Peer-reviewed version available at *Information* **2017**, *8*, 71; doi:10.3390/info8030071

Article

Pokémon GO Forensics: An Android Application Analysis

Joshua Sablatura[1,*] and Umit Karabiyik[2*]

[1] Department of Computer Science, Sam Houston State Univeristy, Huntsville, Texas; jds088@shsu.edu
[2] Department of Computer Science, Sam Houston State University, Huntsville, Texas; umit@shsu.edu
* Author to whom correspondence should be addressed.

Abstract: As the geolocation capabilities of smartphones continue to improve, developers have continued to create more innovative applications that rely on this location information for their primary function. This can be seen with Niantic's release of Pokémon GO, which is a massively multiplayer online role playing and augmented reality game. This game became immensely popular within just a few days of its release. However, it also had the propensity to be a distraction to drivers resulting in numerous accidents, and was used to as a tool by armed robbers to lure unsuspecting users into secluded areas. This facilitates a need for forensic investigators to be able to analyze the data within the application in order to determine if it may have been involved in these incidents. Because this application is new, limited research has been conducted regarding the artifacts that can be recovered from the application. In this paper, we aim to fill the gaps within the current research by assessing what forensically relevant information may be recovered from the application, and understanding the circumstances behind the creation of this information. Our research focuses primarily on the artifacts generated by the Upsight analytics platform, those contained within the bundles directory, and the Pokémon Go Plus accessory. Moreover, we present our new application specific analysis tool that is capable of extracting forensic artifacts from a backup of the Android application, and presenting them to an investigator in an easily readable format. This analysis tool exceeds the capabilities of UFED Physical Analyzer in processing Pokémon GO application data.

Keywords: digital forensic tool, mobile application forensics, geolocation, Upsight, Pokémon GO, Pokémon GO Plus

1. Introduction

On July 6, 2016 Niantic, a subsidiary of Google, in conjunction with Nintendo and The Pokémon Company released Pokémon GO [1] a massively multiplayer online role playing and augmented reality game for mobile devices that is based upon the late 1990s Pokémon cartoon series, and the Nintendo Gameboy game series. The game which was built upon the same framework as Niantic's earlier game, Ingress, utilizes the GPS capabilities provided by most smartphones to allow the user to locate, capture, and train virtual creatures within the real world. These creatures appear based on the physical location of the user, and the type of pokémon encountered.

Soon after its release Pokémon GO quickly became one of the most popular mobile applications on the market. Within just five days of its release Pokémon GO was installed on 10.8% of U.S. android phones [2]. Furthermore, based on application usage data from July 8^{th}, users were spending an average of 43 minutes a day within the application [3]. According to the 2017 U.S. Cross-Platform Future in Focus report published by comScore, within just seven days of its release the popularity of Pokémon GO peaked at over 28.5 million daily unique visitors, and has since leveled out at around five million daily unique visitors in December 2016, with approximately 60% of its user base comprised of 18-34 year-old Millennials who grew up with the cartoon and game series [4]. This steep decline can largely be attributed to a lack of long term goals in the game, and certain features that were expected by the users such as the pokémon tracker, player-versus-player battling, and pokémon trading. However, despite these declining usage numbers Pokémon GO has generated over one

Peer-reviewed version available at Information 2017, 8, 71; doi:10.3390/info8030071

billion dollars in revenue since its launch [5], and as of April 2017, maintains a user base of 65 million monthly active users. These numbers are predicted to increase as Niantic has announced plans to release several long-awaited features including player-versus-player battling, and pokémon trading in a coming update [6]. Pokémon GO has also won several awards including The Game Award for Best Mobile/Handheld Game, and Best Family Game in 2016.

Pokémon GO was an immensely popular game that succeeded in increasing the physical activity of many people as it required them to go outside and walk around in order to find pokémon. A study that monitored users for 30 days shortly after the games release, estimated that on average users increased their activity by 1,473 steps while playing the game [7]. This surge of physical activity could be seen in news reports as hundreds of people flocked to sightings of rare pokémon within New York City's Central Park [8], and Santa Monica Pier in California [9]. This huge surge of people taking to the outdoors to seek pokémon lead to some very interesting discoveries as was reported by the media in the case of a 19 year-old Wyoming girl who stumbled upon a dead body while searching for a water type pokémon by a river near her home [10]. Pokémon GO has also been attributed to helping two marines, who were visiting a pokéstop in Fullerton Park in California, catch a murder suspect after they noticed a suspicious man approaching a mom and her three kids [11].

However, not all of the news stories about Pokémon GO are positive in nature. In O'Fallon Missouri four armed robbers placed a lure module on a secluded pokéstop to lure in unsuspecting pokémon trainers [12]. There have also been numerous reports of distracted pedestrians being injured, and/or trespassing on private property or restricted areas to capture pokémon prompting the Joint Base Lewis-McChord military base near Tacoma, Washington to issue a warning to pokémon trainers [13]. Furthermore, it did not take pokémon trainers long to realize they could cover more ground by driving instead of walking. John Ayers of San Diego State University searched through social media posts on Twitter and examined news stories for evidence of people having accidents while driving and playing Pokémon GO during the course of 10 days shortly after the games release. The researchers discovered that there were 14 unique crashes, while approximately 18% of tweets indicated that a person was playing and driving, while 11% of tweets indicated that a passenger was playing [14]. In fact there have been numerous news reports of Pokémon GO players who have crashed their automobiles into a tree, parked police car, school, and at least one reported instance of a woman who was killed when a distracted driver playing Pokémon GO struck two pedestrians [15–18].

Because of Pokémon GO's widespread popularity, propensity to be a distraction to pedestrians and divers, and its use as a tool to attract victims of armed robbery it is likely that a forensic investigator will encounter a mobile investigation where the data contained in the application will need to be analyzed. Unfortunately, since Pokémon GO is a new application little research has been conducted that focuses on understanding the forensically relevant artifacts found within the application, and the circumstances that surround their creation. This paper seeks to expand upon the current research in order to discover any forensically relevant information that may be contained in the Pokémon GO application, and develop an understanding of how this information was created by the application. This information may include corroborative evidence that may place a user at the location of a crime, or may support the idea that a user was distracted by the application while walking or driving. Furthermore, this research expands upon prior research by including the Pokémon GO Plus device, and examining the metadata contained within images generated by the application.

This research furthers the understanding of Pokémon GO application forensics by making the following contributions to the current state-of-the-art:

i) Identifies the most recent time frame in which the application was run in the foreground of a mobile device by examining the session information contained within the *upsight.xml* file.
ii) Identifies the relative geolocation of the user at the end of the most recent time frame in which the application was run in the foreground of a mobile device by examining the information stored in the *upsight.db* file.

iii) Identifies additional methods that can indicate user activity prior to and during the most recent time in which the application was run in the foreground of a mobile device by examining the timestamps of files contained in the *bundles* directory, the timestamp contained in the *lastPushTokenRegistrationTime* value, and the entries contained in the Pokémon Trainer's "journal".
iv) Determines the use of a Pokémon GO Plus accessory with the application by the presence of the *pgp.xml* file. This artifact contains the MAC address of the device connected to the mobile device, and can be used to link a specific Plus accessory to the mobile device.
v) Determines the effects of the Pokémon GO Plus accessory on the artifacts that are generated by the application.
vi) Determines that no geolocation information can be found within the metadata of the images created with the in-game camera feature.
vii) Provides additional discussion pertaining to the geolocation information that can be found in the Cell ID values within the Crittercism logs of applications used prior to July 31, 2016.
viii) Develops an application specific analysis tool capable of extracting these artifacts from a backup of the Android application, and presenting them to an investigator in an easily readable format. This analysis tool exceeds the capabilities of UFED Physical Analyzer in processing Pokémon GO application data.

