Effort-dependent technologies for multi-domain risk-based security testing (DIAMONDS[1])

As computerized systems, services and infrastructures have become an important part of society, the need for security has become increasingly evident. Today, particularly in light of the evolution and increasing use of the Internet, the need for security concerns nearly every user of computerized systems, be it private users, industrial users, or government users.

While the future internet creates new business opportunities (e.g. online banking services) and security mechanisms (e.g. authentication by mobile phone), it also creates new security threats and vulnerabilities as connectivity and the multi-domain created by trust and organizational boundaries increases. This adds an extra level of complexity, as both risks and assumptions are hard to anticipate and yet they cannot be deemed indefinitely. On the contrary, they must be monitored and reassessed continuously.

The aim of the DIAMONDS project is to strengthen the ability of Norwegian companies to face the new security challenges posed by the future internet by transferring state-of-the-art security assessment techniques to the industry. In particular, as illustrated in Figure 1, we aim to develop industrial guidelines and a supporting framework to help businesses find a balanced approach within the three-dimensional space of invested *effort*, *security testing* and *risk analysis*.

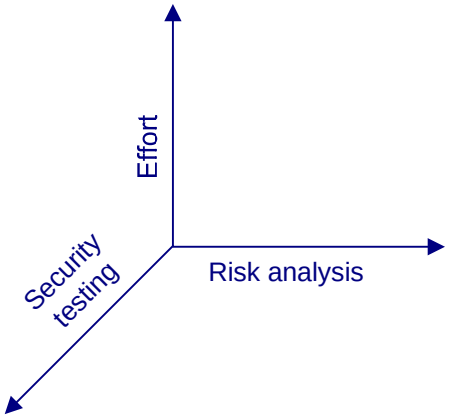

Figure 1 - Dimensions of DIAMONDS

Security testing is a widely used technique for assessment. It is one of the few techniques that can be used to gain confidence that a system (not just its specification) together with is environment (e.g., operating system, network, and legacy code) is secure. Security testing is particularly useful in light of the dynamic and evolving multi-domain of trust envisioned by the future internet where, for instance, end users are more and more empowered and therefore decide (often on the fly) on how content and services are shared and composed.

The challenge with security testing, however, is that only some aspects of a system can be tested. In response to this, many advocate the notion of risk-based testing. Its main idea is to use risk analysis to identify and prioritize those important parts of systems that need to be tested. One of the key challenges of risk-based testing is to relate risk analysis results at a high-level of abstraction (e.g. business level) to test-cases at a low-level of abstraction (e.g. implementation level). A particular challenge is how to relate risks and security test cases to facilitate assurance and maintenance in the multi-domain created by the fragmentation of trust boundaries as envisioned the future internet.

In practice, security assessments are always constrained by cost and time. The effort available for doing a security assessment can vary a great deal depending on e.g. target of analysis and business process, yet effort is one of the most important factors for determining the scope, depth, and (aspects of) techniques used for the security assessment. Any general technique for security assessment which fails to take effort into account is not likely to be very practical. We therefore aim to have a strong emphasis on effort-dependence.

[1] This project will be a part of an international ITEA2 project named DIAMONDS currently under evaluation. The Norwegian project has the same short-name, but a different title that better reflects the work tasks assigned to the Norwegian consortium within the international project, where the Norwegian consortium is leading work package “security testing methodology”.

In summary, our main objective is to develop industrial guidelines and a supporting framework for adapting risk-based testing techniques in the multi-domain created by trust and organizational boundaries envisioned by the future internet. The guidelines and its supporting framework should take effort in account as a key factor in determining what aspects of the guidelines and framework to use and how to use them.

1. Objectives

1.1. Three motivating scenarios

To motivate and illustrate the relevance of DIAMONDS we describe three scenarios in which the results of the project are relevant.

