Forensics & Malware Analysis

Dee Dixon: dxd946@student.bham.ac.uk
Kyle MacQueen: knm986@student.bham.ac.uk

Case No: 001
Case Scope: 18th February 2020 to 24th February 2020
SHA256 Initial Hash: 48152f0d86c78c7d80479b714fd48d3464cd944c3d6b5a08af7d28b368a58e1d
SHA256 Final Hash: 48152f0d86c78c7d80479b714fd48d3464cd944c3d6b5a08af7d28b368a58e1d

Part 1: Picture documentation

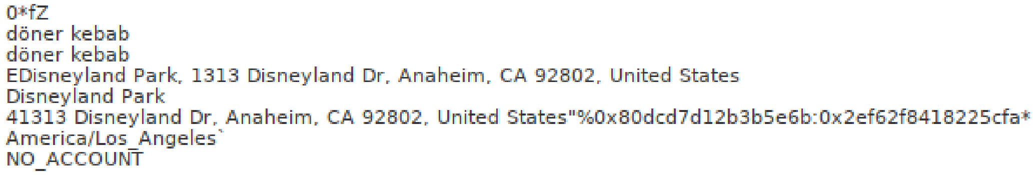
Due to the high volume of pictures recovered, some considered non-essential will be grouped together or omitted.

Name	Brief description	Timestamp	Geolocation
productNew images	Group of app icon images	2020-02-18 15:16:33 –15:16:36	N/A
homescreenPreview.png	Screenshot of home screen	2020-02-18 11:16:58	N/A
lockscreen_capture_port .png	Screenshot of lock screen	2020-02-18 15:16:06	N/A
Screenshot_20200218-152721.png	Screenshot	2020-02-18 15:27:23	N/A
Screenshot_20200218-152336.jpg	Edited screenshot of Love Island Google Search	2020-02-18 15:23:36	N/A
Screenshot_20200218-152724.png	Screenshot of [Code Art] Fairy Tale	2020-02-18 15:27:24	N/A
Screenshot_20200218-171729.png	Screenshot of YouTube video (ABO channel)	2020-02-18 17:17:30	N/A
Screenshot_20200218-171734.png	Screenshot of YouTube video (ABO channel)	2020-02-18 17:17:34	N/A
Screenshot_20200221-145547.png	Screenshot of Fortnite Hack Aimbot + ESP	2020-02-21 14:55:47	N/A
Thumbnail images	Reduced versions of already stored images	2020-02-18 15:23:45 to 2020-02-20 14:23:10	N/A
20200218-171304.jpg	Flowers in a vase	2020-02-18 17:13:04	N/A
20200220_115211.jpg	Cold meats / cheese section of shop	2020-02-20 11:52:11	N/A
20200220_115524.jpg	Sausage Rolls (from Spar)	2020-02-20 11:55:24	N/A
Photo_1567201080580-bfcc97dae346.jpg	Downloaded image of Pig	2020-02-18 15:31:14	N/A
Whatsapp Media	WhatsApp features / how to use statuses	2020-02-20 11:42:56 – 11:42:57	N/A

IMG-20200222-WA0000.jpg	Outside of Indian Restaurant Mr. Idly	2020-02-22 23:58:50	N/A
Whatsapp Profile pictures (447458937615.jpg)	Profile picture (lady with glasses)	2020-02-21 14:36:28	N/A
128_task_thumbnail.png	Google search for Donor Kebab	2020-02-21 14:46:36	N/A
Samsung theme overlays	Preview images of Samsung theme	2018-01-01 00:05:47	N/A
Clipboard images	Images that were captured from screenshots. Shown above	2020-02-18 14:23:13 – 2020-02-21 14:55:49	N/A
Google Map cache images	Shows Disneyland Castle and German Donor Kebab	2020-02-18 14:47:30	N/A

Most of the images found on the device are thumbnails and small application icon images. A lot of images are stored more than once, through the clipboard (screenshots) and through thumbnails (smaller image used for indexing/organising). WhatsApp was used to download a few emoticon images and filters. Duplicate images are stored in in the clipboard. These images are stored in the clipboard because they were captured from screenshots. Screenshots of an image are automatically stored in clipboard for easy copying and pasting, to WhatsApp for example. While exploring the filesystem for more data, we have also come across images that have been stored in cache. These images all show a restaurant, German Donor Kebab and the Disneyland Castle. We also attempted to use geolocation plugins but also had no luck with finding location data. Two videos were also found, one is a video we believe is linked to WhatsApp as it states the user can now edit images, the other shows a gif of Billie Eilish pointing and laughing.

File Directory used: */full.img/data/com.google.android.apps.maps/files*
File Names used: *new_recent_history_cache_search.cs*



```
0*fZ
döner kebab
döner kebab
EDisneyland Park, 1313 Disneyland Dr, Anaheim, CA 92802, United States
Disneyland Park
41313 Disneyland Dr, Anaheim, CA 92802, United States"%0x80dcd7d12b3b5e6b:0x2ef62f8418225cfa*
America/Los_Angeles`
NO_ACCOUNT
```

This screenshot shows the search history within Google Maps. We can see the user searched for kebabs and Disneyland. We have also found images that match this search history in the cache for the map application. We can assume that the phone has stored these images under cache for use later. We can safely assume these images were not saved by the user because there is no history of this in downloads. We believe that these searches were carried out in order to add noise to the phone image to try and deceive what the phone was used for.

