Sports Corruption Case

Jack Sime

CMP416-Advanced Digital Forensics

BSc Ethical Hacking Year 4

2022/23

Note that information contained in this document is for educational purpose.

.

Contents

.

1 INTRODUCTION & AIM

The investigator has been asked to work on an international sporting corruption case through a national security agency. Three network packet captures have been provided for investigation, along with a list of some necessary details the security agency aims to recover relating to the case.

The aim is to recover evidence of the case within the provided network captures. This overall aim comprises several sub-aims:

- Locate areas of interest within the provided packet captures
- Extract related information from within the packet captures
- Ensure forensic integrity by using read-only folders for provided evidence and noting evidence hash values, and performing regular comparisons
- Present the found evidence from each packet capture provided

2 METHODOLOGY

Before conducting the investigation, the investigator was provided with intel on each packet capture. The provided intel outlined what was suspected to be within the packets, any known protocols or methods used, and the evidence that was a priority to extract. Forensic integrity was ensured throughout the investigation through hash calculations of files and read-only folders to ensure files weren't modified. For this investigation, the forensic investigator followed a general methodology when investigating each packet capture and handling evidence found. The methodology was adapted during each capture to apply to the situation and data found within each capture, and this allowed the investigator to uncover information efficiently. The general methodology can be seen below for the packet captures and once any potential evidence has been extracted.

- Packet Capture Method
    - Conduct Statistical Flow analysis to gather intel on potential IP addresses and ports
    - Analyse capture utilising provided intelligence and information from Flow analysis
    - Locate Items of interest
    - Filter out unrelated traffic within the capture to isolate items of interest
    - Examine packets for crucial details such as protocols, IP addresses, distinguishable details
    - Export files that could potentially be evidence

- Evidence Analysis Method
    - Anti-forensic methods (Steganography, obfuscation, encoding)
    - Examine timestamps to establish a timeline
    - Reconstruct any potential evidence that has been fragmented

3 INVESTIGATION TOOLS

During the investigation, a wide range of tools was used to aid in the forensic analysis of the packet captures and the successful recovery of any evidence that was found.

Application	Function	Key features utilised	Captures used in
Wireshark	It can be used to capture and analyse network traffic	The packet analysis feature was extensively used to analyse suspect packets and gather necessary details on potential suspects, like their IP addresses and devices being used. The export function was also extensively used to extract identified artefacts within the packet captures. The filter feature within Wireshark allowed for the investigator to quickly filter out traffic that was deemed to be irrelevant and allowed them to filter data by key details noticed within packets.	Wireshark was used within every packet capture due to its large utility
YAF & SiLK	YAF was used to convert the PCAP file to an IPFIX format to allow it to be further analysed. SiLK was used to analyse the newly created IPFIX files through several different tools and scripts that make up SiLK	Several essential scripts within SiLK were used on the Captures. rwfilter allowed the investigator to filter the traffic based on supplied conditions. This was used to filter for specific ports and was then passed to another script. rwstats returned a few fields depending on what was selected. This was used to return bytes and packets sent and received from addresses that were over 0.5% of the total packets in the capture.	Used on Capture 1 as Capture 2 & 3 had intel to suggest the area of the packet to investigate
Binwalk	Allows for the searching of a binary image for embedded files and code	The ability to search image files was utilised within the investigation to uncover hidden files, including a python script. Binwalk was also used to extract these hidden files from the image files.	Binwalk was used within Capture 1 & 2 to check the image files found within the captures.
CyberChef	A tool created by GCHQ for analysing and decoding data	CyberChef was mainly used to decode the word documents from Base64 to allow the true nature of the documents to be revealed	Used extensively throughout Capture 1 with every word document found
Ngrep	A packet analyser that allowed for the searching of packets for keywords	Ngrep was mainly used to search captures for the appearance of keywords that related to the case. This allowed for the discovery of further suspicious packets related to suspects and details relating to them	Used within Capture 3 to search the capture for suspect names and keywords
CSV to KML converter	Simple web application for converting CVS to KML	Utilised to convert a created CSV to a KML file supported by Google Earth online	Used within Capture 3 for converting the CSV file containing the location data

Google Earth Online	Web-based Google Earth allows for the plotting of points and exploration of Earth	It was used to import a KML file with locational data to plot the series of points. This allowed for the uncovering of hidden data that was revealed by the pattern of all the plotted points.	This was only used for the locational data that was found within Capture 3.
Python & VSC	Visual Studio Code is an IDE, and Python is a programming language	VSC and Python were used to create a script to strip the excess data from the location requests extracted. This reduced the time taken by the investigator during the investigation when handling the data	These were exclusively used when handing the location data after it had been extracted and before it was converted to a KML file

4 PACKET CAPTURE 1

4.1 PACKET CAPTURE ANALYSIS

Following the methodology set out above, the investigator conducted Statistical flow analysis on the capture to establish key addresses and protocols, which can be viewed in Appendix A1. This provided a good starting point and allowed subsequent filters to be applied to the capture within Wireshark. Two addresses were found to be receiving many packets, and the investigator filtered the capture using these addresses to investigate the traffic further. This was key to establishing the method of file download mentioned within the intel for the capture, as this established potential suspects. In addition, SMB requests were found within the filtered packets, which was further investigated as the potential transfer method. SMB allows users to communicate with other systems and access files, printers and serial ports within the same network and over the internet (Andriekutė, 2021).

Using this information, the capture was further filtered for SMB packets to exclude the suspected irrelevant data from the investigator and to isolate the suspected target packets. After filtering, the investigator could investigate suspicious packets that included the two suspect addresses that had been identified. Investigating the packets allowed the investigator to observe and note down the username, IP address and MAC address of the suspect's machine and the host machine, which can be seen in Figure 1. In addition, the host username was seen in "TREE Connect" requests, and some directory names were also visible.

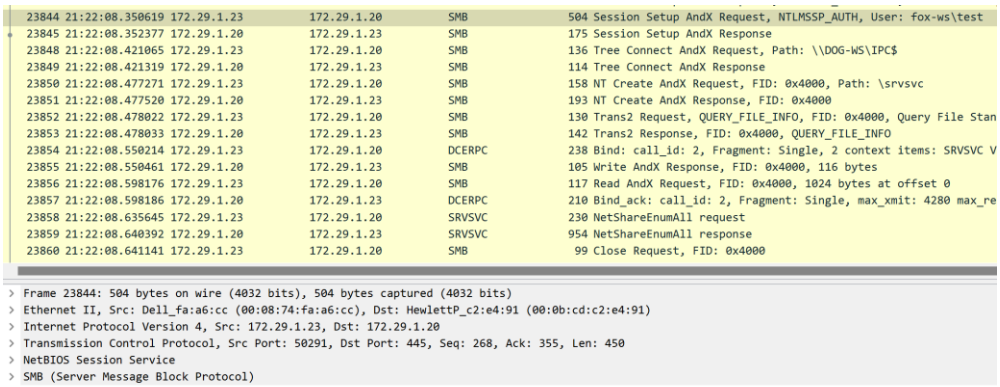


Figure 1: Suspect and Host information

Within further packets, a complete list of files accessible to the suspect was viewable. Packet 23917 contained a "FIND_FIRST2" response containing the list of files, which confirmed to the investigator that SMB had been used to transfer files that could be evidence. Through the export objects function within Wireshark, the investigator could quickly and efficiently extract the SMB objects that had been transferred. The investigator could then view them, and it was noted that the suspect did not download every file accessible to them. The host's computer username was confirmed as it was seen during the exportation process, which can be seen in Figure 2.

23854	\\DOG-WS\IPC$	PIPE (Not Implemented) (0/0) W [0.00%]	0 bytes	\srvsvc
23902	\\DOG-WS\DOCUMENTS	FILE (129/129) R [100.00%]	129 bytes	\desktop.ini
23924	\\DOG-WS\DOCUMENTS	FILE (151/151) R [100.00%]	151 bytes	\My Music\desktop.ini
23932	\\DOG-WS\DOCUMENTS	FILE (150/150) R [100.00%]	150 bytes	\My Pictures\desktop.ini
23940	\\DOG-WS\DOCUMENTS	FILE (151/151) R [100.00%]	151 bytes	\My Videos\desktop.ini
24021	\\DOG-WS\DOCUMENTS	FILE (42/42) R [100.00%]	42 bytes	\My Pictures\Sample Pictures\desktop.ini
24186	\\DOG-WS\BLAH	FILE (1324022/1324022) W [100.00%]	1324 kB	\Documents.zip
25755	\\DOG-WS\BLAH	FILE (1014/1324022) R [0.00%]	1324 kB	\DOCUME~1.ZIP

Figure 2: List of files exported through Wireshark

Nine files were discovered by the investigator to have been downloaded; five ".ini" files were downloaded, three ".zip" files and a file called "srvsvc". The final file, "srvsvc", was investigated further and related to the inter-process communication share ($IPC), where anonymous users can perform limited activities, including the enumeration of domain accounts and shares (*IPC$ share*, 2021).

After the file recovery, each ".ini" file was examined, and all five were found to be irrelevant to the investigation. The three zip files had their hashes calculated to ensure their forensic integrity throughout the investigation, which can be seen in Appendix A3. This also allowed the investigator to compare the three zip hashes and found that each file was different. This was confirmed when the files all responded differently to the extraction attempts.

Each file was then decompressed using WinRAR (GmbH, no date) by the investigator, and only the file "Documents.zip" was successfully uncompressed; the files can be seen in Appendix A2.

4.2 Recovered File Analysis

Due to only one of the extracted zip files being useable, the "Documents.zip" folder was extracted into a read-only folder to be further investigated whilst not altering any data. Four folders and an additional zip file were discovered within the extracted folder. The investigator first examined the new zip file named "untitled folder.zip" for any evidence. The file contained a sequence of empty folders ending with a folder name "SilentEye". The investigator researched the name and discovered an application that shared the name used for steganography to hide messages in pictures and audio (*SilentEye - Steganography is yours*, no date). This was seen as a tip-off for potential usages of the application to hide messages within the capture.

The investigator then examined the four folders within "Documents.zip", and each folder was examined for evidence following the methodology provided. Several file types were found within the four folders, and the investigator examined each file individually.

Images found were scanned using binwalk for hidden files concerning the "SilentEye" clue found (*binwalk | Kali Linux Tools*, no date). The file "NorthKorea.jpg" contained a python script called "broken.py"; the script contained syntax errors and was non-functional in its current state. The steganography presented a challenge to the investigator initially, but the functionality of binwalk allowed the hidden file to be quickly found and extracted.

Many word documents were extracted, and upon investigating them all, it was seen that everyone was encoded. This proved challenging for the investigator as the information contained was unusable in this state. The investigator utilised CyberChef to help decode each of the files (*CyberChef*, no date). The file's contents were copied to CyberChef, where it detected the encoding being used as Base64 (Kumar, 2021, p. 64). This allowed the information within the files to be revealed to the investigator where one file's content did not match its title. Once decoded, the file "track6" contained a list of potential suspect usernames. This discovery and the apparent attempt to hide the data confirmed to the investigator that the previously identified IP addresses and computer usernames were related to the suspects. All of the recovered files can be viewed in Appendix A4 & A5.

5 PACKET CAPTURE 2

5.1 PACKET CAPTURE ANALYSIS

Statistical flow analysis was skipped as the first step for capture two due to the provided intel providing a solid starting point. Intel, on the capture, detailed the suspected FTP usage between a corrupted official and a foreign national.

The intel was used to initially filter the capture for FTP data, where the download of two files (sandofwhich.zip & ojd34.zip) was noted. After switching the filter to "FTP-DATA", the files could easily be retrieved by the investigator utilising the "Follow TCP Stream" function within Wireshark and saving the data stream as a zip file. Wireshark provided a quick way for the investigator to extract the downloaded files so they could be viewed.

Within the FTP requests, a username of "Ill_Song" was noted as a point of interest, along with the suspect's IP address. After searching packet details using the search function within Wireshark for "Song", an email request was seen. After further investigation utilising the "Follow TCP Stream" function again, a data stream was extracted containing three zip files. Finally, the extracted data stream was opened in the HxD hex editor to allow the carving of the three discovered zips. The five zips can be seen in Appendix B1. The discovery of Ill Song's username allowed the investigator to quickly isolate any files that could have been linked to the initial download and removed the challenge of sifting through the capture for related packets.

