Crypto Lab - 3

Name: H M Mythreya
SRN: PES2UG20CS130

Task 1: Generate Encryption Key in a Wrong Way

With random seed

```
[09/26/22]seed@VM:~/Desktop$ gcc task1.c -o task1
[09/26/22]seed@VM:~/Desktop$ ./task1
1664250718
15139705ecfc076f54eee1169892bc77
[09/26/22]seed@VM:~/Desktop$ ./task1
1664250721
40d1f18c6aa9785d21eafbd5edff8120
[09/26/22]seed@VM:~/Desktop$ ./task1
1664250723
0c7508c7be1f5e706b670ac78d6a6c74
[09/26/22]seed@VM:~/Desktop$
```

With same seed

```
[09/26/22]seed@VM:~/Desktop$ gcc task1.c -o task1
[09/26/22]seed@VM:~/Desktop$ ./task1
1664250853
67c6697351ff4aec29cdbaabf2fbe346
[09/26/22]seed@VM:~/Desktop$ ./task1
1664250855
67c6697351ff4aec29cdbaabf2fbe346
[09/26/22]seed@VM:~/Desktop$ ./task1
1664250856
67c6697351ff4aec29cdbaabf2fbe346
[09/26/22]seed@VM:~/Desktop$
```

In computers, there is no such thing as random, only pseudo or false random. If we use the same seed to call the random function, the output is the same since the random function is simply a mathematical function designed to output false randoms.

As seen above, when the seed is set the same, the output isn't random.

Task 2: Guessing the Key

```
7186 ced0224b6836811253a5c11b2ee3db987
7187 b49f28baf332b6cc551296835f0b06e8
7188 b630b40215eb8c3c006bd5e8401f167c
7189 d5031571d5a1270c6c40c81e150071cf
7190 e2f75b7c0302adb44314decd99ba9dcf
7191 59a07e06ba43dfaa7207b22337d636c2
7192 d25bac21989dcae104b8f2ca345b4735
7193 88d68d8e488bf2e864bfc6df2a91d28d
7194 b8b44167a8ae9f5b84eaf0a9a3c3eb1f
7195 b08c060c79273ec8563657aa773507b4
7196 b5849bfeeb969c3070a7ea36b2694f5c
7197 2c343cd1cd1aa9ab1f141616439bd6fb
7198 a0ae1a29c3a79d7228209ebaa9100c4b
7199 47d111721d2be6c9669bcd467752170e
7200 78bb3b41e3c5251e343ef45ef833b519
7201 93c4ad2033792205fb2a29d1dc7c4f50
```

Plain Text Tab Width: 8 Ln 7201, Col 33 INS

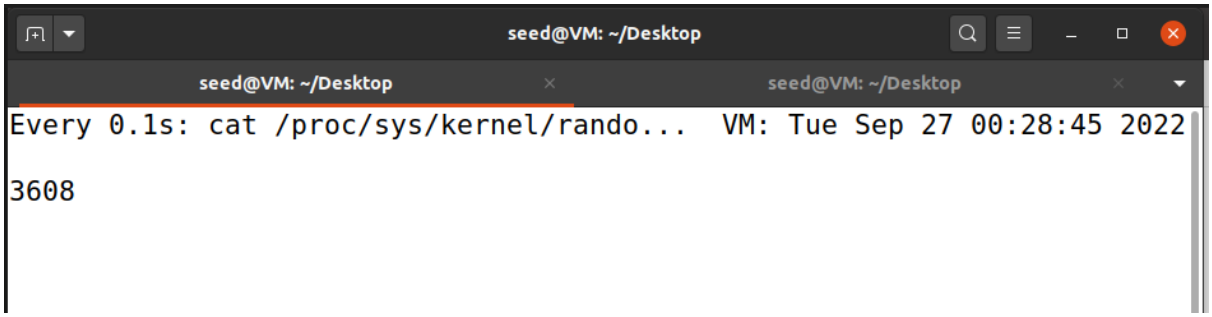
Right Ctrl

Since the seed used by the host is based on time, using the timestamp, we can bruteforce all possible combinations of the seed that could've been used in the timeframe. As seen above, there are only 7201 possible keys from the timestamp which isn't safe since modern computers can brute-force millions of keys a second.

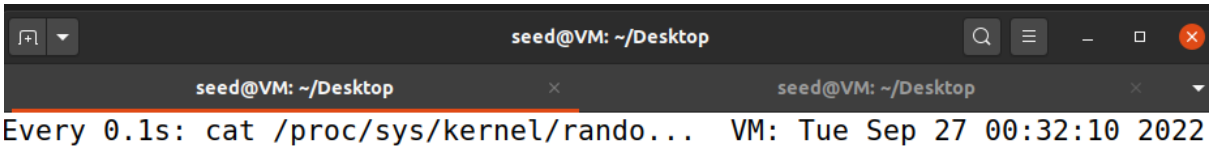
```
[09/27/22]seed@VM:~/Desktop$ python3 task3.py

Match found
key: 95fa2030e73ed3f8da761b4eb805dfd7
Ciphertext: d06bf9d0dab8e8ef880660d2af65aa82
Encrypted: d06bf9d0dab8e8ef880660d2af65aa82

[09/27/22]seed@VM:~/Desktop$
```

By trying every key out of the 7200 keys, we can easily crack the encryption.

