Preventive Approach for Web Applications Security Testing

Luiz Otávio Duarte
Ferrucio de Franco Rosa
Walcir M. Cardoso Jr.
Renato Archer Information Technology Center
Brazilian Ministry of Science and Technology

+55-19-37466241

OWASP
10/30/2009

Copyright © The OWASP Foundation
Permission is granted to copy, distribute and/or modify this document
under the terms of the OWASP License.

The OWASP Foundation
http://www.owasp.org

Agenda

- Introduction
- Key Points
- Techniques for software testing
- Software Quality
- Web application security testing approach
- Practical demonstration
- Conclusions

Agenda

- Introduction
- Key Points
- Techniques for software testing
- Software Quality
- Web application security testing approach
- Practical demonstration
- Conclusions

Motivation

- The concern with security in developing projects for the Web is essential for the protection of information and business continuity.

- In recent years the Web has grown significantly and complexity of applications and services even more. New technologies, tools and architectures are adopted without properly concern for security.

Statistics data

@Risk/SANS Statistics

- SysAdmin, Audit, Network, Security Institute maintains @RISK a weekly vulnerability consensus digest.

NVD/NIST Statistics

- National Vulnerability Database - repository of vulnerability resources (USA).

OSVDB Statistics

Top 10 vulnerabilities by OWASP

Statistics data

@Risk/SANS Statistics

NVD/NIST Statistics

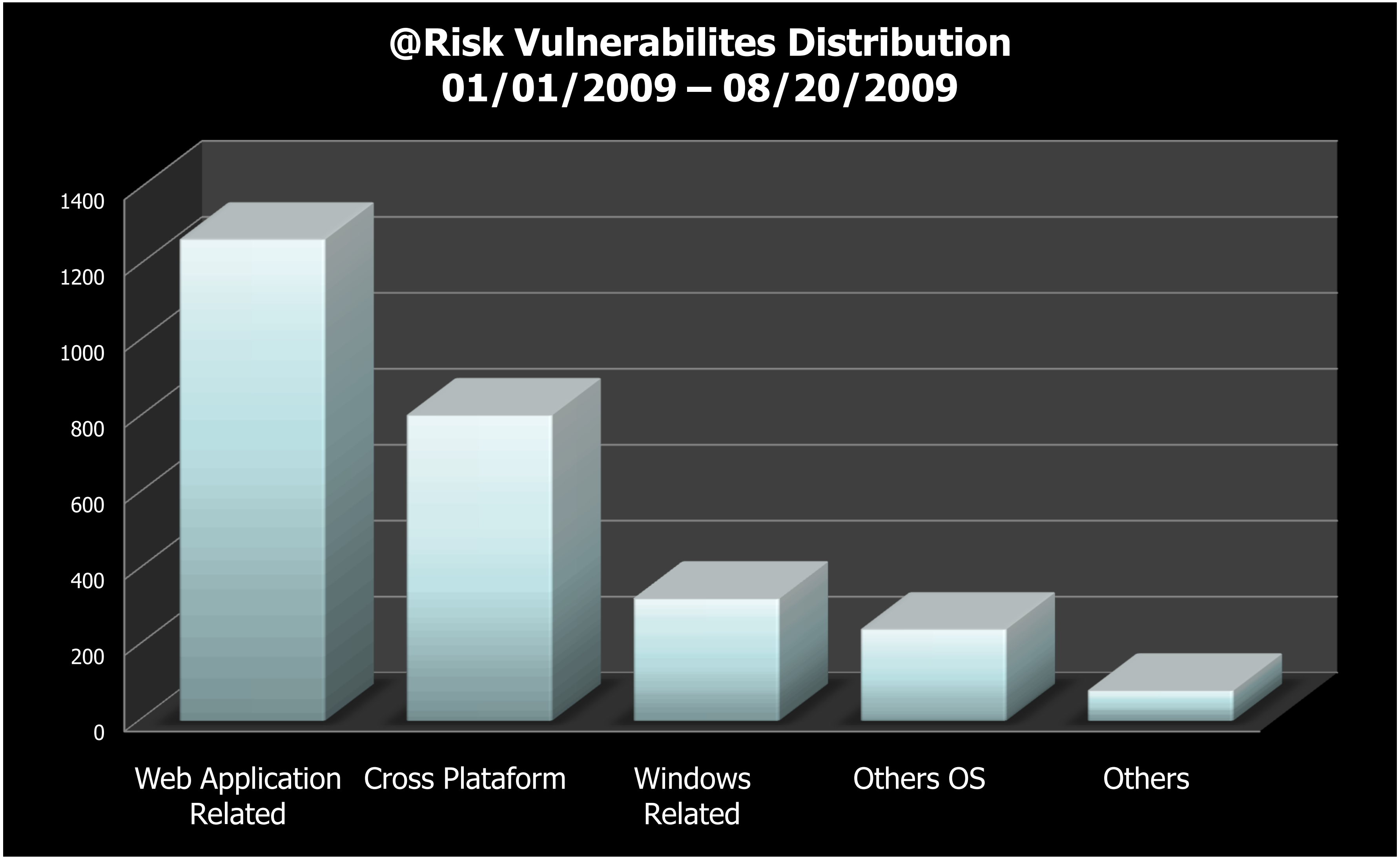
OSVDB Statistics

- Open Source Vulnerability Database, that aims to provide accurate, detailed, current, and unbiased technical information on security vulnerabilities.

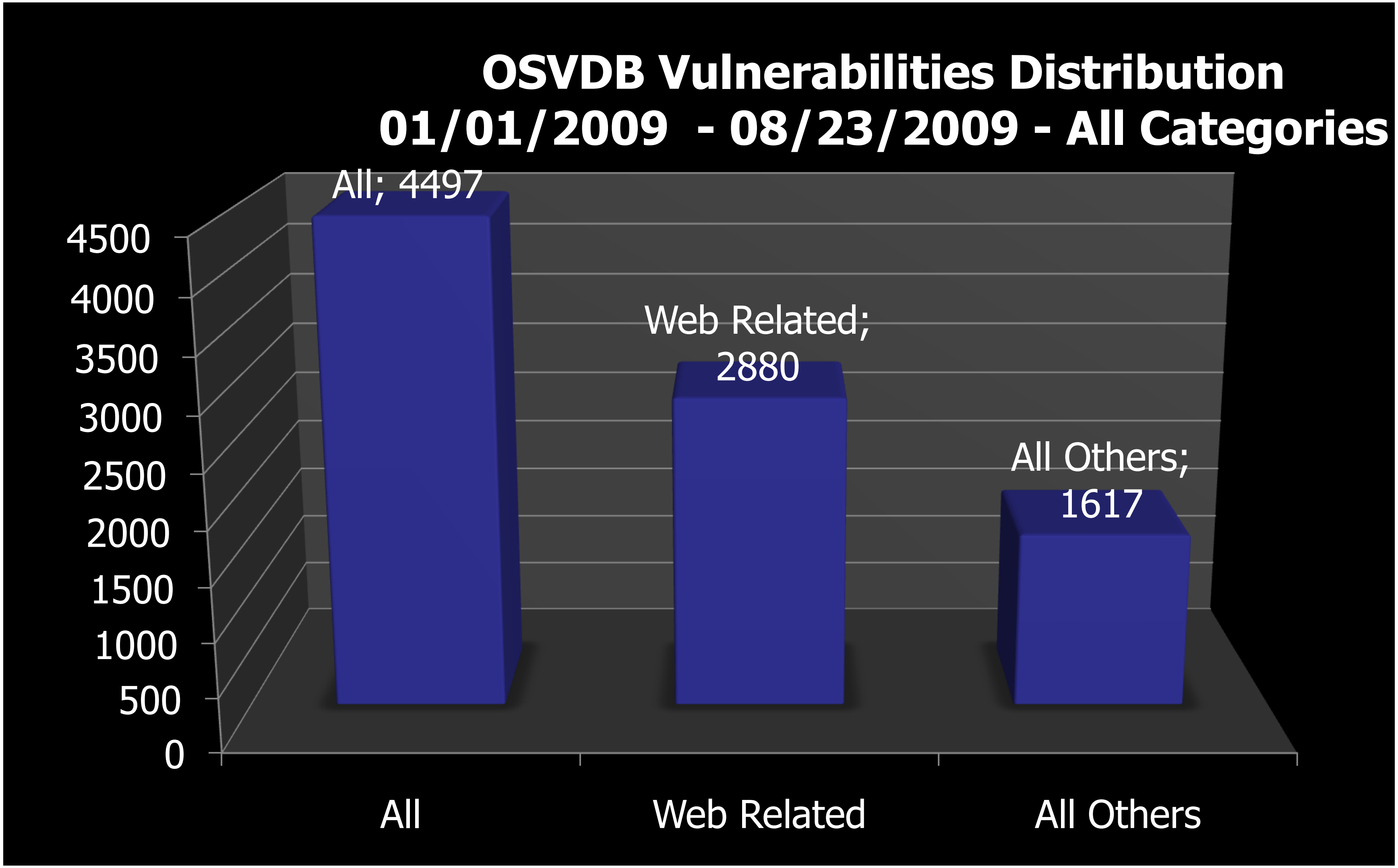
Top 10 vulnerabilities by OWASP

- Open Web Application Security Project, provides the top 10 most critical web application vulnerabilities.

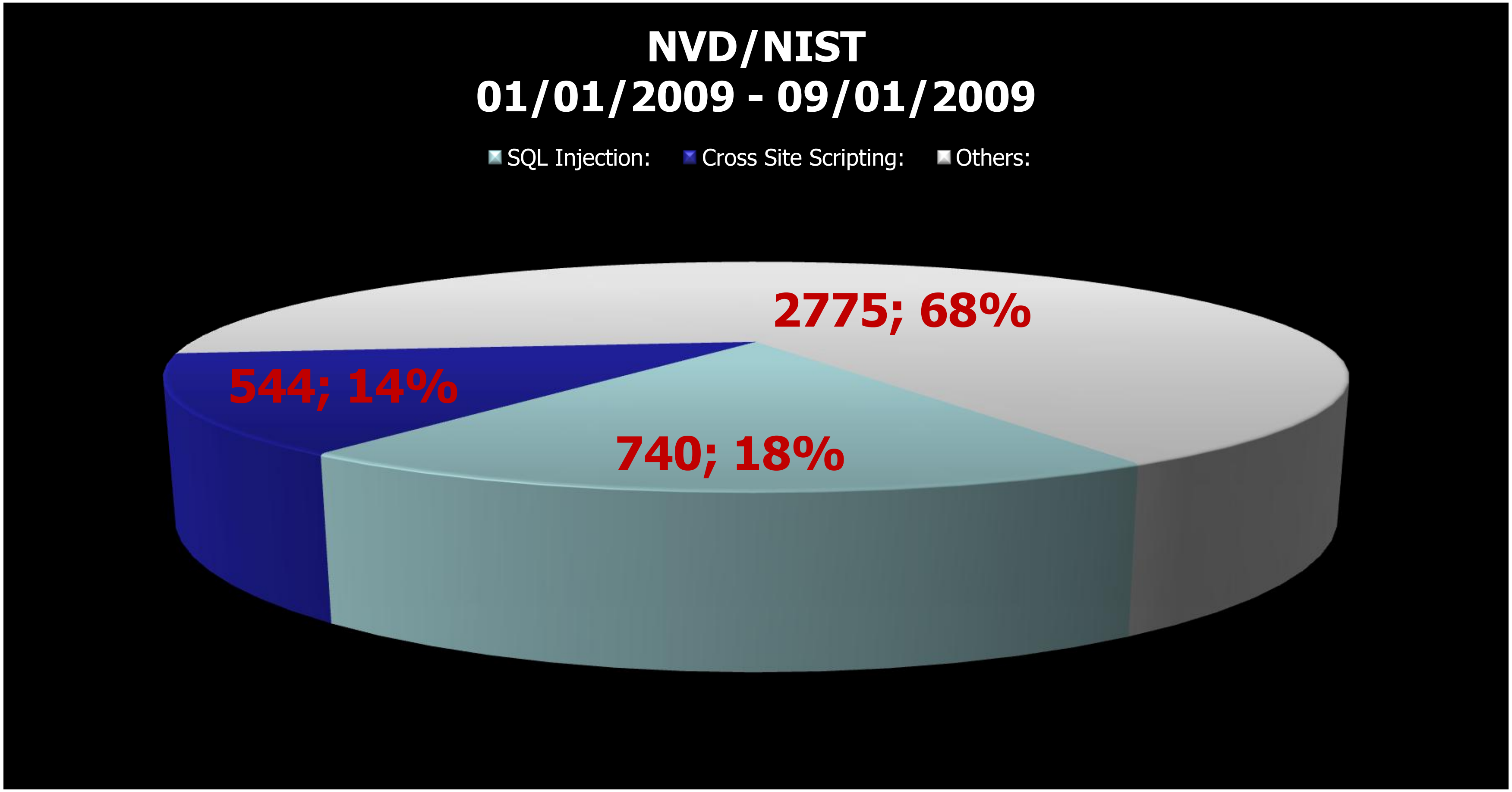
@Risk/SANS


@Risk Vulnerabilites Distribution
01/01/2009 – 08/20/2009
1400
1200
1000
800
600
400
200
0
Web Application Related
Cross Plataform
Windows Related
Others OS
Others