Security audit of supplier services

A company has to purchases services from software suppliers on a regular basis. The security department of the company has to audit both the suppliers and the services before purchasing them, and later during use if they are purchased. Current state-of-the-art methods of service auditing across organisational boundaries (e.g., SAS70[2] and the ISO2700 series) suffer from (1) lack of proper methodological guidelines, (2) security metrics for comparing one service to another, and (3) lack of support for combining high-level security assessment with in-depth security assessments. A better methodology for auditing of supplier services is therefore needed. The methodology should both support an in depth security analysis, but at the same time present an overall risk-picture which enables the decision maker to make decisions without knowing all the low-level details of the security analysis.

The DIAMONDS project will address this problem by developing guidelines and a supporting framework for using risk analysis to manage service auditing across organisational boundaries based on security testing. The guidelines will address the question of relating the in depth test analysis to the overall security risk-picture.

Maintaining security in virtual organizations

The number of services communicating across the boundaries of organizations is already quite large, and it is likely to increase as envisioned by the Future Internet. While these kinds of services offer new possibilities, they also create new security challenges that prevent the potential of the services to be fully exploited. For instance, many service suppliers offer remote updates of their services, yet many companies both lack the access control mechanisms for these updates to be done in a completely trusted manner and tools automating the evaluation of the security level of new services (e.g. Norges Bank has to physically transport in personnel by air to do certain kinds of updates which could have been done remotely). One of the problems in this scenario is that many companies need a centralized security control of distributed systems and information.

In the DIAMONDS-project, we will develop a proof-of-concept authentication scheme which addresses this problem on the basis of an authentication mechanism (developed by Encap) for mobile phones. We will do a security analysis of this based on the guidelines and the framework developed in this project.

Light weight risk and security testing

In order to deal with new security threats, a company has decided to introduce an integrated methodology for risk analysis and security testing which can be carried out continuously as an integrated part of the business process. To ensure that the risk picture is updated regularly, the company needs to enable system owners to do small risk and security testing analyses on their own without heavy involvement from the security department. Hence, the company needs a light weight methodology for doing risk analyses and security testing without much effort.

In the DIAMONDS project, we aim to address this scenario by developing guidelines and a supporting framework in which effort is taken as a key parameter for determining what aspects of the framework and the guidelines to use. Hence, the guidelines are both applicable for an overall business-level methodology, and for the methodology for doing small risk and testing analyses by system owners.

[2] American Institute of Certified Public Accountants, Audit Guide for Service Organizations Applying SAS No. 70, (2006)

1.2. Primary and secondary objectives

The aim of the DIAMONDS project is to strengthen the ability of Norwegian companies to face the new security challenges posed by the future internet by developing guidelines for improved security testing through the use of risk analysis. The guidelines should be sensitive to work-effort (man hours) available for the testing; i.e. the guidelines should be scalable with respect to effort. In particular also offer support when the number of man-hours available is small.

1.3. Secondary objectives

DIAMONDS aims to fulfil the following secondary objectives:

(1) Establish a baseline industrial guideline for the use of risk analysis to improve the quality of security testing based on an adaptation of existing risk management and security testing technologies.

(2) Generalise the baseline industrial guideline to facilitate scaling w.r.t. to work effort.

(3) Provide a framework of methods, techniques, languages and tools to support the guideline.

(4) Demonstrate and evaluate the guideline and its framework in field-trials conducted in collaboration with the industrial partners of the project.

(5) Disseminate the results through publications (popular as well as research), public workshops, future national and international projects, and university lectures.

(6) Graduate two PhD fellows.

1.4. Relevance for call for proposals of BIP projects under the VERDIKT programme

There are two general challenges posed by the **future internet** that DIAMONDS will address:

Fragmentation of organizational and trust boundaries: The future internet will offer a new generation of services (e.g. a hybrid aggregation of content and functionality), service factories (e.g., personal and enterprise mash-ups), and service warehouses (e.g., platform as a service). One specific service instance may thus be created by multiple service development organizations, be hosted and deployed by multiple providers, and be operated and used by a virtual consortium of business stakeholders. While the creative space of service compositions is in principle unlimited, so is the fragmentation of ownership of both services and content, as well as the complexity of implicit or explicit relations between participants in each business value chain that is generated. The future internet stretches the present know how on building secure service-based systems: more stakeholders with different trust levels are involved in a typical services composition and a variety of potentially harmful content sources are leveraged to provide value to the end user. This is attractive in terms of degrees of freedom in the creation of service offerings and businesses, yet this also creates more vulnerabilities and risks as the number of trust domains in an application gets multiplied, the size of attack surfaces grows and so does the number of threats. As the complexity of the risk picture increases, so does the need for having a clear and accurate high-level risk picture on which decisions about risk mitigation/elimination can be made. The DIAMONDS project will develop guidelines for risk-based security testing which particularly takes fragmentation of trust domains into account as well as security across organizational and trust boundaries.

