



The *Spambush* Trojan

A VigilantMinds White Paper

Abstract

The *Spambush* Trojan exists to facilitate the propagation of spam emails via zombie computers that it creates without the knowledge of the end user. The author of the code went to extreme lengths in order to fly under the radar of several popular intrusion detection systems and anti-virus solutions in order to be undetectable. The virus will only attach itself to users with a vulnerable Internet Explorer browser version 6.0 (prior to Hotfix Q822925) and an operating system that utilizes the New Technology File System (NTFS).

The initial attack vector occurs through a “spamvertisement” (a spam e-mail advertising a product) or banner ad that lures unsuspecting users. The attack is initiated when an unsuspecting user follows the link to a web page included in this banner ad or spam. Compromise is accomplished through a combination of JavaScript and vbscript. This attack goes completely unseen by the end-user due to the covert method with which the vbscript is executed. The device then “advertises” to a spammer / hacker website that it has been successfully compromised by posting an external IP address. This is a notification for potential spammers to initiate compromise through this device. Once compromised, spam is generated by an internal SMTP engine and disseminated. The specifics of the dissemination are dependent upon the originating spammer.

The ability of the exploit code to remain undetected during the process of compromise and propagation give this Trojan a rating of **HIGH SEVERITY**. The stealth process used by the *Spambush* Trojan increases the likelihood of continued compromise. The possibility of millions of compromised machines is likely. Though the primary function of this Trojan is spam propagation, the possibility exists that the device could be used in order to launch further attacks. Variations of this Trojan have already been observed with attacks that include further compromises, key logging, and denial-of-service attacks.

In order to best understand this Trojan, VigilantMinds purposely directed an unpatched IE web browser (Version 6.0 - Before Hotfix Q822925) to allweb.dreamhost.com. The server was fully patched, with the exception of Microsoft Internet Explorer web browsers, and was utilizing Windows XP Professional as the underlying operating system. VigilantMinds then recorded a step-by-step compromise analysis that can be located in Appendix A.

This Trojan impacts Microsoft Internet Explorer 6.0 browsers (prior to Hotfix Q822925) and below. VigilantMinds recommends verifying that all devices that utilize Internet Explorer are patched with the most current available updates. In addition, determine that all anti-virus signatures are current and up to date.

VigilantMinds has updated all its proprietary NetXone sensors to detect compromises in progress, and devices that have already been compromised. VigilantMinds is working closely with the FBI and other organizations to combat the effects and propagation of this malicious program.

This white paper describes the steps used to identify the characteristics of the *Spambush* Trojan and provides tested remediation steps to cleanse the Trojan from a compromised system.

Spambush Trojan Overview

The *Spambush* Trojan detailed in this whitepaper has several components that work together to accomplish the desired outcome: compromising vulnerable benign systems into becoming spam sending systems. Some of these components are the result of directed intent to harm vulnerable systems and others have been caught up in the activities of the Trojan as innocent by-standers.

Server-Side – Computer Code

1. A web browser vulnerability checking script (observed as counter.js in this case)
2. A client-machine “infector” script (observed as vbscript in a .hta file)

Client-Side – Computer Code

1. An initial compromising “boot-strap” executable (observed as audio.exe in this case)
2. A follow-up “spam engine” executable (a randomly named seven-letter DLL installed as an ADS of the system32 folder)

