Test Prioritization of Risk-based Security Tests

Michael Berger

Fraunhofer FOKUS, Kaiserin-Augusta-Allee 31, D-10589 Berlin,
michael.berger@fokus.fraunhofer.de

Abstract Many approaches are developed for efficient identification and estimation of security risks. One big challenge is to prioritize the related test cases of identified risks. The effort and costs of security testing can be high and the budget is limited. The challenge is to get a proper proportion between test effort and potential system harm. Based on the results of security testing countermeasures can be implemented to achieve a proper security level for a system. In the RASEN project, one goal is to develop risk-based security testing methods and tools as well as a methodology for risk-based security testing.

1 Security Risk Assessment and Risk-based Security Testing

Security risks can be understood as an intended harming of organization assets by exploiting system vulnerabilities. Each risk has a likelihood value and the effect for the asset can be estimated with a consequence value. The risk value, composed of both likelihood value and consequence value, is used for comparing a risk with each other and to determine if the risk is of interest for further treatment.

The testing of security risks is deeply influenced by the risk analysis. The tests are designed to simulate unknown attacks. That can be trial-and-error tests on interfaces by generating semi-randomized input values to get access to secure data or to force program crashes, also known as fuzzing. A security test case can also generate multiple sessions in order to perform a denial-of-service attack to the server. These parameters, the test data, the amount of inputs and the behavior as well as the reply data of the system are essential parts of a security test case. The test execution is also influenced by the identified risks. The risk value determines not only if related attacks will be tested but also how much test resources can be assigned for each risk. For example test parameters like the amount of inputs can vary with the risk value.

The security risk testing also directly influences the security risk assessment. Unexpected effects while testing or anomalies in logs of test incident reports can detect new vulnerabilities of the system or not examined threat scenarios. As a result, new attacks and risks can be evaluated and analyzed in a following risk assessment. Additionally, test reports can be used for new evaluation and estimation of risk values like likelihood and consequence, e.g. when testing with much effort did not reveal any vulnerability. The likelihood of the related risk can be rated down and with the next test cycle other risks can be tested more intensive.

While security tests are identified by risk assessment, testing helps to improve the risk assessment in each iteration step. At least, the security risk testing verifies installed countermeasures and treatments.

2 Prioritization Approach of Security Testing

In the RASEN project, one goal is to develop tools and techniques for risk assessment, risk-based testing and prioritization of security risk tests. For risk assessment, the CORAS approach is used. The CORAS methodology was developed by the Scandinavian organization SINTEF. The risk assessment methodology is developed according to risk assessment standard ISO 31000. It starts with the identification of all assets of the system that require protection. In the next steps the attack paths with all components are identified. An attack path starts from a threat and ends in harmed assets by exploiting vulnerabilities of the system. Additionally, probabilities of single events and a consequence for each attack are roughly estimated. All data are included in risk diagrams created in the CORAS risk language with help of an editor also developed for CORAS. The most relevant elements of the CORAS risk language are:

- the **Asset**, a system part that requires protection. It can be a real system part like personal data in database or an intangible good e.g. reputation.
- the **Threat**, the initiator of an attack scenario. It could be a malicious attacker, an undeliberate user or a software problem, where the hazard is starting from.
- the **Vulnerability**, an identified weakness that can be exploited by the threat.
- the **Threat Scenario**, a single and concrete event on an attack path.
- the **Unwanted Incident**, a specific threat scenario that immediately harms an asset. The unwanted incident is the last event of an attack path to let the attack be successful if it occurs.
- the **Treatment**, a countermeasure to decrease the exploitability of a weakness or to reduce the consequence of an attack.

For security risk testing the concept of **security test patterns** is developed in RASEN project. A test pattern can be understood as a test design draft to generate test cases for a special scenario. A test pattern guides the test design among others by stimulation and observation strategies that provide instructions or actual code how to stimulate the system under

test in order to perform a certain attack and how to determine whether the attack was successful. These code fragments can be used for automatic test generation. Also, effort for and effectiveness of tests are estimated. Such a security test pattern will be related to an identified vulnerability in the risk model. One approach in RASEN project is to build up a security test pattern catalog that allows to select proper test patterns based on risk assessment.

To combine all steps, risk assessment, risk testing and test prioritization with help of security test patterns, the tool RiskTest is under development in context of RASEN project. RiskTest is designed as a traceability platform that integrates the CORAS editor tool for risk assessment, a requirement tool where the test patterns are embedded, and for the TTCN-3 test suite from Testing Technologies for testing and test reporting. The CORAS editor is designed only for modeling risks and does not provide capabilities for calculation risk values. Therefore, a service has to be developed to collect all attack paths from the risk model, combine the likelihood and consequence values of these paths and assign the paths with a risk value. The risk value can be used for comparing each attack path with each other in order to estimate a priority value for each related test pattern. With the test pattern attributes effort and effectiveness the tester has an additional parameter for prioritization of test cases.

It is possible that a test pattern is related to more than one attack path, which means that related test cases are testing an attack sequence of many different attacks. The assigned attack paths can be of low risk, but the priority that is cumulating the risk value of all attack paths can be higher than a pattern related to only one single attack path where this risk value is very high. Additionally, test prioritization may lead to the decision that cost and time expensive tests won't be generated and executed if the harmed assets are not worth that effort.