Name:
Date:
Org:

Pentest Report

Vulnerabilityanalys of XXXXXX

Methodology

The Penetration Testing Execution Standard (PTES) is a detailed guide for conducting penetration tests, created by security experts. It standardizes testing methods, helping security professionals and businesses. It covers testing from planning to reporting, and emphasizes a tailored approach for each organization's needs. The PTES improves penetration testing effectiveness.

Pre-engagement Interactions

This is the preparation phase for the pen test. It is all about document approvals and tools needed for the test.

Intellegence Gathering

In this phase information about the target system are gathered from external sources like social media websites, official records etc. This phase comes under OSINT (Open-Source Intelligence).

Threat Modelling

It is a procedure for optimizing network security by identifying objectives and vulnerabilities, and then defining countermeasures to prevent, or mitigate the effects of, threats to the system. It is skipped in typical pan tests.

Vulnerability Analysis

This phase discovers and validates vulnerabilities.That is risk that an attacker could exploit and gain authorized access to the system or application.

Exploitation

In this phase, the tester try to reach the security of the target system using the vulnerabilities previously identified and validated.

Post Exploitation

This phase maintains the control over target system and collects data.

Reporting

Documents entire process in a form understandable to the client.

Results

Result presents all the findings, how, and why the finding was discovered. Results also presents if the discovered vulnerability is a threat.

Recommendations

Recommendations presents all the recommendations for mitigating and avoiding these types of vulnerabilities.

How to prevent SQLi (SQL Injection) attacks?

SQLi is a type of attack that is used to exploit SQL statements. It is one of the most common types of attacks that are used by hackers to gain access to a database.
The most popular way to protect against SQL injection is to use prepared statements, which are statements that have been prepared ahead of time. This is done by using the parameterized query. The parameter is the part of a SQL statement that contains the value that will be used in the query, and the placeholders are the values that the user will put in. For example, if the statement is:
SELECT * FROM users WHERE username =?
Then the parameters would be:? and?. If the username is not in a table, then the? will not be there. In this case, the prepared statement will look like this: SELECT * from users where username=?. When the database is executed, it will first check to see if there is an entry for the given username in any of its tables. Then, depending on the result, a different query will execute. So, in this example the first query would execute, but the second query wouldn't. By using prepared queries, you can prevent this type SQL Injections. Here are some of them: SQL injections in PHP
PHP is very popular for web development. PHP is also very easy to learn and use. However, there are a few things that