Each newly extracted file contained several image files, some being viewable and others not viewable. It was clear to the investigator that the original images had been fragmented to prevent them from being viewed, as all of the image files had one-word names, and only three parts of the pictures were viewable. Referring back to the intel provided on this capture assisted with the usage of an Edward Snowden quote is referred to. The quote was identified from the Guardian article, where Snowden reveals that he was the NSA whistle-blower (Greenwald, MacAskill and Poitras, 2013).

5.2 IMAGE RECONSTRUCTION

With the files all copied to the same directory, the investigator pieced the image together using the "cat" command within Linux following the identified Snowden quote. The successful process revealed an image of a chess board seen in Figure 3. The investigator deemed this to be the suspected item received by the corrupt official.

Figure 3: Reconstructed image using the Snowden Quote

The remaining image files were pieced back together using trial and error until two other images were created, which can be viewed in Appendix B2. Piecing the image back together was made easier through the intel provided in the Snowden quote. Without this tip-off or quote, constructing the image would have been made much harder due to the apparent attempt to hide the image's actual value and the large number of images found within the capture used to presumably reduce that chance of successfully recreating the image. These factors would have made brute forcing the image a long and extensive task.

6 PACKET CAPTURE 3

6.1 PACKET ANALYSIS

With the suspect names provided within intel for this capture, the capture was searched using ngrep and Wireshark (*Wireshark · Go Deep.*, no date) for these values. This allowed the investigator to find any evidence that linked to the suspects quickly and provided a solid starting point.

Multiple packets matched the search criteria, and each packet contained JSON data that looked to relate to the sending and receiving of messages. The request URI for the packets was seen to include "http://api.pinger.com/", and upon further research, this related to a company that created the messaging app "TextFree" (*Free Texting and Calling | TextFree*, no date). This was confirmed by investigating the request packet for one of the packets containing "Ill-Song", where both the application "TextFree" was mentioned and the device being used, which was a Nexus 7 running android 4.2.2. The packet capture was then filtered to show only the packets containing message JSON information, and the data was then exported using Wireshark to export HTTP objects.

6.2 TEXT ANALYSIS

The JSON data from the messaging app requests were then viewed using an online JSON viewer, and the text messages were pieced together. A transcript of the conversation and a complete list of the JSON data can be seen in Appendix C1. The JSON data provided crucial evidence on the content of the texts, along with the senders and timestamps of the messages. Timestamps within the data allowed for the conversation to be pieced back together effectively by the investigator, and the usage of the online viewer made it easier for critical information to be extracted from the data.

6.3 LOCATION DATA ANALYSIS

Multiple requests from the same IP address were found after searching the capture for "location". This search term was used to uncover evidence on a potential meeting point. Requests were made to the address "mob.mapquestapi.com", and this find was key to narrowing down these suspect requests. Within this request, the investigator discovered data that looked like coordinates, so the address was used to filter the capture to allow for the HTTP objects to be extracted. This allowed for the key requests containing location data to be exported to a CSV file. The investigator created a python script to quickly strip out the excess data within the requests and leave only the longitude and latitude. The CSV file was then converted to a KML file to be imported into Google Earth to view all of the plotted points. This allowed the plotted points to be viewed where they revealed a "17", which was believed to be the date relating to the meeting; this can be seen in Appendix C2.

7 General Evaluation and Reflection

The investigator utilised many tools and techniques they were familiar with throughout the investigation to uncover critical evidence. Although more efficient tools may have revealed results quicker, the use of familiar tools allowed the investigator to uncover evidence that may have been missed when using a tool they were unfamiliar with. Multiple attempts to conceal evidence were noted throughout the investigation, presenting many challenges for the investigator in uncovering details of the case. The usage of base64 encoding, zip files being concealed and needing manually carved and hiding information through google map plot points all presented challenges for the investigator that ultimately slowed the investigation but did not prevent the recovery of the information.

REFERENCES

Andriekutė, A. (2021) *What is SMB and how does it work? | NordVPN*. Available at: https://nordvpn.com/blog/what-is-smb/ (Accessed: 4 December 2022).

binwalk | Kali Linux Tools (no date) *Kali Linux*. Available at: https://www.kali.org/tools/binwalk/ (Accessed: 13 December 2022).

CyberChef (no date). Available at: https://gchq.github.io/CyberChef/ (Accessed: 13 December 2022).

Free Texting and Calling | TextFree (no date). Available at: https://textfree.us/ (Accessed: 13 December 2022).

GmbH, win rar (no date) *WinRAR*, *WinRAR download free and support*. Available at: https://www.win-rar.com/start.html?&L=0 (Accessed: 13 December 2022).

Greenwald, G., MacAskill, E. and Poitras, L. (2013) 'Edward Snowden: the whistleblower behind the NSA surveillance revelations', *The Guardian*, 11 June. Available at: https://www.theguardian.com/world/2013/jun/09/edward-snowden-nsa-whistleblower-surveillance (Accessed: 9 December 2022).

IPC$ share (2021). Available at: https://learn.microsoft.com/en-us/troubleshoot/windows-server/networking/inter-process-communication-share-null-session (Accessed: 4 December 2022).

Kumar, A. (2021) *What Is Base64 Encoding?*, *Medium*. Available at: https://levelup.gitconnected.com/what-is-base64-encoding-4b5ed1eb58a4 (Accessed: 13 December 2022).

SilentEye - Steganography is yours (no date). Available at: https://achorein.github.io/silenteye/ (Accessed: 13 December 2022).

Wireshark · Go Deep. (no date). Available at: https://www.wireshark.org/ (Accessed: 13 December 2022).

APPENDICES

APPENDIX A - CAPTURE 1 ARTIFACTS

A1 - STATISTICAL FLOW ANALYSIS

Below are the outputs from the carried-out statistical flow analysis. This look at the most used ports based on packets along with specific port investigations.

```
┌──(kali㉿kali)-[~/Desktop/Unit 2 - Case Study - PCAP Files]
└─$ rwstats Capture1.rw --fields=4 --values=packets --count=10
INPUT: 1368 Records for 674 Bins and 30295 Total Packets
OUTPUT: Top 10 Bins by Packets
```

dPort	Packets	%Packets	cumul_%
80	12448	41.089289	41.089289
50180	3547	11.708203	52.797491
443	2431	8.024426	60.821918
445	1336	4.409969	65.231886
50291	942	3.109424	68.341310
1784	460	1.518402	69.859713
1633	224	0.739396	70.599109
1696	224	0.739396	71.338505
1668	224	0.739396	72.077901
1631	224	0.739396	72.817297

```
┌──(kali㉿kali)-[~/Desktop/Unit 2 - Case Study - PCAP Files]
└─$ rwstats Capture1.rw --fields=3 --values=packets --count=10
INPUT: 1368 Records for 686 Bins and 30295 Total Packets
OUTPUT: Top 10 Bins by Packets
```

sPort	Packets	%Packets	cumul_%
80	10051	33.177092	33.177092
50180	6593	21.762667	54.939759
443	2395	7.905595	62.845354
50291	1336	4.409969	67.255323
445	942	3.109424	70.364747
1784	308	1.016669	71.381416
1315	204	0.673378	72.054795
50039	182	0.600759	72.655554
1769	181	0.597458	73.253012
1696	147	0.485229	73.738241

```
┌──(kali㉿kali)-[~/Desktop/Unit 2 - Case Study - PCAP Files]
└─$ rwfilter Capture1.rw --sport=80 --pass=stdout | rwstats --fields=dip --percentage=0.5 --bytes
INPUT: 418 Records for 3 Bins and 6786279 Total Bytes
OUTPUT: Top 2 bins by Bytes (0.5000% == 33931)
```

dIP	Bytes	%Bytes	cumul_%
172.29.1.20	5743361	84.631961	84.631961
172.29.1.23	1029088	15.164245	99.796206

```
┌──(kali㉿kali)-[~/Desktop/Unit 2 - Case Study - PCAP Files]
└─$ rwfilter Capture1.rw --sport=443 --pass=stdout | rwstats --fields=dip --percentage=0.5 --bytes
INPUT: 114 Records for 2 Bins and 1998675 Total Bytes
OUTPUT: Top 2 bins by Bytes (0.5000% == 9993)
```

dIP	Bytes	%Bytes	cumul_%
172.29.1.23	1143086	57.192190	57.192190
172.29.1.20	855589	42.807810	100.000000

```
┌──(kali㉿kali)-[~/Desktop/Unit 2 - Case Study - PCAP Files]
└─$ rwfilter Capture1.rw --sport=443 --pass=stdout | rwstats --fields=sip --percentage=0.5 --bytes
INPUT: 114 Records for 48 Bins and 1998675 Total Bytes
OUTPUT: Top 27 bins by Bytes (0.5000% == 9993)
```

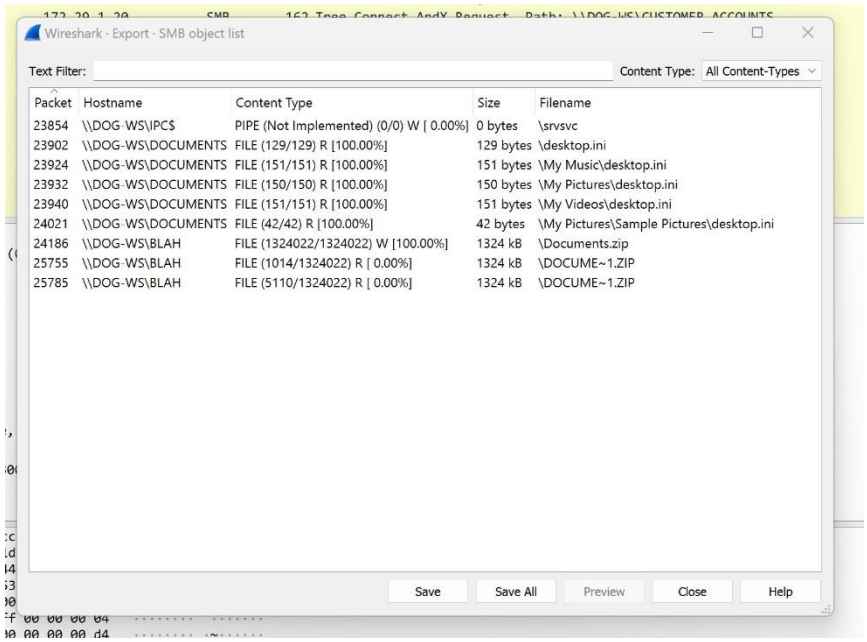
sIP	Bytes	%Bytes	cumul_%
171.161.199.100	309414	15.480956	15.480956
74.125.239.60	308838	15.452137	30.933093
173.194.79.103	300693	15.044617	45.977710
74.125.239.50	278436	13.931029	59.908740
69.172.216.55	138768	6.943000	66.851739
74.125.224.47	58694	2.936646	69.788385
69.172.216.56	52003	2.601874	72.390259
74.125.239.59	50838	2.543585	74.933844
74.125.224.156	45810	2.292018	77.225862
134.170.21.245	39667	1.984665	79.210527
74.125.239.153	36504	1.826410	81.036937
74.125.239.156	35980	1.800193	82.837130
74.125.224.187	30504	1.526211	84.363341
23.23.130.131	25375	1.269591	85.632932
69.172.216.111	23062	1.153864	86.786796
184.25.119.231	19983	0.999812	87.786609
23.40.196.221	19326	0.966941	88.753549
66.117.23.102	16832	0.842158	89.595707
157.56.67.167	15648	0.782919	90.378626
207.200.74.51	13520	0.676448	91.055074
8.21.24.36	13350	0.667943	91.723016
74.201.120.33	11935	0.597146	92.320162
206.190.60.139	11472	0.573980	92.894142
65.171.135.52	11251	0.562923	93.457065
75.101.153.118	10771	0.538907	93.995972
54.241.7.243	10586	0.529651	94.525623
204.236.227.135	10033	0.501983	95.027606

```
┌──(kali㉿kali)-[~/Desktop/Unit 2 - Case Study - PCAP Files]
└─$ rwfilter Capture1.rw --sport=80 --pass=stdout | rwstats --fields=sip --percentage=0.5 --bytes
INPUT: 418 Records for 100 Bins and 6786279 Total Bytes
OUTPUT: Top 15 bins by Bytes (0.5000% == 33931)
```

