ETSI TR 101 583 V1.1.1 (2015-03)

Methods for Testing and Specification (MTS);
Security Testing;
Basic Terminology

Reference
DTR/MTS-101583 SecTest_Terms

Keywords
analysis, security, testing

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

The present document can be downloaded from:
http://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the only prevailing document is the print of the Portable Document Format (PDF) version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

Copyright Notification

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://ipr.etsi.org).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Report (TR) has been produced by ETSI Technical Committee Methods for Testing and Specification (MTS).

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

1 Scope

The present document defines terminology and an ontology which together provide the basis for a common understanding of security testing techniques which can be used in testing communication products and systems. The terminology and ontology have been derived from latest research, but also current standards and best practices specified by a broad range of standards organizations and industry bodies. The present document aims to provide information to practitioners on techniques used in testing, and assessment of security, robustness and resilience throughout the product and systems development lifecycle. The present document lists terms and methods for the following security testing approaches:

- Verification of security functions and risk-based testing.
- Load, stress and performance testing.
- Resilience and robustness testing (fuzzing).
- Penetration testing.

Static Application Security Testing (SAST) tools and techniques are out of scope for the present document.

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the reference document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

Not applicable.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the reference document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI TS 102 165-1: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Methods and protocols; Part 1: Method and proforma for Threat, Risk, Vulnerability Analysis".

[i.2] IEEE St. 610.12-1990: "IEEE Standard Glossary of Software Engineering Terminology".

[i.3] ISO/IEC 9646-1:1994: "Information technology -- Open Systems Interconnection -- Conformance testing methodology and framework -- Part 1: General concepts".

[i.4] ISO/IEC 15408:2009: "Information technology -- Security techniques -- Evaluation criteria for IT security -- Part 1: Introduction and general model".

[i.5] VTT Publications 447: "A Functional Method for Assessing Protocol Implementation Security", 2001, Espoo. Technical Research Centre of Finland,. Kaksonen, Rauli.128 p. + app. 15 p. ISBN 951-38-5873-1 (soft back ed.) ISBN 951-38-5874-X (on-line ed.).

[i.6] "Fuzzing for Software Security Testing and Quality Assurance", 2008. Takanen, Ari, Artech House. 287 p. ISBN-13: 978-1596932142.

[i.7] ETSI TR 101 590 (V1.1.1): "IMS Network Testing (INT); IMS/NGN Security Testing and Robustness Benchmark".

[i.8] ETSI TR 101 577 (V1.1.1): "Methods for Testing and Specifications (MTS); Performance Testing of Distributed Systems; Concepts and Terminology".

[i.9] Recommendation ITU-T X.1524: "Common weakness enumeration".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

asset: anything that has value to stakeholders, its business operation and its continuity (ETSI TS 102 165-1 [i.1])

attack: technique, process or script, malicious code or malware that can be launched to exploit a vulnerability or to bypass security controls on the system

attack surface: consists of user interfaces, target protocol interfaces and reachable data paths that can be attacked within the system

black-box testing: testing that ignores the internal mechanism of a system or component and focuses solely on the outputs generated in response to the selected inputs and execution conditions (IEEE St. 610.12-1990 [i.2])

bottleneck: severe limitation of the throughput capacity of a system service due to a single cause (ETSI TR 101 577 [i.8])

consequence: outcome of an event affecting objectives, in security testing, the impact from the resulting failure to the protected assets after a successful attack or security test case

constant load: load pattern where the SUT is exposed to a fixed rate of service requests per time unit. Constant load is commonly used in performance tests of stability and availability characteristics (ETSI TR 101 577 [i.8])

exploit: security jargon for automated attacks

fail closed: software will attempt to shut itself down in case of an undesired failure to prevent further corruption by attacks

fail open: software will attempt to recover from the failure and maintain service

fail safe: software will attempt to control the failure and restrict the exploitability of the vulnerability

failure: result of a fault, in security testing, an indication of a vulnerability

false negative: in security testing, a vulnerability was not detected even if one existed

false positive: in security testing, a vulnerability was indicated, even if it did not exist or was not possible to exploit

fuzzing, Fuzz testing: negative testing technique for automatically generating and injecting into a target system anomalous invalid message sequences, broken data structures or invalid data, in order to find the inputs that result in failures or degradation of service

known vulnerability: vulnerability in a specific version of software that has been found in the past

likelihood: chance of something happening

load testing: load testing uses large volumes of valid protocol traffic to ensure that a system is able to handle a predefined amount of traffic (ETSI TR 101 590 [i.7])

model-based security testing: approach of automatically generating security tests from behavioural models

negative testing: testing for the absence of undesired functionality

off-line testing: automated test generation technique where series of tests are generated and stored for later execution, typically as a test script or set of individual tests

on-line testing: automated test generation and execution technique where test is generated and executed at the same time, possibly with capability to adjust the functionality of the subsequent test based on earlier test or the current test sequence

performance testing: uses large volumes of valid traffic to find the limits of how much traffic a system is able to handle

penetration testing: practical, proactive and authorized use of social attacks, environment attacks, load attacks, automated input validation attacks, data attacks, logic attacks and other relevant attacks to test for vulnerabilities in a system, and to verify the consequence of successful attacks to the assets protected by the system

