MyDoom

Malware Reverse Engineering

Report Issued: 07.30.2022

Prepared by: Jeff Malavasi
Rochester Institute of Technology, Computing Security

TABLE OF CONTENTS

OVERVIEW **3**

ANALYSIS **4**

Static Analysis **4**

General 4
Basic 4
Strings 4
Findings 4
Floss 6
Findings 6
UPX 7
Findings 8
PE View 8
Findings 8
PEiD 9
Findings 9
Resource Hacker 9
Findings 9
Virus Total 10
Findings 10
Advanced 13
Ida Pro 13
Finding 1: DNS Query Algorithm 13
Finding 2: Mass Mailer Algorithm 15

Dynamic Analysis **16**

Basic 16
RegShot 16
Findings 16
Process Monitor 17
Findings 17
ApateDNS 20
Findings 20
netstat 22
Findings 22
Advanced 23
x32Debug 23
Finding 1: DNS Query Algorithm 23
Finding 2: Mass Mailer Algorithm 24

CONCLUSION **25**

Potential Danger **25**

Malware Removal **25**

APPENDIX A – TOOLS **26**

APPENDIX B – REFERENCES **27**

OVERVIEW

The MyDoom worm was first detected in January 2004. Although its origins are still unknown, artifacts within the binary most likely link it’s development back to Russia. The virus was spread through a self-replicating email campaign that tricked users into opening a malicious attachment. The attachment, which appeared benign, was actually the malicious payload. Upon opening the file, it would silently scan the computer and the internet for any email addresses it could find to continue replicating. Additionally, the virus creates a backdoor on the machine which was used for multiple botnet attacks against large organizations such as Google and Microsoft. In fact, the virus successfully took down Google for nearly a day through a distributed denial of service attack.

When MyDoom was originally released, analysts reported that the virus accounted for one in ten emails sent worldwide. MyDoom crippled internet page load times by 50% globally and caused an estimated of $65 billion dollars. Although today it only accounts for about 1% of all email and is detected by most antivirus software, successful campaigns have been reported as recently as a few years ago.

ANALYSIS

Static Analysis

General

File Information	
Operating System	Windows 10 1709
Description	A Windows Worm that was first sighted in 2004. It is infamously known as the fastest spreading email worm to date and is still heavily used today.
Size (bytes)	26050
Hash (SHA256)	6F064D4987B4202EBE2FAAAB28F3582DD784F24FA1A13F305051A6D7E85A78ED
Source	https://samples.vx-underground.org/samples/Families/MyDoom/

Basic

Strings

Strings is a static command line utility that is used to quickly pull readable text snippets out of a binary. This can be used to get a cursory idea of the function of the binary.

Findings

In this binary, there are a small number of strings (548). This indicates that the file may be obfuscated through packing. Further analysis shows the strings “UPX0” and “UPX1” (Figure 1), which is a specific type of packing technology. Finally, we see the presence of the LoadLibraryA and GetProcAddress API calls (Figure 2), but not many others. This is another strong indicator of packing.

Analysis

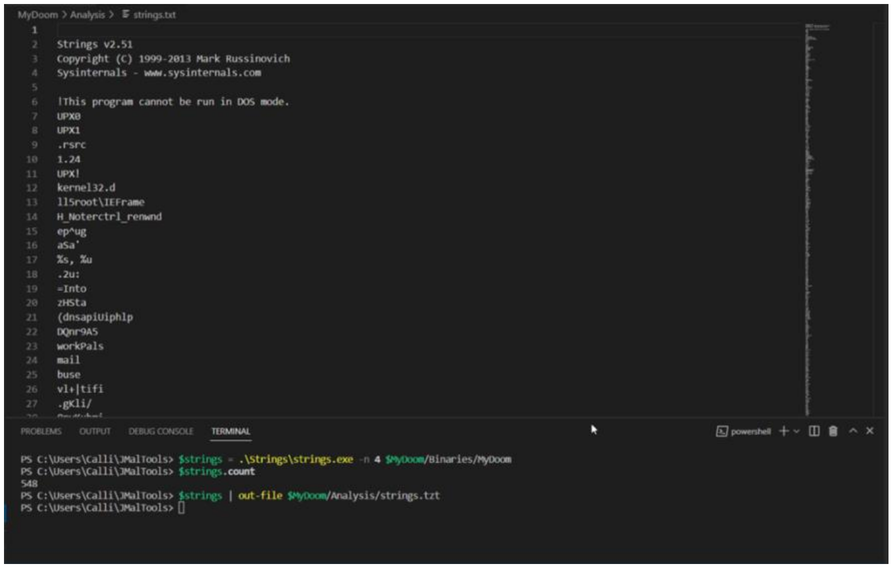


Figure 1 Strings UPX

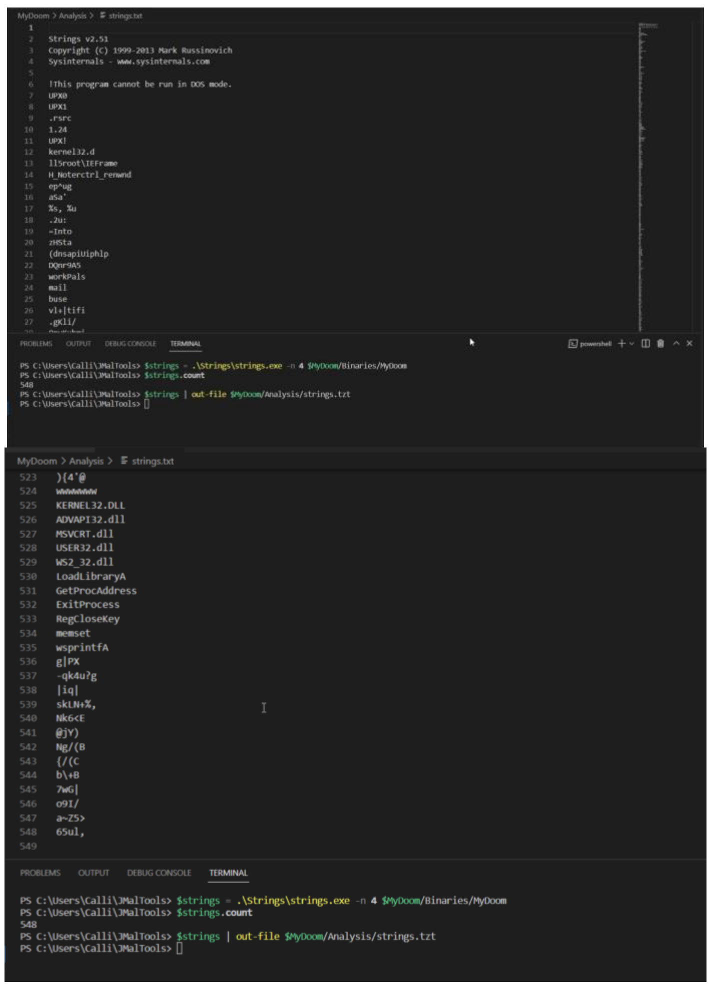


Figure 2 API Calls

Floss

Floss is another CLI tool that was created by the FireEye FLARE team. It builds upon the capabilities of strings by being able to decrypt encoded strings automatically.

Findings

In this binary, FLOSS was able to decode a few different interesting strings. The first appears to be some sort of email template used to help replicate the virus across other hosts (Figure 3). The template appears to be able to send a few different message bodies to make it more difficult to detect. Since there are many variants of MyDoom, the email template can be used to help hone in on the specific version of the malware. This template is a strong indicator of MyDoom.m, which aimed to get people to open an email that mimicked an NDR from their system administrator.

The second string of interest appears to be used to help send the message (Figure 4). It is a series of message headers that will likely be passed to the built in SMTP server.

Finally, we see a series of strings that helps to understand how the worm gathers email recipients (Figure 5). On top of searching the local host for temporary internet files and contacts, the malware also queries search engines for publicly available addresses. This is likely how it became the fastest spreading email virus of all time.

Analysis

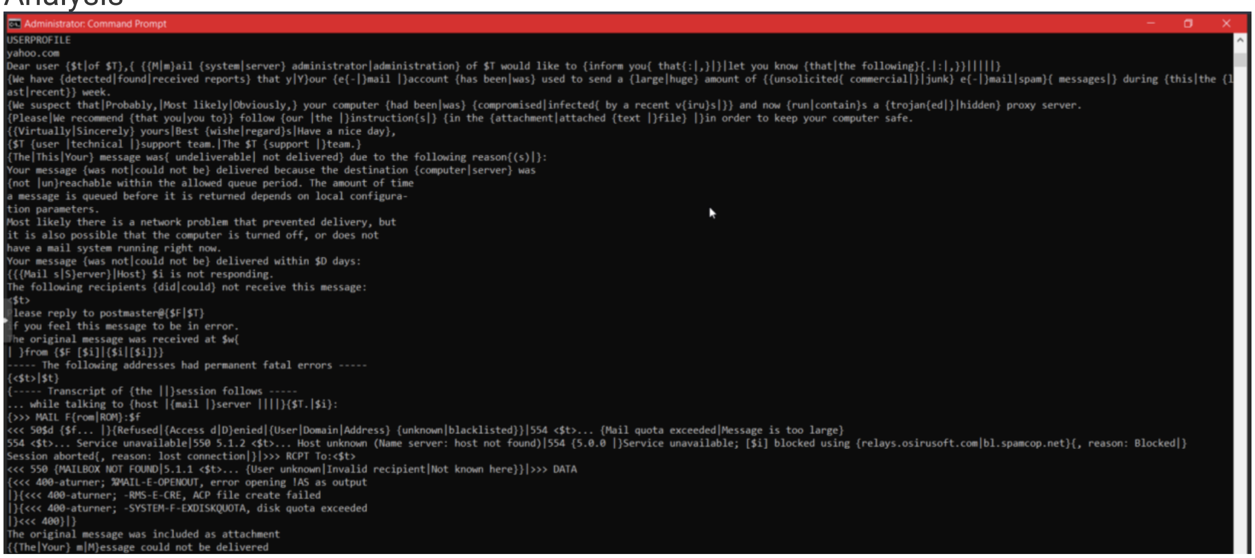


Figure 3 Email Template

```
ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/
X-Priority: 3
X-MSMail-Priority: Normal
X-Mailer: Microsoft Outlook Express 6.00.2600.0000
X-MIMEOLE: Produced By Microsoft MimeOLE V6.00.2600.0000
Content-Type: multipart/mixed;
        boundary="%s"
MIME-Version: 1.0
Date:
Subject: %s
To: %s
From: %s
----=_%s_%.3u_%.4u_%.8X.%.8X
NextPart
--%s--
--%s
Content-Type: application/octet-stream;
        name="%s"
Content-Transfer-Encoding: base64
Content-Disposition: %s;
        filename="%s"
inline
--%s
Content-Type: text/plain;
        charset=us-ascii
Content-Transfer-Encoding: 7bit
This is a multi-part message in MIME format.
```

Figure 4 Email Headers

```
services
urlmon.dll
URLDownloadToCacheFileA
http://search.lycos.com/default.asp?lpv=1&loc=searchhp&tab=web&query=%s
&nbq=%d
http://www.altavista.com/web/results?q=%s&kgs=0&kls=0
&n=%d
http://search.yahoo.com/search?p=%s&ei=UTF-8&fr=fp-tab-web-t&cop=mss&tab=
&num=%d
http://www.google.com/search?hl=en&ie=UTF-8&oe=UTF-8&q=%s
%s+%s
```

Figure 5 Email Recipient Worming

UPX

UPX is a command line utility that can be used to pack and unpack files. Malware developers often pack files to obfuscate the contents from a malware reverse engineer.

