**Security Testing:**

## What is Security Testing?

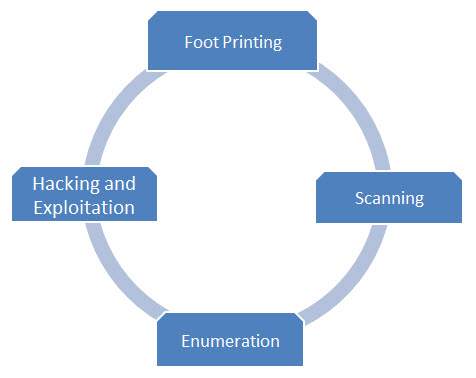
Security testing is a testing technique to determine if an information system protects data and maintains functionality as intended. By Performing security testing, it is no guarantee that systems are secure but it is important to include the security testing as part of the testing process. It also aims at verifying 6 basic principles as listed below:

* **Confidentiality (**is the protection of personal information**)**
* **Integrity (**doing the right thing in a reliable way**)**
* **Authentication (**process of identifying an individual, usually based on a username and password)
* **Authorization (**access policy**)**
* **Availability ()**
* **Non-repudiation (Nonrepudiation** is the assurance that someone cannot deny something**)**

As we are involved in the process, documentation should be done phase so that all the steps necessary to reproduce the attack are available readily which is the basis for the detailed report customers receive at the end of a penetration test.

These phases are re-iterated multiple times in a pentest phase which goes hand in hand with the normal SDLC.

**Pentest Workflow**

`

The four major phases of security Testing are the following. Click on each one of the phases to understand in detail.

* Foot Printing
* Scanning
* [Enumeration](https://www.tutorialspoint.com/security_testing/security_testing_enumeration.htm)
* [Exploitation](https://www.tutorialspoint.com/security_testing/security_testing_exploitation.htm)

## What is Footprinting?

Footprinting, the first step that a penetration tester used to evaluate the security of the web application. It means gathering the blue print of a particular system or network and about the devices that are attached to the network under consideration.

After performing this very first important step, penetration tester can understand the pulse of a hacker. It is good to understand the complete information about the system which will pave the way for the hacker.

This free **website blueprint** is a document you use to plan out the construction of a successful keyword focused content **website**. ... The **blueprint** is also a great planning aid because it allows you to record keyword demand and profitability and keep running notes on the organization of your **website** as it grows.

## FootPrinting Steps

* **Information gathering**
* **Determining the range of the network**
* **Identifying active machines**
* **Identifying open ports and access points**

**OS fingerprinting (OS fingerprinting** is the process of determining the **operating system** used by a host on a network. What is active finger printing?

**Active fingerprinting** is the process of transmitting packets to a remote host and analysing corresponding replies.

* **Fingerprinting services**
* **Mapping the network**

What is the meaning of mapping in networking?

In Windows Vista, a **network map** is a graphical representation of all the computers and devices on your **network** that shows how each is connected. To appear on the **network map**, the device or computer needs to support UPnP technology or Web Services for Devices for Windows.

## What is Scanning?

Scanning, the second step that we perform after Footprinting. It involves scanning Open ports, fingerprinting the operating system, and uncovering services on ports. The ultimate goal in scanning is to find open ports through external or internal network scanning, pinging machines, determining network ranges and port scanning individual systems.

There are various methodologies/approaches which we can make use as a reference for performing the attaks. Below are the following standards one can take into account while making developing their attack model.

Among the below list, OWASP is the most active and there are lot of contributors. We will focus on OWASP Techniques which each development team takes into consideration before designing a web app.

