



---

Annual ADFSL Conference on Digital Forensics, Security and Law

2017  
Proceedings

---

May 15th, 10:00 AM

## Kelihos Botnet: A Never-Ending Saga

Arsh Arora

University of Alabama, Birmingham, [ararora@uab.edu](mailto:ararora@uab.edu)


Max Gannon

University of Alabama, Birmingham, [gannonm@uab.edu](mailto:gannonm@uab.edu)

Gary Warner

University of Alabama, Birmingham, [gar@uab.edu](mailto:gar@uab.edu)

Follow this and additional works at: <https://commons.erau.edu/adfsl>

 Part of the [Defense and Security Studies Commons](#), [Forensic Science and Technology Commons](#), [Information Security Commons](#), [OS and Networks Commons](#), [Other Computer Sciences Commons](#), and the [Science and Technology Studies Commons](#)

---

### Scholarly Commons Citation

Arora, Arsh; Gannon, Max; and Warner, Gary, "Kelihos Botnet: A Never-Ending Saga" (2017). *Annual ADFSL Conference on Digital Forensics, Security and Law*. 4.

<https://commons.erau.edu/adfsl/2017/papers/4>

This Peer Reviewed Paper is brought to you for free and open access by the Conferences at Scholarly Commons. It has been accepted for inclusion in Annual ADFSL Conference on Digital Forensics, Security and Law by an authorized administrator of Scholarly Commons. For more information, please contact [commons@erau.edu](mailto:commons@erau.edu).

**EMBRY-RIDDLE**  
Aeronautical University™  
SCHOLARLY COMMONS

(c)ADFSL



# KELIHOS BOTNET: A NEVER-ENDING SAGA

Arsh Arora, Max Gannon, Gary Warner

University of Alabama at Birmingham  
1201 University Blvd, Birmingham, AL 35233  
{ararora, gannonm, gar}@uab.edu

## ABSTRACT

This paper investigates the recent behavior of the Kelihos botnet, a spam-sending botnet that accounts for many millions of emails sent each day. The paper demonstrates how a team of students are able to perform a longitudinal malware study, making significant observations and contributions to the understanding of a major botnet using tools and techniques taught in the classroom. From this perspective, the paper has two objectives: encouragement and observation. First, by providing insight into the methodology and tools used by student researchers to document and understand a botnet, the paper strives to embolden other academic programs to follow a similar path and to encourage such discovery. Second, the paper shares observations and insights gathered about the botnet’s recent spam activity showing evidence of the “spam as a service” model and demonstrating a variety of unique and dangerous spam campaigns conducted via the Kelihos botnet, including banking trojans, credential phishing, and ransomware attacks.

**Keywords:** Kelihos, Botnet, Malware, Spam, Ransomware, Banking Trojan, Pharma, Pump and Dump, Geo-Targeting

## 1. INTRODUCTION

While the Kelihos botnet first debuted in 2009, the current botnet is a functional derivative of two other famous botnets, Waledac, and the Storm Worm [Adair, 2012, Bureau, 2011]. The Kelihos botnet is known by a number of aliases in the community including Hlux and Slenfbot [Singh et al., 2014]. Kelihos evolved and became sophisticated over time, but no interruption could stop the working of Kelihos. Two widely celebrated takedowns of Kelihos have been performed by security companies. The first was Operation b79, conducted by Microsoft in September of 2009 [Nadji et al., 2013]. In March of 2012,

CrowdStrike, Dell Secureworks, Kaspersky, and others sinkholed 100,000 nodes of the Kelihos.B malware [Kerkers et al., 2014]. At RSA Conference 2013, CrowdStrike demonstrated a repeat performance, targeted Kelihos.C [Rossow et al., 2013, Werner, 2013]. Despite these admirable attempts, Kelihos continues to send significant volumes of spam on a daily basis [Stringhini et al., 2014]. Even after all of the takedown attempts, Kelihos was still ranked as the top spam-sending botnet in 2015 and continues to yield a significant spam volume today [McAfee, 2016, Tech, 2016].

The Kelihos botnet utilizes a peer to peer

network infrastructure that hides the location of the true command and control server [Dietrich et al., 2013]. Kelihos-infected computers communicate with one another and are used in two primary ways, depending on whether they have a publicly reachable IP address or not. Nodes which cannot be addressed from the Internet just send spam. Internet addressable nodes help to anonymize the location of the C&C server by acting as a multi-tiered proxy, receiving requests for spam details from spamming nodes, and requesting those same details from other nodes, eventually leading to a job server which is leased infrastructure and communicates with the C&C server, as documented by Fortinet’s Kyle Yang at BlackHat [Yang, 2012].

The most unique aspect of Kelihos is the diversity in its spam and its delivery mechanism. The spam diversity is primarily because Kelihos provides “spam as a service” offering the use of its spamming infrastructure for hire to deliver any messages for a sender who is willing to pay their fees. The spam messages are dominated by pharmaceutical spam, but are not limited to it. Pharma spam seems to be the fall-back position, when other higher paying customers have not hired the botnet to deliver something else. Kelihos has been observed sending spam for pump and dump manipulation of different stock symbols, money mule job applications, and credential phishing for Polish and French financial institutions. In the summer of 2016, Kelihos began geo-targeting for email delivery sending different messages and payloads based on the country-code Top Level Domain (ccTLD) of the email recipient (’.pl’,’.uk’). Malware, including ransomware and Zeus banking trojans, were delivered only to certain geographies using this technique. Kelihos has been seen spamming ransomware, sending WildFire [Arora & Warner, 2016b], CryptFile2 [Arora & Warner, 2016a] and Troldesh [Arora & Warner, 2016f] encryption ransomware fam-

ilies.

Another addition to the spam campaign is sending links to a Word document that will drop a variant of Zeus, specifically, geo-targeting for German and United Kingdom banks. In the current iteration, it uses stolen SMTP credentials to login as a legitimate user to a large number of mail servers and sends spam.

