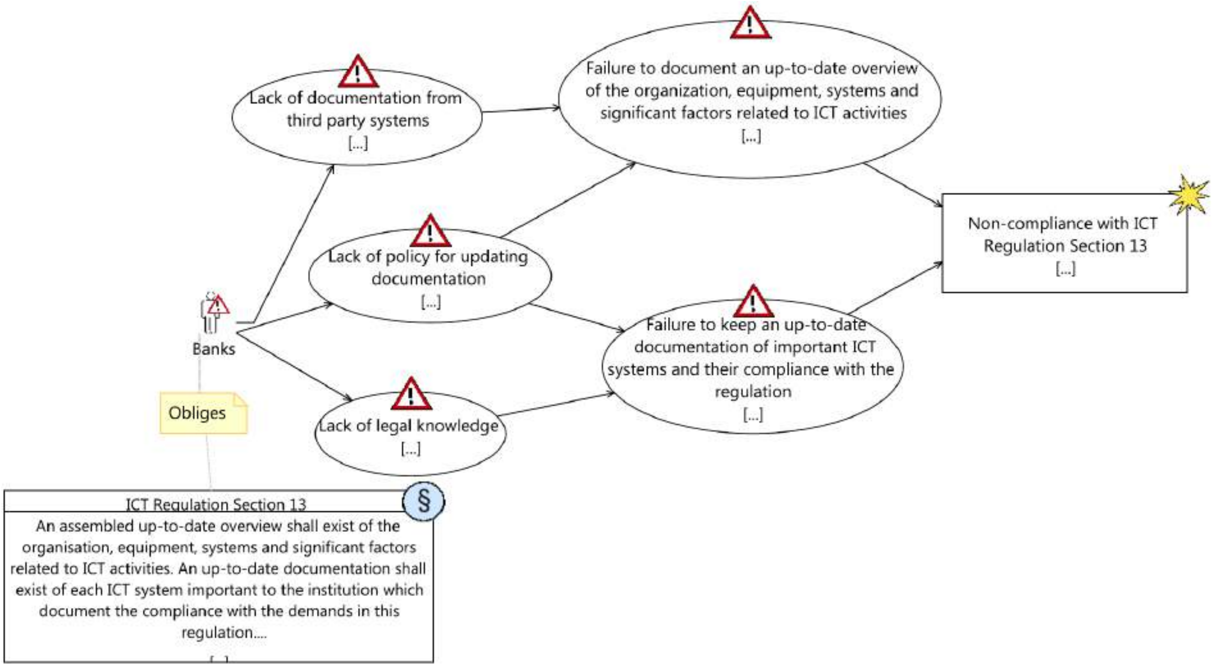
The second activity in this step is to identify the triggers for the respective identified compliance threats. Doing so is important because different factual circumstances could give rise to failure to perform the obligatory activity or the performance of the prohibited activity. In addition, the triggers are what make the risk assessment specific to the client at hand or to the target under analysis. Therefore, all possible causes and triggers of the threat can be identified by asking a relevant guiding question, such as:

- What facts could trigger the threat at hand?

Based on the discussion with the Customer, the following triggers are identified for Section 13:

- Lack of documentation from third-party systems
- Lack of policies for updating documentation
- Lack of legal knowledge

Once the triggers are identified, they are modeled in CORAS as initiating threat scenarios.



Lack of documentation from third party systems [...]
Failure to document an up-to-date overview of the organization, equipment, systems and significant factors related to ICT activities [...]
Non-compliance with ICT Regulation Section 13 [...]
Lack of policy for updating documentation [...]
Banks
Failure to keep an up-to-date documentation of important ICT systems and their compliance with the regulation [...]
Obliges
Lack of legal knowledge [...]
ICT Regulation Section 13
§
An assembled up-to-date overview shall exist of the organisation, equipment, systems and significant factors related to ICT activities. An up-to-date documentation shall exist of each ICT system important to the institution which document the compliance with the demands in this regulation....

Figure 31 – CORAS NORICTR Section 13 with triggers

Step 5: Compliance risk estimation

After completing risk identification, the risks of non-compliance should be analyzed in order to understand the level of the risks, which allows the stakeholder to decide on which issues it is most important to focus. The level of non-compliance risk is determined by considering its negative consequences and likelihood according to criteria established in advance by the stakeholders [9]. In estimating the consequences, account should be taken, if any, to the relevant compliance requirements that specify the consequences of failing to adhere to the obligations or prohibitions. In the case at hand, the analysis team agrees that the 'lack of documentation for third party systems' is 'likely' to happen as those third parties might not be aware of the existence of such obligation and is annotated in the CORAS. The team also indicates that although the customer has a documentation policy, it does not stipulate the timeline for updating documentations and the party responsible for doing so. Nonetheless, the team explains that as far as the documentation policy is well complied, the likelihood of 'lack of policy for updating documentation' leading to both threat scenarios is 'unlikely'. The team also agrees that given most people working in relation to the netback are IT expertise, lack of legal knowledge is 'likely' to exist and to lead to the undersigned threat scenario.