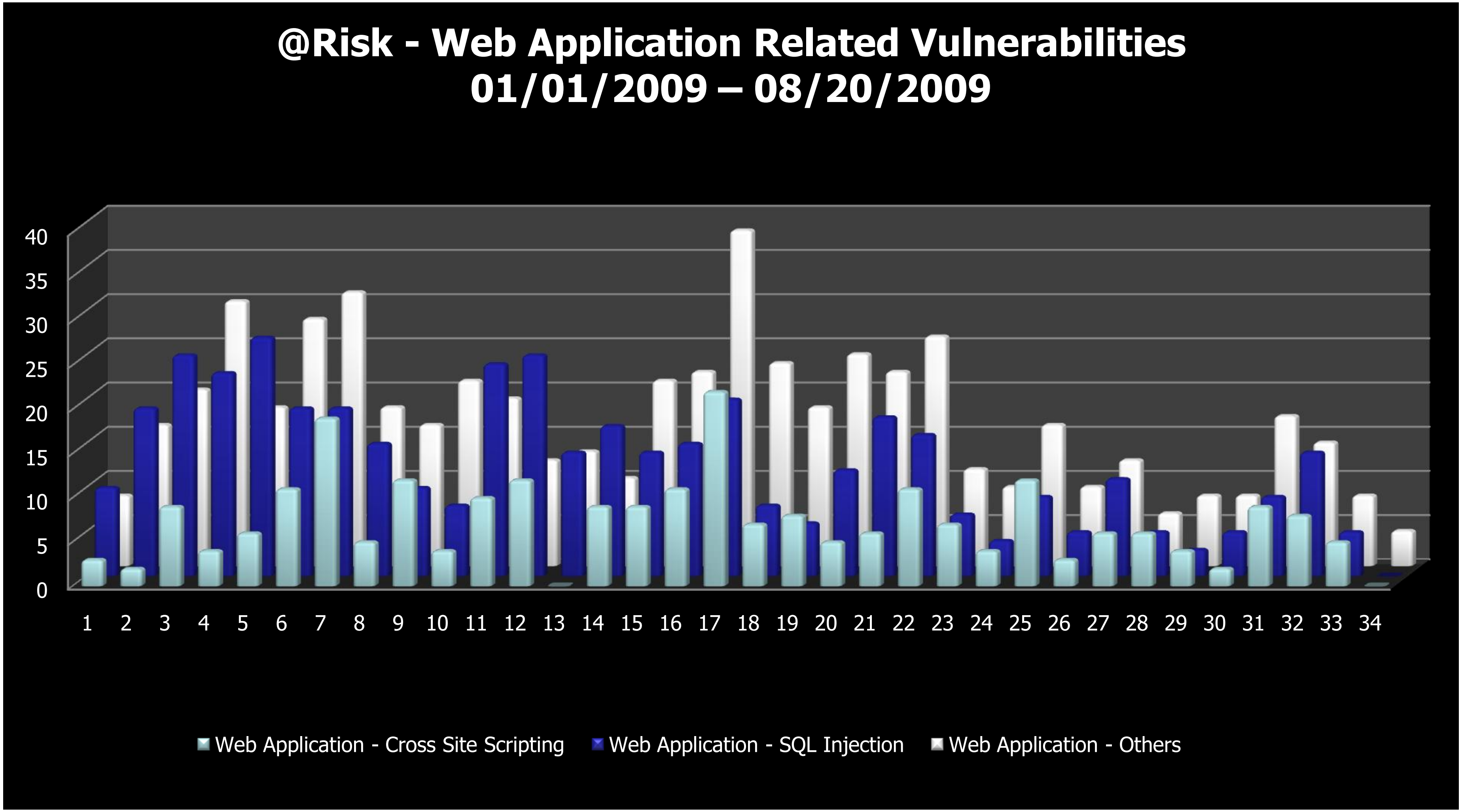
Statistics data – OSVDB



NVD/NIST


NVD/NIST
01/01/2009 - 09/01/2009
SQL Injection:
Cross Site Scripting:
Others:
2775; 68%
544; 14%
740; 18%

@Risk/SANS


@Risk - Web Application Related Vulnerabilities
01/01/2009 – 08/20/2009
0
5
10
15
20
25
30
35
40
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34
Web Application - Cross Site Scripting
Web Application - SQL Injection
Web Application - Others

OWASP- Top 10 Web app. vulns. for 2007

- A1 - Cross Site Scripting (XSS)
- A2 - Injection Flaws
- A3 - Malicious File Execution
- A4 - Insecure Direct Object Reference
- A5 - Cross Site Request Forgery (CSRF)
- A6 - Information Leakage and Improper Error Handling
- A7 - Broken Authentication and Session Management
- A8 - Insecure Cryptographic Storage
- A9 - Insecure Communications
- A10 - Failure to Restrict URL Access

Aspects of the problem

Developers/Architects are making the same kind of mistakes.

Many of software development life cycles (SDLC) do not provide an effectively approach for security aspects.

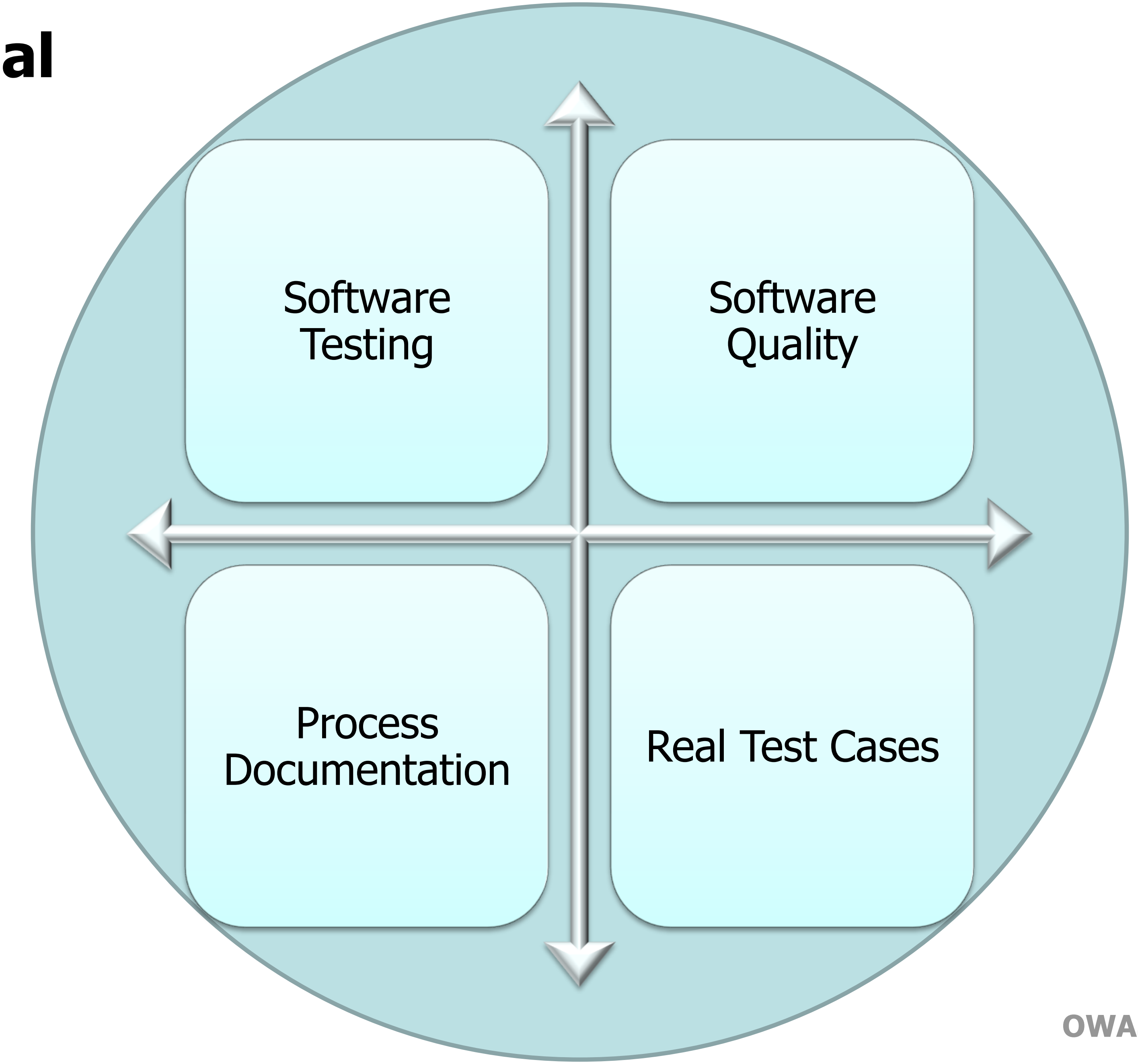
How to spot web vulnerabilities and the correct then are well known, but repairing could be expensive.

Applications are not properly tested in terms of security.

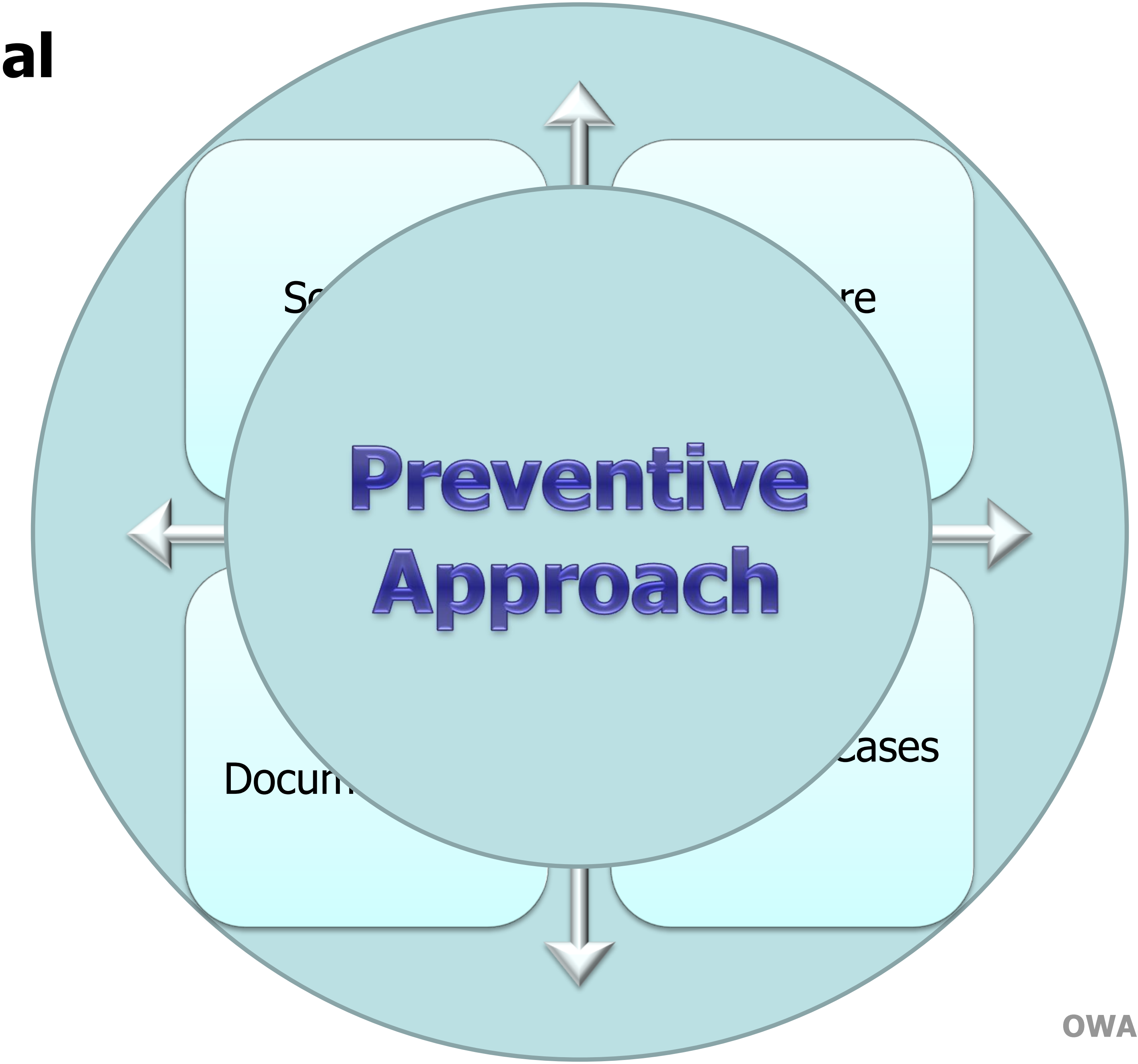
Lack of a *de facto* standard or approach to conduct a good software security testing for web applications.

Know-how on penetration test is far from a good software security testing.

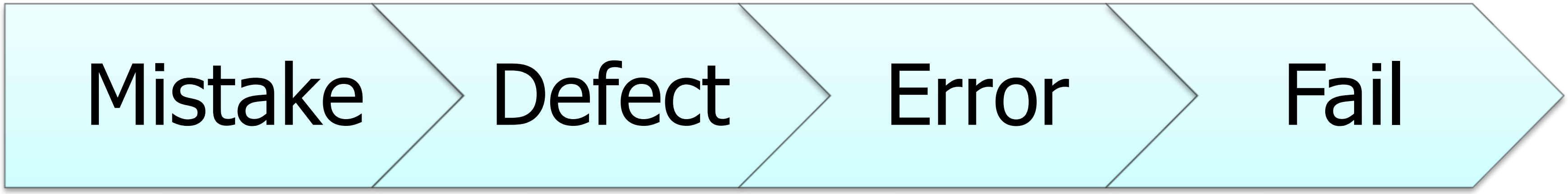
Goal


Software Testing
Software Quality
Process Documentation
Real Test Cases

Goal



Initial Concepts



Mistake → Defect → Error → Fail

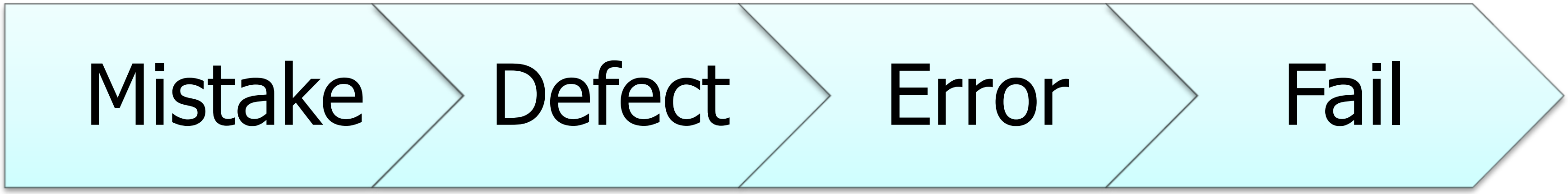
Mistake

- A human action that can introduce some kind of defect in the software.

Defect, Fault

- It is a deficiency (step, process or incorrect definition) resident in the software. The defect when executed can cause an error.

Initial Concepts



Mistake Defect Error Fail

Error

- It is caused by the execution of a defect and is characterized by an unexpected or inconsistent program state. It is the incorrect value in a given state of the program.

Fail

- It is an observable event that the software infringed its specification. It is the noticed spread of an error.

Initial Concepts

Vulnerability

- A weakness in a computing system which can be exploited and harmed by one or more threats. Making the system do what it was not designed to do.

