ANALYSIS OF WEB WORM ATTACK ON WEB APPLICATION

AMALINA MOHD GHAZZALI

A project report submitted in partial fulfillment of the
requirements for the award of the degree of
Master of Computer Science (Information Security)

Faculty of Computer Science and Information Systems
Universiti Teknologi Malaysia

OCTOBER 2008

ABSTRACT

This study of web worms attack on web application can be implemented to enhance the security of current web application. Nowadays, attack from worms and viruses on web application come with several motives, whether to gain administrative access or even for stealing. This can be happening from a simple attack which will lead to a devastating effect to the organization. This analysis consists of several steps from analyzing worms attack to producing a guideline for secure web application development. The worms attack is based on a web application model developed using PHP as programming language and using MySQL database. In this case, the vulnerabilities found on the web application will be match to the method of attack from worms, and finally come out with a guideline to prevent such attacks. Even this guideline will not hundred percent prevent the attack, hopefully for anyone who follow this guideline will be on the safer side and at least minimized the possibility of attack to happen on their web application. Finally, the guideline produce from this analysis can be use for developing a secure web application. This guideline will be a framework for those who are new in this field to prevent themselves from being a targeted attack from this internet attacks.

ABSTRAK

Proses pembangunan aplikasi web membawa masalah yang serius bagi kebanyakan organisasi. Tambahan pula dengan kehadiran pelbagai jenis masalah keselamatan baik dari segi pembangunan web aplikasi itu sendiri mahupun masalah konfigurasi. Pada masa sekarang, serangan dari *worm* mahupun virus terhadap web aplikasi dating dengan pelbagai motif, antaranya ialah untuk mendapatkan akses administrator yang mempunyai lebih fungsi terhadap web aplikasi, ataupun untuk mencuri data, wang dan sebagainya. Semuanya bermula dengan satu serangan mudah ynag akhirnya membawa kepada kemusnahan sistem itu sendiri. Analisis yang dilakukan terhadap *web worm* ini terdiri daripada beberapa peringkat. Bermula daripada menganalisis serangan *worm* tersebut, sehinggalah kepada penghasilan panduan dalam menghasilakn apliksi web yang selamat. Serangan *web* worm ini akan dijalankan ke atas model web aplikasi yang telah dibangunkan dengan menggunakan PHP sebagai bahasa pengatucaran dan MySQL sebagai pangkalan data. Dalam hal ini, kekurangan yang diperolehi daripada imbasan yang dilakukan terhadap aplikasi web akan dibandingkan bersama dengan method serangan yang akhirnya membawa kepada panduan untuk menghalang daripada serangan berlaku. Walaupun panduan ini tidak akan mampu menghalang serangan internet dengan seratus peratus, namun sesiapa yang menggunakan panduan ini diharap akan dapat mengurangkan kemungkinan dari terkena serangan tersebut. Secara tidak langsung, panduan ini akan menjadi garis dasar bagi meraka yang baru dalam bidang ini untuk mengelakkan organisasi dari menjadi bahan serangan dari internet.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 Preface

Web application has become a very popular application nowadays. Lots of people get connected to the internet regularly in order to fulfill their needs through all sorts of web application. From a provider's perspective, such applications, as an example e-banking, e-learning, picture and music sharing are easy to manage, and it is easier if only one application on the web is open for access by people around the world than to manage an application installed at specific clients' computers.

Even though the managing aspect of web application had become easy, its security aspect becomes a big problem to handle. This happens due to the fact that security aspect always comes last during any application development process. Lack of security will lead to vulnerabilities that will open hundreds of ways for malicious code to be implemented in the application. Valuable information can be easily traced, worms and virus can be spread to all clients' computer and major problems may occur from one simple mistake in the development process.

Beside the vulnerabilities in the web application itself, web browser also plays important roles in preventing malicious code attack. Web applications developers also have to be aware on different types of web browser's vulnerabilities, so that any problem that might arise from the weaknesses of a web browser can be handled by the application. Another important key is to keep the web browser updated with the most recent patches available and to use the most current version of

the web browser in order to make sure that the possibilities of being attacked can be minimized.