NOTE: In formal audits, penetration testing should be performed by professionals and experienced penetration testers.

response time: elapsed time from receiving a service request to the beginning of sending the response to the request (ETSI TR 101 577 [i.8])

risk: combination of the consequences of an event with respect to an objective and the associated likelihood of occurrence

risk-based security testing: integration of security risk assessment results in the testing process aiming for systematic guidance, prioritization and optimization of security testing activities

robustness: degree to which a system or component can function correctly in the presence of invalid inputs or stressful environmental conditions (IEEE St. 610.12-1990 [i.2])

robustness testing: sends large volumes of invalid, malformed or otherwise unexpected traffic to the SUT in order to make it fail (ETSI TR 101 590 [i.7])

security requirement: statements about security functions, performance limitations and software reliability for a piece of software, sub-component or system

security test case: set preconditions, inputs (including actions, where applicable), and expected results, developed to determine whether the security features of the SUT have been implemented correctly or to determine whether or not the covered part of the SUT has vulnerabilities that may harm the availability, confidentiality and integrity of the SUT

susceptibility testing: informal penetration test or other type of security review, not necessarily conducted by professionals or experienced penetration testers

system Under Test (SUT): set of hardware and software components constituting the tested object

test-based risk assessment: risk assessment approach that modifies the analysis based on results of security tests

threat: potential for violation of security, which exists when there is a circumstance, capability, action, or event that could cause harm

unknown vulnerability, Zero-day vulnerability: vulnerability that is hidden in software waiting for later discovery and potential exploitation, and which are unknown to the software developer and/or the public

vulnerability: any weakness in software that can be used to cause a failure in the operation of the software

weakness: shortcoming or imperfection in the software code, design, architecture, or deployment that, could, at some point become a vulnerability, or contribute to the introduction of other vulnerabilities (Recommendation ITU-T X.1524 [i.9])

zero-day attack: special form of attack that exploits an unknown vulnerability, and therefore cannot be protected against

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

CC	Common Criteria
CTMF	Conformance Test Methodology and Framework
DAST	Dynamic Application Security Testing
DDoS	Distributed Denial of Service
DoS	Denial of Service
MBST	Model-Based Security Testing
MBT	Model-Based Testing
SAST	Static Application Security Testing
SDLC	System/Software Development Lifecycle
SUT	System Under Test
TOE	Target Of Evaluation
TSFI	TOE Security Functional Interface
TVRA	Threat, Vulnerability and Risk Analysis

4 Introduction to security testing

In software engineering terms, security can be seen as an umbrella activity. The assessment of the security of a system is not a single, stand-alone activity but, rather, takes place at a number of different stages of the System or Software Development Lifecycle (SDLC).

The purpose of security testing is to find weaknesses in software implementation, configuration or deployment. These weaknesses can potentially create or become vulnerabilities in the system. Various security testing techniques are applied at various phases in the product/system lifecycle, starting from requirements definition and analysis and continuing through design, implementation, verification, operations and maintenance (figure 1).

Security tests can be performed in two complementary approaches. Security tests using Static Analysis, also called Static Application Security Testing (SAST), analyse the source code or the binary for security weaknesses without executing it. Problem with SAST tools is the number of **false positives**, indications of security flaws that cannot be triggered to cause security **failures**. Security tests using Dynamic Analysis, or Dynamic Application Security Testing (DAST), execute the code and analyse the behaviour. DAST tools can have **false negatives** caused by bad test design, missing tests that do not trigger the security failures due to bad test coverage or bad choice of tools. In the present document, our focus is in dynamic tests and our use of the term security testing refers to dynamic security testing only.

The actors in the security testing activities include developers, internal testers and external security evaluators. Our focus in this article is in the activities related to internal testing typically performed by internal testers during Verification and Validation (V&V). These activities include:

- Risk Assessment and Risk-based Security Testing (clause 5).
- Functional Testing of Security Features (clause 6).
- Performance Testing (clause 7).
- Robustness Testing (clause 8).
- Penetration Testing (clause 9).