## 2. NETWORK

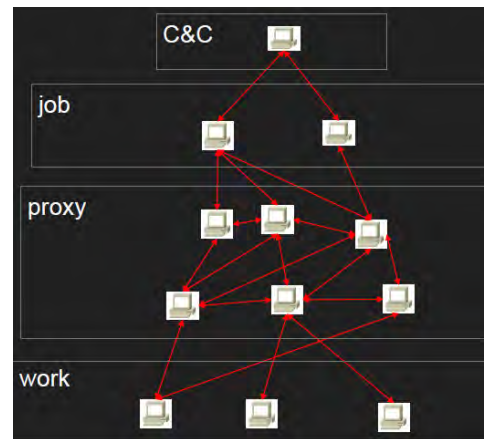


Figure 1. Courtesy: Kyle Yang outlining Kelihos distribution network [Yang, 2012]

Although the division of the botnet’s operation appears to be clear in Figure 1, the path is actually complex. Starting from the bottom, the chart displays the worker or client nodes, which sends spam to the end user and only communicate to the servers at the proxy level. The proxy level is the most interesting one as it connects with other proxies, clients, and job servers. This 3-way communication makes it the most attractive and the one that contains maximum information. The proxy level receives templates from other proxies and also communicates with the job servers. While Kelihos-infected Windows computers serve as the proxy and worker nodes, job servers are leased hardware controlled by the criminals. Job servers are the ones that serve as the main operators to the command and control center. Job servers are often in disguise and try to be invisible under the shadow

of different proxies; thus, making detection extremely complex and difficult. Lastly, the command and control center is handled by the bot herder and provides the information, spam templates, and updates needed to be distributed across the botnet.

### 3. INITIALIZATION

In this section, researchers will describe the pattern that is followed by the Kelihos binary to initiate, then communicate to its command and control center and eventually send spam. The process will be illustrated with screen captures from OllyDbg and Wireshark. The details are shared so that they can be easily replicated by any individual who has expertise with OllyDbg. The following analysis was performed in a virtual environment (VMWare) operating in NAT network mode.

#### 3.1 Processes

In the subsequent section, researchers will describe the process of how a Kelihos binary performs when we try to self-infect. The preliminary observations can be viewed with the help of programs such as Wireshark, Process Hacker, and OllyDbg 2.0 version. Next, we will explain the process of self-infecting with a complete description of the process as depicted in Figure 2. As soon as we launch the binary, it initiates and starts a child process. Once the child process gets fully activated, it terminates the parent process and starts the communication process. Initially, it tries to make a TCP connection with several Internet Protocol (IP) addresses which are hard-coded in the binary. There is a unique identifier in the message that helps to authenticate the communication coming from the Kelihos binary. Once a connection is established, it makes a request for configuration (config) files. The config files contain the information about the spam template and the email addresses to be used for sending spam messages.

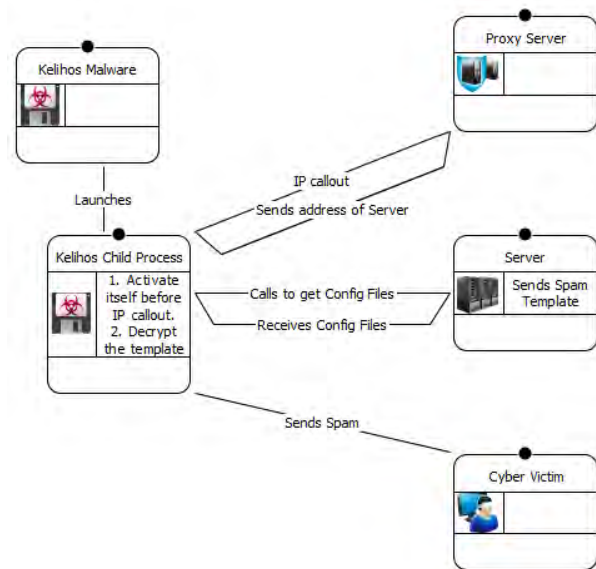


Figure 2. Architecture of Kelihos Malware

#### 3.1.1 Parent Process

In the following sub-section, researchers will demonstrate some breakpoints in OllyDbg that are helpful in performing the analysis. To attain optimum results, the following steps should be followed in the order mentioned.

After introducing a Kelihos binary into the virtual environment, start Process Hacker and OllyDbg as an administrator. Next, open the Kelihos binary in OllyDbg. Now, one can see a process being activated in Process Hacker. The following breakpoints should be enabled under the Executables (E) tab of OllyDbg:

1. Ntdll.dll
  - (a) RtlReportSilentProcessExit
  - (b) NtTerminateProcess
  - (c) NtResumeThread
2. kernel32.dll
  - (a) Resume Thread

Address	Module	Status	Disassembly	Comment
75C80F1C	kernel32	Active	MOV EDI,EDI	UINT kernel32.ResumeThread(Thread)
77836A08	ntdll	Active	MOV EAX,130	ntdll.NtResumeThread(guessed Arg1,Arg2)
778368C8	ntdll	Active	MOV EAX,172	ntdll.NtTerminateProcess(guessed Arg1,Arg2)
77840FDC	ntdll	Active	MOV EDI,EDI	ntdll.RtlReportSilentProcessExit(guessed Arg1,Arg2)

Figure 3. Breakpoints for Parent Process

The researchers provide the specific breakpoints in Figure 3, so that the results may be

easily replicated. Once we start debugging the program, it will stop at the breakpoints in the following manner:

1. RtlReportSilentProcessExit
2. NtTerminateProcess
3. RtlReportSilentProcessExit
4. NtTerminateProcess
5. ResumeThread
6. NtResumeThread

The process is complicated, but this method seems to obtain the maximum information from the parent process before moving to the child process. This child process can be observed in Process Hacker. The timing of when to attach to the child process is crucial, if one does it before or after the following steps mentioned above, the desired result may not be obtained. Next, we attach the child process in a new OllyDbg window with administrative rights.

### 3.1.2 Child Process

Once the child process is attached successfully, we check the Executables (E) box and set a breakpoint in the child process at the following location displayed in figure 4.

1. kernel32.dll

(a) Virtual Protect



Figure 4. Breakpoints for Child Process

By setting a breakpoint on kernel32.VirtualProtect, we can pause the malware execution only after the code has been unpacked. One way to understand this unpacking is to observe the DLLs listed in the Executables window in OllyDBG. Before running to the next breakpoint, there are only four DLLs under the Executable section, as shown in Figure 5.

After the execution pauses at the Virtual Protect breakpoint, we can see in figure 6, that the number is significantly greater than what was observed previously.

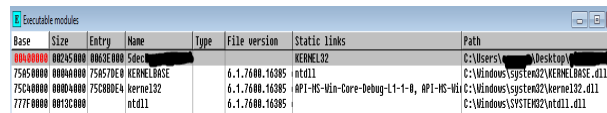


Figure 5. Pre-loaded Dlls at Virtual Protect

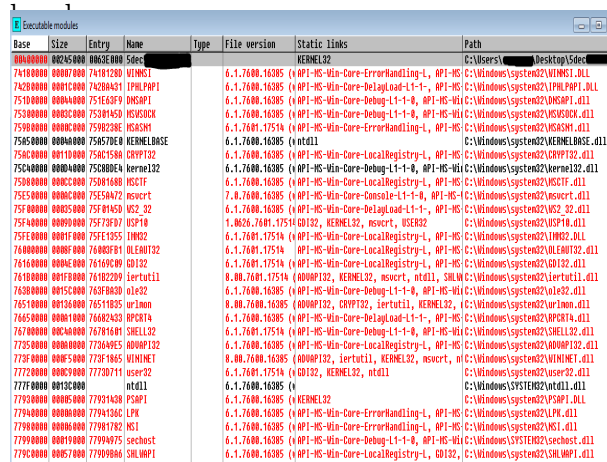


Figure 6. New DLLs loaded shown in Red

Next, the binary will decrypt itself and then proceed towards its communication channel.