Recent research by The Gartner Group (2006) estimated about 97 percent of more than 300 website inspected is exposed to various type of web application attacks. Most of the organizations using web application usually are not aware of the attack that could bring damage to their entire network and information system. Even though the consequences of the attack will be on the organization side, the developer should not ignore this potential problem. Developers should aware of the existence of various vulnerable attributes in the application which will lead to malicious code attack.

1.2 Background of Problem

Web application attacks can be divided into two types, namely the client side attack and server side attack. Client side attack is based on malicious code attack to the web browser vulnerability, while server side attack focuses on the web application. Ed Skoudis[1] defined malware as a set of instructions that run on your computer and change your system according to directions given by the attacker. Malware is another name for malicious code which came from the word *malicious software*. There are various types of malicious codes which were being developed to take advantage of different kind of web browser's and web application's vulnerabilities and to enter any particular user's computer.

A lot of major issues arise from the possible damage that could be caused by malicious code, ranging from installing spyware for monitoring user behavior, to secretly damaging user's hard disk on their computer. Most of the web application program developed was focused on its function, leaving out an important part, which

[1] Ed Skoudis with Lenny Zeltser. Malware: Fighting Malicious Code. New Jersey : Prentice Hall. 2004

is the security. The design was poor and lack of security features, such as unavailability of exception handling which will give technical error message to user when anomaly happen during runtime. As a result, it is easier for hacker to find out sequence of vulnerabilities on those web applications.

Beside web application issues arising around internet user, web browser also plays important roles in causing malware attack. As an example, using unpatched or older version of web browser can lead to multiple vulnerabilities. Problem rises when those vulnerabilities lead to remote code execution, which will attack without the needs of user's interaction.

Web application also communicates directly to the server that store millions of sensitive information. Through web browser, one single modification to the web application request to the server can cause all the information be deleted of modified or copied to attacker's hard disk. Information such as username and password, credit card number and other important information can be used by the attacker in a false manner and would jeopardize the owner's safety and confidentiality.

Many companies are using web application in their information system and they are not aware of the web application's vulnerabilities that hackers could exploit. To them, using anti-virus, firewall and other security substance are enough to prevent hacking process from happening, and this is indeed a very dangerous perception.

There are several types of malware that can infect a web application, such as worms and viruses. Different type of malware has different objective carried from the creator. For example, worms actually self-circulate from one host to another, infecting them and do not depend on the language used to develop the web application. Unlike worms, viruses need human interaction to reproduce, for example by opening a file, restarting the system, executing contaminated application and many other ways.

With these kinds of problems appear regularly, something has to be done to keep web application and everything connected to it from getting infected and damaged. The best way to deal with it is to find the vulnerabilities in those web applications and overcome the problem. For example, if the attack came from user input, all user input must be validated thoroughly before accepting it to database.

1.3 Problem Statement

Attacks by different kinds of malicious code lead to different types of problems. Lots of effort has been done to minimize these attacks, but as time goes on, it becomes an obvious fact that the attacking technology is always getting one step ahead than the preventing technology. The problems statements that lead to this topic proposal are as follows:

1. Malicious code attacks on web application appear more frequent lately.
2. Lots of valuable information being stolen everyday because of malicious code attacks.
3. Different programming language leads to different vulnerabilities.
4. All the vulnerabilities can cause major effect on the applications and user.
5. Bad programming practice by developer and the habit of ignoring type of malicious code attacks.

1.4 Aim of Project

To analyze the attacking method of malicious code on web applications, to analyze the common existing vulnerability and to propose a guideline on developing secure web application.

1.5 Project Objectives

This analysis on malicious code will be done to fulfill several objectives. Stated below are all the objectives:

1. To analyze vulnerabilities in various web applications
2. To find out the probability of malicious code attack on web application.
3. To measure processing time of attack via malicious code – this objective cannot accomplish during this study because of the lack of time, hardware and software.
4. To verify the method of malicious code attack on web application.
5. To come out with a guideline to develop secure web application.

