ZEUS **https://github.com/Visgean/Zeus**

WRITTEN in C

Zeus, ZeuS, or Zbot is a Trojan horse malware package that runs on versions of Microsoft Windows. While it can be used to carry out many malicious and criminal tasks, it is often used to steal banking information by man-in-the-browser keystroke logging and form grabbing. It is also used to install the CryptoLocker ransomware.[1] Zeus is spread mainly through drive-by downloads and phishing schemes. First identified in July 2007 when it was used to steal information from the United States Department of Transportation,[2] it became more widespread in March 2009. In June 2009 security company Prevx discovered that Zeus had compromised over 74,000 FTP accounts on websites of such companies as the Bank of America, NASA, Monster.com, ABC, Oracle, Play.com, Cisco, Amazon, and *BusinessWeek*.[3] Similarly to Koobface, Zeus has also been used to trick victims of tech support scams into giving the scam artists money through pop-up messages that claim the user has a virus, when in reality they might have no viruses at all. The scammers may use programs such as Command prompt or Event viewer to make the user believe that their computer is infected.[4]

PHP Malware Finder **https://github.com/nbs-system/php-malware-finder**

WRITTEN in PHP

```
 _______  __   __  _______
|  ___  ||  |_|  ||       |
| |   | ||       ||    ___|
| |___| ||       ||   |___   Webshell finder,
|    ___||       ||    ___|   kiddies hunter,
|   |    | ||_|| ||   |        website cleaner.
|___|    |_|   |_||___|
```

Detect potentially malicious PHP files.

What does it detect?

PHP-malware-finder does its very best to detect obfuscated/dodgy code as well as files using PHP functions often used in malwares/webshells.

The following list of encoders/obfuscators/webshells are also detected:

- Best PHP Obfuscator
- Carbylamine
- Cipher Design
- Cyklodev
- Joes Web Tools Obfuscator
- P.A.S
- PHP Jiami
- Php Obfuscator Encode
- SpinObf
- Weevely3
- atomiku
- cobra obfuscator
- phpencode
- tennc
- web-malware-collection
- webtoolsvn
- novahot

Of course it's **trivial** to bypass PMF, but its goal is to catch kiddies and idiots, not people with a working brain. If you report a stupid tailored bypass for PMF, you likely belong to one (or both) category, and should re-read the previous statement.

How does it work?

Detection is performed by crawling the filesystem and testing files against a set of YARA rules. Yes, it's that simple!

Instead of using an *hash-based* approach, PMF tries as much as possible to use semantic patterns, to detect things like "a $_GET variable is decoded two times, unziped, and then passed to some dangerous function like system".

WiFiPhisher (https://github.com/wifiphisher/wifiphisher)   written in Python

About

Wifiphisher is a security tool that mounts automated victim-customized phishing attacks against WiFi clients in order to obtain credentials or infect the victims with malwares. It is primarily a social engineering attack that unlike other methods it does not include any brute forcing. It is an easy way for obtaining credentials from captive portals and third party login pages (e.g. in social networks) or WPA/WPA2 pre-shared keys.

Wifiphisher works on Kali Linux and is licensed under the GPL license.

MALTRAIL   written in PYTHON

https://github.com/stamparm/maltrail

https://github.com/stamparm/maltrail/blob/master/README.md

Introduction

Maltrail is a malicious traffic detection system, utilizing publicly available (black)lists containing malicious and/or generally suspicious trails, along with static trails compiled from various AV reports and custom user defined lists, where trail can be anything from domain name (e.g. zvpprsensinaix.com for Banjori malware), URL (e.g. http://109.162.38.120/harsh02.exe for known malicious executable), IP address (e.g. 185.130.5.231 for known attacker) or HTTP User-Agent header value (e.g. sqlmap for automatic SQL injection and database takeover tool). Also, it uses (optional) advanced heuristic mechanisms that can help in discovery of unknown threats (e.g. new malware).