The contributions made by this paper can be used by forensic investigators to establish a timeline of user activity within the Pokémon GO application, establish the relative location of the user, and to determine if the user was utilizing a Pokémon GO Plus accessory. This information can be useful during investigations involving pedestrians or drivers that may have been involved in an accident that resulted in death or serious injury to themselves or others as a result of their failure to pay attention to their surroundings while playing the game. Using the information contained within the application the investigator is capable of determining if the application was running in the foreground of the mobile device at the time of the incident, and/or if additional indicators of game activity exist close to the time at which the incident occurred. Furthermore, the geolocation information contained within the application can be used in conjunction with the timestamp information to provide supporting evidence that an individual witnessed, or participated in a crime or incident. While this location information cannot specifically place an individual at a location, it can be used to support the idea that they were in the vicinity of the crime at the time it occurred, or it may be used to exclude them from the crime by indicating they were located elsewhere at the time of the incident.

The remainder of this paper is structured as follows. A description of the basic gameplay, discussion of forensically relevant artifacts that are expected to found within the game, and the gaps in the prior research are presented in Section 2. Section 3 describes the methodology, and results of our analysis on the artifacts found in the application. The results of this analysis are used to develop an analysis tool that can extract this information from the application. Discussion on the tool we developed presented in Section 4. Finally, concluding remarks and future work are discussed in Section 5.

2. Literature Review

Due to the limited research available on the forensic artifacts that can be found in the application a brief overview of Pokémon GO gameplay is presented in Section 2.1. Section 2.2 describes the forensically relevant information that may be found within the application. Finally the gaps within the prior research are presented in Section 2.3.

2.1. Pokémon GO: Gameplay

The majority of gameplay takes place within a simple map that corresponds to the buildings and streets around the user's physical location. In order to move the user's avatar around the map, the user must physically move to the corresponding location. When a user is close to a wild pokémon, an image of the creature appears on the map. The user can then select the image to encounter the

Peer-reviewed version available at Information 2017, 8, 71; doi:10.3390/info8030071

pokémon. During the encounter, the smartphone's camera is used to superimpose an image of the pokémon into the physical world and the user is given the opportunity to capture the creature. The application includes an in-game camera that allows users to capture pictures of the pokémon they encounter.

Pokémon GO uses two in game locations that are tied to physical locations in the real world: *Pokémon Gyms* and *Pokéstops*. These locations correspond to public buildings, parks, or statues that exist in the real world. Pokémon gyms allow pokémon trainers, which belong to one of three different teams, to battle for control of the gym. On the other hand, pokéstops provide pokémon trainers with items such as pokéballs, potions, raspberries, incense, pokémon eggs, and lure modules. A pokéstop can be equipped with a lure module in order to attract pokémon to that location. This effect benefits, and is visible to all pokémon trainers in the area. Therefore, by placing a lure module on a pokéstop a user is not only going to attract pokémon, but will likely attract other trainers looking to benefit from its effects. This creates an interesting social component to the game, as users are likely to interact with each other at pokéstops, gyms, or as they are searching for pokémon.

Pokémon GO supports the Pokémon GO Plus accessory, which is a small Bluetooth low energy device that allows pokémon trainers to play the game without having to look at their smartphone. The device will notify a user when they are near a pokémon or pokéstop by vibrating and flashing an LED. The user can then push the button on the device to attempt to capture the pokémon or collect items from the pokéstop.

2.2. Forensically Relevant Artifacts

Maus et al. noted that many smartphone applications make use of geolocation information, and this information can be valuable to forensic investigators. However, because there is no standard format regarding how this information is stored, it can be difficult to extract valuable information from the application [19]. Based on the description of gameplay it becomes evident that Pokémon GO can provide a wealth of geolocation information; as not only is the user's physical location used to determine the location of the pokémon trainer's avatar within the game, but two key components of gameplay, pokémon gyms and pokéstops, are tied to physical locations. Therefore, any artifacts that reference a pokéstop, or a pokémon gym within the application can place the user at a specific location.

Other forensically relevant artifacts would include those that can indicate when the application was being used by the user. These could be timestamps of files that are modified during gameplay, or logs that indicate periods of user activity. Additional forensically relevant artifacts could be those that may indicate that a Pokémon GO Plus device was recently used, and any metadata found within the images created by the in-game camera.

2.3. Limitations of Prior Research

The most definitive work on the artifacts created by the Pokémon GO application comes from Cindy Murphy, the co-owner and president of Gillware Digital Forensics. This work focused on the artifacts that can be found in both iOS and android devices. Several *.sqlite* databases and *.plist* files were found in the iOS version of the application. The most forensically relevant data could be found within the *com.nianticlabs.pokemongo.plist* file that contained information that describes the user's device, and username. Murphy also noted that the timestamps for the files within the *bundles* directory appeared to correspond to game activity. Murphy also noted that the *com_upsight* directory contained three SQLite databases that contained multiple *.plist* formatted entries that are believed to be related to a marketing and analytics platform for web and mobile application developers. The timestamps for these files also corresponded to user activity within the game.

On the other hand, in the android version of the application the email address associated with the user's account was found inside the *com.nianticlabs.pokemongo.PREFS.xml* file. The *bundles* directory was also found within this version of the application, and as with the iOS version the timestamps of the files within this directory corresponded to when the game was played. Furthermore, the android

Peer-reviewed version available at Information 2017, 8, 71; doi:10.3390/info8030071

version of the application contained several log files within the *com.crittercism* directory that are associated with the Apteligent (formally known as Crittercism) application performance monitoring system. These logs, known as breadcrumbs, allow the developer to create custom logs, and contained information such as application errors, pokémon encounters, caught pokémon, encounter updates, attempted encounters, and cell removals. Two interesting pieces of information were contained in these logs: Encounter IDs, and Cell IDs. The encounter ID number is a 19 digit number that is believed to be a 64 bit representation of the attributes of a specific pokémon. The cell ID number was a little more interesting as Murphy with the assistance of Warren Raquel discovered that this was a 19 digit integer that when converted into hexadecimal was a representation of a small region of the global map on a Hilbert Curve [21] that corresponds to the user's geolocation.

Additional research in to the forensic artifacts found within Pokémon GO was conducted by The Security Sleuth [22]. This research focused only on android devices and used open-source acquisition and analysis tools. During the study the research arrived at the same conclusions about the artifacts found in the file system, the ability to recover the user's email address in the *com.nianticalbs.pokemon.PREFS.xml* file, and the information obtained from the log files in the *com.crittercism* directory. However, the researchers were unable to convert the cell ID number into physical GPS coordinates. Similarly, additional research was conducted by Head [23] that focused on both iOS and android devices. While Head was unable to translate the Cell ID numbers to physical GPS coordinates, this research corresponds to the findings presented by Murphy and The Security Sleuth.

During the course of this research, additional research on the artifacts contained within Pokémon GO was conducted by Lawson [24], who examined both iOS and android devices and focused on the artifacts that could be recovered from the application in three different states: (1) installed but never opened, (2) after one hour of game play, (3) after the application has been deleted. This research mentions that some geolocation information, and timestamps indicating game activity can be found within the upsight.xml, and upsgitht.db files within android applications. However, no additional research was conducted by Lawson to discover the circumstances behind this information's creation. Furthermore, none of the prior research examines the artifacts created by the Pokémon GO Plus accessory, or the metadata contained within the images created by the in-game camera.

3. Forensic Analysis of Pokémon GO

The methodology used to conduct this analysis of the Pokémon GO application was twofold. First, the prior research conducted by Murphy was validated to identify the artifacts that can be recovered from the current version of the Pokémon GO application. Then based on these artifacts, a more detailed testing methodology was developed to assess the effect that different user actions have on these artifacts. Note that the scope of this research project is limited to the android version of the application run on Samsung Galaxy S6 and Samsung Galaxy S7 devices (the most recent devices at the time of research).

