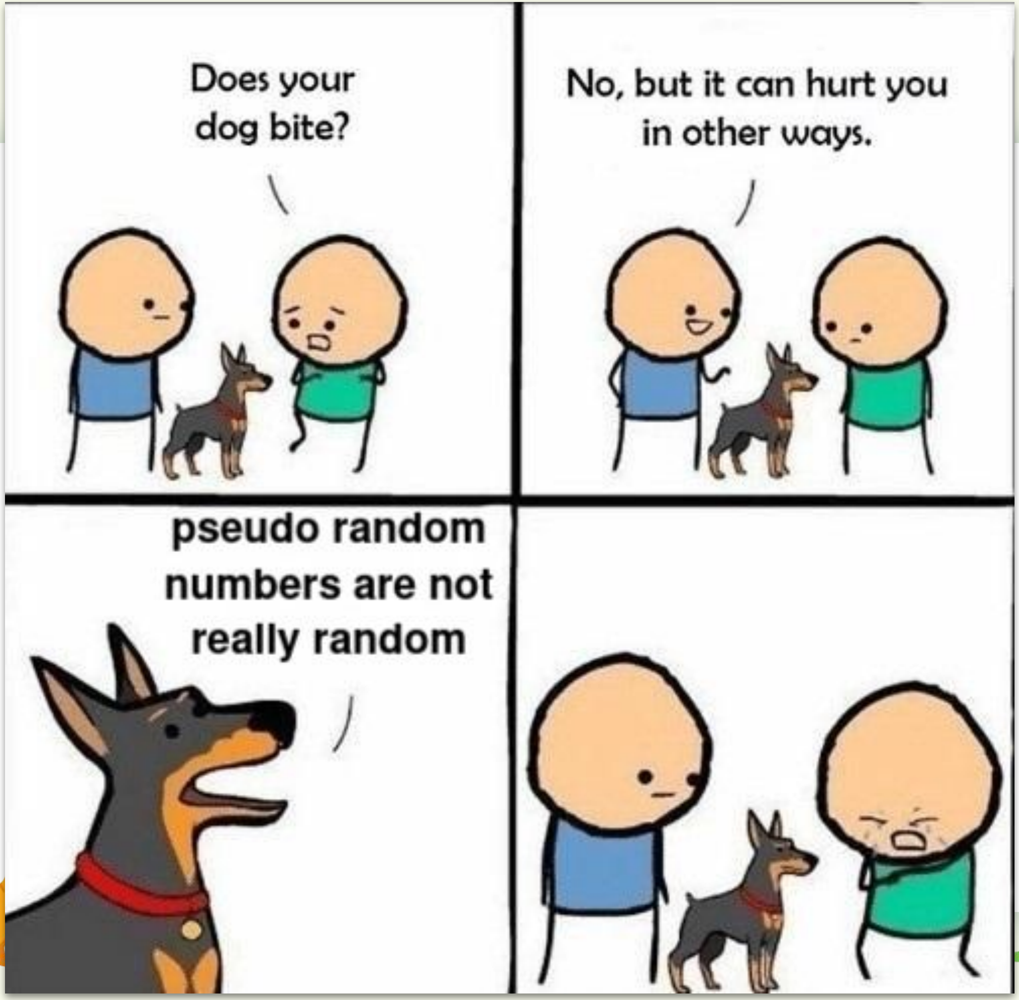

Does your dog bite?
No, but it can hurt you in other ways.
pseudo random numbers are not really random

