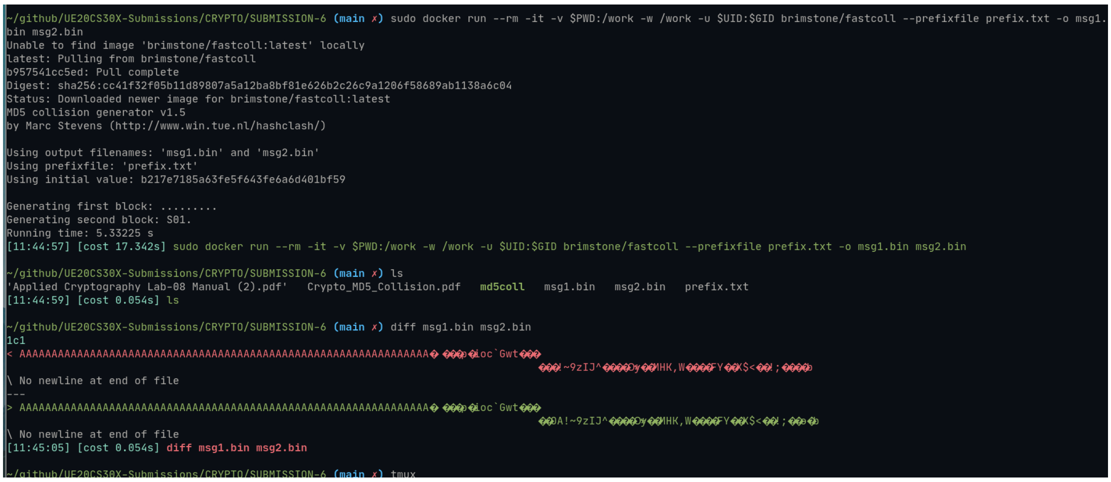
Cryptograhphy Hands-On submission 6 | MD5

Details :

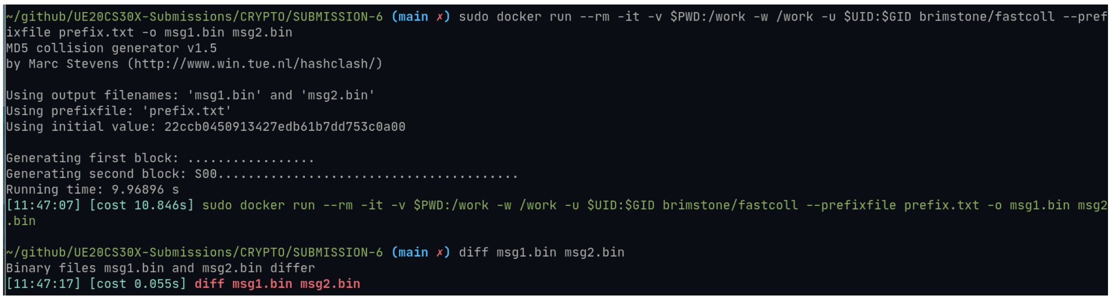
- SRN : PES2UG20CS237
- Name : P K Navin Shrinivas
- Section : D