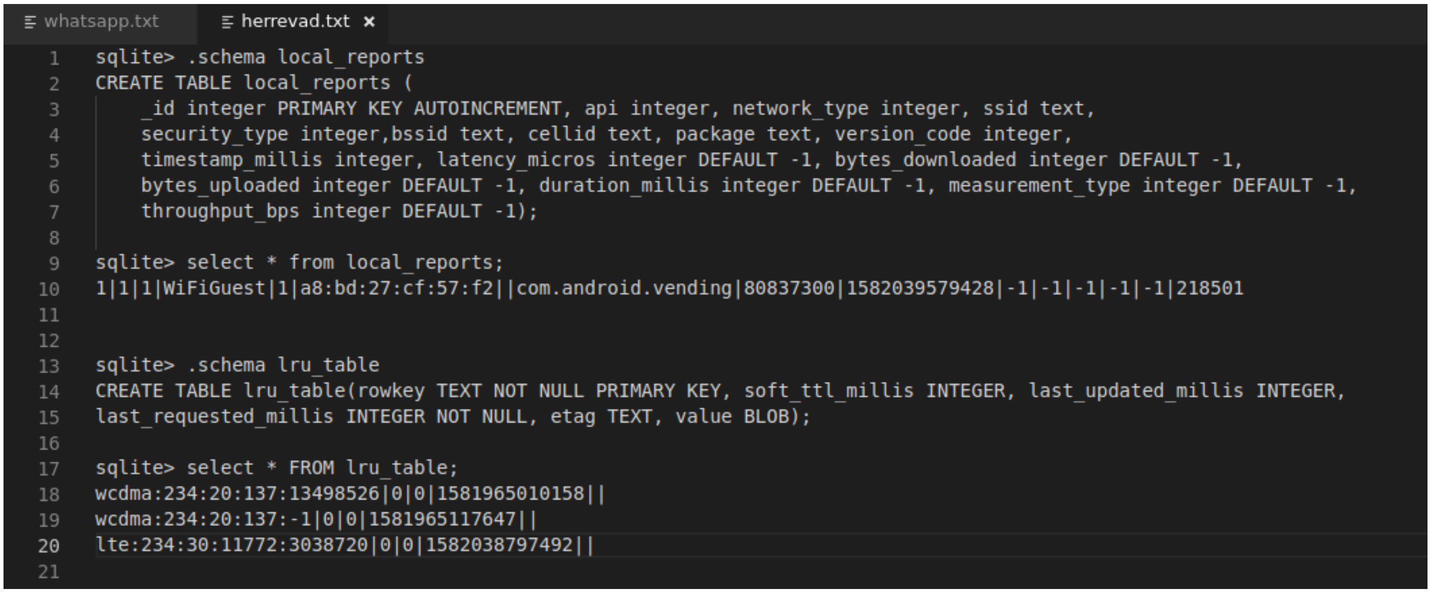
Part 2: Wireless documentation

File Directory used: */full.img/misc/wifi*
File Names used: *networkHistory.txt, default_ap.conf, p2p_supplicant.conf*

Wireless Name (BSSID)	Authentication	Connection Type	Connected (Y/N)
ASK4 Wireless (802.1x)	WPA-EAP	Wireless	N
WifiGuest	None	Wireless	Y
ASK4 Wireless	None	Wireless	Y
Wifi Extra/O2 Wifi/ O2 Wifi.1x/ Wifi +	Unsure	SIM/Mobile	Unsure
BRUH	Unsure	P2P	Unsure

From personal experience we know that *WifiGuest* is the extended university wireless channel for Guests, where a logon page pops up. *ASK4 Wireless* is linked to the accommodations of the university, particularly those within the Vale. Wifi Extra and its many other names are services linked to mobile connections as part of the SIM deal on the phone, all can be used as the Pay as you go option or the monthly allowance. Lastly BRUH seems to be a device that the phone connected to via Peer-to-Peer, possibly to transfer over data, more research is needed.

File Directory used: */full.img/data/com.google.android.gms/databases*
File Names used: *herrevad.db, herrevad.db-journal*

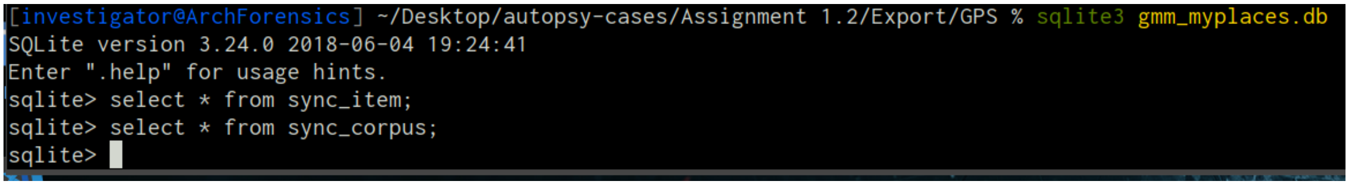


```
sqlite> .schema local_reports
CREATE TABLE local_reports (
    _id integer PRIMARY KEY AUTOINCREMENT, api integer, network_type integer, ssid text,
    security_type integer,bssid text, cellid text, package text, version_code integer,
    timestamp_millis integer, latency_micros integer DEFAULT -1, bytes_downloaded integer DEFAULT -1,
    bytes_uploaded integer DEFAULT -1, duration_millis integer DEFAULT -1, measurement_type integer DEFAULT -1,
    throughput_bps integer DEFAULT -1);

sqlite> select * from local_reports;
1|1|1|WiFiGuest|1|a8:bd:27:cf:57:f2||com.android.vending|80837300|1582039579428|-1|-1|-1|-1|-1|218501


sqlite> .schema lru_table
CREATE TABLE lru_table(rowkey TEXT NOT NULL PRIMARY KEY, soft_ttl_millis INTEGER, last_updated_millis INTEGER,
last_requested_millis INTEGER NOT NULL, etag TEXT, value BLOB);

sqlite> select * FROM lru_table;
wcdma:234:20:137:13498526|0|0|1581965010158||
wcdma:234:20:137:-1|0|0|1581965117647||
lte:234:30:11772:3038720|0|0|1582038797492||
```

Herrevad confirms the use of WifiGuest with the data shown above in the local report table. From discussion with other forensics operators we believe that our local report table is a lot emptier than would normally be expected. We were interested in the data within *lru_table*, although research into the table itself found nothing the *rowkeys* start with *wcdma* and *lte*, both denoting mobile connection types.

Part 3: Location documentation

File Directory used: */full.img/ data/com.google.android.apps.maps/databases*
File Names used: *gmm_myplaces.db, gmm_myplaces.db-journal, gmm_storage.db, gmm_storage.db-journal, gmm_sync.db, gmm_sync.db-journal, ugc_photos_location_data.db, ugc_photos_location_data.db-journal*



```
[investigator@ArchForensics] ~/Desktop/autopsy-cases/Assignment 1.2/Export/GPS % sqlite3 gmm_myplaces.db
SQLite version 3.24.0 2018-06-04 19:24:41
Enter ".help" for usage hints.
sqlite> select * from sync_item;
sqlite> select * from sync_corpus;
sqlite> 
```

Unfortunately, we have been unable to find any location data from the device. Looking through location files and databases from google maps did not produce any positional data. Here the screenshot shows the same for all databases used, the tables are completely devoid of usable data. Looking through the database files we were unable to locate *node.db*. This is another file where location data could potentially be stored, however this file did not exist on the system. We are confident that the device's location-based services were not running due to all databases with any positional fields returning no data.