3.1. Data Acquisition Method

Data acquisition with Pokémon GO, on the Samsung Galaxy S6 device proved to be difficult as the application utilizes Google's SafetyNet API, which examines the software and hardware information of the device to assess its integrity. This API performs a check to determine if the device has been tampered with or has been rooted, and if so, will prevent the user from logging into Pokémon GO's application servers. Further complicating the data acquisition process, rooting the Samsung Galaxy S6 device involved utilizing ODIN to flash a rooted firmware patch into the operating system. The only way that this could be undone was to re-flash the device with a clean firmware image. Because multiple data acquisitions after various amounts of application usage were required for later tests, an acquisition strategy had to be devised that did not involve rooting the android device.

Reverse engineering the Pokémon GO application with *jadx* revealed that the application manifest file does not specify a value for the *allowBackup* attribute. According to the android documentation the default for this attribute is "true" indicating that the application and all of its data is capable of being backed up using *android debug bridge (ADB)*. Therefore, *ADB* was used to create a backup of the *com.nianticlabs.pokemongo* application data. Once a backup of the application has been created, the information must be extracted from the backup file using *Android Backup Extractor*. This will produce a compressed tar archive that contains all of the Pokémon GO application data within the *apps/com.nianticlabs.pokemongo* directory.

This acquisition methodology imposes two limitations on this research. First, this method is only able to obtain the equivalent of a logical acquisition of the device. Therefore, any files that are deleted will not be recoverable during the analysis. Additionally, this acquisition method will only retrieve artifacts that reside within the Pokémon GO Application directory and data storage areas. Artifacts that are created by the android operating system and may reside in other areas of the device would not be acquired.

3.2. Preliminary Data Analysis

After the Pokémon GO application was installed on the Samsung Galaxy S6 device it was opened for the first time at 1:34 PM on January 17, 2017. During this time a new character was created with the username 'ForensicGuy1394', and Charmander was captured as the starter pokémon. Three additional pokémon Exeggcute, Venonat, and Weedle were captured before the application was terminated at 1:56 PM on January 17, 2017. A full backup of the phone was taken using *ADB*. This backup was then extracted and analyzed to validate the prior research, and identify any forensically relevant information.

The findings of this preliminary data collection correlate with those presented by Murphy, with the exception of the *com.crittecism* logs within the *f* directory. These logs which contained a wealth of geolocation information were not discovered within the backup of the application. Reverse engineering the Pokémon GO application revealed numerous references to the Crittercism platform indicating that it is still included within the application, but has been disabled or is logging information to a remote server instead of writing the logs to the local device.

Based on the results of this analysis, there are several artifacts that could be useful to an investigator and could indicate when and where the application was being used. The first of these include the session information contained within the *upsight.xml* file. Using this information it may be possible to determine when the application was being used based on the session start timestamp, and last known session timestamp which corresponded to when the application was launched and terminated. This file also contained statistical information about the player such as experience points earned, player level, and the number of items and pokémon in the user's possession. Other forensically relevant data includes the presence of geolocation information within the *upsight.db* file. The UNIX timestamp in the *lastPushTokenRegistrationTime* variable within the *com.upsight.android.googleadvertisingid.internal.registration.xml* file which correlates with when the application was launched. The email address of the user is stored in the *com.nianticlabs.pokemongo.PREFS.xml* file, and the timestamps associated with the files are located in the *bundles* directory. These files are believed to contain Unity 3D models for pokémon, items, and other assets in game. These models are downloaded dynamically as they are needed by the application and could therefore indicate user activity.

3.3. Targeted Data Analysis

During the data collection phase a series of twelve data dumps were taken of the Pokémon GO application. Before each data dump the application was used for a short period of time without being backgrounded, and/or terminated. These tests had four goals: (1) determine the effects of running the application in the foreground and background of the mobile device, (2) determine the effects of

Table 1. Summary of data generated by Upsight

Item	Description
Session Number	Number of the current session.
Current Session Duration	Duration of the current session in seconds.
Session Start Timestamp	Date and time that the current session started represented as a UNIX timestamp.
Last Known Session Timestamp	Date and time that the current session ended represented as a UNIX timestamp.
Sequence ID	A continually incrementing number that appears to be proportional to the players activity.
Install Time Stamp	Date and time at which the application was installed represented as a UNIX timestamp.
SID	Security Identifier. This value never changes.
Player XP	Player's experience level at the end of the current session.
Item Count	Number of items the user has in their possession at the START of the current session.
Pokémon Count	Number of pokémon a user has in their possession at the START of the current session.
Player Level	Level of the player at the end of the current session.
Past Session Time	Total active game play since the game was installed in seconds.
Upsight.model.location (upsight.db)	GPS coordinates of the last location of game play. Stored within a SQLite database.
lastPushTokenRegistrationToken	Value within *com.upsight.android.googleadvertisingid.internal.registration.xml* that may indicate user activity around a certain point in time.

the starting and ending location of the user on the data within the artifacts created by the Upsight platform, (3) gaining additional insight into each of the values contained within the *upsight.xml* file, and (4) to determine the effects of using the Pokémon GO Plus accessory on these artifacts. Appendix A describes the purpose, and actions taken prior to each dump of the application. It is important to note that encounters with previously caught pokémon are not recorded in this table.

Each of these dumps were extracted, and the information contained within the *upsight.xml* and *upsight.db* files as well as the *lastPushTokenRegistrationTime* value were analyzed. During this analysis forensically important values, which can be found in Appendix B, were compared with their previous values from the prior dump, and were cross referenced with the actions taken by the user prior to generating the dump as shown in Appendix A.

A summary of the results of this analysis for the information generated by the *Upsight* platform are found within Table 1.

3.4. Upsight Session Information

Based on this analysis several trends begin to emerge. First, it becomes evident that the session number is a sequentially incrementing value representing the current session for the device. This holds true until Dump4 where the session number increments from 5 to 8. This can be explained by examining the session start time and last known session time values within Appendix B. These values for Dumps 1, 2, and 3 correspond to when the application was started and terminated as indicated in Appendix A. However, in Dump4 the session start time corresponds to the time the application was resumed at 3:38 PM on February 24. During the data collection for Dump4, the application was backgrounded twice. The first occurred at 3:14 PM with the application being resumed at 3:30 PM, and the second occurred at 3:33 PM with the application being resumed at 3:38 PM. Therefore, the session start time corresponds to the time at which the user begins running the application within the foreground of the mobile device either by starting the application, or resuming an application that was previously running in the background.

Likewise, the last known session time values recorded within Appendix B for Dumps 6, 7, 11, and 12 correlate to when the application was backgrounded (see Appendix A). This indicates that the last known session time value corresponds to when the application was terminated, or backgrounded by switching to another application, receiving a phone call, or by locking the device. In other words, the current session ends when the application is either terminated or backgrounded. If the application was backgrounded, and then brought back to the foreground a new session would start, and the session number would be incremented. Because the application was backgrounded twice during Dump4 the session number was incremented twice. Hence, instead of Dump4 having a session number of 6, the data in Appendix B indicates a value of 8.

Therefore, it is possible to define a session that is recorded in the *upsight.xml* file as the time in which the application was running in the foreground of the user's mobile device. In other words, this corresponds to the most recent time when the application was actively played. This information can be used by a forensic investigator to provide definitive evidence of the last time the application was actively played by the user.

Furthermore, by examining the current session duration value given in Appendix B, and comparing this value to the session start and last known session time values, it becomes evident that this value represents the length of the current session in seconds. Likewise, by examining the past session time and current session time values, it becomes evident that the past session time value is a counter of the total time spent actively playing the game in seconds excluding the time spent in the current session.