TASK 1 : Creating colliding MD5 hashes

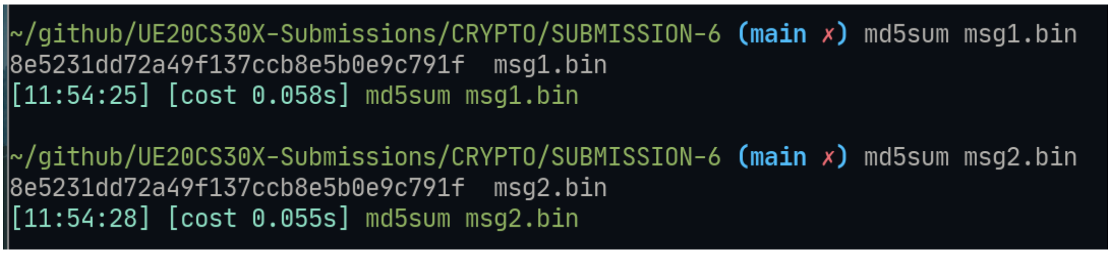
Screenshots :



```
~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) sudo docker run --rm -it -v $PWD:/work -w /work -u $UID:$GID brimstone/fastcoll --prefixfile prefix.txt -o msg1.
bin msg2.bin
Unable to find image 'brimstone/fastcoll:latest' locally
latest: Pulling from brimstone/fastcoll
b957541cc5ed: Pull complete
Digest: sha256:cc41f32f05b11d89807a5a12ba8bf81e626b2c26c9a1206f58689ab1138a6c04
Status: Downloaded newer image for brimstone/fastcoll:latest
MD5 collision generator v1.5
by Marc Stevens (http://www.win.tue.nl/hashclash/)

Using output filenames: 'msg1.bin' and 'msg2.bin'
Using prefixfile: 'prefix.txt'
Using initial value: b217e7185a63fe5f643fe6a6d401bf59

Generating first block: .........
Generating second block: S01.
Running time: 5.33225 s
[11:44:57] [cost 17.342s] sudo docker run --rm -it -v $PWD:/work -w /work -u $UID:$GID brimstone/fastcoll --prefixfile prefix.txt -o msg1.bin msg2.bin

~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) ls
'Applied Cryptography Lab-08 Manual (2).pdf'   Crypto_MD5_Collision.pdf   md5coll   msg1.bin   msg2.bin   prefix.txt
[11:44:59] [cost 0.054s] ls

~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) diff msg1.bin msg2.bin
1c1
< AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA� ���p�ioc`Gwt���
                                                                                 ���!~9zIJ^����D�y��MHK,W����FY��X$<��!;����b
\ No newline at end of file
---
> AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA� ���p�ioc`Gwt���
                                                                                 ��DA!~9zIJ^����D�y��MHK,W����FY��X$<��!;��e�b
\ No newline at end of file
[11:45:05] [cost 0.054s] diff msg1.bin msg2.bin

~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) tmux
```



```
~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) sudo docker run --rm -it -v $PWD:/work -w /work -u $UID:$GID brimstone/fastcoll --pref
ixfile prefix.txt -o msg1.bin msg2.bin
MD5 collision generator v1.5
by Marc Stevens (http://www.win.tue.nl/hashclash/)

Using output filenames: 'msg1.bin' and 'msg2.bin'
Using prefixfile: 'prefix.txt'
Using initial value: 22ccb0450913427edb61b7dd753c0a00

Generating first block: ................
Generating second block: S00......................................
Running time: 9.96896 s
[11:47:07] [cost 10.846s] sudo docker run --rm -it -v $PWD:/work -w /work -u $UID:$GID brimstone/fastcoll --prefixfile prefix.txt -o msg1.bin msg2
.bin

~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) diff msg1.bin msg2.bin
Binary files msg1.bin and msg2.bin differ
[11:47:17] [cost 0.055s] diff msg1.bin msg2.bin
```



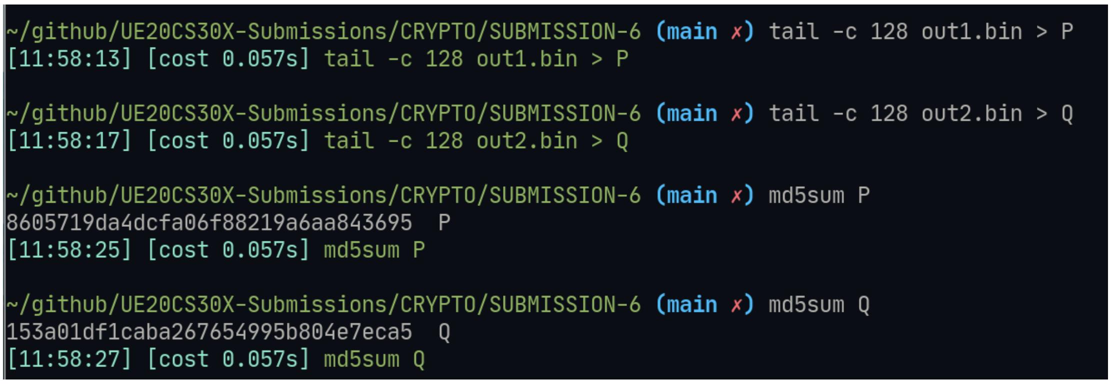
```
~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) md5sum msg1.bin
8e5231dd72a49f137ccb8e5b0e9c791f  msg1.bin
[11:54:25] [cost 0.058s] md5sum msg1.bin

~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) md5sum msg2.bin
8e5231dd72a49f137ccb8e5b0e9c791f  msg2.bin
[11:54:28] [cost 0.055s] md5sum msg2.bin
```

Observation :