From the pictures and messages found, we can figure out some of the locations they have visited, e.g. Spar University Centre. Positional data wasn't shown in the messages database for WhatsApp. This could be because location-based services were not enabled on the phone or could be the possibility that WhatsApp was not allowing location access.

Part 4: Contacts documentation

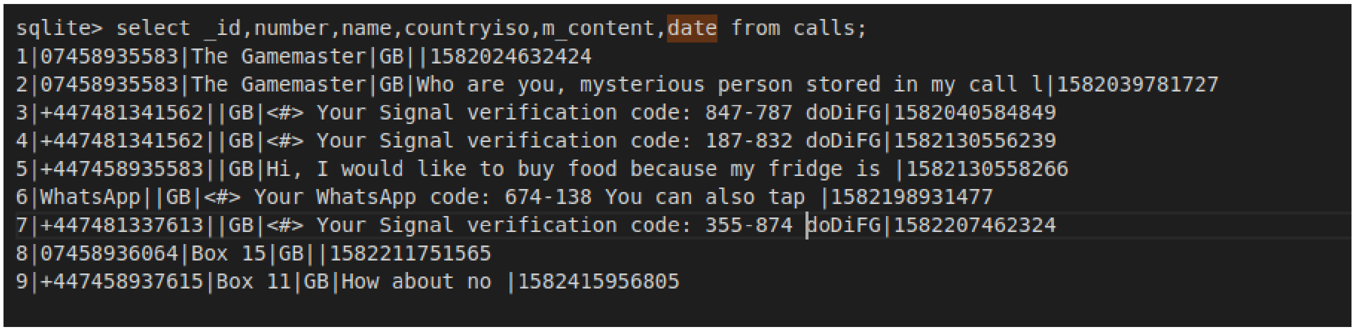
File Directory used: */full.img/data/com.android.providers.contacts/databases*
File Names used: c*ontacts2.db*

Name	Number
The Gamemaster	07458 935583
Box 2	07458 933888
Box 3	07458 937005
Box 5	07458 937882
Box 6	07458 934328
Box 7	07458 933703
Box 9	07458 936822
Box 10	07458 933830
Box 11	07458 937615
Box 13	07458 933106
Box 15	07458 936064
Box 17	07458 934348

This table shows the names of the contacts on the phone and their numbers. Below we can see messages from the call log which shows the phone receiving a message from *The Gamemaster*.

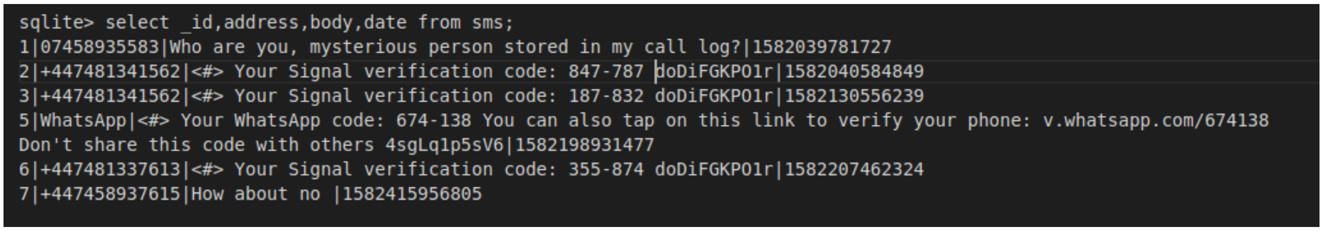
Part 5: Call and Message documentation

File Directory used: */full.img/data/com.android.providers.contacts/databases*
File Names used: *calllog.db, calllog.db-journal*



```
sqlite> select _id,number,name,countryiso,m_content,date from calls;
1|07458935583|The Gamemaster|GB||1582024632424
2|07458935583|The Gamemaster|GB|Who are you, mysterious person stored in my call l|1582039781727
3|+447481341562||GB|<#> Your Signal verification code: 847-787 doDiFG|1582040584849
4|+447481341562||GB|<#> Your Signal verification code: 187-832 doDiFG|1582130556239
5|+447458935583||GB|Hi, I would like to buy food because my fridge is |1582130558266
6|WhatsApp||GB|<#> Your WhatsApp code: 674-138 You can also tap |1582198931477
7|+447481337613||GB|<#> Your Signal verification code: 355-874 doDiFG|1582207462324
8|07458936064|Box 15|GB||1582211751565
9|+447458937615|Box 11|GB|How about no |1582415956805
```