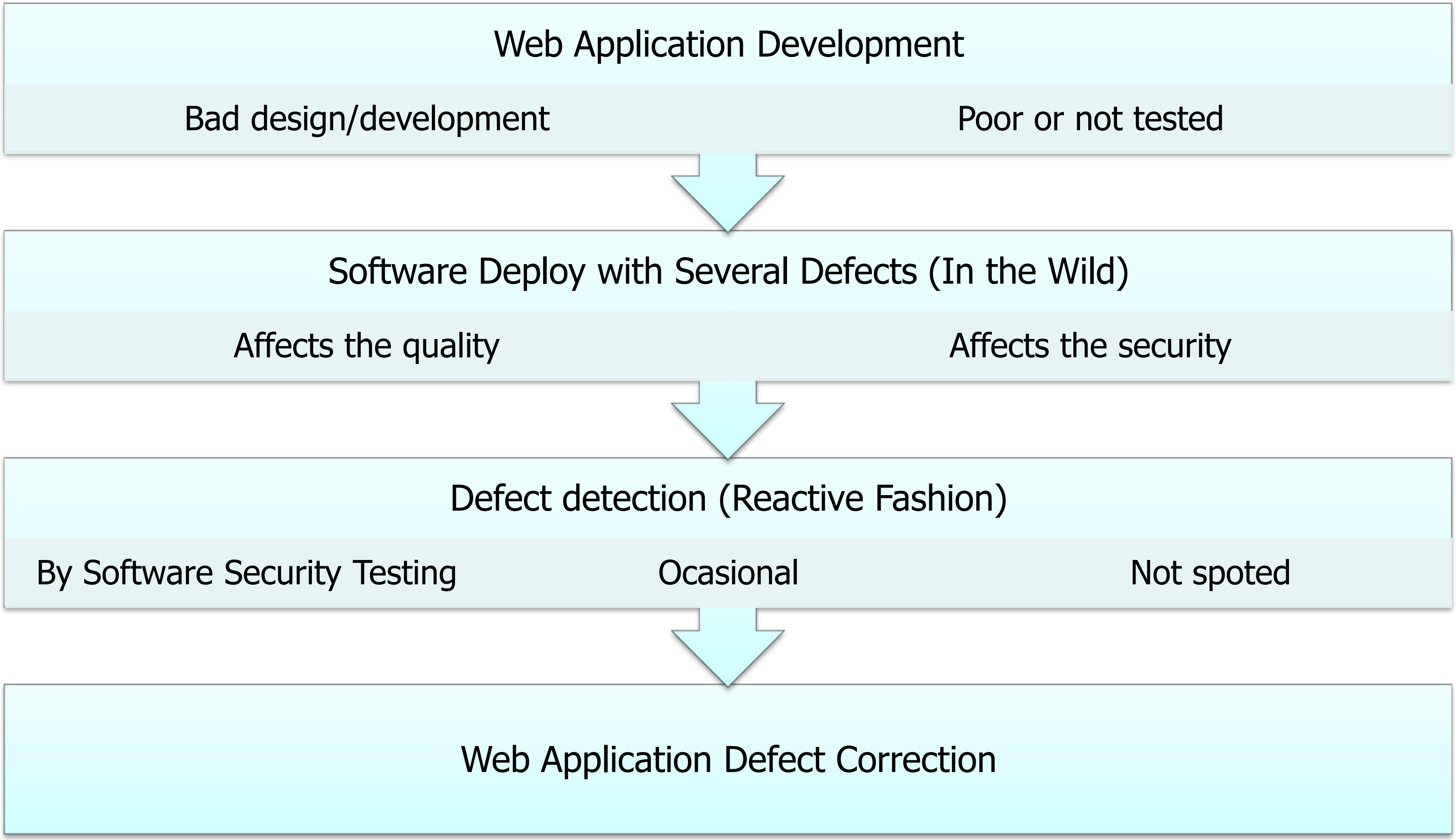
Web Application (In this presentation)

- Is an application coded in a browser-supported language that is acessed via a web browser over a network such as Internet. Some times we will use webapp as a synonymous of Web Application.

Agenda

- Introduction
- Key Points
- Techniques for software testing
- Software Quality
- Web application security testing approach
- Practical demonstration
- Conclusions

Web Application Vulnerabilities Life-Cicle



Web Application Development

Bad design/development

Poor or not tested

Software Deploy with Several Defects (In the Wild)

Affects the quality

Affects the security

Defect detection (Reactive Fashion)

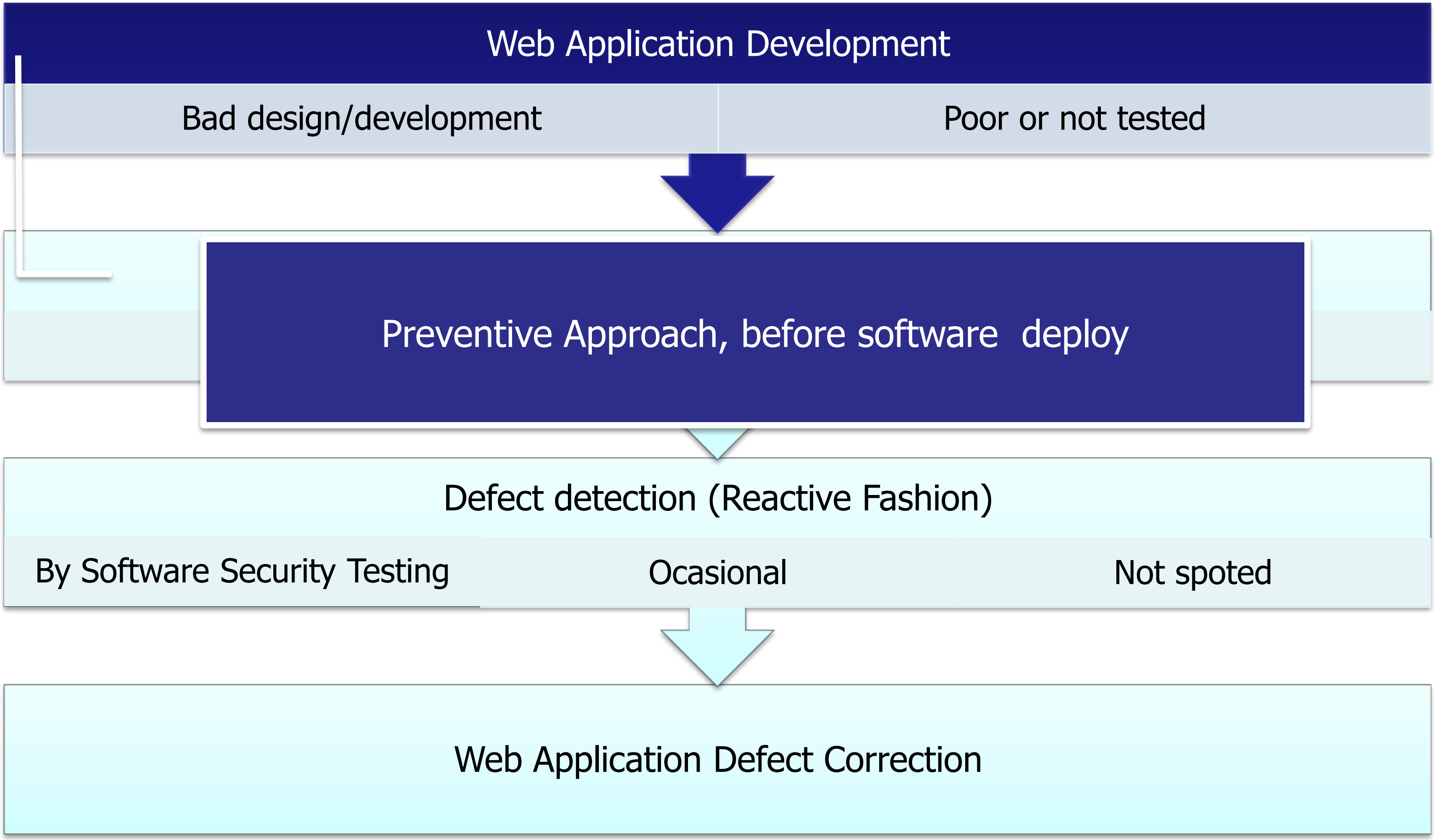
By Software Security Testing

Ocasional

Not spoted

Web Application Defect Correction

Web Application Vulnerabilities Life-Cicle



Web Application Development

Bad design/development

Poor or not tested

Preventive Approach, before software deploy

Defect detection (Reactive Fashion)

By Software Security Testing

Ocasional

Not spoted

Web Application Defect Correction

Preventive Approaches – Line Fronts

Secure Development

- Secure Design
- Secure Coding
- Education

Software Security Testing

- Structural
- Functional

Preventive Approaches – Line Fronts

Secure Development

- Secure Design
- Secure Coding
- Education

Software Security Testing

- Structural
- Functional

Agenda

- Introduction
- Key Points
- Techniques for software testing
- Software Quality
- Web application security testing approach
- Practical demonstration
- Conclusions

Software Testing

Software testing is the process of running the software in a **controlled** way to evaluate whether it **behaves as specified**.

The main goal of software testing is to **detect software failures** so **defects may be uncovered and corrected**.

Software Testing - Aspects

A fundamental software testing problem is that testing **all possible test cases** is not feasible. (combinatorial explosion)

The solution is the definition of a **good test criterion**, that selects a reduced set of test cases that have high probability to find defects.

There are several criteria types and objectives:

- Test criteria;
- Test case selection criteria;
- Adequacy criteria;
- …

Software Testing

We test the software only partially?

- Yes.
- We can not conclude that the software is free from defects, even after running several test cases.
- The cost of testing to ensure that software has no defects can overcome the project budget.

We should plan what to test:

- The targets must be critical points, those most vulnerable, more susceptible to defects, which offer greater risk to the business.

Technical aspects of software testing

What to test – Types of testing

- Functionality Testing
- Interface Testing
- Stress Testing
- Usability Testing
- Security Testing
- ...

How to test – Test Criteria

When to test - SDLC

Technical aspects of software testing

What to test – Types of testing

How to test – Test Criteria

- Functional Testing
  - Equivalence Partition
  - Boundary Value Analysis
  - …
- Structural Testing
  - Branch Testing
  - Condition Testing
  - Function Testing
  - …

When to test - SDLC

Technical aspects of software testing

What to test – Types of testing

How to test – Test Criteria

When to test - SDLC

- Unit Testing
- Integration/Module Testing
- System Testing / Validation Testing
- Acceptance Testing
- Regression Testing

Software Test Documentation – IEEE STD 829-2008

Defines several documents to make the testing more reliable.

- Test Plan;
- Test Design Specification;
- Test Case Specification;
- Test Procedure Specification;
- Test Item Transmittal Report;
- Test Log;
- Test Incident Report;
- Test Summary Report.

Software Test Documentation – IEEE STD 829-2008

Defines several documents to make the testing more reliable

Those documents can introduce unnecessary overhead to testing.

We can use a different and more effective approach to deal with documentation.

Agenda

- Introduction
- Key Points
- Techniques for software testing
- Software Quality
- Web application security testing approach
- Practical demonstration
- Conclusions

Software Quality

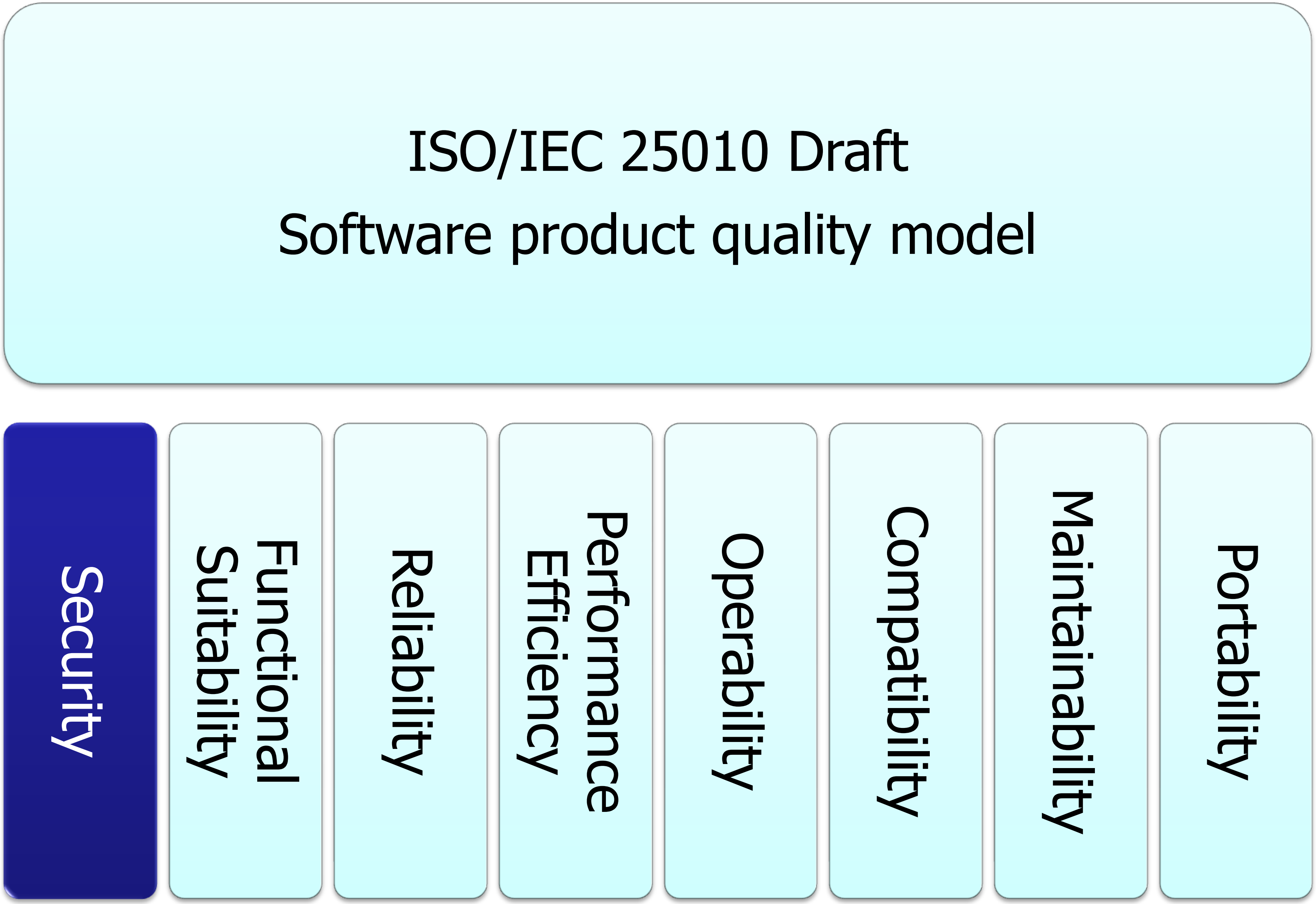
- Is the degree in which the software product meets **stated and implied needs** when used under specified conditions.

ISO/IEC 25010 Draft

- Software product Quality Requirements and Evaluation (SQuaRE) – Quality models for software product quality and system quality in use.

ISO/IEC 14598

- Information technology - Software product evaluation - Part 1: General overview.
- Software product evaluation - Part 5: Process for evaluators.