Websites – Innocent By-standers

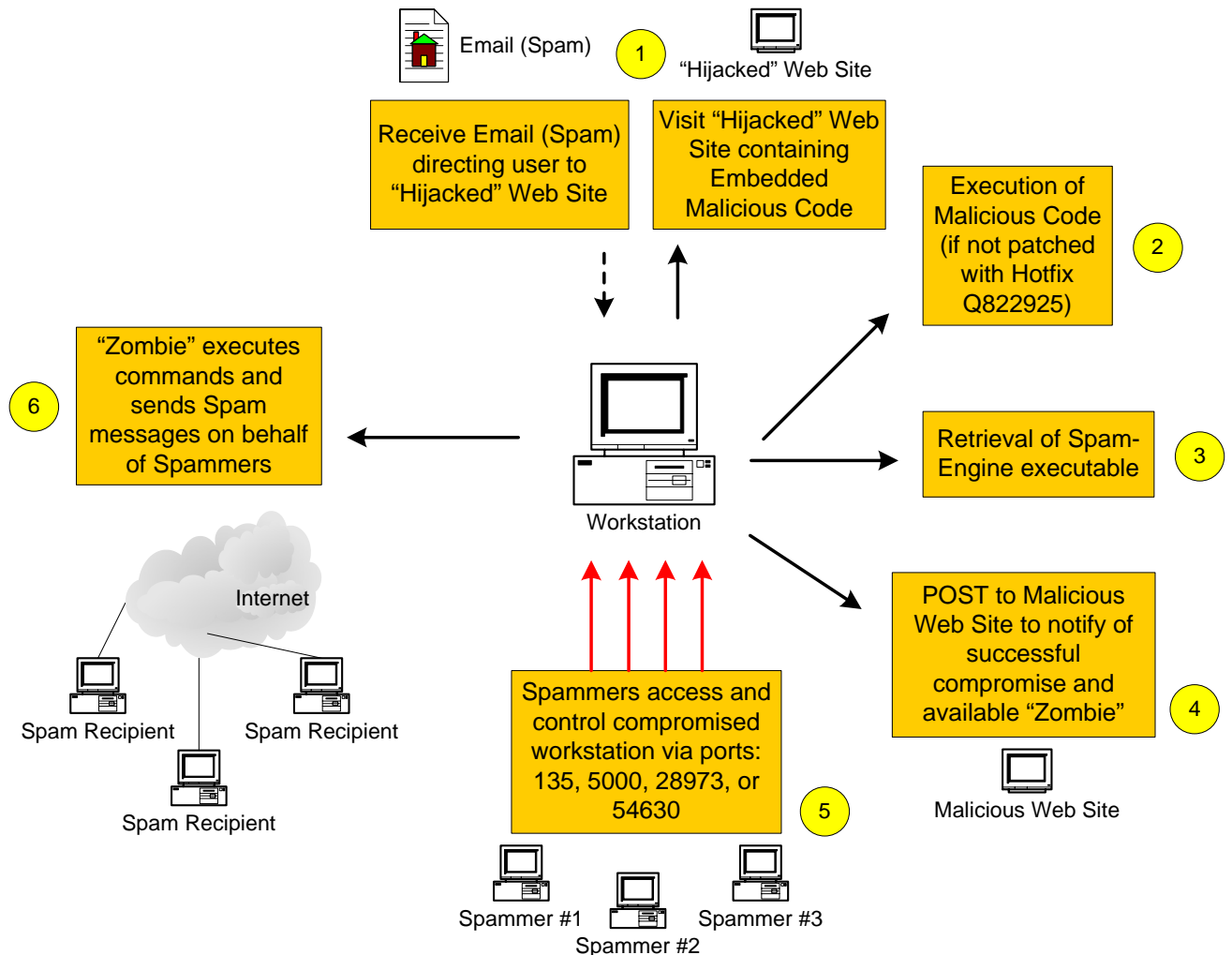
1. Legitimate website copied for use by the hijacked web-site (observed as www.truesuite.com & www.trueswitch.com in this case)

It is important to mention that at no time during this investigation did VigilantMinds find www.truesuite.com or www.trueswitch.com voluntarily involved in the compromise or propagation of this Trojan. These sites were unfortunately copied for their appearance and content in an attempt to make the hijacked website appear legitimate.

Websites – Complicit in the Attack

1. Hijacked website – provider of the browser vulnerability checking script, client-machine “infector” script, and the initial compromising “boot-strap” executable (observed as allweb.dreamhost.com in this case)
2. Spam enabling website – provider of the “spam-engine” executable (observed as YaMX.com in this case)
3. Compromised system reporting website (observed as 66.98.188.91 in this case)

The following graphic illustrates the Spambush Trojan's path of compromise.



The following steps detail the path of the attack.

1. A vulnerable Microsoft Internet Explorer (IE) browser is lured to a "hijacked" website (containing copied page contents from a legitimate web site) via spam or a pop-up window.
2. The "hijacked" website (decoy) determines if the browser is vulnerable and infects it with the "boot-strap" executable.
3. The "boot-strap" executable contacts the spam-enabling website and the "spam-engine" executable is downloaded and installed on the compromised system.
4. The "spam-engine" executable announces to a compromised system reporting website that the host system has been breached and is available to spammers.
5. Once the system announces that it has been compromised, spammers or other malicious individuals gain access and control of this workstation.
6. The workstation is fully compromised and becomes a link in the propagation of various spam chains.