Dynamic multi-domains of trust: The future internet will be an intrinsically dynamic and evolving place where, for instance, end users are more and more empowered and therefore decide (often on the fly) on how content and services are shared and composed. This adds an extra level of complexity, as both risks and assumptions are hard to anticipate and yet they cannot be deemed indefinitely. On the contrary, they must be monitored and reassessed continuously. In this setting, an accurate security assessment of a service must not only take the service into consideration, but also the dynamic environment in which it executes. Security testing is a particularly useful technique in this setting, because its main strength over other techniques (such as static analysis or specification based verification) is that the real environment (not just a model of it) in which the service executes is taken into account. DIAMONDS aims to address security testing in the setting of dynamic trust domains, and develop guidelines and a supporting framework where issues related to this scenario are taken into account.

With respect to the VERDIKT matrix of research topics and research pillars, the DIAMONDS project is in particular positioned in the intersection between the research pillar **security, privacy and vulnerabilities** and the research topics **internet of things** and **mobile internet.**

With the evolvement of future internet we see an increased connectivity of devices. Internet devices can communicate through more than one channel, and devices which traditionally have not been connected will

have the capability of being connected. Examples of such devices are credit cards and other 'tagged' objects (key fobs, watches and simpler objects). In this context, DIAMONDS will explore the potential of using multi channel verification of (connected) objects as risk mitigation for sophisticated online attacks. Furthermore, we will explore and assess the possibilities of using e.g. the connectivity of NFC-enabled credit cards and the multi channel capabilities of mobile terminals to design security measures against sophisticated man-in-the-middle (MITM), man-in-the-browser and online phising attacks. Based on the case study experiences, we will develop guidelines both addressing the security issues posed by increased connectivity of devices, and for assessing the security of mobile terminals with multi-channel capabilities.

2. Frontiers of knowledge and technology

In this section, we survey the literature of the relevant research areas.

2.1. Risk Analysis

This section provides a brief overview of some of the methods related to risk analysis. Our aim in DIAMONDS is not to focus on one particular approach, but rather develop guidelines that can be used in combination with most leading approaches to risk analysis, some of which are mentioned blow.

OCTAVE[3] (Operationally Critical Threat, Asset, and Vulnerability Evaluation) is a general approach for evaluating and managing information security risks. As information security includes issues related to both business and technology, an interdisciplinary analysis team that includes people from both the business units and the information technology (IT) department performs the evaluation. OCTAVE provides a snapshot analysis at a given point in time. Therefore, OCTAVE advises that the organization either performs new evaluations periodically or triggered by major events, such as corporate reorganization or redesign of the computing infrastructure. OCTAVE comes with predefined templates for documenting information during the analysis.

CRAMM[4] (CCTA Risk Analysis and Management Method) provides a stepwise and disciplined risk analysis method that takes both technical and non-technical (e.g. physical and human) aspects of security into account. In its original form, it was adopted as a standard by the U.K. government organization CCTA (Central Computer and Telecommunications Agency). During a CRAMM analysis, the reviewer gathers information by interviewing the asset owners, system users, technical support staff and security manager. A standardized CRAMM format is used for documenting results, mostly in the form of specialized tables.

Microsoft has developed their own risk guideline[5]. This guideline defines the Microsoft Security Risk Management Process, which has four primary phases. As risk management is viewed as an ongoing process, these phases constitutes the parts of a risk management cycle. The guideline also goes further than defining this cycle. For example, it provides lists of common assets, threats and vulnerabilities. However, these are not intended to be comprehensive, and analysts are encouraged to add or delete items as necessary.