Next, the team notes that although it is 'unlikely' that lack of policy for updating documentation leads to 'failure to document an up-to-date overview of the organization, equipment, systems and significant factors related to ICT activities', the 'lack of documentation for third party systems' is 'likely' to do so. Similarly, the team explains that at least the lack of legal knowledge is 'likely' to cause failure of documenting compliance. Taking account of the contractual framework in place with third parties, the legal awareness of people working with the netbank as well as the significance of the documentation provision assigned by the Financial Supervisory Authority of Norway (Finanstilsynet), the group agrees that *Non-compliance with NORICTR Section 13* is ‘likely’ to happen with ‘moderate’ effect on the compliance of the organization as it can be fixed at the management level. Next, the risk level (e.g., high or low risk) is established based on the consequences and likelihood of occurrence. With the likelihood value of ‘likely’ and consequence value of ‘moderate’, non-compliance with Section 13 is ranked as a ‘medium risk’.

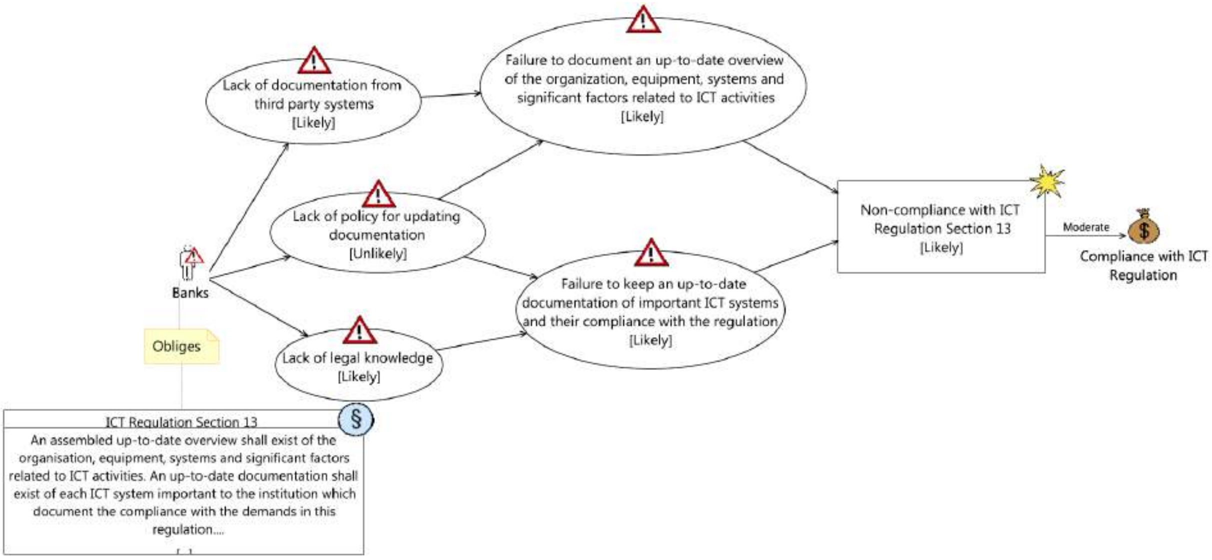

Lack of documentation from third party systems [Likely]
Failure to document an up-to-date overview of the organization, equipment, systems and significant factors related to ICT activities [Likely]
Lack of policy for updating documentation [Unlikely]
Banks
Non-compliance with ICT Regulation Section 13 [Likely]
Moderate
Compliance with ICT Regulation
Failure to keep an up-to-date documentation of important ICT systems and their compliance with the regulation [Likely]
Obliges
Lack of legal knowledge [Likely]
ICT Regulation Section 13
§
An assembled up-to-date overview shall exist of the organisation, equipment, systems and significant factors related to ICT activities. An up-to-date documentation shall exist of each ICT system important to the institution which document the compliance with the demands in this regulation....
[...]

