



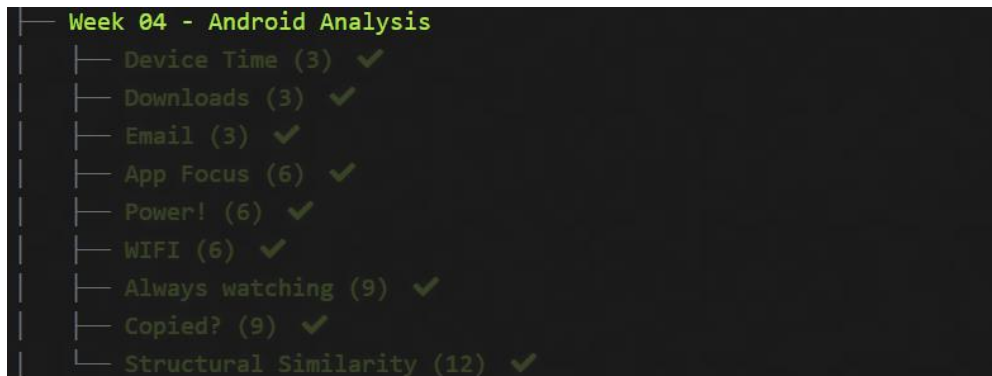
CYBERCRIME

Author: Kharim Mchatta

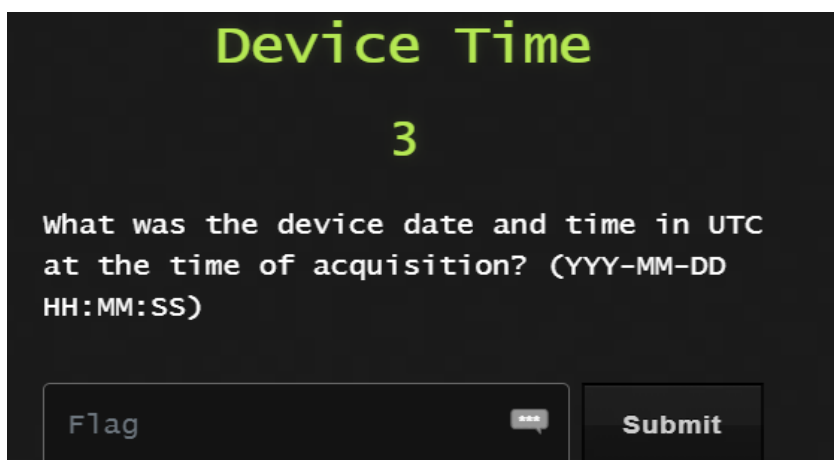
Title: Africa Digital Forensics CTF Competition

Date: 5/28/2021

On 5/24/2021 the fourth week of the competition of the United Nation Office of Drugs and Cybercrime (UNODC) CTF had begun. This was the last week of the competition. The fourth week challenge was based on analysis of android logical file. This was a very interesting challenge to me due to the way the questions were set. Some questions were very trick but doable. In this week we were faced with 9 questions which were supposed to be answered based on the android files which were provided.



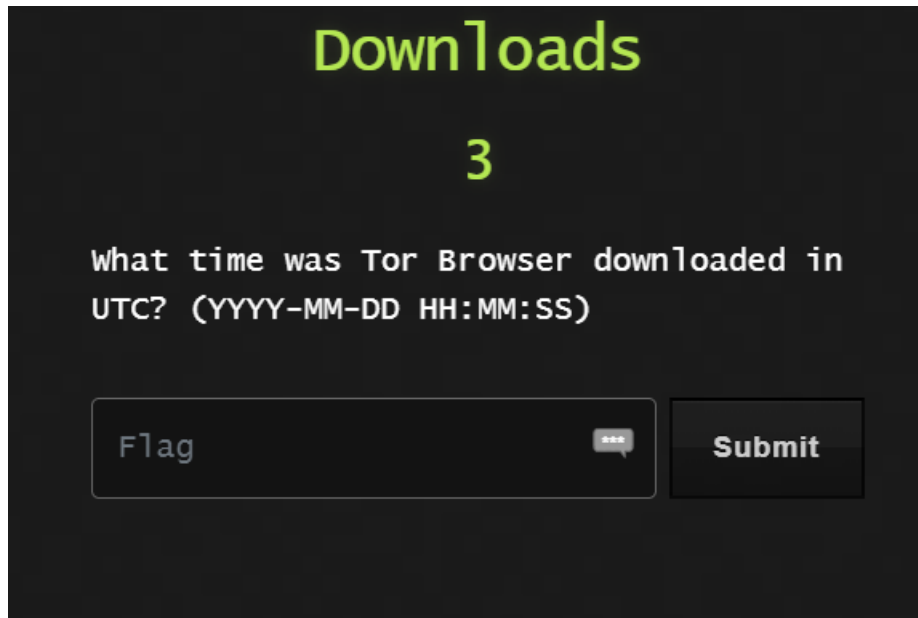
Starting off with the first challenge which was called device time where we were supposed to find the device date and time of acquisition.