CORAS[6] is a method for conducting security risk analysis. CORAS provides a customised graphical language for threat and risk modelling, and comes with detailed guidelines explaining how the language should be used to capture and model relevant information during the various stages of the security analysis. In this respect CORAS is model-based. Special CORAS diagrams are used for documenting intermediate results, and for presenting the overall conclusions. The CORAS language is supported by a structured semantics translating CORAS diagrams into natural language (English) sentences[7].

---

[3] Alberts, C. J. and Dorofee; A. J., "OCTAVE Criteria Version 2.0", Tech. report CMU/SEI-2001-TR-016. ESC-TR-2001-016, 2001.

[4] www.cramm.com

[5] Microsoft, "The Security Risk Management Guideline", Microsoft Solutions for Security and Compliance, Microsoft Centre of Excellence, 2006

[6] Folker den Braber, Ida Hogganvik, Mass Soldal Lund, Ketil Stølen, Fredrik Vraalsen. Model-based security analysis in seven steps – a guided tour to the CORAS method. BT Technology Journal, 25(1):101-117, January 2007

[7] Heidi E. I. Dahl, Ida Hog-ganvik, Ketil Stølen. Structured semantics for the CORAS security risk modelling language. Pre-proceedings of the 2nd International Workshop on Interoperability solutions on Trust, Security, Policies and QoS for Enhanced Enterprise Systems (IS-TSPQ'07). Report B-2007-3, pages 79-92, Department of Computer Science, University of Helsinki, 2007

2.2. Security testing

Within academia, testing a system for its security is surprisingly (considering the importance of the issue and the wealth of research on other methods of addressing security in software-based systems) a relatively new concern that has started to be addressed in the last few years, and the first research workshop on this topic was held in 2008 (SecTest08, Lillehammer). Of course, a number of tools have long been around to target specific attacks on systems (e.g. vulnerability scanners), but we refer here to the more systematic testing of systems w.r.t. specified policies or security properties/requirements.

Security testing is used to determine if a software-based system protects data and maintains functionality as intended. Security testing tests for confidentiality, integrity, authentication, authorization, availability, and non-repudiation. In service-base environments, security testing is in particular compelling as service-based networked applications are distributed systems crossing network domains, are systems in dynamic configurations and have to cope with usages in unknown contexts and user scenarios.

Security control is typically defined by security policies. Most work related to security policy can be mainly divided into two parts: the description of the policy itself and the verification of the rules. Until recently, in many systems, there was no real policy specification, only a description in terms of low-level mechanisms such as access control lists. Concerning the verification of the rules, most work in the field deals with testing of firewall rules. The first proposals consisted in performing testing of rules by hand. This implies that test construction is performed by human experts who focus on detecting traces of known attacks. Most recently, research tended to concentrate on the verification of security rules in order to detect errors or mis-configurations such as redundancy, contradiction or collision[8][9]. Some approaches propose to focus on validation by checking the conformance of a system with respect to a security policy. In[10], authors show how an organisation's network security policy can be formally specified in a high-level way, and how this specification can be used to automatically generate test cases to test a deployed system. In contrast to other firewall testing methodologies, such as penetration testing, this approach tests conformance to a specified policy.

2.3. Risk-based testing

The combination of risk-analysis and testing is known as risk-based testing (RBT). Bach[11] is one of the first to consider the notion of RBT. He discusses two heuristic approaches to risk-based testing. One approach is based on starting with the system parts and identifying risks for those. The other approach is based on starting from the risks and identifying system parts that are relevant for those risks. A similar informal discussion of RBT is given by Redmill[12][13]. In particular, he discusses how risk analysis can be used prioritize those parts of a software system that needs to be tested during software development. He also discuses what kind of risks should be sought, and how they can be identified an analysed. A more structured approach to RBT is proposed by Amland[14]. He also discusses experiences gained from applying the approach in financial application case study. The reported experiences seem encouraging. Stallbaum et.al.[15] goes one step further than the previously cited authors by proposing an automated technique for risk-based test case generation and prioritization. The technique is based on augmenting test models (specified by UML activity diagrams) with risk information. From these test models, a prioritized set of test scenarios is generated.