Figure 32 – CORAS NORICTR Section 13 with likelihood and consequences

Step 6: Compliance risk evaluation

In the standard risk analysis process, this step involves making decisions on whether to treat or accept risks. However, in dealing with legal compliance or mandatory requirements, accepting risks might imply that the organization is prepared to allow violations of law or regulations. Therefore, the primary goal of such step should focus on managing and mitigating compliance risk. The evaluation is conducted based on the risk estimation conducted in step 6 and the risk evaluation criteria established in step 1. Furthermore, in order to avoid unethical business conduct, the risk-based compliance evaluation should also take consideration of ethical issues. When non-compliance would seem to imply limited risk there will often nevertheless be valid ethical reasons for not accepting non-compliance.

In the case at hand, given their likelihood of occurrence, the Customer and the team decided that the lack of documentation for third party systems and lack of legal knowledge should be prioritized over the lack of policy for updating documentation. The team agrees that the lack of documentation for third party systems has to be deal in the contract negotiation phase with third parties. The team explains that at least contracts currently under negotiation should specifically address such issue. The *Lack of legal knowledge* has to be addressed by awareness raining mechanisms within the Customer organization.

Step 7: Treatment

Once the evaluation is completed, compliance measures that can address the risks according to their risk level and the most suitable compliance measures are selected based on an assessment of the costs and benefits of each measure. In this context, the process involves analyzing available risk control options and finally implementing the selected control mechanisms. Therefore, the team identifies the following measures to mitigate the identified risks:

- Include contractual provision for third party system documentations. Contracts currently under negotiation should be prioritized

- Enhance the legal awareness of the NORICTR through training, particularly those working with the netbank.

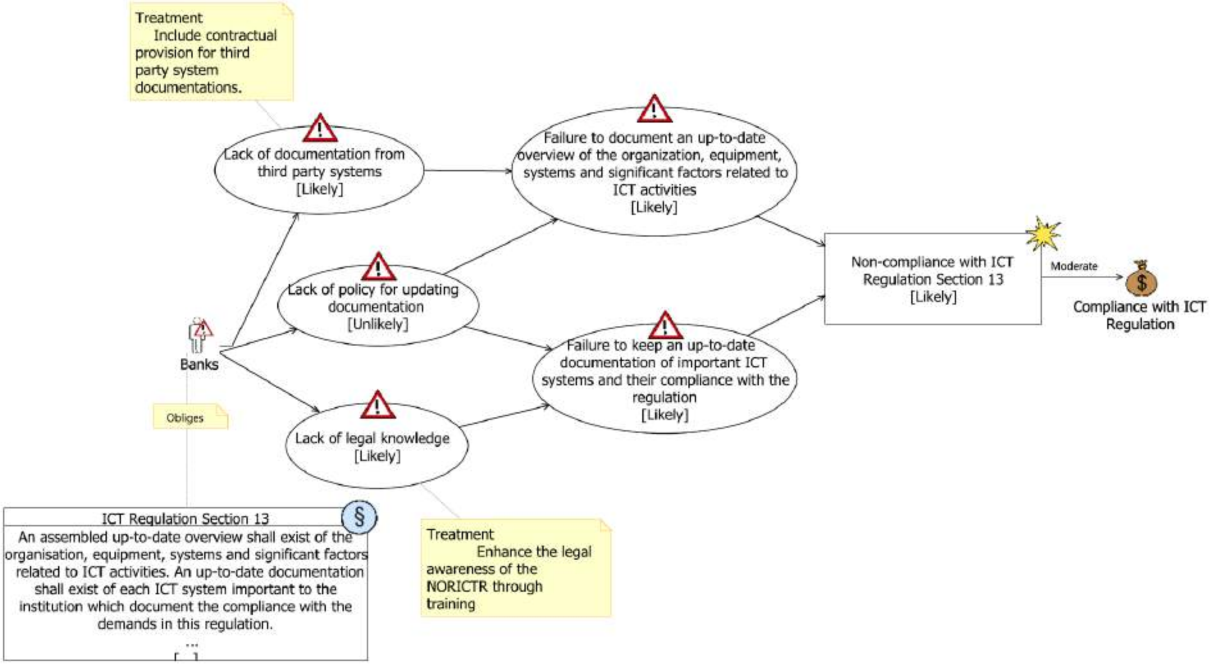

Treatment
Include contractual provision for third party system documentations.
Lack of documentation from third party systems [Likely]
Failure to document an up-to-date overview of the organization, equipment, systems and significant factors related to ICT activities [Likely]
Lack of policy for updating documentation [Unlikely]
Non-compliance with ICT Regulation Section 13 [Likely]
Moderate
Compliance with ICT Regulation
Banks
Failure to keep an up-to-date documentation of important ICT systems and their compliance with the regulation [Likely]
Obliges
Lack of legal knowledge [Likely]
ICT Regulation Section 13
An assembled up-to-date overview shall exist of the organisation, equipment, systems and significant factors related to ICT activities. An up-to-date documentation shall exist of each ICT system important to the institution which document the compliance with the demands in this regulation.
...
Treatment
Enhance the legal awareness of the NORICTR through training