Any workstation running Microsoft Internet Explorer 6.0 - prior to Hotfix Q822925 –and utilizing NTFS, is vulnerable to this ambush, and would have its system threatened or compromised in a number of ways, including:

- Administrative level ownership of the system by a remote intruder
- Advertisement of the system as a compromised system
- Use of the system in the generation of spam e-mails

Initial research and testing indicates that attackers are primarily utilizing the compromised workstation as a spam engine. However, variants of the *Spambush* Trojan have already been seen that use the compromised systems for further malicious activities, such as key logging. VigilantMinds is continuing its research in these areas.

Anti-Virus Detection

The *Spambush* Trojan client-side computer code as first observed by VigilantMinds consists of two main components:

3. An initial compromising “boot-strap” executable (observed as audio.exe in this case)
4. A follow-up “spam engine” executable (a randomly named seven-letter DLL installed as an ADS of the system32 folder)

An up-to-date anti-virus solution is necessary in detecting the initial compromising “boot-strap” executable. An anti-virus program that is actively running should detect the initial compromising “boot-strap” executable and halt its execution. If an actively running anti-virus solution does not detect the “boot-strap” executable file, a manual anti-virus search should be initiated on the hard drive. Several popular anti-virus solutions were run against a device that was compromised in the VigilantMinds testing lab, with various conclusions:

<u>Anti-Virus Solution</u>	<u>Detected Name</u>	<u>File found</u>	<u>Remediation</u>
Sophos	Troj / Brok-A	audio.exe	delete file
TrendMicro	Troj Brok.A	audio.exe	delete file
Norton's Corp. Edition	download.Trojan	audio.exe	delete file

Anti-virus solutions will successfully pick up the “boot-strap” executable portion of the Trojan, though they have different interpretations of what the Trojan is. The contents (code) of the “boot-strap” executable are the main trigger for the signatures; however, **none of the applications address the “spam engine” of this Trojan**, which is a randomly named seven-letter DLL file. This file is hidden as an ADS (Alternate Data Stream) of the system32 directory itself. ADS's are similar to invisible attachments to a file or a directory. A directory list would not include the ADS, nor would an ADS show up in a search of a Windows Explorer folder. The Removal Instructions section on the next page identifies steps to locate, view, and remove the ADS file.

All three anti-virus solutions detected and deleted the “**boot-strap**” executable file. After a reboot, **a follow-up scan was then run on the device, and each of the solutions claimed that the device was no longer compromised. However, process listings confirmed that the virus was still functioning and was attempting to communicate outbound.** The Trojan remained in memory and was attached to three different processes: explorer.exe, wpabaln.exe, agentsvr.exe.

As an additional preventative measure, VigilantMinds designed and implemented two intrusion detection signatures that are capable of identifying the work performed by the “boot-strap” executable during different stages of its processing. These two intrusion detection signatures (or NetXone signatures to VigilantMinds clients) are located below.

```
alert TCP $OWNED any -> $EXTERNAL_NET 80 ( sid:1000648; rev:1; uricontent:"GET";  
uricontent:"/test/tracker.exe"; flow:to_server,established; classtype: VM; )
```

//This signature identifies the compromise in progress

```
alert TCP $OWNED any -> $EXTERNAL_NET 80 ( sid:1000649; rev:1;  
uricontent:"POST"; uricontent:"/cgi-bin/ref.cgi?"; flow:to_server,established; classtype:  
VM; )
```

//This signature identifies a compromised device attempting to signal an external IP that it is currently available and compromised.

If a previously compromised machine enters a monitored environment, it will trigger one of the two signatures created by VigilantMinds.

Removal Instructions

The complete removal of this particular Trojan requires identifying and removing the “spam-engine” executable stored as an ADS on the compromised system. First, the “spam-engine” executable must be identified. It was observed installed as a randomly named seven-letter DLL. Follow these steps to identify the “spam-engine” executable:

1. Run an anti-virus solution on the hard drive of the device. The application will identify the “boot-strap” executable **only**. Allow the anti-virus software to eliminate the “boot-strap” executable.
2. Click the **Start** button on your desktop. Select **Run**. Type **regedit** and click **OK**.
- 3 The Registry Editor window will open. Navigate to the following registry folder:
HKEY_Local_Machine\Software\Microsoft\Windows\CurrentVersion\Run
4. In the Run folder look for an entry that looks similar to the example below.