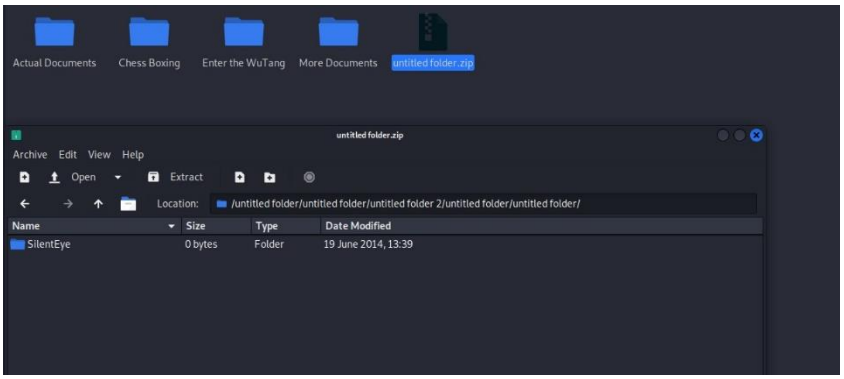
sIP	Bytes	%Bytes	cumul_%
93.184.215.248	3713668	54.723185	54.723185
184.28.16.25	659370	9.716223	64.439408
23.216.11.91	360167	5.307283	69.746690
23.212.52.51	247748	3.650719	73.397410
23.212.52.49	245350	3.615383	77.012793
63.141.196.249	214685	3.163516	80.176309
64.12.132.55	182423	2.688115	82.864424
23.216.11.113	140947	2.076941	84.941365
23.212.52.56	102991	1.517636	86.459001
69.172.216.55	87582	1.290575	87.749575
23.212.52.42	84568	1.246162	88.995737
207.200.111.13	81051	1.194336	90.190073
152.163.13.68	72119	1.062718	91.252791
23.212.52.50	62635	0.922965	92.175756
205.188.16.197	60753	0.895233	93.070989

A2 - SMB Recovered File Evidence

Below is the list of recovered evidence that was found within the SMB files recovered.



Files containing evidence, including the list of usernames that were identified as the target evidence, can be seen below. The folder clue relating to SilentEye can also be seen within this collection.



Output

```
The Mystery of Chess Boxing:
(usernames)

Mr. Method

Kim Ill-Song

Mr. Razor

Mr. Genius

Mr. G. Killah

Matt Cassel

Mr. I. Deck

Mr. M Killa

Mr. O.D.B.

Mr. Raekwon

Mr. U-God

Mr. Cappadonna (possibly)

John Woo?

Mr. Nas
```

```
┌──(kali㉿kali)-[~/Desktop/Doc 1/Documents/More Documents]
└─$ binwalk -b NorthKorea.jpeg

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, JFIF standard 1.01
3453          0xD7D           Zip archive data, at least v2.0 to extract, name: untitled/
3492          0xDA4           Zip archive data, at least v2.0 to extract, compressed size: 604, uncompressed size: 1397, name: untitled/broken.py
4263          0x10A7          End of Zip archive, footer length: 22
```

Below is the "broken.py" file that was found within "NorthKorea.jpg" and can be seen above.

```python
def fileToString(pathToFile):
        f = open(pathToFile, "r")
        strs = ""
        #adds each line of the file to the strs string
        for line in f.readlines():
                strs+=line
        return strs
def ASCII():
        #number of ASCII characters
        NumOfASCII = 0
        #returns list of all ASCII characters
        return "".join([chr(i) for i in range(NumOfASCII)])
def sumName(name):
        sums=0
        #sums the indices in ASCII of all the characters in name
        for x in name:
                sums+=ord(x
        return sums
def indexInFile(password):
        indices = []
        ASCIIArray = ASCII()
        #populates an array of indices to be used by the encoder
        for chrs in password:
                indices.append(ASCIIArray.index(chrs)+sumName(name)*2
        return indices
def indexInASCII(name):
        indices = []
        ASCIIArray = ASCII()
        #split on all non-numeric characters
        #remove first index because it is blank
        indexList = re.split("[^\d]",encoded)[1:]
        #converts encoded characters to ASCII
        for index in indexList:
                indices.append(ASCIIArray[int(index) - (sumName(name)*2)])
        #returns decoded message
        return "".join(indices)
def encode(name):
        #returns a list of indices to be used for encoding
        indices = indexInFile(password,name)
        #convert file associated with name to a string
        bill = fileToString("./%s.txt"%name)
        encoded = ""
        #add letter in file plus index of the letter in the file to the encoded string
        for index in indices:
                encoded+=bill[index]+str(index)

        return encoded
```

A3 – ZIP FOLDER HASH VALUES

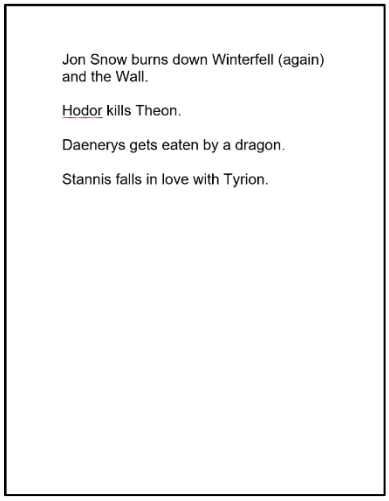
The hashes for the three zip files can be seen within the image below. This was used to ensure forensic validity and proved to the investigator that the files were individually different.

```
┌──(kali㉿kali)-[~/Desktop/Capture 1 SMB]
└─$ sha256sum %5cDocuments.zip
afa54cb6c0b3f2d606a9a0e9e424d4af1b4d0f03b8e5338e95acb6ec495772eb  %5cDocuments.zip

┌──(kali㉿kali)-[~/Desktop/Capture 1 SMB]
└─$ sha256sum %5cDOCUME~1\(1\).ZIP
2b7caaa212c99bb1c2ec2ceeb7c444e0c692be1683c6e22fae70d50d50ca9aab  %5cDOCUME~1(1).ZIP

┌──(kali㉿kali)-[~/Desktop/Capture 1 SMB]
└─$ sha256sum %5cDOCUME~1.ZIP
a5f11d6c4f09ce88b6d6b57752fa41b3a640649b0d31de4f8e74242e5a585e2b  %5cDOCUME~1.ZIP
```

A4 – Recovered Excess Individual Files

Below are encoded and decoded versions of each word document found within capture 1. This highlights the usage of anti-forensic techniques to conceal the evidence from investigators.

Sm9uIFNub3cgYnVybnMgZG93biBXaW
50ZXJmZWxsIChhZ2FpbikgYW5klHRo
ZSBXYWxsLg0KDQplb2RvciBraWxscy
BUaGVvbi4NCg0KRGFlbmVyeXMgZ2V
0cyBlYXRlbiBieSBhIGRyYWdvbi4NCg0
KU3Rhbm5pcyBmYWxscyBpbiBsb3ZlIH
dpdGggVHlyYW9uLiANCg0KDQo=



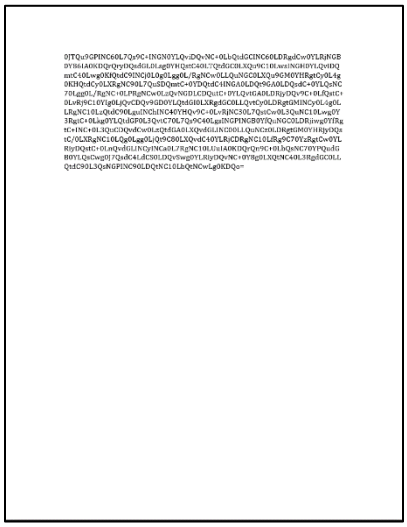
Jon Snow burns down Winterfell (again) and the Wall.

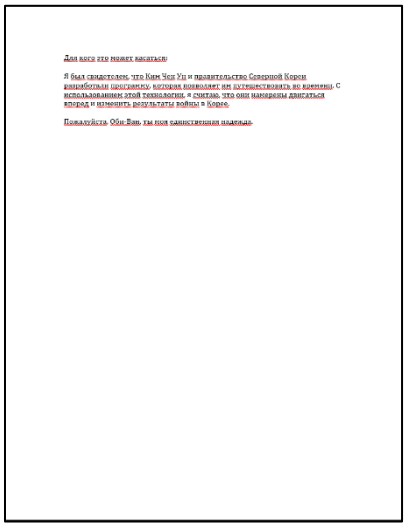
Hodor kills Theon.

Daenerys gets eaten by a dragon.

Stannis falls in love with Tyrion.

GoT Spoilers.docx





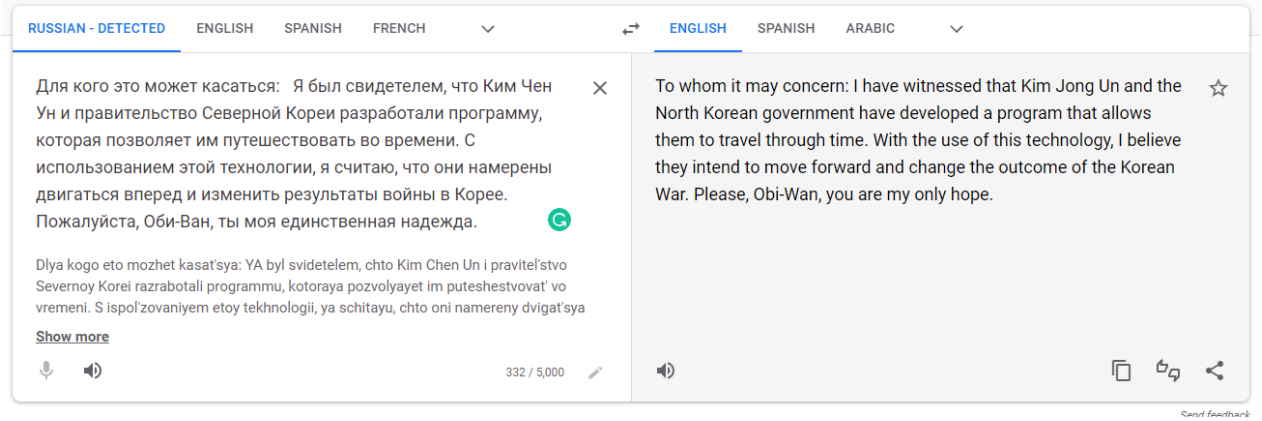
Для кого это может касаться:

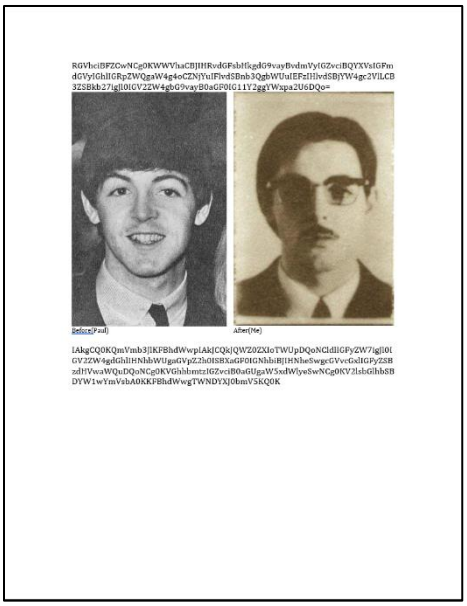
Я был свидетелем, что Ким Чен Ун и правительство Северной Кореи разработали программу, которая позволяет им путешествовать во времени. С использованием этой технологии, я считаю, что они намерены двигаться вперед и изменить результаты войны в Корее.

Пожалуйста, Оби-Ван, ты моя единственная надежда.