### 3.2 Preparing for Take-Off

This section demonstrates a decoding loop that is run across the code within the binary. Figure 7 displays a certain memory address with non-human readable content.

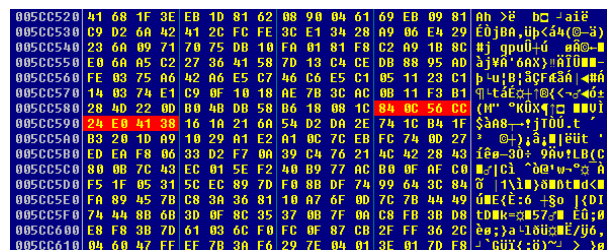


Figure 7. Encrypted Text

After applying this decoding function at the same memory location, displayed in Figure 8, now the same memory location contains readable ASCII code that will be used to request a configuration file from a peer node, displayed in Figure 9. The request is accompanied by NET\_SERVER\_WORKER ID and an IP address.

After decrypting and activating itself, the binary again stops at Virtual Protect to take

```

0063BE88 01DB ADD EBX,EBX
0063BE8A 75 07 JNE SHORT 0063BEC3
0063BE8C 8B1E MOV EBX,DWORD PTR DS:[ESI]
0063BE8E 83EE FC SUB ESI,-4
0063BE91 11DB ADC EBX,EBX
0063BE93 11C0 ADC EAX,EAX
0063BE95 01DB ADD EBX,EBX
0063BE97 73 0B JAE SHORT 0063BED4
0063BE99 75 28 JNE SHORT 0063BEF3
0063BE9B 8B1E MOV EBX,DWORD PTR DS:[ESI]
0063BE9D 83EE FC SUB ESI,-4
0063BE9F 11DB ADC EBX,EBX
0063BED2 72 1F JB SHORT 0063BEF3
0063BED4 48 DEC EAX
0063BED5 01DB ADD EBX,EBX
0063BED7 75 07 JNE SHORT 0063BEE0
0063BED9 8B1E MOV EBX,DWORD PTR DS:[ESI]
0063BEDB 83EE FC SUB ESI,-4
0063BEDE 11DB ADC EBX,EBX
0063BEE0 11C0 ADC EAX,EAX
0063BEE2 EB D4 JMP SHORT 0063BEB8

```

Figure 8. Function Call used for Decryption

```

005C0520 65 20 72 65 71 75 65 73 74 65 64 20 55 52 4c 20 e requested URL
005C0530 0d 0a 00 00 77 61 73 20 6e 6f 74 20 66 6f 75 6e was not found
005C0540 64 20 6f 6e 20 74 68 69 73 20 73 65 72 76 65 72 d on this server
005C0550 2e 3c 2f 70 3e 00 00 3c 2f 62 6f 64 79 3e 3c 2f .</p> </body></
005C0560 68 74 6d 6c 0e 00 00 47 45 54 20 00 00 00 00 html> GET
005C0570 50 4f 53 54 00 00 00 5b 4e 45 54 5f 53 45 52 POST [NET SER
005C0580 56 45 52 5f 57 4f 52 4b 45 52 20 00 61 72 6f 67 UER WORKER aron
005C0590 6f 72 69 00 58 20 52 65 61 6c 2d 40 79 2d 49 50 ori X-Real-Ip:IP
005C05A0 00 00 00 00 5b 4d 41 49 4e 5d 00 00 52 75 73 73 [MAIN] Russ
005C05B0 69 61 6e 0e 4e 6f 74 20 46 6f 75 6e 64 00 00 00 ian Not Found
005C05C0 3c 21 44 4f 43 54 59 50 45 20 48 54 4d 4c 2d 50 <DOCTYPE HTML P
005C05D0 55 42 4c 49 43 2d 22 2f 2f 49 45 54 46 2f 2f UBLIC "-//IETF//
005C05E0 44 54 44 20 48 54 4d 4c 2d 32 2e 30 2f 2f 45 4e DTD HTML 2.0//EN
005C05F0 22 3c 3c 6f 74 6d 6c 3c 68 65 61 64 3e 3c 74 >><html><head><t
005C0600 69 74 6c 65 3e 3c 30 24 0e 6f 74 2d 46 6f 75 ttle></html> Not Fou
005C0610 6e 64 3c 2f 74 69 74 6c 65 3e 3c 2f 68 65 61 64 nd</title></head

```

Figure 9. Decrypted text of the binary

care of some unfinished business. It loads nine new DLLs to perform internet related activities, as displayed in Figure 10. This behavior can be inferred because downloaded DLLs are related to wpcap, dhcp, and tcpip. Wpcap provides network intercept capabilities to the Kelihos binary, which is part of the method for gathering new user ID and password pairs for FTP and SMTP email traffic that are later used by other nodes for spamming and creating temporary internet file storage on stolen web servers.

Name	Path	File version	Static links
00400000 00400000 00000000	C:\Users\... \Desktop\...	1.0.0.0	...
00400000 00400000 00000000	C:\Windows\System32\...	6.0.6002.18005	...
00400000 00400000 00000000	C:\Windows\System32\...	6.0.6002.18005	...
00400000 00400000 00000000	C:\Windows\System32\...	6.0.6002.18005	...
00400000 00400000 00000000	C:\Windows\System32\...	6.0.6002.18005	...
00400000 00400000 00000000	C:\Windows\System32\...	6.0.6002.18005	...
00400000 00400000 00000000	C:\Windows\System32\...	6.0.6002.18005	...
00400000 00400000 00000000	C:\Windows\System32\...	6.0.6002.18005	...
00400000 00400000 00000000	C:\Windows\System32\...	6.0.6002.18005	...

Figure 10. Nine new DLLs loaded by Binary

To reiterate, there has not been any internet communication or activity so far in the process.

### 3.3 Communication

In this segment, the communication channel will be discussed in detail; reaching out to its peers, establishing a connection, getting the config files, decrypting the config files and preparing them to send spam messages.