---

[8] V. Darmaillacq, J.-C. Fernandez, R. Groz, L. Mounier, and J.-L. Richier. Test generation for network security rules. In Proc TestCom, (2006).

[9] J. Garcıa-Alfaro, F. Cuppens, and N. Cuppens-Boulahia. Towards filtering and alerting rule rewriting on single-component policies. In Proc SAFECOMP, (2006).

[10] D. Senn, D. A. Basin, and G. Caronni. Firewall conformance testing. In Proc TestCom, (2005).

[11] J. Bach. Heuristic risk-based testing. Software Testing and Quality Engineering Magazine, 1(6), 1999.

[12] F. Redmill. Exploring risk-based testing and its implications. Software Testing Verification and Reliability, 14(1):3-15, 2004.

[13] F. Redmill. Theory and practice of risk-based testing: Research articles. Software Testing Verification and Reliability, 15(1):3-20, 2005.

[14] S. Amland. Risk-based testing: risk analysis fundamentals and metrics for software testing including financial application case study. Journal of Systems and Software, 53(3):287-295, 2000.

[15] H. Stallbaum, A. Metzger, and K. Pohl. An automated technique for risk-based test case generation and prioritization. In Proc. of the 3rd international workshop on Automation of software test (AST'08), pages 67-70. ACM, 2008.

2.4. Modelling in relation to risk-based testing

There are many documentation techniques that are used to support risk identification and estimation ranging from FMEA (Failure Mode Effect Analysis) tables[16], fault trees[17], threat trees[3], Bayesian networks, to the UML (Unified Modeling Language) based CORAS language[6]. However, little research has been done on documenting the relationship between risk analysis results and testing. The relationship of system models its security requirements and testing (without taking risk into account) has however, been addressed in previous approaches. These approaches consider test generation from e.g. abstract system models[18][19][20], models of security requirements[21][22], or threat models[23]. While none of these approaches address risk analysis, it would be interesting to consider the threat modelling approach[23] as an intermediate step in deriving test cases from risk analysis documentation. In this approach, UML sequence diagrams are used to specify threat a model, i.e., event sequences that should not occur during the system execution. The threat model is then used as a basis for code instrumentation. Finally, the instrumented code is recompiled and executed using randomly generated test cases. If an execution trace matches a trace described by the threat model, security violations are reported and actions should be taken to mitigate the threat in the system.

3. R&D Challenges

Risk analysis and security testing are often used in isolation, and not much research has been done on combining the two techniques, particularly with emphasis on effort-dependence. This leads to the following main research questions:

- **RQ1**: *What are good industrial guidelines for an effort-dependent use of risk analysis to improve the quality of security testing?*
- **RQ2**: *How should we best describe risk analysis and security test related documentation and their relationships?*

The following sub-questions are associated with **RQ1**:

- How can we use risk-analysis to prioritize security test cases?
- What security testing techniques are most suitable in combination with risk analysis?
- How can we use security testing results to influence the overall outcome of the risk analysis?
- How can we use risk analysis to determine when to stop security testing?
- How can we scale the risk and security test analysis w.r.t. effort?

The following sub-questions are associated with **RQ2**:

- How can we combine existing risk analysis languages with languages for specifying security test cases and test results?

---

[16] A. Bouti and A. D. Kadi. A state-of-the-art review of FMEA/FMECA, International Journal of Reliability. Quality and Safety Engineering, vol. 1, 1994.

[17] IEC61025, Fault Tree Analysis (FTA), 1990

[18] J. Jurjens. Model-based security Testing Using UMLSec. Electronic Notes in Theoretical Computer Science 220(1): 93-104, 2008

[19] J. Jurjens and G. Wimmel. Specification-Based Testing of Firewalls. In proceedings of the 4th International Andrei Ershov Memorial Conference on Perspectives of System Informatics (PSI'01), pages 308-316, Springer 2001.

[20] G. Wimmel and J. Jurjens. Specification-based Test Generation for Security-Critical Systems Using Mutations. In proceedings of the International Conference on Formal Engineering Methods (ICFEM'02), Springer, 2002.

