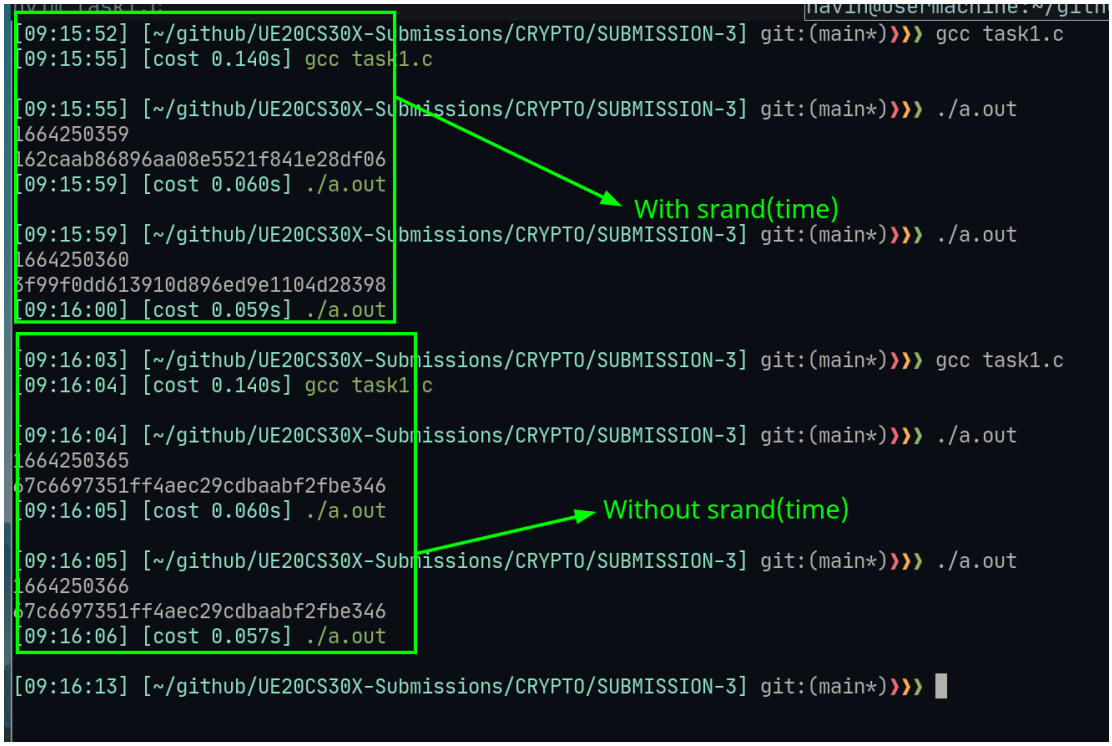
Cryptograhphy Hands-On submission 3

Details :

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

Task 1 : Generating prng the wrong way

Screenshot :



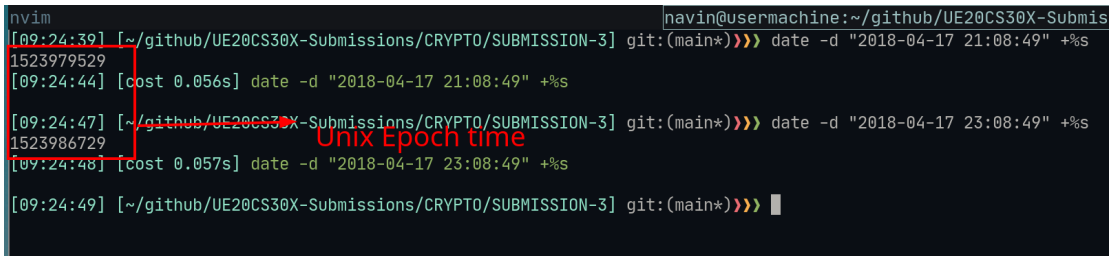
Observation :

- The time stamp is being used as key for the
- If srand(time()) is removed, then the generated number remains the same

Task 2 : Guessing the key

Step 1 : getting UNIX time to bruteforce decryption

Screenshot :



Observation :

- Above two are the UNIX epoch timestamps using which we generate all possible keys for decrypting
- The above two are different from the ones shown in video as I am using IST and not UTC on the contrary. Do note the times in code is following UTC.