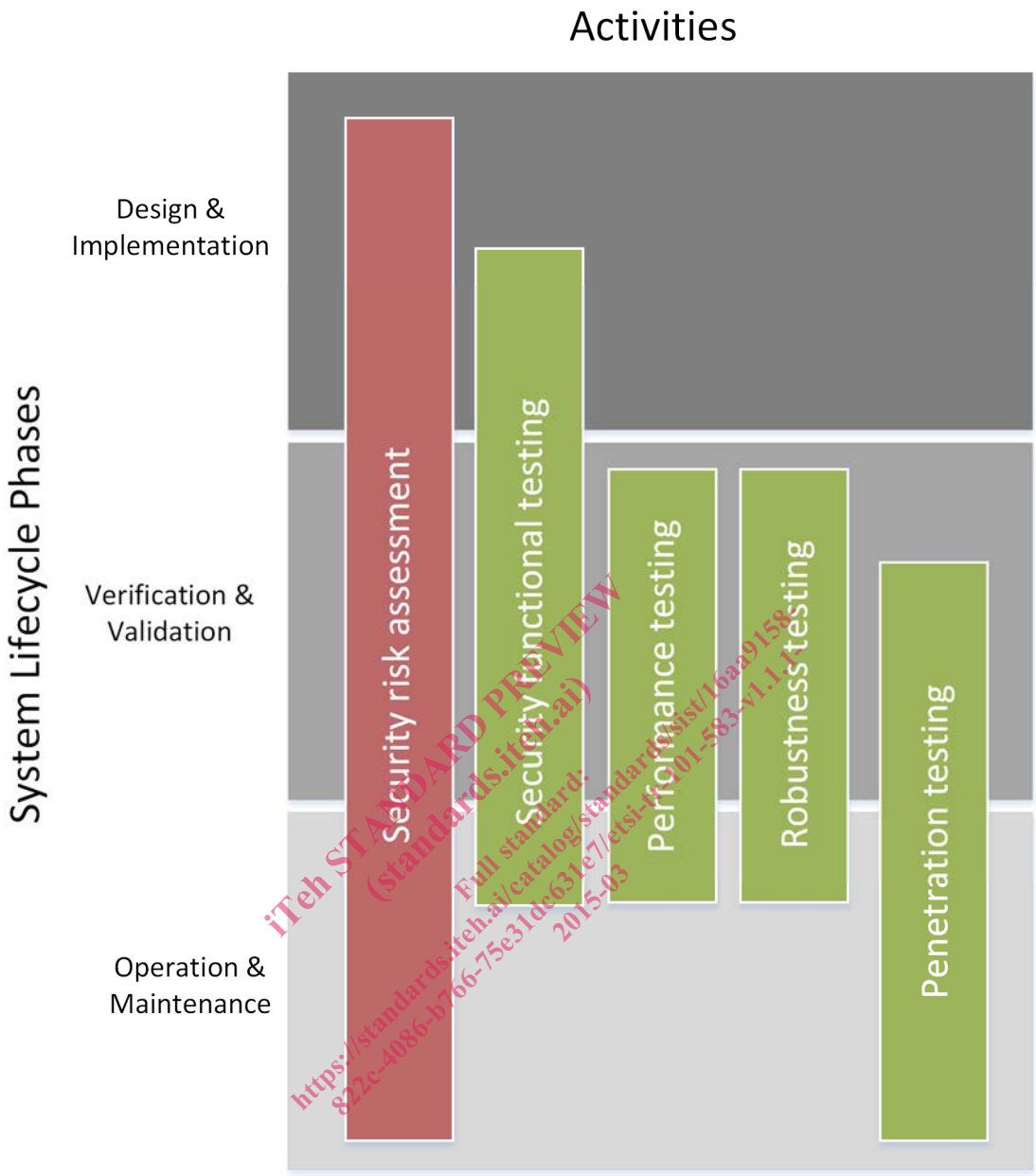

Figure 1: Mapping the security assessment and testing techniques to different phases of the system lifecycle phases

The purpose of security testing is to determine whether a system meets its specified security requirements. The requirements should include statements about security functions, performance limitations and software reliability.

Risk assessment can provide information on potential threats, potential vulnerabilities and the identification of the most critical areas of the system. Moreover the execution of tests can be prioritized based on the risk analysis so that the most relevant test cases are executed with high priority (clause 5).

In penetration testing and third party security audits, additional tests for **attack surface** analysis and scanning for **known vulnerabilities** are used (clause 9).

Several of the testing categories discussed in the present document have other names in different standardization bodies and industry practises. The definition for security testing itself is often limited to only touch the tests focused on security functionality only, and excludes performance and robustness. **Performance testing** can be called stress test or **load test**. **Penetration tests** can be called vulnerability tests or susceptibility tests. **Robustness testing** is often called **fuzzing**.