ISO/IEC 25010 Draft

Software product quality model

Security

Functional Suitability

Reliability

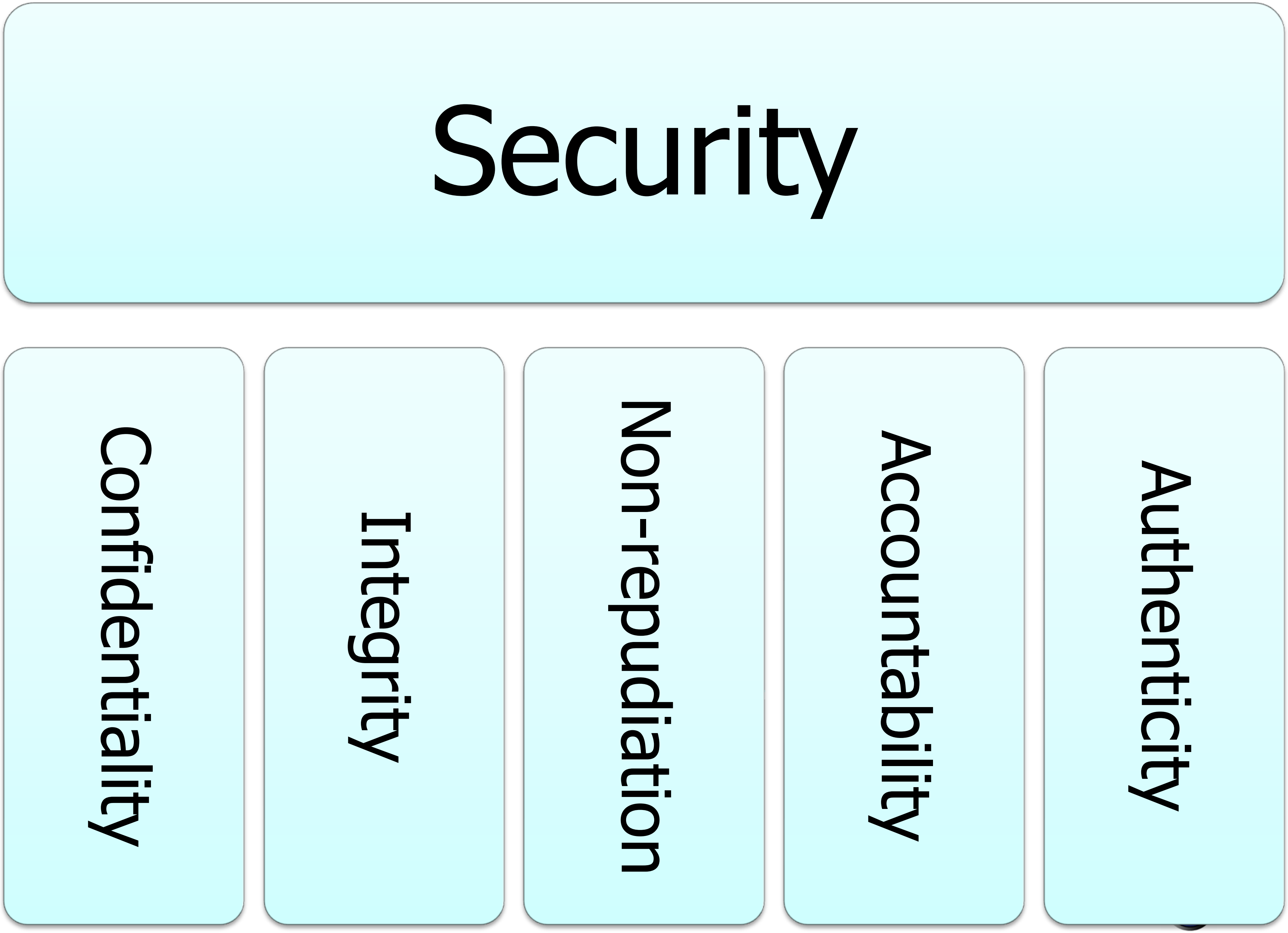
Performance Efficiency

Operability

Compatibility

Maintainability

Portability



Security

Confidentiality

Integrity

Non-repudiation

Accountability

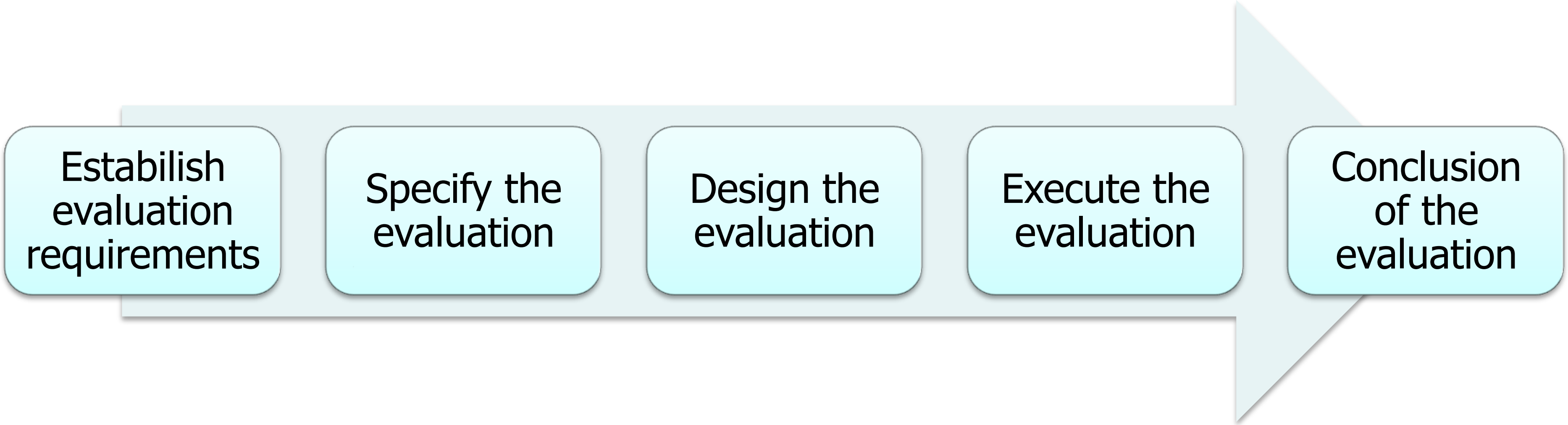
Authenticity

ISO/IEC 14598-1 (ISO/IEC 14598-5)

- Provides basis for a systematic evaluation process and general software quality.

Establishes evaluation process, such as:

- Repeatability / Reproducibility / Impartiality / Objectivity (RRIO).



Estabilish evaluation requirements

Specify the evaluation

Design the evaluation

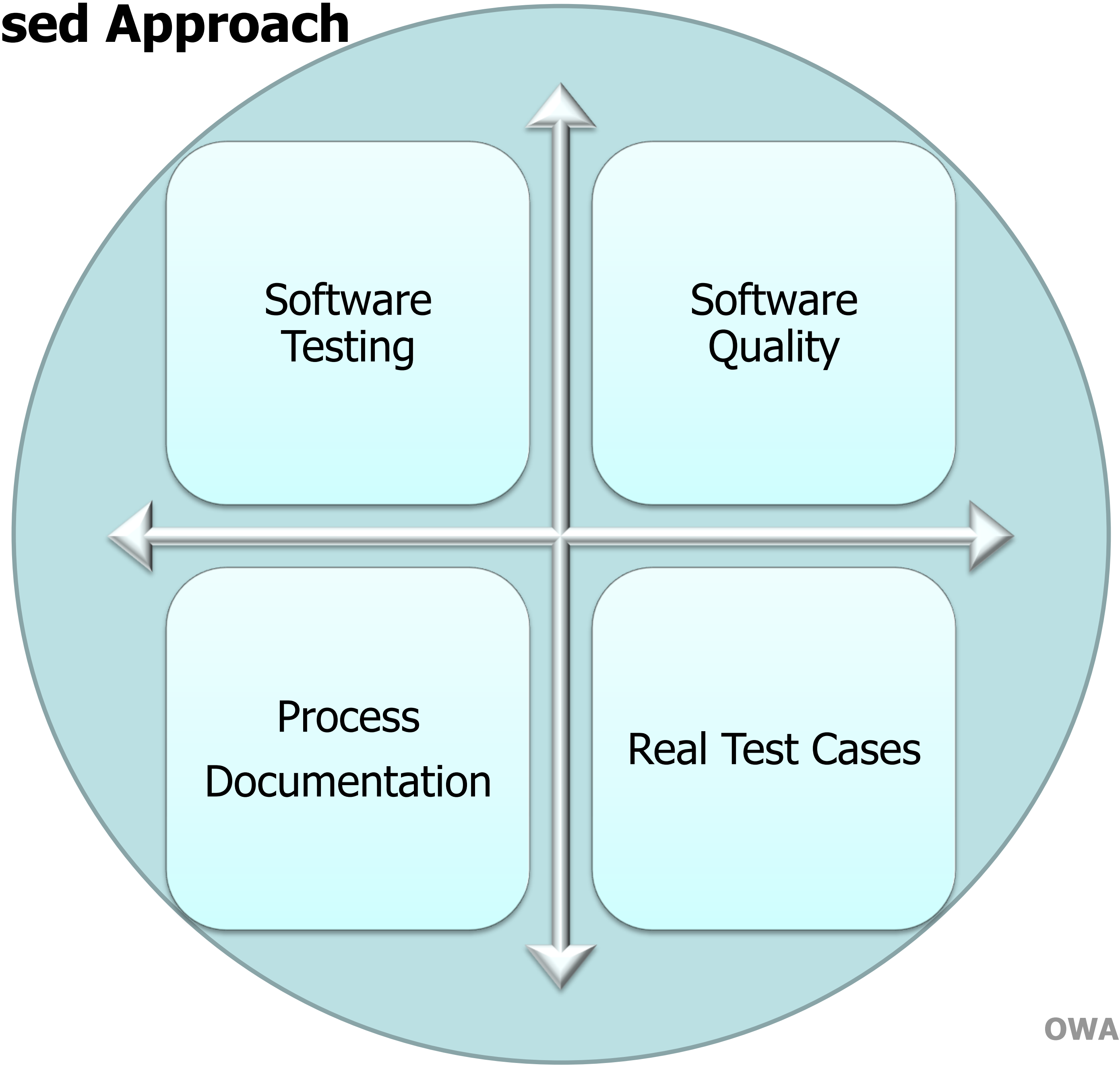
Execute the evaluation

Conclusion of the evaluation

Agenda

- Introduction
- Key Points
- Techniques for software testing
- Software Quality
- Web application security testing approach
- Practical demonstration
- Conclusions

Proposed Approach



Software Testing

Software Quality

Process Documentation

Real Test Cases

Proposed Approach

Software Testing - Security

- Will define the web application security testing criteria;
- Will define the test case selection criteria;
- Will define the adequacy criteria.

Software Quality – Security

- The evaluation phases are extends to software security testing, defining **The process**:
  - Establish security testing requirements;
  - Especify the security testing;
  - Design the security testing;
  - Execute the security testing;
  - Conclusion of the security testing.
- Based on ISO/IEC 14598-5:

Proposed Approach

Process Documentation – Security

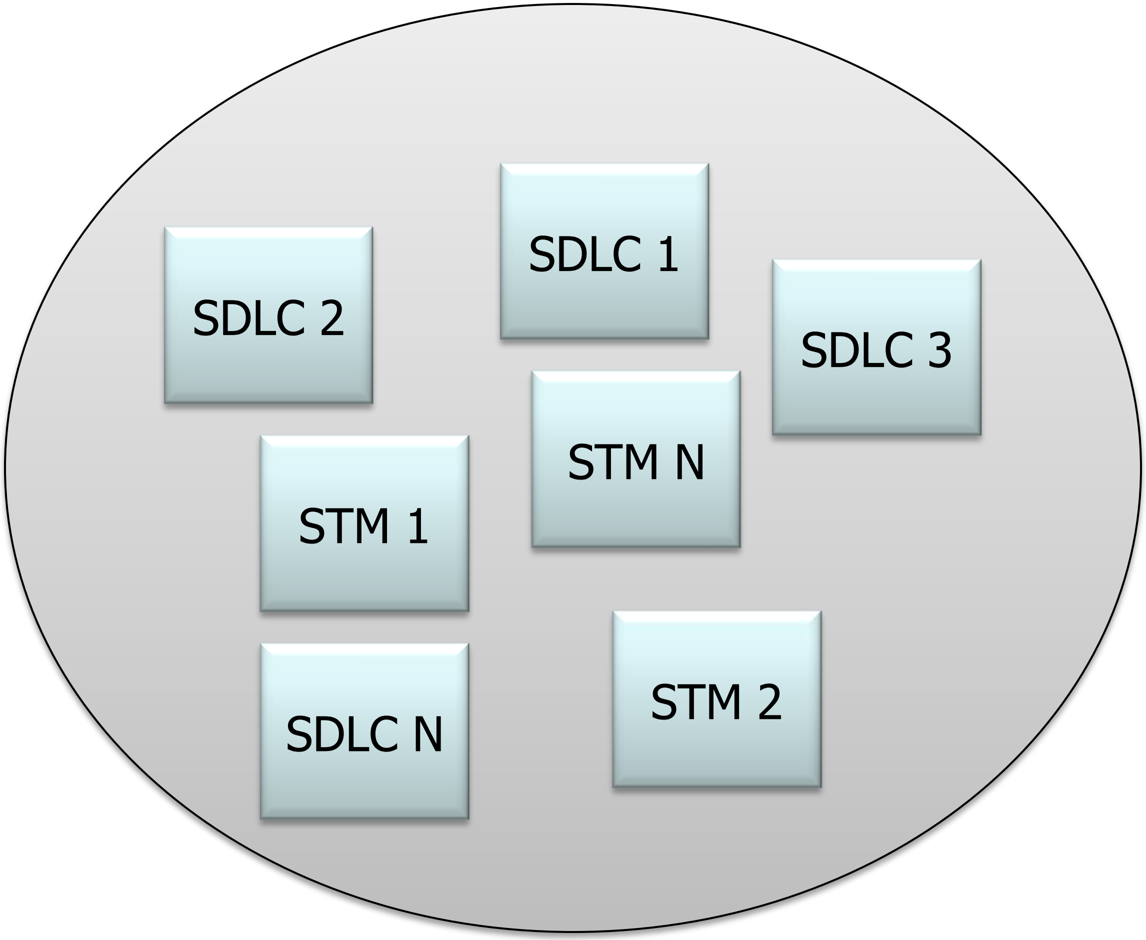
- When not based on IEEE 829-2008, is based on the documentation of each ISO/IEC 14598-5 evaluation phase.