Step 2 : Bruteforce decrypting

Code :

```c
/* task2.c */
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#define KEYSIZE 16
void main() {
  int i, j;
  FILE *f;
  char key[KEYSIZE];
  int value1, value2;
  value1 = 1524013729;
  value2 = 1524020929;
  f = fopen("keys.txt", "w");
  for (j = value1; j <= value2; j++) {
    srand (j);
    for (i = 0; i< KEYSIZE; i++) {
      key[i] = rand()%256;
      fprintf(f, "%.2x", (unsigned char)key[i]);
    }
    fprintf(f,"\n");
  }
}
```

Screenshot :

```
[09:32:52] [~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-3] git:(main*)❯❯❯ gcc task2.c
[09:32:54] [cost 0.140s] gcc task2.c

[09:32:54] [~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-3] git:(main*)❯❯❯ ./a.out
[09:32:57] [cost 0.086s] ./a.out

[09:32:58] [~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-3] git:(main*)❯❯❯ cat keys.txt | head 5
head: cannot open '5' for reading: No such file or directory
[09:33:07] [cost 0.055s] cat keys.txt | head 5

[09:33:25] [~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-3] git:(main*)❯❯❯ head -n 5 keys.txt
f243bb05ec67c06c7a6a820137a98789
7ed87572ab02a5fa85aac9244eaa19a8
701ce20ebe9ccf0cfd17f805ee9369af
c99e333e665ce424a0ac1682d5121b5c
c29a311f4233068c4019b3330bbe56a2
[09:33:29] [cost 0.057s] head -n 5 keys.txt

[09:33:30] [~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-3] git:(main*)❯❯❯
```

```
[09:40:23] [~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-3] git:(main*)❯❯❯ wc -l keys.txt
7201 keys.txt
[09:40:26] [cost 0.069s] wc -l keys.txt

[09:40:27] [~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-3] git:(main*)❯❯❯
```

Observation :

- Here we have generated all possible keys (of which few are seen in the Screenshot) that are possible between the two time stamps
- We also see we have 7200 keys which we will be using for bruteforce