NorthKorea.docx

Decoded and then translated from Russian





RGVhciBFZCwNCg0KWWVhaCBjIHRvdGFsbHkgdG9vayBvdmVyIGZvciBQYXVsIGPm
dGVyIGhlIGRpZWQgaW4g4oCZNjYulFlvdSBnb3QgbWUulEFzIHlvdSBjYW4gc2VlLCB
3ZSBkb27lgIl0lGV2ZW4gb G9vayB0aGF0IG11Y2ggtWxpa2U6DQo=

Before(Paul) After(Me)

IAkgCQ0KQmVmb3JlKFBhdWwpIAkjCQkjQWZ0ZXIoTWUpDQoNCldlIGFyZW7igll0t
GV2ZW4gdGhlIHNhbWUgaGVpZ2h0ISBXaGF0IGNhbiBjIHNhaSwgcGVvcGxlIGFyZSB
zdHVwaWQuDQoNCg0KVGhlbhmtzIGZvciB0aGUgaW5xdWlyeSwNCg0KV2lsbGlhbSB
DYW1wYmVsbA0KKFBhdWwgTWNDYXJ0bmV5KQ0K

Before(Paul) After(Me)

Dear Ed,

Yeah I totally took over for Paul after he died in '66. You got me. As you can see, we don't even look that much alike:

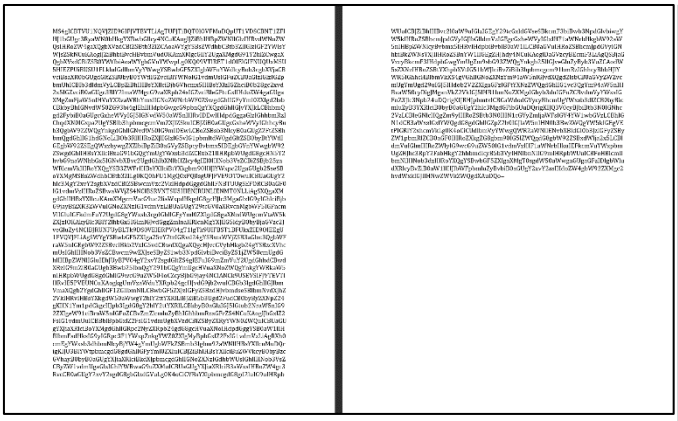
Before(Paul) After(Me)

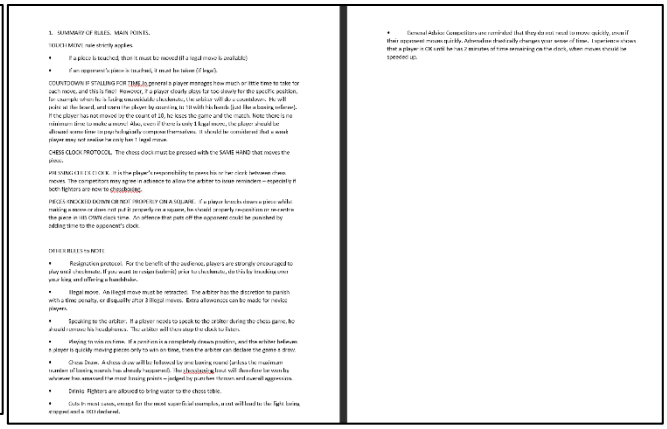
We aren't even the same height! What can I say, people are stupid.

Thanks for the inquiry,

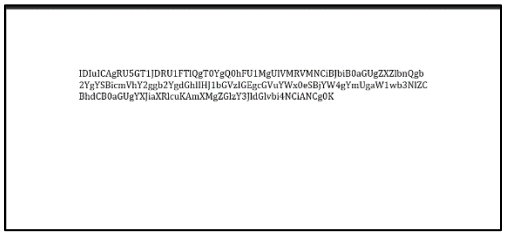
William Campbell
(Paul McCartney)

PiD.docx

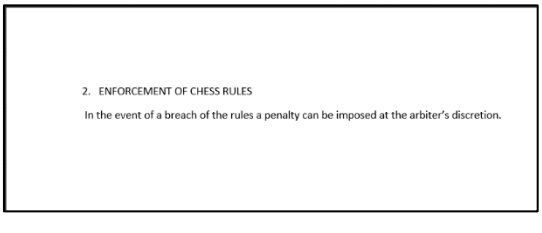


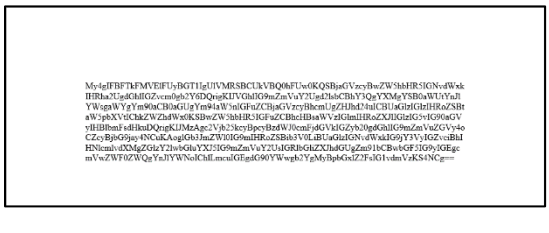


Rules 1.docx

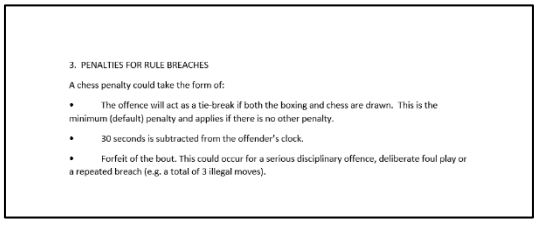


Rules 2.docx





Rules 3.docx



NC4gICBDSEVTUyBDTE9DSyBNQUxGVU5DVElPTg0KSW4gdGhlIHVubGlrZWx5IGV
2ZW50IHRoZSBlbGVjdHJvbmljIGNoZXNzIGNsb2NrIGNlYXNlcyB0byBvcGVyYXRlIGR
1cmluZyBhIGNoZXNzIHJvdW5kLCB0aGUgYXJiaXRlciB3aWxsIGRvIG9uZSBvZiBmb2
xsb3dpbmcsIGRlcGVuZGluZyBvbiB0aGUgZXN0aW1hdGVkIGRpc3J1cHRpb24gdG8g
dGhlIHBsYXllcnMgYW5kIHNwZWN0YXRvcnM6DQrigKIJU3RvcCB0aGUgY2xvY2sgY
W5kIHJlc29sdmUgdGhlIHByb2JsZW0uDQrigKIJU3RvcCB0aGUgY2xvY2sgYW5kIHJlc
GxhY2UgaXQgd2l0aCBhIG5ldyBjbG9jay4gIFRoaXMgYWN0aW9uIGlzIG1vc3QgbGlrZ
Wx5IGlmIHRoZXJlIGlzIGEgcmVwZWF0ZWQgbWFsZnVuY3Rpb24sIG9yIGl04oCZcyB
vbmUgb2YgdGhlIGxhdGVyIGNoZXNzIHJvdW5kcyB3aGVyZSBhIHBsYXllciBpcyBzaG9
ydCBvZiB0aW1lLg0KDQoNCg==

4. CHESS CLOCK MALFUNCTION

In the unlikely event the electronic chess clock ceases to operate during a chess round, the arbiter will do one of following, depending on the estimated disruption to the players and spectators:

- Stop the clock and resolve the problem.
- Stop the clock and replace it with a new clock. This action is most likely if there is a repeated malfunction, or it's one of the later chess rounds where a player is short of time.

Rules 4.docx

Rules 5.docx

5. WCBA CHESS RULES FOR CHESSBOXING

Chess tournament rules have legal points that casual players may be unfamiliar with. The official laws of chess are on the website of FIDE, the chess governing body http://www.fide.com/component/handbook/?id=32andview=category.

Highlighted below are legal points that cause most disputes in tournament chess situations.

In addition, some chessboxing laws differ from FIDE rules in order to (i.) ensure the paying public is entertained, (ii.) keep the game flowing with minimal disruption, and (iii.) minimise verbal communication with the competitors. These differences are highlighted where they occur.

Touch move

- Once a piece is touched it MUST be moved, unless "J'adoube" is indicated before touching the piece. If no legal move is admissible, then any other piece can be moved without punishment.
- Once an opponent's piece is touched it must be captured if there is such a legal move. If it cannot be captured the offender receives no penalty and is free to move without restriction.

Castling touch move

When castling you MUST touch the king first. If you touch the rook first, then you cannot castle, but you must move the rook because of the touch-move rule.

Hand is taken off a piece

When a piece is moved and the hand taken off the piece, the move cannot be retracted – the piece cannot be moved to a different square.

Illegal move

The arbiter will point out the illegal move if it goes unnoticed. Since the punishment for an illegal move is not as severe in chessboxing as in FIDE blitz chess laws, the arbiter will not allow the possibility of an illegal move going uncorrected.

"J'Adoube" rule.

Normal Chess Rules

- If a piece is off centre and is annoying you, state "J'adoube" or "I adjust" BEFORE adjusting its position on the square. One of these phrases should be used regardless of the player's home language.
- If you state "J'adoube" after or during the piece adjustment, then it counts as a touch move.
- You should only adjust pieces whilst your clock is running. Adjusting during your opponent's time is forbidden as it is a distraction.

Chessboxing Rules (adapted because both players have headphones)

- With headphones on it is simplest if players don't try to J'adoube. Pieces will be nicely centred by the arbiter between each chess round. However, if the urge to J'adoube becomes irresistible, follow the below procedure...
- Clearly turn to the arbiter and mouth "J'adoube" AND give the J'adoube hand signal specially developed for chessboxing. Then adjust the piece as in a normal chess game.
- The J'adoube hand signal is the 'OK' hand gesture, creating a circle with the thumb and first finger.

Pawn promotion

A key difference between casual chess and tournament rules. When promoting a pawn to a second queen, do NOT use an upside-down rook (as the electronic chessboard will not recognise it). Even if you shout "queen" as you do so, it is still a rook! The chessboxing arbiter will ensure a spare queen is on the table for you to use.

Clock

- The clock MUST be pressed with the same hand that makes the move
- Running out of time. If a player has no time remaining, then he is lost if his opponent can checkmate him assuming the most unskilled play, otherwise the game is a draw. For example, if Player A has three queens and a king, and Player B has one pawn and a king, then Player B wins if Player A runs out of time.
- A player should not start to make his move until the opponent has physically pressed his clock.
- Time scramble – disputes can arise when 1 or both players are short of time and moving extremely quickly:

o A player should not start to make his move until the opponent has physically pressed his clock. i.e. you should not rush to move a piece in the brief time between your opponent moving his piece and pressing his clock.

o If a player knocks down pieces during a move, he should reset them in his own time before pressing his clock. If he presses his clock without resetting the pieces on their squares, then the opponent can immediately bounce the clock back without making a move, whilst pointing to the offending piece(s) that have been knocked down. The first player should then properly reset the pieces in his own time. [This completely differs from FIDE laws, where the innocent party should stop the clocks and inform the arbiter]. The same action can be performed if a piece is not clearly on a square but significantly overlaps another square such that its position is ambiguous. The arbiter can stop the clocks if there is a flurry of poorly placed pieces, and intervene to reset the board. The arbiter can penalise the offender.

o Drawn position – playing to win on time

- If the arbiter judges the position is a dead draw (e.g. opposite colour bishop ending, or R+K vs R+K), then the arbiter can intervene and declare a draw if a player is simply trying to win on time and not making a concerted effort to win the game. The defender does not need to request the arbiter to make such a judgement; the arbiter will assume the request exists as soon as a player has less than 2 minutes remaining. [This differs from the FIDE laws, which requires the defender to stop the clocks BEFORE he gets into critical time trouble, and ask the arbiter to observe whether the attacker is making a concerted effort to win the game or is just aiming to win on time in a dead drawn position.]
- Losing position – playing to win on time
- Note that if a player is in a winning position but is close to losing on time, the arbiter will not intervene in his favour. If he loses on time before he checkmates the opponent, this is more a consequence of time mismanagement than having to make countless moves shuffling pieces in a dead drawn position.
- Slow playing a lost position – a rule developed for chessboxing to prevent stalling for time.
- If a player takes too much time in a lost position where he would be expected to play much quicker in a normal chess game, the arbiter can give him a count of 10. The arbiter will visually count with his hands. If no move is made on the count of 10, the player forfeits the game.