0x0POSEC 0x0D - The Meet Challenge Write-up

Zezadas

✉ zezadas@sefod.eu

🌐 https://sefod.eu

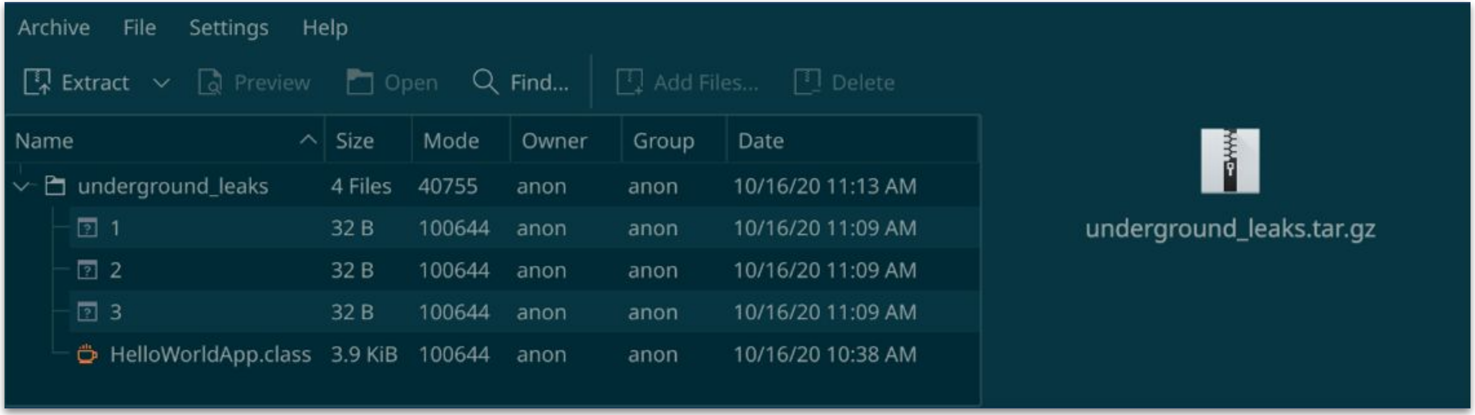
🐦 @0xz3z4d45

Rui Tinto admitted having some information about the snitch from the 0XOPOLEAKS. However, he's unwilling to give the passwords unless his demands of having Kripthor on 0xOPOSEC get fulfilled. Meanwhile, government forces tried to crack the passwords with no success.

We know from underground sources that Rui Tinto does cipher each file individually for higher security and privacy. We manage to put our hands on his top-notch closed source software used for encrypting the information.

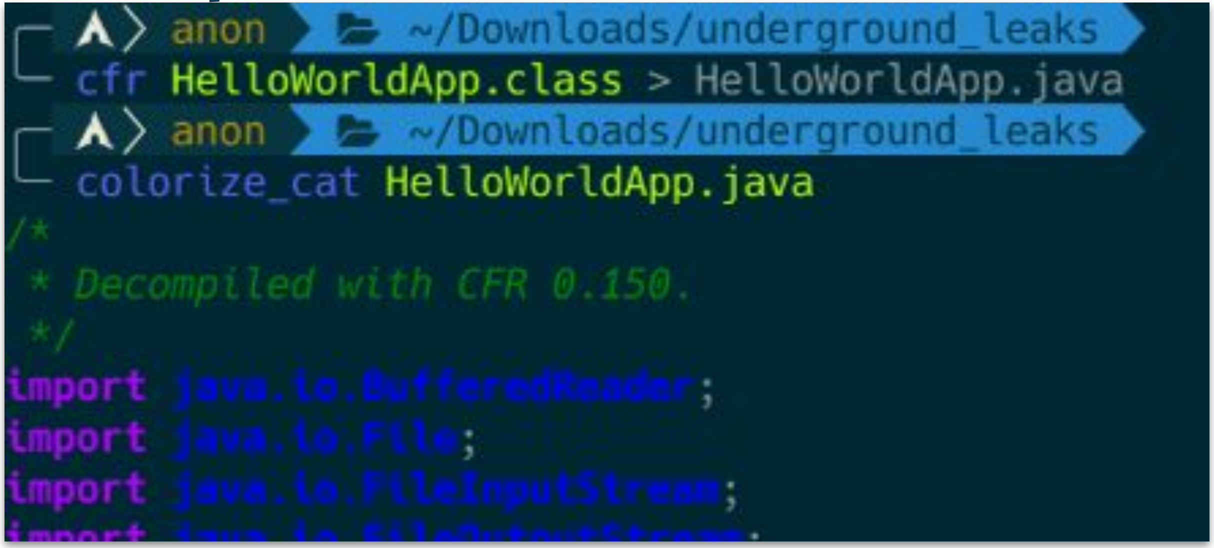
Leak: https://sefod.eu/oposec/underground_leaks.tar.gz