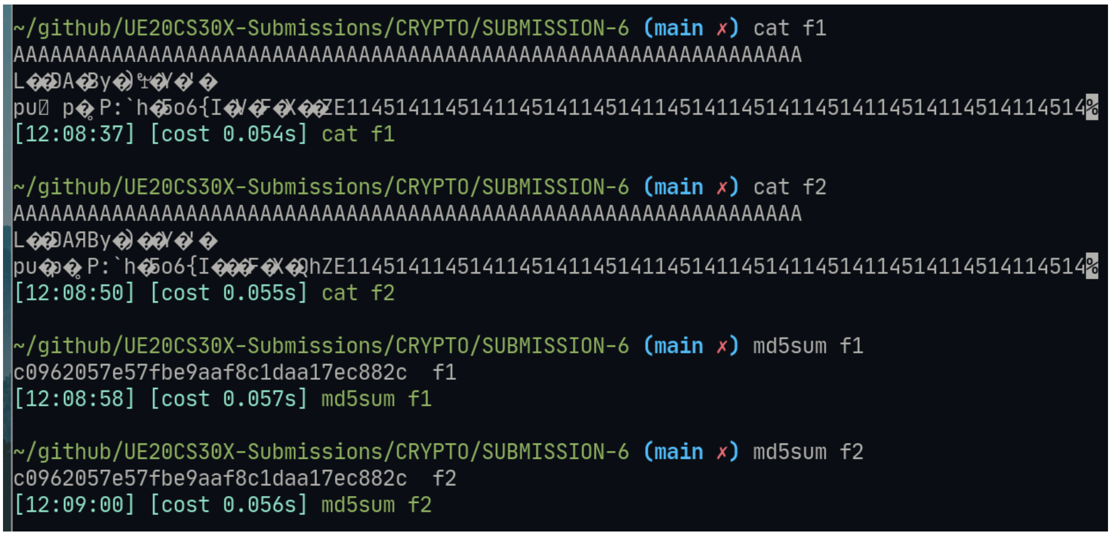
- When the input file is 64 letter long, the md5 hashes do collide.
- When the input file is not 64 letter long, the md5 hashes do not collide.
- But even when the hashes are diff, the md5sum output is the same

TASK 2 : Prefix and Suffix collisions

Screenshots :



```
~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) tail -c 128 out1.bin > P
[11:58:13] [cost 0.057s] tail -c 128 out1.bin > P

~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) tail -c 128 out2.bin > Q
[11:58:17] [cost 0.057s] tail -c 128 out2.bin > Q

~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) md5sum P
8605719da4dcfa06f88219a6aa843695  P
[11:58:25] [cost 0.057s] md5sum P

~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) md5sum Q
153a01df1caba267654995b804e7eca5  Q
[11:58:27] [cost 0.057s] md5sum Q
```



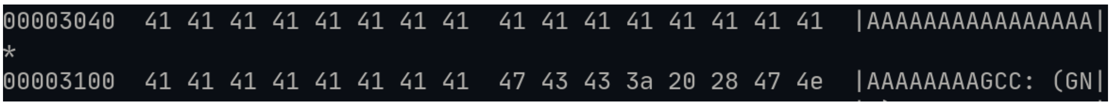
```
~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) cat f1
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
L��DA�By�)�Y�'�
pu p�P:`h�5o6{I�V�F�X��ZE1145141145141145141145141145141145141145141145141145141145141145141114514%
[12:08:37] [cost 0.054s] cat f1

~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) cat f2
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
L��DAЯBy�)��Y�'�
pu�p�P:`h�5o6{I���F�X�QhZE1145141145141145141145141145141145141145141145141145141145141145114514%
[12:08:50] [cost 0.055s] cat f2

~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) md5sum f1
c0962057e57fbe9aaf8c1daa17ec882c  f1
[12:08:58] [cost 0.057s] md5sum f1

~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) md5sum f2
c0962057e57fbe9aaf8c1daa17ec882c  f2
[12:09:00] [cost 0.056s] md5sum f2
```

Observation :

- postfix of the 2 hashes are not the same as md5sum is turning out to be different
- Using the same post and pre fix leads to same md5 hashes, this is due the same contents in the entire of md5's bit len