Task 3: Measure the Entropy of Kernel



Task 4: Get Pseudo Random Numbers from /dev/random

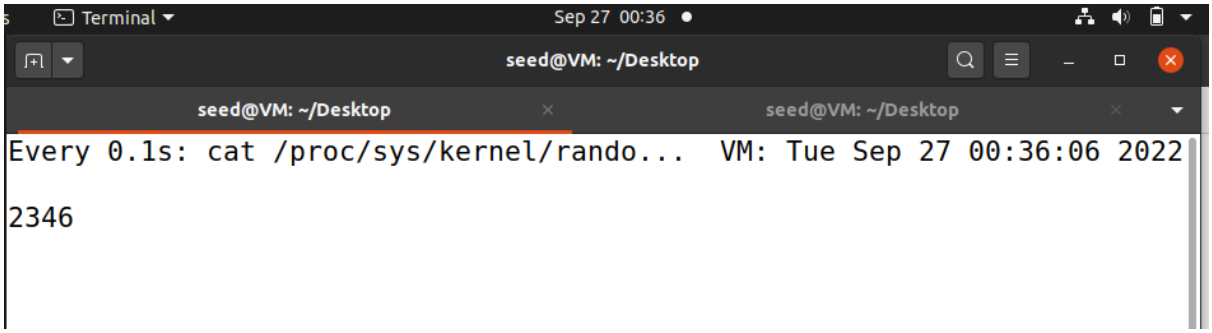


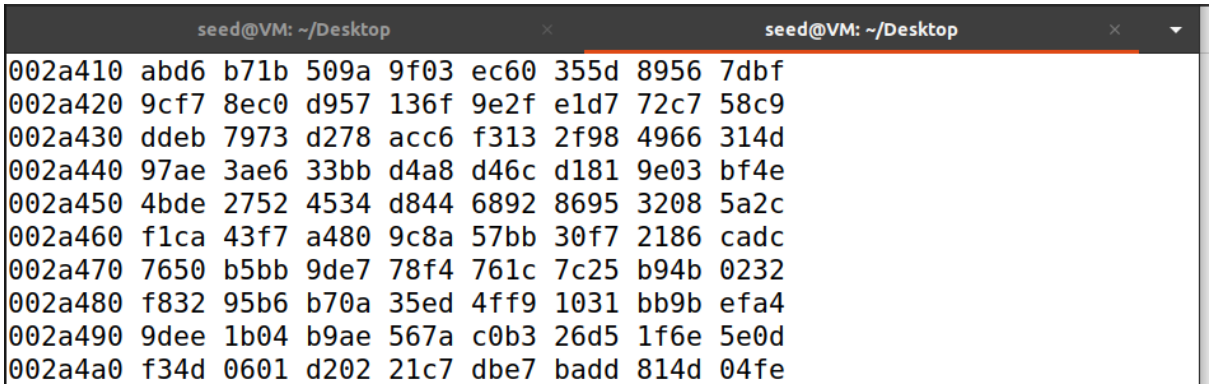

```
[09/27/22]seed@VM:~/Desktop$ cat /dev/random | hexdump
0000000 3559 7ad5 c01e e022 34ea 3ab0 c75f e9c8
0000010 d950 5538 f573 77d8 7c16 c247 603c 8a39
0000020 bba0 1b47 f8fe 9de3 030f aa97 f841 f66a
0000030 ba38 34a1 7649 7717 ba44 1eec b0af 71a5
0000040 7f29 ddb6 accd f3b2 574a 58f9 02f5 756a
```

The entropy of the kernel depends on the state of the system. When there is no activity, i.e, keyboard or mouse movement, there is no change. When there is an activity, the entropy changes.

Task 5: Get Pseudo Random Numbers from /dev/urandom

Entropy of kernel and the hexdump output when using /dev/urandom:





Here we are checking the quality of the randomness using ent

```
[09/27/22]seed@VM:~/Desktop$ head -c 1M /dev/urandom > output.bin
[09/27/22]seed@VM:~/Desktop$ ent output.bin
Entropy = 7.999835 bits per byte.

Optimum compression would reduce the size
of this 1048576 byte file by 0 percent.

Chi square distribution for 1048576 samples is 240.40, and randomly
would exceed this value 73.55 percent of the times.

Arithmetic mean value of data bytes is 127.5264 (127.5 = random).
Monte Carlo value for Pi is 3.143086026 (error 0.05 percent).
Serial correlation coefficient is -0.001238 (totally uncorrelated = 0.0
).
[09/27/22]seed@VM:~/Desktop$
```

Instead of using time for a random seed, we can use the entropy of the kernel with dev/urandom for better randomness and hence make it harder to break the encryption.

```
[09/27/22]seed@VM:~/Desktop$ gcc task5.c -o task5
[09/27/22]seed@VM:~/Desktop$ ./task5
3b784697e757203ca48944c492f46d17
[09/27/22]seed@VM:~/Desktop$ ./task5
829149703499457474b3012daa56abfc
[09/27/22]seed@VM:~/Desktop$
```