Underground Leaks



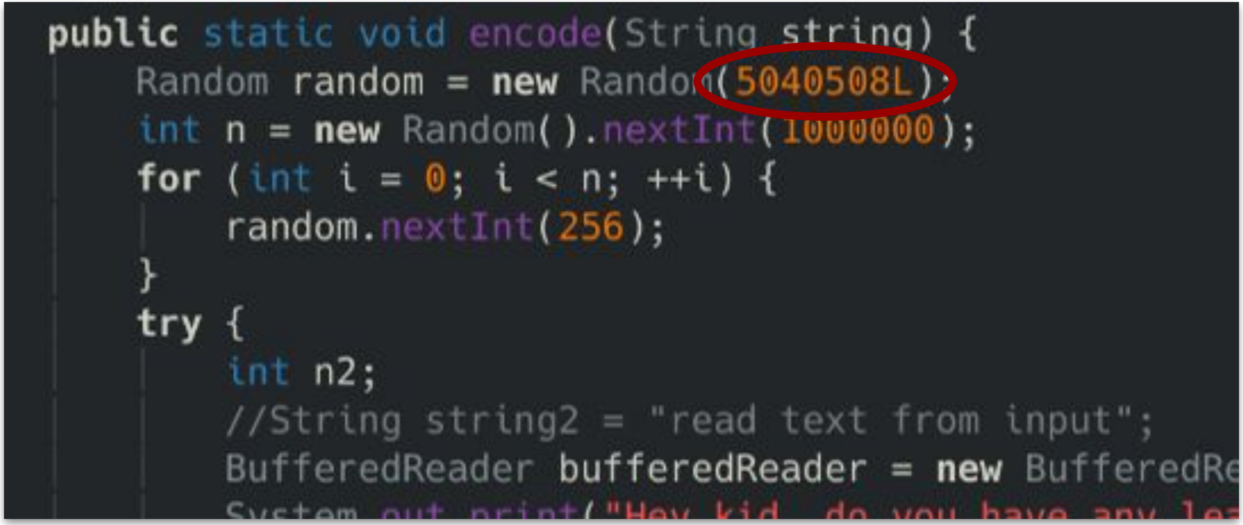
Name	Size	Mode	Owner	Group	Date
underground_leaks	4 Files	40755	anon	anon	10/16/20 11:13 AM
1	32 B	100644	anon	anon	10/16/20 11:09 AM
2	32 B	100644	anon	anon	10/16/20 11:09 AM
3	32 B	100644	anon	anon	10/16/20 11:09 AM
HelloWorldApp.class	3.9 KiB	100644	anon	anon	10/16/20 10:38 AM

CFR Decompiler



```
┌ A〉 anon  ~/Downloads/underground_leaks
└ cfr HelloWorldApp.class > HelloWorldApp.java
┌ A〉 anon  ~/Downloads/underground_leaks
└ colorize_cat HelloWorldApp.java
/*
 * Decompiled with CFR 0.150.
 */
import java.io.BufferedReader;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
```

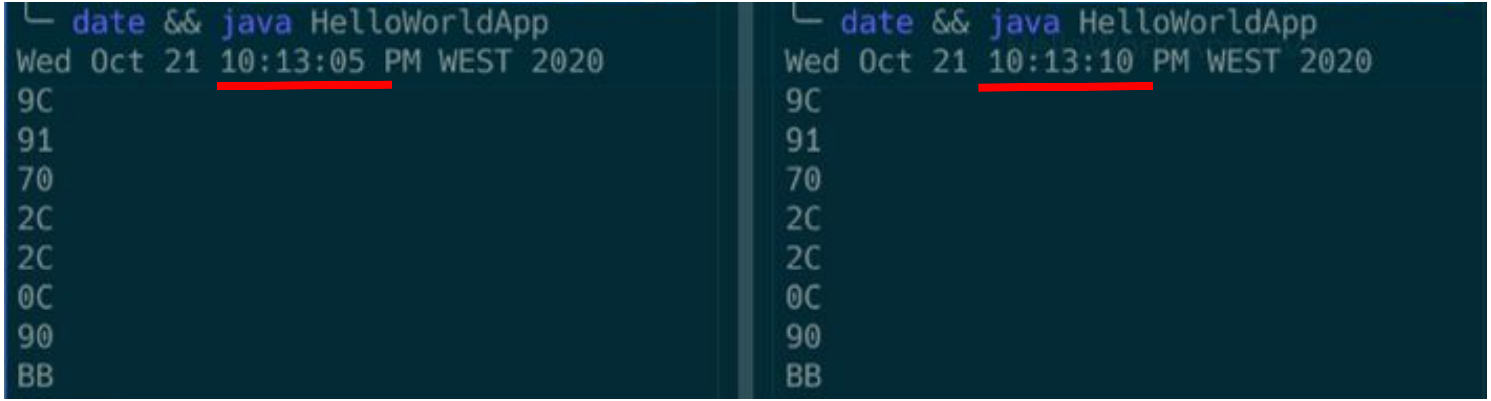
Reverse



```
public static void encode(String string) {
    Random random = new Random(5040508L);
    int n = new Random().nextInt(1000000);
    for (int i = 0; i < n; ++i) {
        random.nextInt(256);
    }
    try {
        int n2;
        //String string2 = "read text from input";
        BufferedReader bufferedReader = new BufferedRe
        System.out.print("Hey kid, do you have any lea
```