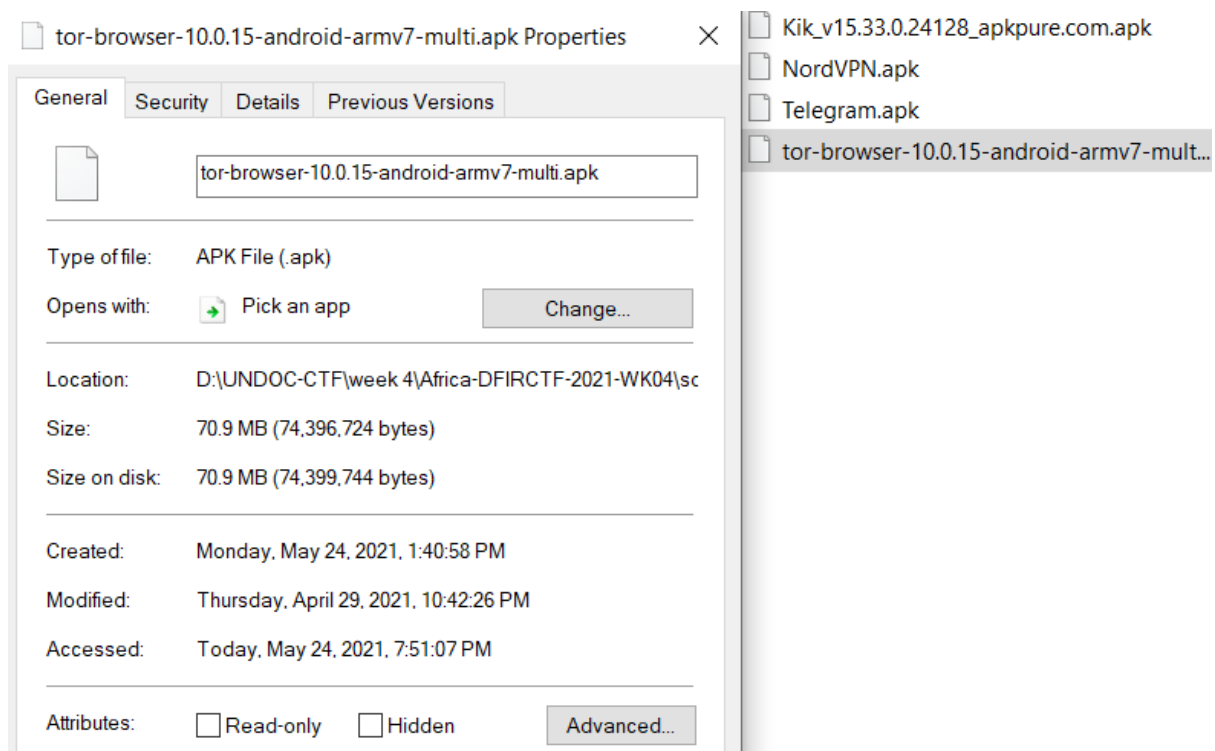
This was a straightforward question. When you go to live data you will see a file named device_datetime_utc and there would be your answer.

LGE LM-Q725K Quick Image > Live Data			
Name	Date modified	Type	Size
 Dumpsys Data	5/24/2021 12:15 PM	File folder	
 device_datetime_utc	5/22/2021 3:17 AM	Text Document	1 K
 device_datetime_utc - Notepad			
File Edit Format View Help			
2021-05-21 18:17:56			