Figure 33 – CORAS treatment diagram for NORICT Section 13

Step 8: Design and implement compliance measures

As a result of the suggested treatments, a contractual provision is drafted that requires third party provider to send documentation of their ICT systems every four months. This has been included in currently ongoing or under revision contracts. Dates have been set to start revisit existing contracts in that respect. A plan has been put in place on raising the legal awareness of the NORICTR, particularly for employees working with the netbank. The awareness will be conducted in two phases.

3.3.4.2 Exemplification of the Facts-Centered Compliance Risk Assessment

Step 1: Understanding the business and regulatory environment

Step 1 of the process remains the same for both the requirement-centered and facts-centered compliance risk assessment. Therefore, it can be referred from Section 3.3.4.1.

Step 2: Requirement identification

In the facts-centered approach compliance risk assessment, this step focuses on identifying relevant facts or other risks that might imply compliance risk. In the cloud context, the compliance manager, with the help of cloud experts, identifies ‘distributed server location’ as one inherent feature of cloud services, which might impact compliance. The Customer wants to know if the identified fact can imply compliance risk.

- Fact: Distributed server location

Step 3: Identify compliance issues

Activity 1: Make a list of obligations and prohibitions

As part of this task, the following provisions are identified that might be infringed by the identified fact.

List of prohibitions and obligations from Data Protection Act

- *Articles 29 of the Data Protection Act: Basic conditions on transfer of personal data to other countries*

List of prohibitions and obligations from Rundskriv 14/2014

- *Paragraph 10 of the Rundskriv 14/2010: Banks responsibility in outsourcing Banks' ICT activities*

Activity 2: Structure compliance requirements in RASEN template

The fact and the relevant requirements can then be structured in RASEN template as follows:

Facts	Distributed server location
Legal source	Data Protection Act Sec 29 Rundskriv 14/2010, para 10
Modality	Prohibitions
Actor	Bank
Role	Data controller Owner of ICT systems
Activity	Data Protection Act Sec 29: Personal data may not be transferred to countries which do not ensure an adequate level of protection of the data Rundskriv 14/2010, para 10: Banks shall not outsource their critical ICT systems to high risk countries[3]
Target	Personal data ICT systems
Threat scenario	Data Protection Act Sec 29: Personal data transferred to countries which do not ensure adequate level protection of data Rundskriv 14/2010: Outsourcing critical ICT systems to high risk countries
Unwanted incident	Data Protection Act Sec 29: Non-compliance with Data Protection Act Section 29 Rundskriv 14/2010: Non-compliance with Rundskriv, para 10