https://youtu.be/GYvJEFkpCbo?t=91

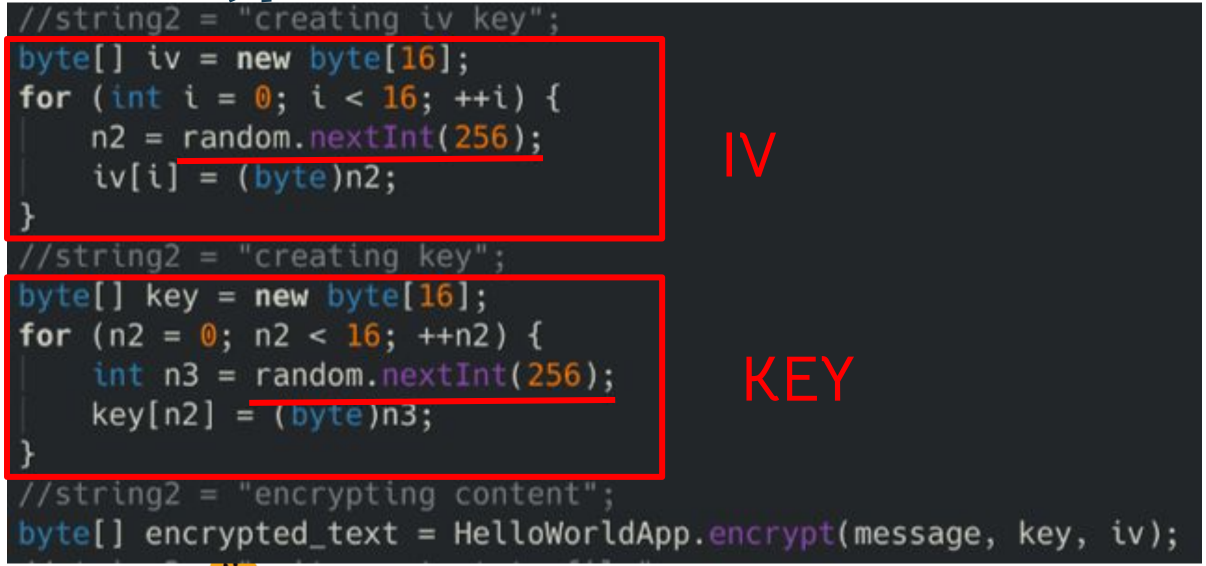
Java Pseudo Random



```
└ date && java HelloWorldApp
Wed Oct 21 10:13:05 PM WEST 2020
9C
91
70
2C
2C
0C
90
BB
```

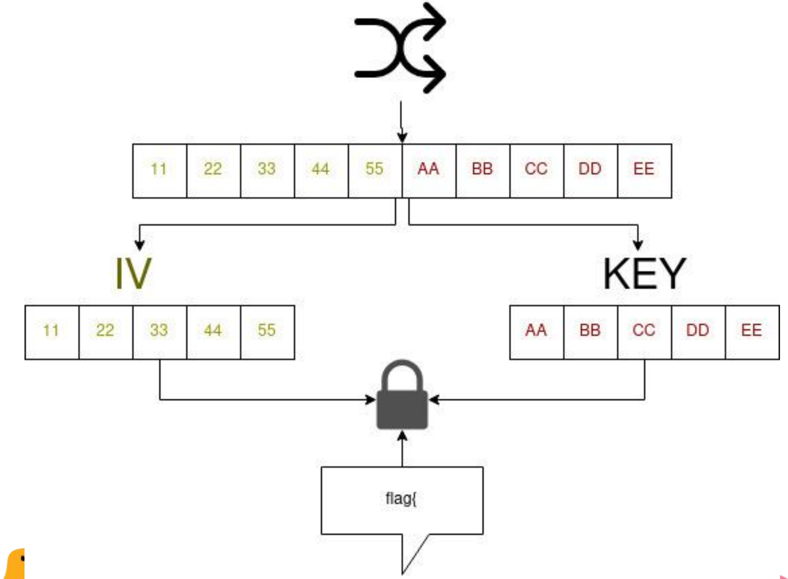
```
└ date && java HelloWorldApp
Wed Oct 21 10:13:10 PM WEST 2020
9C
91
70
2C
2C
0C
90
BB
```

Broken Encryption

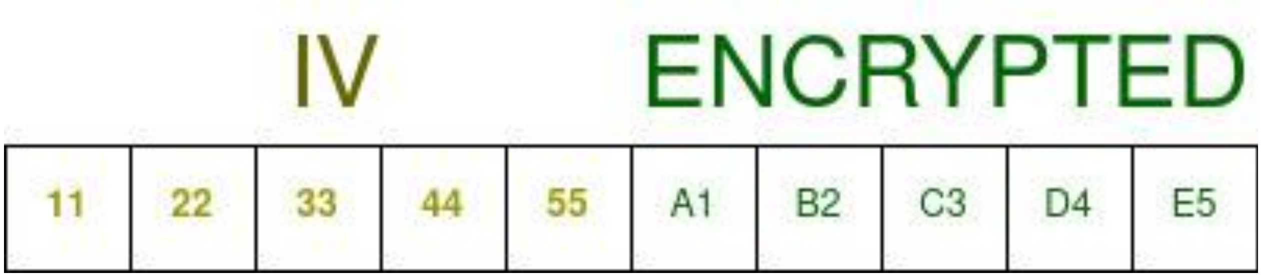


```
//string2 = "creating iv key";
byte[] iv = new byte[16];
for (int i = 0; i < 16; ++i) {
    n2 = random.nextInt(256);
    iv[i] = (byte)n2;
}
//string2 = "creating key";
byte[] key = new byte[16];
for (n2 = 0; n2 < 16; ++n2) {
    int n3 = random.nextInt(256);
    key[n2] = (byte)n3;
}
//string2 = "encrypting content";
byte[] encrypted_text = HelloWorldApp.encrypt(message, key, iv);
```