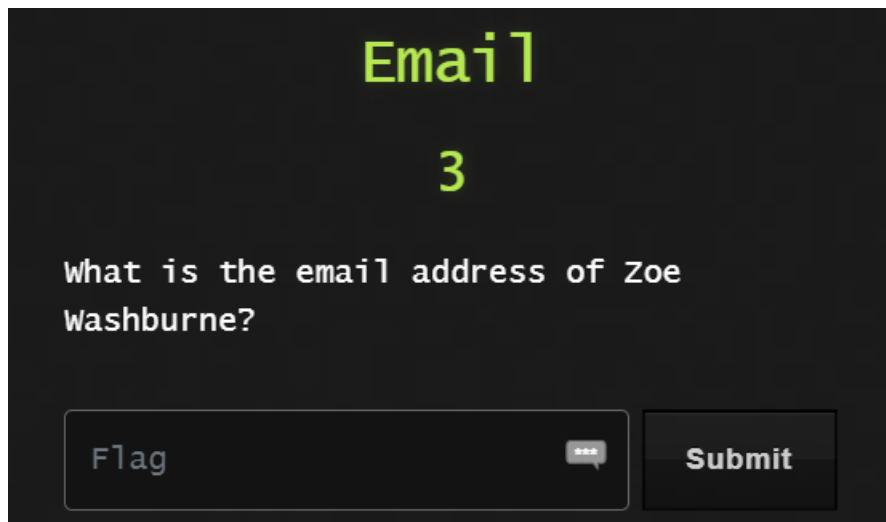
The next challenge was named Downloads, and the question asked about when the time was the TOR browser was downloaded, the question is a bit tricky in the sense that the time is in UTC so make sure your system time or your autopsy is in UTC time zone otherwise you will not get the answer.



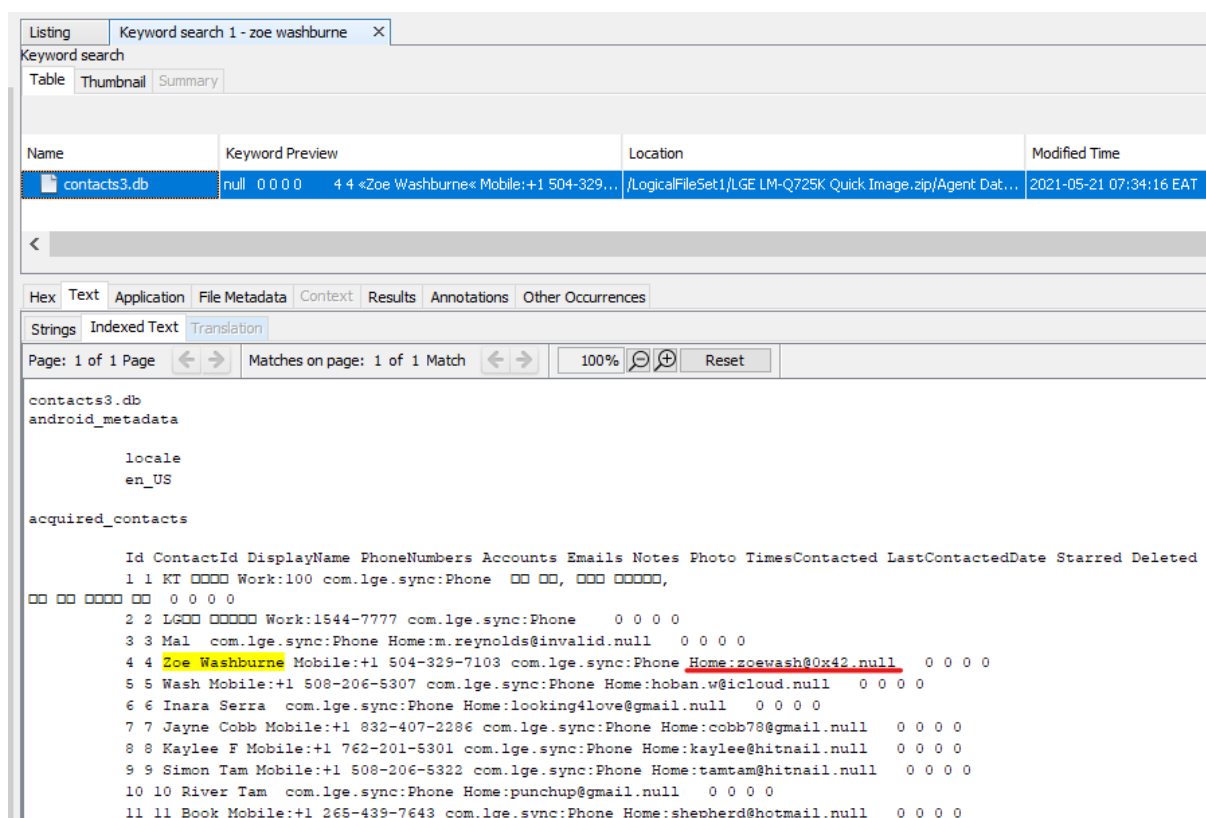
This could be found in the downloads folder and when you right click and check the properties you will see the time