Findings

In both the strings and PEiD analysis, we see that this file is packed with UPX. I was successfully able to unpack the file, which appeared to be compressed by about 70% (Figure 6).

Analysis

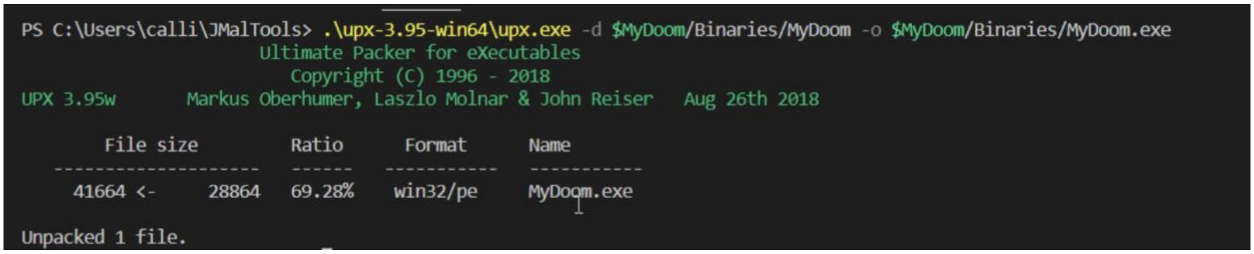


Figure 6 Unpacked Binary

PE View

PE View is a graphical tool that can be used to quickly find embedded files, learn about when the binary was created, as well as quickly see the imported and exported API calls.

Findings

In this binary we can see references to a packed UPX file (Figure 7).

Analysis

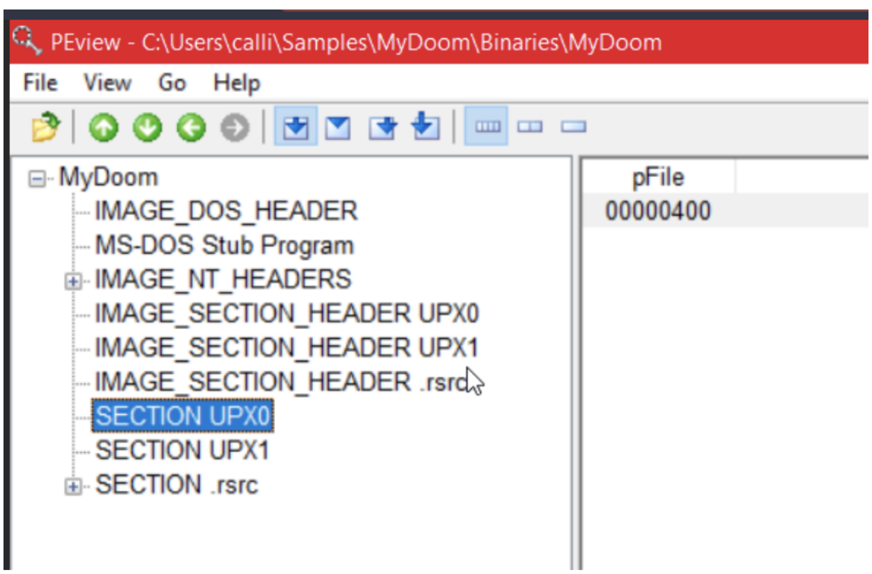


Figure 7 PE View Analysis

PEiD

PEiD is another graphical tool that can be used to detect which packing technology was used to obfuscate a particular binary.

Findings

PEiD was able to detect that the binary was packed with UPX v0.89.6 (Figure 8).

Analysis

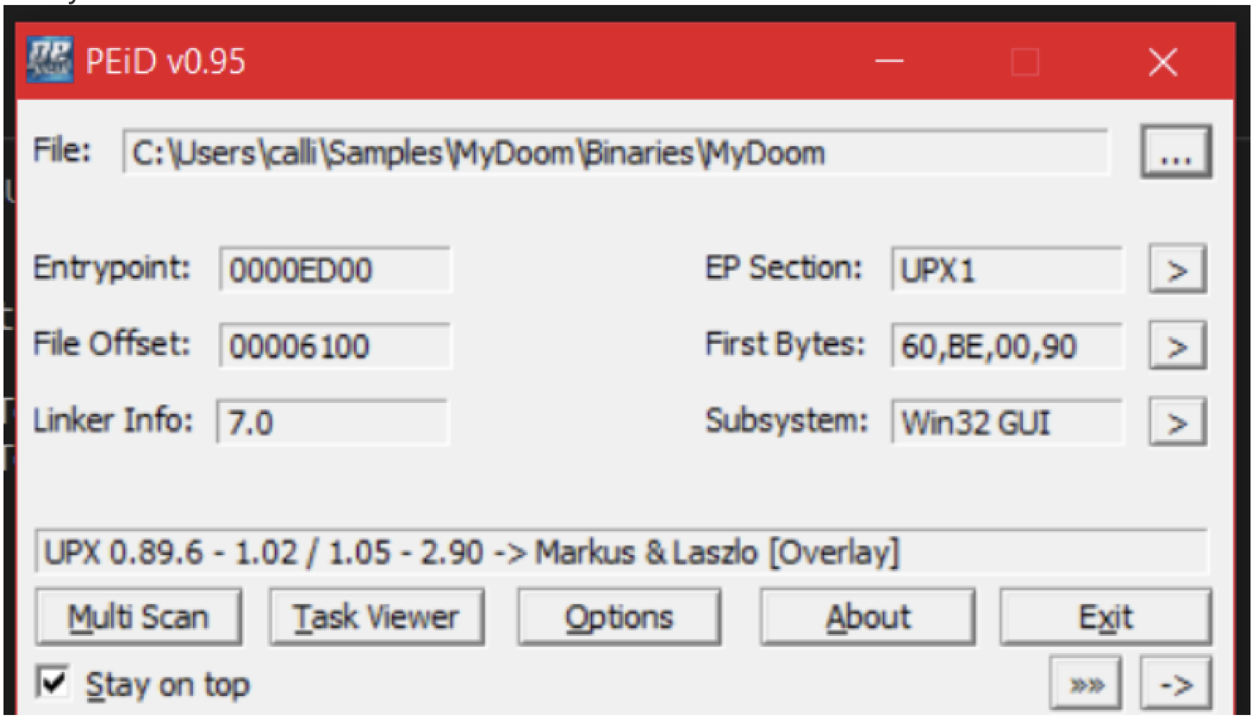


Figure 8 Evidence of Packing

Resource Hacker

Resource Hacker is used to find hidden embedded binaries and other file artifacts. It can also be used to export different subfiles for isolated analysis.

Findings

Although there weren't any hidden files in this binary, the embedded icon changed with each variant of MyDoom. This icon further validates our prediction that this sample is variant M (Figure 9).

Analysis

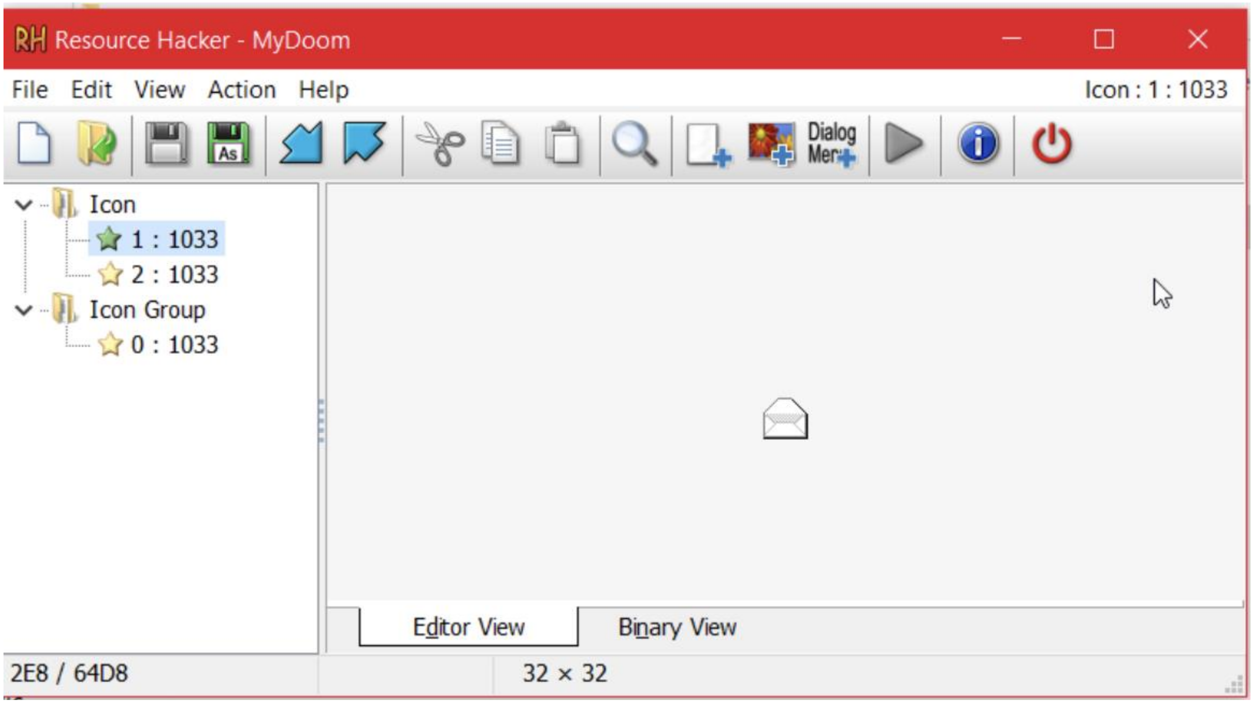


Figure 9 MyDoom.m Icon

Virus Total

Virus Total is an online tool that maintains a database of malicious files. It can be used to determine if a binary has already been deemed malicious as well as provide a basic report on what it may do.