Real Test Cases – Security

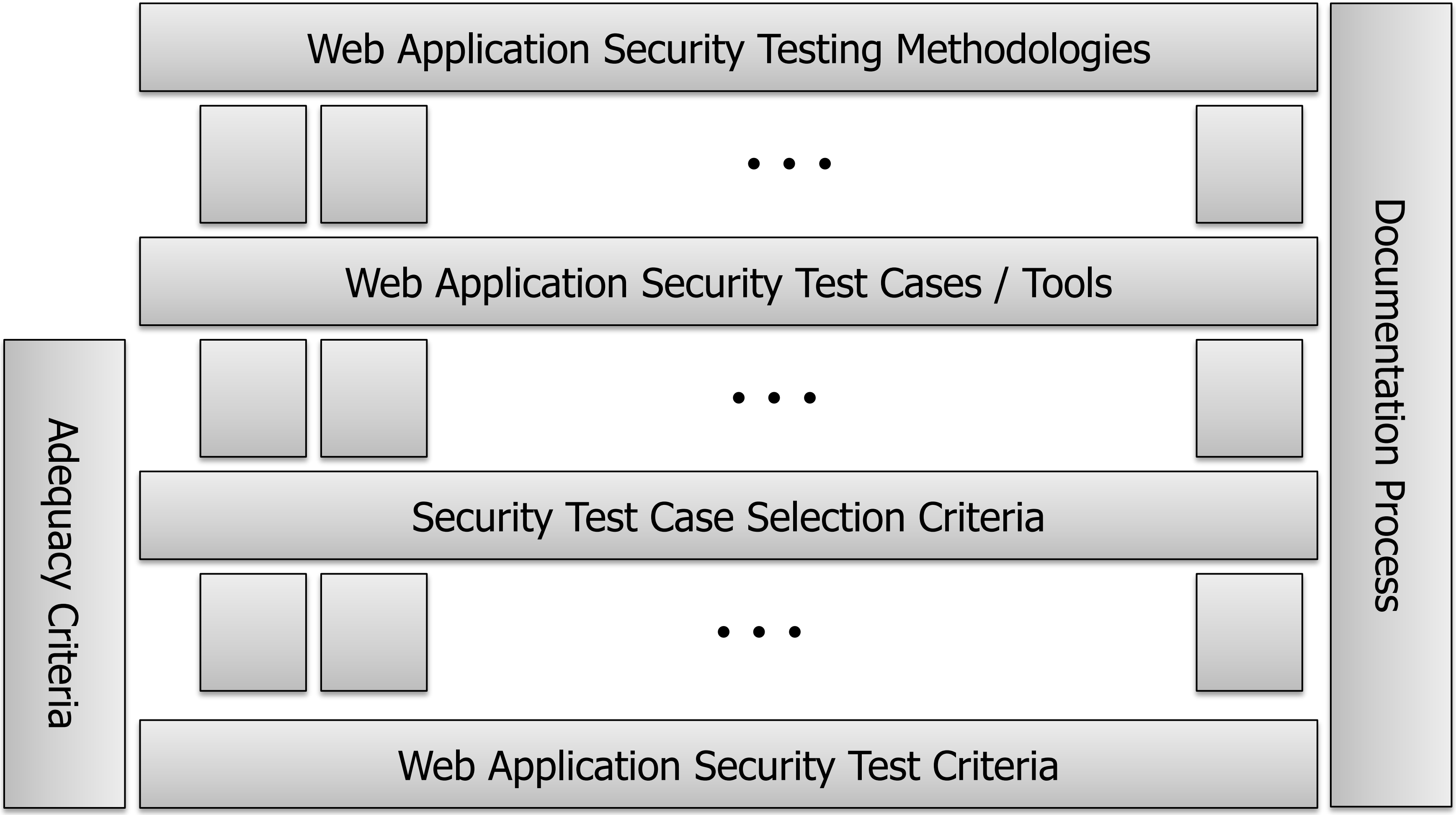
- The easiest part for security guys. We have several test cases to identify different types of vulnerabilities.

Web Application Security Testing Approach

Aims to be SDLC and STM independent.

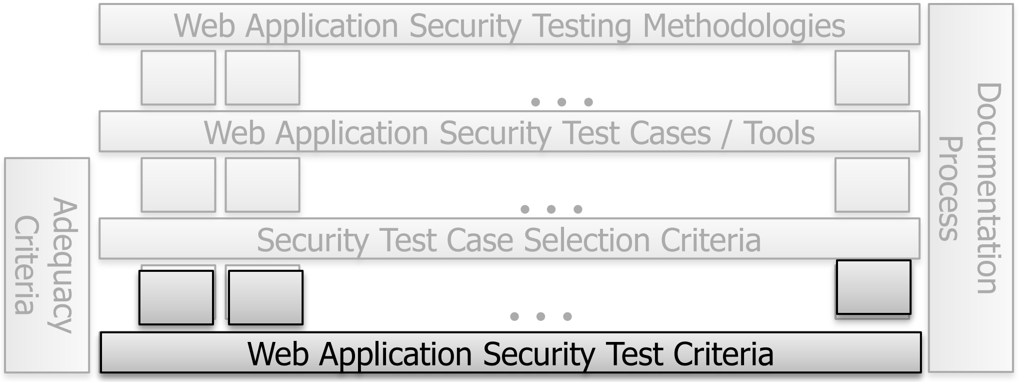


Web Application Security Testing Approach


Web Application Security Testing Methodologies
. . .
Web Application Security Test Cases / Tools
. . .
Security Test Case Selection Criteria
. . .
Web Application Security Test Criteria
Adequacy Criteria
Documentation Process

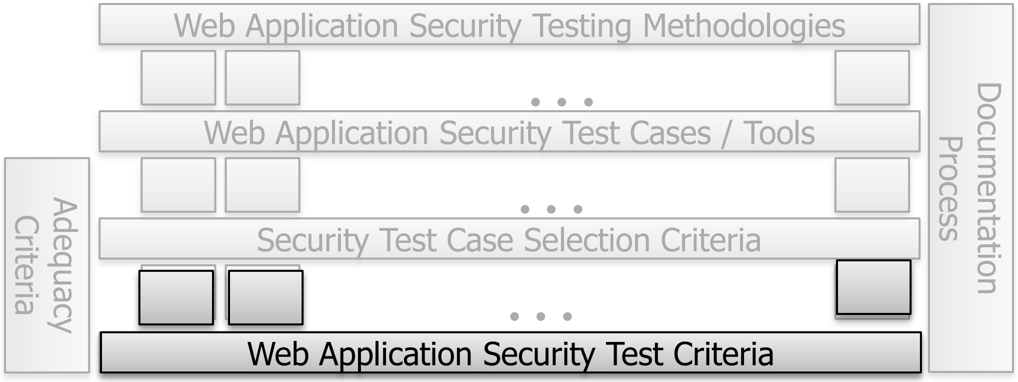
A web application security testing criterion

- The webapp security testing criterion will define what is the **prioritization of security control or threat** that must be **exercised in the testing**, based on **security requirements**. Furthermore will establish security metrics for testing.
- What security testing criterion can we use?
  - Web Application **Risk Analisys**?
  - Web Application **Threat Modeling**?
  - Web Application **Security Use Cases**?
  - ...


Web Application Security Testing Methodologies
. . .
Web Application Security Test Cases / Tools
. . .
Security Test Case Selection Criteria
. . .
Web Application Security Test Criteria
Adequacy Criteria
Documentation Process

A web application security testing criterion

- Almost all security test case will cause **an abnormal behavior** in the structure under testing. The **oracle** must be able to determine if this abnormal behavior is related or not with **some kind of security flaw.**
- If the system's behavior is "normal“ then, either the flaw was not spotted or does not exists.

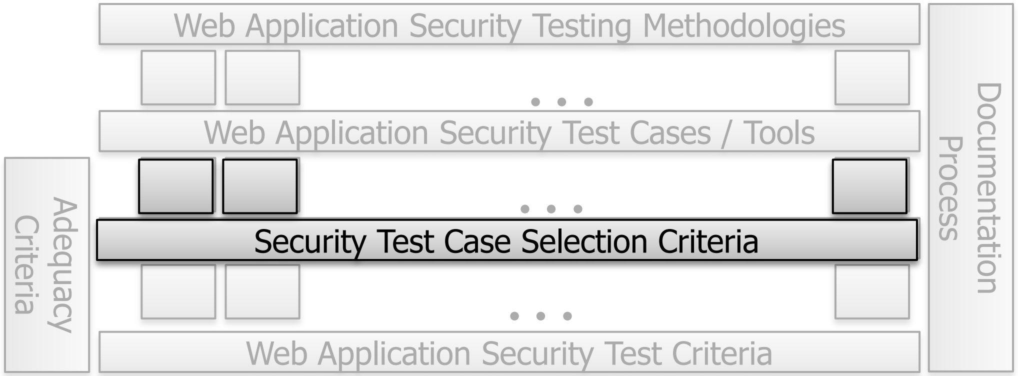

Web Application Security Testing Methodologies
. . .
Web Application Security Test Cases / Tools
. . .
Security Test Case Selection Criteria
. . .
Web Application Security Test Criteria
Adequacy Criteria
Documentation Process

A web application security testing criterion

- Is any of those webapp security testing criteria a valid criterion?
  - All of those criteria, in addition to an adequate test case set, have the ability to identify security flaws presence. But we have not yet been able to prove that all flaws (defined by the criterion) can be exercised.
- Once established the webapp security testing criterion, we must determine the set of case tests. This is the test case select criterion.

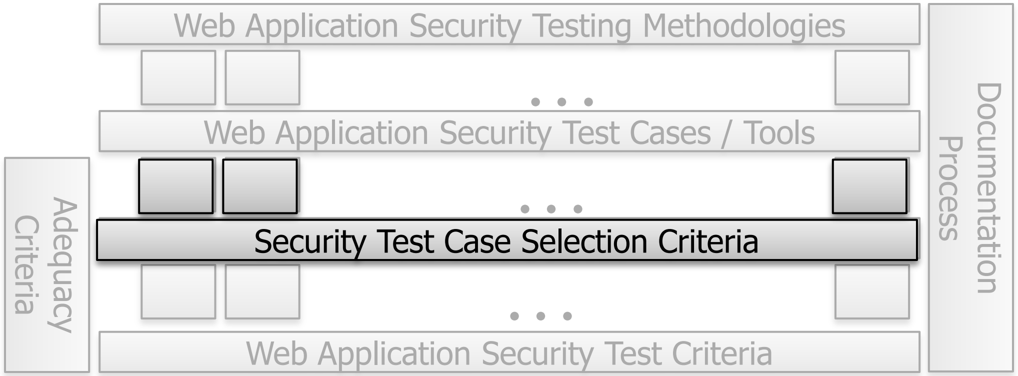
A security test case selection criterion

- We already know what kind of security control or threat to prioritize, now we need to determine a criterion to select good test cases.
- How to define the test data selection criterion?
  - How to select properly "real test cases" and automated test case generation tools, so those threats and security controls can be well exercised?


Web Application Security Testing Methodologies
. . .
Web Application Security Test Cases / Tools
. . .
Security Test Case Selection Criteria
. . .
Web Application Security Test Criteria
Adequacy Criteria
Documentation Process

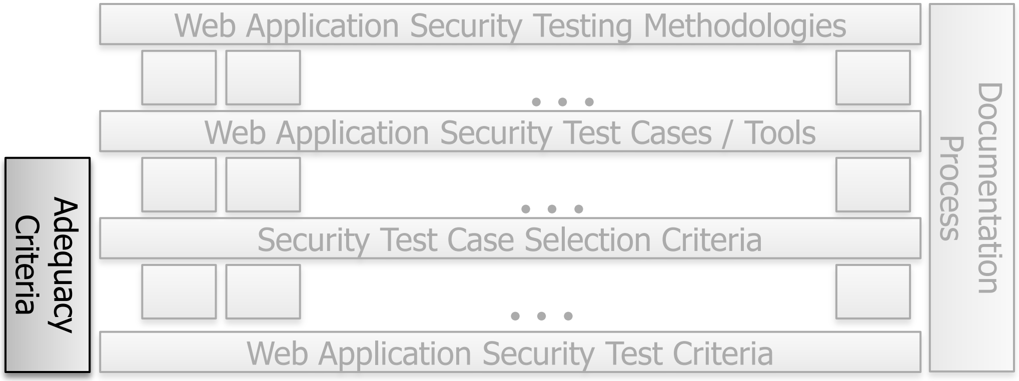
A security test case selection criterion

- We can:
  - Derive vulnerabilities from threat;
  - Derive attack trees from security use cases or risk analysis, defining vulnerabilities;
- The aim is to reduce the testing efforts to the identified vulnerabilities that could be exploited by threats or could be present on security controls.


Web Application Security Testing Methodologies
. . .
Web Application Security Test Cases / Tools
. . .
Security Test Case Selection Criteria
. . .
Web Application Security Test Criteria
Adequacy Criteria
Documentation Process

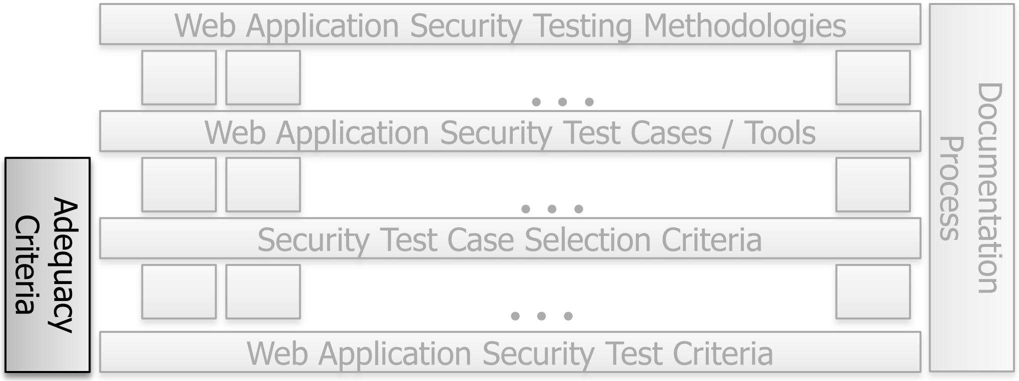
The Adequacy Criterion

- The adequacy criterion is used to determine whether a "program" has been tested "enough".
- It is based on the evaluation of the security test case set: the set is adequate if and only if it satisfies each sequence defined by the webapp security testing criterion.
- In other words: is the confrontation of test case set and webapp security test criterion requirements.