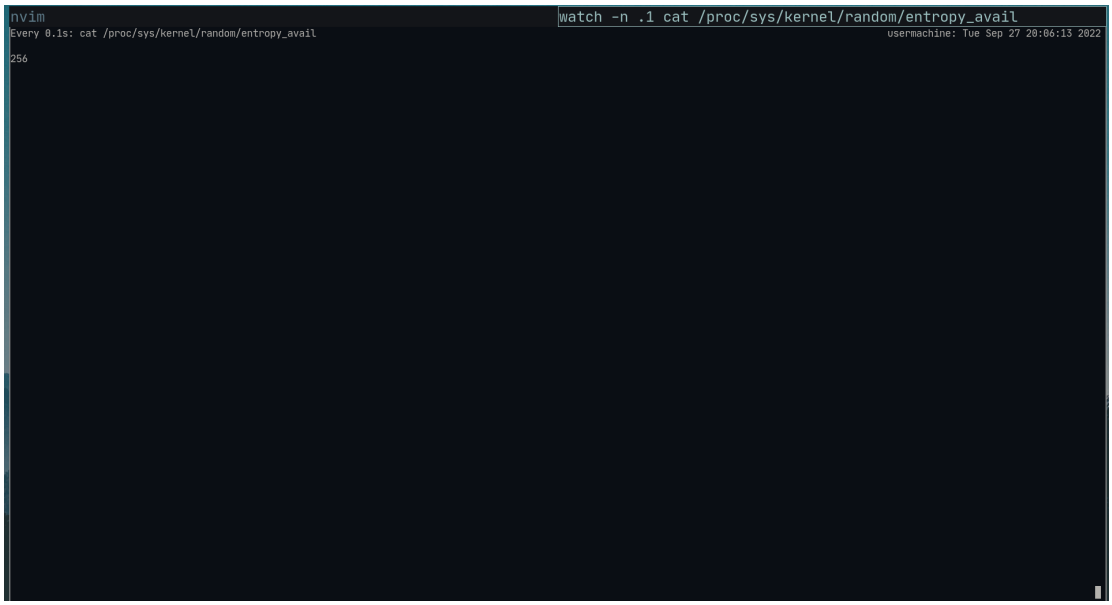
Step 3 : Breaking encrpyted text

Screenshot :

```
Encrypted:  aac4b8af02903130bd4af52facc4ddfa
Encrypted:  18689ed3655a69d5bd96b3035b3eaeee
Encrypted:  845272c5bae60ddbfc358a4d4cfc733a
Encrypted:  d75b4f283a7961979e33cd718e074ada
Encrypted:  07b912cc5244db291f274ebca4f45268
Encrypted:  c92d7aa443327892fb20a74c3d31eb18
Encrypted:  39e3d058c81a41aabc4dfb2b88e71aab
Encrypted:  8b29cc48852675fa11162be89af8e6f7
[19:45:37] [cost 0.969s] python3 task3.py

[19:45:55] [~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-3] git:(main*)❯❯❯ python3 task3.py

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

[19:45:56] [cost 0.770s] python3 task3.py

[19:46:14] [~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-3] git:(main*)❯❯❯
```

task 3 : Seeing kernel entropy :

Screenshot :

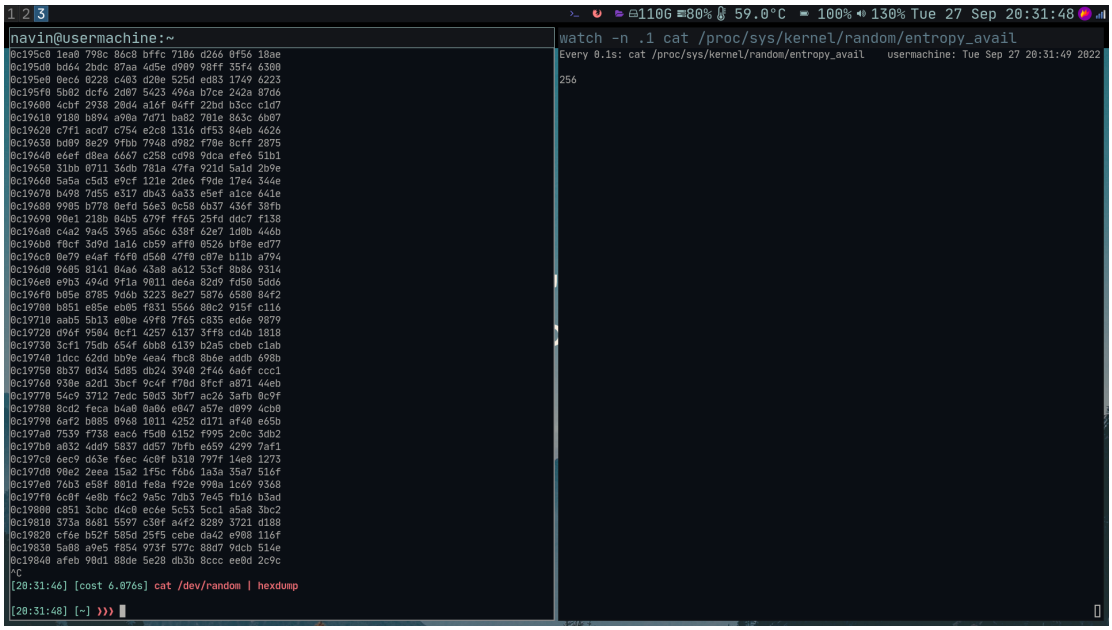


Observation :

- The change in entropy is too little and very rapid

Task 4 : /dev/urandom :

Screenshot :