3.4.1. Geolocation Information

Mapping the geolocation information found in the *upsight.model.location* record within the *upsight.db* file indicate that these coordinates approximate the last known location of the user at the end of the current session. This can be seen within Dump4, which initially started at the Catholic Student Center, and then ended by the Don Sanders Baseball Stadium. The coordinates in the Upsight database (30.71, -95.54) placed the user about 1,000 feet south of the true final location at the Don Sanders Stadium. Furthermore, if the application was backgrounded in one location before being terminated in another location the coordinates in the Upsight database will reflect the location where the application was backgrounded. This can be observed within Dump7, which started at the Catholic Student Center, and then proceeded to Walmart where the application was backgrounded. The application was then terminated 20 minutes later at another location. The coordinates within the Upsight database (30.71, -95.47) placed the user about 1,400 feet south of the true location within the Walmart parking lot.

A significant amount of precision is lost when these values are stored by the Upsight platform because they are rounded to two decimal places. Therefore, these values can be two to three blocks off from the user's true last known location. These coordinates can be used by an investigator to place the user somewhere in the vicinity of an incident or location, but cannot be used as definitive evidence that a user participated in or witnessed the incident.

3.4.2. Gameplay Prior to Current Session

The timestamps of the files collected from the bundles directory may also be used by an investigator to develop a timeline that may indicate game activity prior to the most recent active session as indicated by information in the *upsight.xml* file. The timestamps of these files correspond to the times when new pokémon are encountered, or an update within the game has altered the 3D models, animations, or sounds for an existing pokémon. These files are downloaded dynamically during gameplay.

Similarly, the *lastPushTokenRegistrationTime* value found within the *com.upsight.android.googleadvertsingid.internal.registration.xml* file can indicate gameplay. However, this value is sporadically updated as indicated by the data within Appendix B. This value corresponded

Peer-reviewed version available at Information 2017, 8, 71; doi:10.3390/info8030071

with the session start time of Dump1, but the value was never updated during the other six dumps which occurred in the same day. This value was still not updated during Dump8, which occurred on the following day. Furthermore, for Dumps 9, 10 and 11, which took place on March 19, this value contained a timestamp indicating that it was last updated on March 16, three days before the current gameplay session. Therefore, contrary to the research presented by Lawson this value cannot be used as an indicator of when the most recent game activity began. However, it could indicate prior activity.

It is also possible to determine a player's activity by examining the pokémon trainer's "journal" in the application. The "journal" records the 50 most recent pokémon encounters, and pokéstop visits. Unsuccessful visits to pokémon gyms do not appear to be recorded within the "journal" logs. Visits in which a pokémon trainer wins against the gyms opponents may be recorded too. However, due to the high combat strength of the pokémon that reside in gyms the logging of successful gym battles within the pokémon trainer's "journal" was unable to be tested. The information within the "journal" should be checked after an image of the device has been acquired as this will cause the upsight information to be updated.

3.4.3. Pokémon GO Plus

Dumps 9, 10, 11, and 12 focused on determining the effects of the Pokémon Go Plus accessory on the artifacts generated by the application. The most notable artifact relating to the use of the Pokémon GO Plus accessory is the creation of the *pgp.xml* artifact in the *sp* directory. This artifact contains the Bluetooth MAC address of the device that was connected to the mobile application, and the Bluetooth encryption key. This file is only modified when the encryption key needs to be updated. The content of the *pgp.xml* file is shown below:

```
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
    <string name="BT_KEY_98:B6:E9:0F:53:0C">HSYcDgN5pyxJFGwX7x0mbw==
    </string>
</map>
```

Furthermore, by comparing the session start, and last known session time values in Appendix B with their corresponding entries in Appendix A, it becomes evident that the device has no effect on these artifacts. As noted within Appendix A, during the data collection process for Dump10, the application had to be brought to the foreground at 4:50 PM to reconnect the Pokémon GO Plus accessory, and prior to terminating the application at 5:13 PM the application was briefly brought to the foreground when the screen was unlocked. This caused the Upsight session information to update as indicated by the session numbers, session start and end time values given in Appendix B.

This research indicates that the use of the Pokémon GO Plus accessory will prevent the download of additional "bundles" files, even for pokémon that are encountered for the first time. Instead, these files will be downloaded during the next session. However, it is still possible to determine a player's activity by examining the pokémon trainer's "journal" in the application.

3.5. Legacy Artifacts: Crittercism Logs

During the preliminary data collection phase a backup of a Samsung Galaxy S7 was created to provide an additional data benchmark using a device that has had Pokémon GO installed since July 7, 2016 shortly after the game was released. In addition to the forensically relevant data that was present on the test device, the Apteligent (Crittercism) logs which were mentioned as part of Murphy's research were also discovered. However, these logs were not actively maintained by the application. The timestamps associated with these files indicated that the last time they were updated was July 31, 2016. It appears that when the Crittercism platform was disabled, or modified to submit the logs to a remote server the directories containing the log files were not removed. Therefore, it may be possible for an investigator to encounter a mobile device that still has these logs intact provided that

the application was installed prior to July 31, 2016. These logs may be relevant to an investigation, despite not being actively maintained by the application, depending on the investigation timeline.

These logs were analyzed for forensically relevant data, however because these logs are no longer generated by the application it is impossible to determine the exact circumstance behind their creation. Since these logs were created prior to engaging in this research the results of this analysis may not be completely accurate.

As indicated by Murphy's research these logs were discovered within the *f/com.crittercism* directory inside of the *current_bcs* and *previous_bcs* directories. Each of these directories contains 50 files that may contain geolocation information encoded inside of the Cell ID numbers within the logs. Each log has a naming convention with the following format: 1.1469994833869.000000X, where X represents a sequentially incrementing number. According to the data collected from the Galaxy S7 phone, this Cell ID information can be discovered in two different logs: "Removing Cell ID" logs, and "Updating Encounter" logs. As indicated by Murphy, the Cell ID number is a 19 digit integer that should be converted into a hexadecimal number. This hex value, which is a representation of an area of the global map on a Hilbert Curve, can then be converted to GPS coordinates with the use of Google's S2 geometry library.

Converting and mapping the Cell ID number found in the Galaxy S7 dump reveal that these logs date back to a trip to Tombstone, Arizona during the summer. During this trip a significant amount of time was spent visiting the shops along Allen St. and Fremont St., including the Old Tombstone Wild West Theme Park. The coordinates contained within the "Removing Cell ID" logs appear to occur on the edges of where the application was likely to be used. On the other hand, the coordinates in the "Updating Encounter" logs appear to correspond to areas where user activity is likely to have occurred. Analyzing the timestamps of these log files indicate that the log entries were created within 2 to 3 minutes of each other. This indicates that the information contained within these logs is a snapshot of the user's activity. This snapshot may correspond to the latest activity of the user.

3.6. Image Metadata

As shown in Appendix A, during Dump4 two pictures were taken with the applications in-game camera during an encounter with a pokémon. The first image, *IMG_2017-02-24-15473410.png*, was a Wooper taken at the Intramural Fields. This was captured at 3:47 PM on February 24, 2017. The second image, *IMG_2017-02-24-15513605.png*, was a Weedle taken at Bowers Stadium. This image was captured at 3:52 PM on February 24, 2017.

Both of these images were analyzed with ExifTool as shown in Figure 1, and Figure 2. The file modification date/time information correlate with when the images were created. Furthermore, the file naming convention indicates the date in which the image was created. Additional metadata within the image indicates the size, compression algorithm used, bit depth, color type, filter, and the megapixels within the image. No metadata exists in those images that may provide additional geolocation information. The results of this analysis were confirmed by using other metadata extraction tools.

4. Pokémon GO: Forensic Analysis Tool

The findings from this research were used to create a tool that is capable of aiding an investigator in the analysis of a mobile device that contains the Pokémon GO application. This analysis tool is capable of creating an android backup of the device, and then analyzing the backup to present an overview of the forensically relevant data within the application.