rundll32 C:\Windows\system32\xxxxxxx.dll

5. Record the name of this file.

6. To view the contents of this file, click the **Start** button. Select **Run**. Type the following in the Run command prompt in order to view this ADS file. **Note:** replace the “xxxxxxx” with the file name that was recorded in Step 5.

notepad C:\Windows\system32\xxxxxxx.dll

7. Once the file has been identified and if desired, viewed; click the **Start** button. Select **Run**. Type the following in the Run command prompt in order to remove this ADS file. **Note:** replace the “xxxxxxx” with the file name that was recorded in Step 5.

rundll32 C:\Windows\system32\xxxxxxx.dll, Uninstall

Click **OK** to prompt the removal of the ADS.

8. Reboot the computer.

9. Execute the command identified in Step #6 to verify that the file has been eliminated. Notepad should display a message indicating that the file does not exist. Cancel out of Notepad without saving after getting this message.

Post Compromise

The individual who controls the compromised host can specify the type and destination of the spam. For the *Spambush* Trojan documented in this white paper, at no time during the compromise did the device attempt to compromise other devices. However, variants of this Trojan that do attempt further malicious activity are currently under investigation by VigilantMinds.

At no time during the investigation did TrueSuite.com or TrueSwitch.com appear to be enabling this type of activity. Only allweb.dreamhost.com, yamx.com, and the various other IP addresses that were noted actively participated in the compromise and propagation of the *Spambush* Trojan.

Note: It is likely that code embedded into the allweb.dreamhost.com website is a common tool that is utilized by spammers to hijack thousands of other websites attempting to achieve the same results.

After the compromised machine was removed from the Internet - therefore severing the connection to the outside world - various tools were utilized in an attempt to connect to the compromised box. The scans revealed the following open ports; 135, 139, 445, 1025, 5000. Note that these ports may vary due to the nature of the compromise. Subsequent tcpdumps revealed that other ports were open and communicating to other external devices. Several internal attempts were made to further exploit the device via the RPC DCOM exploit that was observed during the tcpdump. After numerous efforts, no connections were established. **The spammers that compromise the machine do not want the device to be utilized by hackers and / or other hacker communities. However, this DOES NOT mean that the compromised device cannot be used as a platform for other types of attacks in the future.**

VigilantMinds identified several files that imbed themselves into the compromised workstation. The Trojan will also attach itself to the following legitimate processes:

Windows/system32/explorer.exe
Windows/system32/wpabaln.exe

Windows/msagent/agentsvr.exe

The provided removal instructions will remove all files and attachments to processes that are installed by this Trojan.

Conclusions

The *Spambush* Trojan is a particularly difficult Trojan to identify and successfully remove from a compromised system due to the covert nature of the JavaScript and vbscript. The end user will likely be unaware that a compromise is taking place. An organization with a reliable intrusion detection system in place, coupled with an actively running and up-to-date anti-virus solution is best positioned to identify this Trojan as it attempts to compromise a system. However, it may be necessary to manually initiate an anti-virus search of a machine's hard drive in order to locate the Trojan files. As testing indicated, simply rebooting a machine after the anti-virus ran turned up the false indicator that the device was no longer infected, even though the Trojan remained in memory and was attached to three different processes. The Removal Instructions in this white paper provide tested solutions to identifying and removing the Trojan files.

VigilantMinds feels that this type of compromise will become more common and could very likely be used as the transport mechanism to enable the next mass attack on Internet connected systems. As operating system vendors continue to address vulnerabilities and weaknesses in the underlying operating systems used by the majority of users, Trojan developers will increasingly target weaknesses within the applications running under these operating systems.

As more Trojans of this nature emerge, the need for reliable intrusion detection and intrusion prevention manners will increase. VigilantMinds recommends a proactive plan for intrusion detection and prevention, a tested anti-virus solution, and an active anti-spam solution to reduce the possibility of an end user falling prey to the 'latest' variant of the Trojan.

Appendix A - Step-by-Step Spambush Compromise Notes

The following is a description of events that occurred in the VigilantMinds' test lab on December 31 2003 regarding the Spambush Trojan compromise. VigilantMinds conducted a thorough tcpdump (packet sniffing) during the course of the investigation. This tcpdump initiated when the device first contacted the malicious website, and terminated once the device was fully compromised and was generating spam.