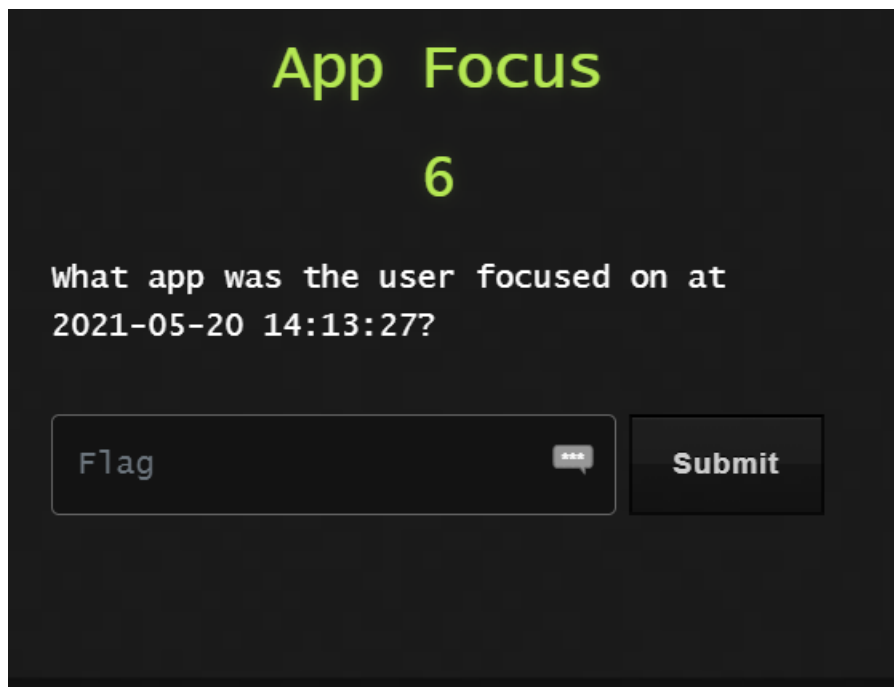
The next challenge was called email where we were supposed to look for the email address of ZOE



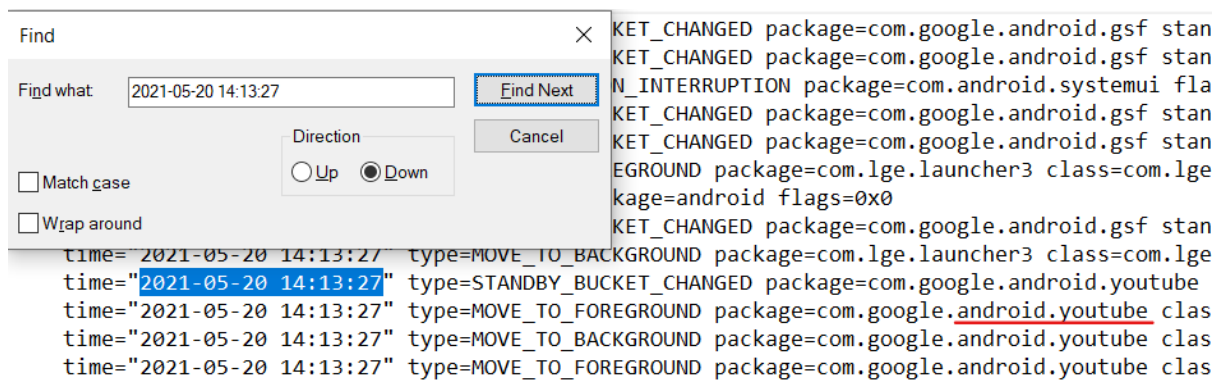
Getting the answer was a very straight forward process where I did a key word search of Zoe Washburne, and I got the flag.