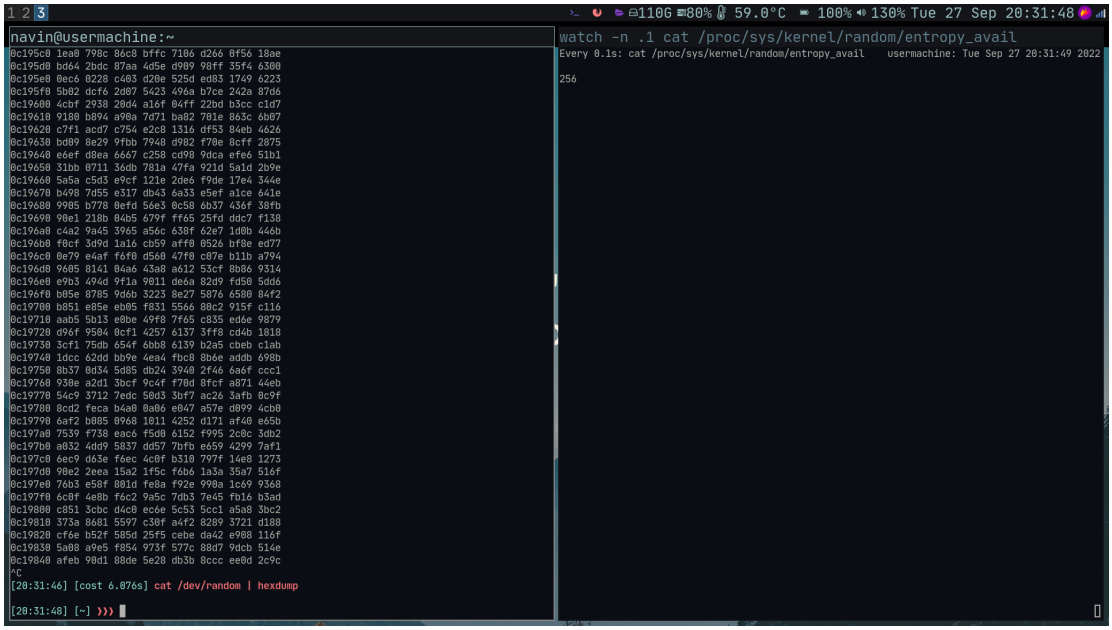
Observation :

- On mouse movement we see large change in generate hex dump

Task 5 : /dev/urandom :

Step 1 :

Screenshot :



Observation :

- On mouse movement we see large change in generate hex dump

Step 2 : Generate random and check quality

Screenshot :

Observation :

- Here we stats about the random number generated through /dev/urandom

Step 3 : Using dev/uradom in task 1 :

Screenshot :

```
[20:47:31] [cost 43.554s] yay -Syy ent

[20:47:32] [~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-3] git:(main*)❯❯❯ head -c 1M /dev/urandom > output.bin
[20:47:40] [cost 0.059s] head -c 1M /dev/urandom > output.bin

[20:47:45] [~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-3] git:(main*)❯❯❯ ent output.bin
Entropy = 7.999817 bits per byte.

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

Chi square distribution for 1048576 samples is 265.64, and randomly
would exceed this value 31.06 percent of the times.

Arithmetic mean value of data bytes is 127.4850 (127.5 = random).
Monte Carlo value for Pi is 3.144344880 (error 0.09 percent).
Serial correlation coefficient is 0.003824 (totally uncorrelated = 0.0).
[20:47:46] [cost 0.084s] ent output.bin

[20:50:28] [~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-3] git:(main*)❯❯❯ gcc task1.c
[20:50:32] [cost 0.126s] gcc task1.c

[20:50:32] [~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-3] git:(main*)❯❯❯ ./a.out
a9581c6cd00f079dcca4eaf636e0c2e2
[20:50:34] [cost 0.048s] ./a.out

[20:50:39] [~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-3] git:(main*)❯❯❯ ./a.out
45ddd23dd66f20cb8b359f7557b6cdeb
[20:50:40] [cost 0.051s] ./a.out

[20:50:40] [~/github/UE20CS30X-Submissions/CRYPTO/SUBMISSION-3] git:(main*)❯❯❯ ./a.out
ce901e7865da2f5087bb4eec6d3205db
[20:50:41] [cost 0.050s] ./a.out
```

Observation :

- Here we see that the generated ranomd number done see to follow a patterns unlike before when we did using timestamps.