#### 3.3.1 First Communication

From now on, we will require simultaneous monitoring of the network traffic which will be performed by Wireshark in the demo case. The initial communication starts immediately after the child process takes control of the operation after loading the required DLLs for proper working of the malware. During our research, it was found that the binary uses a particular location at a precise location to communicate with its peers. We concluded that there were approximately 145 hard-coded Internet Protocol (IP) addresses that the malware tries to reach out in sets of five each. The precise location can be found under the Memory (M) tab as shown in figure 11 and 12.

1. MSWsock.dll under second section

(a) NtDeviceIoControlFile at location - 7530616E

Address	Disasm	Module	Comment	Argv	Arg1	Arg2	Arg3	Arg4	Arg5
75300000	00001000	MSWSOCK	PE header	Inq	R	RWE	CopyOnWr		
75301000	00005000	MSWSOCK	.text,_SAMONTCP	Inq	R	E	RWE	CopyOnWr	
75302000	00002000	MSWSOCK	.data	Inq	RW	Copy	RWE	CopyOnWr	
75303000	00001000	MSWSOCK	.rsrc	Inq	R	RWE	CopyOnWr		
75303900	00003000	MSWSOCK	.reloc	Inq	R	RWE	CopyOnWr		

Figure 11. Location under MSWsock

Address	Module	Disasm	Comment
75301000	MSWSOCK	CALL 00001000 [C:\Windows\System32\MSWSOCK.dll]	Argv = 0, Arg1 = 12345, Arg2 = 1A, Arg3 = 0, Arg4 = 0
75301000	MSWSOCK	CALL 00001000 [C:\Windows\System32\MSWSOCK.dll]	Argv = 0, Arg1 = A, Arg2 = 12345, Arg3 = 0

Figure 12. Breakpoints under MSWsock

The location may vary due to using different machines, but the last four digits will remain constant. When performing the steps one can add 616E at the end of the location of the starting point of the MSWsock.dll. Once

you set the breakpoint at the following address, debug the program. The program will stop at the place with the function call:

```
MOV EBX, EAX
```

This is the location where the IP address gets stored before making a Transmission Control Protocol (TCP) connection. The value is stored in hex in the EAX registry and can be confirmed with Wireshark. As soon as the call is made, Wireshark displays a TCP connection request to the IP in the EAX registry. This can be viewed in Figure 13.

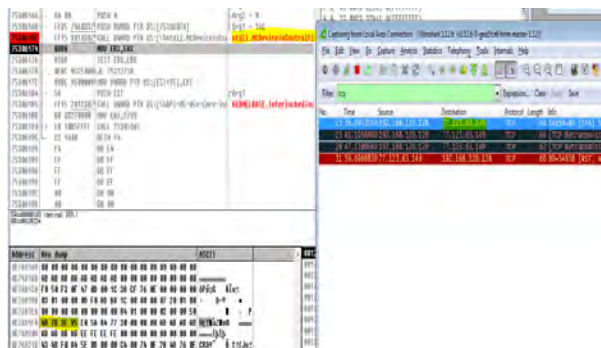


Figure 13. Proof of Hard-coded IP address

Since we knew the IP address from the Wireshark connection (77.123.63.149) it can be safely assumed that the IP is hardcoded in the binary. When we visited the hex address in the EAX Registry and checked for the value in '0E76D1F8' location it contained:

Hex 4D 7B 3F 95 to IP 77.123.63.149

This is the first check done by Kelihos binary to the following IP address, based on the hard-coded IPs in the binary. The binary attempts to establish the connection with the IP addresses one by one in the hope of securing a connection with one of them. In almost all cases, the binary is able to establish connection with one of the 145 IP addresses. If it is not able to establish the connection, it falls back to a list of hard coded domains to try and establish a connection. A list of these domains are shown in the figure 14.

The dominant ones are 'gorodkoff.com' and 'goloduha.info.' During the past year, Pas-

- 2014br.biz
- abrorra.biz
- avroran.biz
- bayermun.biz
- bypomsa.info
- chemp14.biz
- demyator.biz
- ecuad69.biz
- ekidjop.info
- fahhtaz.biz
- fucmethve.info
- goloduha.info
- gorodkoff.com
- gorotza.biz
- hockelen.info
- ibayermun.biz
- jagesxij.info
- jiqnipun.ru
- meuvbayt.info
- mydear.name
- newcounter.biz
- niggawhat.net
- niwrebsa.info
- pasbuyr.info
- pookagyx.info
- segbuktem.info
- usdivqo.info
- omyxiglet.info
- onabgitry.info
- ggabwav.info
- zadofadsun.info

Figure 14. List of the Fast Flux domain names sive DNS systems at Internet Identity (IID) recorded 3,261 distinct IP addresses used by goloduha.info. The researchers' monitoring station was observed by IID resolving gorodkoff.com, goloduha.info, and zavodchik-shop.com at least once during each month from July 2016 through December 2016, proving that our "proxy node" was being used as part of the Fast Flux hosting.

### 3.3.2 Second Communication

Once the initial TCP connection is established with one of the IP addresses, the binary tries to initiate a 164 bytes encrypted conversation with the IP address. The consistent thing in the communication is the presence of string 'IUUE..H@' at the starting of the message in the first 12 bytes, as depicted in Figure 15. This is the unique identifier that is used to confirm that Kelihos binary indeed sends the communication. Before this communication gets sent, it makes the following call:

```
Nt Device IO Control File 753044EE
```

The location can be found by the same manner as you found the previous location. In this case, you add '44EE' to the MSWsock.dll location and put a breakpoint at the following address. One may notice that the function call is the same, but it is the address which is the key here. Therefore, one has to be careful about the address because the function call is mentioned in ten

other places in the binary when searched in OllyDbg. Once the call is made, it uses 'WSA Send' to send the initial communication.

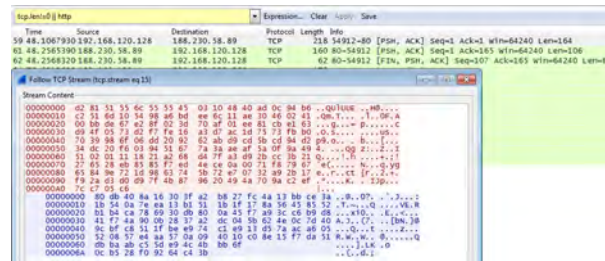


Figure 15. Initial communication with unique identifier 'IUUE..H@'

Next, it waits for a reply from the connected IP; in case it does not receive a response, it moves on to the next IP and performs the same process over and over again. Finally, it uses 'WSA Receive' when the binary can establish a connection and then it receives a consistent string of 227 bytes.