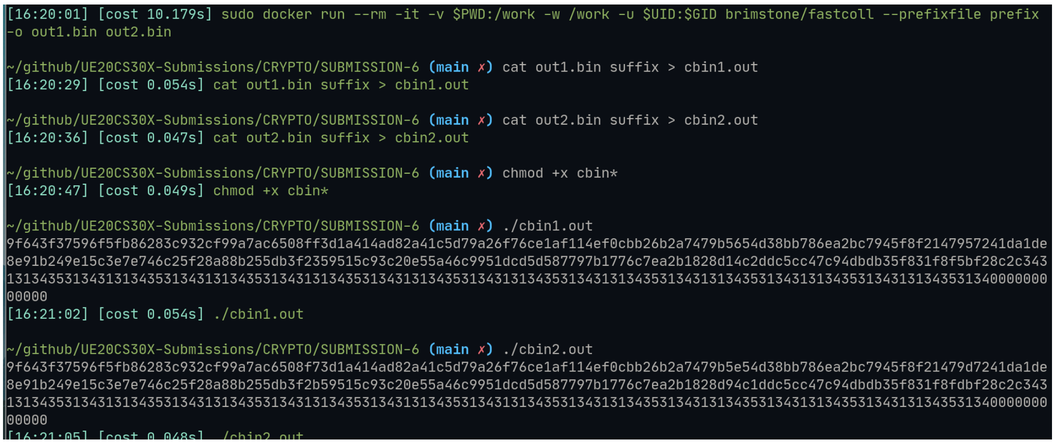
TASK 3

Screenshots :



```
00003040  41 41 41 41 41 41 41 41  41 41 41 41 41 41 41 41  |AAAAAAAAAAAAAAAA|
*
00003100  41 41 41 41 41 41 41 41  47 43 43 3a 20 28 47 4e  |AAAAAAAAGCC: (GN|
```

- Starting index : 12352
- Endind index : 12553



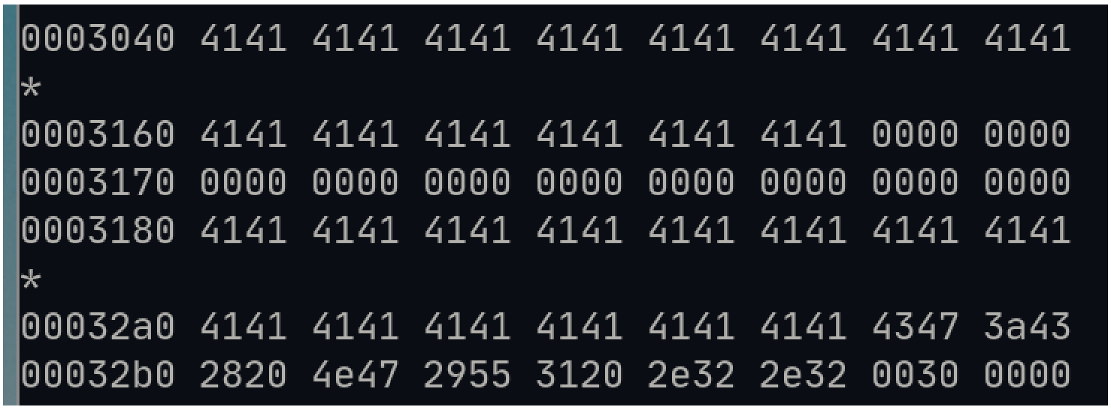
```
[16:20:01] [cost 10.179s] sudo docker run --rm -it -v $PWD:/work -w /work -u $UID:$GID brimstone/fastcoll --prefixfile prefix
-o out1.bin out2.bin

~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) cat out1.bin suffix > cbin1.out
[16:20:29] [cost 0.054s] cat out1.bin suffix > cbin1.out

~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) cat out2.bin suffix > cbin2.out
[16:20:36] [cost 0.047s] cat out2.bin suffix > cbin2.out

~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) chmod +x cbin*
[16:20:47] [cost 0.049s] chmod +x cbin*

~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) ./cbin1.out
9f643f37596f5fb86283c932cf99a7ac6508ff3d1a414ad82a41c5d79a26f76ce1af114ef0cbb26b2a7479b5654d38bb786ea2bc7945f8f2147957241da1de
8e91b249e15c3e7e746c25f28a88b255db3f2359515c93c20e55a46c9951dcd5d587797b1776c7ea2b1828d14c2ddc5cc47c94dbdb35f831f8f5bf28c2c343
131343531343131343531343131343531343131343531343131343531343131343531343131343531343131343531343131343531343131343531340000000
00000
[16:21:02] [cost 0.054s] ./cbin1.out

~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) ./cbin2.out
9f643f37596f5fb86283c932cf99a7ac6508f73d1a414ad82a41c5d79a26f76ce1af114ef0cbb26b2a7479b5e54d38bb786ea2bc7945f8f21479d7241da1de
8e91b249e15c3e7e746c25f28a88b255db3f2b59515c93c20e55a46c9951dcd5d587797b1776c7ea2b1828d94c1ddc5cc47c94dbdb35f831f8fdbf28c2c343
131343531343131343531343131343531343131343531343131343531343131343531343131343531343131343531343131343531343131343531340000000
00000
[16:21:05] [cost 0.048s] ./cbin2.out
```

Observation :

- Here we see that using md5colgen we generated 2 program that have the same hash despite their outputs.