TheFatRat (Unit for bypass av)   WRITTEN in C

https://github.com/Screetsec/TheFatRat

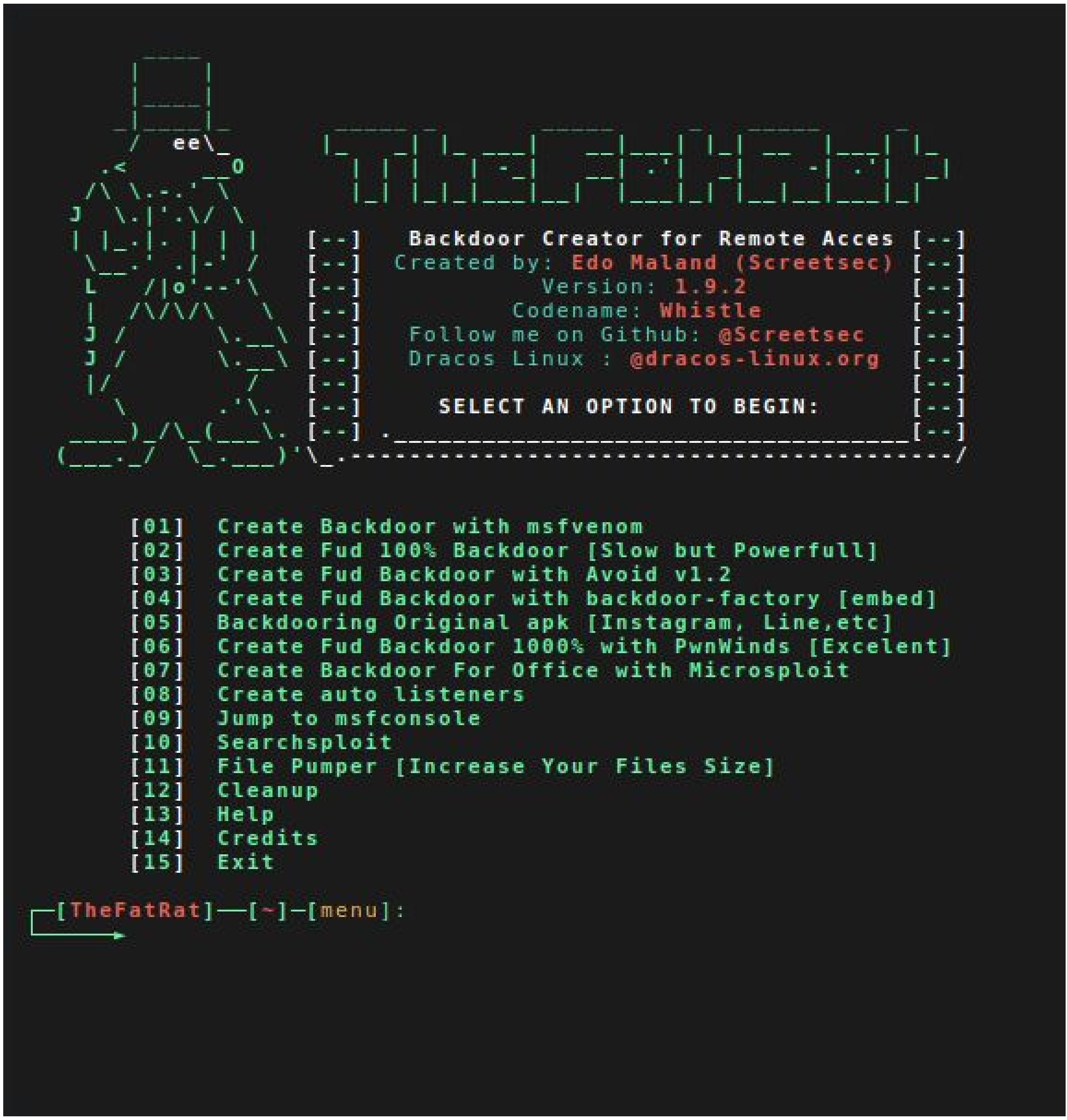
https://github.com/Screetsec/TheFatRat/blob/master/README.md