[21] M. Blackburn, R. Busser, and A. Nauman. Model-based Approach to Security Test Automation. In proceedings of the 13th International Symposium on Software Reliability Engineering (ISSRE'02), 1999.

[22] T. Mouelhi, F. Fleurey, B. Baudry, and Y. L. Traon. A Model-Based Framework for Security Policy Specification, Deployment and Testing. In proceedings of the 11th International Conference on Model Driven Engineering Languages and Systems, pages 537-552, Springer, 2008.

[23] L. Wang, E. Wong, and D. Xu. A Threat Model Driven Approach for Security Testing. In proceedings of the 3rd International Workshop on Software Engineering for Secure Systems (SESS'07), pages 10-16, IEEE Computer Society, 2007.

- To what extent can we formalize and possible semi-automate the generation of security test case specifications from risk documentation and the system model?
- What distinguishes model based testing of security requirements from traditional testing of functional requirements?

The two research questions represent two focus areas addressed by the two PhD-fellows we apply for.

4. Research approach/methods

In Section 4.1 we describe our general research method and in Sections 4.2 and 4.3 we describe our approach for addressing research questions **RQ1** and **RQ2**, respectively.

4.1. Research Method

Experimental Computer Science and Engineering (ECSE) is defined as "the building of, or the experimentation with or on, nontrivial hardware or software systems". ECSE is a synthetic discipline, studying phenomena created by humans rather than those given by nature, and there is much room for creativity and few direct physical constraints. The primary focus of ECSE is on artefacts – software and/or hardware which is the subject of study, apparatus used to conduct the study, or both. Often a significant part of the intellectual effort of the experimental research is embodied in the artefact. Artefacts in ECSE can have one of the following three roles in the research:

- Proof-of-performance. The artefact shows that a certain performance can be achieved, or that it is in some other measurable way an improvement of previous implementations. The results are usually quantitative.
- Proof-of-concept. The existence of an artefact in this role proves that a concept is possible to realise, at least in one configuration. The artefact is usually too complex to derive the behaviour of by only using logical reasoning or abstract argument.
- Proof-of-existence. An artefact in this role demonstrates a new phenomenon through its existence which is impossible or difficult to grasp only through documentation. A historical example of this is the computer mouse.

In DIAMONDS we will develop artefacts as proofs-of-concept. The artefacts will be validated against requirements from end users. To this end DIAMONDS conducted according to an iterative process where new ideas and artefact prototypes are tried out in industrial field trials. The results from these field trials will indicate the extent to which the expectations are likely to be fulfilled and will thus help direct the next iteration of research.

4.2. Effort-dependent risk-based security testing guidelines

Figure 2 illustrates the overall approach of combining model-based risk analysis with the general approach to model based testing[24]. In model-based testing, the model is used to generate traces, or test cases for an implementation, according to a test case specification. The test case specification is a selection criterion on the set of the traces of the model. The risk analysis, which is conducted on the basis of the system model, is used to define the appropriate test case specifications.

[24] A. Pretschner and J. Philipps. Methodological issues in model-based testing. In Model-Based Testing of Reactive Systems, volume 3472 of Lecture Notes in Computer Science, pages 281-291. Springer, 2005.

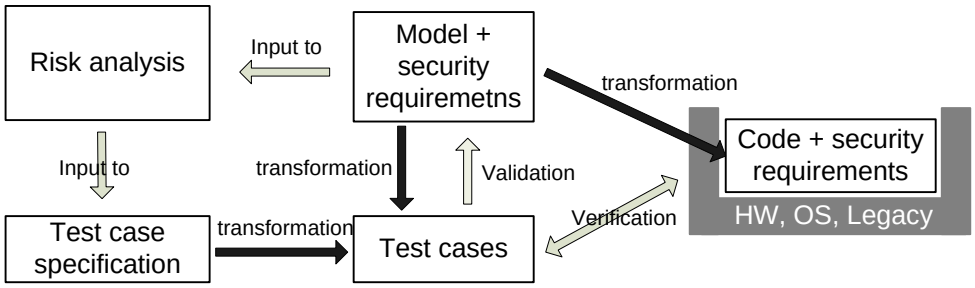

Figure 2 - Risk-based testing

The risk-based testing is useful as it offers the possibility of:

(1) using the risk analysis to select those parts of the system model that needs to be tested;

(2) to assess whether a system is ready for deployment;

(3) and to relate high-level risks to low-level errors uncovered by testing.