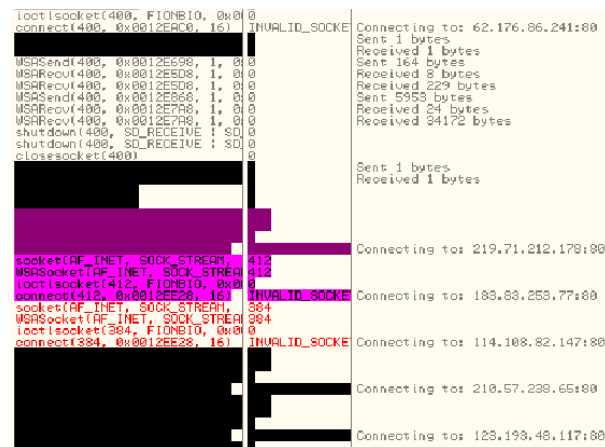


Figure 16. Olly Socket Trace displaying the back and forth communication

Next step, the binary sends a reply and communicates back and forth exchanging pertinent information with the established IP connection about how to obtain the config files. The information is encrypted, but the network communication can be viewed in Wireshark and can be confirmed with a special plugin named 'Olly Socket Trace,' being displayed in Figure 16.

### 3.3.3 Third Communication

This is the most important section of the communication cycle. The section includes the process of receiving the encrypted config files and then displaying the decryption of the config files into the various fields of 'To,' 'From,' 'Subject' and 'Email Body' for the spam template.

After the above communication is established, the binary requests to receive config files. For the following, we would need to continuously monitor Wireshark and for better results of the experiment use the filter 'http.request.method == GET.' The binary uses HTTP protocol, as shown in Figure 17, to request for config files with the usage of the 'GET' command.

GET /file.htm HTTP/1.1

Info	Protocol
GET /start.htm	HTTP/1.1
GET /start.htm	HTTP/1.1
GET /home.htm	HTTP/1.1
GET /home.htm	HTTP/1.1
GET /main.htm	HTTP/1.1
GET /main.htm	HTTP/1.1
GET /main.htm	HTTP/1.1
GET /search.htm	HTTP/1.1
GET /login.htm	HTTP/1.1
GET /login.htm	HTTP/1.1
GET /search.htm	HTTP/1.1
GET /file.htm	HTTP/1.1
GET /default.htm	HTTP/1.1
GET /default.htm	HTTP/1.1
GET /file.htm	HTTP/1.1

Figure 17. Some examples of GET config file request

Figure 18 displays a list of different '.html' file names that are used to obtain various GET config files:

- default
- file
- home
- index
- install
- login
- main
- online
- search
- setup
- start
- welcome

Figure 18. GET config files

After the files are received, the process of decrypting the data starts. Multiple levels of decoding, involving six to seven permutations, are used to convert the encrypted text into readable text. Figures 19 and 20 will give a more picturesque view of the commands use to decrypt the files and the decoded text.





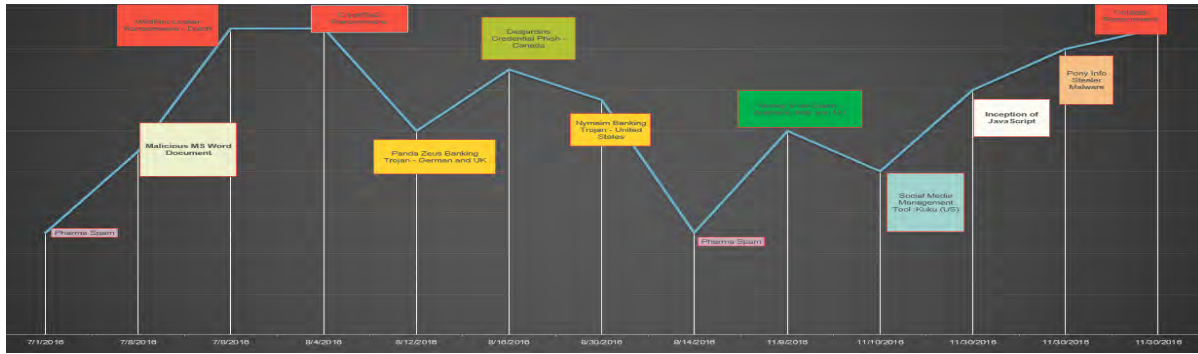


Figure 22. Timeline of recent Kelihos botnet spam campaign

Since taking down the Kelihos botnet with our partners Kyrus Inc. and Kaspersky Labs in September, the Microsoft Digital Crimes Unit has continued to actively investigate the case and pursue new leads with the goal of holding the perpetrators behind the botnet accountable for their actions.

In an amended complaint filed today with the U.S. District Court for the Eastern District of Virginia, Microsoft alleges that **Andriy N. Sabelnikov**, a citizen of Russia, is responsible for the operations of the **Kelihos botnet**.

Figure 23. Sabelnikov alleged to operate and control the Kelihos botnet

Canadian Health & Care Mall for the majority of its spam.



Figure 24. Snapshot of Canadian Health & Care Mall

#### 4.1.1 Domain Names

When the Kelihos botnet is spamming URLs for pharmaceutical products, the URL used in the spam does not point directly to the pharmaceutical website, but rather to a redirector site. However some domains have been used consistently over a period of many months, including those listed in Figure 25 25. Researchers used these domain names as beacons to identify spam-sending IP addresses infected with Kelihos.

The domains displayed in Figure 25 are different and recurring at times in the spam campaign.

asvhpwti[.]ru	curativeonlinegroup[.]be
bleguzjn[.]ru	gnhwigm[.]be
blvqekte[.]ru	bestpillmart[.]com
canadianrxsale[.]ru	canadianherboutlet[.]com
curativebestcompany[.]ru	curativepillshop[.]com
curativebestinc[.]ru	fasthealingelement[.]com
curativeonlinedeal[.]ru	organicremedyreward[.]com
curingfamilymall[.]ru	perfectcuringshop[.]com
curingfamilytrade[.]ru	secureorganictrade[.]com
curingfirsttrade[.]ru	smartpharmacymarket[.]com
curinghotpurchase[.]ru	sveuxfrd[.]com
curingonlineservice[.]ru	thepillpurchase[.]com
dgwghslr[.]ru	onlineremedydart[.]eu
familydrugsreward[.]ru	abargainingassociation[.]ru
firstdruginvestment[.]ru	arbjubfn[.]ru

Figure 25. Domain names spammed by Kelihos

## 4.2 Pump and Dump Spam

The second most dominant spam affiliated to Kelihos botnet is pump & dump spam. These types of campaigns last a couple of weeks on average. The pump & dump operators trick people into buying penny stocks advertised in the spam messages [Bouraoui, 2015] [Nelson et al., 2013]. The spam promises that the stock is on the rise and its value will increase drastically, often with fraudulent or misleading statements about a new contract, patent, or discovery that is about to make the company much more valuable. Generally, these stocks are priced for less than 50 cents while some are 10-12 cents. The spam messages promise closing prices of 3-4 dollars in a short period of time. Stock symbols spammed by Kelihos over the last year or so were:

- GRYN Green Hygienes Holdings Inc.
- APTY APT Systems Inc.