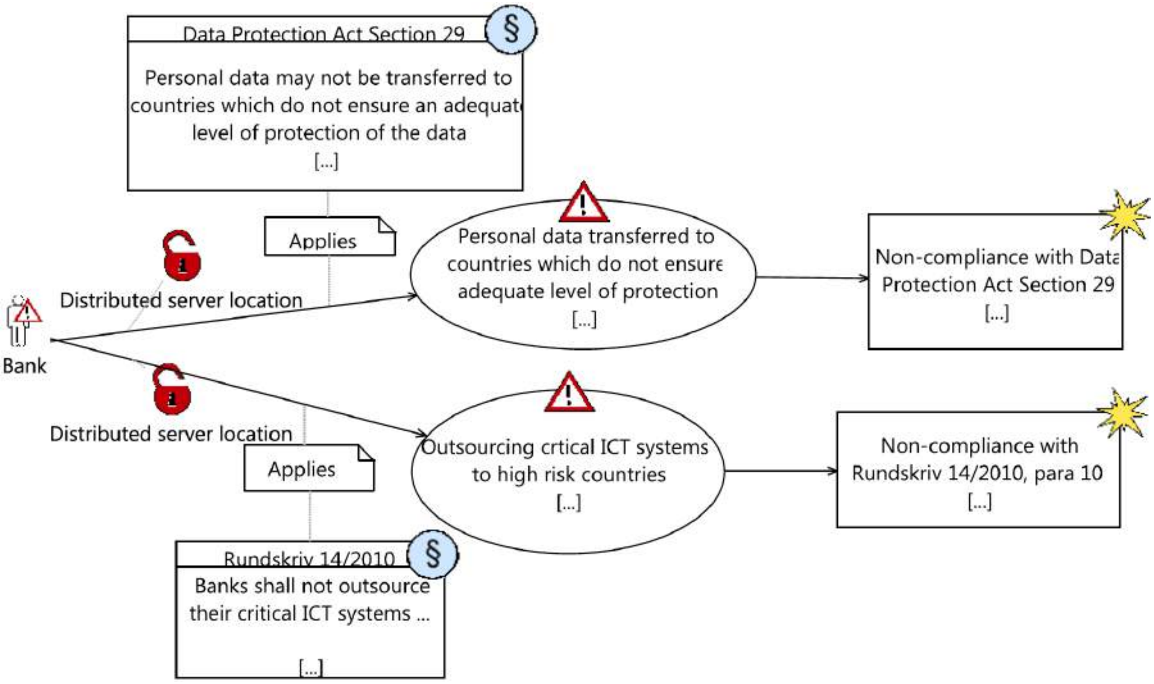
Table 47 – Distributed server location

Step 4: Compliance risk identification

Activity 1: Model risk in CORAS

The artifacts in the RASEN template are then modeled in CORAS as follows:

[3] Paragraph 10 reads as 'Finanstilsynet er av den oppfatning at ovennevnte IKT-oppgaver ikke kan utkontrakteres til landområder med høy risiko'.



Data Protection Act Section 29
§
Personal data may not be transferred to countries which do not ensure an adequate level of protection of the data
[...]
Applies
Distributed server location
Bank
Personal data transferred to countries which do not ensure adequate level of protection
[...]
Non-compliance with Data Protection Act Section 29
[...]
Distributed server location
Applies
Outsourcing crtical ICT systems to high risk countries
[...]
Non-compliance with Rundskriv 14/2010, para 10
[...]
Rundskriv 14/2010
§
Banks shall not outsource their critical ICT systems ...
[...]

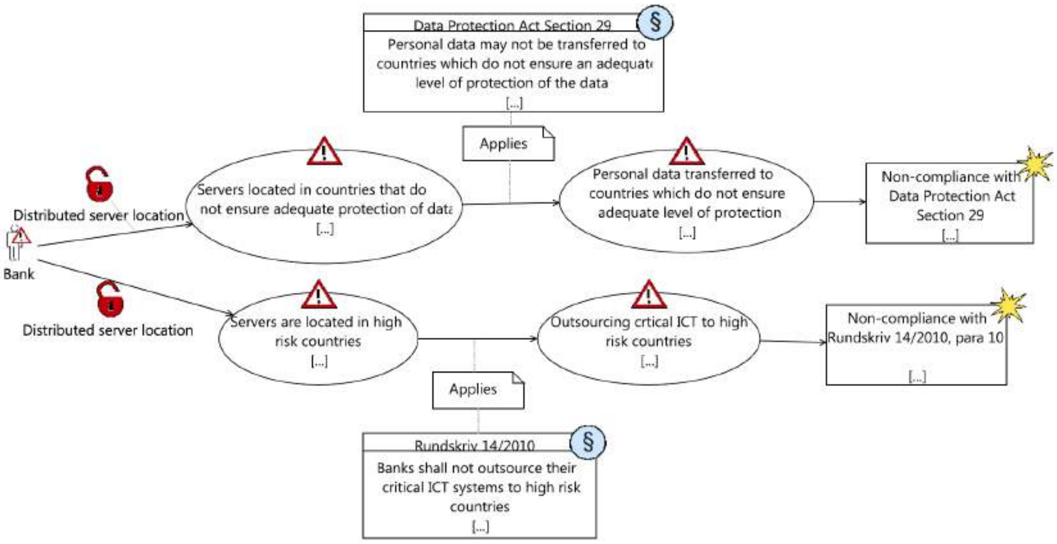
Figure 34 – Facts-based: distributed server location

Activity 2: Identify triggers and model them in CORAS

In the fact-based approach, the meeting or brainstorming activity with the stakeholders focuses on how the distributed server location could affect the Customer's compliance with these rules and on identifying the specific circumstances in which the distributed location might lead to non-compliance. The legal and compliance team identifies the following triggers:

- The servers used to store personal data are located in countries that do not ensure adequate protection.
- The servers are located in high-risk countries.

These aspects can further be modeled in the CORAS approach as threat scenarios in order to get the full picture of the compliance risks in the given context.