Findings

From the Virus Total report, we can see that this binary has been deemed malicious by 66 different anti-virus vendors. Additionally, Virus total can detect when the file was first seen in the wild (6/4/2022) as well as the packing technology (Figure 10). From the behaviors tab, we can see what the file is expected to do once run. From our analysis, we can see that the file makes many DNS requests, as well as creates a startup task that automatically starts the binary on boot (Figure 11). Finally, we get a better look at the various search queries the binary makes to find more victims.

Analysis

66 / 71

Community Score

66 security vendors and 1 sandbox flagged this file as malicious

6f064d4987b4202ebe2faaab28f3582dd784f24fa1a13f305051a6d7e85a78ed
MyDoom

28.19 KB Size

2022-07-25 02:16:57 UTC a moment ago

EXE

checks-network-adapters · direct-cpu-clock-access · long-sleeps · nxdomain · overlay · peexe · persistence · runtime-modules · suspicious-dns · upx

DETECTION · DETAILS · RELATIONS · BEHAVIOR · COMMUNITY

Security Vendors' Analysis

Ad-Aware	Worm.Generic.24461	AhnLab-V3	Win32/Mydoom.worm.49344.B
Alibaba	Malware:Win32/Dorpal.ali1000029	ALYac	Worm.Mydoom
Antiy-AVL	Trojan/Generic.ASMalwS.118	Arcabit	Worm.Generic.D5F8D
Avast	Win32:Banker-FNW [Trj]	AVG	Win32:Banker-FNW [Trj]
Avira (no cloud)	WORM/Mydoom.O.1	Baidu	Win32.Worm-Email.Mydoom.a
BitDefender	Worm.Generic.24461	BitDefenderTheta	AI:Packer.6236D6581F
Bkav Pro	W32.MyDoom.M.Worm	ClamAV	Win.Worm.Mydoom-90
Comodo	Worm.Win32.Mydoom.R@348l	CrowdStrike Falcon	Win/malicious_confidence_100% (W)
Cybereason	Malicious.28f59b	Cylance	Unsafe

Basic Properties

MD5	13c05f728f59b645759ccff2469dd2b2
SHA-1	a2879876885d68be54bc0d9307a8ea0b4182560b
SHA-256	6f064d4987b4202ebe2faaab28f3582dd784f24fa1a13f305051a6d7e85a78ed
Vhash	02403e0f7d1019z301lz15z17z
Authentihash	8275243725139b9e8403115b265b8c532e6590f2c5a37f272350c7b64de96452
Imphash	98cd465c2ab2841f9fd90d5e847563f4
SSDEEP	384:1vxBbK26lj5ld8SpHx9jLhsznnVxA1WmP5w7GGCJlqqwMyNTbs4:Dv8lRRdsxq1DjJcqfo7
TLSH	T150D2C085B050FAA2C01682331D86C471FD119C611AAAD2CBBA24BF7FFDB17850B0CD2B
File type	Win32 EXE
Magic	PE32 executable for MS Windows (GUI) Intel 80386 32-bit
TrID	UPX compressed Win32 Executable (34.7%)
TrID	Win32 EXE Yoda's Crypter (34.1%)
TrID	Win32 Dynamic Link Library (generic) (8.4%)
TrID	Win16 NE executable (generic) (6.4%)
TrID	Win32 Executable (generic) (5.7%)
File size	28.19 KB (28864 bytes)
PEiD packer	UPX 2.90 [LZMA] -> Markus Oberhumer, Laszlo Molnar & John Reiser
Cyren packer	UPX

History

First Seen In The Wild	2022-06-04 15:39:32 UTC
First Submission	2021-07-26 10:48:24 UTC
Last Submission	2022-07-25 02:20:07 UTC
Last Analysis	2022-07-25 02:16:57 UTC

Names

MyDoom
mydoom.exe
instruction.html .com

Figure 10 VT Basic

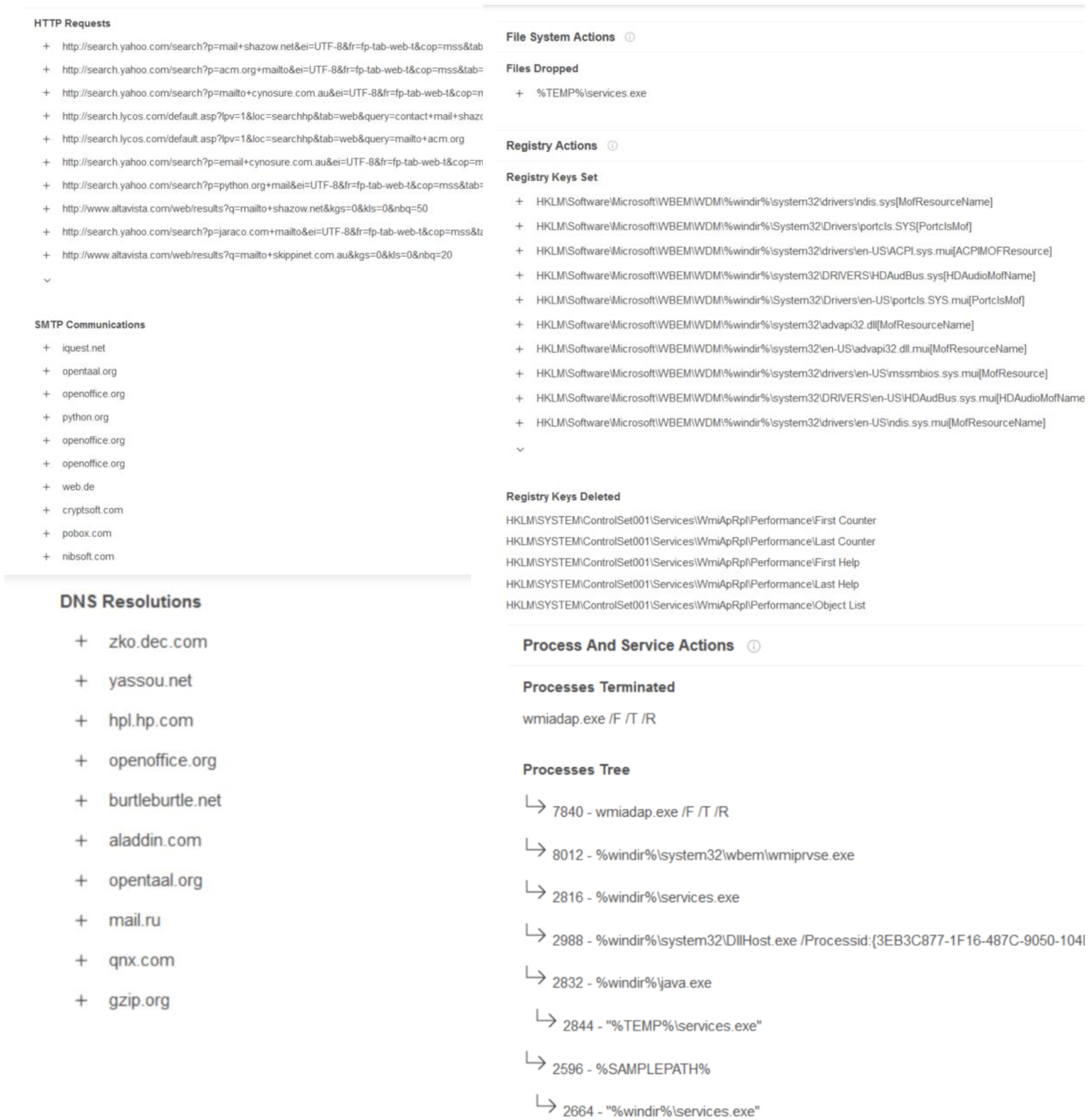


Figure 11 VT Behaviors

Advanced

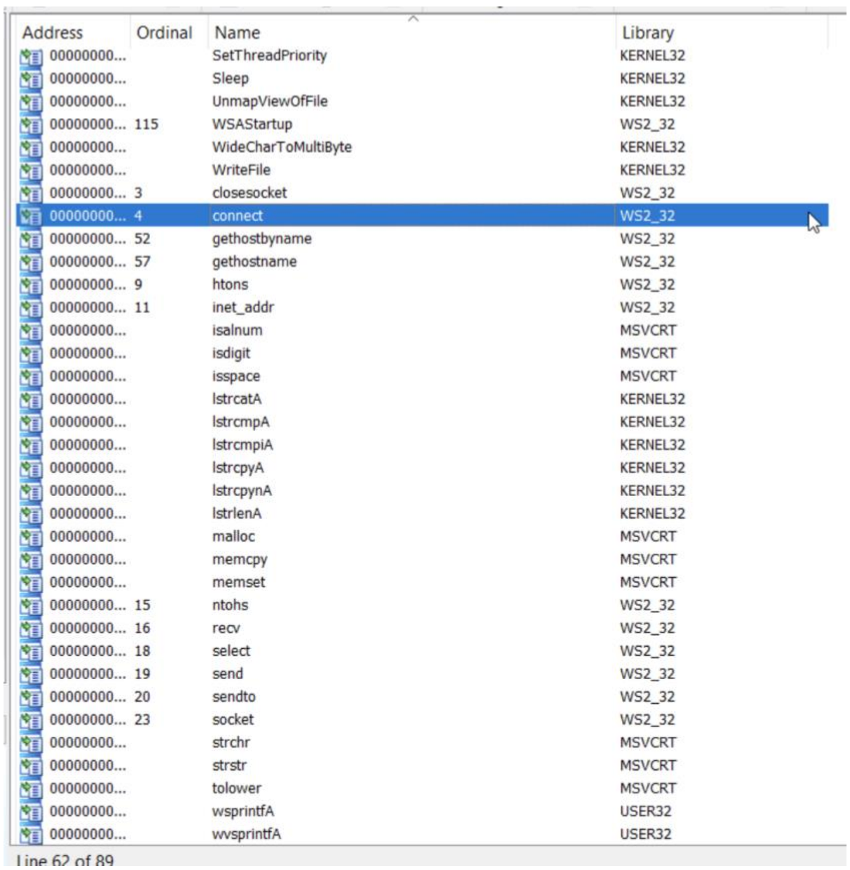
Ida Pro

Ida Pro is a commonly used disassembler, that can be used get a detailed picture of all the functions within a binary.