- FZRO FlashZero Corp.
- CWTC Clearwave Telecommunication Inc.
- UPOT Indie Growers Association, Inc.
- SNXG Sunx Energy

Most of the stock symbols were spammed for a couple of weeks, but a few of them came back after taking a break. For example, APTY was first seen on January 22, 2016 and then seen frequently in January, February, and March. It was the most consistent symbol that was being spammed by Kelihos botnet for a long period.

### 4.3 Money Mule Spam

The Kelihos botnet also often sends spam for money mule or work at home position vacancies. The targets are mostly based on language and location. In this particular spam, the operators seek for unemployed workers or people looking for a job. They urge the reader to reply back to an email address with his/her personal information, promising to pay handsomely with flexible work hours. One of the biggest factors for this spam campaign was that the employee would have the option to work from home, which allowed readers to reply back with their information promptly. Different languages such as English, French, Italian, Spanish, Portuguese, and Dutch have been noted in this particular spam campaign. Notable organizations like Apple and Walmart were also used in some spam campaigns to attract readers.

### 4.4 Credential Phish Spam

Frequently, we encounter multi-language credential phish. In recent months, the Kelihos botnet targets mostly Polish and French banks, and attempts to steal user credentials by asking the victim to verify their information via clicking a link. The link leads to a phishing website and tries to steal the victims username and password. An example of such

an email was documented in April 2016 and can be seen in the figure 26:



Figure 26. Polish Bank Phish Email

## 4.5 Kronos Banking Trojan

An interesting thing that was observed on November 9, 2016 was that, in addition to spamming the world, the Kelihos binary behaved covertly and dropped Microsoft Word document on the infected machines desktop [Arora & Warner, 2016e]. The document name can be viewed in Figure 27.



Figure 27. Pictorial view of the document link on the infected user's desktop

The intention was to surprise the user and hope that the document will be viewed out of curiosity. As expected, the 'oldversion.doc' was a malicious document and after clicking 'Enable Content,' it downloaded Kronos banking trojan malware. This was confirmed with Figure 28 that contains a string found in Ollydbg

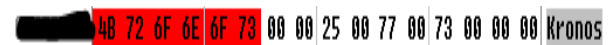


Figure 28. Confirmation of Kronos malware with the help of OllyDbg

## 5. RANSOMWARE

### 5.1 WildFire Ransomware

The latest addition to the Kelihos botnet was WildFire Ransomware. On July 8, 2016, a Kelihos sample was encountered that was

sending a Dutch-language spam with a link to a Microsoft Word document [Arora & Warner, 2016b]. The behavior has never been seen before and it was the first time that Kelihos botnet was distributing links to ransomware. Ransomware has evolved drastically and seems to be the most profitable form of malware distributed by malware authors. Through a single click on a malicious link or enabling a macro script, the malware has the ability to encrypt all the files on a victim’s machine and requests a payment in return for decrypted files. Figures 29 and 30 display the original email in the Dutch language and its translation to English.



Figure 29. Word document link being distributed in the email

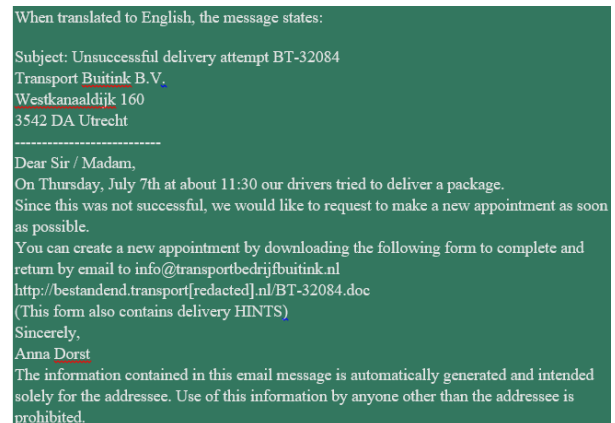


Figure 30. Word document email translated in English

In this particular case, the message informs the user of an unsuccessful package delivery. The user is required to fill a form in order to reschedule the delivery and is requested to click the embedded link to download a Microsoft Word document. Once the Word document opens, the document is presented in “Protected View” and requires “Enable Editing”.



Figure 31. Word document in protected view

Once enabled, it followed a similar process required by most hostile macro scripts “Enable Content”. After fulfilling all requirements by the Word document, the macro leverages a connection with its command and control center in order to encrypt the files on the machine. The process can be better understood with help of Figures 31, 32 and 33



Figure 32. Word document asking to enable content



Figure 33. Wildfire Encryption Ransomware ransom note

Different payment sites are listed where the user can pay the ransom amount in Bitcoin and receive the decryption key. This was a significant change for the Kelihos botnet given that it normally targets victims with

less aggressive methods. The introduction of ransomware made the long lasting Kelihos botnet extremely treacherous.

## 5.2 CryptFile2 Ransomware

On August 4, 2016, another interesting addition to the long-lasting Kelihos spam was that American Airlines themed spam messages delivered Microsoft Word documents, which eventually delivered CryptFile2 ransomware [Arora & Warner, 2016a]. The ransom note is mentioned in Figure 34



Figure 34. CryptFile2 Encryption Ransomware ransom note

## 5.3 No\_More\_Ransom Ransomware

On November 30, 2016, Kelihos spammed a wide variety of payloads, including a banking trojan, credential phishing, and a new ransomware variant. The criminals altered their malware to change the extension of encrypted files to “.nomoreransom” poking fun at the “No More Ransomware” campaign [No More Ransom, 2016] started by National High Tech Crime Unit of the Netherlands police, Europol’s European Cybercrime Centre, Kaspersky Lab, and Intel Security. Rather than being cowed by the new public service campaign, the criminals challenged them. This particular spam campaign was geo-targeting Australian email addresses ending in ‘.au’ with a lure impersonating Bank of America to trick the users to believe that the spam is legitimate and coming from a well-established corporation.

### 5.3.1 Inauguration of JavaScript

In this campaign, the infection vector changed from a Microsoft Word document to a JavaScript file. Clicking the link in the email downloaded a zip file which contained a small JavaScript program [Arora & Warner, 2016f]. The necessary details of the zip file are visible in Figure 35

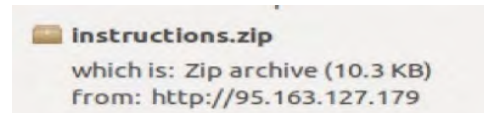


Figure 35. Downloaded zip file containing JavaScript

After the JavaScript is executed, the malware contacts the command and control center and encrypts the file system. Afterwards, it displays the following ransom note, noted in Figure 36, in Russian as well as in English.