AES Encryption

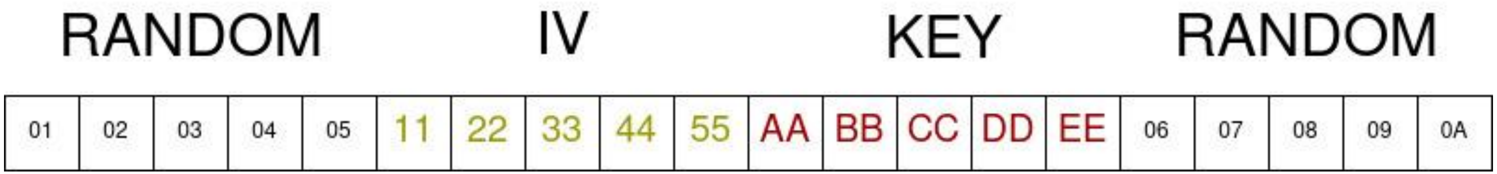


Encrypted File

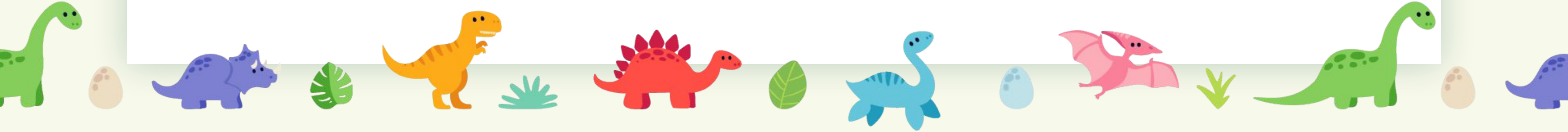


IV					ENCRYPTED				
11	22	33	44	55	A1	B2	C3	D4	E5

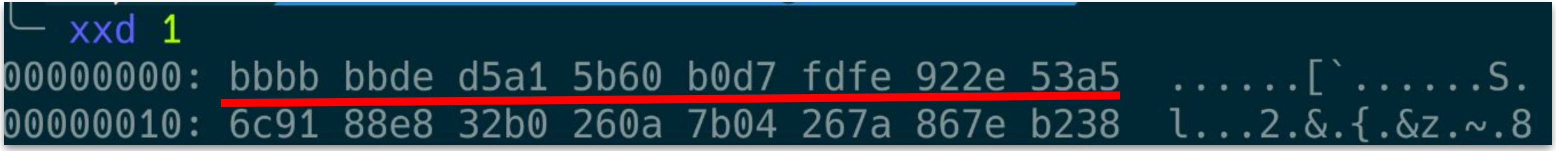
Solver



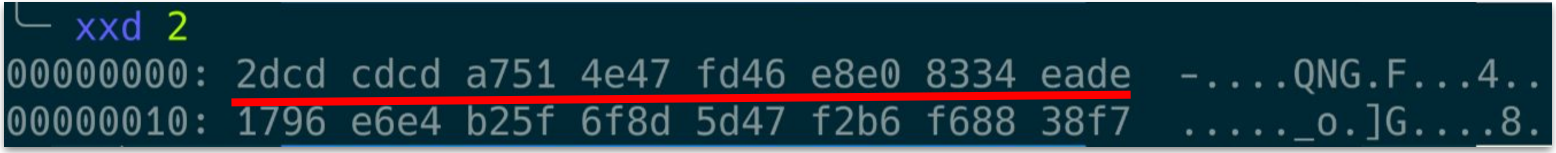
RANDOM					IV					KEY					RANDOM				
01	02	03	04	05	11	22	33	44	55	AA	BB	CC	DD	EE	06	07	08	09	0A



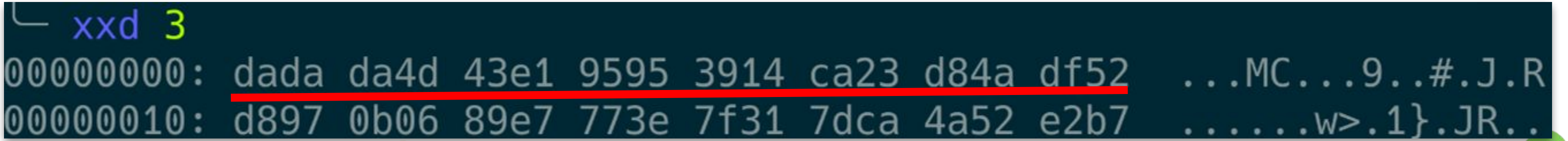
Encrypted Files



```
└─ xxd 1
00000000: bbbb bbde d5a1 5b60 b0d7 fdfe 922e 53a5  ......[`......S.
00000010: 6c91 88e8 32b0 260a 7b04 267a 867e b238  l...2.&.{.&z.~.8
```