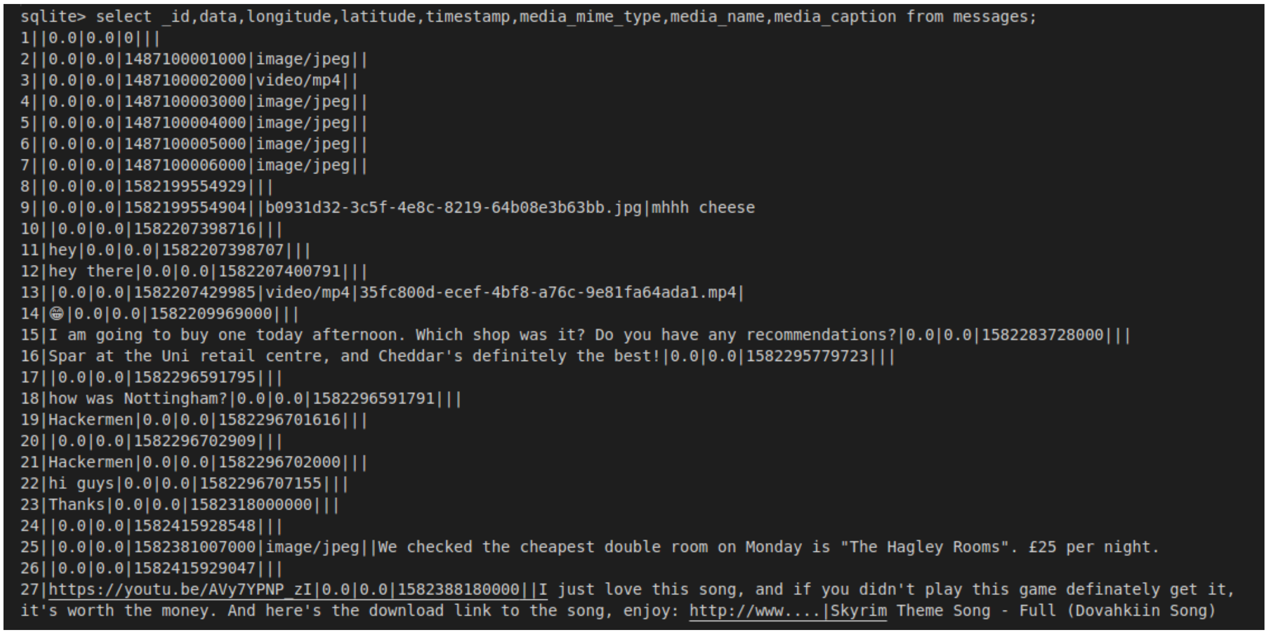
File Directory: */img_full.img/user_de/0/com.android.providers.telephony/databases/mmssms.db*
File Names: *mmssms.db, mmssms.db-wal, mmssms.db-shm*



```
sqlite> select _id,address,body,date from sms;
1|07458935583|Who are you, mysterious person stored in my call log?|1582039781727
2|+447481341562|<#> Your Signal verification code: 847-787 doDiFGKPO1r|1582040584849
3|+447481341562|<#> Your Signal verification code: 187-832 doDiFGKPO1r|1582130556239
5|WhatsApp|<#> Your WhatsApp code: 674-138 You can also tap on this link to verify your phone: v.whatsapp.com/674138
Don't share this code with others 4sgLq1p5sV6|1582198931477
6|+447481337613|<#> Your Signal verification code: 355-874 doDiFGKPO1r|1582207462324
7|+447458937615|How about no |1582415956805
```

These screenshots show the SMS messages that interacted with the phone. The data shows the number, name, country and content within these messages. We believe that message 5 from the *callog* database is the task given by the "Gamemaster" as it both discusses food and does not exist in other databases, insinuating that it was deleted. We can see that this message does not appear within the *sms* database but does appear within the *callog* database as a small snippet to show it did exist.

File Directory used: */full.img/data/com.whatsapp/databases*
File Names used: *msgstore.db, msgstore.db-wal, msgstore.db-shm*



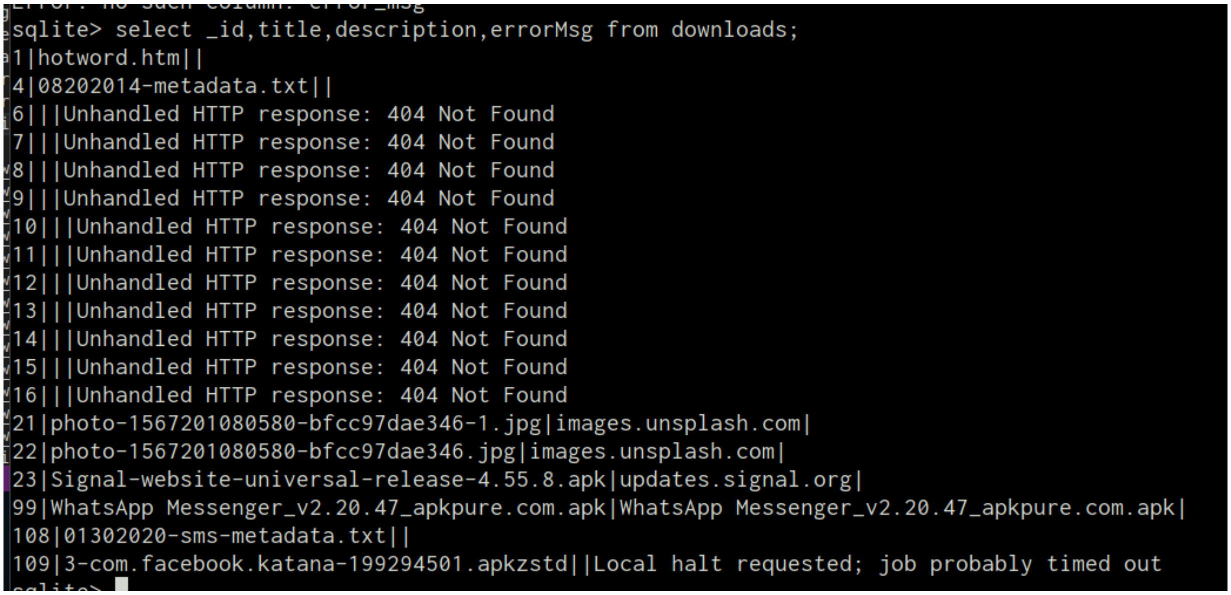
```
sqlite> select _id,data,longitude,latitude,timestamp,media_mime_type,media_name,media_caption from messages;
1||0.0|0.0|0|||
2||0.0|0.0|1487100001000|image/jpeg||
3||0.0|0.0|1487100002000|video/mp4||
4||0.0|0.0|1487100003000|image/jpeg||
5||0.0|0.0|1487100004000|image/jpeg||
6||0.0|0.0|1487100005000|image/jpeg||
7||0.0|0.0|1487100006000|image/jpeg||
8||0.0|0.0|1582199554929|||
9||0.0|0.0|1582199554904||b0931d32-3c5f-4e8c-8219-64b08e3b63bb.jpg|mhhh cheese
10||0.0|0.0|1582207398716|||
11|hey|0.0|0.0|1582207398707|||
12|hey there|0.0|0.0|1582207400791|||
13||0.0|0.0|1582207429985|video/mp4|35fc800d-ecef-4bf8-a76c-9e81fa64ada1.mp4|
14|😬|0.0|0.0|1582209969000|||
15|I am going to buy one today afternoon. Which shop was it? Do you have any recommendations?|0.0|0.0|1582283728000|||
16|Spar at the Uni retail centre, and Cheddar's definitely the best!|0.0|0.0|1582295779723|||
17||0.0|0.0|1582296591795|||
18|how was Nottingham?|0.0|0.0|1582296591791|||
19|Hackermen|0.0|0.0|1582296701616|||
20||0.0|0.0|1582296702909|||
21|Hackermen|0.0|0.0|1582296702000|||
22|hi guys|0.0|0.0|1582296707155|||
23|Thanks|0.0|0.0|1582318000000|||
24||0.0|0.0|1582415928548|||
25||0.0|0.0|1582381007000|image/jpeg||We checked the cheapest double room on Monday is "The Hagley Rooms". £25 per night.
26||0.0|0.0|1582415929047|||
27|https://youtu.be/AVy7YPNP_zI|0.0|0.0|1582388180000||I just love this song, and if you didn't play this game definately get it,
it's worth the money. And here's the download link to the song, enjoy: http://www....|Skyrim Theme Song - Full (Dovahkiin Song)
```