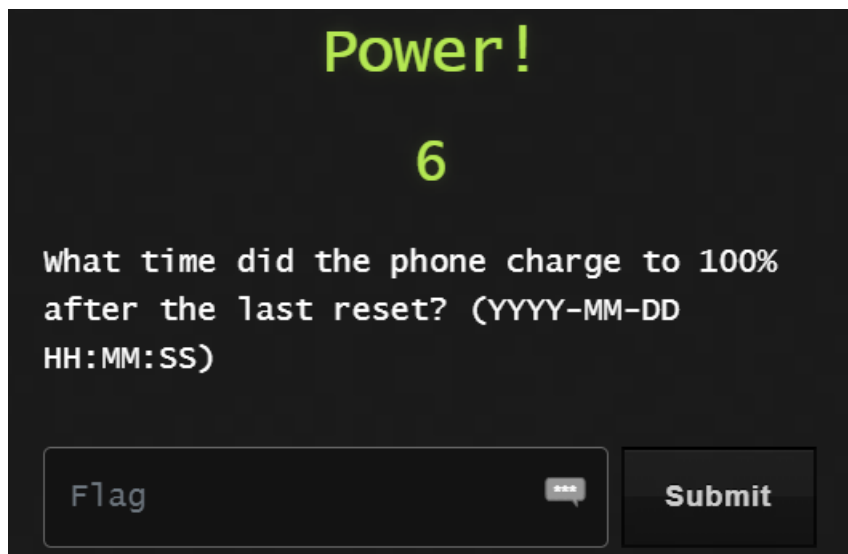
The next question was called App Focus where we were supposed to look for the app that was running at the specified time



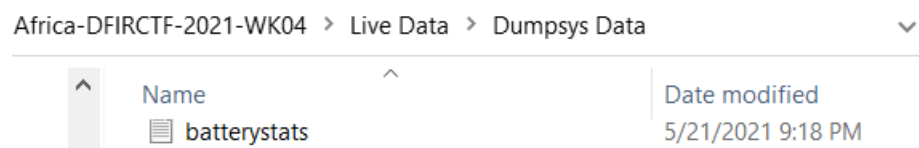
When you go live data then usage stats and search for the specified time you will see that the app that was used was YouTube



The next challenge was named power which required us to find when the last time was the phone was fully charged to 100% after the last reset.



First, we needed to find the file which contained information about the battery life which could be found in **live data – dumphsys data – batterystats** as shown in the below image



Opening the file and searching the word **status=full** you would see that the battery was fully charged in 5mins 01 secs and 459 milli secs

batterystats - Notepad

File Edit Format View Help

```
+4m52s141ms (2) 099 brightness=dark
+5m01s459ms (3) 100 status=full charge=2665
```

And scrolling up the file you could see that the phone was reset at 13:12:19

batterystats - Notepad

File Edit Format View Help

```
Battery History (0% used, 908 used of 512KB, 17 strings using 1928):
0 (10) RESET:TIME: 2021-05-21-13-12-19
0 (2) 099 status=discharging health=good plug=none
```

To get the flag you take the time of reset and add the time the battery took to be fully charged.

The following question was named WIFI where we were supposed to look for the password of the Access point.



This could be found in the app folder under the android providers setting folder as shown in the folder below.

Africa-DFIRCTF-2021-WK04 > apps > com.android.providers.settings > k

Name	Date modified	Type	Size
_manifest	5/21/2021 9:22 PM	File	3 KB
com.android.providers.settings.data	5/21/2021 9:22 PM	DATA File	5 KB

Once you open the file and scroll down you will find the preshared key which is the password of the WIFI.

Name	Size	Date modified	Type
_manifest	3	0000-00-00 00:00:00	File
com.android.providers.settings.data	3	2021-05-21 21:22:19 EAT	DATA File

Hex	Text	Application	File Metadata	Context	Results	Annotations	Other Occurrences
Strings Indexed Text Translation Page: 1 of 1 Page Matches on page: - of - Match 100% Reset							
<pre> <float name="Version" value="1.0" /> <NetworkList> <Network> <WifiConfiguration> <string name="ConfigKey">&quot;Hot_SSL&quot;WPA_PSK</string> <string name="SSID">&quot;Hot_SSL&quot;</string> <null name="BSSID" /> <boolean name="ShareThisAp" value="false" /> <string name="PreSharedKey">&quot;ThinkingForest!&quot;</string> </pre>							

The next question was called always watching, where we were required to find out how much time the suspect spent on YouTube.

