

WEB ATTACKS AND COUNTERMEASURES

February 2008

© The Government of the Hong Kong Special Administrative Region

The contents of this document remain the property of, and may not be reproduced in whole or in part without the express permission of the Government of the HKSAR.

Disclaimer: Whilst the Government endeavours to ensure the accuracy of the information in this paper, no express or implied warranty is given by the Government as to the accuracy of the information. The Government of HKSAR accepts no liability for any error or omission arising from or related to the use of the information.

TABLE OF CONTENTS

Summary	2
I. Contemporary Web Attacks.....	3
Trends & Potential Impacts.....	3
II. Behaviour of Attacks	5
III. Countermeasures and Responses	7
Guidlines for Web Application Owners	7
Tips for End-users.....	9

SUMMARY

Web applications are vulnerable to attacks from the moment they go online. Over the past few years, we have witnessed an explosion in the number of web attacks that exploit vulnerabilities in web servers, and programming flaws in web applications. More recently, end-users and their workstations have become the latest targets for web attacks focused on the growing sphere of the Internet communities¹ such as MySpace, Facebook, Wikipedia, as well as other community chat rooms, discussion forums, and so on.

As more ingenious attack strategies and schemes appear on the Internet, end-users and the organisations that provide web services need to protect their systems from being compromised. These could in turn become a weapon for attacking other machines.

¹ <http://www.firstmonday.org/issues/issue4/valauskas/>.

I. CONTEMPORARY WEB ATTACKS

TRENDS & POTENTIAL IMPACTS

In addition to exploiting the vulnerabilities inherent in web servers, or making use of the loopholes and flaws within web applications, attackers are also taking advantage of the trust their victims have in the sites they visit. Victims of web attacks are either tricked into accessing a malicious website, or redirected to a malevolent site when they access sites providing popular information such as music, movies, collectables, and so on. Organisations and individuals who do not safeguard their computer systems properly run the risk of considerable financial loss or destruction of reputation. Key examples of major web attacks that target end-users or their PCs are described below:

1. The 'Italian job' Web attack

In June 2007, more than 10,000 websites, including many Italian government websites, were compromised. Infected websites had a short piece of HTML "iFrame" code inserted that would redirect visitors to another website, where a malicious JavaScript would install a keylogger and a Trojan downloader program on their PCs to test and see if they could be compromised further^{2,3}.

2. The MySpace Phish / Drive-by attack

Also in June 2007, several hundred MySpace profiles were discovered injected with links to phishing⁴ sites. Users of MySpace ran the risk of being infected

² http://www.infoworld.com/article/07/06/18/italian-job-Web-attack-hits-10000-sites_1.html

³ <http://blog.trendmicro.com/another-malware-pulls-an-italian-job/>

⁴ <http://www.scmagazineus.com/MySpace-users-warned-of-drive-by-exploit-attack/article/35125/>

when they visited any MySpace profile page containing malicious JavaScript that would silently redirect them to a malicious site attempting to exploit the vulnerability in Internet Explorer. A commonly known proxy network bot, “flux bot”, would be installed in an attempt to hide the phishing sites behind constantly changing proxy servers⁵.

3. Cross-Site Scripting (“XSS”) Worms

In October 2005, an XSS vulnerability in MySpace was exploited by the author of the Samy worm who was able to upload his infected XSS code to his personal profile page on MySpace. When other authenticated MySpace users viewed Samy’s profile, the worm forced their web browsers to add Samy as a friend, and alter their profiles with a copy of the malware code. The Samy worm continued to spread exponentially when a user viewed Samy’s or any other infected users’ profiles. More than one million MySpace user profiles were infected⁶ this way.

4. Other attacks

Phishing can be termed a social engineering attack whereby criminals attempt to lure unsuspecting web surfers into logging into a fraudulent website that looks like a real website, such as eBay, or the website of an online bank⁷.

Internet search engines can also help web attacks. In December 2004, the web worm Santy.A exploited the vulnerability in the bulletin board software phpBB. Instead of randomly guessing a target IP address, the worm used the Google search engine to help find new vulnerable targets in order to launch defacement attacks via the vulnerability in phpBB⁸.

⁵ <http://isc.incidents.org/diary.html?storyid=3060>

⁶ <http://www.whitehatsec.com/downloads/WHXSSThreats.pdf>

⁷ <http://googleonlinesecurity.blogspot.com/2007/06/thwarting-large-scale-phishing-attack.html>

⁸ <http://isc.sans.org/diary.html?date=2004-12-21>

II. BEHAVIOUR OF ATTACKS

Web attacks like the Italian job, MySpace phish / drive-by attack and other XSS worms roughly follow this pattern:

1. The attacker locates a web server with a vulnerability that he/she can leverage to launch an XSS or code injection attack.
2. The attacker performs either of the following actions:
 - a. They succeed in inserting code (e.g. JavaScript code) in the vulnerable web server that allows a cross-site scripting attack to take place against client users connecting to the victim's web server; or
 - b. They create a URL embedded with malicious script in a website with an XSS vulnerability. By enticing a target user to click on this URL, an embedded script would run on the user's browser causing more malignant attacks, such as downloading a Trojan horse or sending cookie information to the attacker.