The Pokémon GO: Forensic Analysis Tool was written for Python 2.7.8, and is compatible with both Windows, and Linux operating systems. The analysis tool also requires the *Android Debugging Bridge* in order to generate backups from target devices, and Google's S2 geometry library in order to convert the Cell ID's found within the Crittercism logs into physical GPS coordinates. The S2 geometry library is available on GitHub, but must be compiled to run on the analysis system. The S2 library

Peer-reviewed version available at *Information* **2017**, *8*, 71; doi:10.3390/info8030071


```
ExifTool Version Number         : 10.46
File Name                       : IMG_2017-02-24-15473410.png
Directory                       : C:/Users/Joshua/Desktop
File Size                       : 1904 kB
File Modification Date/Time     : 2017:02:24 15:48:05-06:00
File Access Date/Time           : 2017:03:16 11:06:28-05:00
File Creation Date/Time         : 2017:03:16 11:06:28-05:00
File Permissions                : rw-rw-rw-
File Type                       : PNG
File Type Extension             : png
MIME Type                       : image/png
Image Width                     : 864
Image Height                    : 1536
Bit Depth                       : 8
Color Type                      : RGB
Compression                     : Deflate/Inflate
Filter                          : Adaptive
Interlace                       : Noninterlaced
Image Size                      : 864x1536
Megapixels                      : 1.3
-- press RETURN --
```

Figure 1. Extracted metadata from IMG_2017-02-24-15473410.png.

```
ExifTool Version Number         : 10.46
File Name                       : IMG_2017-02-24-15513605.png
Directory                       : C:/Users/Joshua/Desktop
File Size                       : 2028 kB
File Modification Date/Time     : 2017:02:24 15:51:38-06:00
File Access Date/Time           : 2017:03:16 11:06:28-05:00
File Creation Date/Time         : 2017:03:16 11:06:28-05:00
File Permissions                : rw-rw-rw-
File Type                       : PNG
File Type Extension             : png
MIME Type                       : image/png
Image Width                     : 864
Image Height                    : 1536
Bit Depth                       : 8
Color Type                      : RGB
Compression                     : Deflate/Inflate
Filter                          : Adaptive
Interlace                       : Noninterlaced
Image Size                      : 864x1536
Megapixels                      : 1.3
-- press RETURN --
```

Figure 2. Extracted metadata from IMG_2017-02-24-15513605.png.

requires the following dependencies: development version of OpenSSL; CMake; and SWIG. Our analysis tool also requires an internet connection in order to generate the maps containing geolocation information.

When the analysis application is launched, the investigator is presented with a welcome screen that provides them with a quick overview of the forensically relevant artifacts that can be found in the application, and a getting started section that covers the basic usage of the application. From the File Menu the investigator has several options to begin the analysis of a phone. This menu allows the investigator to perform the following actions:

- **Capture Pokémon GO Backup:** This obtains an ADB backup of only the Pokémon GO application from a target mobile device.
- **Capture Full Backup:** This obtains an ADB backup of the entire target mobile device.
- **Create New Analysis from Backup:** This creates a new case folder in the current working directory named after the selected android backup file. The android backup is then extracted to this directory, and the relevant forensic information is parsed from the application.
- **Open Case Folder:** This opens a case folder containing an extracted android backup file, and parses the relevant forensic information from the application.

In order to begin an analysis, the investigator must select the "Create New Analysis from Backup" option. This will prompt the investigator to select the android backup file they wish to use, and will create a case directory named after this file. The case directory contains an *apps* directory which is the uncompressed backup of the mobile device, a *maps* directory which contains all of the static maps generated by the analysis application, and the compressed tar archive that was extracted from the backup file.

Once the application has finished processing the newly created case, an "Overview" tab is created within the application as shown within Figure A1 in Appendix C. This tab contains a consolidated view of all of the forensically relevant artifacts from the *upsight.xml*, *upsight.db*, *com.nianticlabs.pokemongo.PREFS.xml*, and *com.upsight.googleadvertisingid.internal.registration.xml* files. Using the information within this overview tab the investigator is able to determine when the last time the game was actively played by examining the Session Start and Session End Time values. This "Overview" tab also contains a button to map the relative GPS coordinates obtained from the Upsight database. If the application detects the presence of the Crittercism log files, then the option to map the geolocation information within these logs will also be presented to the investigator. When the investigator elects to map this information then a new tab is created as shown in Figure A2 within Appendix C. This tab provides the investigator with the ability to alter the zoom level of the static map with the use of the "Zoom In" and "Zoom Out" buttons. Each map image is also stored in the *maps* directory of the case folder.

As shown in Figure A2 of Appendix C, when the user maps the Crittercism logs a tab will be created for both the *current_bcs* and *previous_bcs* folders. In these maps markers that originated from data in "Removing Cell" logs are represented with a red marker, while markers that originated from data in "Updating Encounter" logs are represented with a blue marker. As indicated in Section 3.4 the "Updating Encounter" log data provides a better approximation of the areas the user is likely to have played, while the "Removing Cell" logs appear to be clustered along the edges of the user's location. Furthermore, each marker has a unique label that corresponds to an entry within the text box in the upper right corner of the tab. These entries correspond to the text in the log which generated that marker.

As shown in Figure A1 and Figure A2 within Appendix C, the analysis application also provides the investigators with a file browser on the left hand side of the application that lists all files found in the application. This provides the investigator with a quick overview of all the files in the application, and allows the investigator to examine the timestamps of certain files such as those contained within the *bundles* directory that may indicate when the game was actively played prior to the current session. This file browser also provides the investigator with the ability to open a hexdump of any file by

double clicking on the file in the file browser. This will open a new tab that displays the raw contents of the file. This allows the application to be more versatile in the event that artifacts within the Pokémon GO application are altered, or additional artifacts are discovered that contain forensically relevant information.

The Pokémon GO: Forensic Analysis Tool has a built in event viewer in the lower right of the application. This event viewer records all actions taken by the investigator as they conduct their analysis, and reports any problems that may arise. In addition to each log entry being recorded within the log viewer window, the logs are written to a text file called *activityLog.txt* which resides in the same directory as the analysis application.

5. Conclusion

Through this in depth analysis of the Pokémon GO application several forensically useful artifacts were discovered. We believe these findings could help an investigator develop a timeline of application use that could indicate that a user was distracted at the time of an incident, or could place the user around a particular location at a specific time. Most of these artifacts are created by the Upsight marketing and analytics platform, and can be found in the *upsight.xml*, and *upsight.db* files. By examining the data contained in the *upsight.xml* file it is possible to determine the start time, end time, and duration of the most recent session. This provides evidence of when the game was last actively run in the foreground of the user's device. Furthermore, utilizing the location data contained within the *upsight.db* file it is possible to determine the last known relative location of the user during the most recent session. Because these artifacts are generated by the Upsight platform, it may be possible to apply the knowledge learned about the behavior of these artifacts to other applications that use this platform.

Game activity outside of the most recent session could potentially be determined by analyzing the timestamps of the files contained within the bundles directory. These files consist of Unity 3D models that are downloaded dynamically during gameplay from the server. Therefore, the timestamps from these files correspond to when new pokémon or items are encountered for the first time, or if an update has occurred and the pokémon or item animations/graphics have been altered. Furthermore, the *lastPushTokeRegistrationTime* value could also be used as an indicator of game activity. Finally, the pokémon trainer's "journal" can be used to indicate recent game activity including pokémon encounters, and visits to pokéstops.

The use of a Pokémon GO Plus accessory can be determined by the presence of the *pgp.xml* file within the *sp* directory in the application. This artifact contains the Bluetooth MAC address of the Pokémon GO Plus accessory, and the Bluetooth encryption key. Using the Bluetooth MAC address it is possible to determine which Pokémon GO Plus accessory was connected to the phone, as this should be unique for each device. This research has determined that the use of the Pokémon GO Plus accessory does not have any effects on the session information generated by the Upsight platform, or on the timestamps of the files within the bundles directory. In other words, provided the application was not brought back into the foreground of the mobile device, the Pokémon GO Plus accessory will not cause the session or geolocation information recorded by the Upsight platform to be updated. Furthermore, any new "bundles" required will be downloaded when the application is brought back into the foreground of the mobile device. Other forensically relevant information includes the email address of the account holder which can be found within the *accountName* value.