Web Application Security Testing Methodologies
. . .
Web Application Security Test Cases / Tools
. . .
Security Test Case Selection Criteria
. . .
Web Application Security Test Criteria
Adequacy Criteria
Documentation Process

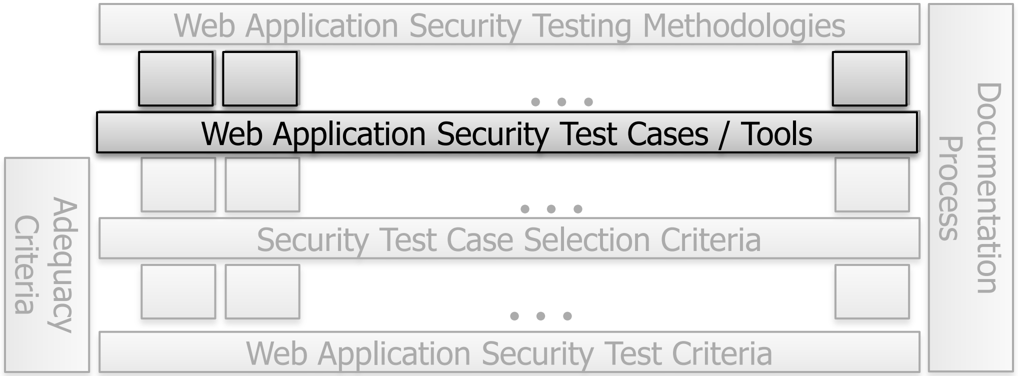
The Adequacy Criterion

- It is a hard work to state the adequacy of a selected set of security test case. But we can estimate the coverage test:
  - Were all attack surface tested?
  - Were all defined threat tested?
  - Were all defined security controls tested?
  - Were all defined vulnerabilities spottable?
  - Were all attach tree tested?
  - …


Web Application Security Testing Methodologies
. . .
Web Application Security Test Cases / Tools
. . .
Security Test Case Selection Criteria
. . .
Web Application Security Test Criteria
Adequacy Criteria
Documentation Process

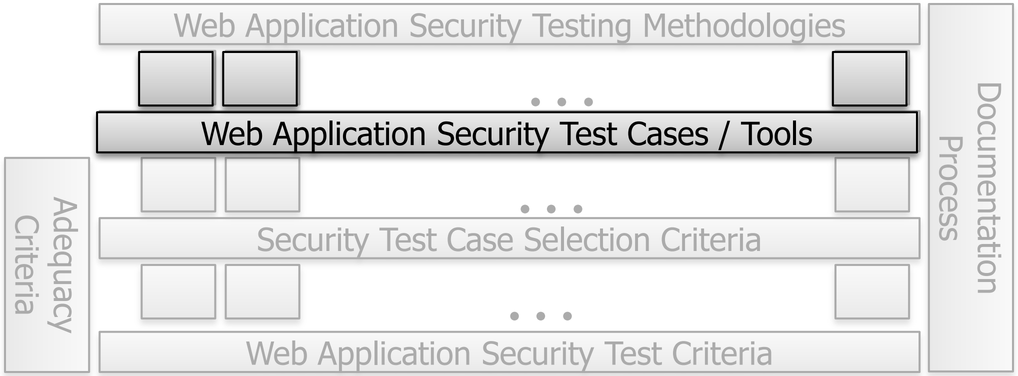
Web Application Security Test Case / Tools

- We select the security test case and tools based on the security test case criterion, because on the security field we have many ways/techniques and tools that helps the process to spot a vulnerability.
- What we need to do is to generate test cases from those huge quantity of ways/techniques and tools.
  - Remember the testing must be RRIO.
  - We can use pentest techniques, but in a structured way.


Web Application Security Testing Methodologies
. . .
Web Application Security Test Cases / Tools
. . .
Security Test Case Selection Criteria
. . .
Web Application Security Test Criteria
Adequacy Criteria
Documentation Process

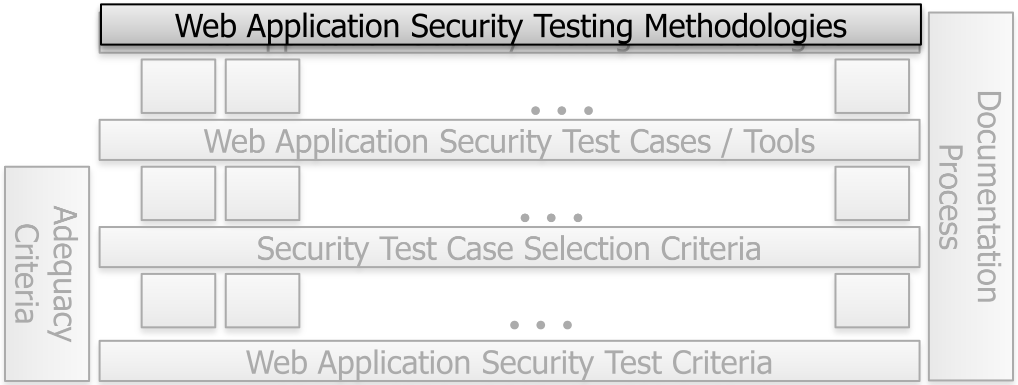
Web Application Security Test Case / Tools

- So now it is time to use the security skills to generate test cases. Who can save us?
  - OSSTMM;
  - OWASP;
  - Pentesting techniques;
  - Exploitation techniques;
  - Security testing tools.

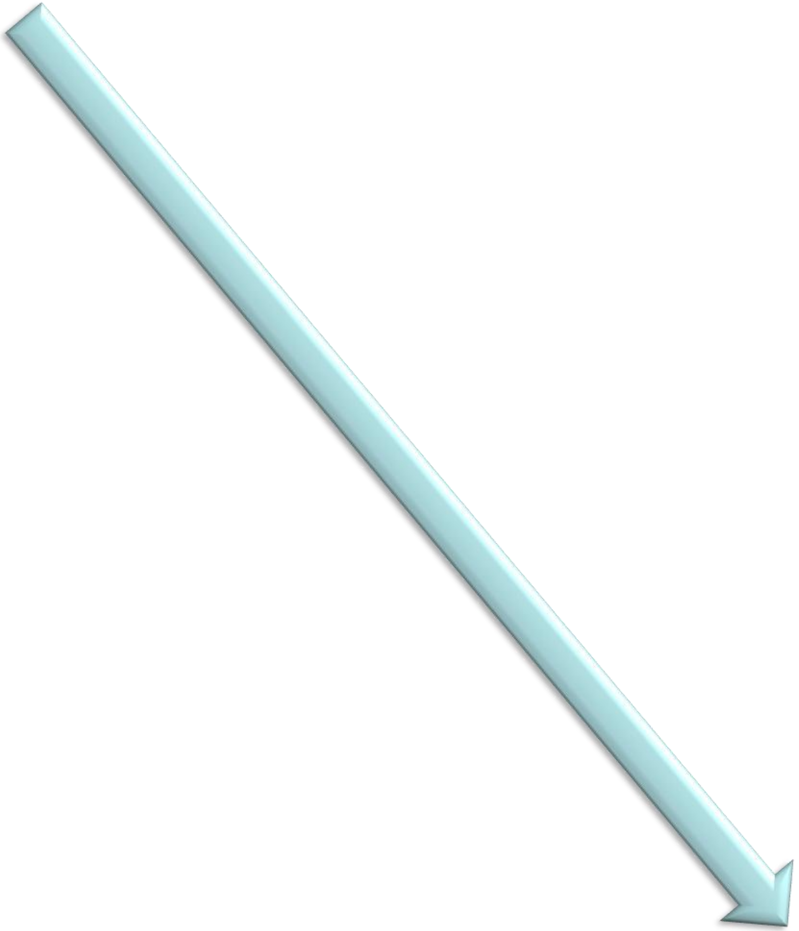

Web Application Security Testing Methodologies
. . .
Web Application Security Test Cases / Tools
. . .
Security Test Case Selection Criteria
. . .
Web Application Security Test Criteria
Adequacy Criteria
Documentation Process

Web Application Security Test Case / Tools

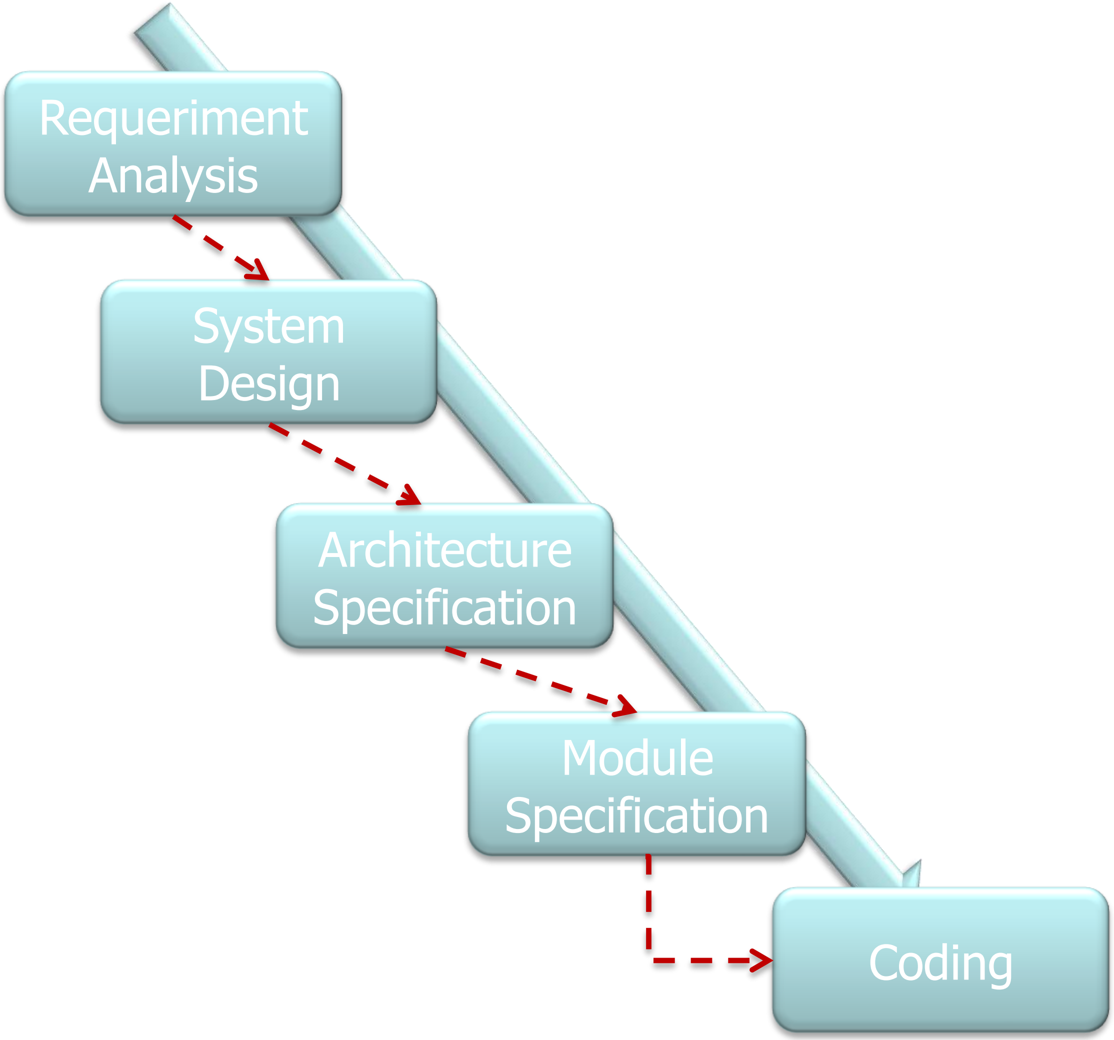
- As the used approach is broad, there might have methodologies that allow the use of the suggested approach.
- As our goal is a preventive approach, the methodology must be defined during the development of the software product. That is, during the SDLC.
- Let's take the Vee model as an example.


Web Application Security Testing Methodologies
. . .
Web Application Security Test Cases / Tools
. . .
Security Test Case Selection Criteria
. . .
Web Application Security Test Criteria
Adequacy Criteria
Documentation Process

Software Testing – Vee model (Generic)



Software Testing – Vee model (Generic)



Requeriment Analysis

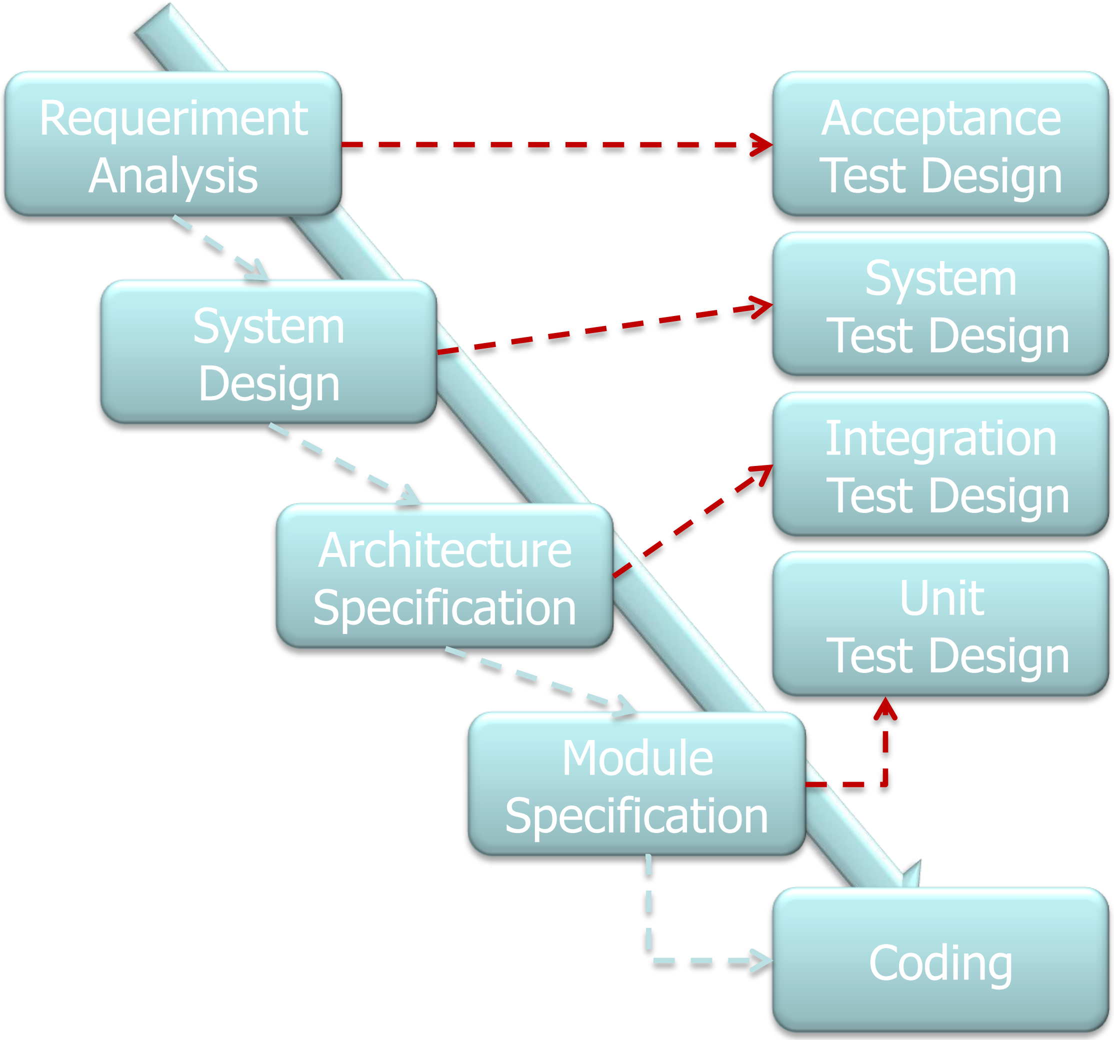
System Design

Architecture Specification

Module Specification

Coding

Software Testing – Vee model (Generic)