Update: Version 1.9.2

Codename: Whistle

Thefatrat a massive exploiting tool revealed

An easy tool to generate backdoor and easy tool to post exploitation attack like browser attack,dll . This tool compiles a malware with popular payload and then the compiled malware can be execute on windows, android, mac . The malware that created with this tool also have an ability to bypass most AV software protection .



```
[--]   Backdoor Creator for Remote Acces [--]
[--]  Created by: Edo Maland (Screetsec) [--]
[--]             Version: 1.9.2          [--]
[--]           Codename: Whistle         [--]
[--]   Follow me on Github: @Screetsec   [--]
[--]   Dracos Linux : @dracos-linux.org  [--]
[--]                                     [--]
[--]     SELECT AN OPTION TO BEGIN:      [--]
[--] .___________________________________[--]


      [01]  Create Backdoor with msfvenom
      [02]  Create Fud 100% Backdoor [Slow but Powerfull]
      [03]  Create Fud Backdoor with Avoid v1.2
      [04]  Create Fud Backdoor with backdoor-factory [embed]
      [05]  Backdooring Original apk [Instagram, Line,etc]
      [06]  Create Fud Backdoor 1000% with PwnWinds [Excelent]
      [07]  Create Backdoor For Office with Microsploit
      [08]  Create auto listeners
      [09]  Jump to msfconsole
      [10]  Searchsploit
      [11]  File Pumper [Increase Your Files Size]
      [12]  Cleanup
      [13]  Help
      [14]  Credits
      [15]  Exit

┌─[TheFatRat]──[~]─[menu]:
└────→
```

Pafish

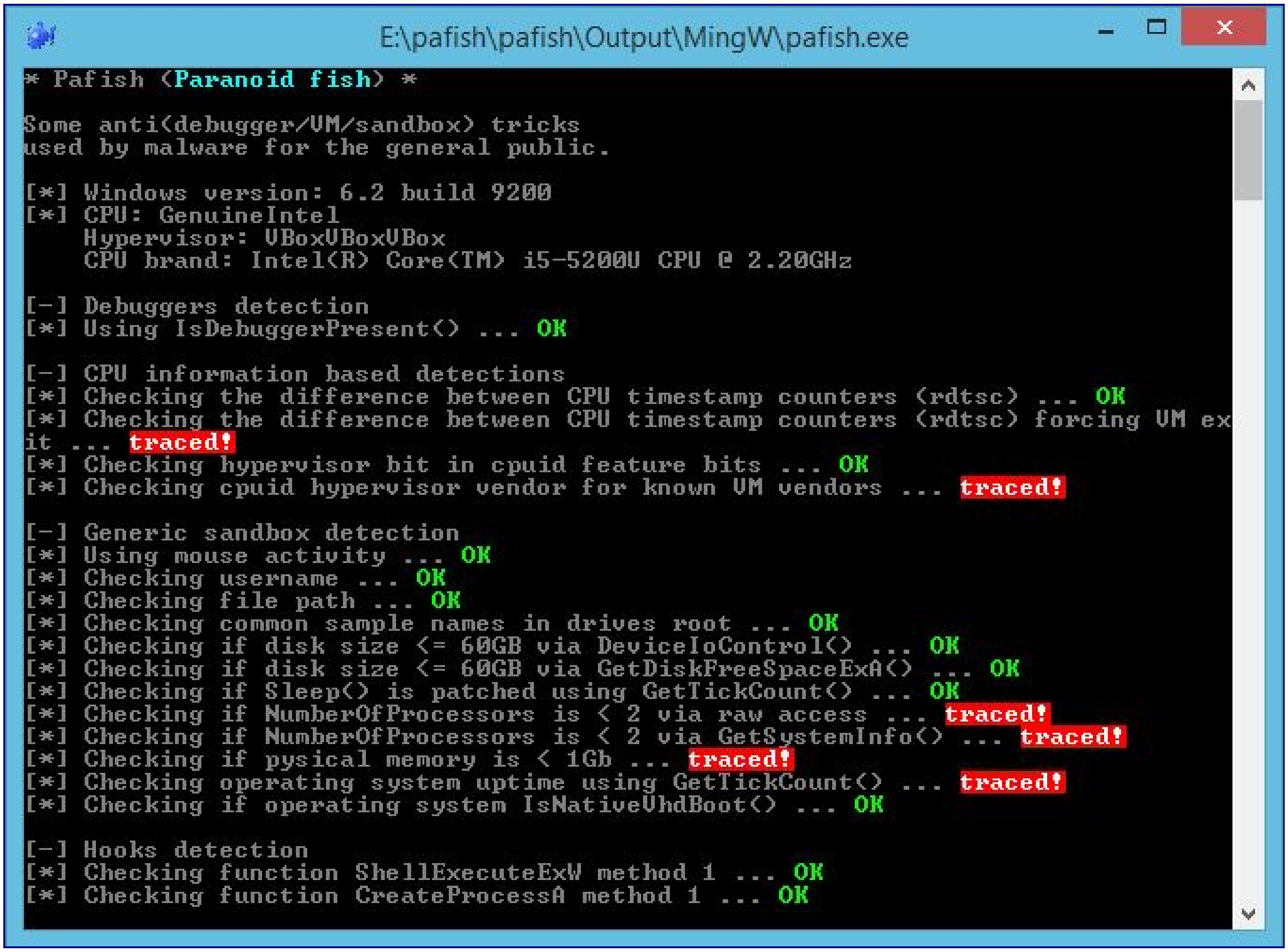
https://github.com/a0rtega/pafish   **WRITTEN in C**