The results of this research were utilized to create an analysis application that can assist an investigator by parsing the relevant information from the Pokémon GO application files, and present it to the investigator in an easy to read format. The analysis application provides the investigator with all of the session information contained within the *upsight.xml* file, the email address associated with the user's account, and an easy means to map any geolocation information discovered within the application. The data that this application was able to retrieve was compared to the data presented to the investigator by Cellebrite's UFED Physical Analyzer. In comparison, the only information that was

automatically presented to the investigator with Physical Analyzer was the email address associated with the user account. The session information contained within the *upsight.xml* file was not included within the time created by Physical Analyzer. Furthermore, the geolocation information in the upsight database was not presented to the investigator either.

The artifacts contained in the Pokémon GO application should periodically be reanalyzed as the application is still under development and new features and bug fixes are being introduced with each update. These updates could change the artifacts contained within the application, or present new artifacts that provide additional information. Future development within the analysis application should focus on the creation of a logging and reporting system that is specific to each case. This would provide the investigator with an easy means to get information out of the analysis application. Additional work should focus on the creation of fully dynamic maps, the creation of additional file viewing options, and adding file content searching capabilities.

Peer-reviewed version available at Information 2017, 8, 71; doi:10.3390/info8030071

Appendix A Actions taken prior to each dump of the application and their purposes

Data Dump Name	Activity	Notes
Dump1	**Start:** February 24, 2017 0:41 AM Application required update and restart.	Updated to version 0.57.2Updated to version 0.57.2
Dump2	**Start:** February 24, 2017 1:36 PM **Location:** Apartment **Activity:** • Captured Pidgey @ 1:38 PM • Level 3 • Captured Pikachu @ 1:42 PM **Terminated Application:** February 24, 2017 1:42 PM	This dump is designed to act as a baseline image for the application.
Dump3	**Start:** February 24, 2017 2:32 PM **Location:** Catholic Student Center **Activity:** • Captured Hoppip @ 2:35 PM • Level 4 • Visited 4 pokéstops around St. Thomas **Terminated Application:** February 24, 2017 2:40 PM	Compared to the previous dump, the starting, and ending location for Dump3 has changed substantially. A few pokémon were captured, and several pokéstops were visited to help determine the creation of new items within the bundles directory.
Dump4	**Start:** February 24, 2017 2:45 PM **Location:** Catholic Student Center **Activity:** • Captured Sentret @ 2:47 PM • Captured Spinarak @ 2:48 PM • Captured Murkrow @ 2:55 PM • Level 5 • Captured Natu @ 3:03 PM • Captured Totodile @ 3:08 PM • Captured Wooper @ 3:09 PM • Captured Caterpie @ 3:11 PM **Backgrounded App:** Phone call @ 3:14 PM **Resumed App:** 3:30 PM • Level 6 **Backgrounded App:** Locked screen @ 3:33 PM in front of Library **Resumed App:** 3:38 PM in triangle garden with Swing Statue • Captured Jiggly Puff @ 3:41 • Picture with app camera at intramural fields @ 3:47 PM • Picture with app camera at Bowers Stadium @ 3:51 • Captured Spearow @ 3:52 PM **Terminated Application:** February 24, 2017 3:57 PM **Location:** Don Sanders Stadium	This dump covers a large amount of ground, and backgrounds the application multiple times to help determine how the Upsight platform handles the creation of session information. The starting location for this dump also varies significantly from the location in which the application was terminated. **Ending Statistics:** Pokémon in Possession: 56 Pokémon Eggs: 7 Items: 189 Caught Pokémon: 18 Seen Pokémon: 31

Peer-reviewed version available at Information 2017, 8, 71; doi:10.3390/info8030071

Dump5	**Start:** February 24, 2017 4:23 PM **Location:** Catholic Student Center **Activity:** • Transferred 31 pokémon • Visit pokéstop **Terminated Application:** February 24, 2017 4:29	This dump takes place at the Catholic Student Center. The only actions that occurred was the transfer of 31 pokémon, and a pokéstop visit. This is to help determine how the pokémon count and item count variables are updated.
Dump6	**Start:** February 24, 2017 4:33 PM **Location:** Catholic Student Center **Activity:** • Captured Kakunna @ 4:35 PM • Captured Staryu @ 4:36 PM **Backgrounded App:** February 24, 2017 4:38 PM **Acquired Backup of App (Still running in background):** February 24, 2017 4:41 PM **Application restarted.** • Visit Pokéstop **Terminated Application:** February 24, 2017 4:42 PM	This dump also takes place at the Catholic Student Center. A few pokémon were captured, and then the application was backgrounded and a backup was taken. This caused the application to restart. **Ending Statistics:** Pokémon in Possession: 30 Pokémon Eggs: 8 Items: 203 Caught Pokémon: 20 Seen Pokémon: 33
Dump7	**Start:** February 24, 2017 4:47 PM **Location:** Catholic Student Center then drove to Walmart **Activity:** • Captured Marill @ 4:55 PM • Level 7 • Captured Horsea @ 4:57 PM **Backgrounded App:** February 24, 2017 4:57 PM at Walmart **Terminated Application:** February 24, 2017 5:19 at Apartment	This dump was started at the Catholic Student Center. The application was utilized while driving to Walmart. The application was then backgrounded at Walmart. The application was then terminated at the Apartment. This was designed to help determine how the Upsight platform logs geolocation information, and generates session information.
Dump8	**Start:** February 25, 2017 3:40 PM **Terminated Application:** February 25, 2017 3:40 PM	During this dump the application was started, and then quickly exited after it finished loading. This is to help determine how the Upsight platform creates session information.
Dump9	**Start:** March 19, 2017 4:14 PM Connected Pokémon Go Plus device **Terminated Application:** March 19, 2017 4:15 PM	This dump was created to act as a baseline for testing the Pokémon GO Plus device.
Dump10	**Start:** March 19, 2017 4:43 PM **Location:** Apartment Connected Pokémon GO Plus device **Backgrounded App:** 4:45 PM (locked phone) Went driving around campus. Visited numerous pokéstops and caught several pokémon with Pokémon GO Plus **Reconnected Pokémon GO Plus:** 4:50 PM **Terminated Application:** March 19, 2017 5:13 PM after successful catch. **Location:** One-way street by Library	This dump focuses on determining the effect that using the Pokémon Go Plus device has on the session information generated by the Upsight platform. The starting and ending locations for the data acquisition very significantly. The application briefly opened before being terminated, as a result of the screen being locked while still running the application.