We were able to acquire the messages sent via WhatsApp without decrypting them, we had located the key used for encryption of the backups, so this was feasible if it had been necessary. The data unfortunately does not give the phone numbers at work but does give an insight into the messages that were occurring. There seems to be discussion about locating cheese at a store and they state that they went to the spar store and believe cheddar to be a good choice. We know this to be true due to the images we located on the device showing an image of cheese within a Spar shop. The screenshot also shows when images and videos were sent into the conversation. We have also converted the timestamps to confirm this was within the scope of our search.

We have had issues throughout this report with finding GPS and location data, we believe with the evidence above showing no longitude or latitude for any of the messages that it may be a possibility that the GPS was never turned on (or only turned on briefly) throughout the subjects having the phone in their possession.

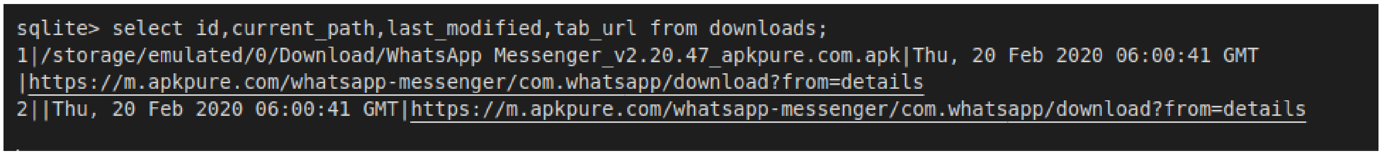
Part 6: Apps and services documentation

File Directory used: */full.img/data/com.android.providers.downloads/databases*
File Names used: *downloads.db, downloads.db-journal*



```
Error: no such column: error_msg
sqlite> select _id,title,description,errorMsg from downloads;
1|hotword.htm||
4|08202014-metadata.txt||
6|||Unhandled HTTP response: 404 Not Found
7|||Unhandled HTTP response: 404 Not Found
8|||Unhandled HTTP response: 404 Not Found
9|||Unhandled HTTP response: 404 Not Found
10|||Unhandled HTTP response: 404 Not Found
11|||Unhandled HTTP response: 404 Not Found
12|||Unhandled HTTP response: 404 Not Found
13|||Unhandled HTTP response: 404 Not Found
14|||Unhandled HTTP response: 404 Not Found
15|||Unhandled HTTP response: 404 Not Found
16|||Unhandled HTTP response: 404 Not Found
21|photo-1567201080580-bfcc97dae346-1.jpg|images.unsplash.com|
22|photo-1567201080580-bfcc97dae346.jpg|images.unsplash.com|
23|Signal-website-universal-release-4.55.8.apk|updates.signal.org|
99|WhatsApp Messenger_v2.20.47_apkpure.com.apk|WhatsApp Messenger_v2.20.47_apkpure.com.apk|
108|01302020-sms-metadata.txt||
109|3-com.facebook.katana-199294501.apkzstd||Local halt requested; job probably timed out
sqlite>
```

File Directory used: */full.img/data/com.android.app.sbrowser/app_sbrowser/Default*
File Names used: *History, History-journal*

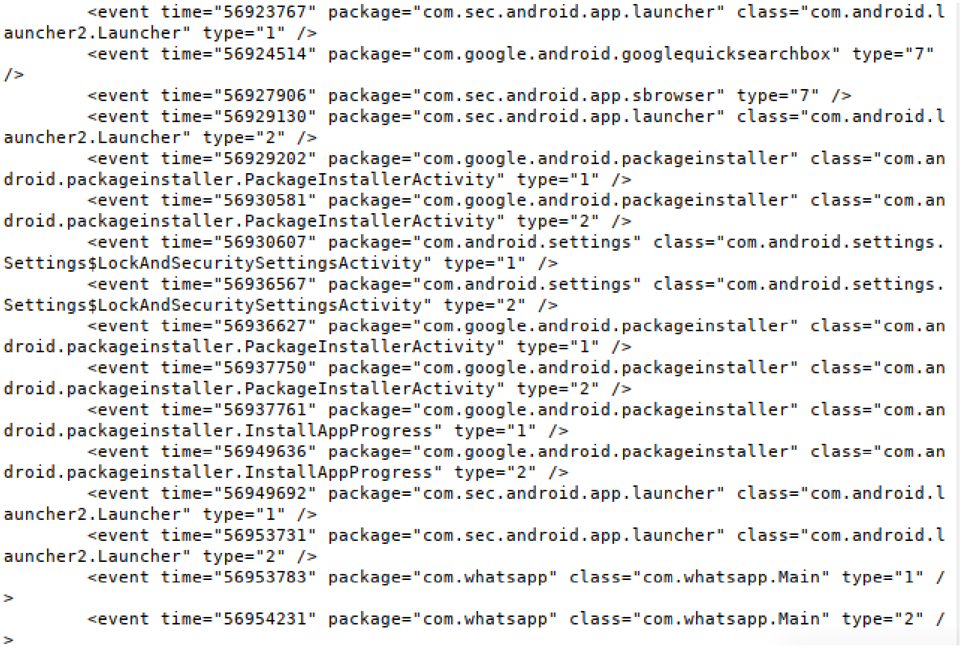


```
sqlite> select id,current_path,last_modified,tab_url from downloads;
1|/storage/emulated/0/Download/WhatsApp Messenger_v2.20.47_apkpure.com.apk|Thu, 20 Feb 2020 06:00:41 GMT
|https://m.apkpure.com/whatsapp-messenger/com.whatsapp/download?from=details
2||Thu, 20 Feb 2020 06:00:41 GMT|https://m.apkpure.com/whatsapp-messenger/com.whatsapp/download?from=details
```

Here we can see the files that were downloaded. We can see that the name of the photos downloaded matches the first table which shows the images found on the phone. These images were downloaded from *unsplash*.com. We can also see two applications have been downloaded: Signal Messenger and WhatsApp Messenger. The screenshot shows that Signal was downloaded from the applications main website and WhatsApp was downloaded from *apkpure.com*. It seems from the second screenshot that two WhatsApp downloads took place but the second one was cancelled.