(Paranoid Fish)

Pafish is a demonstration tool that employs several techniques to detect sandboxes and analysis environments in the same way as malware families do.

The project is open source, you can read the code of all anti-analysis checks. You can also **download** the executable of the latest stable version.

It is licensed under GNU/GPL version 3.



```
* Pafish (Paranoid fish) *

Some anti(debugger/VM/sandbox) tricks
used by malware for the general public.

[*] Windows version: 6.2 build 9200
[*] CPU: GenuineIntel
    Hypervisor: VBoxVBoxVBox
    CPU brand: Intel(R) Core(TM) i5-5200U CPU @ 2.20GHz

[-] Debuggers detection
[*] Using IsDebuggerPresent() ... OK

[-] CPU information based detections
[*] Checking the difference between CPU timestamp counters (rdtsc) ... OK
[*] Checking the difference between CPU timestamp counters (rdtsc) forcing VM ex
it ... traced!
[*] Checking hypervisor bit in cpuid feature bits ... OK
[*] Checking cpuid hypervisor vendor for known VM vendors ... traced!

[-] Generic sandbox detection
[*] Using mouse activity ... OK
[*] Checking username ... OK
[*] Checking file path ... OK
[*] Checking common sample names in drives root ... OK
[*] Checking if disk size <= 60GB via DeviceIoControl() ... OK
[*] Checking if disk size <= 60GB via GetDiskFreeSpaceExA() ... OK
[*] Checking if Sleep() is patched using GetTickCount() ... OK
[*] Checking if NumberOfProcessors is < 2 via raw access ... traced!
[*] Checking if NumberOfProcessors is < 2 via GetSystemInfo() ... traced!
[*] Checking if pysical memory is < 1Gb ... traced!
[*] Checking operating system uptime using GetTickCount() ... traced!
[*] Checking if operating system IsNativeVhdBoot() ... OK

[-] Hooks detection
[*] Checking function ShellExecuteExW method 1 ... OK
[*] Checking function CreateProcessA method 1 ... OK
```

Scope

The objective of this project is to collect usual tricks seen in malware samples. This allows us to study them, and test if our analysis environments are properly implemented.

theZoo aka Malware DB (http://thezoo.morirt.com/)

A repository of LIVE malwares for your own joy and pleasure   WRITTEN IN PYTHON

View the Project on GitHub ytisf/theZoo / https://github.com/ytisf/theZoo

Download **ZIP File** / https://github.com/ytisf/theZoo/zipball/master
Download **TAR Ball** / https://github.com/ytisf/theZoo/tarball/master
View On **GitHub** / https://github.com/ytisf/theZoo

About

theZoo is a project created to make the possibility of malware analysis open and available to the public. Since we have found out that almost all versions of malware are very hard to come by in a way which will allow analysis we have decided to gather all of them for you in an available and safe way. theZoo was born by Yuval tisf Nativ and is now maintained by Shahak Shalev.

theZoo is open and welcoming visitors!