We aim to investigate research question **RQ1** by developing guidelines (according to the general approach illustrated in Figure 2) for how to performing risk-based testing based on an adaptation of existing tools and methodology.

To ensure good scalability w.r.t. available effort, we aim to partition the guidelines into modules which can be selected and combined according to effort.

4.3. Modelling risk-based security testing

As illustrated in Figure 2, there are many artefacts (e.g. the system model, test cases, risk analysis results) that must be taken into account in a risk-based testing approach. These artefacts and their relationships must be characterized and described.

Characterization of the system model, the system implementation, the security requirements, and the relationship between these has been studied previously. The general set-up is illustrated in the lower-most box of Figure 3. Here two levels of abstraction are illustrated, but in general there may be many. The idea is as follows: the system and its security requirements are described by models which are at a higher level of abstraction than the code level. At this level, we can show that the system model is in *adherence* with its security requirements, i.e. that the system model is secure. From the abstract level, a system implementation and its associated concrete security requirements can be derived (manually, semi-automatically, or automatically). Also at the concrete, we can show that the implementation is in adherence with its security requirements. To avoid doing a completely new security analysis all over again at the concrete level, we can ensure that the transformation from the abstract to the concrete level preserves the adherence relationship for those part of the implementation that are described by the abstract system model. This is known as *adherence preservation*.

The general approach of investigating research question **RQ2** is to characterize the risk model and the test cases in addition to the system and its security requirements. The idea, as illustrated in Figure 3, is to use risk analysis to assess the adherence between the (abstract) system model and its security requirements, and to use testing to analyse the adherence at the implementation level.

The use of security testing to analyse the concrete adherence relationship is interesting for the following reason: although the code might be a correct refinement of the model, one must in addition take into account its environment consisting of hardware, operating system, and legacy software components which are not feasible to describe by the abstract model. It is therefore in many cases impossible to ensure completeness of the adherence preservation. Security testing, particularly in the setting of the dynamic trust domains envisioned by the future internet, is one of the few feasible solutions to the problem gaining confidence that the code together with its environment does not violate the security requirements.

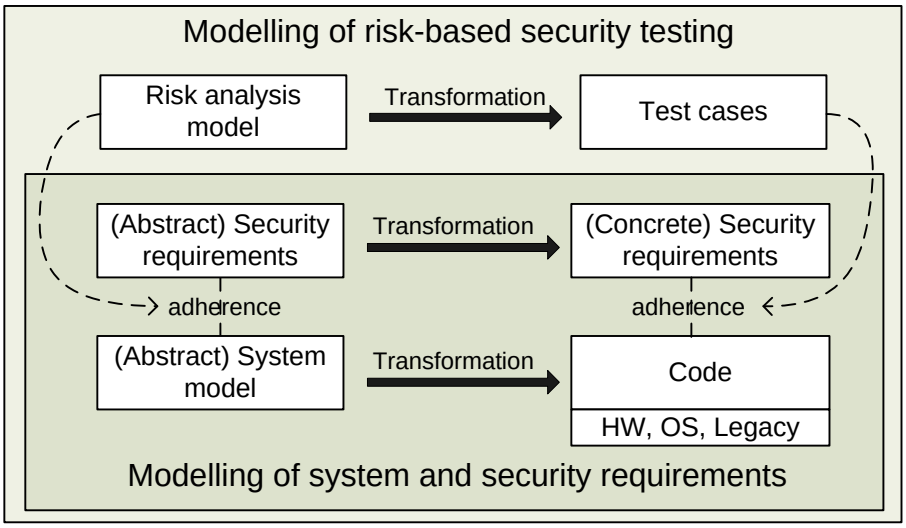

Figure 3 – Modelling of risk-based testing artefacts