Requeriment Analysis

System Design

Architecture Specification

Module Specification

Coding

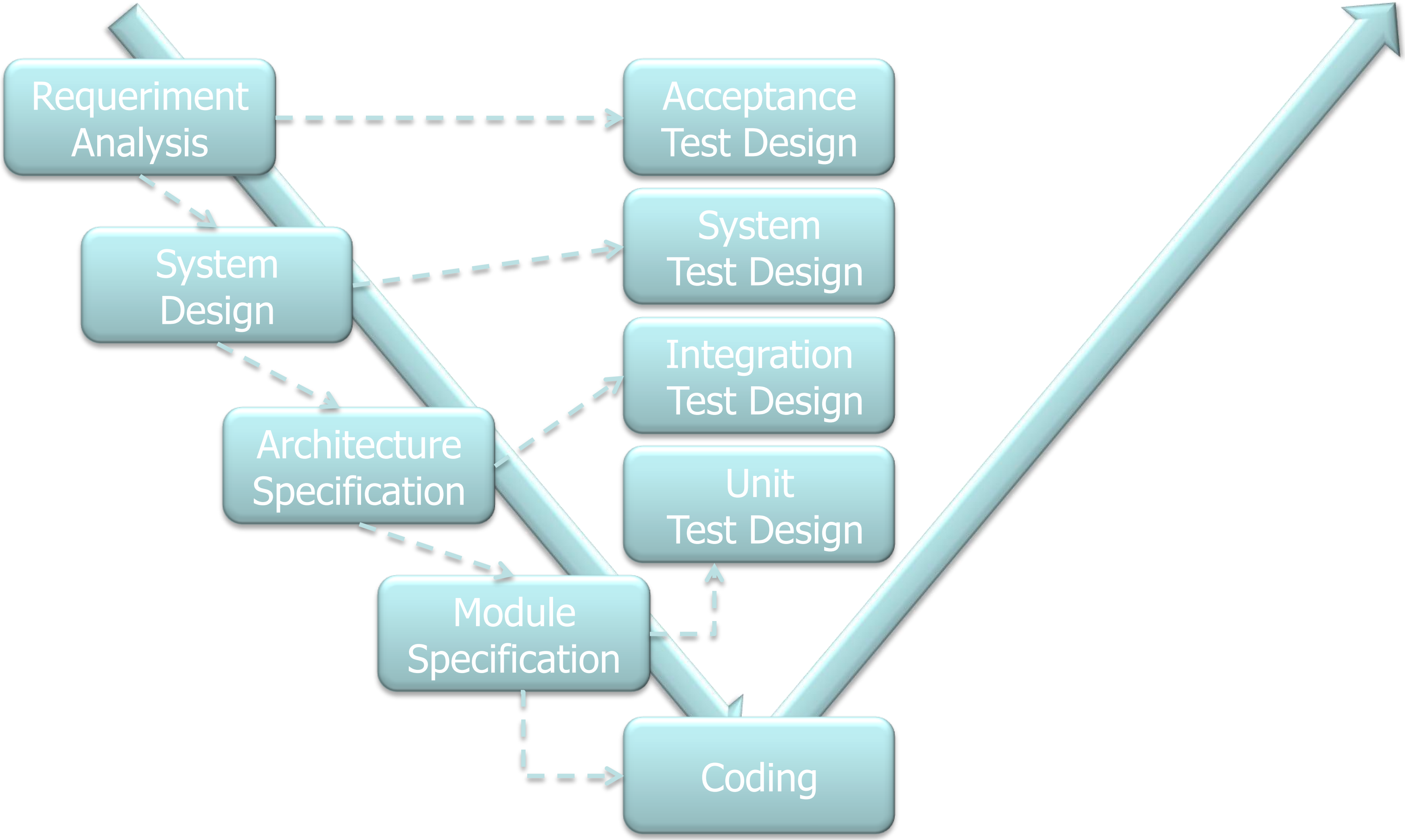
Acceptance Test Design

System Test Design

Integration Test Design

Unit Test Design

Software Testing – Vee model (Generic)



Requeriment Analysis

System Design

Architecture Specification

Module Specification

Coding

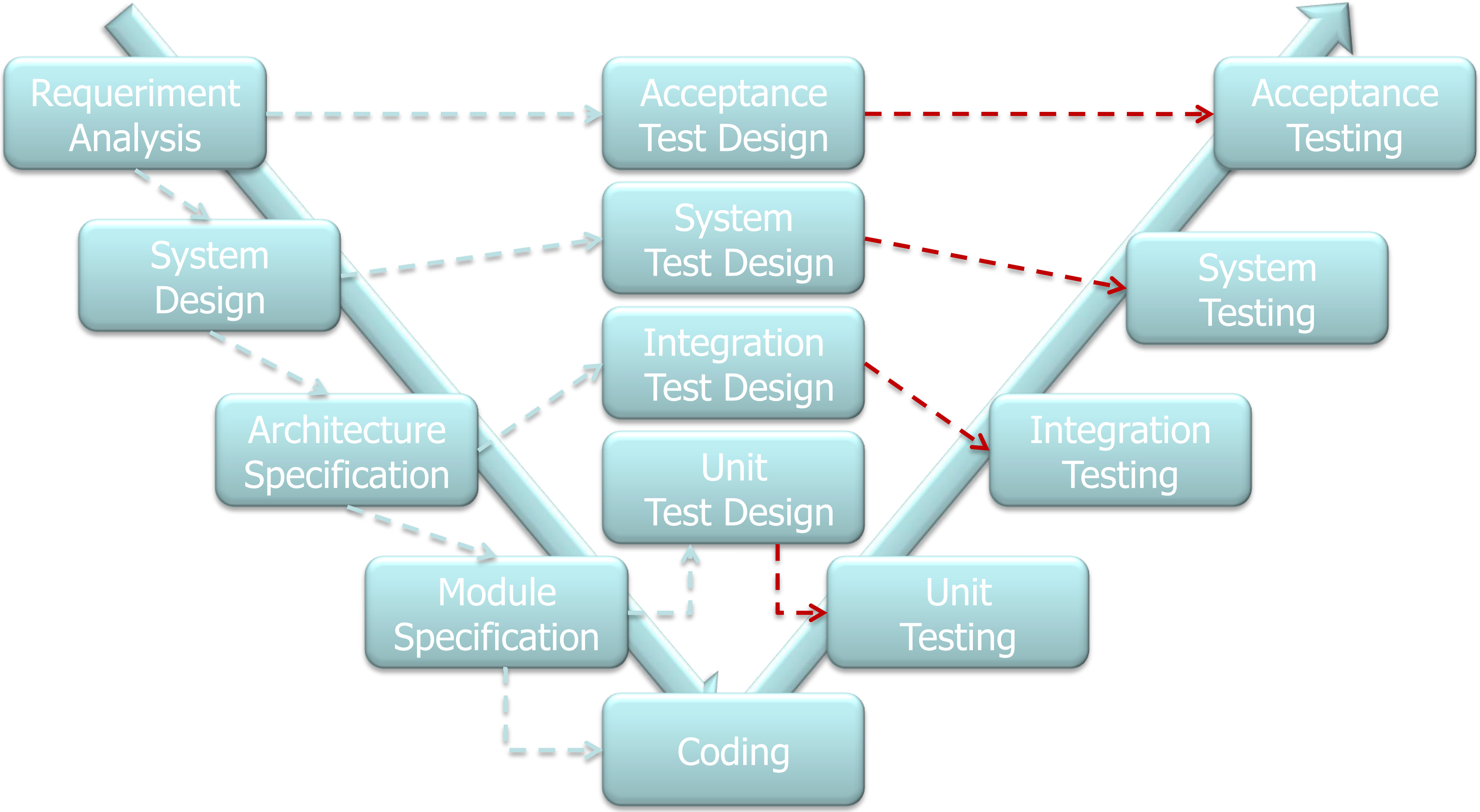
Acceptance Test Design

System Test Design

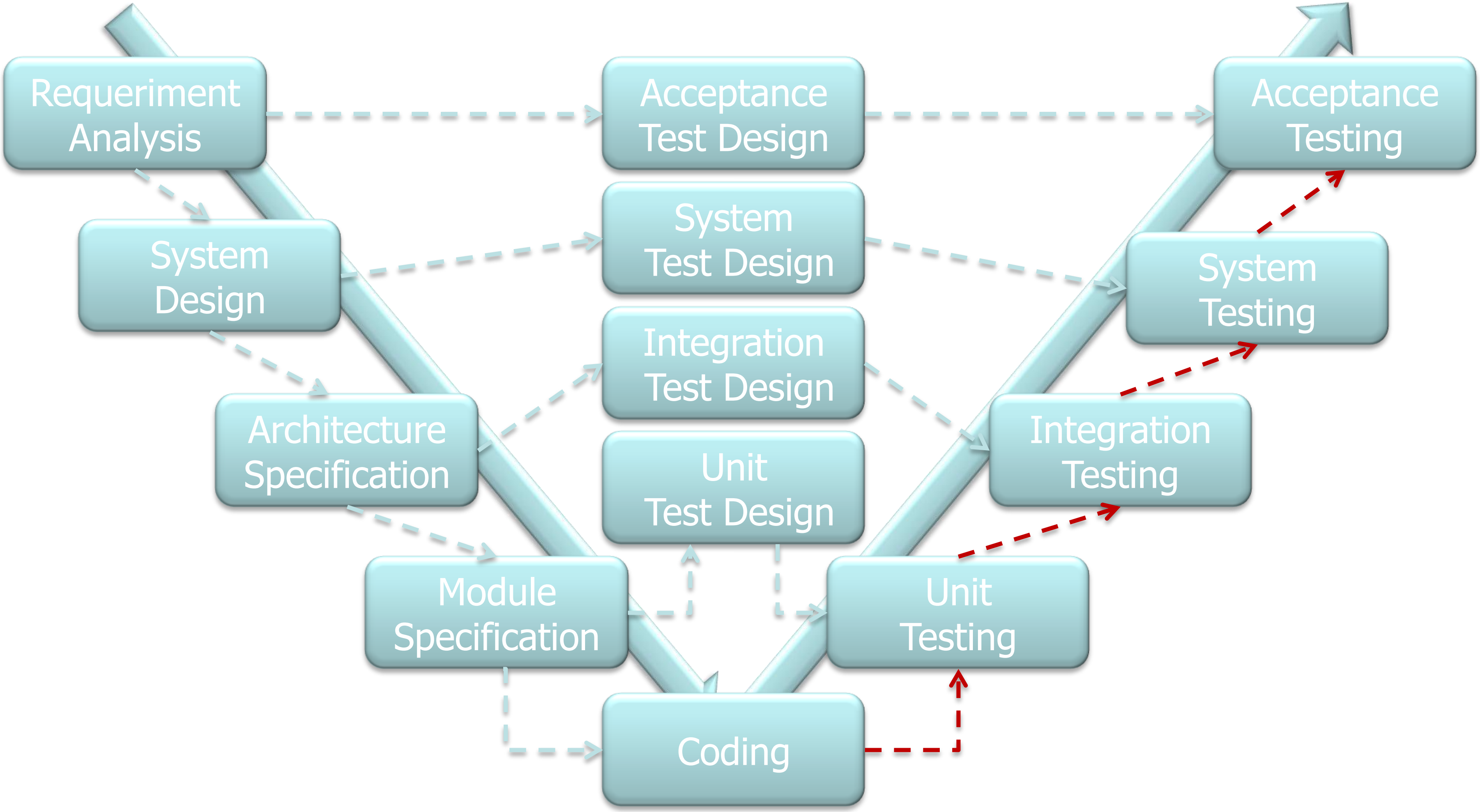
Integration Test Design

Unit Test Design

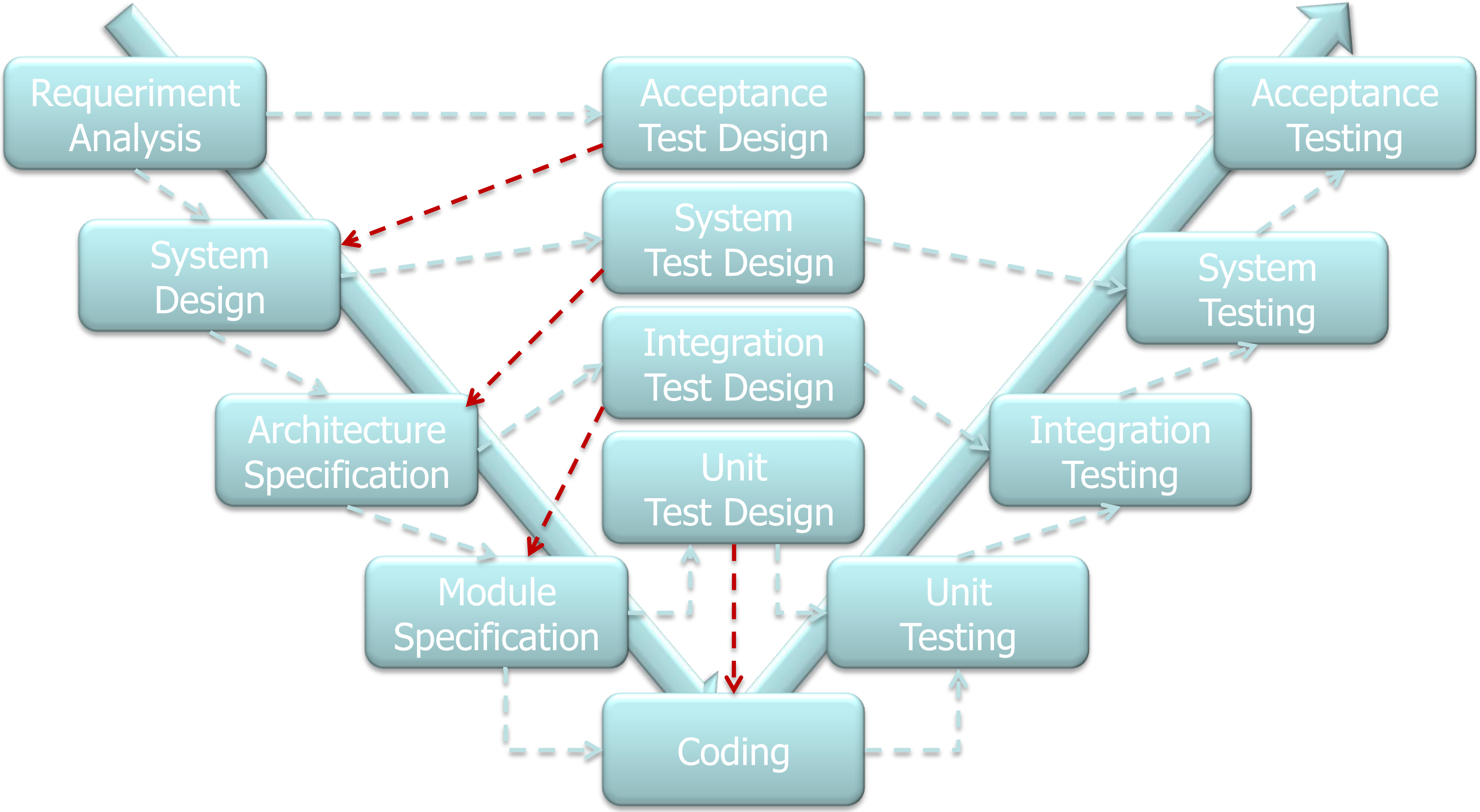
Software Testing – Vee model (Generic)



