



# Web Application Penetration Testing

July 2009



## Introduction

Owing to the ubiquity, ease of access, cost effectiveness and provision of service, the Web Application has emerged as a driving force of adoption. With the advent of web 2.0 and web 3.0 technologies, web application has evolved to be more advanced, quicker in response times than that of desktop applications. Also, the user interfaces are getting more and more spontaneous and sophisticated. Today Web Applications are more functional and flexible, which increases their value to business operations. It is this wide acceptability and adaptability of web applications that make them an enticing target for malicious users. The increasing complexity and use of new technologies has opened doors to greater and more devastating security risks.

## Incidents of Security Breaches

- ▶ Fraudsters broke into the Glo promo website and stole the details of ATM account and PIN numbers of unsuspecting ATM users, reports Daily Trust, a National daily based in Abuja, Nigeria.
- ▶ 40,000 websites worldwide have fallen under the spell of a sneaky piece of attack code as per the latest report released by the UK security experts, says Websense, a company specialized in Web security gateway software.
- ▶ The website of 'Popular Holding,' a local stationery and book store in Singapore, was hacked, reports Websense.
- ▶ Hacker deletes 100,000 websites reports MXLogic a leading provider of online security services.
- ▶ Virginia Department of Health Professionals (VDHP) website attacked, the hacker erased all the records on the VDHP servers and now demanding \$10M ransom to return the files reported EHealthEurope, a news portal.
- ▶ Hundreds of UK Government, School & University Websites hacked, reported the Telegraph.
- ▶ Verizon Business Services reports that there were more web sites breaches in 2008 than in all four previous years combined.
- ▶ As stated by the media reports, U.S. President Barack Obama, whose own presidential campaign computers were hacked, recently said such attacks are so widespread that they have cost more than \$8 billion in damages over the past two years in the United States.

Intending to commit corporate espionage, identity theft, fraud, and other illegal activities, hacker barge into websites resulting in costly and embarrassing service disruptions, down-time, lost productivity, stolen data, regulatory fines, angry users and irate customers. To address these security threats and to prevent the associated negative consequences, companies need frequent and thorough web application penetration testing.

## What is Web Application Penetration Testing (WAPT)

Web Application Penetration Testing (WAPT) is a legally authorized, non-functional assessment of a given web application, carried out to identify loopholes or weaknesses, otherwise known as vulnerabilities. These vulnerabilities, exploited by a malicious user (attacker/hacker), may affect the confidentiality, integrity, availability of the web application and/or information distributed by it. Some the loopholes or vulnerabilities plaguing web applications are SQL Injection(Structured Query Language Injection), XSS(Cross Site Scripting), CSRF(Cross Site Request Forgery), Remote File Include, etc. Apart from these, vulnerabilities may exist in the underlying infrastructure like Operation System, Web Server, Application Server, Database Server, etc. Thereby, WAPT aims at identifying and reporting the presence of these vulnerabilities.

### Benefits of WAPT

- ▶ Proactive protection of information assets against hacking and unauthorized intrusions
- ▶ Provides an insight into the current security posture of the given web application
- ▶ Provides a hacker's eye view of the web application
- ▶ Aids in mitigating costs improving goodwill and brand value

### WAPT Methodology

WAPT should be carried out in a phased manner in order to ensure optimum coverage and at the same time simulate the fluid actions of a real time hacker. The following figure depicts the flow:



Figure 1: WAPT Methodology

## Information Gathering

This is the most critical phase in the methodology as all further phases depend on this. As a part of this phase, information about the target web application should be collected: type of web application (e-commerce, social networking, e-retailing, etc), technology used (J2EE, .NET, PHP, PERL, etc), WHOIS, and traceroute.

Also, publicly available information from search engines and archive sites should be gathered. Other useful sites that aid in gathering information on a given web application are [www.netcraft.com](http://www.netcraft.com) and [www.dnsstuff.com](http://www.dnsstuff.com). Apart from the above, spidering tools such as Paros should also be used to 'crawl' the given web application and reveal all publicly available web pages (URLs) e.g. the admin pages.

## Planning and Analysis

All the data gathered in the above phase, is converted into usable information, in the form of a customized test plan. An important step in this phase is to prepare a checklist of tasks or areas (URLs) or applicable vulnerabilities to cover.

## Vulnerability Assessment

This phase can also be dubbed as active information gathering phase. Various automated scans are run against the target application and its underlying infrastructure (server(s) and network); a web application is only as strong as the infrastructure it is hosted on. Vulnerability

in any of the underlying infrastructure components could compromise the security of the web application.

As part of vulnerability assessment port scans and services, identifications are conducted to determine what other possible paths an attacker might take to gain access. Attempts are also made to identify the patch levels of the operating system and other running services.

Tools: Nmap, Nessus, Nikto

## Attack/Penetration

It is under this phase that the actions of a web application hacker are emulated. Based on the information gathered and analyzed in previous phases and following the customized testplan, attacks are carried out to identify the presence of vulnerabilities in the application.

The techniques and tools used are the same as those used by a real hacker. This is done in order to gain a hacker's eye view of the application. A combination of both, free and commercial tools may be used in order to gain a greater insight into the security posture of the application and also to be able to filter false positives.

A combination of manual and automated methods should be used to evaluate the security of web applications. Automated tools are limited in their ability to access and exercise an application like a real attacker. Therefore, manual methods employed by an experienced and creative security tester are a necessary compliment to automated tools.

Tools: WebScarab, Paros, Metasploit framework, SQLiX, and other commercial tools

There are many other open source and commercial tools other than the mentioned tools that may be used.

## Reporting

This is the final and probably the most important phase as it is here that all the findings are documented for presentation to the stake holders. Successful implementation of the above phases would be futile if they are not properly documented or reported.

At the end of the Attack/Penetration phase, a comprehensive report should be prepared detailing each finding, assigning a suitable severity level to each, delineating the steps necessary to reproduce the vulnerability, and suggesting recommendations to address every vulnerability.

The intent of the final report should be to provide the stakeholders with all of the information necessary to fix the vulnerabilities and test the fixes put in place.

## Conclusion

In light of the growing numbers of web applications, advancements in technology employed by web applications, the constant evolution of features in web applications, and the frequent discovery of new vulnerabilities, the preferred way of ensuring security in web applications is to include security testing as part of the SDLC. However, the reality is that the ease of developing a web application and the focus on functionality and user interface has pushed security testing to the background if it happens at all. Nonetheless, Web Application Penetration Testing should be an integral part of the roll-out and life cycle of every web application.