Charles Wallis
CS4398.001
04/14/2023
Assignment 6

1. Create a PNG file
    a. Image is scaled down to 120x80, 19.5kb size file

2. The inode of this file is 14

```
forensics@forensics:/media/forensics/8ccfa268-2699-474d-ab2f-c7b92f637dc0$ ls -i image.png
14 image.png
```

3. I ran the inode tool from my previous assignment to find the i_block[0] location, which is 33792. This is where my PNG start signature should be located at. The Inode analyser below prints information about inode 14 (which is image.png)

```
forensics@forensics:~/Desktop/Assignment6$ sudo ./inode /dev/sda1
Total inodes: 61056, Inodes per group: 7632, Size of inode: 256
Block Size: 4096 Bytes
Inode Analyser# 14
 Invalid Command ! Try 'help' for available commands
Inode Analyser# inode 14
 i_mode = 33204
 i_uid = 1000
 i_size = 20257
 i_atime = 1681533121
 i_ctime = 1681533121
 i_mtime = 1681533121
 i_dtime = 0
 i_gid = 1000
 i_links_count = 1
 i_blocks = 40
    i_block[0] = 33792
    i_block[1] = 33793
    i_block[2] = 33794
    i_block[3] = 33795
    i_block[4] = 33796
    i_block[5] = 0
    i_block[6] = 0
    i_block[7] = 0
    i_block[8] = 0
    i_block[9] = 0
    i_block[10] = 0
    i_block[11] = 0
    i_block[12] = 0
    i_block[13] = 0
    i_block[14] = 0
 i_flags = 0
 i_generation = 4031516515
 i_file_acl = 0
 i_dir_acl = 0
 i_faddr = 0
 i_extra_isize = 32
 i_pad1 = 0
Inode Analyser#
```

4. Now I am going to scan the partition, one block at a time. The program will match PNG signatures to find the blocks where PNG files exist. Below is the image of my python code

```python
import os
import sys

def main():
    #PNG filetype signature (first 8 bytes) using HEX
    png_signature = bytes.fromhex('89 50 4E 47 0D 0A 1A 0A')

    block_size = 4096

    #path of my partition
    partition_path = "/dev/sda1"
    print(f"partition_path set to {partition_path}")

    with open(partition_path, "rb") as partition:
        block_number = 0
        while True:
            block = partition.read(block_size)

            if not block:
                print(f"Reached end of partition after reading {block_number} blocks")
                break

            #print(f"Block {block_number}: {block[:8]}")
            if block[:8] == png_signature:
                print(f"Found PNG signature at block {block_number}")

            block_number += 1

if __name__ == "__main__":
    main()
```

5. My program detected more than one block of PNGs (matching PNG signatures), because my partition has several PNG files in it. Some PNG files could have come from other programs, or from my previous downloaded files. It could even be remnants of older PNG files as well.

```
lubuntu@lubuntu:~/Downloads$ sudo python3 find_blocks.py image.png
partition_path set to /dev/zram0
Found PNG signature at block 223552
Found PNG signature at block 223560
Found PNG signature at block 223568
Found PNG signature at block 223576
Found PNG signature at block 333456
Found PNG signature at block 510664
Found PNG signature at block 954507
Found PNG signature at block 954508
Found PNG signature at block 1050320
Found PNG signature at block 1189208
Found PNG signature at block 1623416
Found PNG signature at block 1744400
Found PNG signature at block 1788592
Reached end of partition after reading 2022944 blocks
lubuntu@lubuntu:~/Downloads$
```

To fix this issue, I tried re-creating the partition and zeroing but since this was attempted on my ubuntu emulator, I was not able to accomplish that- so I switched boot mode so I can try this again in linux.

(also, the above image was done with the code reading with block_size = 512, which has now been fixed to 4096.

6. The first block found did match the first block in the inode printout (33792), given from the inode analyzer tool, which indicates that my program worked, the PNG file is correct, and my PNG signature was correctly detected.