Security testing – An Introduction

Contents

[Introduction 3](#_Toc489959577)

[What Is Vulnerability Assessment and Penetration Testing? 3](#_Toc489959578)

[Features and Benefits of VAPT 4](#_Toc489959579)

[Types of Security Testing 4](#_Toc489959580)

[Attacks 4](#_Toc489959581)

[OWASP TOP 10 5](#_Toc489959582)

[Security Testing Approach 5](#_Toc489959583)

[Security Testing Tools 6](#_Toc489959584)

# Introduction

Security testing is a testing technique to determine if an information system protects data and maintains functionality as intended.

Security testing is a process intended to reveal flaws in the security mechanisms of an information system that protect data and maintain functionality as intended.

Vulnerability assessment is the process of identifying and quantifying security vulnerabilities in an environment. It is an in-depth evaluation of your information security posture, indicating weaknesses as well as providing the appropriate mitigation procedures to eliminate those weaknesses.

The goal of a Security test, is to evaluate the current security status of IT systems. It is a controlled attack which uncovers security flaws in a realistic way.

It also aims at verifying 6 basic principles as listed below:

* Authentication: The origin of the application and its data is genuine.
* Authorization: Specific users should only get access to authorized functions.
* Confidentiality: Data/information is secure from theft.
* Integrity: The application and its data is not altered in course of time during transmission.
* Non repudiation: Guarantee that sender and receiver of information cannot deny having sent or received the data.

# What Is Vulnerability Assessment and Penetration Testing?

Vulnerability Assessment and Penetration Testing (VAPT) are two types of vulnerability testing.

The tests have different strengths and are often combined to achieve a more complete vulnerability analysis.

In short, Penetration Testing and Vulnerability Assessments perform two different tasks, usually with different results, within the same area of focus.

Vulnerability assessment tools discover which vulnerabilities are present, but they do not differentiate between flaws that can be exploited to cause damage and those that cannot.

Vulnerability scanners alert companies to the preexisting flaws in their code and where they are located. Penetration tests attempt to exploit the vulnerabilities in a system to determine whether unauthorized access or other malicious activity is possible and identify which flaws pose a threat to the application.

Penetration tests find exploitable flaws and measure the severity of each.

A penetration test is meant to show how damaging a flaw could be in a real attack rather than find every flaw in a system.

Together, penetration testing and vulnerability assessment tools provide a detailed picture of the flaws that exist in an application and the risks associated with those flaws.

# Features and Benefits of VAPT

Vulnerability Assessment and Penetration Testing (VAPT) provides enterprises with a more comprehensive application evaluation than any single test alone.

Using the Vulnerability Assessment and Penetration Testing (VAPT) approach gives an organization a more detailed view of the threats facing its applications, enabling the business to better protect its systems and data from malicious attacks.

Vulnerabilities can be found in applications from third-party vendors and internally made software, but most of these flaws are easily fixed once found.

Using a VAPT provider enables IT security teams to focus on mitigating critical vulnerabilities while the VAPT provider continues to discover and classify vulnerabilities.

# Types of Security Testing

* Web Application
* Mobile Application
* Desktop Application

# Attacks

* **SQL Injection**:

Any application that passes SQL queries through URL or text fields is potentially vulnerable to manual editing of these fields. SQL injection can result in the returning some confidential data or the granting unauthorized access.

* **Cross Site Scripting (XSS)**:

This allows a user to embed a JavaScript, ActiveX, HTML or other scripts in the client side web page and then gather the client’s confidential information. The most common form of XSS involve the unauthorized addition of advertisements, links, and offers on various dynamic web pages especially social networking sites that benefit the cracker.

* **Url Manipulation**:

By changing certain parts of a URL, a malicious cracker may get access to unauthorized pages.

For example, changing a URL from /login to /play on a gaming site shouldn’t allow direct access to the games.

* **Brute Force Attack**:

This type of attack requires automated software. The idea is to try a large combinations of username/passwords to match with a valid combination and get unauthorized access.

Using automated tools, a number of username/password combinations can be combined and tested on an application. The application, if properly secured, should not allow repeated login attempts after a limited number of invalid attempts. An unsecured site will allow repeated and frequent login attempts and may give access after some time to a combination/input.

* **Session Hijacking**:

Here the malicious cracker keeps sniffs on user login/transaction activities. As soon a successful session is created between user and web server, the cracker hijacks or simply steals the session activities, or session ID. These session IDs contains confidential information which the cracker can then use to gain unauthorized access to that account and possibly the web server. Using a sniffer like Fiddler, the cracker watches activity that passes between an application and a web browsers looking for login requests. Once a session id/key is spotted, the cracker copies and pastes it in to a browser. If the session, and app, is secured, the cracker will be sent to a login or similar page. Otherwise, the cracker has all of the access privileges of the account.

# OWASP TOP 10

* Injection
* Broken Authentication and Session Management
* Cross-Site Scripting (XSS)
* Insecure Direct Object References
* Security Misconfiguration
* Sensitive Data Exposure
* Missing Function Level Access Control
* Cross-Site Request Forgery (CSRF)
* Using Components with Known Vulnerabilities
* Invalidated Redirects and Forwards

# Security Testing Approach

We can take the following approach while preparing and planning for Security testing:

* *Security Architecture Study*: The first step is to understand the business requirements, security goals, and objectives in terms of the security compliance of the organization. The test planning should consider all security factors, like the organization might have planned to achieve PCI compliance.
* *Security Architecture Analysis*: Understand and analyze the requirements of the application under test.
* *Classify Security Testing*: Collect all system setup information used for development of Software and Networks like Operating Systems, technology, hardware. Make out the list of Vulnerabilities and Security Risks.
* *Threat Modelling*: Based on above step, prepare Threat profile.
* *Test Planning*: Based on identified Threat, Vulnerabilities and Security Risks prepare test plan to address these issues.
* *Traceability Matrix Preparation*: For each identified Threat, Vulnerabilities and Security Risks prepare Traceability Matrix.
* *Security Testing Tool identification*: All security testing cannot be executed manually, so identify the tool to execute all security test cases faster & more reliably.
* *Test Case Preparation*: Prepare the Security tests case document.
* *Test Case Execution*: Perform the Security Test cases execution and retest the defect fixes. Execute the Regression Test cases.
* *Reports*: Prepare detailed report of Security Testing which contains Vulnerabilities and Threats contained, detailing risks, and still open issues etc.

# Security Testing Tools

* OWASP ZAP
* IRON WASP
* VEGA
* Burp Suite
* Web Scarab
* Nessus
* IBM AppScan
* Nikto
* OpenVAS