In the Samy worm case, the malicious code stayed and infected authenticated users only within the MySpace community, which had been large enough for the spread of the worm. In some cases, the malicious code could not connect to servers outside.

During a phishing attack, victims are tricked into giving out their identities, credit card numbers and even login credentials for bank accounts through social engineering channels such as emails. No compromise in the security of the legitimate website is needed, and it simply involves setting up a fraudulent website and throwing out bait to catch out careless or unsuspecting users who fall into the trap.

The convenience and accuracy of Internet search engines now enables exploitative code to find new targets much more easily and more accurately than the random IP guess approach. In addition, if an organisation's sensitive information is not properly protected, Internet search engines might be able to index such information. If the information involved appears on a user's screen, within a search context, data leakage may well results.

III. COUNTERMEASURES AND RESPONSES

As more ingenious web attacks appear on the Internet, end-users as well as organisations providing web services need to protect their systems from being compromised, and which could in turn become a weapon to attack other machines. Appropriate actions from both end-users and web application owners are required.

GUIDLINES FOR WEB APPLICATION OWNERS

To avoid being exploited by attacks targeting web applications, certain technical measures can be implemented to help prevent and detect any abnormal incidents. As there is no guarantee of a perfectly secure website, a proper incident handling procedure should be implemented.

It might be some time before the operator of a web application is aware that the website has been compromised, or that customer security has been breached after visiting the website. In many cases, it is third parties like customers who first report that the website hosting a web application might have a problem. In the case of a phishing attack, the fraudulent website is often hosted under a different jurisdiction. Operators of the genuine website can only warn customers not to visit fraudulent websites which might look similar to the legitimate site. Another possible action is contacting the Internet Service Provider hosting the fraudulent website in the hope that they can take it offline.

Studying the system and application logs may help in uncovering web attack incidents. In the XSS worms case described earlier, the victim's MySpace pages only played a role in

directing a customer to a malicious website without any trace of hacking being left on a customer's PC. Consequently, a victim's page or website needs to include a way to trace pages that have been exposed to any cross-site scripting attack, and be able to clean up any infected pages so as to stop further infection.

An attacker needs to be able to insert malicious code into a victim's web application before attacks like XSS worms or similar are successful. To prevent this from happening, malicious user input into the web applications needs to be sanitised. In the case of the Samy worm, MySpace did have user input validation systems in place, but these proved to be inadequate. In addition to removing special characters from allowed input character sets, and encoding dynamic output elements, a white-list approach should be followed. In a white-list approach, only inputs matching pre-defined patterns are allowed through, while all others are filtered out. Compared to the black-list approach where pre-defined invalid character sets or patterns are blocked, the benefit of a white-list approach is that it enables the web application to allow through exact approved inputs, something that cannot be guaranteed in the black-list approach.

An incident detection and monitoring mechanism to expose, contain and prevent security incidents should be established. System logs and other supporting information should be retained and archived to provide proof when tracing back through security incidents. To prepare for a worst case scenario, a security incident handling and reporting procedure applicable to the web application should be established, documented and maintained. Awareness training should be conducted for all staff to ensure that they are fully aware of handling and reporting procedures for security incidents. Immediate follow-up action is required for any suspected system intrusion, and should follow procedures laid out in the security incident handling and reporting guidelines.

In addition, web-based information systems should be periodically evaluated by auditors employed by an independent, trusted third party to determine whether the minimum set of

controls required to contain risks at an acceptable level is being maintained. Security risk assessments should also be performed prior to any major enhancement or change to web systems and/or web applications.

An additional possible preventive measure is hiring outside professionals to periodically check for the existence of fraudulent websites over the Internet. Customers and website users can be notified immediately as soon as fraudulent websites posing as the legitimate site are discovered. This can help minimise phishing attack incidents.

Other possible technical and administrative measures can be found in the paper on “Web Application Security”.

TIPS FOR END-USERS

To avoid your PC being compromised and becoming a weapon to attack other machines, web application and the Internet users are advised to:

1. ensure that your operating system and key system components such as the web browser is fully patched and up to date;
2. install a personal firewall along with anti-virus tools with the latest virus signatures that can detect malware such as keyloggers,;
3. employ different sets of login and password combinations for different web applications and services you use;
4. regularly change your passwords in critical web applications if a one-time password system is not supported; and
5. turn off all JavaScript or ActiveX support in your web browser before you visit any unfamiliar websites.