1.6 Project Scopes

Project scope limits the analysis that will be conduct to specific types of malicious code and the platform for the application to runs on. The scope is as follows:

1. Analysis on Blaster, Ajax and XSS worm only.
2. Study will be carried out on a web applications model developed using PHP, MySQL as the database and running on Apache Web Server.
3. Study on the method of attack on selected worms.
4. Analysis based on the method available to defend and prevent worms attack on web applications.
5. Use the output from analysis to come out with a guideline for developing a secure web application.

1.7 Importance of the Study

In this day and age, almost every single application can be transform into a web application. Building a web application is not a real problem since there are currently many software developers coming into the scene. The real crisis happen when the software engineers themselves did not aware of the vulnerabilities of the applications they have written, let alone the defect on the web browser. In some situation, some of the defect in browser can be overcome during the application development. But the main problem was arising due to the availability of various types of web browsers used by the users, which lead to difficulties in handling such problem.

This kind of defect will lead to various kind of attack from public once the vulnerabilities get to be known. In addition, there are lots of tools available in the internet that can be use to scan the vulnerabilities on any web application. So, it is indeed a software engineer's responsibility to make sure that their applications are not vulnerable to be attacked from inside and outside of the network.

REFERENCES

Billy Hoffman, *Analysis of Web Application Worms and Viruses*. SPI Labs Security Researcher.

Bryan Sullivan, *Malicious Code Injection: It's Not Just for SQl Anymore*.

Chris Lambert (2003), *Web Application Security*. Boston : MIT Security Camp.

Dancho Danchev (2005), *Malware – future trends*.

Daniel Estermann (2006), *Web Application Security 2.0: How to face current web security problems.*

Dharmesh M Mehta(2004), *Jeopardy in Web 2.0, The Next Generation Web*. The Open Web Application Security Project.

Ed Skoudis with Lenny Zeltser (2004). *Malware: Fighting Malicious Code*. New Jersey : Prentice Hall.

Gartner Group Web Sites

Information Technology Security Report Lead Agency Publication (2006), *Future Trends in Malicious Code – 2006 Report*. Canada : Royal Canadian Mounted Police.

Jose Nazario, with Jeremy Anderson, Rick Wash and Chris Connelly (2003), *The Future of Internet Worms*. Crimelabs Research.

Micheal Cobb (2005), Introduction to Web Application Attacks, SearchSecurity's Web Security School.

Mike Shema (2007), *Web Application Worms: The Future of Browser Insecurity*. InfoSecrurity, New York.

Mohammad Omar Khan (2007), *Automated, self-propagating attacks on custom Web application code.*

Norhazimah Abdul Malek (2005), *Securing Application From Hackers*, Computimes.

Open Web Application Security Project (OWASP)(2003), *OWASP's Top Vilnerabilities in Web Applications.*

Peleus G. Uhley (2003), *Web Browser Vulnerabilities 101*. Anonymizer Inc.

Peter Sayer (2004), *Santy.E Worm Poses Threat to Sites Badly Coded in PHP*, IDG News Services.

Sheeraj Shah (2005), *Web Application Kung Fu, The Art of Defense*. Malaysia : Net-Square Solutions Pvt. Ltd.

Ulfar Erlingsson, Benjamin Livshits, Yinglian Xie (2007), *End-to-end Web Application Security*. Microsoft Research.

Ulrich Bayer, Andreas Moser, Christopher Ktuegel, Engin Kirda(2006), *Dynamic Analysis of Malicious Code*. France : Springer.

Vern Paxson (2005), *Addressing the Threat of Internet Worms.* ICSI Center for Internet Research and Lawrence Berkeley National Laboratory.

WebSense (2004), *Avoiding the Newest Security Threats From Web-Based Attacks*. California : Websense, Inc.