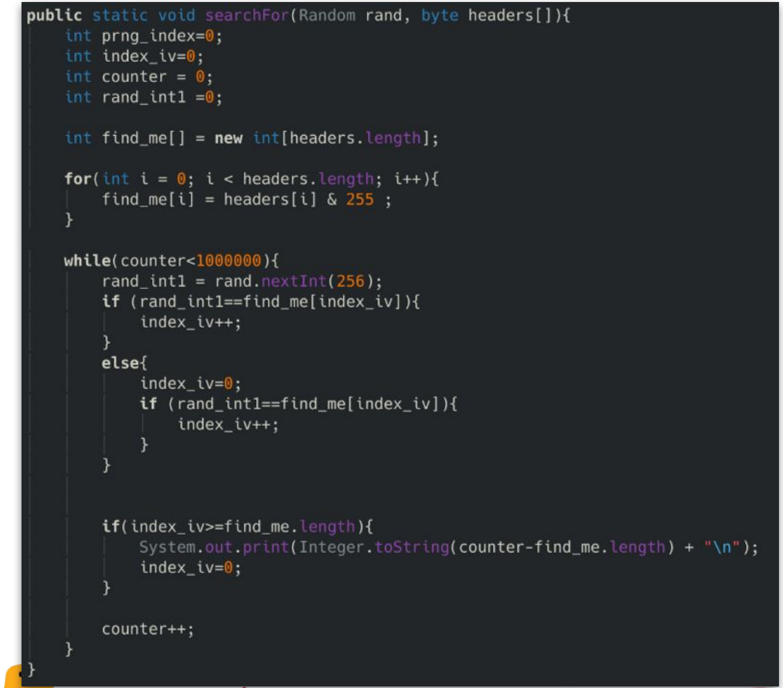


```
└─ xxd 2
00000000: 2dcd cdcd a751 4e47 fd46 e8e0 8334 eade  -....QNG.F...4..
00000010: 1796 e6e4 b25f 6f8d 5d47 f2b6 f688 38f7  ....._o.]G....8.
```



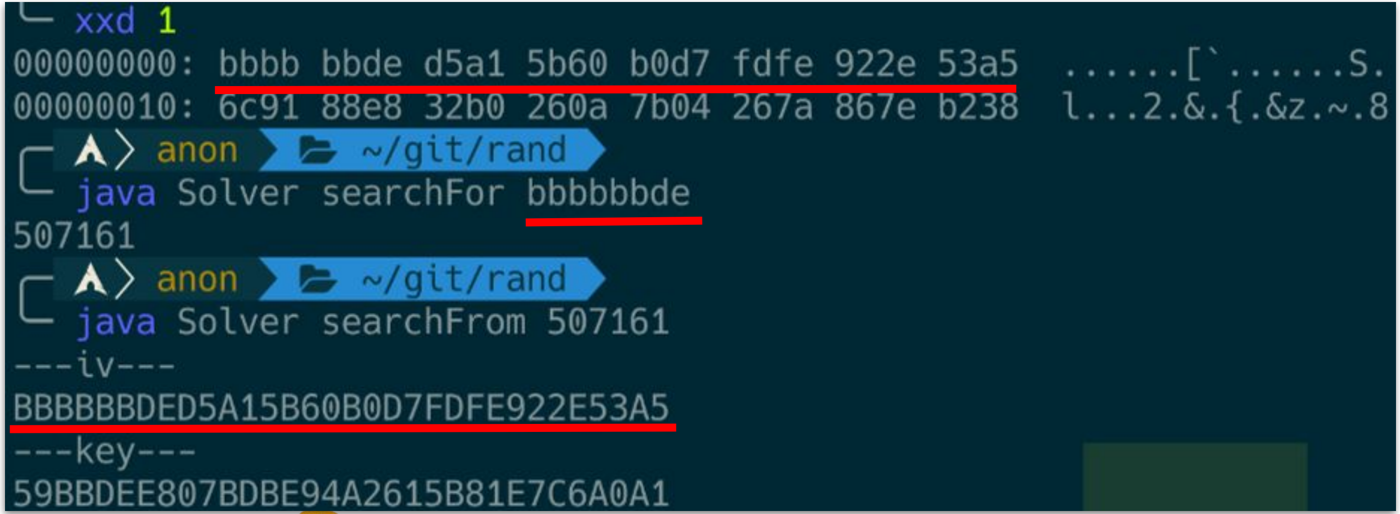
```
└─ xxd 3
00000000: dada da4d 43e1 9595 3914 ca23 d84a df52  ...MC...9..#.J.R
00000010: d897 0b06 89e7 773e 7f31 7dca 4a52 e2b7  ......w>.1}.JR..
```