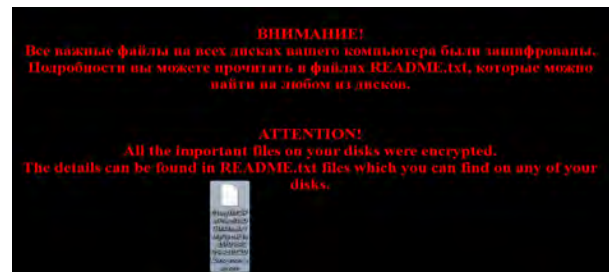


Figure 36. Ransom note of Troldesh ransomware in Russian and English

As can be seen in the ransom note, the extension of the file is changed to ‘no\_more\_ransom’. On further analysis, it was found that it was ‘Troldesh’ ransomware.

### 5.3.2 Pony Info Stealer Malware

After encrypting the files on the victim’s machine with the Troldesh ransomware, the malware also downloaded the Pony information stealer malware. This behavior is unique and has never been associated with Kelihos malware. Even though the files are encrypted, the malware still wants to acquire additional information from the victim’s computer. Figure 37 depicts the admin page of Pony malware.

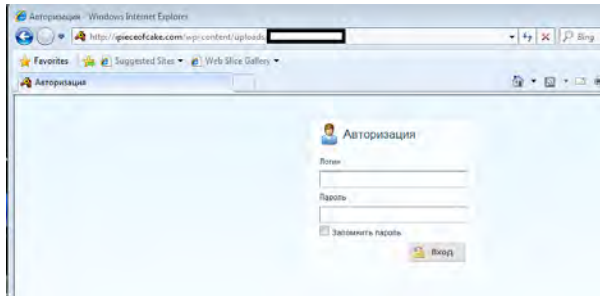


Figure 37. Pony Info Stealer Malware Admin Panel

## 6. GEO-TARGETED SPAM

### 6.1 Europe Geo-Targeted with Panda Zeus

Perhaps the clearest example of geo-targeting comes from a set of spam campaigns conducted on August 12, 2016 by Kelihos. The botnet was geo-targeting based on the top level domain of the recipient’s email addresses [Arora & Warner, 2016d]. Some examples of the specific targeted spam campaigns were:

- *If you are a “.de” you get “German banking” spam that drops a Word doc that leads to a Panda Zeus infection*
- *If you are a “.co.uk” you get “British banking” spam that drops a Word doc that leads to a Panda Zeus infection*
- *If you are a “.it” you get invited to start a romance with lyudmilafedoji@gmail[.]com*
- *If you are anyone else, you get pill spam.*

The Zeus banking trojan is well-known among the information security community. Zeus is a form of malware that targets various operating systems and tries to steal financial data; especially, banking information, credit card, social security, and so on [Kaspersky, 2010]. While this seemed a dramatic change in the Kelihos botnet spamming techniques,

it is merely a sign of the “Spam as a Service” model used by Kelihos. Any malware actor can hire the Kelihos spammers to deliver their message or payload. The same mechanism, as previously mentioned, was used to spread several different malware campaigns and target the geographies desired by the customer.

### 6.2 Canada Geo-Targeted with Desjardins Phish

On August 16, 2016, another example of geo-targeting was observed. Email addresses ending with “.ca” were targeted with a French language spam message for one of many Desjardins phishing websites [Arora & Warner, 2016c]. The link advertised in the spam went was of a credential phishing form, intended to steal the username and password of users of Desjardins Bank. The spam message in French and English is shown in Figure 38.

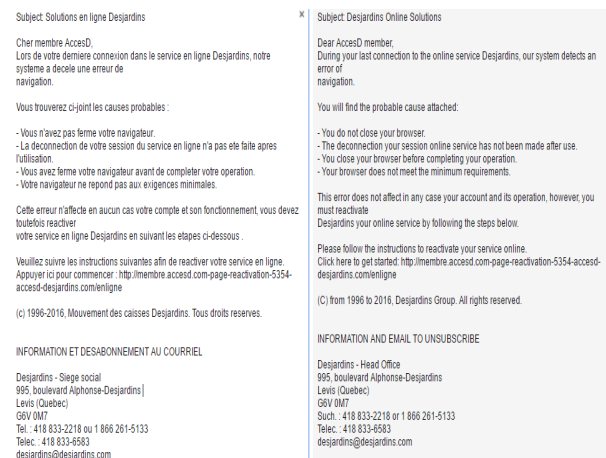


Figure 38. French Desjardins Phishing Email; Google Translate

### 6.3 United States of America Geo-Targeted with Nymaim

On August 30, 2016, Kelihos began geo-targeted US based emails ending with the top level domain .us by spamming the online shopping site Amazon. The Kelihos botnet

was delivering Nymaim banking trojan with an Amazon Gift Card theme [Arora et al., 2016]. In this case, the link in the spam body downloaded a Word document, which downloaded a Nullsoft Installer, and eventually executed the Nymaim banking trojan.

## 7. BEHAVIOR

It is well known that the Kelihos botnet behaves differently based on the network connection type [Yang, 2012]. The behavior changes in accordance to the network communication and ability to send spam. The network connections that will be referred in the following section are:

1. Network Address Translation (NAT) - Private IP and internal facing
2. Bridged Mode - Public IP and external facing

### 7.1 Network Address Translation

When the Virtual Machine or the infected computer is in NAT mode, with an IP address that is not directly reachable from the Internet, the Kelihos botnet behaves like a client or work server. It connects to a single node and receives the information. This means that it receives a single spam template and sends out the same spam message to everyone with an identical subject and message content. A single URL and similar text is used in the spam message that is distributed in a single run.

### 7.2 Bridged

When the Virtual Machine or infected computer is in bridged mode, meaning that it requests or is assigned a publicly addressable IP number, the Kelihos botnet drastically changes its behavior. Kelihos acts as a proxy server, where it receives not one but many spam templates and send them in different directions. Other Kelihos nodes approach

the server to obtain various spam messages for them to send. In bridged mode, it acts as a proxy as well as a client; thus, receiving spam templates and forwarding them to various other clients or servers.