Finding 1: DNS Query Algorithm

The first algorithm that was investigated was the DNS query algorithm. This function is used to find potential victims on the internet to aid in replication of the worm. I first started by looking at any imports related to networking (Figure 12). Next, I examined any cross reference to GetHostByName until I found the function of interest (Figure 13). The address of this function was later used for dynamic analysis.

Analysis



Address	Ordinal	Name	Library
00000000...		SetThreadPriority	KERNEL32
00000000...		Sleep	KERNEL32
00000000...		UnmapViewOfFile	KERNEL32
00000000...	115	WSAStartup	WS2_32
00000000...		WideCharToMultiByte	KERNEL32
00000000...		WriteFile	KERNEL32
00000000...	3	closesocket	WS2_32
00000000...	4	connect	WS2_32
00000000...	52	gethostbyname	WS2_32
00000000...	57	gethostname	WS2_32
00000000...	9	htons	WS2_32
00000000...	11	inet_addr	WS2_32
00000000...		isalnum	MSVCRT
00000000...		isdigit	MSVCRT
00000000...		isspace	MSVCRT
00000000...		lstrcatA	KERNEL32
00000000...		lstrcmpA	KERNEL32
00000000...		lstrcmpiA	KERNEL32
00000000...		lstrcpyA	KERNEL32
00000000...		lstrcpynA	KERNEL32
00000000...		lstrlenA	KERNEL32
00000000...		malloc	MSVCRT
00000000...		memcpy	MSVCRT
00000000...		memset	MSVCRT
00000000...	15	ntohs	WS2_32
00000000...	16	recv	WS2_32
00000000...	18	select	WS2_32
00000000...	19	send	WS2_32
00000000...	20	sendto	WS2_32
00000000...	23	socket	WS2_32
00000000...		strchr	MSVCRT
00000000...		strstr	MSVCRT
00000000...		tolower	MSVCRT
00000000...		wsprintfA	USER32
00000000...		wvsprintfA	USER32

Line 62 of 89

Figure 12 GetHostByName

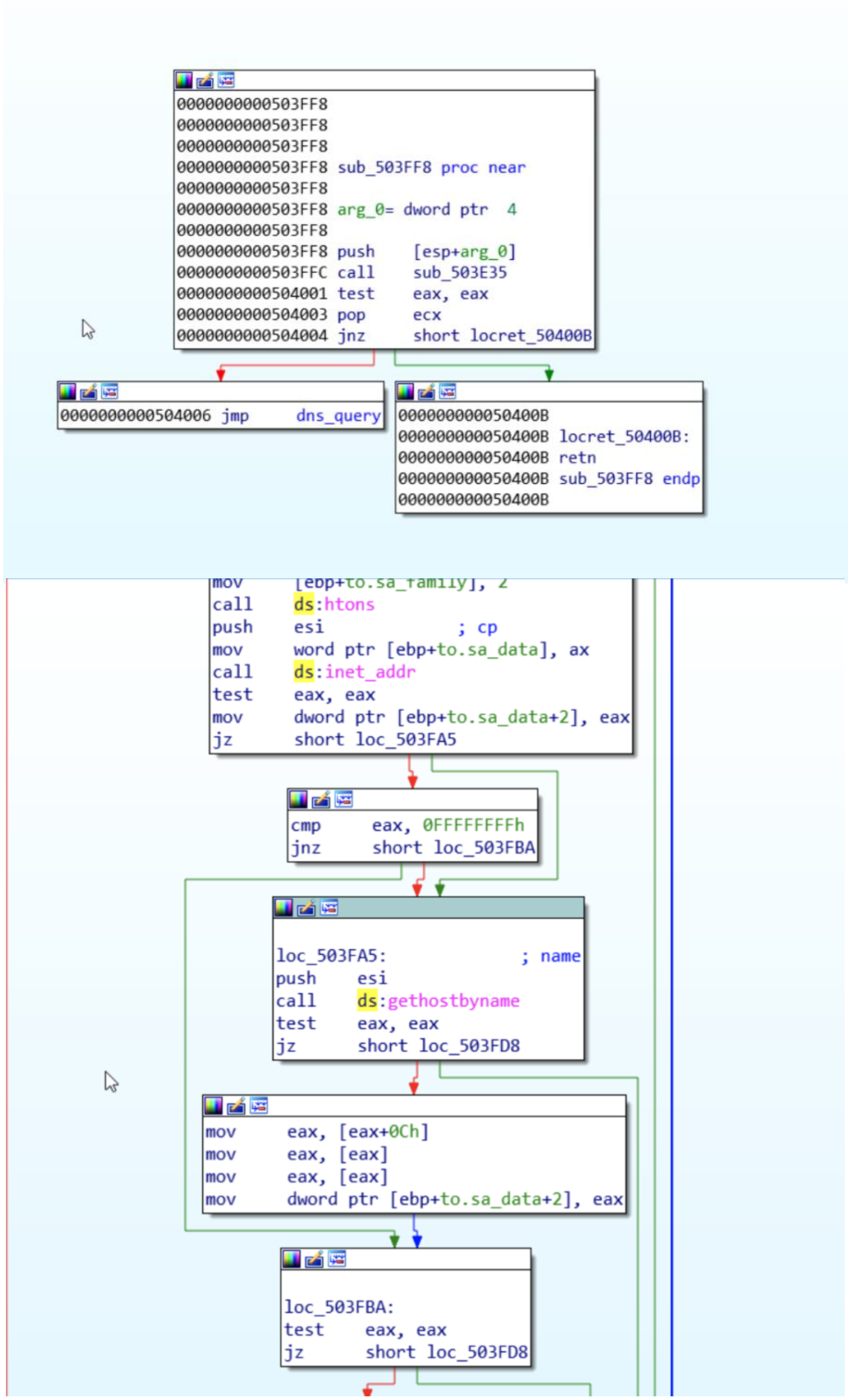


Figure 13 DNS Algorithm

Finding 2: Mass Mailer Algorithm

The second finding is the function that is used to construct the phony email message. To find this function, I looked for the cross references to PostMessageA, which is used to queue messages (Figure 13). Next, I looked for any calls to this function, which would be used to initiate the email phishing campaign (Figure 14).

Analysis

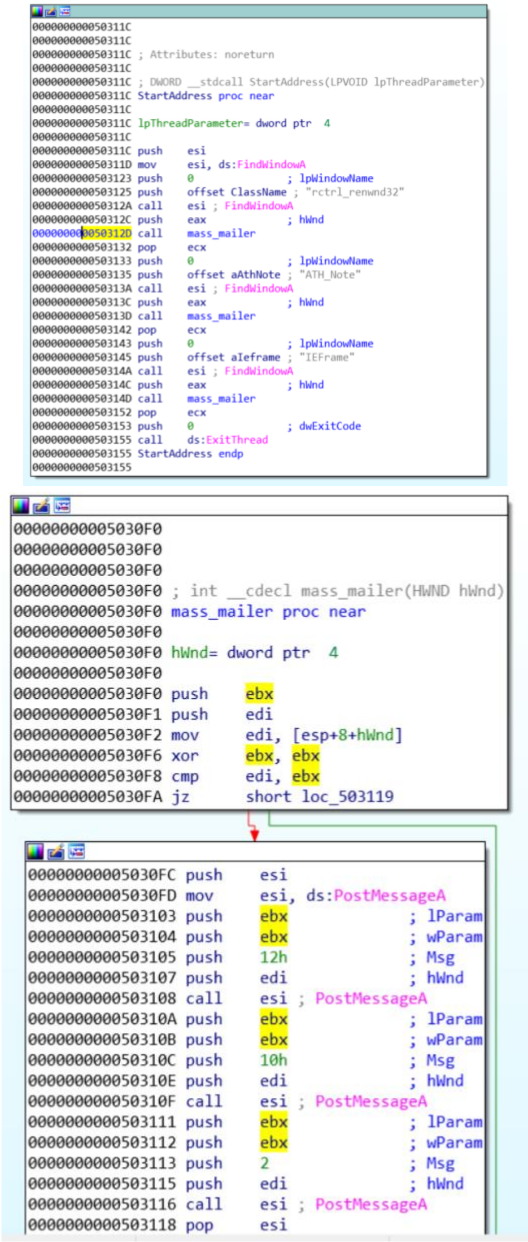


Figure 14 Mass Mailer Algorithm

Dynamic Analysis

Basic

RegShot

RegShot is a graphical dynamic tool that will create a report of all the modified files and registry keys during a specific time window.

Findings

The binary was run for a period of five minutes on a virtual air gapped network. In this period a total of 14 registry keys and 6 files were added. Additionally, 17 files and keys were modified (Figure 15). Some of the files are in the system directory, which is often a technique used to hide malicious files. Some variants of MyDoom will replace the taskmon.exe with a malicious version. Additionally, the registry keys indicate the creation of a startup task which runs the binary each time the host boots.