Draw by threefold repetition

- If the same position occurs 3 times (and with the same player to move), the player can claim a draw ONLY WHEN IT IS HIS MOVE. He should stop the clock after the opponent's last move, remove his headphones and TELL the arbiter what move he WOULD play to get into the 3rd repetition. DO NOT PLAY THE MOVE, DO NOT PRESS THE CLOCK. If the player is unsure how to pause the clock, then he can take off his headphones and claim the draw. The arbiter will stop the clock as the headphones come off. If the draw claim is correct and the claimant runs out of time after removing his headphones, the draw will hold.
- A draw by repetition normally occurs by perpetual check so is easy to identify.

50 move rule

A draw can be claimed if neither a piece is taken nor a pawn moved in 50 moves (i.e. 50 White and 50 Black moves). As players are not writing a game score, the arbiter will monitor on their behalf – this is most likely to occur in an ending B+N+K vs. K.

Draw Offer

- Contrary to FIDE rules, players will not be able to offer a draw unless the position is a 'dead draw', as judged by the arbiter.
- The offer of a draw must be made through the arbiter. Make your move, do not press your clock, and then remove the headphones to speak to the arbiter. The arbiter will stop the clock and judge whether a draw offer is acceptable. If so, he will convey to the opponent for consideration and restart the clock (as the opponent can consider the draw offer until he makes his next move).

Verbal Communication with the arbiter

- If a player wants to speak to the arbiter during the game he should remove his headphones. The arbiter will stop the clock to talk. The other player can remove his headphones to listen to the conversation.

Arbiter's decision

- The arbiter's decision is final. The finer rules of chessboxing will no doubt evolve with the sport. Any unanticipated circumstances will be judged considering the official FIDE chess laws, the need for sporting fair play in relation to the tournament chess experience of the chessboxers, and the need to entertain a paying audience.

Ni4gICBDSEVTUyBEUkFXIElOIFJFTEFUSU9OIFRPIFRIRSBDSEVTU0JPWElORyBCT1
VUDQogDQogSWYgYSBjaGVzcyBkcmF3IGlzIGRlY2xhcmVkIGluIGFueSByb3VuZCwg
dGhlcmUgd2lsbCBiZSBhdCBtb3N0IG9ubHkgb25lIGJveGluZyByb3VuZCB0aGVyZWF
mdGVyLiAgSWYgdGhlIGNoZXNzIGRyYXcgb2NjdXJzIGluIHRoZSBmaW5hbCByb3VuZ
CwgdGhlbiB0aGVyZSB3aWxsIGJlIG5vIGZ1cnRoZXIgYm94aW5nIHJvdW5kLCBpbiBs
aW5lIHdpdGggdGhlIG9yaWdpbmFsIHNjaGVkdWxlLg0KSW4gdGhlIHVubGlrZWx5IG
V2ZW50IHRoYXQgdGhlIGNoZXNzIGdhbWUgaXMgZHJhd24gQU5EIHRoZSBib3hpbm
cgaXMgYSB0aWUgb24gcG9pbnRzLCB0aGVuIHRoZSBwbGF5ZXIgd2l0aCB0aGUgZm
V3ZXN0IGNoZXNzIHBlbmFsdGllcyBpcyB0aGUgd2lubmVyLiBJZiB0aGVzZSBhcmUgZ
XF1YWwgdGhlIGJvdXQgd2lsbCBiZSBkZWNsYXJlZCBhIGRyYXcuDQogDQoNCg==

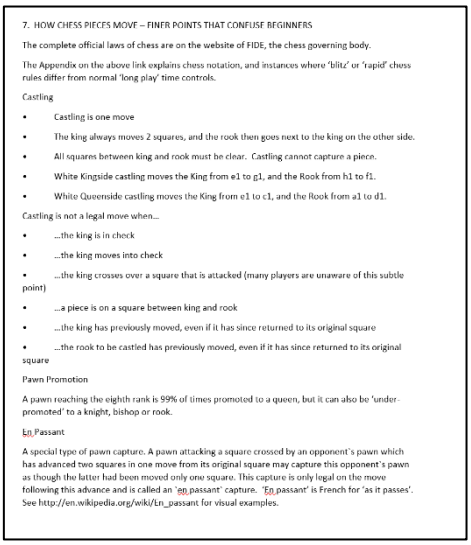
6. CHESS DRAW IN RELATION TO THE CHESSBOXING BOUT

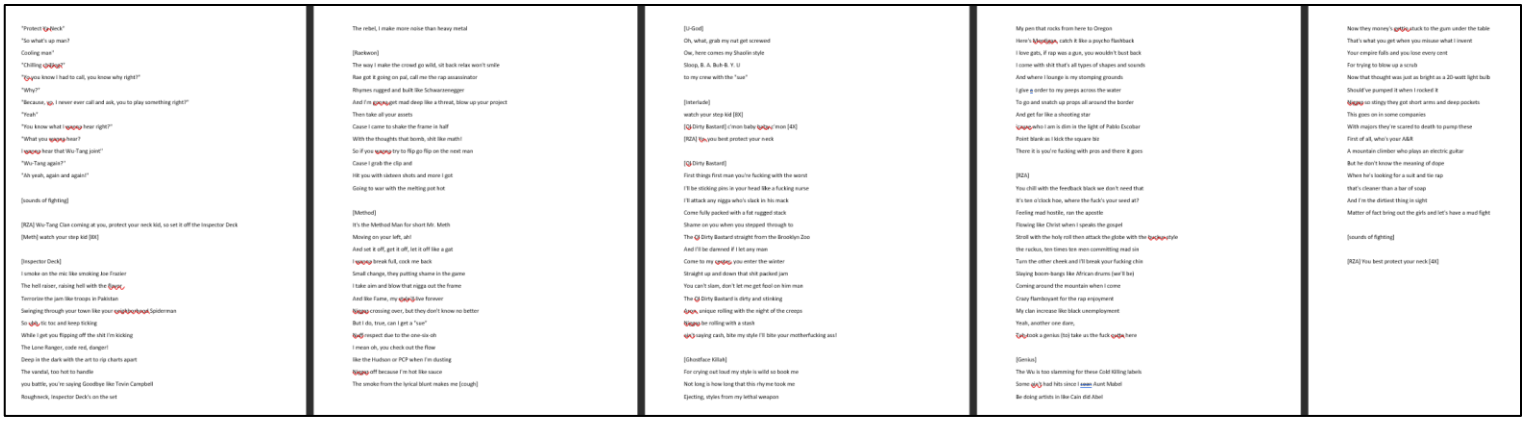
If a chess draw is declared in any round, there will be at most only one boxing round thereafter. If the chess draw occurs in the final round, then there will be no further boxing round, in line with the original schedule.

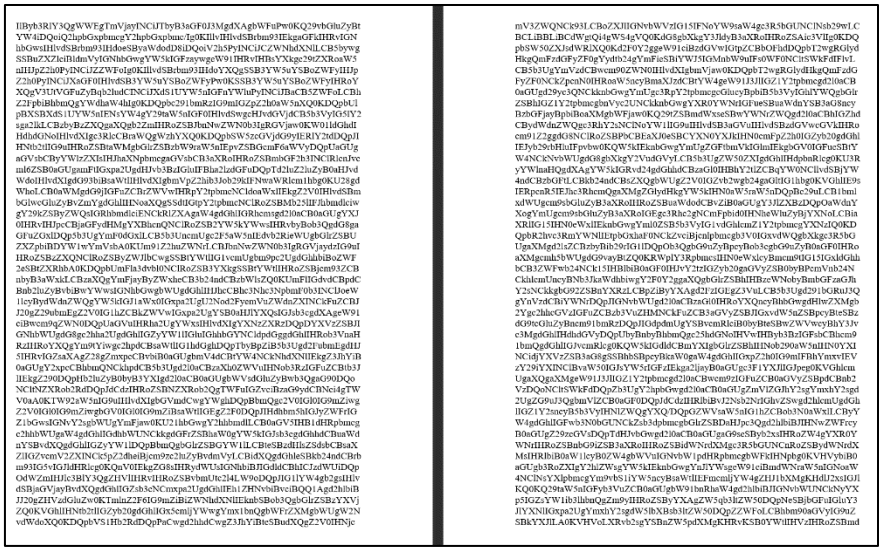
In the unlikely event that the chess game is drawn AND the boxing is a tie on points, then the player with the fewest chess penalties is the winner. If these are equal the bout will be declared a draw.

Rules 6.docx

Rules 7.docx



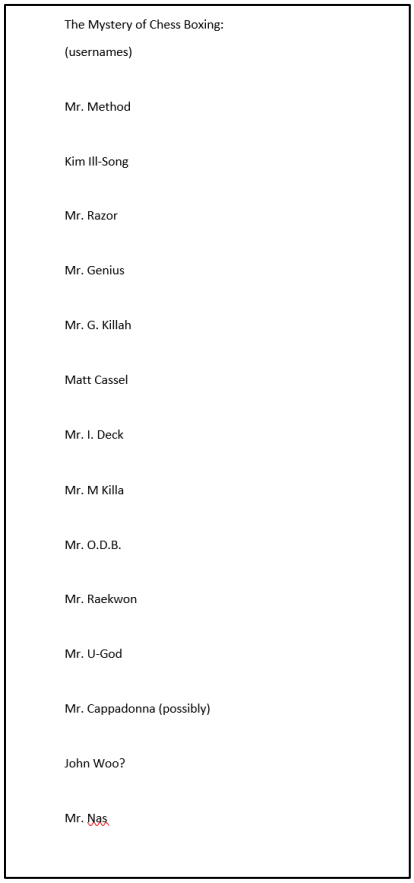




Track10.docx

A5 - Suspected Target Evidence

VGhlIE15c3Rlcnkgb2YgQ2hlc3MgQm94aW5nOg0KKHVzZXJuYW1lcykNCg0KTXIuIE
1ldGhvZA0KDQpLaW0gSWxsLVNvbmcNCg0KTXIuIFJhem9yDQoNCk1yLiBHZW5pd
XMNCg0KTXIuIEcuIEtpbGxhaA0KDQpNYXR0IENhc3NlbA0KDQpNci4gSS4gRGVjaw
0KDQpNci4gTSBLaWxsYQ0KDQpNci4gTy5ELkIuDQoNCk1yLiBSYWVvd29uDQoNCk
1yLiBVLUdvZA0KDQpNci4gQ2FwcGFkb25uYSAocG9zc2libHkpDQoNCkpvaG4gV29
vPw0KDQpNci4gTmFzDQo=



The Mystery of Chess Boxing:

(usernames)

Mr. Method

Kim Ill-Song

Mr. Razor

Mr. Genius

Mr. G. Killah

Matt Cassel

Mr. I. Deck

Mr. M Killa

Mr. O.D.B.

Mr. Raekwon

Mr. U-God

Mr. Cappadonna (possibly)

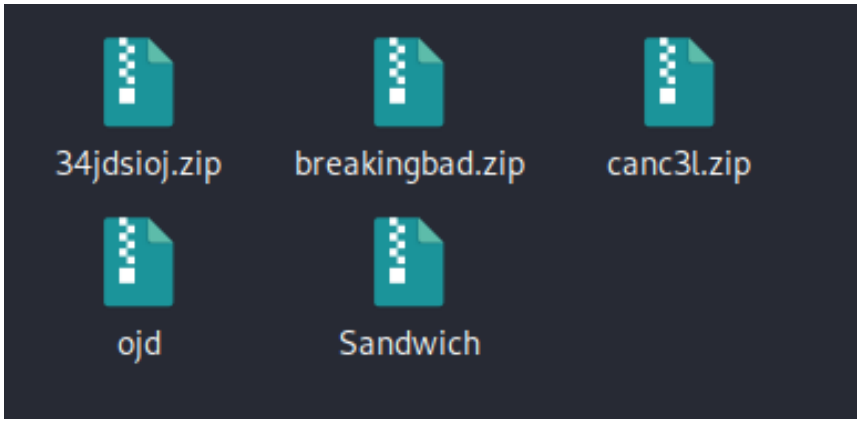
John Woo?