Dump11	**Start:** March 19, 2017 7:09 PM **Location:** Apartment Connected Pokémon GO Plus **Backgrounded App:** March 19, 2017 7:12 PM Caught numerous Pokémon at apartment with Pokémon GO Plus. **Terminated Application:** March 19, 2017 9:30 PM	This dump is focused on determining the effect that using the Pokémon Go Plus device has on the timestamps of files within the bundles directory. There was no change in the starting and ending location of this dump.
Dump12	**Start:** April 13, 2017 3:20 PM **Location:** Apartment Connected Pokémon Go Plus device. **Backgrounded App:** April 13, 2017 3:21 PM **Activity:** • Captured Pokémon @ 3:35 PM • Captured Pokémon @ 3:37 PM • Captured Pokémon @ 3:42 PM **Terminated Application:** April 13, 2017 3:53 PM	This dump was focused on determining the effect that using the Pokémon GO Plus device has on the Upsight session information. All captured pokémon were recorded

Appendix B Pokémon Go Targeted Data Analysis

Table A2. Comparing forensically important values for application dumps 1-4

Item	Dump1	Dump2	Dump3	Dump4
Session Number	3	4	5	8
Current Session Duration	0	283	491	1118
Session Start Timestamp	2/24/2017, 10:41:40 AM	2/24/2017, 1:36:58 PM	2/24/2017, 2:32:14	2/24/2017, 3:38:19 PM
Last Know Session Time	2/24/2017, 10:41:40 AM	2/24/2017, 1:41:41 PM	2/24/2017, 2:40:25 PM	2/24/2017, 3:56:57 PM
Sequence ID	46	69	95	265
Install Timestamp	1/17/2017, 1:34:00 PM	1/17/2017, 1:34:00 PM	1/17/2017, 1:34:00 PM	1/17/2017, 1:34:00 PM
SID	9249052116809215539	9249052116809215539	9249052116809215539	9249052116809215539
Player XP	2920	5270	7190	18545
Player Avatar	0	0	0	0
Item Count	54	59	64	90
Pokémon Count	0	3	7	12
Player Level	2	3	4	6
Past Session Time	1364	1364	1647	4056
Upsight DB Cords		30.72, -95.56	30.72, -95.55	30.71, -95.54
lastPushTokenRegistrationTime	2/24/2017, 10:41:40 AM	2/24/2017, 10:41:40 AM	2/24/2017, 10:41:40 AM	2/24/2017, 10:41:40 AM

Peer-reviewed version available at Information 2017, 8, 71; doi:10.3390/info8030071

Table A3. Comparing forensically important values for application dumps 5-8

Item	Dump5	Dump6	Dump7	Dump8
Session Number	9	10	12	13
Current Session Duration	310	289	609	20
Session Start Timestamp	2/24/2017, 4:23:56 PM	2/24/2017, 4:33:26 PM	2/24/2017, 4:47:29 PM	2/25/2017, 3:40:21 PM
Last Know Session Time	2/24/2017, 4:29:06 PM	2/24/2017, 4:38:15 PM	2/24/2017, 4:57:39 PM	2/25/2017, 3:40:41 PM
Sequence ID	309	335	377	385
Install Timestamp	1/17/2017, 1:34:00 PM	1/17/2017, 1:34:00 PM	1/17/2017, 1:34:00 PM	1/17/2017, 1:34:00 PM
SID	9249052116809215539	9249052116809215539	9249052116809215539	9249052116809215539
Player XP	18595	20345	22045	22045
Player Avatar	0	0	0	0
Item Count	189	193	206	232
Pokémon Count	56	25	31	36
Player Level	6	6	7	7
Past Session Time	5174	5484	5827	6436
Upsight DB Cords	30.72, -95.55	30.72, -95.55	30.71, -95.57	30.72, -95.56
lastPushTokenRegistrationTime	2/24/2017, 10:41:40 AM	2/24/2017, 10:41:40 AM	2/24/2017, 10:41:40 AM	2/24/2017, 10:41:40 AM

Table A4. Comparing forensically important values for application dumps 8-12

Item	Dump9	Dump10	Dump11	Dump12
Session Number	18	21	22	28
Current Session Duration	95	2	143	68
Session Start Timestamp	3/19/2017, 4:14:00 PM	3/19/2017, 5:12:04 PM	3/19/2017, 7:09:51 PM	4/13/2017, 3:20:14 PM
Last Know Session Time	3/19/2017, 5:12:06 PM	3/19/2017, 5:12:06 PM	3/19/2017, 7:12:15 PM	4/13/2017, 3:21:22 PM
Sequence ID	435	454	471	530
Install Timestamp	1/17/2017, 1:34:00 PM	1/17/2017, 1:34:00 PM	1/17/2017, 1:34:00 PM	1/17/2017, 1:34:00 PM
SID	9249052116809215539	9249052116809215539	9249052116809215539	9249052116809215539
Player XP	23375	30800	32080	38335
Player Avatar	0	0	0	0
Item Count	225	224	260	249
Pokémon Count	41	42	63	72
Player Level	7	8	8	9
Past Session Time	6749	6996	6998	7542
Upsight DB Cords	30.72, -95.56	30.72., -95.55	30.72, -95.56	30.72, -95.56
lastPushTokenRegistrationTime	3/16/2017, 11:04:15 AM	3/16/2017, 11:04:15 AM	3/16/2017, 11:04:15 AM	4/7/2017, 3:51:21 PM

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 May 2017 doi:10.20944/preprints201705.0199.v1

Peer-reviewed version available at Information 2017, 8, 71; doi:10.3390/info8030071

Appendix C Pokémon GO: Forensic Analysis Tool

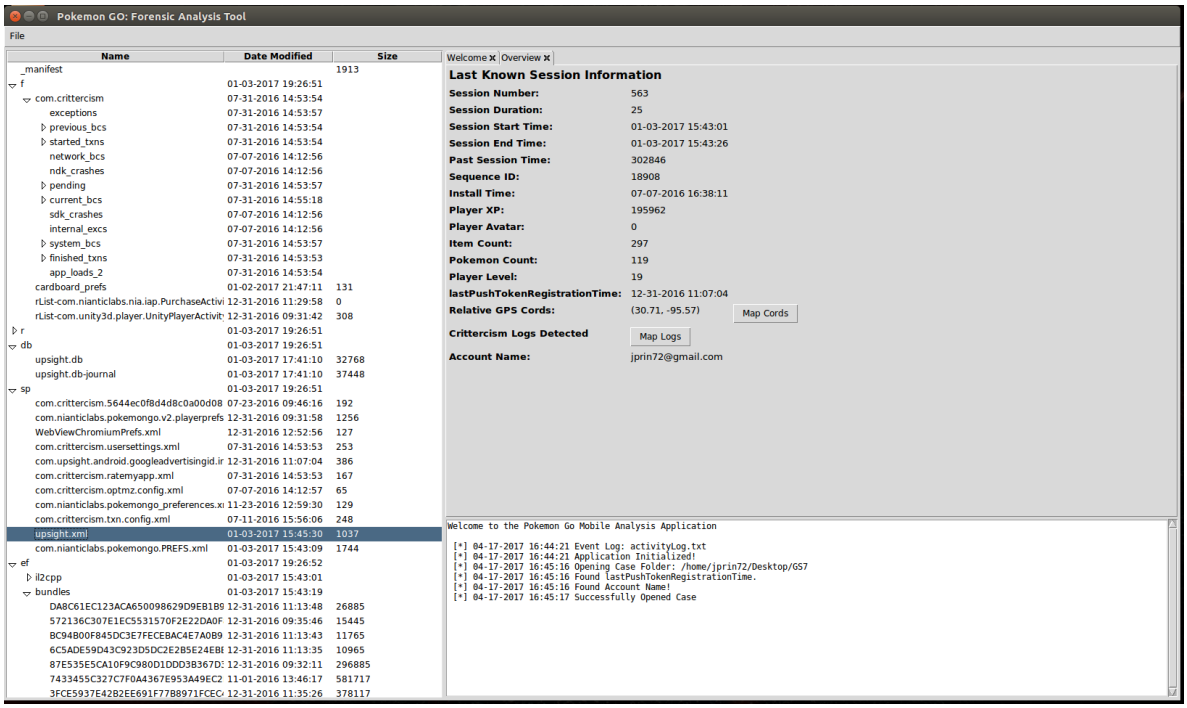

Figure A1. Pokémon GO: Overview.