Analysis

```
~res-x64.txt - Notepad
File Edit Format View Help
Computer: DESKTOP-OKR609Q , DESKTOP-OKR609Q
Username: calli , calli

----------------------------------
Keys added: 8
----------------------------------
HKU\S-1-5-21-2023141954-689327518-1290971125-1001\Software\Microsoft\Windows\CurrentVersion\Search\RecentApps\{CBF8C622-7C34-4E22-BE63-57FA163E7703}
HKU\S-1-5-21-2023141954-689327518-1290971125-1001\Software\Microsoft\Daemon
HKU\S-1-5-21-2023141954-689327518-1290971125-1001\Software\Classes\VirtualStore\Machine\Software\WOW6432Node
HKU\S-1-5-21-2023141954-689327518-1290971125-1001\Software\Classes\VirtualStore\Machine\Software\WOW6432Node\Microsoft
HKU\S-1-5-21-2023141954-689327518-1290971125-1001\Software\Classes\VirtualStore\Machine\Software\WOW6432Node\Microsoft\Daemon
HKU\S-1-5-21-2023141954-689327518-1290971125-1001_Classes\VirtualStore\Machine\Software\WOW6432Node
HKU\S-1-5-21-2023141954-689327518-1290971125-1001_Classes\VirtualStore\Machine\Software\WOW6432Node\Microsoft
HKU\S-1-5-21-2023141954-689327518-1290971125-1001_Classes\VirtualStore\Machine\Software\WOW6432Node\Microsoft\Daemon

----------------------------------
Values added: 6
----------------------------------
HKU\S-1-5-21-2023141954-689327518-1290971125-1001\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\{CEBFF5CD-ACE2-4F4F-9178-9926F41749EA}\Count\P:\Hfref\pnyyv\Fnzcyrf\ZlQbbz\Ovanevrf\ZlQbbz.rkr:  00 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
HKU\S-1-5-21-2023141954-689327518-1290971125-1001\Software\Microsoft\Windows\CurrentVersion\Search\RecentApps\{CBF8C622-7C34-4E22-BE63-57FA163E7703}\LastAccessedTime:  A0 69 F3 A6 16 A2 D8 01
HKU\S-1-5-21-2023141954-689327518-1290971125-1001\Software\Microsoft\Windows\CurrentVersion\Search\RecentApps\{CBF8C622-7C34-4E22-BE63-57FA163E7703}\AppId: "C:\Users\calli\Samples\MyDoom\Binaries\MyDoom.exe"
HKU\S-1-5-21-2023141954-689327518-1290971125-1001\Software\Microsoft\Windows\CurrentVersion\Search\RecentApps\{CBF8C622-7C34-4E22-BE63-57FA163E7703}\LaunchCount: 0x00000001
HKU\S-1-5-21-2023141954-689327518-1290971125-1001\Software\Microsoft\Windows\CurrentVersion\Search\RecentApps\{CBF8C622-7C34-4E22-BE63-57FA163E7703}\AppPath: ""
HKU\S-1-5-21-2023141954-689327518-1290971125-1001\Software\Microsoft\Windows NT\CurrentVersion\AppCompatFlags\Compatibility Assistant\Store\C:\Users\calli\Samples\MyDoom\Binaries\MyDoom.exe:  53 41 43 50 01 00 00 00 00 00 00 00 07 00 00 00 28 00 00 00 C0 A2 00 00

----------------------------------
Values modified: 5
----------------------------------
HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Notifications\Data\418A073AA3BC3475:  74 16 00 00 00 00 00 00 04 00 04 00 01 02 03 00 00 00 00 00 12 00 00 00 0D 78 79 00 02 00 00 00 87 DE 83 00 09 00 00 00 A1 9F 5E 00 01 00 01 00 00 00 17 00 00 00 56 73 7D 00 01
HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Notifications\Data\418A073AA3BC3475:  76 16 00 00 00 00 00 00 04 00 04 00 01 02 03 00 00 00 00 00 12 00 00 00 0D 78 79 00 02 00 00 00 87 DE 83 00 09 00 00 00 A1 9F 5E 00 01 00 01 00 00 00 19 00 00 00 56 73 7D 00 01
HKU\S-1-5-19\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Wpad\74-ac-b9-49-60-29\WpadDecisionTime:  A3 FC 24 DB 14 A2 D8 01
HKU\S-1-5-19\Software\Microsoft\Windows\CurrentVersion\Internet Settings\Wpad\74-ac-b9-49-60-29\WpadDecisionTime:  53 D6 B9 C7 16 A2 D8 01
HKU\S-1-5-21-2023141954-689327518-1290971125-1001\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\{CEBFF5CD-ACE2-4F4F-9178-9926F41749EA}\Count\HRZR_PGYFRFFVBA:  00 00 00 00 51 01 00 00 D9 08 00 00 C9 9D 9A 06 41 00 00 00 0C 01 00 00 F9 B9 6C 00 4D 00
HKU\S-1-5-21-2023141954-689327518-1290971125-1001\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\{CEBFF5CD-ACE2-4F4F-9178-9926F41749EA}\Count\HRZR_PGYFRFFVBA:  00 00 00 00 52 01 00 00 DC 08 00 00 A4 35 98 06 41 00 00 00 0D 01 00 00 C2 FE 6C 00 4D 00
HKU\S-1-5-21-2023141954-689327518-1290971125-1001\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\{CEBFF5CD-ACE2-4F4F-9178-9926F41749EA}\Count\Zvpebfbsg.Jvaqbjf.Rkcybere:  00 00 00 00 41 00 00 00 0C 01 00 00 F9 B9 6C 00 00 00 80 BF 00 00 80 BF 00 00
HKU\S-1-5-21-2023141954-689327518-1290971125-1001\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\{CEBFF5CD-ACE2-4F4F-9178-9926F41749EA}\Count\Zvpebfbsg.Jvaqbjf.Rkcybere:  00 00 00 00 41 00 00 00 0D 01 00 00 C2 FE 6C 00 00 00 80 BF 00 00 80 BF 00 00
HKU\S-1-5-21-2023141954-689327518-1290971125-1001\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\{CEBFF5CD-ACE2-4F4F-9178-9926F41749EA}\Count\P:\Hfref\pnyyv\WZnyGbbyf\Ertfubg-1.9.0\Ertfubg-k64-NAFV.rkr:  00 00 00 00 01 00 00 00 03 00 00 00 2F 56 00
HKU\S-1-5-21-2023141954-689327518-1290971125-1001\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\{CEBFF5CD-ACE2-4F4F-9178-9926F41749EA}\Count\P:\Hfref\pnyyv\WZnyGbbyf\Ertfubg-1.9.0\Ertfubg-k64-NAFV.rkr:  00 00 00 00 01 00 00 00 05 00 00 00 41 A9 00

----------------------------------
Files added: 6
----------------------------------
C:\Users\calli\AppData\Local\Temp\Iigiry.log
C:\Users\calli\AppData\Local\Temp\java.exe
C:\Users\calli\AppData\Local\Temp\services.exe
C:\Users\calli\AppData\Local\Temp\zincite.log
C:\Windows\Prefetch\MYDOOM.EXE-CEA0452A.pf
C:\Windows\Prefetch\SERVICES.EXE-2F1F0CB6.pf

----------------------------------
Files [attributes?] modified: 12
----------------------------------
C:\Users\calli\AppData\Local\Microsoft\Windows\UsrClass.dat.LOG1
C:\Users\calli\ntuser.dat.LOG1
C:\Windows\Prefetch\SMARTSCREEN.EXE-9B5E4173.pf
C:\Windows\Prefetch\SVCHOST.EXE-5AAAB524.pf
C:\Windows\ServiceProfiles\LocalService\NTUSER.DAT.LOG1
C:\Windows\ServiceProfiles\NetworkService\NTUSER.DAT.LOG1
C:\Windows\System32\config\DEFAULT.LOG2
C:\Windows\System32\config\SOFTWARE.LOG1
C:\Windows\System32\config\SOFTWARE.LOG2
C:\Windows\System32\config\SYSTEM.LOG1
C:\Windows\System32\winevt\Logs\Microsoft-Windows-Application-Experience%4Program-Compatibility-Assistant.evtx
C:\Windows\System32\winevt\Logs\Microsoft-Windows-UAC-FileVirtualization%4Operational.evtx

----------------------------------
Total changes: 37
----------------------------------
```

Figure 15 File Modifications

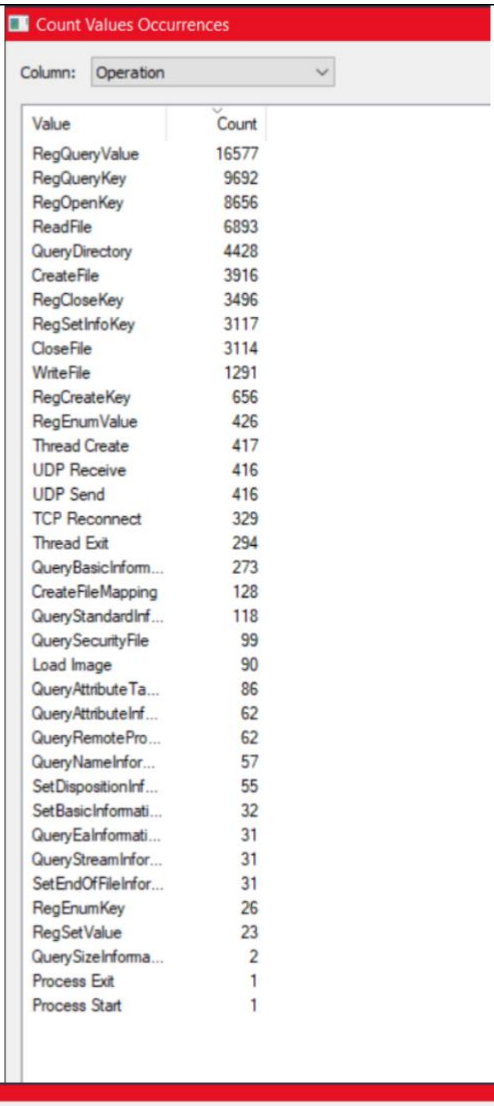
Process Monitor

Process Monitor is used to closely examine a particular running process. This is achieved by stacking filters that target process names, IDs, categories, and more.

Findings

After launching the binary, I started with a basic filter on the entire process (Figure 16). A summary was run on operation names, which revealed many registry operations. Due to this, another summary was run against the registry paths modified. This revealed changes in both the user and local machine hive. Next, I wanted to look at the network operations, as the previous analysis had revealed many indicators of worming/mass mailer activity. A TCP filter was added, which revealed thousands of reconnect requests to various MX addresses (Figure 17). Finally, to further analyze where file artifacts where placed, the TCP filter was replaced with a CreateFile filter (Figure 18). This revealed artifacts in SysWow64 and the user's AppData folder.