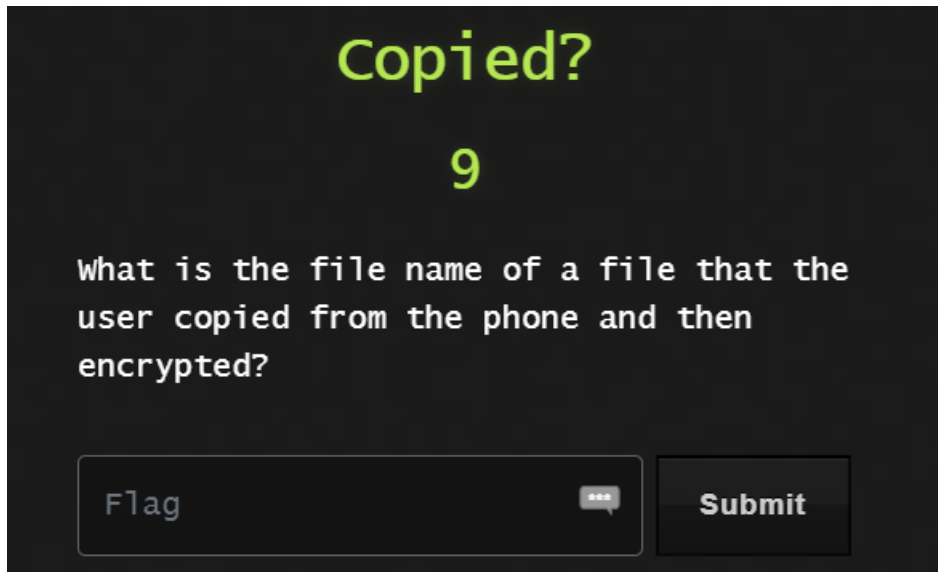


When you go live data then usage_stats all you have to do is filter out the unwanted information as shown below

```
# cat usage_stats.txt | grep 2021-05-20 | grep com.google.android.youtube
time="2021-05-20 14:13:27" type=STANDBY_BUCKET_CHANGED package=com.google.android.youtube
time="2021-05-20 14:13:27" type=MOVE_TO_FOREGROUND package=com.google.android.youtube clas
time="2021-05-20 14:13:27" type=MOVE_TO_BACKGROUND package=com.google.android.youtube clas
time="2021-05-20 14:13:27" type=MOVE_TO_FOREGROUND package=com.google.android.youtube clas
time="2021-05-20 15:13:27" type=STANDBY_BUCKET_CHANGED package=com.google.android.youtube
time="2021-05-20 15:18:37" type=STANDBY_BUCKET_CHANGED package=com.google.android.youtube
time="2021-05-20 16:13:28" type=STANDBY_BUCKET_CHANGED package=com.google.android.youtube
time="2021-05-20 16:23:28" type=STANDBY_BUCKET_CHANGED package=com.google.android.youtube
time="2021-05-20 18:13:29" type=STANDBY_BUCKET_CHANGED package=com.google.android.youtube
time="2021-05-20 18:23:29" type=STANDBY_BUCKET_CHANGED package=com.google.android.youtube
time="2021-05-20 20:13:30" type=STANDBY_BUCKET_CHANGED package=com.google.android.youtube
time="2021-05-20 20:23:30" type=STANDBY_BUCKET_CHANGED package=com.google.android.youtube
time="2021-05-20 22:13:30" type=STANDBY_BUCKET_CHANGED package=com.google.android.youtube
time="2021-05-20 22:23:30" type=STANDBY_BUCKET_CHANGED package=com.google.android.youtube
time="2021-05-20 22:47:57" type=MOVE_TO_BACKGROUND package=com.google.android.youtube clas
time="2021-05-20 22:47:57" type=STANDBY_BUCKET_CHANGED package=com.google.android.youtube
time="2021-05-20 23:47:57" type=STANDBY_BUCKET_CHANGED package=com.google.android.youtube
package=com.google.android.youtube totalTime="8:34:29" lastTime="2021-05-20 22:47:57" ap
package=com.google.android.youtube totalTime="8:34:29" lastTime="2021-05-20 22:47:57" ap
package=com.google.android.youtube totalTime="8:34:29" lastTime="2021-05-20 22:47:57" ap
```

As we can see below the total time is **8:34:29** and the last time was **22:47:57** since the total time is 8:34:29 and it counts from 0:00:00 then we add seconds to the total time which will be 8:34:30.

The next challenge was called copied, the question wants us to find the file the user copied from the phone and encrypted.



Copied?

