PandaLabs Bulletins:

Legitimate Webs in jeopardy

www.pandasecurity.com/homeusers/security_info | www.pandalabs.com

Index

1.- Introduction

Although awareness regarding Internet threats has evolved, many users still believe that if you keep away from dubious Web pages you will avoid malware infection.

Malware on the Internet is usually associated with malicious or suspicious Web pages, but not with legitimate ones.

However, since no system is 100% secure, cyber-crooks take advantage of the trust users have of specific domains to drop malicious code on their systems.

This technique is mainly being used to spread malware, but it could also be employed to distribute spam or store stolen data.

This article provides an insight into these types of attacks, specific cases and possible future trends. Additionally, several tips have been included to prevent you from falling victim to these attacks.

2.- Modus operandi

Legitimate Web page infection consists of modifying the Web page source code by adding an iframe-type reference pointing to a malicious server.

Cyber-crooks can infect Web pages in several different ways:

1. By exploiting vulnerabilities in the software installed on a server.
2. Through bad configuration of the programs installed and running.
3. By stealing passwords for accessing the server using Trojans.

At least one of the above conditions must be met to modify the Web page source code.

These techniques allow cyber-crooks, in addition to infecting the corporate website, to use the servers for a range of malicious actions, including hosting a program designed to infect visitors, distributing spam or storing stolen data.

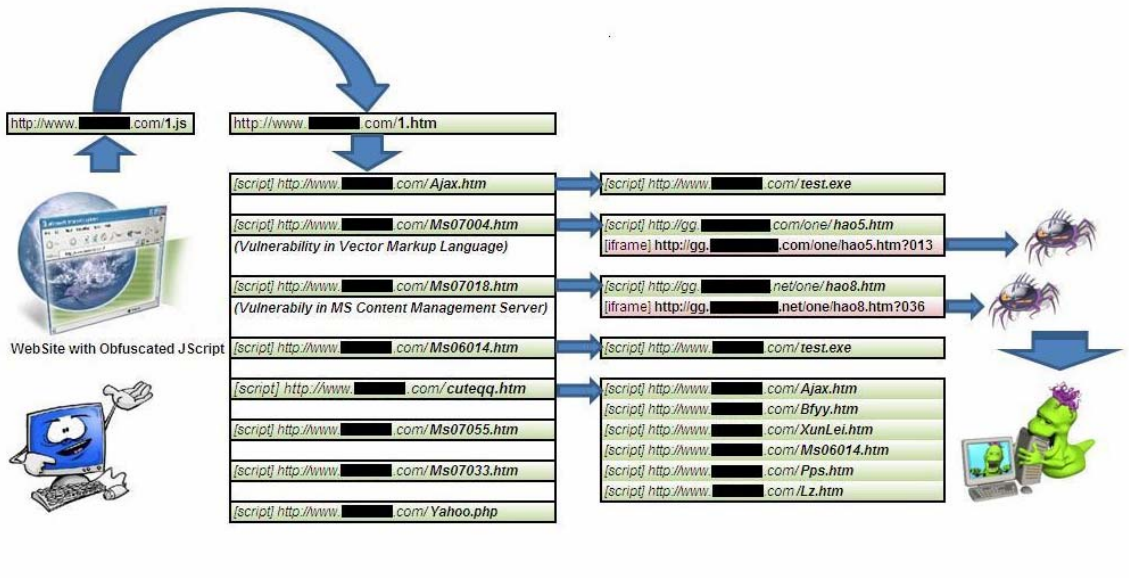
Once they manage to access the Web page, cyber-crooks add an iframe-type reference at the end of the file loaded by default, pointing to the malicious server. Initially, users don't suspect a thing since the modification is made on the HTML code of the legitimate Web page, and is invisible to users.

This way, when users visit a Web page to which cyber-crooks have added a malicious iframe, the iframe establishes a connection (transparent to users) with a page that checks the computer for specific vulnerabilities. If the computer is updated against the vulnerabilities, users will not be infected. If it isn't, the malware –which is normally designed to steal passwords- will be automatically downloaded.

The fact the malware distributed is a password stealer is not a coincidence, since maximum efficiency would land large financial benefits in cyber-crooks' hands. This malware captures all types of confidential information (passwords, user names, email addresses) which hackers can use for subsequent fraudulent actions.

As for the exploits used to infect users, initially they were often related to the operating system. However, users that had applied the corresponding patches to their operating systems weren't infected other than by zero-day exploits for which no patch had been released. Consequently, attackers began to widen their scope, targeting browsers such as Internet Explorer or Firefox, and popular applications; Windows Media Player, QuickTime, Acrobat, Flash Player, etc.

The diagram below has been included to better explain these types of attacks:



Example of an iframe-type attack

The process described is as follows:

1. To start with, several legitimate Web pages have been modified through the insertion of a malicious iframe, *http://www.*<blocked>*.com/1.js* in the example.
2. When users visit an infected legitimate Web page, the iframe connects to a Web page, in this case: *http://www.*<blocked>*.com/1.htm.*
   This connection is imperceptible to users.
3. This Web page has a list of vulnerabilities it tries to exploit on the affected system.
4. Upon detecting a vulnerability, the malware is downloaded.

3.- The most notable case

One of the most notable trends this year has been the SQL Injection attacks affecting hundreds of thousands of servers. These types of attacks have enabled iframe insertion on Web pages. Numerous compromised servers were detected at the beginning of April. Their pages were modified to include an iframe that pointed to a server which exploited several vulnerabilities, such as:

- MS06-014: Vulnerability in the Microsoft Data Access Components (MDAC) Function.
- MS07-004: Vulnerability in Vector Markup Language.
- MS07-018: Vulnerabilities in Microsoft Content Management Server.
- MS07-033 : Cumulative Security Update for Internet Explorer

- MS07-055 : Vulnerability in Kodak Image Viewer.

These vulnerabilities were exploited to distribute different types of malware.

These were highly organized SQL Injection attacks. Due to the large number of affected servers, the attack must have been automated, using a tool specifically developed to scan servers and analyze the chances of SQL Injection attacks on each server.

One of the most successful SQL Injection attacks was detected at the beginning of April and affected half a million Web pages. A problem programming specific asp-type pages allowed the insertion of the malicious iframe on hundreds of thousands of pages.

4.- Trends

When these cases came to light, users panicked, since millions of legitimate and reliable pages were infected.

Despite the seriousness of the situation, the number of affected pages has decreased significantly, since their administrators took the necessary measures to resolve the problem and prevent their servers' pages from being infected.

The number of infected pages is expected to continue decreasing, since Web administrators are now better informed. However, if a zero-day exploit appears, until the corresponding patch is published, cyber-crooks could revive these attacks.

On the other hand, before these attacks appeared, several kits for installing malware through exploits were developed, e.g. Mpack. These types of tools are designed to exploit vulnerabilities for malware distribution. In this case, they are pages designed by cyber-crooks who usually use names similar to the legitimate ones to fool users.

5.- How to ensure you are protected?

From a user's point of view, to avoid infection systems must be up-to-date against known vulnerabilities, and complemented with an antivirus with up-to-date proactive technologies.

From an administrator's point of view, servers must be up-to-date so no vulnerabilities appear, and administrators must make sure the pages that can access their database are correctly programmed and the server passwords are frequently modified to prevent people with malicious intentions from using them.