* **PTES - Penetration Testing Execution Standard**
* **OSSTMM - Open Source Security Testing Methodology Manual**
* **OWASP Testing Techniques - Open Web Application Security Protocol**

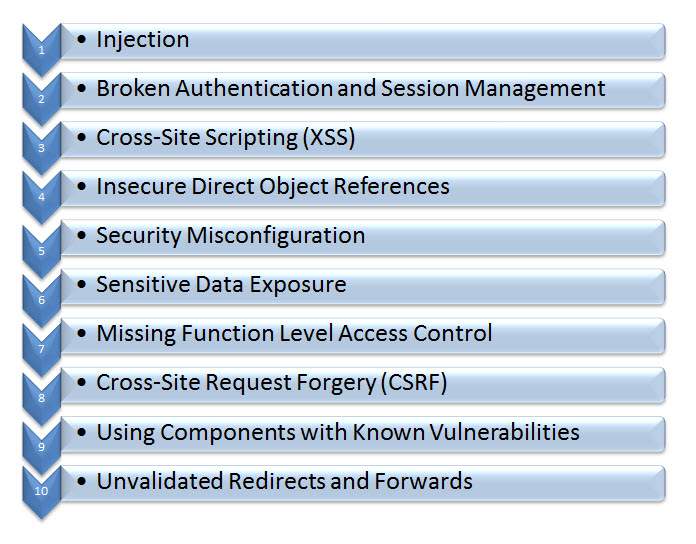
## What is Enumeration?

Enumeration is the last step and performed as a result of scanning. The Goal of enumeration is to get the complete picture of the target. In this phase, a penetration tester tries to identify valid user accounts or poorly-protected shared resources using active connections to systems.

# What is Exploitation?

In this last phase, a security tester actively exploit security weakness. Once the attack is successful, it is possible to penetrate more systems in the domain, because the pentesters now have the access to more potential targets that were not available before.

## OWASP Top 10



**HTTPS Protocol**

HTTPS (Hypertext Transfer Protocol over Secure Socket Layer) or HTTP over SSL is a web protocol developed by Netscape. It is not a protocol but it is just the result of layering the HTTP on top of SSL/TLS (Secure Socket Layer/Transport Layer Security).

**Inshort, HTTPS = HTTP + SSL**

**When Https Required**

When we browse, we normally send and receive information using HTTP protocol. So this leads anyone to eavesdrop on the conversation between our computer and the web server. Many a times we need to exchange sensitive information which needs to be secured and to prevent unauthorized access.

Https protocol used in the following scenarios

* Banking Websites
* Payment Gateway
* Shopping Websites
* All Login Pages
* Email Apps

**Basic Working of HTTPS**

* Public key and signed certificates are required for the server in HTTPS Protocol.
* Client requests for the https:// page
* When using an https connection, the server responds to the initial connection by offering a list of encryption methods the webserver supports.
* In response, the client selects a connection method, and the client and server exchange certificates to authenticate their identities.
* After this is done, both webserver and client exchange the encrypted information after ensuring that both are using the same key, and the connection is closed.
* For hosting https connections, a server must have a public key certificate, which embeds key information with a verification of the key owner's identity.
* Almost all certificates are verified by a third party so that clients are assured that the key is always secure.

## What is Encoding?

Encoding is the process of putting a sequence of characters such as letters, numbers and other special characters into a specialized format for efficient transmission while Decoding is the process of converting an encoded format back into the original sequence of characters. It is completely different from Encryption which we usually misinterpret.

Encoding and decoding are used in data communications and storage. Encoding should NOT be used for transporting sensitive information.

## URL Encoding

URLs can only be sent over the Internet using the ASCII character-set and there are instances when URL contains special characters apart from ASCII characters, it needs to be encoded. URLs donot contain spaces and are replaced with a plus (+) sign or with %20.

## What is a cookie?

cookie, a small piece of info sent by web server to store on a web browser so that it can later read by the browser. Hence browser remembers some specific personal information. If a Hacker gets hold of the cookie information, will lead to security issues.

