Reversing 102 (Easy) - Rev

Sunday, June 11, 2023 12:47 AM

Description

"Uncover the hidden secrets of the binary underworld and decode the flag that lies within."

Solution

Running the file.

```
(kali@ kali)-[~/_/rev/unused/reversing102_r2.0/challenge]
$ file reversing102
reversing102: ELF 64-bit LSB pie executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, BuildID[sha1]=4bb9686dc21373f8b7a6
a00f5eb6ec4cc095b984, for GNU/Linux 3.2.0, not stripped
```

Running the binary.

Nothing pops out in Itrace either.

Ghidra.

Upon decompilation on ghidra, we see some variable initializations and a XOR operation being performed in a loop.

```
local_128[0] = 0xdede;
local_128[1] = 0xded8;
local_128[2] = 0xde9d;
local_128[3] = 0xde9e;
local_118 = 0xdedf;
local_114 = 0xdede;
local_110 = 0xde9e;
local_10c = 0xdece;
local_108 = 0xdedf;
local_104 = 0xde9e;
local_100 = 0xded9;
```

```
local 1c = 0xb;
 for (local_c = 0; local_c < local_1c; local_c = local_c + 1) {
  local_128[local_c] = local_128[local_c] ^ 0xdead;
Further ahead, there's our comparison for the secret key.
printf("Enter the secret key: ");
  isoc99 scanf(&DAT 001021a5,local 88);
 sVar1 = strlen(local_88);
 if (sVar1 == 0xb) {
  for (local_10 = 0; local_10 < 0xb; local_10 = local_10 + 1) {
   if ((int)local_88[local_10] != local_128[local_10]) {
    printf("Access Denied! You\'re not allowed in here.");
   }
,,,}
Notice that it checks each character of our input with secret key that is local_128.
To get the actual secret key, we can use the following script.
key = [0xdede,0xded8,0xdedd,0xde9e,0xdedf,0xdede,0xde9e,0xdece,0xdedf,0xde9e,0xded9]
for i in range(len(key)):
      print(chr(key[i] ^ 0xdead), end="")
```

We get the secret.

```
(kali@ kali)-[~/.../rev/unused/reversing102_r2.0/original-code]
$ python3 script.py
sup3rs3cr3t
```

Enter the secret in the binary and you'll get the flag.



Flag: NCC{1ts_0nly_th3_b3ginning}