Data Protection Act Section 29
§
Personal data may not be transferred to countries which do not ensure an adequate level of protection of the data
[...]
Applies
Distributed server location
Bank
Servers located in countries that do not ensure adequate protection of data
[...]
Personal data transferred to countries which do not ensure adequate level of protection
[...]
Non-compliance with Data Protection Act Section 29
[...]
Distributed server location
Servers are located in high risk countries
[...]
Outsourcing crtical ICT to high risk countries
[...]
Non-compliance with Rundskriv 14/2010, para 10
[...]
Applies
Rundskriv 14/2010
§
Banks shall not outsource their critical ICT systems to high risk countries
[...]

Figure 35 – Facts-based: distributed server location with triggers

Step 5: Compliance risk estimation

The analysis team agrees that for a globally operating cloud provider it is ‘very likely´ that servers are distributed across different geographical locations including countries that do not ensure adequate protection of data. Given the number of high risk countries is lesser than the number of countries that do not ensure adequate protection, the team reduces the likelihood of the *servers are located in high-risk countries* to ´likely´. However, the team explains that the mere fact that the provider uses servers in countries that do not ensure adequate protection does not necessarily lead to personal data transfer. This is the case so far as the provider does not use the servers to store the data of the customer under analysis. This led to the team reducing the likelihood value of the *Personal data transferred to countries which do not ensure adequate level protection of data* to ´likely´. Similarly, the team agrees that if *Personal data is transferred to countries which do not ensure adequate level protection of data,* then it is ´possible´ that *Non-compliance with Data Protection Act Section 29* is found to occur.

The team was rather uncertain about whether the storage of the data by the cloud provider in high risk countries would constitute outsourcing. After deliberating on the issue, the team agrees that it is ´possible´ that *Outsourcing of critical ICT systems to high risk countries* might happen if the cloud provider uses servers in these countries. Although the mere fact that the provider uses servers in high risk countries does not mean that the client´s data will be stored in such countries, the team indicates that if the client´s data is stored in such countries, it is ‘possible’ that *Non-compliance with Rundskriv 14/2010, para 10* might occur. Next, the team considers the consequences of both *Non-compliance with Data Protection Act Section 29* and *Non-compliance with Rundskriv 14/2010, para 10 as having ‘major’* impact on the company compliance objective as both could lead to regulatory penalties followed by media publicity. Based on the likelihood and consequence values, the risk of *Non-compliance with Data Protection Act Section 29* is ranked as ‘high’ risk whereas the risk of *Non-compliance with Rundskriv 14/2010, para 10 i*s ranked as ‘medium’ risk. All these likelihood and consequence values are then annotated in CORAS as follows.