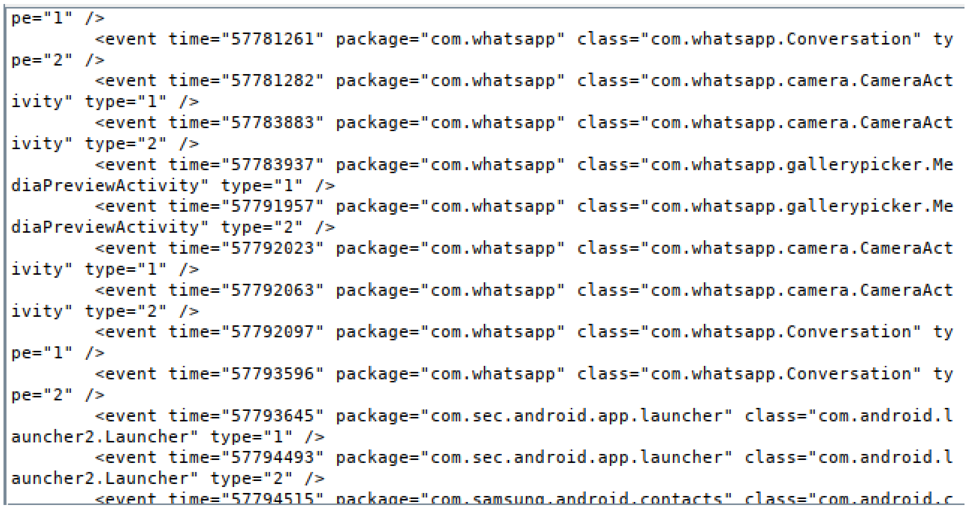
Part 7: Additional documentation

Within *system/usagestats* we were able to locate log files of the activity conducted on the phone. The phone stores log files for different time lengths: yearly, weekly, monthly and daily. The daily folder has four logs for each day the phone was used. These logs show the last active times of different services running on the phone.



```
        <event time="56923767" package="com.sec.android.app.launcher" class="com.android.l
auncher2.Launcher" type="1" />
        <event time="56924514" package="com.google.android.googlequicksearchbox" type="7"
/>
        <event time="56927906" package="com.sec.android.app.sbrowser" type="7" />
        <event time="56929130" package="com.sec.android.app.launcher" class="com.android.l
auncher2.Launcher" type="2" />
        <event time="56929202" package="com.google.android.packageinstaller" class="com.an
droid.packageinstaller.PackageInstallerActivity" type="1" />
        <event time="56930581" package="com.google.android.packageinstaller" class="com.an
droid.packageinstaller.PackageInstallerActivity" type="2" />
        <event time="56930607" package="com.android.settings" class="com.android.settings.
Settings$LockAndSecuritySettingsActivity" type="1" />
        <event time="56936567" package="com.android.settings" class="com.android.settings.
Settings$LockAndSecuritySettingsActivity" type="2" />
        <event time="56936627" package="com.google.android.packageinstaller" class="com.an
droid.packageinstaller.PackageInstallerActivity" type="1" />
        <event time="56937750" package="com.google.android.packageinstaller" class="com.an
droid.packageinstaller.PackageInstallerActivity" type="2" />
        <event time="56937761" package="com.google.android.packageinstaller" class="com.an
droid.packageinstaller.InstallAppProgress" type="1" />
        <event time="56949636" package="com.google.android.packageinstaller" class="com.an
droid.packageinstaller.InstallAppProgress" type="2" />
        <event time="56949692" package="com.sec.android.app.launcher" class="com.android.l
auncher2.Launcher" type="1" />
        <event time="56953731" package="com.sec.android.app.launcher" class="com.android.l
auncher2.Launcher" type="2" />
        <event time="56953783" package="com.whatsapp" class="com.whatsapp.Main" type="1" /
>
        <event time="56954231" package="com.whatsapp" class="com.whatsapp.Main" type="2" /
>
```

This screenshot shows what we believe to be the user installing WhatsApp. We can see the user uses the quick search box, which opens the Samsung default browser. From there we can assume the user attempts to download the APK, but then switches to change their settings to allow downloads of unknown sources. The user then installs the APK. We can see this with *android.packageinstaller.*