Solver



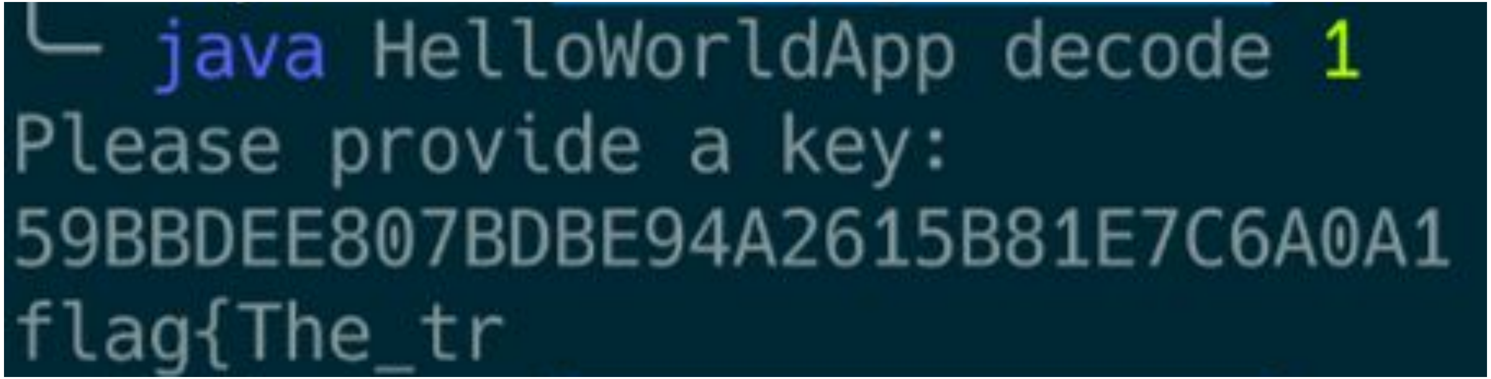
```java
public static void searchFor(Random rand, byte headers[]){
    int prng_index=0;
    int index_iv=0;
    int counter = 0;
    int rand_int1 =0;

    int find_me[] = new int[headers.length];

    for(int i = 0; i < headers.length; i++){
        find_me[i] = headers[i] & 255 ;
    }

    while(counter<1000000){
        rand_int1 = rand.nextInt(256);
        if (rand_int1==find_me[index_iv]){
            index_iv++;
        }
        else{
            index_iv=0;
            if (rand_int1==find_me[index_iv]){
                index_iv++;
            }
        }


        if(index_iv>=find_me.length){
            System.out.print(Integer.toString(counter-find_me.length) + "\n");
            index_iv=0;
        }

        counter++;
    }
}
```

Retrieving Key



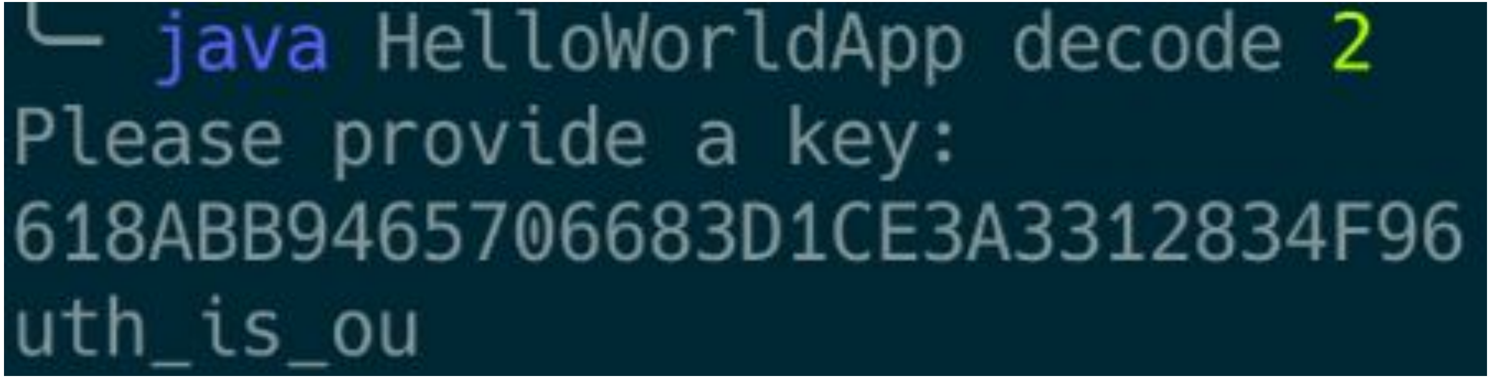
```
└ xxd 1
00000000: bbbb bbde d5a1 5b60 b0d7 fdfe 922e 53a5  ......[`......S.
00000010: 6c91 88e8 32b0 260a 7b04 267a 867e b238  l...2.&.{.&z.~.8
┌ A> anon  ~/git/rand
└ java Solver searchFor bbbbbbde
507161
┌ A> anon  ~/git/rand
└ java Solver searchFrom 507161
---iv---
BBBBBBDED5A15B60B0D7FDFE922E53A5
---key---
59BBDEE807BDBE94A2615B81E7C6A0A1
```

