**Guess My Number**

***Misc***

**Challenge:**

My friend said if i can guess the right number, he will give me something. Can you help me?

nc ctf.tcp1p.com 7331

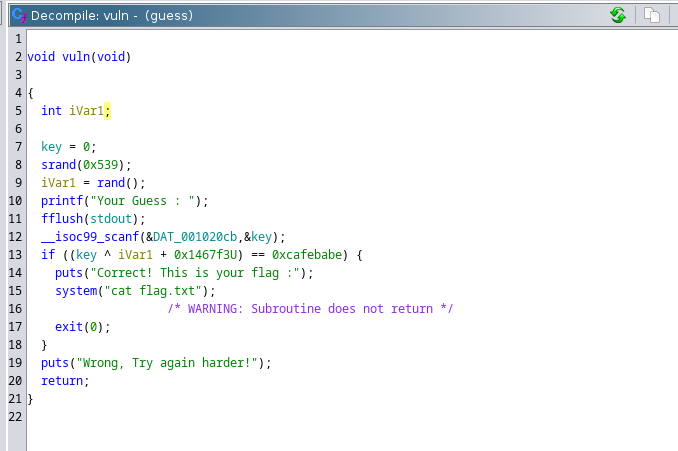
Attached - dist.zip

**Approach:**

The zip file contained a single file named ‘guess’ which looks like a ELF executable file:



We open it up in ghidra and navigate to the function that lokos interesting:



If we read through it, we’ll see it asks the user for a ‘guess’ and then performs some operation on the number entered, then it compares the result with a hex number, if it matches, then it prints the flag.

Let’s debug it step by step

First of all, 0xcafebabe is the hex number that it compares to at the end, which translates to 3405691582 in the decimal system, so that’s the number we need to reach at the end.

There’s a variable iVar1 that we’ll require the value of so let’s check that out first.

The iVar1 variable is generated by the random function using the rand() function, so its actually a random number??

Not really, if we check the previous line, we see:

“srand(0x539);”

What this function does is set a seed for the random method, in short, setting a seed sets the starting point of the random number generator. This means that the first number generated by the random generator could be calculated.

For this we open a online C compiler, use the srand(0x539) and generate a random (not so random) number, and we get the number:

292616681

Now just basic maths:

Input ^ (292616681 + 0x1467f3) = 0xcafebabe

Input ^ (292616681 + 1337331) = 3405691582

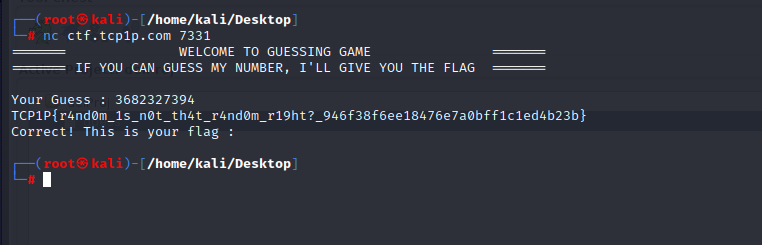
Input ^ (292616681 + 1337331) ^ (292616681 + 1337331) = 3405691582 ^ (292616681 + 1337331)

Input = 3405691582 ^ (292616681 + 1337331)

Input = 3405691582 ^ (293954012)

Input = 3682327394

We connect using the netcat client and enter the number, and boom!



**Flag: TCP1P{r4nd0m\_1s\_n0t\_th4t\_r4nd0m\_r19ht?\_946f38f6ee18476e7a0bff1c1ed4b23b}**

Congrats!!

Happy Hacking!