## Cookies - Properties

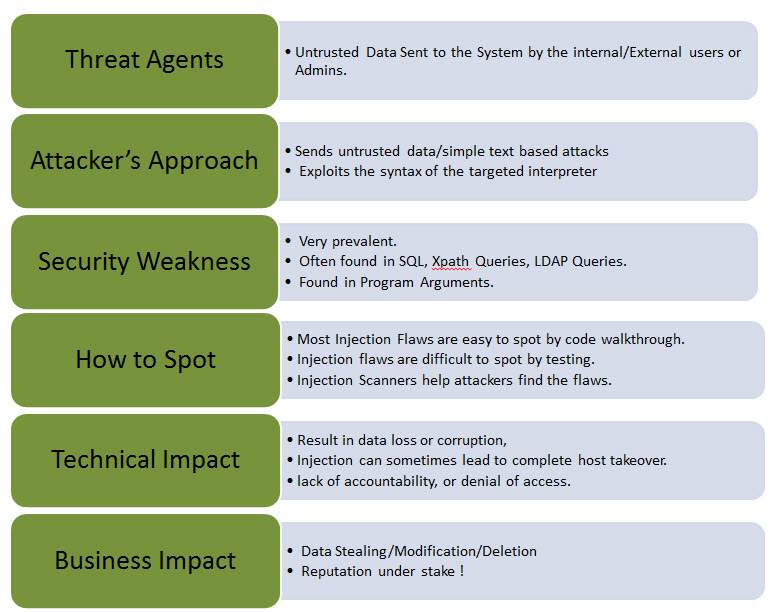
* It is Usually small text files, given ID tags that are stored on your computer's browser directory
* It is Used by web developers to help users navigate their websites efficiently and perform certain functions
* When the user browses the same website again, the data stored in the cookie is sent back to the web server to notify the website of the users previous activity.
* Cookies are unavoidable for websites that have huge databases, need logins, have customizable themes.

**Editing Cookies:** If the application uses cookies to store login information then as a tester we should try changing the user in the cookie or address bar to another valid user. Editing the cookie should not let you log in to a different users account.

**Web Application - Injection**

Injection technique consists of injecting a SQL query or a command using the input fields of the application. A successful SQL injection can read, modify sensitive data from the database and it can also to delete data from database. It also enables the hacker to perform administrative operations on the database such as shutdown the DBMS/dropping databases.

Let us understand Threat Agents, Attack Vectors, Security Weakness, Technical Impact and Business Impacts of this flaw with the help of simple diagram.



## Preventing SQL Injection

There are plenty of ways to prevent SQL injection. When developers write

the code they should ensure that they handle special characters accordingly. Using Parameterized Queries , Escaping all User Supplied Input, Enable Least Privilege for the database for the end users

# AJAX Security

Asynchronous Javascript and XML (AJAX) is one of the latest techniques used to develope web application inorder to give a rich user experience. Since it is a new technology there are many security issues that are yet to be completed established and below are the few security issues in AJAX.

* The attack surface is more as there are more inputs to be secured.
* It also Exposes the internal functions of the applications.
* Failure to protect authentication information and sessions.
* A very narrow line between client-side and server-side hence there are possibilities of committing security mistakes.

## Preventing Mechanisms

In Client side :

* Use .innerText instead of .innerHtml.
* Don't use eval. (The eval() function evaluates JavaScript code represented as a string. ... If you construct an arithmetic expression as a string, you can use eval() to evaluate it at a later time)
* Don't rely on client logic for security.
* Avoid writing serialization code.
* Avoid building XML dynamically.
* Never transmit secrets to the client.
* Don't perform encryption in client side code.
* Don't perform security impacting logic on client side.

In Server side:

* Use CSRF protection. (Cross-Site Request Forgery (CSRF) is an attack that forces an end user to execute unwanted actions on a web application in which they're currently authenticated. CSRF attacks specifically target state-changing requests, not theft of data, since the attacker has no way to see the response to the forged request.)
* Avoid writing serialization code.
* Services can be called by users directly.
* Avoid building XML by hand, use the framework.
* Avoid building JSON by hand, use an existing framework.