#### 7.2.1 Credential Exchange

Each Kelihos-infected computer has the capacity to steal userids and passwords from the user logged in at that station, particularly focusing on email and FTP credentials. Stolen credentials are passed upstream to the C&C server. When a node is in bridge mode, Kelihos receives some of these stolen credentials to test and logs into various targets using the FTP addresses and checking for the ability to write files. The SMTP credentials are tested and, if they work, are used to send spam disguising itself as the user whose credentials were used. Kelihos receives a list that includes ‘Username’, ‘Password’, ‘From’, and ‘To’ to be used for the spam message. After a successful establishment of connection and logging into the machine as the user, the spam is sent out using the same email address within one or two seconds. Another interesting observation is that the same server IP address is used for different username and password logins. Once the email is sent, the malware logs in with a different account and keeps on repeating the process. On average, for a 30 minute run it exchanges approximately 700-800 FTP and SMTP credentials combined. With the given quantity, one can imagine how many credentials are being circulated across the globe. This is one of the main reasons why Kelihos is considered to be the most poisonous and resilient spam botnet in the entire world.

## 8. FUTURE WORK

While the Kelihos botnet has successfully endured through multiple take-down attempts, the researchers share everything they learn about the botnet with law enforcement and the greater anti-virus community. We will

continue to document the Kelihos botnet's regular spam campaigns and seek a greater understanding of its infection and communication methodologies as we train additional team members and begin to pursue other botnets in a similar fashion. We would welcome contact with other researchers who would like to see similar programs adopted in their research or teaching curriculum.



## REFERENCES

- Adair, S. (2012, December). *Could it be storm worm 3.0/ waledac 2.0?* Retrieved from <http://www.shadowserver.org/wiki/pmwiki.php/Calendar/20101230>
- Arora, A., Gannon, M., & Warner, G. (2016, August). *Amazon gift card from kelihos!* garwarner.blogspot.com. Retrieved from <http://garwarner.blogspot.com/2016/08/amazon-gift-card-from-kelihos.html>
- Arora, A., & Warner, G. (2016a, August). *American airlines spam from kelihos delivers ransomware.* garwarner.blogspot.com. Retrieved from <http://garwarner.blogspot.com/2016/08/american-airlines-spam-from-kelihos.html>
- Arora, A., & Warner, G. (2016b, July). *Kelihos botnet delivering dutch wildfire ransomware.* garwarner.blogspot.com. Retrieved from <http://garwarner.blogspot.com/2016/07/kelihos-botnet-delivering-dutch.html>
- Arora, A., & Warner, G. (2016c, August). *Kelihos botnet sending geo-targeted desjardins phish to canadians.* garwarner.blogspot.com. Retrieved from <http://garwarner.blogspot.com/2016/08/kelihos-botnet-sending-geo-targeted.html>
- Arora, A., & Warner, G. (2016d, August). *Kelihos botnet sending panda zeus to german and uk banking customers.* garwarner.blogspot.com. Retrieved from <http://garwarner.blogspot.com/2016/08/kelihos-botnet-sending-panda-zeus-to.html>
- Arora, A., & Warner, G. (2016e, November). *Kronos banking trojan and geo-targeting from kelihos.* garwarner.blogspot.com. Retrieved from <http://garwarner.blogspot.com/2016/11/kronos-banking-trojan-and-geo-targeting.html>
- Arora, A., & Warner, G. (2016f, November). *Nomoreransom aka troldesh ransomware delivered by kelihos.* garwarner.blogspot.com. Retrieved from <http://garwarner.blogspot.com/2016/11/nomoreransom-ransomware-by-kelihos.html>
- Bouraoui, T. (2015). Does pump and dump affect stock markets? *International Journal of Trade Economics and Finance*, 45.
- Bureau, P.-M. (2011). Same botnet, same guys, new code. Virus Bulletin International Conference.
- Commission, F. T. (2016, February). *Cases and proceedings: Sale slash, llc.* Retrieved from <https://www.ftc.gov/enforcement/cases-proceedings/142-3247/sale-slash-llc>
- Dietrich, C. J., Rossow, C., & Nobert, P. (2013). Cocospot: Clustering and recognizing botnet command and control channels using traffic analysis. *Elsevier*, 475-486.
- Kaspersky. (2010). *Kaspersky lab.* Retrieved from <https://usa.kaspersky.com/internet-security-center/threats/zeus-trojan-malware#.V97n0vArKM8>
- Kerkers, M., Santanna, J. J., & Sperotto, A. (2014). Characterisation of the kelihos. b botnet. Springer Berlin Heidelberg: IFIP International Conference on Autonomous

- Infrastructure, Management and Security.
- McAfee. (2016, June). *McAfee labs threats report*. Retrieved from <http://www.mcafee.com/us/resources/reports/rp-quarterly-threats-may-2016.pdf>
- Meisner, J. (2011, September). *Microsoft neutralizes kelihos botnet, names defendant in case*. Microsoft. Retrieved from <http://blogs.microsoft.com/blog/2011/09/27/microsoft-neutralizes-kelihos-botnet-names-defendant-in-case-2/#sm.0000i7mgzgrjpfddy8k1vahomseh5>
- Nadji, Y., Antonakakis, M., Perdisci, R., Dagon, D., & Lee, W. (2013). Beheading hydras: performing effective botnet takedowns. New York, NY, USA: ACM SIGSAC Conference.
- Nelson, K. K., Price, R. A., & Rountree, B. R. (2013). Are individual investors influenced by the optimism and credibility of stock spam recommendations? *Journal of Business Finance & Accounting*, 1155-1183.
- No more ransom*. (2016, July). Retrieved from <https://www.nomoreransom.org/>
- Rossow, C., Andriess, D., & Werner, T. (2013). Sok: P2pwned-modeling and evaluating the resilience of peer-to-peer botnets. Security and Privacy (SP), 2013 IEEE Symposium.
- Singh, K., Guntuku, S. C., Thakur, A., & Hota, C. (2014). Big data analytics framework for peer-to-peer botnet detection using random forests, information sciences. *Elsevier*, 278(0020-0255), 488-497.
- Stringhini, G., Hohlfeld, O., Kruegel, C., & Vigna, G. (2014). The harvester, the botmaster, and the spammer: on the relations between the different actors in the spam landscape. New York, NY, USA: 9th ACM symposium on Information computer and communications security (ASIA CCS '14).
- Tech, M. (2016, August). *Significant increase in kelihos botnet activity*. Retrieved from <https://www.malwaretech.com/2016/08/significant-increase-in-kelihos-botnet-activity.html>
- Werner, T. (2013, February). *Kelihos botnet taken down live on stage*. Retrieved from <http://www.h-online.com/security/news/item/Kelihos-botnet-taken-down-live-on-stage-1813580.html>
- Yang, K. (2012, July). *Fighting against kelihos botnet*. BlackHat. Retrieved from <https://media.blackhat.com/ad-12/Yang/bh-ad-12-the-endless-game-yang-slides.pdf>