Mr. Nas

Track6.docx

Appendix B – Capture 2 Artifacts

B1 – Recovered Zips



The 5 different zip files that were found within the capture, followed by the hash values for each one. The contents of each folder once extracted can be seen through the use of the tree command

```
┌──(kali㉿kali)-[~/Desktop/Capture 2 Zips]
└─$ sha256sum 34jdsioj.zip
98a0af05448513bbd602b077c32b3af0389efda724b37b702279aadd7ee5caa8  34jdsioj.zip

┌──(kali㉿kali)-[~/Desktop/Capture 2 Zips]
└─$ sha256sum breakingbad.zip
24e12f53d88a9dcbaef60a5c182de6491ff48562555273bc5183a6240781e76f  breakingbad.zip

┌──(kali㉿kali)-[~/Desktop/Capture 2 Zips]
└─$ sha256sum canc3l.zip
e53bbc0db3f490a307cd936ca5ab9512c911bed4c84c88a59a76b5c913063b98  canc3l.zip

┌──(kali㉿kali)-[~/Desktop/Capture 2 Zips]
└─$ sha256sum ojd
6dbf0194a952678813affeac112fe45840a37e8254fad4ecca2f9cb0201fe96f  ojd

┌──(kali㉿kali)-[~/Desktop/Capture 2 Zips]
└─$ sha256sum Sandwich
4c8873d7adf85608c19bb05c2b84b37496b7588818c758d2ecee36df1a3fb52b  Sandwich
```

```
root@kali:~/Desktop/Capture2Extracted# tree
.
├── 34jdsioj
│   ├── corrupt.jpg
│   ├── doors.jpg
│   ├── human.jpg
│   ├── liberties.jpg
│   ├── machine.jpg
│   ├── massive.jpg
│   ├── the.jpg
│   ├── theyre.jpg
│   ├── this.jpg
│   ├── with.jpg
│   └── world.jpg
├── breaking_bad_season_6
│   ├── a.jpg
│   ├── because.jpg
│   ├── but.jpg
│   ├── communism.jpg
│   ├── it.jpg
│   ├── nor.jpg
│   ├── secretive.jpg
│   ├── secret.jpg
│   ├── their.jpg
│   ├── there.jpg
│   └── unconstitutional.jpg
├── canc3l
│   ├── American.jpg
│   ├── behind.jpg
│   ├── closed.jpg
│   ├── condone.jpg
│   ├── constructing.jpg
│   ├── internet.jpg
│   ├── people.jpg
│   ├── privacy.jpg
│   ├── secretly.jpg
│   ├── surveillance.jpg
│   ├── to.jpg
│   └── U.S..jpg
├── ojd34
│   ├── allow.jpg
│   ├── and.jpg
│   ├── around.jpg
│   ├── basic.jpg
│   ├── building.jpg
│   ├── cant.jpg
│   ├── conscience.jpg
│   ├── terrorism.jpg
│   ├── Watergate.jpg
│   └── web-based.jpg
└── sandofwhich
    ├── destroy.jpg
    ├── for.jpg
    ├── freedom.jpg
    ├── good.jpg
    ├── government.jpg
    ├── I.jpg
    ├── in.jpg
    ├── NSA.jpg
    ├── rights.jpg
    └── security.jpg
```

B2 – IMAGE RECONSTRUCTION

Below is the “cat” command used to reconstruct the image that was created using the Edward Snowden quote. This chess board is believed to be the item that was exchanged.

```
└─$ cat I.jpg cant.jpg in.jpg good.jpg conscience.jpg allow.jpg the.jpg U.S..jpg government.jpg to.jpg destroy.
jpg privacy.jpg internet.jpg freedom.jpg and.jpg basic.jpg liberties.jpg for.jpg people.jpg around.jpg world.jp
g with.jpg this.jpg massive.jpg surveillance.jpg machine.jpg theyre.jpg secretly.jpg building.jpg > Snowden.jpg
```

Below are the two other images that were pieced back together using the left-over image files along with the commands used to do so.

```
┌──(kali㉿kali)-[~/Desktop/Capture 2 Zips/Complete Photo Collection]
└─$ cat condone.jpg American.jpg web-based.jpg rights.jpg constructing.jpg security.jpg terrorism.jpg NSA.jpg Watergate.jpg corrupt.jpg human.jpg behind.jpg closed.jpg doo
rs.jpg > OtherImage2
```

```
┌──(kali㉿kali)-[~/Desktop/Capture 2 Zips/Complete Photo Collection]
└─$ cat there.jpg their.jpg a.jpg it.jpg but.jpg communism.jpg nor.jpg because.jpg unconstitutional.jpg secretive.jpg secret.jpg >OtherImage1
```

APPENDIX C – CAPTURE 3 ARTIFACTS

C1 – TEXT CONVERSATION

Sender	Message	Timestamp
Kim Ill-Song	“Good afternoon, Ann.”	2014-07-02 22:38:55
Ann Dercover	“who is this?”	2014-07-02 22:39:15
Kim Ill-Song	“Castling”	2014-07-02 22:39:31
Ann Dercover	“where are you?”	2014-07-02 22:39:46
Kim Ill-Song	"I know I can't tell you that."	2014-07-02 22:40:05
Ann Dercover	"Do you know that there are people investigating Kim Ill-Song?"	2014-07-02 22:41:25
Kim Ill-song	"Of course. However, they will never know it is me behind the bribes."	2014-07-02 22:41:47
Ann Dercover	"still we should be careful. Pay attention. I want to meet in September at 5PM."	2014-07-02 22:42:54
Kim Ill-song	"At our old meetup spot?"	2014-07-02 22:43:06
Ann Dercover	"yes"	2014-07-02 22:43:28
Kim Ill-song	"What day?"	2014-07-02 22:43:44
Ann Dercover	"I told you to pay attention."	2014-07-02 22:50:32

(json.value.string == "Ann Dercover") && (json.value.string == "Kim Ill-song")

No.	Time	Source	Destination	Protocol	Length	Info
1	17:39:39.404130	199.87.160.87	192.168.1.5	HTTP/JSON	1158	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
3	17:40:10.379257	199.87.160.87	192.168.1.5	HTTP/JSON	1257	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
4	17:40:38.857756	199.87.160.87	192.168.1.5	HTTP/JSON	1143	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
5	17:42:16.120636	199.87.160.87	192.168.1.5	HTTP/JSON	1210	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
6	17:43:40.163433	199.87.160.87	192.168.1.5	HTTP/JSON	1270	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
8	17:44:23.764745	199.87.160.87	192.168.1.5	HTTP/JSON	1250	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)
9	17:51:31.711472	199.87.160.87	192.168.1.5	HTTP/JSON	1158	HTTP/1.1 200 OK , JavaScript Object Notation (application/json)

The sorted conversation can be seen above along with the timestamps for each message. Below are the 7 different JSON files used to piece together the conversation.


```
{
 "success": "messages retrieved",
 "result": {
  "recMessages": [
   {
    "messageId": "45b537c51e5cf2f90f31779e9ec8fc46",
    "messageType": "normal",
    "messageText": "Good afternoon,  Ann.",
    "recipientType": "phone",
    "recipientId": "14068522589",
    "senderType": "phone",
    "senderId": "14069243754",
    "senderName": "Kim Ill-song",
    "time": "2014-07-02 22:38:55",
    "messageStatus": "unread",
    "deliveryMethod": "onnet"
   }
  ],
  "sentMessages": [
   {
    "messageId": "d275712ce4c2b1b420bd1ba0728b79af",
    "messageType": "normal",
```

```
      "messageText": "this is a test",
      "recipientType": "phone",
      "recipientId": "14069243754",
      "senderType": "phone",
      "senderId": "14068522589",
      "senderName": "Ann Dercover",
      "time": "2014-07-02 22:34:13",
      "messageStatus": "read",
      "deliveryMethod": "onnet"
    }
  ],
  "brandedSystemMessages": [],
  "calls": [],
  "voicemails": [],
  "now": "2014-07-02 22:38:57",
  "largestCount": 1,
  "smsCreditBalance": 0,
  "callingCreditBalance": 0,
  "numTextsSent": 0,
  "numTextsRec": 0,
  "inviteCount": 0
 }
}
```

```
{
 "success": "messages retrieved",
 "result": {
  "recMessages": [
   {
    "messageId": "45b537c51e5cf2f90f31779e9ec8fc46",
    "messageType": "normal",
    "messageText": "Good afternoon,  Ann.",
    "recipientType": "phone",
    "recipientId": "14068522589",
    "senderType": "phone",
    "senderId": "14069243754",
    "senderName": "Kim Ill-song",
    "time": "2014-07-02 22:38:55",
    "messageStatus": "read",
    "deliveryMethod": "onnet"
   },
   {
    "messageId": "c113ed366ab0fba64f6215f41d6fb127",
    "messageType": "normal",
    "messageText": "Castling.",
    "recipientType": "phone",
    "recipientId": "14068522589",
    "senderType": "phone",
    "senderId": "14069243754",
    "senderName": "Kim Ill-song",
    "time": "2014-07-02 22:39:31",
    "messageStatus": "unread",
    "deliveryMethod": "onnet"
```

```
    }
  ],
  "sentMessages": [
   {
    "messageId": "eb232446d54193d00876830421797030",
    "messageType": "normal",
    "messageText": "who is this?",
    "recipientType": "phone",
    "recipientId": "14069243754",
    "senderType": "phone",
    "senderId": "14068522589",
    "senderName": "Ann Dercover",
    "time": "2014-07-02 22:39:15",
    "messageStatus": "read",
    "deliveryMethod": "onnet"
   }
  ],
  "brandedSystemMessages": [],
  "calls": [],
  "voicemails": [],
  "now": "2014-07-02 22:39:32",
  "largestCount": 2,
  "smsCreditBalance": 0,
  "callingCreditBalance": 0,
  "numTextsSent": 0,
  "numTextsRec": 0,
  "inviteCount": 0
 }
}
```

```
{
 "success": "messages retrieved",
 "result": {
  "recMessages": [
   {
    "messageId": "c113ed366ab0fba64f6215f41d6fb127",
    "messageType": "normal",
    "messageText": "Castling.",
    "recipientType": "phone",
    "recipientId": "14068522589",
    "senderType": "phone",
    "senderId": "14069243754",
    "senderName": "Kim Ill-song",
    "time": "2014-07-02 22:39:31",
    "messageStatus": "read",
    "deliveryMethod": "onnet"
   }
  ],
  "sentMessages": [
   {
    "messageId": "4125737ad17157e816310b4f2f98752a",
    "messageType": "normal",
    "messageText": "where are you?",
    "recipientType": "phone",
    "recipientId": "14069243754",
    "senderType": "phone",
    "senderId": "14068522589",
    "senderName": "Ann Dercover",
    "time": "2014-07-02 22:39:46",
```

```
      "messageStatus": "read",
      "deliveryMethod": "onnet"
     }
    ],
    "brandedSystemMessages": [],
    "calls": [],
    "voicemails": [],
    "now": "2014-07-02 22:39:54",
    "largestCount": 1,
    "smsCreditBalance": 0,
    "callingCreditBalance": 0,
    "numTextsSent": 0,
    "numTextsRec": 0,
    "inviteCount": 0
   }
}
{
  "success": "messages retrieved",
  "result": {
    "recMessages": [
     {
      "messageId": "dc821c4eeacd713cfef5cea15e803040",
      "messageType": "normal",
      "messageText": "I know I can't tell you that.",
      "recipientType": "phone",
      "recipientId": "14068522589",
      "senderType": "phone",
      "senderId": "14069243754",
      "senderName": "Kim Ill-song",
```

```
    "time": "2014-07-02 22:40:05",
    "messageStatus": "read",
    "deliveryMethod": "onnet"
   }
 ],
 "sentMessages": [
   {
    "messageId": "bdc2b81acb8e3bff28a1e87ff44ee5d7",
    "messageType": "normal",
    "messageText": "Do you know that there are people investigating Kim Ill-Song?",
    "recipientType": "phone",
    "recipientId": "14069243754",
    "senderType": "phone",
    "senderId": "14068522589",
    "senderName": "Ann Dercover",
    "time": "2014-07-02 22:41:25",
    "messageStatus": "read",
    "deliveryMethod": "onnet"
   }
 ],
 "brandedSystemMessages": [],
 "calls": [],
 "voicemails": [],
 "now": "2014-07-02 22:41:31",
 "largestCount": 1,
 "smsCreditBalance": 0,
 "callingCreditBalance": 0,
 "numTextsSent": 0,
 "numTextsRec": 0,
```

```
  "inviteCount": 0
 }
}

{
 "success": "messages retrieved",
 "result": {
  "recMessages": [
   {
    "messageId": "8197385d4b4222e32ec474fa497b70d8",
    "messageType": "normal",
    "messageText": "Of course. However,  they will never know it is me behind the bribes.",
    "recipientType": "phone",
    "recipientId": "14068522589",
    "senderType": "phone",
    "senderId": "14069243754",
    "senderName": "Kim Ill-song",
    "time": "2014-07-02 22:41:47",
    "messageStatus": "read",
    "deliveryMethod": "onnet"
   }
  ],
  "sentMessages": [
   {
    "messageId": "700b4051723f212b979cf068e59067b9",
    "messageType": "normal",
    "messageText": "still we should be careful. Pay attention. I want to meet in September at 5PM.",
    "recipientType": "phone",
    "recipientId": "14069243754",
```

```
      "senderType": "phone",
      "senderId": "14068522589",
      "senderName": "Ann Dercover",
      "time": "2014-07-02 22:42:54",
      "messageStatus": "read",
      "deliveryMethod": "onnet"
     }
    ],
    "brandedSystemMessages": [],
    "calls": [],
    "voicemails": [],
    "now": "2014-07-02 22:42:58",
    "largestCount": 1,
    "smsCreditBalance": 0,
    "callingCreditBalance": 0,
    "numTextsSent": 0,
    "numTextsRec": 0,
    "inviteCount": 0
  }
}
{
  "success": "messages retrieved",
  "result": {
    "recMessages": [
     {
      "messageId": "e5d6be661c5ed90cfb27a0fb50b33bf2",
      "messageType": "normal",
      "messageText": "At our old meetup spot?",
      "recipientType": "phone",
```

```
    "recipientId": "14068522589",
    "senderType": "phone",
    "senderId": "14069243754",
    "senderName": "Kim Ill-song",
    "time": "2014-07-02 22:43:06",
    "messageStatus": "read",
    "deliveryMethod": "onnet"
   },
   {
    "messageId": "b5860bdea833df4231c31dfbecbedf0d",
    "messageType": "normal",
    "messageText": "What day?",
    "recipientType": "phone",
    "recipientId": "14068522589",
    "senderType": "phone",
    "senderId": "14069243754",
    "senderName": "Kim Ill-song",
    "time": "2014-07-02 22:43:44",
    "messageStatus": "unread",
    "deliveryMethod": "onnet"
   }
  ],
  "sentMessages": [
   {
    "messageId": "9854f7107287ad4d6a6a69b25fc3da57",
    "messageType": "normal",
    "messageText": "yes",
    "recipientType": "phone",
    "recipientId": "14069243754",
```