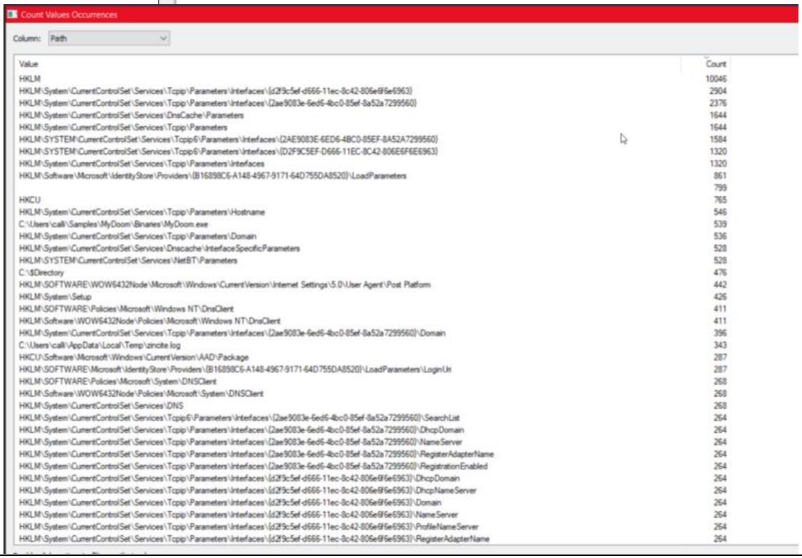
Analysis



Count Values Occurrences

Column: Operation

Value	Count
RegQueryValue	16577
RegQueryKey	9692
RegOpenKey	8656
ReadFile	6893
QueryDirectory	4428
CreateFile	3916
RegCloseKey	3496
RegSetInfoKey	3117
CloseFile	3114
WriteFile	1291
RegCreateKey	656
RegEnumValue	426
Thread Create	417
UDP Receive	416
UDP Send	416
TCP Reconnect	329
Thread Exit	294
QueryBasicInform...	273
CreateFileMapping	128
QueryStandardInf...	118
QuerySecurityFile	99
Load Image	90
QueryAttributeTa...	86
QueryAttributeInf...	62
QueryRemotePro...	62
QueryNameInfor...	57
SetDispositionInf...	55
SetBasicInformati...	32
QueryEaInformati...	31
QueryStreamInfor...	31
SetEndOfFileInfor...	31
RegEnumKey	26
RegSetValue	23
QuerySizeInforma...	2
Process Exit	1
Process Start	1



Count Values Occurrences

Column: Path

Value	Count
HKLM	10046
HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{d2f9c5ef-d666-11ec-8c42-806e6f6e6963}	2904
HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{2ae9083e-6ed6-4bc0-85ef-8a52a7299560}	2376
HKLM\System\CurrentControlSet\Services\DnsCache\Parameters	1644
HKLM\System\CurrentControlSet\Services\Tcpip\Parameters	1644
HKLM\SYSTEM\CurrentControlSet\Services\Tcpip6\Parameters\Interfaces\{2AE9083E-6ED6-4BC0-85EF-8A52A7299560}	1584
HKLM\SYSTEM\CurrentControlSet\Services\Tcpip6\Parameters\Interfaces\{D2F9C5EF-D666-11EC-8C42-806E6F6E6963}	1320
HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Interfaces	1320
HKLM\Software\Microsoft\IdentityStore\Providers\{B16898C6-A148-4967-9171-64D755DA8520}\LoadParameters	861
	799
HKCU	765
HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Hostname	546
C:\Users\cali\Samples\MyDoom\Binaries\MyDoom.exe	539
HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Domain	536
HKLM\System\CurrentControlSet\Services\Dnscache\InterfaceSpecificParameters	528
HKLM\SYSTEM\CurrentControlSet\Services\NetBT\Parameters	528
C:\$Directory	476
HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\User Agent\Post Platform	442
HKLM\System\Setup	426
HKLM\SOFTWARE\Policies\Microsoft\Windows NT\DnsClient	411
HKLM\Software\WOW6432Node\Policies\Microsoft\Windows NT\DnsClient	411
HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{2ae9083e-6ed6-4bc0-85ef-8a52a7299560}\Domain	396
C:\Users\cali\AppData\Local\Temp\zincite.log	343
HKCU\Software\Microsoft\Windows\CurrentVersion\AAD\Package	287
HKLM\SOFTWARE\Microsoft\IdentityStore\Providers\{B16898C6-A148-4967-9171-64D755DA8520}\LoadParameters\LoginUri	287
HKLM\SOFTWARE\Policies\Microsoft\System\DNSClient	268
HKLM\Software\WOW6432Node\Policies\Microsoft\System\DNSClient	268
HKLM\System\CurrentControlSet\Services\DNS	268
HKLM\System\CurrentControlSet\Services\Tcpip6\Parameters\Interfaces\{2ae9083e-6ed6-4bc0-85ef-8a52a7299560}\SearchList	264
HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{2ae9083e-6ed6-4bc0-85ef-8a52a7299560}\DhcpDomain	264
HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{2ae9083e-6ed6-4bc0-85ef-8a52a7299560}\NameServer	264
HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{2ae9083e-6ed6-4bc0-85ef-8a52a7299560}\RegisterAdapterName	264
HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{2ae9083e-6ed6-4bc0-85ef-8a52a7299560}\RegistrationEnabled	264
HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{d2f9c5ef-d666-11ec-8c42-806e6f6e6963}\DhcpDomain	264
HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{d2f9c5ef-d666-11ec-8c42-806e6f6e6963}\DhcpNameServer	264
HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{d2f9c5ef-d666-11ec-8c42-806e6f6e6963}\Domain	264
HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{d2f9c5ef-d666-11ec-8c42-806e6f6e6963}\NameServer	264
HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{d2f9c5ef-d666-11ec-8c42-806e6f6e6963}\ProfileNameServer	264
HKLM\System\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\{d2f9c5ef-d666-11ec-8c42-806e6f6e6963}\RegisterAdapterName	264

Figure 16 ProcMon Filter Name

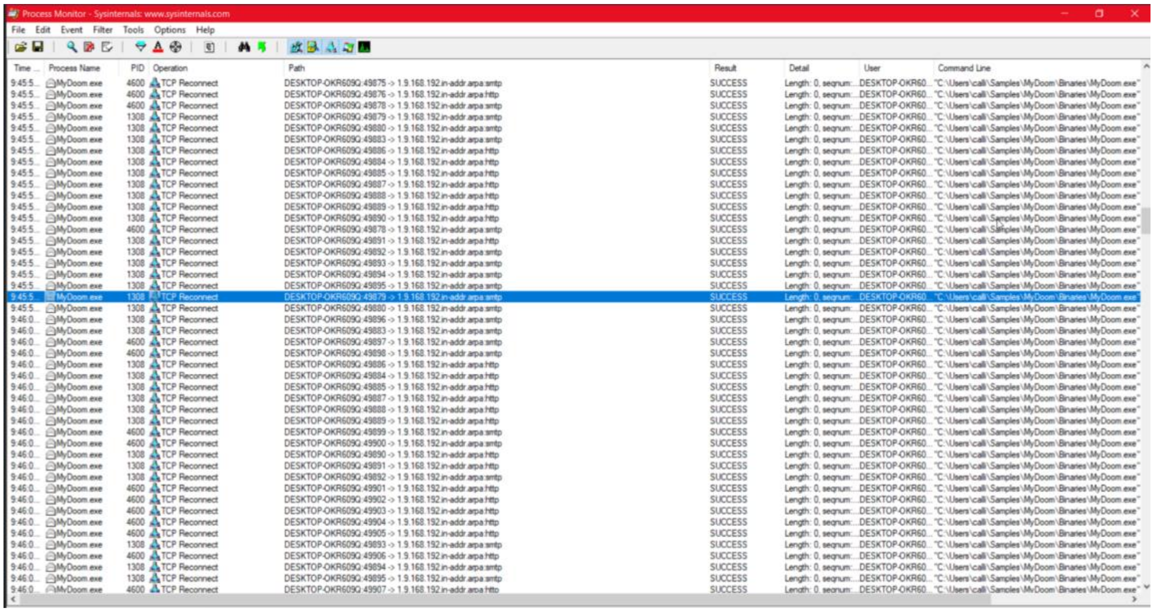


Figure 17 ProcMon Filter TCP

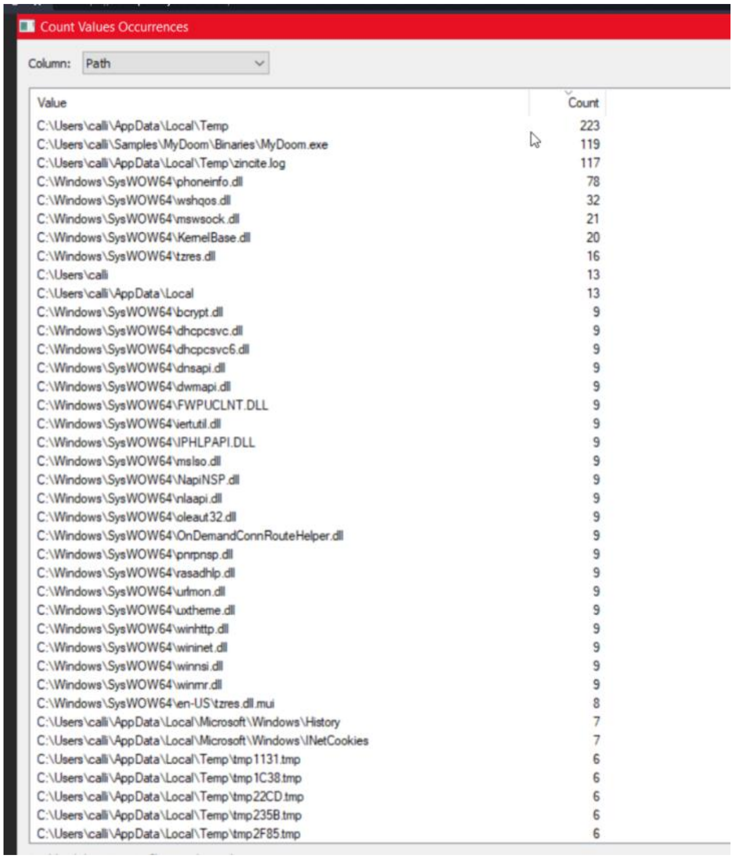


Figure 18 ProcMon Filter CreateFile

ApateDNS

ApateDNS helps malware analysts by setting up a fake DNS server on the local host (Figure 19). This redirects all the DNS requests from the binary to a console where the researcher can monitor the outbound connection requests.