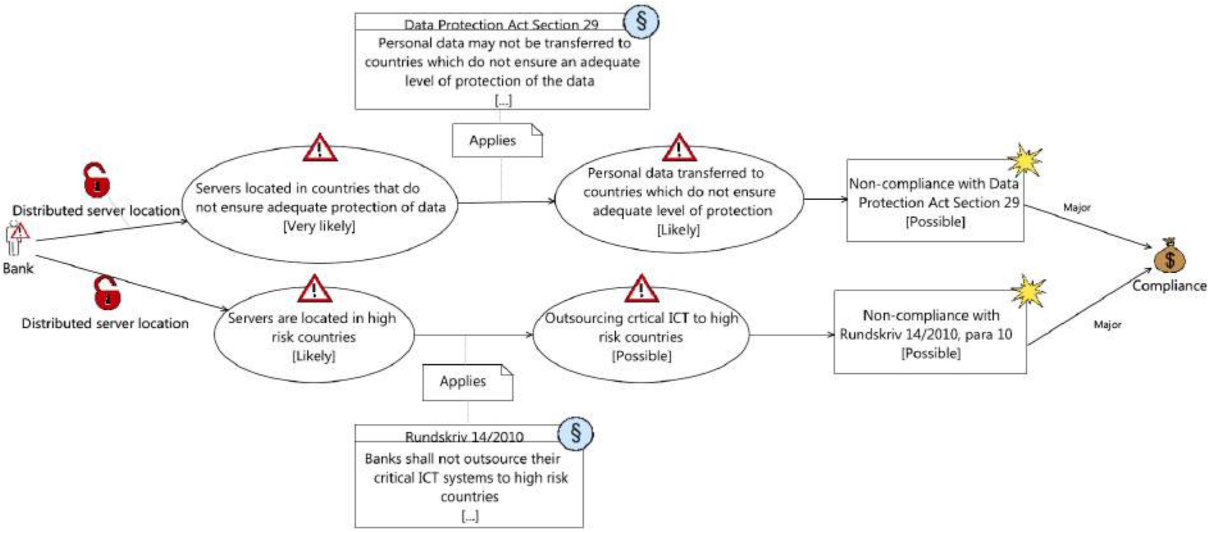

Data Protection Act Section 29
§
Personal data may not be transferred to countries which do not ensure an adequate level of protection of the data
[...]
Applies
Distributed server location
Bank
Servers located in countries that do not ensure adequate protection of data [Very likely]
Personal data transferred to countries which do not ensure adequate level of protection [Likely]
Non-compliance with Data Protection Act Section 29 [Possible]
Major
Compliance
Distributed server location
Servers are located in high risk countries [Likely]
Outsourcing crtical ICT to high risk countries [Possible]
Non-compliance with Rundskriv 14/2010, para 10 [Possible]
Major
Applies
Rundskriv 14/2010
§
Banks shall not outsource their critical ICT systems to high risk countries
[...]

Figure 36 – Facts-based: distributed server location with likelihood and consequences

Step 6: Compliance risk evaluation

Based on step 5, the Customer wanted to spend more resources on the first the *Non-compliance with Data Protection Act Section 29*. However, the team indicates that one and the same measure can be used to treat both risks. Different treatment measures are discussed including the use of a clue provider that only uses servers within Norway. But most team members did not buy into the idea since there are very few providers and due to lack of diversity in the services. The other option on the table was to negotiate contract terms for the cloud provider to store the data in Norway but this option was also seen as less viable. Finally, the team agrees on the use of a cloud provider that uses serves within the European Economic Area (EEA) and provide assurance through a third party for its compliance to do so.

Step 7: Treatment

The team identifies the following measures to mitigate the risks.

- Choose a cloud provider that uses servers within the EEA and that can produce an assurance from trusted third part for doing so. Or
- Negotiate such contractual clauses into the cloud computing agreement with the provider.

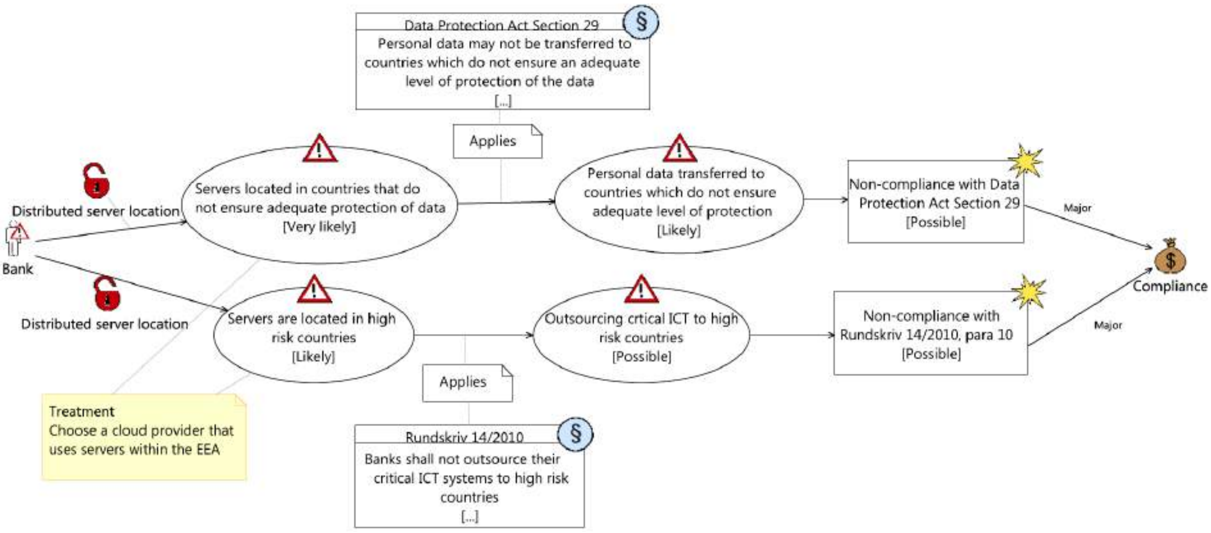

Data Protection Act Section 29
Personal data may not be transferred to countries which do not ensure an adequate level of protection of the data
[...]
Applies
Distributed server location
Bank
Servers located in countries that do not ensure adequate protection of data
[Very likely]
Personal data transferred to countries which do not ensure adequate level of protection
[Likely]
Non-compliance with Data Protection Act Section 29
[Possible]
Major
Compliance
Distributed server location
Servers are located in high risk countries
[Likely]
Outsourcing crtical ICT to high risk countries
[Possible]
Non-compliance with Rundskriv 14/2010, para 10
[Possible]
Major
Applies
Treatment
Choose a cloud provider that uses servers within the EEA
Rundskriv 14/2010
Banks shall not outsource their critical ICT systems to high risk countries
[...]