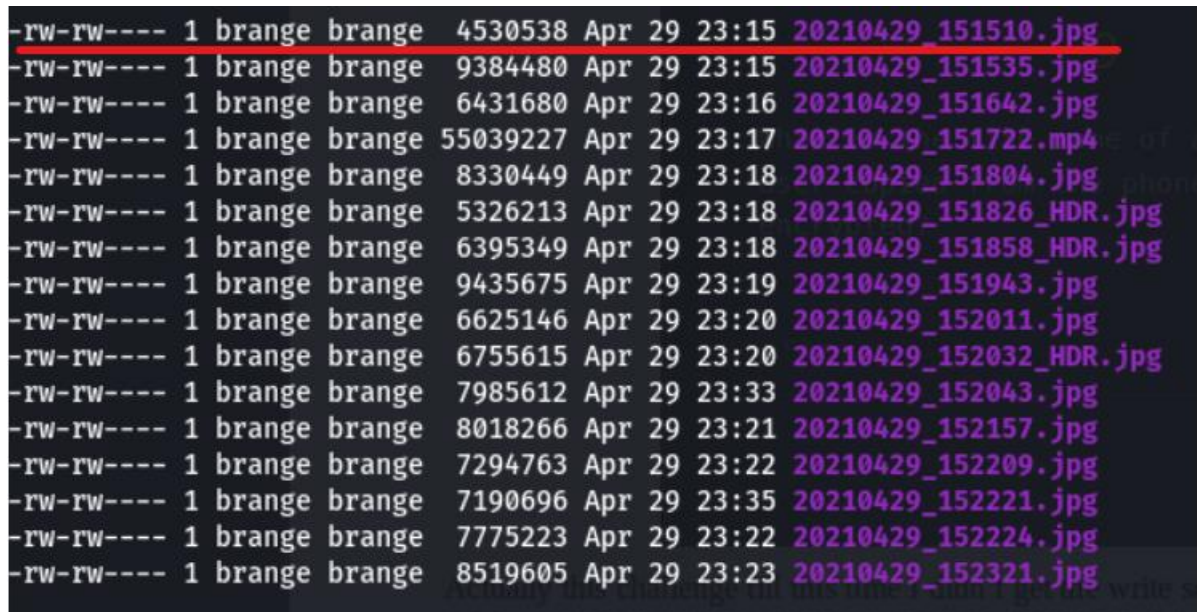
9

What is the file name of a file that the user copied from the phone and then encrypted?

Flag

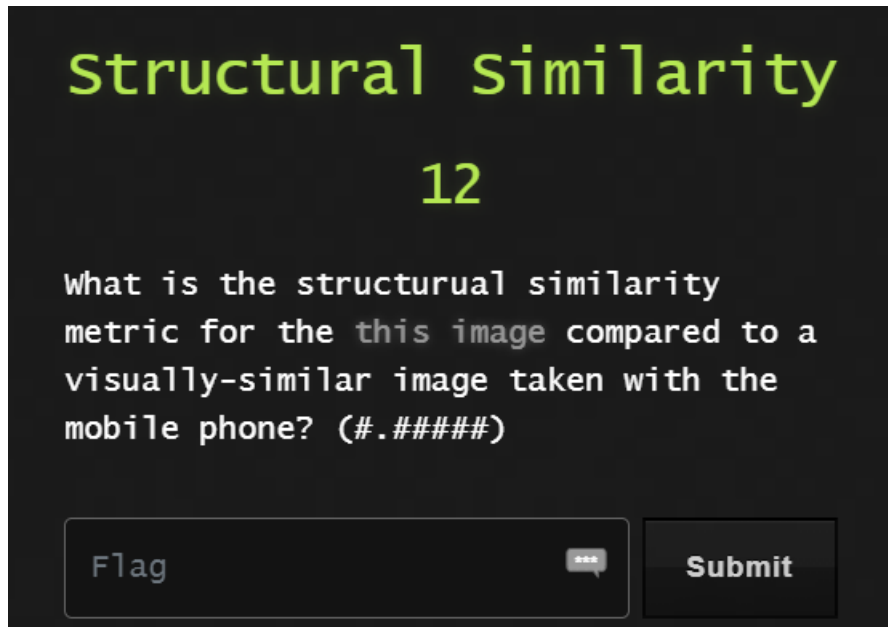
Submit

The key to this challenge was to brute force the file of camera in **Africa-DFIRCTF-2021-WK04/LGE LM-Q725K Quick Image/adb-data/shared/0/DCIM/Camera**. Once your brute force the file once brute forced the image **20210429_151510.jpg**