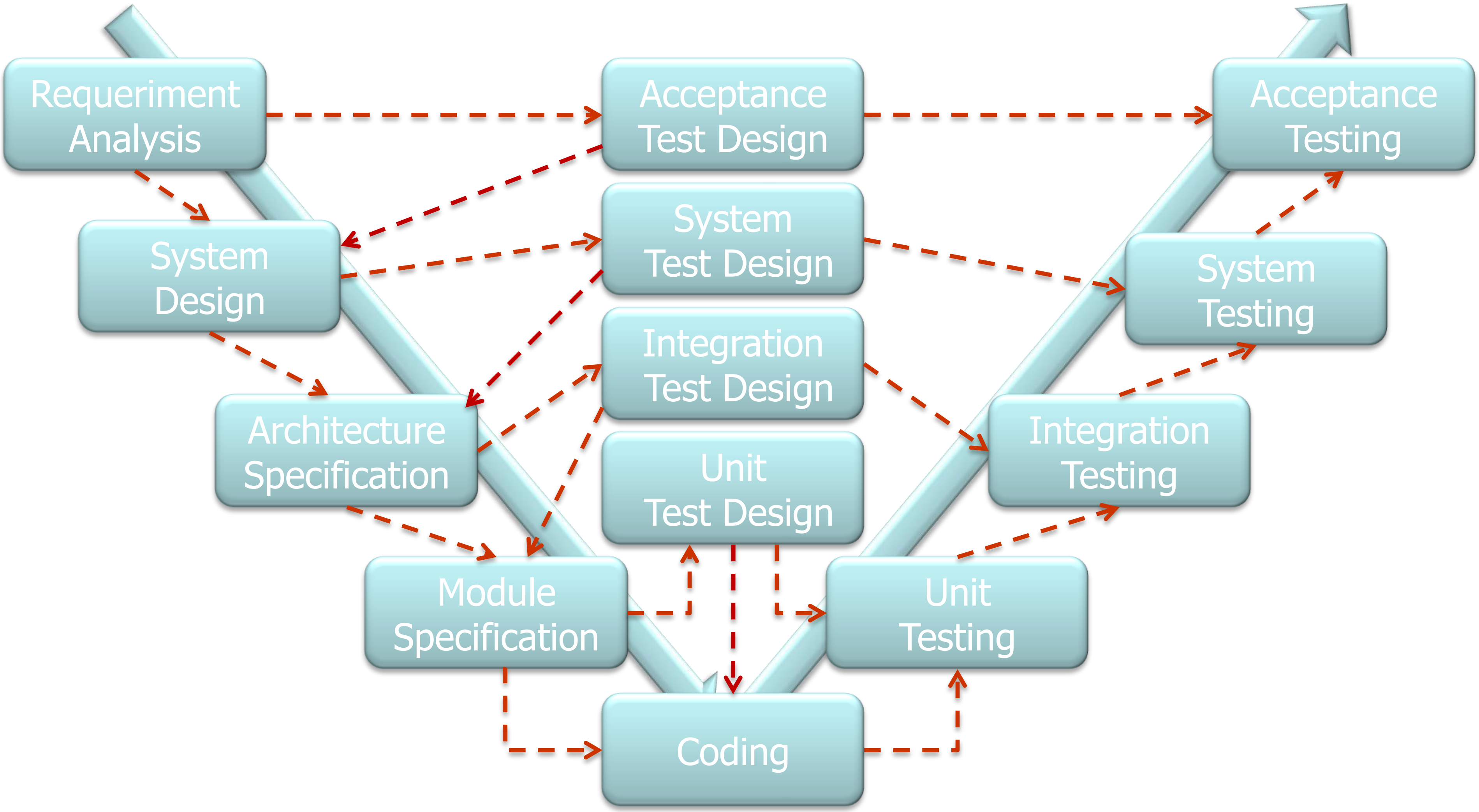
Software Testing – Vee model (Generic)



Software Testing – Vee model (Generic)

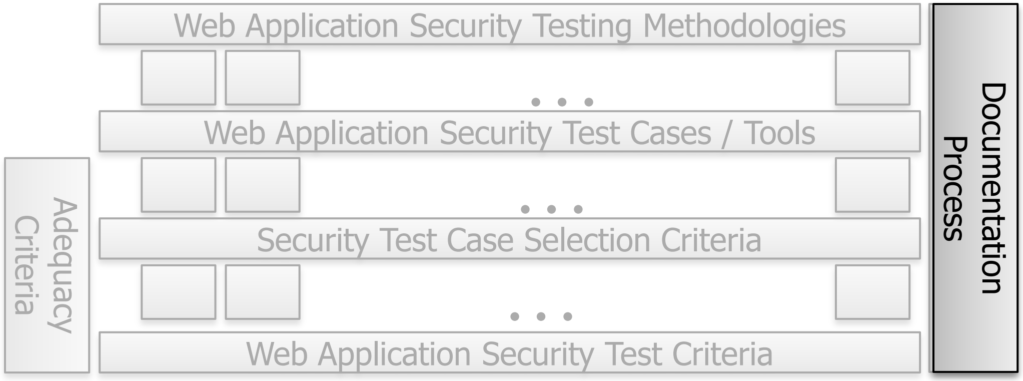


Software Testing – Vee model (Generic)

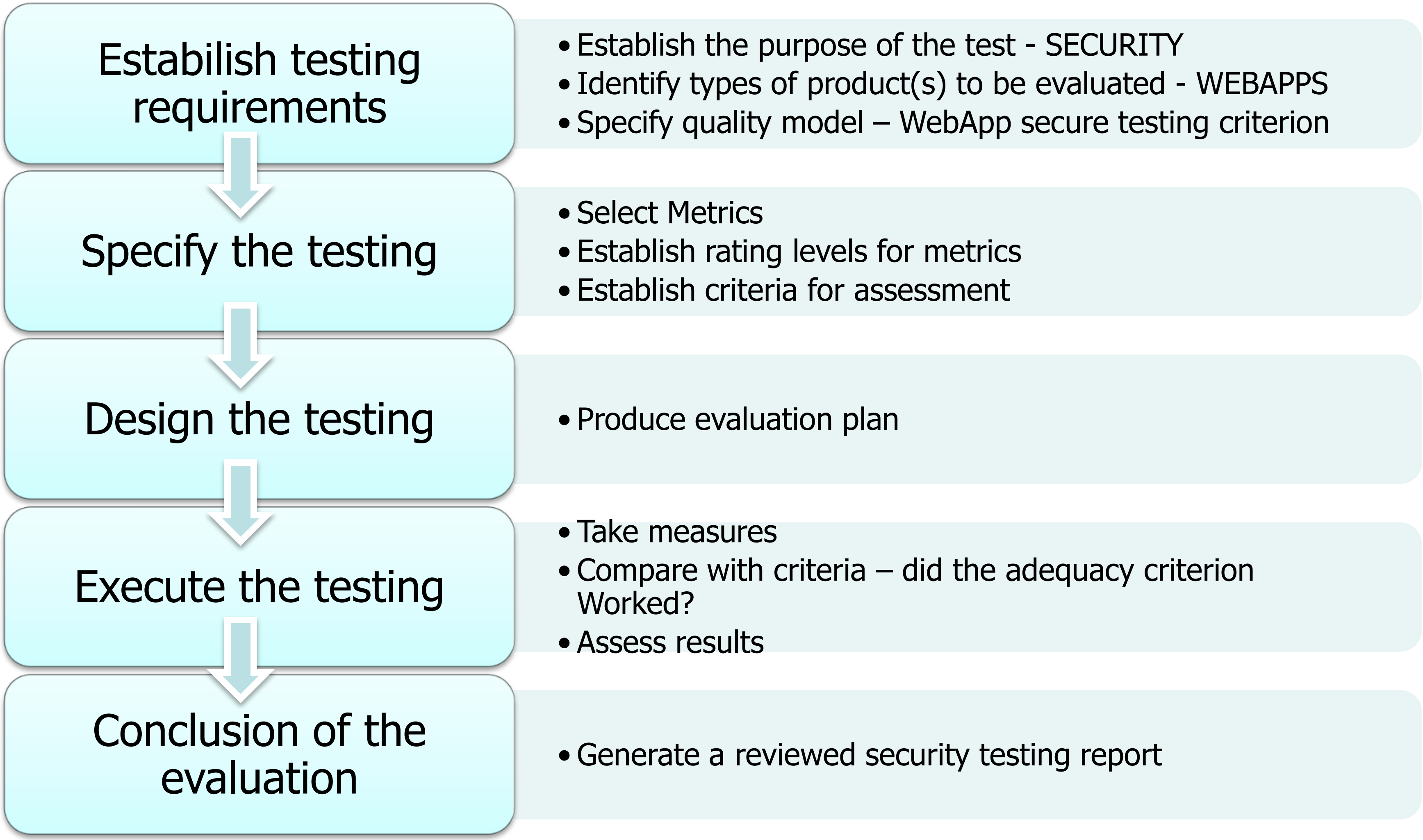


Documentation Process

- The documentation process is the base of RRIO. It is important to use a well-established process of testing so the documentation process is naturally achieved.
- Remember ISO/IEC 14598-5? We extended it to security testing. So the process of testing follows this ISO and so does documentation.


Web Application Security Testing Methodologies
Web Application Security Test Cases / Tools
Security Test Case Selection Criteria
Web Application Security Test Criteria
Adequacy Criteria
Documentation Process

Testing phases, based on ISO/IEC 14598-5



Estabilish testing requirements

- Establish the purpose of the test - SECURITY
- Identify types of product(s) to be evaluated - WEBAPPS
- Specify quality model – WebApp secure testing criterion

Specify the testing

- Select Metrics
- Establish rating levels for metrics
- Establish criteria for assessment

Design the testing

- Produce evaluation plan

Execute the testing

- Take measures
- Compare with criteria – did the adequacy criterion Worked?
- Assess results

Conclusion of the evaluation

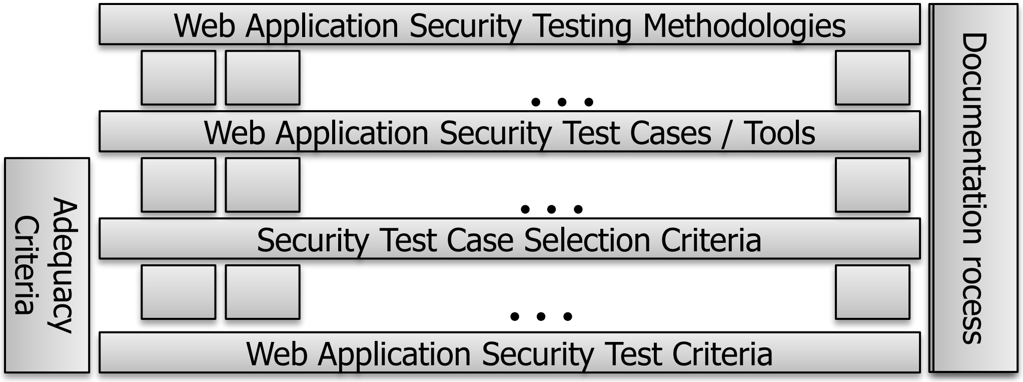
- Generate a reviewed security testing report

Testing phases, based on ISO/IEC 14598-5

- Depending on what is being tested (acceptance, system, integration, unit) we must use the test phases in different ways.
  - For example, the test criteria may be different for each phase;
    The test case selection criterion for unit testing uses knowledge of the webapps source-code, code review and inspection tools might be used as well as patterns of input validation.
  - The criterion for selection of test cases for integration testing can involve manipulation of input and communication between modules.
- That is, for each type of test this set of phases should be re-drawn, re-structured, re-thought.

Testing phases, based on ISO/IEC 14598-5

- The steps should be repeated until the program reaches the desired level of security. (Maturity models?)
- Remember:
  - New risk analysis should be made;
  - New types of threats;
  - New security use cases;
  - New attacks trees;
  - New tests case.


Web Application Security Testing Methodologies
. . .
Web Application Security Test Cases / Tools
. . .
Security Test Case Selection Criteria
. . .
Web Application Security Test Criteria
Adequacy Criteria
Documentation rocess

Agenda

- Introduction
- Key Points
- Techniques for software testing
- Software Quality
- Web application security testing approach
- Practical demonstration
- Conclusions

Agenda

- Introduction
- Key Points
- Techniques for software testing
- Software Quality
- Web application security testing approach
- Practical demonstration
- Conclusions

Conclusions

- It is a preventive approach (and practice), using risk analysis (threat modeling) and other technologies, in order to establish guidelines for test criteria for sensitive applications.
- The key point is how to use technology properly in a determinated flow of phases in order to achieve a given result.
- It is fully applicable to any type (adherent) of software testing model and any SDLC.

Thanks