Figure 37 – CORAS treatment diagram for facts-centered: distributed server location

Step 8: Design and implement compliance measures

As a result of the suggested treatments, a contractual provision is drafted that requires the cloud provider to store all data within the EEA. The contractual provision also requires that servers used for backup purposes should also be located within the EEA. The contract also addresses that every six months the cloud provider will submit a report from trusted third party that the data has not been stored outside the EEA.

4 Summary

This document constitutes the second and intermediate version of the RASEN methodologies. The final version will be described in the RASEN deliverable D5.3.3.

In this document, we have described a unified process that combines three domains, which are traditionally considered distinct: security risk assessment, security testing, and legal compliance. Furthermore, we have described how the unified process may be instantiated to support specific combinations of the three domains, and discussed the integration points involved in these combinations.

Finally, we have described specific RASEN processes, which may be seen as instantiations of the unified RASEN process. Each step of these specific processes has been described in detail, and examples demonstrating the use of the processes have been given.

References

[1] Article 29 Data Protection Working Party, ‘Opinion 05/2012 on Cloud Computing’ adopted on 1 July 2012 (WP 196)

[2] Australian Standard AS 3806-2006 Compliance programs.

[3] Commission, ‘Proposal for a Regulation of the European Parliament and of the Council on the protection of individuals with regard to the processing of personal data and on the free movement of such data (Draft General Data Protection Regulation)’ Com (2012) 11 final.

[4] Commission, ‘Proposal for a Directive of the European Parliament and of the Council concerning measures to ensure a high common level of network and information security across the Union’ COM (2013) 48 fina

[5] Commission, ‘Proposal for a directive of the European parliament and of the Council on payment services in the internal market and amending Directives 2002/65/EC, 2013/36/UE and 2009/110/EC and repealing Directive 2007/64/EC, COM(2013) 547 final.

[6] COSO, ‘Enterprise Risk Management: An Integrated Framework. Committee of Sponsoring Organizations of the Treadway Commission, 2004

[7] European Network and Information Security Agency (ENISA).: Data Protection Notification in the EU. (2011).

[8] FERMA. A Report by Harvard Business Review Analytic Services: Meeting the Cyber Risk Challenge. (2012). http://www.computerweekly.com/blogs/public-sector/Meeting%20the%20Cyber%20Risk%20Challenge%20-%20Harvard%20Business%20Review%20-%20Zurich%20Insurance%20group.pdf

[9] Harvard Business Review Analytic Services.: Meeting the Cyber Risk Challenge. (2012).

[10] International Standards Organization. ISO 29119 Software and system engineering - Software Testing-Part 2 : Test process (draft), 2012

[11] International Standards Organization. ISO 31000:2009(E), Risk management – Principles and guidelines, 2009.

[12] Mahler T.: Legal Risk Management: Developing and Evaluating Elements of a Method for Proactive Legal Analyses, with a Particular Focus on Contracts “, University of Oslo, PhD thesis, 2010, unpublished.

[13] Mahler, T.: Defining legal risk. Proc. Commercial Contracting for Strategic Advantage – Potentials and Prospects, Turku University of Applied Sciences, 2007, pp. 10-31.

[14] Lund M.S, Solhaug B., and Stølen K.: Model-Driven Risk Analysis, The CORAS Approach, Springer Verlag Berlin Heidelberg 2011, ISBN: 978-3-642-12322-1

[15] Regulation on the Processing of Personal Data (Personal Data Regulation) Regulation No.1265 of 15 December 2000

[16] Vraalsen, F., Lund, M.S., Mahler, T., Parent, X., Stølen, K..: Specifying Legal Risk Scenarios Using the CORAS Threat Modelling Language: Experiences and the Way Forward. In: Herrmann, P. et al. (eds.): iTrust 2005. LNCS, vol. 3477, pp. 45–60. Springer, Heidelberg (2005).

[17] Esayas, S.Y.: Utilizing Security Risk Analysis and Security Testing in the Legal Domain’ in: T. Bauer et al. (Eds.): ‘Risk Assessment and Risk-driven Testing’ LNCS, vol. 8418, pp. 51–67, Springer, Switzerland 2014.