**Testing Strategies**

The languages used by our team is JavaScript, SQL, PHP predominantly.

**Types of testing methodologies that are to be used for this project::**

1. Penetration testing
2. Security testing
3. Unit testing
4. Integration testing

**Tools to be used ::**

**Penetration testing:**

1. **Metasploit**

[Metasploit](https://www.cybrary.it/course/metasploit/) is a very popular collection of various [penetration tools](https://www.metasploit.com/). Cyber security professionals and other IT experts have used it for years to accomplish various objectives, including discovering vulnerabilities, managing security evaluations, and formulating defense methodologies.

You can use the Metasploit tool on servers, online-based applications, networks, and several other places. If a new security vulnerability or exploit is reported, the utility will have it. If you need to evaluate the security of your infrastructure against older vulnerabilities, Metasploit will have you covered.

1. **Nmap**

[Nmap](https://nmap.org/), also known as network mapper, is a free and open source tool for scanning your systems or networks for vulnerabilities. The tool is also helpful in carrying out other activities, including monitoring host or service uptime and performing mapping of network attack surfaces.

Nmap runs on all the major operating systems and is suitable for scanning both large and small networks.

With the utility, you can understand the various characteristics of any target network, including the hosts available on the network, the type of operating system running, and the type of packet filters or firewalls in place.

**Security testing :**

1. **Wapiti**

One of the leading web application security testing tools, [Wapiti](http://wapiti.sourceforge.net/) is a free of cost, open source project from SourceForge and devloop. In order to check web applications for security vulnerabilities, Wapiti performs black box testing. As it is a command-line application, it is important to have a knowledge of various commands used by Wapiti.

Wapiti is easy to use for the seasoned but testing for newcomers. But don’t worry, you can find all the Wapiti instructions on the official documentation. For checking whether a script is vulnerable or not, Wapiti injects payloads. The open source security testing tool provides support for both GET and POSTHTTP attack methods. Vulnerabilities exposed by Wapiti are:

* Command Execution detection
* CRLF injection
* Database injection
* File disclosure
* Shellshock or Bash bug
* SSRF (Server Side Request Forgery)
* Weak .htaccess configurations that can be bypassed
* XSS injection
* XXE injection

Key highlights:

* Allows authentication via different methods, including Kerberos and NTLM
* Comes with a buster module, allowing brute force directories and files names on the targeted web server
* Operates like a [fuzzer](https://en.wikipedia.org/wiki/Fuzzinghttps:/en.wikipedia.org/wiki/Fuzzing)
* Supports both GET and POSTHTTP methods for attacks

1. **W3af**

One of the most popular web application security testing frameworks that are also developed using Python is [W3af](http://w3af.org/). The tool allows testers to find over 200 types of security issues in web applications, including:

* Blind SQL injection
* Buffer overflow
* Cross-site scripting
* CSRF
* Insecure DAV configurations

Key highlights:

* Authentication support
* Easy to get started with
* Offers intuitive GUI interface
* Output can be logged into a console, a file or email

Unit Testing :

1. **SimpleTest**

* SimpleTest is an open – source unit testing framework dedicated to PHP Programming Language
* This framework supports SSL, forms, proxies and basic authentication
* The test case classes in SimpleTest are being extended from base test classes along with methods and codes
* SimpleTest includes autorun.php.file to transform test cases into executable test scripts

1. **Karma**

* Karma is an open source testing framework that provides productive testing framework
* It is a test runner for JavaScript that runs tests on real devices
* Facilitates easy debugging and efficiently integrated with Jenkins, Travis or Semaphore
* Karma is known as ‘Testacular’ that is Spectacular Test Runner for JavaScript

Integration testing :

### Jasmine

### Jasmine is a behavioral driven development (BDD) framework. Using this tool tests can be run in isolation. Jasmine tool supports various browsers like Chrome, Internet Explorer, Safari, Firefox etc. It suits for websites where JavaScript runs.

### Citrus

Citrus is a test framework written in Java that assists in automated integration testing of message based application and data formats. Citrus validates for JSON, XML and plain text messaging request and response data.

Consider an example in which application under test is available on the application server. This application interacts with Citrus using various message transports like SOAP, HTTP, and JMS. In this case, Citrus acts as a both client and server side and simulate the request and response messages.

**Features:**

* Citrus is open source and licensed under Apache License 2.0
* Set sequence of messages
* Create error messages
* Message Header Validation
* Sending and receiving messages
* Wait for message and trigger another message
* Supports integration testing for message transport connectivity
* Validation of XML response
* Validate existence of data

**DAST Tools**

Dynamic application security testing (DAST) is a process of testing an application or software product in an operating state. This kind of testing is helpful for industry-standard compliance and general security protections for evolving projects.

**Tool Used :: VEGA**

Vega is a free and open source web security scanner and web security testing platform to test the security of web applications. Vega can help you find and validate SQL Injection, Cross-Site Scripting (XSS), inadvertently disclosed sensitive information, and other vulnerabilities. It is written in Java, GUI based, and runs on Linux, OS X, and Windows.

Vega can help you find vulnerabilities such as: reflected cross-site scripting, stored cross-site scripting, blind SQL injection, remote file include, shell injection, and others. Vega also probes for TLS / SSL security settings and identifies opportunities for improving the security of your TLS servers.

Vega includes an automated scanner for quick tests and an intercepting proxy for tactical inspection. The Vega scanner finds XSS (cross-site scripting), SQL injection, and other vulnerabilities. Vega can be extended using a powerful API in the language of the web: Javascript.