Flag_1



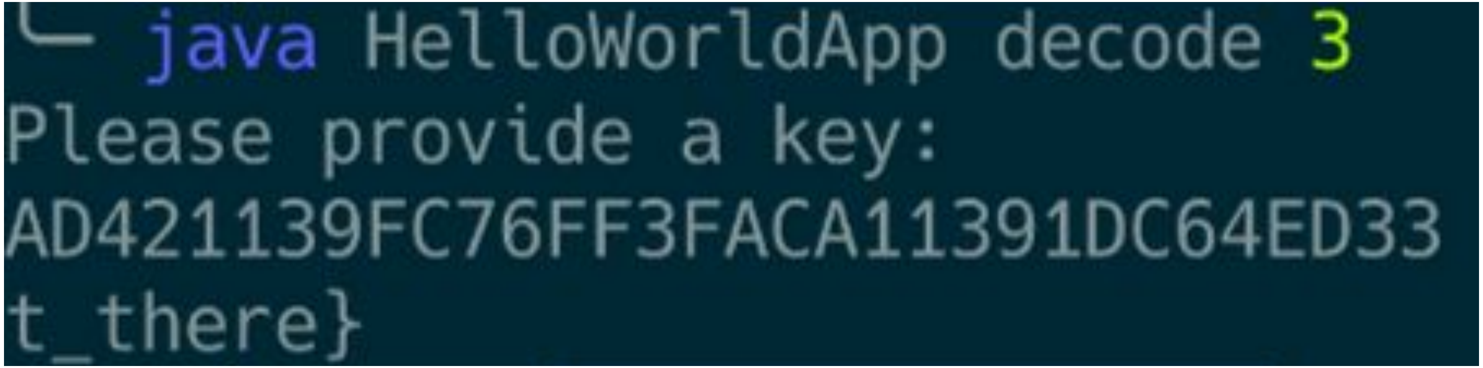
```
└─ java HelloWorldApp decode 1
Please provide a key:
59BBDEE807BDBE94A2615B81E7C6A0A1
flag{The_tr
```

Flag_2



```
└─ java HelloWorldApp decode 2
Please provide a key:
618ABB9465706683D1CE3A3312834F96
uth_is_ou
```

Flag_3

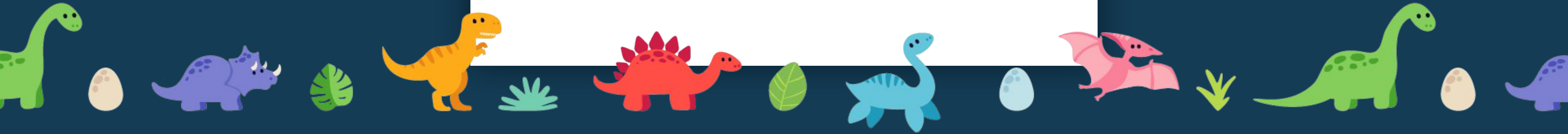


```
└─  java HelloWorldApp decode 3
Please provide a key:
AD421139FC76FF3FACA11391DC64ED33
t_there}
```

flag{The_truth_is_out_there}

Solvers

- @nunohumberto (First Blood!)
- @Guillaume
- @jp
- @mluis
- @miguelpduarte
- @k414x
- @ines



https://github.com/zezadas/0xOPOSEC_0xOD_PSEUDO_RANDOM

Thanks!

Any questions?

You can find me at
@0xz3z4d45 and
zezadas@sefod.eu

- .... / .- .. .. -. . / .. ... / -. --- - / ... .-
..-. . .-.-.- / .-- .- .. - / ..-. --- .-. / - . . . --
/ ..-. ...- .-. ..- -.-.- ..-.- --- -.... . .-.-.- / ..-.
.-.. .- --. .-..-. ----- -..-- -- .. - .--- -.-- .. .-.. -.. -.--...
... . -.-.-. .. ... .-.-.-... -.. .-.-- ...-- .. .. -..-.