TASK 4 : Changin behaviours of file more drastically.

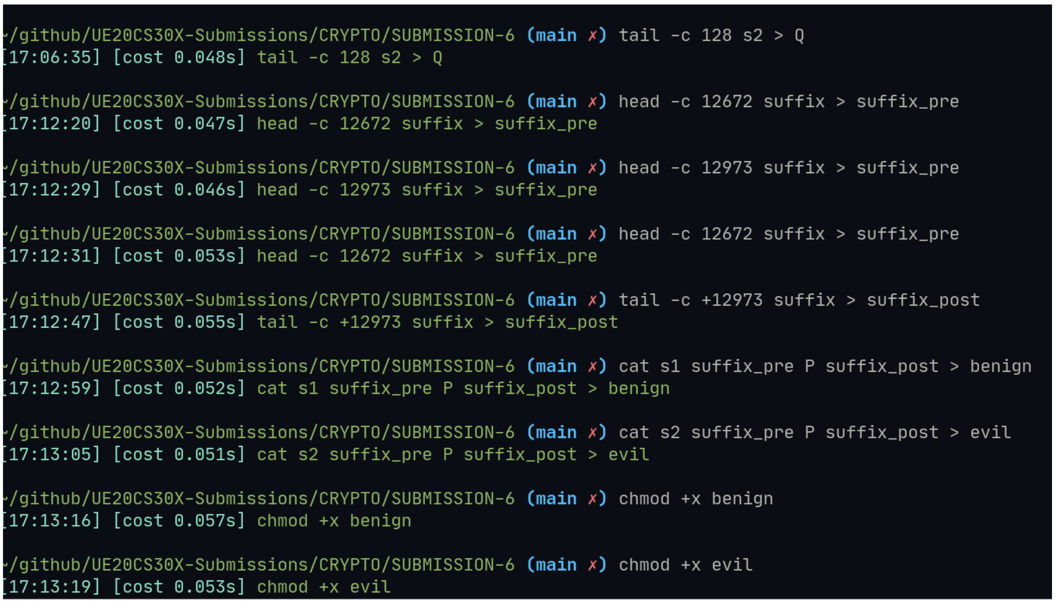
Screenshots and Observations :



```
0003040 4141 4141 4141 4141 4141 4141 4141 4141
*
0003160 4141 4141 4141 4141 4141 4141 0000 0000
0003170 0000 0000 0000 0000 0000 0000 0000 0000
0003180 4141 4141 4141 4141 4141 4141 4141 4141
*
00032a0 4141 4141 4141 4141 4141 4141 4347 3a43
00032b0 2820 4e47 2955 3120 2e32 2e32 0030 0000
```

- Start : 12552
- End : 12653
- Start of y : 12672

- End of y : 12973



```
/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) tail -c 128 s2 > Q
[17:06:35] [cost 0.048s] tail -c 128 s2 > Q

/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) head -c 12672 suffix > suffix_pre
[17:12:20] [cost 0.047s] head -c 12672 suffix > suffix_pre

/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) head -c 12973 suffix > suffix_pre
[17:12:29] [cost 0.046s] head -c 12973 suffix > suffix_pre

/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) head -c 12672 suffix > suffix_pre
[17:12:31] [cost 0.053s] head -c 12672 suffix > suffix_pre

/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) tail -c +12973 suffix > suffix_post
[17:12:47] [cost 0.055s] tail -c +12973 suffix > suffix_post

/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) cat s1 suffix_pre P suffix_post > benign
[17:12:59] [cost 0.052s] cat s1 suffix_pre P suffix_post > benign

/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) cat s2 suffix_pre P suffix_post > evil
[17:13:05] [cost 0.051s] cat s2 suffix_pre P suffix_post > evil

/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) chmod +x benign
[17:13:16] [cost 0.057s] chmod +x benign

/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-6 (main ✗) chmod +x evil
[17:13:19] [cost 0.053s] chmod +x evil
```

- Here we observe that the files have been modified and yet their hash remains same and the program cant tell that it has been hacked.