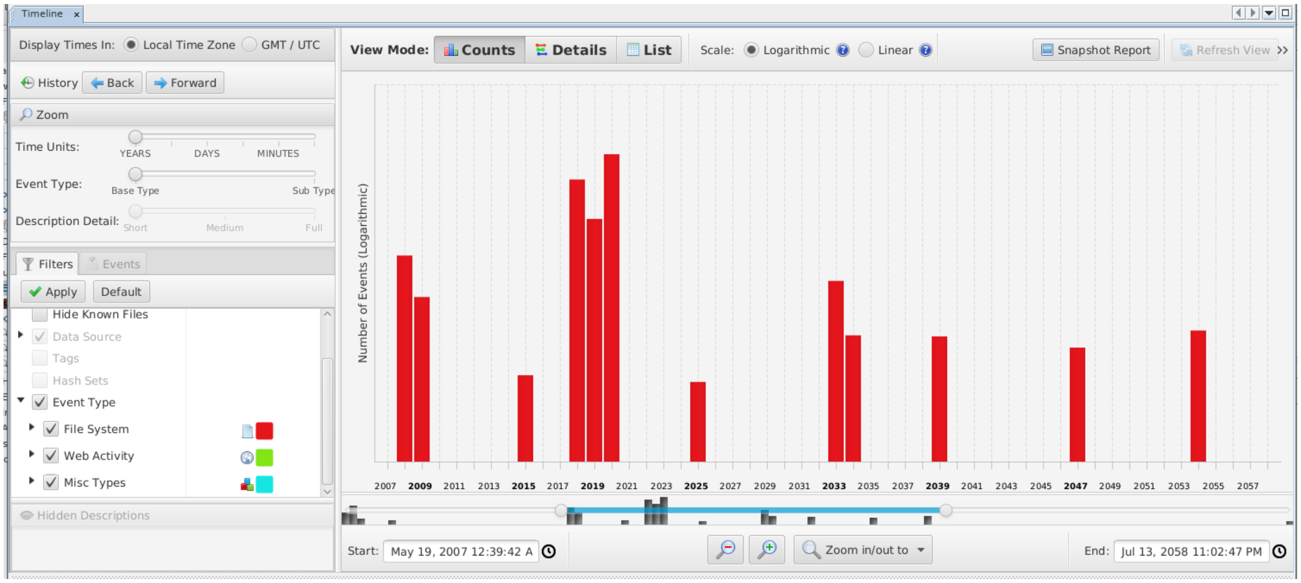


```
pe="1" />
        <event time="57781261" package="com.whatsapp" class="com.whatsapp.Conversation" ty
pe="2" />
        <event time="57781282" package="com.whatsapp" class="com.whatsapp.camera.CameraAct
ivity" type="1" />
        <event time="57783883" package="com.whatsapp" class="com.whatsapp.camera.CameraAct
ivity" type="2" />
        <event time="57783937" package="com.whatsapp" class="com.whatsapp.gallerypicker.Me
diaPreviewActivity" type="1" />
        <event time="57791957" package="com.whatsapp" class="com.whatsapp.gallerypicker.Me
diaPreviewActivity" type="2" />
        <event time="57792023" package="com.whatsapp" class="com.whatsapp.camera.CameraAct
ivity" type="1" />
        <event time="57792063" package="com.whatsapp" class="com.whatsapp.camera.CameraAct
ivity" type="2" />
        <event time="57792097" package="com.whatsapp" class="com.whatsapp.Conversation" ty
pe="1" />
        <event time="57793596" package="com.whatsapp" class="com.whatsapp.Conversation" ty
pe="2" />
        <event time="57793645" package="com.sec.android.app.launcher" class="com.android.l
auncher2.Launcher" type="1" />
        <event time="57794493" package="com.sec.android.app.launcher" class="com.android.l
auncher2.Launcher" type="2" />
        <event time="57794515" package="com.samsung.android.contacts" class="com.android.c
```

This screenshot of the log shows the user sending a picture in a WhatsApp conversation. We can see the user open the camera activity and choose an image from the gallery;

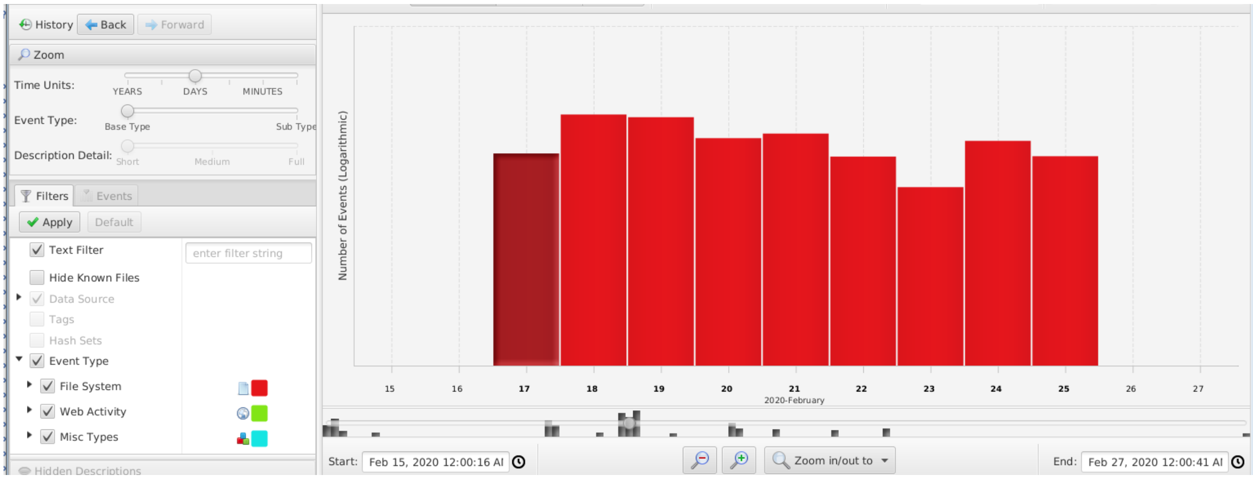
CameraActivity, gallerypicker. We can also see when the user is in the conversation. The WhatsApp messages show that a picture was sent from the user. This adds further proof to our assumptions from previously collected data about how the phone was used.

The below images show two timelines of data usage. Firstly one showing the full span of the phones data (some of which exists before the phone was ever released and after the current date) the second shows the uniform use of the device over the days the users had access to it. Whether this is just background noise and the phone was left on for the entire time, or whether the users did several things each day remains to be seen.