Findings

As expected, the binary created thousands of DNS requests. These requests varied from search engine queries, MX hosts, and internal mail portals. Over a period of a few minutes nearly 1500 requests were made (Figure 20 DNS Requests).

Analysis

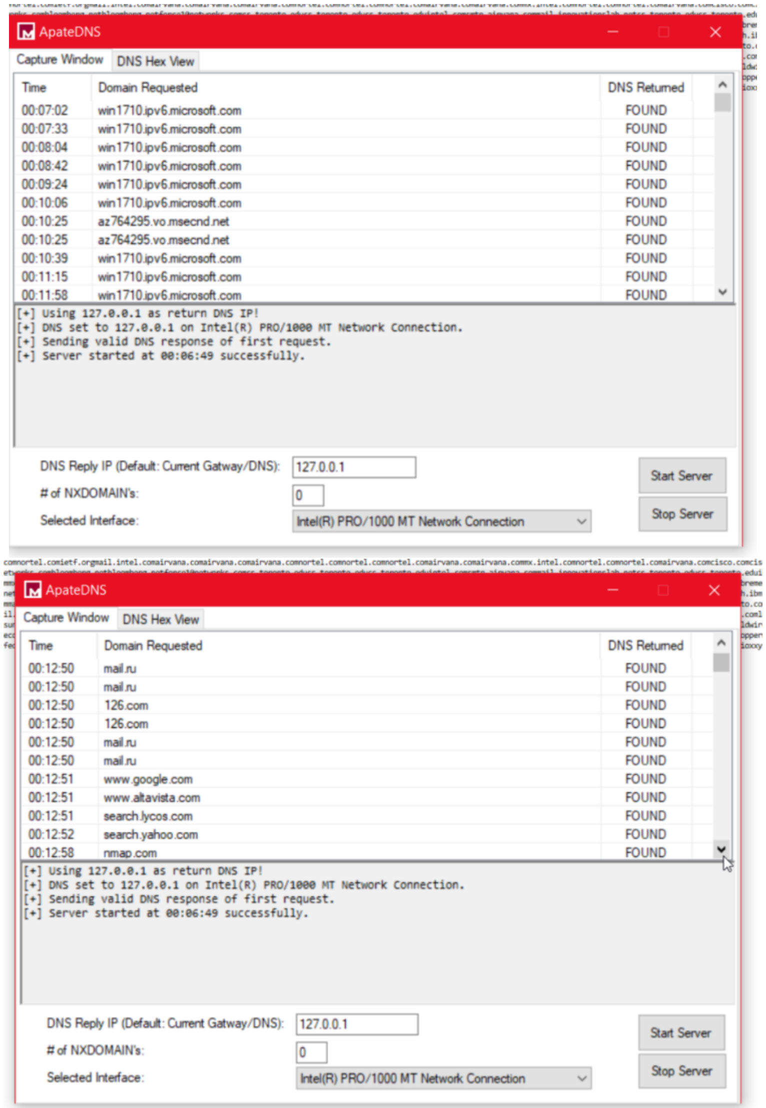


Figure 19 Fake DNS Server

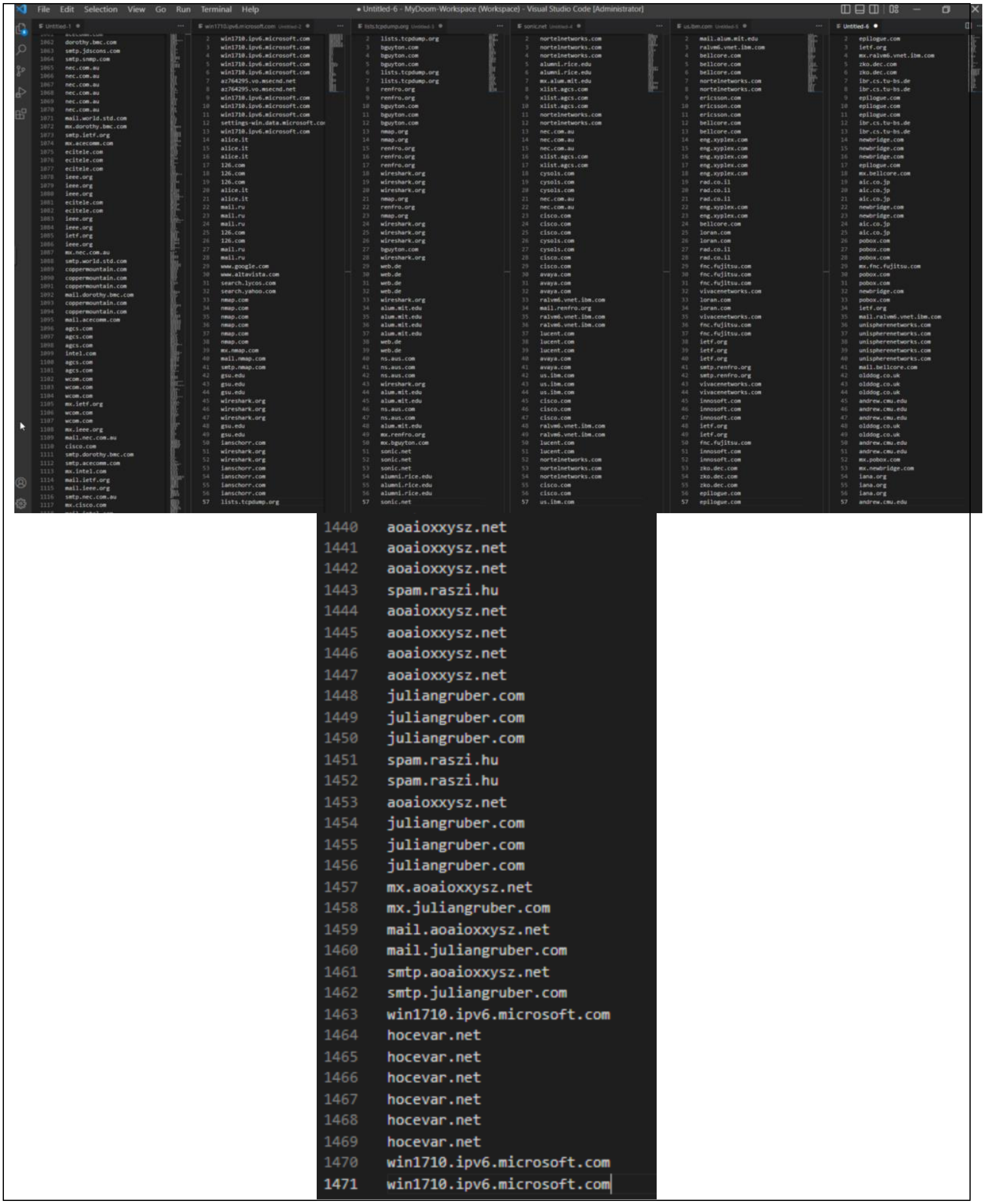


Figure 20 DNS Requests

netstat

Netstat is a basic networking command line tool that is used to show connections and listening ports on a host.

Findings

As indicated in the research, MyDoom creates a TCP listener that can be used to transfer files or command and control a large botnet of affected devices (Figure 21). Each variant opens a different port. As previously hypothesized, this further confirms that the sample is variant M (Port 1034).

Analysis

```
c:\Users\calli\Samples\MyDoom\Binaries>netstat -an

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    0.0.0.0:135            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:445            0.0.0.0:0              LISTENING
  TCP    0.0.0.0:1034           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:3389           0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49664          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49665          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49666          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49667          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49668          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49669          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49670          0.0.0.0:0              LISTENING
  TCP    0.0.0.0:49671          0.0.0.0:0              LISTENING
  TCP    127.0.0.1:55624        127.0.0.1:80           SYN_SENT
  TCP    127.0.0.1:55625        127.0.0.1:80           SYN_SENT
  TCP    127.0.0.1:55626        127.0.0.1:80           SYN_SENT
  TCP    127.0.0.1:55627        127.0.0.1:80           SYN_SENT
  TCP    127.0.0.1:55628        127.0.0.1:80           SYN_SENT
  TCP    127.0.0.1:55629        127.0.0.1:80           SYN_SENT
  TCP    127.0.0.1:55630        127.0.0.1:80           SYN_SENT
  TCP    127.0.0.1:55631        127.0.0.1:80           SYN_SENT
  TCP    169.254.214.217:5040   0.0.0.0:0              LISTENING
  TCP    192.168.9.14:139       0.0.0.0:0              LISTENING
  TCP    192.168.9.14:445       192.168.9.13:64321     ESTABLISHED
  TCP    192.168.9.14:55477     16.50.1.85:1034        SYN_SENT
  TCP    [::]:135               [::]:0                 LISTENING
  TCP    [::]:445               [::]:0                 LISTENING
  TCP    [::]:3389              [::]:0                 LISTENING
  TCP    [::]:49664             [::]:0                 LISTENING
  TCP    [::]:49665             [::]:0                 LISTENING
  TCP    [::]:49666             [::]:0                 LISTENING
  TCP    [::]:49667             [::]:0                 LISTENING
  TCP    [::]:49668             [::]:0                 LISTENING
  TCP    [::]:49669             [::]:0                 LISTENING
  TCP    [::]:49670             [::]:0                 LISTENING
  TCP    [::]:49671             [::]:0                 LISTENING
  UDP    0.0.0.0:53             *:*
  UDP    0.0.0.0:123            *:*
  UDP    0.0.0.0:500            *:*
  UDP    0.0.0.0:3389           *:*
  UDP    0.0.0.0:4500           *:*
  UDP    0.0.0.0:5050           *:*
  UDP    0.0.0.0:5353           *:*
  UDP    0.0.0.0:5355           *:*
  UDP    127.0.0.1:1900         *:*
```

Figure 21 Backdoor Listener

Advanced

x32Debug

x32Debug is a dynamic debugging tool that can be used to step through a binary as well as provides the ability to manipulate the CPU, heap and stack in real time.