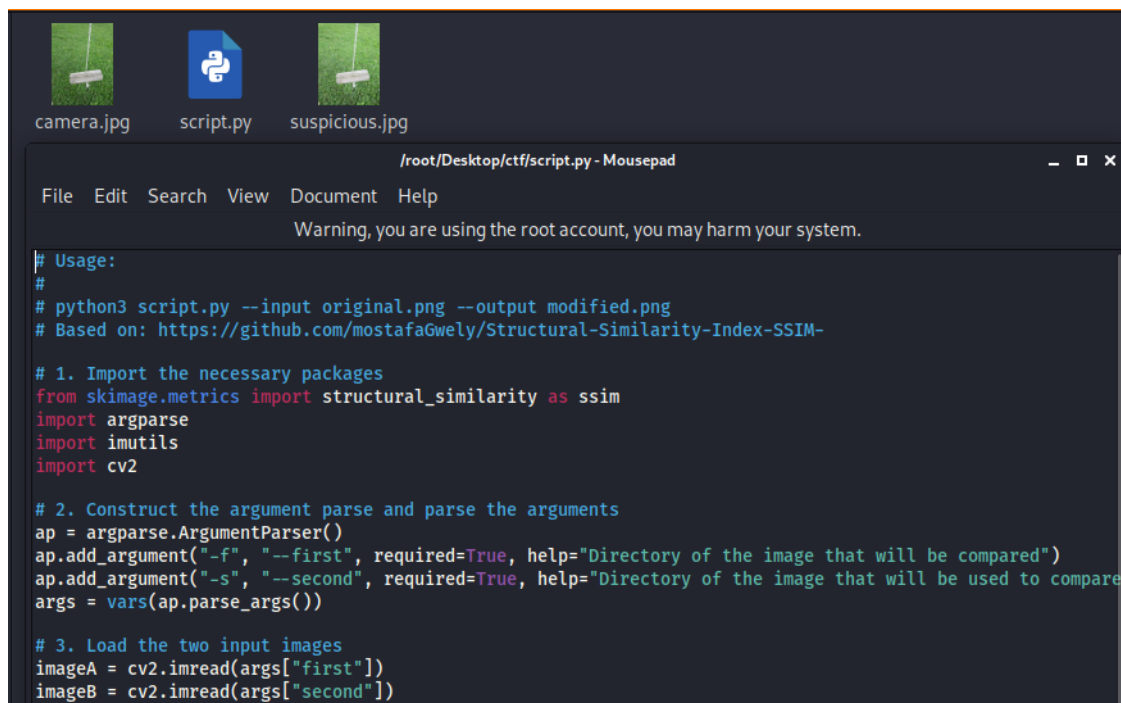


```
-rw-rw---- 1 brange brange 4530538 Apr 29 23:15 20210429_151510.jpg
-rw-rw---- 1 brange brange 9384480 Apr 29 23:15 20210429_151535.jpg
-rw-rw---- 1 brange brange 6431680 Apr 29 23:16 20210429_151642.jpg
-rw-rw---- 1 brange brange 55039227 Apr 29 23:17 20210429_151722.mp4
-rw-rw---- 1 brange brange 8330449 Apr 29 23:18 20210429_151804.jpg
-rw-rw---- 1 brange brange 5326213 Apr 29 23:18 20210429_151826_HDR.jpg
-rw-rw---- 1 brange brange 6395349 Apr 29 23:18 20210429_151858_HDR.jpg
-rw-rw---- 1 brange brange 9435675 Apr 29 23:19 20210429_151943.jpg
-rw-rw---- 1 brange brange 6625146 Apr 29 23:20 20210429_152011.jpg
-rw-rw---- 1 brange brange 6755615 Apr 29 23:20 20210429_152032_HDR.jpg
-rw-rw---- 1 brange brange 7985612 Apr 29 23:33 20210429_152043.jpg
-rw-rw---- 1 brange brange 8018266 Apr 29 23:21 20210429_152157.jpg
-rw-rw---- 1 brange brange 7294763 Apr 29 23:22 20210429_152209.jpg
-rw-rw---- 1 brange brange 7190696 Apr 29 23:35 20210429_152221.jpg
-rw-rw---- 1 brange brange 7775223 Apr 29 23:22 20210429_152224.jpg
-rw-rw---- 1 brange brange 8519605 Apr 29 23:23 20210429_152321.jpg
```

Next was the challenge which was called Structural similarity where we were required to get the structural similarity of the two images, one which was provided and the other which could be found in the logical drive.



In order to solve this challenge, you had to make a script which is going to assist you in calculating the structural differences of the two images.



The code can be found: <https://ourcodeworld.com/articles/read/991/how-to-calculate-the-structural-similarity-index-ssim-between-two-images-with-python> or <https://github.com/mostafaGwely/Structural-Similarity-Index-SSIM->