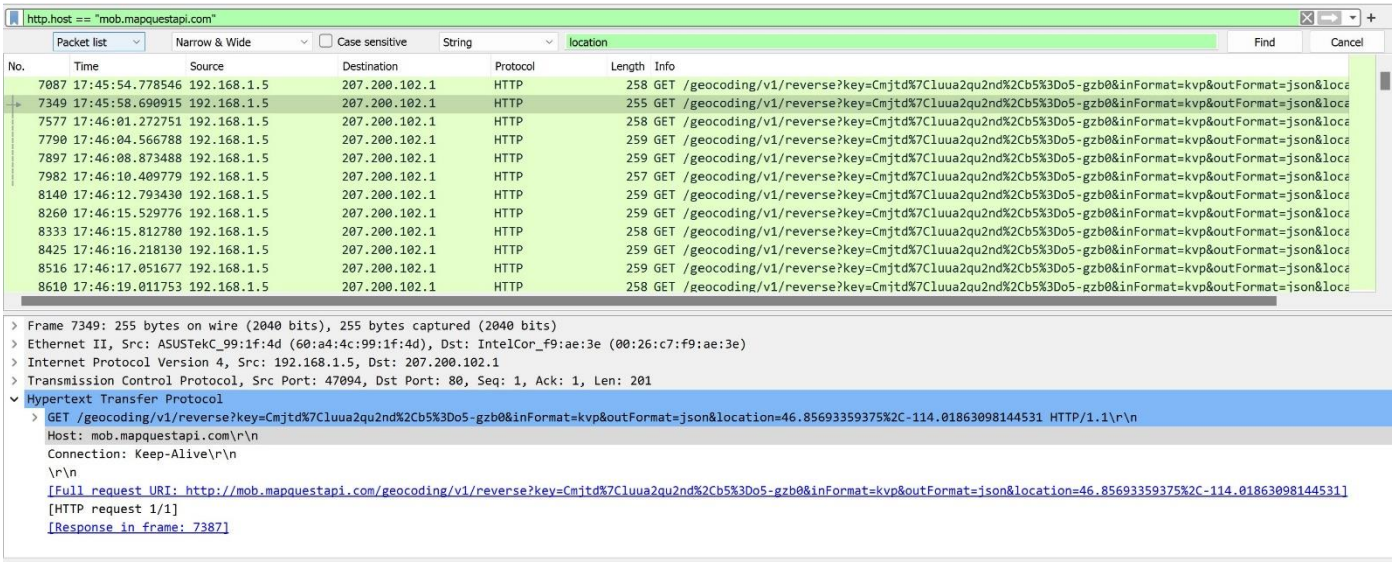
```
      "senderType": "phone",
      "senderId": "14068522589",
      "senderName": "Ann Dercover",
      "time": "2014-07-02 22:43:28",
      "messageStatus": "read",
      "deliveryMethod": "onnet"
    }
  ],
  "brandedSystemMessages": [],
  "calls": [],
  "voicemails": [],
  "now": "2014-07-02 22:43:45",
  "largestCount": 2,
  "smsCreditBalance": 0,
  "callingCreditBalance": 0,
  "numTextsSent": 0,
  "numTextsRec": 0,
  "inviteCount": 0
 }
}
{
 "success": "messages retrieved",
 "result": {
  "recMessages": [
   {
    "messageId": "b5860bdea833df4231c31dfbecbedf0d",
    "messageType": "normal",
    "messageText": "What day?",
    "recipientType": "phone",
```

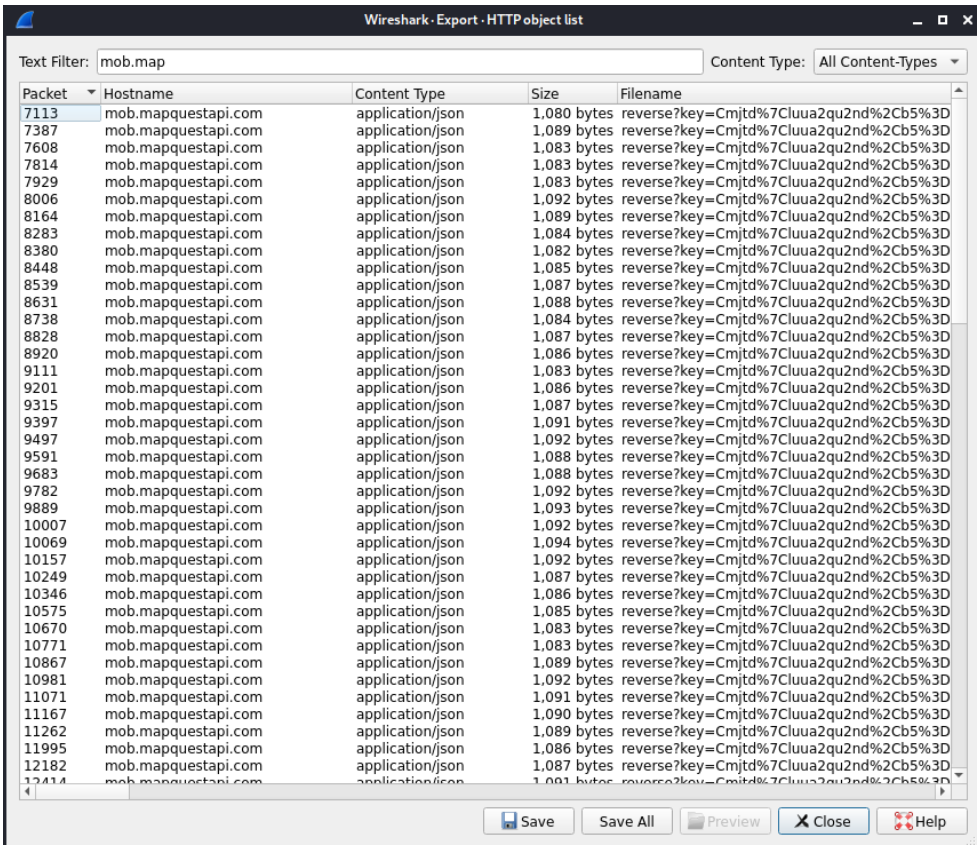
```
    "recipientId": "14068522589",
    "senderType": "phone",
    "senderId": "14069243754",
    "senderName": "Kim Ill-song",
    "time": "2014-07-02 22:43:44",
    "messageStatus": "read",
    "deliveryMethod": "onnet"
  }
],
"sentMessages": [
  {
    "messageId": "3ceeadc119a0225656c73b3fbfd3418f",
    "messageType": "normal",
    "messageText": "I told you to pay attention.",
    "recipientType": "phone",
    "recipientId": "14069243754",
    "senderType": "phone",
    "senderId": "14068522589",
    "senderName": "Ann Dercover",
    "time": "2014-07-02 22:50:32",
    "messageStatus": "read",
    "deliveryMethod": "onnet"
  }
],
"brandedSystemMessages": [],
"calls": [],
"voicemails": [],
"now": "2014-07-02 22:50:45",
"largestCount": 1,
```

```
    "smsCreditBalance": 0,
    "callingCreditBalance": 0,
    "numTextsSent": 0,
    "numTextsRec": 0,
    "inviteCount": 0
  }
}
```

C2 – LOCATION DATA

Below is the process of filtering the map requests and exporting the location data contained within each request.