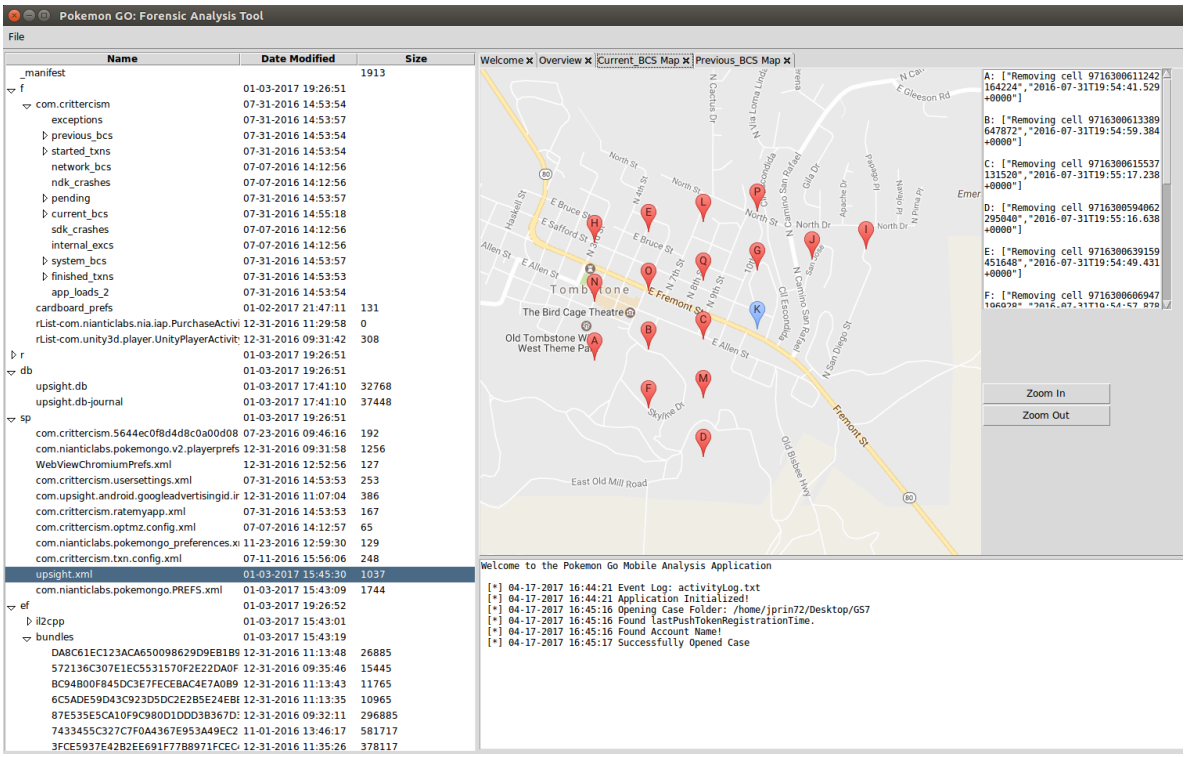

Figure A2. Pokémon GO: Map Crittercism Logs.

References

1. PokémonGO. Available online: http://pokemongo.nianticlabs.com/en/. (Accessed 9 April 2017)
2. Schwartz, J. Pokémon GO Compared to Other Popular Apps. Available online: https://www.similarweb.com/blog/pokemon-go-compared. (accessed 9 April 2017).
3. Schwartz, J. Pokémon GO: The Data Behind America's Latest Obsession. Available online: https://www.similarweb.com/blog/pokemon-go. (accessed on 9 April 2017).
4. 2017 U.S. Cross-Platform Future in Focus. Available online: http://www.comscore.com/Insights/Presentations-and-Whitepaper (accessed on 9 April 2017).
5. Thier, D. Firm: 'Pokémon GO' Has Made $1 Billion Since Launch. Available online: https://www.forbes.com/sites/davidthier/2017/01/31/firm-pokemon-go-has-made-1-billion-since-launch/#433f79ba20e3. (accessed 9 April 2017).
6. Weinberger, M. The fad may be over, but Pokémon GO still has 65 million monthly active players. Available online: http://www.businessinsider.com/pokemon-go-65-million-monthly-active-players-2017-4. (accessed on 9 April 2017).
7. Althoff, T.; White, R.W.; Horvitz, E. Influence of Pokémon GO on physical activity: Study and Implications. *Journal of Medical Internet Research* **2016**, vol. 18, no. 12.
8. Smith, D. Hundreds of people mobbed Central Park to catch a Vaporeon in Pokémon GO. Available online: http://www.businessinsider.com/pokemon-go-mob-runs-after-vaporeon-video-2016-7. (accessed on 9 April 2017).
9. Gilbert, B. This 40-second video will convince you that Pokémon GO is an insane phenomenon. Available online: http://www.businessinsider.com/pokemon-go-mob-runs-after-squirtle-video-2016-7. (accessed on 9 April 2017).
10. Bowerman, M. Woman discovers body while playing 'Pokémon Go'. [Online]. Available online: https://www.usatoday.com/story/tech/nation-now/2016/07/11/woman-playing-pokemon-go-discovers-dead-body-river-pla (accessed on 9 April 2017).

11. Daye, A. Pokémon Go helps Marines to catch suspect. Available online: http://www.cnn.com/2016/07/13/us/pokmon-go-helps-marines-to-catch-suspect/. (accessed on 9 April 2017).
12. Miller, R. Teens used Pokémon Go app to lure robbery victims, police say. Available online: https://www.usatoday.com/story/tech/2016/07/10/four-suspects-arrested-string-pokemon-go-related-armed-robberies/8692 (accessed on 9 April 2017).
13. Military base issues 'Pokémon GO' warning. Available online: http://www.foxnews.com/tech/2016/07/19/military-base-issues-pokemon-go-warning.html. (accessed on 9 April 2017).
14. Roberts, A. 'Pokémon Go' Is A Major Distraction For Drivers According To A New Study. Available online: http://uproxx.com/gaming/pokemon-go-too-distracting-new-study/. (accessed on 9 April 2017).
15. Lahman, S. Pokémon Go player crashes his car into a tree. Available online: https://www.usatoday.com/story/news/nation/2016/07/14/pokmon-go-player-crashes-his-car-into-tree/87074762/. (accessed on 9 April 2017).
16. Birch, N. This Distracted 'Pokémon Go' Player Crashed Into A Police Car On Camera. Available online: http://uproxx.com/gammasquad/pokemon-go-police-car/. (accessed on 9 April 2017).
17. Pokémon Go player crashes car into school while playing game. Available online: https://www.theguardian.com/australia-news/2016/jul/29/pokemon-go-player-crashes-car-into-school-while-playing-game. (accessed on 9 April 2017).
18. Inada, M. 'Pokémon Go' - Related Car Crash Kills Woman in Japan. Available online: https://www.wsj.com/articles/woman-killed-in-pokemon-go-related-car-crash-in-japan-1472107854. (accessed on 9 April 2017).
19. Maus, S.; Hofken, H.; Schuba, M. Forensic analysis of geodata in android smartphones. *Internation Conference on Cybercrime, Security and Digital Forensics* **2011**.
20. Murphy, C. A Sneak Peek at Pokemon Go Application Forensics. Available online: https://digital-forensics.sans.org/blog/2016/08/09/a-sneak-peek-at-pokemon-go-application-forensics. (accessed on 9 April 2017).
21. Hilbert, D. On the continuous representation of a line on a surface part. *Mathematical Annals* 38.3 (1891): 459-460.
22. Call me Ash Ketchum: Open Source Forensics with Pokemon Go. Available online: https://www.security-sleuth.com/sleuth-blog/2016/8/13/call-me-ash-ketchum-open-source-forensics-with-pokemon-go. (accessed on 9 April 2017).
23. Head, N. Pokémon Go: An Introductory Forensic Study. Available online: https://www.intaforensics.com/2016/08/05/pokemon-go-an-introductory-forensic-study/. (accessed on 9 April 2017).
24. Lawson, V. A Forensic Examination of Pokémon Go. *ProQuest Dissertations & Theses* **2016**. Utica, New York.