Start: May 19, 2007 12:39:42 A

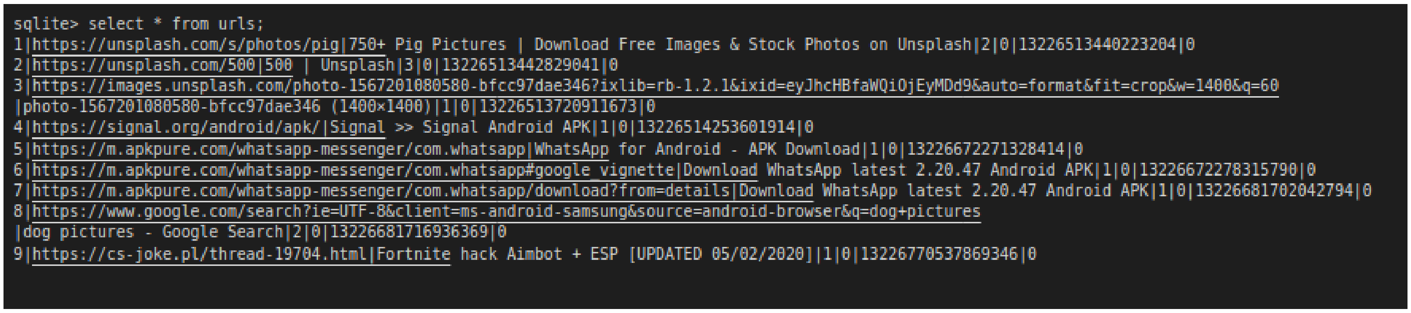
End: Jul 13, 2058 11:02:47 PM



Start: Feb 15, 2020 12:00:16 AI

End: Feb 27, 2020 12:00:41 AI

File Directory used: */full.img/data/com.android.app.sbrowser/app_sbrowser/Default*
File Names used: *History, History-journal*



```
sqlite> select * from urls;
1|https://unsplash.com/s/photos/pig|750+ Pig Pictures | Download Free Images & Stock Photos on Unsplash|2|0|13226513440223204|0
2|https://unsplash.com/500|500 | Unsplash|3|0|13226513442829041|0
3|https://images.unsplash.com/photo-1567201080580-bfcc97dae346?ixlib=rb-1.2.1&ixid=eyJhcHBfaWQiOjEyMDd9&auto=format&fit=crop&w=1400&q=60
|photo-1567201080580-bfcc97dae346 (1400×1400)|1|0|13226513720911673|0
4|https://signal.org/android/apk/|Signal >> Signal Android APK|1|0|13226514253601914|0
5|https://m.apkpure.com/whatsapp-messenger/com.whatsapp|WhatsApp for Android - APK Download|1|0|13226672271328414|0
6|https://m.apkpure.com/whatsapp-messenger/com.whatsapp#google_vignette|Download WhatsApp latest 2.20.47 Android APK|1|0|13226672278315790|0
7|https://m.apkpure.com/whatsapp-messenger/com.whatsapp/download?from=details|Download WhatsApp latest 2.20.47 Android APK|1|0|13226681702042794|0
8|https://www.google.com/search?ie=UTF-8&client=ms-android-samsung&source=android-browser&q=dog+pictures
|dog pictures - Google Search|2|0|13226681716936369|0
9|https://cs-joke.pl/thread-19704.html|Fortnite hack Aimbot + ESP [UPDATED 05/02/2020]|1|0|13226770537869346|0
```

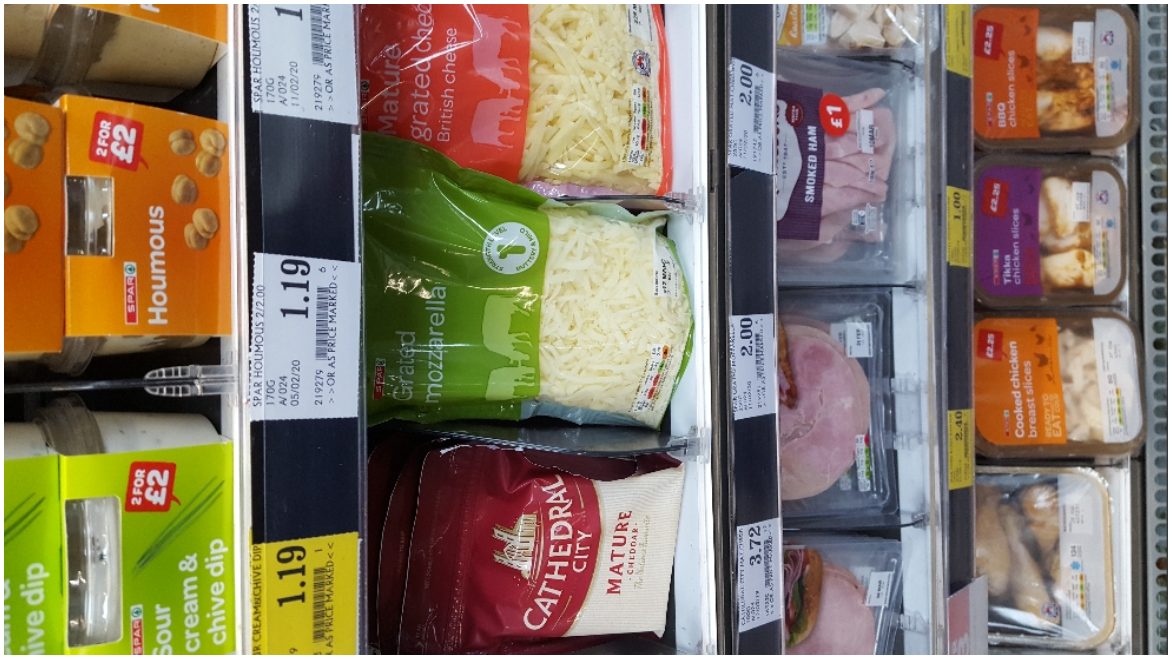
This screenshot shows the browser history of the default Samsung browser on the device. This information aligns with previous data we have extracted from the phone. We can see that pictures of a pig were downloaded and the APKs for WhatsApp and Signal Messenger. We can also see that the user searched for dog pictures as well, however no dog images were downloaded. We checked the search history for the chrome application as well did not find any search information within.

Final reconstruction:

We believe the subject was given the task of taking a picture of some cheese, sending the image along with a recommendation to the recipient. We can't be fully comprehensive as the ability to cross-reference the image location and the location it was sent from was very limited. The reconstruction goes as follows:

1. **Wednesday, February 19, 2020 4:42PM:** Message sent from GameMaster to Subjects giving task
1. **Thursday, February 20, 2020 06:00AM:** Approximate WhatsApp installation time
2. **Thursday, February 20, 2020 11:52AM:** Subjects sends image of cheese to GameMaster with caption "mhhh cheese"
3. **Friday, February 21, 2020 11:15AM:** GameMaster replies that they want more details
4. **Friday, February 21, 2020 2:36PM:** Subjects reply with more information, including SPAR being the location of the shop and that they recommend cheddar

Appendix



20200220_115211.jpg

This was the image taken of the cheese and sent to the GameMaster.