Below is a complete list of the extracted location data:

No.	Time	Source	Destination	Protocol	Length	Info
7087	45:54.8	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85661315917969%2C-114.01860809326172 HTTP/1.1
7349	45:58.7	192.168.1.5	207.200.102.1	HTTP	255	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85693359375%2C-114.01863098144531 HTTP/1.1
7577	46:01.3	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85727310180664%2C-114.01868438720703 HTTP/1.1
7790	46:04.6	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.857601165771484%2C-114.01866912841797 HTTP/1.1
7897	46:08.9	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.858055114746094%2C-114.01866149902344 HTTP/1.1
7982	46:10.4	192.168.1.5	207.200.102.1	HTTP	257	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.8582878112793%2C-114.01864624023438 HTTP/1.1
8140	46:12.8	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.858524322509766%2C-114.01863861083984 HTTP/1.1
8260	46:15.5	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.858734130859375%2C-114.01864624023438 HTTP/1.1
8333	46:15.8	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-

						gzb0&inFormat=kvp&outFormat=json&location=46.8 5884475708008%2C-114.01864624023438 HTTP/1.1
8425	46:16.2	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 58943939208984%2C-114.01864624023438 HTTP/1.1
8516	46:17.1	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 59046936035156%2C-114.01864624023438 HTTP/1.1
8610	46:19.0	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 5914993286133%2C-114.01864624023438 HTTP/1.1
8714	46:22.2	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 59466552734375%2C-114.01864624023438 HTTP/1.1
8788	46:22.4	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 5957717895508%2C-114.01864624023438 HTTP/1.1
8899	46:23.3	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 5969161987305%2C-114.01864624023438 HTTP/1.1
9080	46:24.4	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 5980987548828%2C-114.01864624023438 HTTP/1.1
9177	46:25.5	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 5993194580078%2C-114.01864624023438 HTTP/1.1
9286	46:28.8	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 6029052734375%2C-114.01863098144531 HTTP/1.1

9374	46:30.8	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86052322387695%2C-114.01863861083984 HTTP/1.1
9473	46:32.5	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86075592041015%2C-114.01863098144531 HTTP/1.1
9567	46:34.4	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86098861694336%2C-114.01863098144531 HTTP/1.1
9663	46:36.4	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86122942871094%2C-114.01863861083984 HTTP/1.1
9758	46:37.9	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86147689819336%2C-114.01863098144531 HTTP/1.1
9871	46:40.1	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86159896850586%2C-114.01863098144531 HTTP/1.1
10000	46:42.5	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86183547973633%2C-114.01862335205078 HTTP/1.1
10043	46:43.2	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86206436157226%2C-114.01861572265625 HTTP/1.1
10133	46:45.2	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86228179931640%2C-114.01860046386719 HTTP/1.1
10229	46:47.2	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2

						Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.86248779296875%2C-114.01860046386719 HTTP/1.1
10317	46:48.8	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.86260223388672%2C-114.01859283447266 HTTP/1.1
10552	46:50.7	192.168.1.5	207.200.102.1	HTTP	257	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.86282730102539%2C-114.0185775756836 HTTP/1.1
10646	46:52.5	192.168.1.5	207.200.102.1	HTTP	257	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.86306381225586%2C-114.0185775756836 HTTP/1.1
10747	46:54.2	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.86330032348633%2C-114.01856231689453 HTTP/1.1
10843	46:55.5	192.168.1.5	207.200.102.1	HTTP	255	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.86342608496094%2C-114.0185546875 HTTP/1.1
10955	46:57.0	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.86355209350586%2C-114.01854705810547 HTTP/1.1
11047	46:57.7	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.86367416381836%2C-114.01853942871094 HTTP/1.1
11142	46:58.6	192.168.1.5	207.200.102.1	HTTP	257	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.86377809753418%2C-114.01853942871094 HTTP/1.1
11239	46:59.6	192.168.1.5	207.200.102.1	HTTP	257	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-

						gzb0&inFormat=kvp&outFormat=json&location=46.8 6387252807617%2C-114.0185317993164 HTTP/1.1
1194 9	47:4 9.0	192.168 .1.5	207.200.1 02.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 63704681396484%2C-114.01164245605469 HTTP/1.1
1215 6	47:5 0.7	192.168 .1.5	207.200.1 02.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 6370849609375%2C-114.01163482666016 HTTP/1.1
1238 3	47:5 3.1	192.168 .1.5	207.200.1 02.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 64017486572266%2C-114.01107025146484 HTTP/1.1
1259 5	47:5 5.3	192.168 .1.5	207.200.1 02.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 64044189453125%2C-114.01074981689453 HTTP/1.1
1276 3	47:5 6.8	192.168 .1.5	207.200.1 02.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 6404800415039%2C-114.01071166992188 HTTP/1.1
1294 5	47:5 8.9	192.168 .1.5	207.200.1 02.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 6408996582031%2C-114.01042175292969 HTTP/1.1
1313 5	48:0 0.6	192.168 .1.5	207.200.1 02.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 6408996582031%2C-114.01012420654297 HTTP/1.1
1327 7	48:0 2.1	192.168 .1.5	207.200.1 02.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 64078521728516%2C-114.00962829589844 HTTP/1.1
1337 6	48:0 3.6	192.168 .1.5	207.200.1 02.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 64070892333984%2C-114.0094223022461 HTTP/1.1

13464	48:05.5	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86406707763672%2C-114.00910186767578 HTTP/1.1
13555	48:06.8	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86407470703125%2C-114.00875854492188 HTTP/1.1
13651	48:09.6	192.168.1.5	207.200.102.1	HTTP	257	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86408233642578%2C-114.0084228515625 HTTP/1.1
13884	48:15.4	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86405818847656%2C-114.0074691772461 HTTP/1.1
13975	48:17.3	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86404189453125%2C-114.00716400146484 HTTP/1.1
14073	48:19.4	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86404189453125%2C-114.00694274902344 HTTP/1.1
14169	48:21.5	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86404800415039%2C-114.00680541992188 HTTP/1.1
14265	48:23.5	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86405563354492%2C-114.00670623779297 HTTP/1.1
14361	48:26.0	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86405818847656%2C-114.00662231445313 HTTP/1.1
14458	48:27.1	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2

						Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 64051818847656%2C-114.00646209716797 HTTP/1.1
1454 7	48:2 9.1	192.168 .1.5	207.200.1 02.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 64051818847656%2C-114.00627899169922 HTTP/1.1
1470 9	48:3 1.1	192.168 .1.5	207.200.1 02.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 64051818847656%2C-114.00605773925781 HTTP/1.1
1480 7	48:3 2.7	192.168 .1.5	207.200.1 02.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 64051818847656%2C-114.00592803955078 HTTP/1.1
1490 1	48:3 4.5	192.168 .1.5	207.200.1 02.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 6405944824219%2C-114.00563049316406 HTTP/1.1
1499 9	48:3 6.7	192.168 .1.5	207.200.1 02.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 6405944824219%2C-114.00534057617188 HTTP/1.1
1509 6	48:3 8.4	192.168 .1.5	207.200.1 02.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 6405563354492%2C-114.00506591796875 HTTP/1.1
1519 2	48:4 0.3	192.168 .1.5	207.200.1 02.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 64051818847656%2C-114.00477600097656 HTTP/1.1
1528 8	48:4 2.4	192.168 .1.5	207.200.1 02.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 64051818847656%2C-114.00452423095703 HTTP/1.1
1540 9	48:4 5.2	192.168 .1.5	207.200.1 02.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5-

						gzb0&inFormat=kvp&outFormat=json&location=46.864044189453125%2C-114.0042724609375 HTTP/1.1
15481	48:45.7	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.864044189453125%2C-114.00414276123047 HTTP/1.1
15592	48:47.7	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86404037475586%2C-114.00392150878906 HTTP/1.1
15987	48:51.1	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.863983154296875%2C-114.00354766845703 HTTP/1.1
16084	48:55.2	192.168.1.5	207.200.102.1	HTTP	257	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86393356323242%2C-114.0035171508789 HTTP/1.1
16181	48:57.2	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86381912231445%2C-114.00352478027344 HTTP/1.1
16268	48:59.1	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.863643646240234%2C-114.0035400390625 HTTP/1.1
16374	49:00.7	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86354446411133%2C-114.00354766845703 HTTP/1.1
16468	49:03.9	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86325454711914%2C-114.00360107421875 HTTP/1.1
16567	49:05.3	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86309051513672%2C-114.00376892089844 HTTP/1.1

16669	49:07.0	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86293411254883%2C-114.00396728515625 HTTP/1.1
16762	49:09.0	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86286163330078%2C-114.00408172607422 HTTP/1.1
16856	49:10.5	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86270141601562%2C-114.00432586669922 HTTP/1.1
16948	49:12.1	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86253356933594%2C-114.00457763671875 HTTP/1.1
17057	49:15.3	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86236190795898%2C-114.00481414794922 HTTP/1.1
17257	49:17.2	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86210632324219%2C-114.00520324707031 HTTP/1.1
17419	49:20.8	192.168.1.5	207.200.102.1	HTTP	257	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86183547973633%2C-114.0055923461914 HTTP/1.1
17519	49:22.6	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86166000366211%2C-114.00584411621094 HTTP/1.1
17612	49:24.5	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.86148452758789%2C-114.00609588623047 HTTP/1.1
17706	49:27.4	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2

						Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 6122131347656%2C-114.00647735595703 HTTP/1.1
1779 7	49:2 9.4	192.168 .1.5	207.200.1 02.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 6103057861328%2C-114.00672912597656 HTTP/1.1
1788 6	49:3 1.5	192.168 .1.5	207.200.1 02.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 60843658447266%2C-114.00699615478516 HTTP/1.1
1798 8	49:3 3.7	192.168 .1.5	207.200.1 02.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 6065673828125%2C-114.00727081298828 HTTP/1.1
1809 0	49:3 6.7	192.168 .1.5	207.200.1 02.1	HTTP	257	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 6037063598633%2C-114.0076675415039 HTTP/1.1
1837 2	49:4 0.7	192.168 .1.5	207.200.1 02.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 59989166259766%2C-114.00820922851563 HTTP/1.1
1847 3	49:4 2.0	192.168 .1.5	207.200.1 02.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 5979080200195%2C-114.00848388671875 HTTP/1.1
1856 8	49:4 3.8	192.168 .1.5	207.200.1 02.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 5969161987305%2C-114.00862121582031 HTTP/1.1
1866 4	49:4 5.4	192.168 .1.5	207.200.1 02.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5- gzb0&inFormat=kvp&outFormat=json&location=46.8 59500885009766%2C-114.00887298583984 HTTP/1.1
1877 4	49:4 7.5	192.168 .1.5	207.200.1 02.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2 Cb5%3Do5-

						gzb0&inFormat=kvp&outFormat=json&location=46.85930252075195%2C-114.00914001464844 HTTP/1.1
18857	49:48.9	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85910415649414%2C-114.00941467285156 HTTP/1.1
18957	49:50.6	192.168.1.5	207.200.102.1	HTTP	256	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.8590087890625%2C-114.0095443725586 HTTP/1.1
19058	49:52.3	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.858829498291016%2C-114.00979614257813 HTTP/1.1
19156	49:54.2	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.858646392822266%2C-114.01005554199219 HTTP/1.1
19251	49:57.6	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.858375549316406%2C-114.01044464111328 HTTP/1.1
19438	50:00.7	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.858123779296875%2C-114.01079559326172 HTTP/1.1
19538	50:02.3	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85795211791992%2C-114.01103973388672 HTTP/1.1
19651	50:04.4	192.168.1.5	207.200.102.1	HTTP	257	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.857880859375%2C-114.01127624511719 HTTP/1.1
19735	50:06.3	192.168.1.5	207.200.102.1	HTTP	257	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85765838623047%2C-114.0114517211914 HTTP/1.1

19848	50:10.0	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85751342773437 5%2C-114.01164245605469 HTTP/1.1
19929	50:12.1	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85749053955078%2C-114.01168823242188 HTTP/1.1
20021	50:14.0	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85747146606445%2C-114.01171112060547 HTTP/1.1
20113	50:15.2	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85741806030273 4%2C-114.01179504394531 HTTP/1.1
20209	50:17.2	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85733413696289%2C-114.01190948486328 HTTP/1.1
20299	50:19.1	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85723495483398 4%2C-114.01204681396484 HTTP/1.1
20388	50:20.8	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85718154907227 6%2C-114.01212310791016 HTTP/1.1
20496	50:22.7	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85708236694336%2C-114.01225280761719 HTTP/1.1
20593	50:24.8	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85697937011719%2C-114.01237487792969 HTTP/1.1
20686	50:26.9	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2

						Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.856834411621094%2C-114.01256561279297 HTTP/1.1
20915	50:29.4	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85672378540039%2C-114.01271057128906 HTTP/1.1
21006	50:31.9	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85659790039062 5%2C-114.01287078857422 HTTP/1.1
21105	50:33.5	192.168.1.5	207.200.102.1	HTTP	258	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85647201538086%2C-114.01302337646484 HTTP/1.1
21214	50:36.1	192.168.1.5	207.200.102.1	HTTP	259	GET /geocoding/v1/reverse?key=Cmjtd%7Cluua2qu2nd%2Cb5%3Do5-gzb0&inFormat=kvp&outFormat=json&location=46.85631942749023 4%2C-114.01313018798828 HTTP/1.1

Below is the script created by the investigator to strip the excess data from the CSV file.

```
import os

filepath = "cat Exportedlocations.csv | grep 'location' | cut -d ',' -f 7 | cut -b 102-150 | cut -d ' ' -f 1 | sed 's/%2C/,/g' >Locations.txt"

os.system(filepath)

```

After converting the CSV to a KML file it was imported to google Earth where the number 17 can be seen below.