1. VigilantMinds purposely (with the intent to infect) directed an un-patched IE web browser (Version 6.0 - Before Hotfix Q822925) to allweb.dreamhost.com. The server was fully patched, with the exception of Microsoft Internet Explorer web browsers, and was utilizing Windows XP Professional as the underlying operating system.

VigilantMinds noted that as of January 20, 2004, the allweb.dreamhost.com web site was no longer accessible.

2. The default page of allweb.dreamhost.com is called index.html and contains a reference to a JavaScript file called **counter.js**, which is executed on the vulnerable client browser.

Approximately 95% of the contents of this **index.html** page are identical to this page: www.truesuite.com/products_trueblock.htm. It even contains near identical links of the website. For instance, truesuite.css as well as numerous TrueSuite.com images in gif format are retrieved from IP 64.90.160.222 (www.truesuite.com).

3. The file **counter.js** runs a directed set of checks against the user's browser type and version to determine if it is Microsoft Internet Explorer 6.0 prior to Hotfix Q822925.

If the user is running Microsoft Internet Explorer 6.0 prior to HotFix Q822925, **counter.js** creates a popup window that is programmed to display at a size of one pixel by one pixel. This window flashes quickly and is virtually invisible to the end user.

4. The body of this "invisible" popup window is set to allweb.dreamhost.com/index.htm, which does not appear to exist on the server, but instead gets redirected to allweb.dreamhost.com/index.htm.hta.

5. allweb.dreamhost.com/index.htm.hta then runs a vbscript in the user's browser that saves an executable (**audio.exe**) to the %WINDOWS SYSTEM% directory. It then executes this file using the Windows Scripting Host shell object.

6. The browser is then directed to another site at IP address 207.44.244.61 (YaMX.com) and retrieves the file: /test/tracker.exe. The device is then directed to save this file as **C:\Windows\system32:xxxxxxx.dll** (the file will be a randomly generated seven character string). The file is saved as an ADS (Alternate Data Stream) file within the system32 folder, making it virtually impossible to locate with conventional tools.

7. The browser then posts to IP address 66.98.188.91 (no website /Everyone's Internet Inc.) the following; POST / cgi-bin/ref.cgi?Tue%20Dec%2030%2012%3A54%3A09.218%202003 HTTP/1.0. This post gives the IP address a timestamp / identification number that references the newly compromised host. IP address 66.98.188.91 then disseminates the IP address of the compromised host to various locations throughout the Internet, notifying other attackers of the newly compromised host.

8. Shortly after the post, a barrage of SYN packets from various external IP addresses hit the compromised machine. These hosts will rotate through a multitude of possible ports in order to identify its listening port. Note: periodically, the compromised box will report back to 66.98.188.91 and repost the previously noted string. This allows the IP (66.98.188.91) to maintain a current list of available compromised devices.
9. IP address 81.202.35.190 / ANeuilly-109-1-4-168.w81-49.abo.wanadoo.fr, was the first to initiate a full exploit against the compromised device. The IP address (81.202.35.190) communicates and exploits the device via port 135 (RPC DCOM). Please note that the Windows operating system had been patched for the known RPC vulnerability prior to the testing. The initial exploiter then shuts the RPC services down and logs out of the box. Moments later, the malicious intruder returns and communicates via port 5000 (upnp).
10. Throughout the course of the compromise process, various hosts attempt to contact the device, and numerous TCP RST packets are sent back to these hosts for ports that the box is not listening on. VigilantMinds compiled a list of over 50 possible listening ports. For the hosts that receive TCP SYN/ACK packets, the communication was observed on ports 135 and 5000. The compromised machine received various instructions via these two ports from numerous IP addresses.
11. Approximately ten minutes from the beginning of the tcpdump, the compromised device's first SMTP attempt to 66.159.80.150 / mercury.toad.net is conducted. The internal spam engine is then continuously running and generates spam with a pornographic or prescription pill nature. This comprises the bulk of the traffic, with the occasional IP address attempting to determine which port is open on the compromised box.

Appendix B – Other References

VigilantMinds conducted searches on specific characteristics of the Trojan via the Internet. This Trojan is comparable to the Aflooder Trojan that is explained in the following links:

<http://forums.spywareinfo.com/index.php?showtopic=10456&hl=aflooder>
<http://www.helpdesk.umd.edu/virus/alerts/aflooder.shtml>