Finding 1: DNS Query Algorithm

In the below analysis we can see the DNS requests being loaded into the stack before being registered in ApateDNS (Figure 22). This function is repeated numerous times as the binary makes its way through thousands of domains. With Wireshark, we can see each packet as the binary calls the GetHostByName API (Figure 23).

Analysis

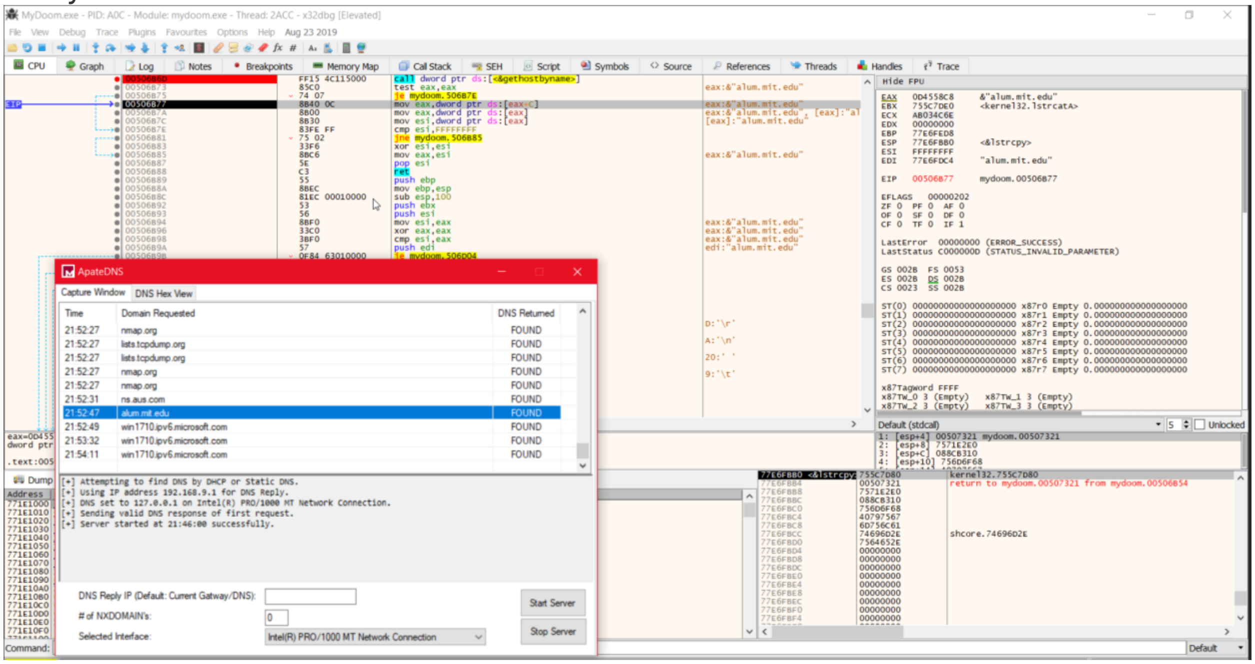


Figure 22 DNS Requests

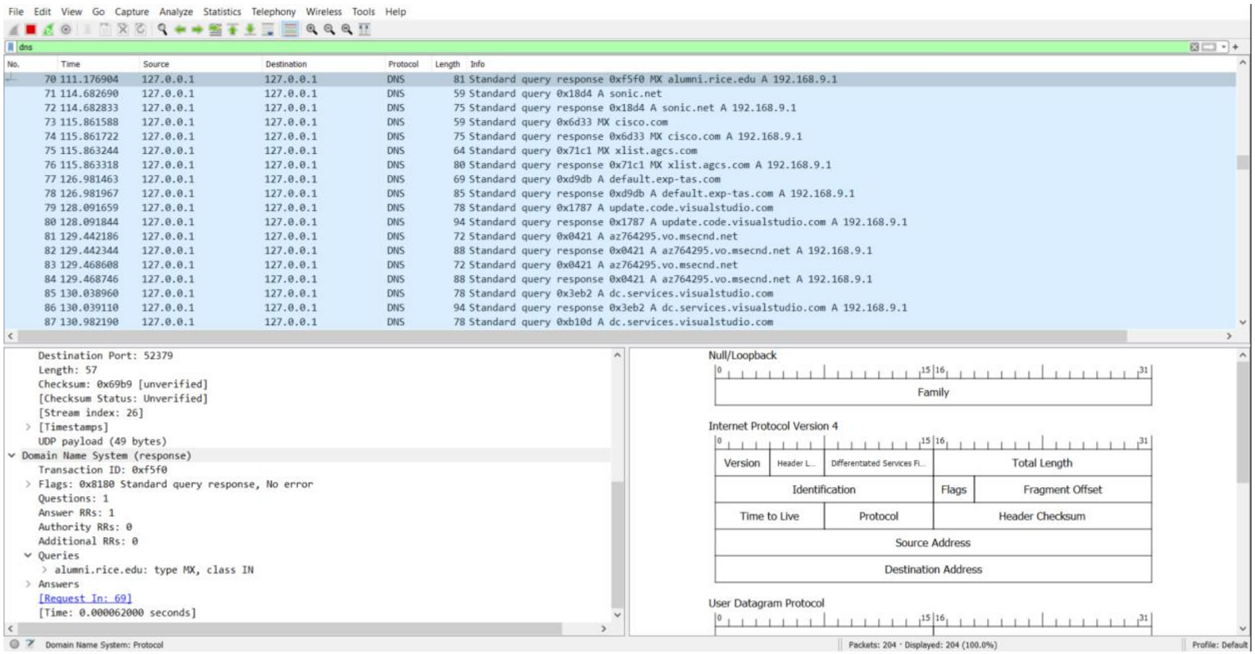


Figure 23 DNS Packets

Finding 2: Mass Mailer Algorithm

In the below analysis we can see where the binary calls the PostMessageA API (Figure 24). This call is repeated with each call to ESI, as the dereferenced memory location is loaded to this register on 005030FD.

Analysis

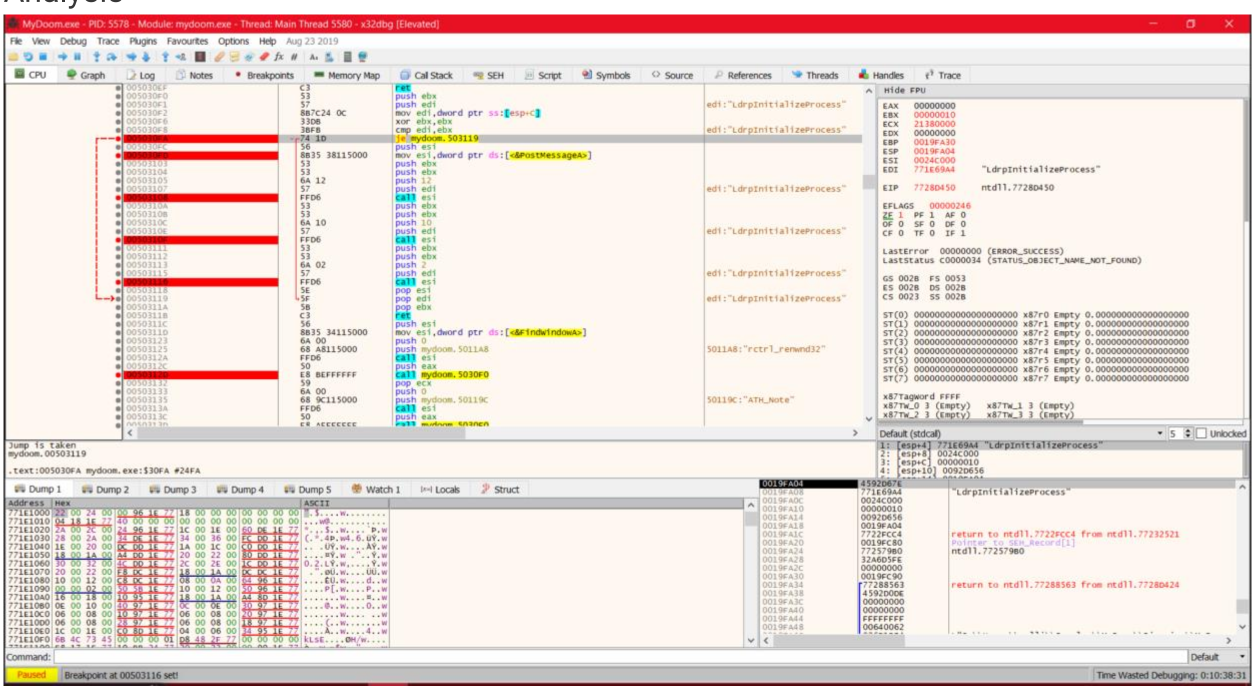


Figure 24 PostMessageA Debug

CONCLUSION

Potential Danger

The MyDoom malware has successfully been used in email phishing campaigns as well as distributed denial of service attacks against large enterprises. Once a host is infected, the malware configures itself to launch automatically and can use a significant number of resources attempting to replicate. This can cause the local host to slow down, and some variants will additionally lock files to cause even more damage to the host. Additionally, a backdoor is created that can be later used to deliver more malware or command affected hosts to attack another organization.

Malware Removal

The easiest way to detect and remove the MyDoom malware is to ensure your host has up to date antivirus software. Due to the age of this malware, most software will detect the hash, or signature of the file, as malicious. However, if the file is launched, it maintains persistence by modifying the registry and file system. System administrators should delete any unknown or suspicious startup tasks as well as the registry keys added from the RegShot Analysis.

It is important to note that this malware is typically delivered via social engineering techniques. With these types of attacks, prevention is key. To properly mitigate against this malware, a robust security awareness program should be implemented alongside signature-based detection.

APPENDIX A – TOOLS

1. Strings
2. FLOSS
3. UPX
4. PE View
5. PEiD
6. Resource Hacker
7. Virus Total
8. Ida Pro
9. RegShot
10. Process Monitor
11. ApateDNS
12. Netstat
13. Wireshark
14. X32Debug

APPENDIX B – REFERENCES

1. MyDoom Wikipedia
2. https://www.f-secure.com/v-descs/novarg.shtml
3. https://www.youtube.com/watch?v=cRH-khasTfg