Binary-гаа ажилуулна.

```
┌──(user㉿user)-[~/shared/nest/anti_debug]
└─$ ./nest
Flag: asafdas
Wrong!
```

Source code-ийг харахад гараас оруулж буй утгыг dummy_function-ийн утгатай тулгаж байна.

```
lStack_10 = *(long *)(in_FS_OFFSET + 0x28);
uStack_78 = 0;
uStack_70 = 0;
uStack_68 = 0;
uStack_60 = 0;
uStack_58 = 0;
uStack_50 = 0;
uStack_48 = 0;
uStack_40 = 0;
uStack_38 = 0;
uStack_30 = 0;
uStack_28 = 0;
uStack_20 = 0;
uStack_18 = 0;
lVar2 = ptrace(0,0,0,0);
if (lVar2 == -1) {
  puts("Debugging detected! Exiting...");
  FUN_001010a0(1);
}
dummy_function(&uStack_78);
printf("Flag: ");
fgets(auStack_e8,100,stdin);
lVar2 = strcspn(auStack_e8,&DAT_001020be);
auStack_e8[lVar2] = 0;
iVar1 = strcmp(auStack_e8,&uStack_78);
if (iVar1 == 0) {
  printf("Correct!");
}
else {
  printf("Wrong!");
}
uVar3 = 0;
if (lStack_10 != *(long *)(in_FS_OFFSET + 0x28)) {
  uVar3 = __stack_chk_fail();
}
return uVar3;
```

Үүнд strcmp функцийг ашигласан байх ба энэ нь library trace хийхэд харьцуулж буй 2 утгыг харж болох функц юм.

```
┌──(user㉿user)-[~/shared/nest/anti_debug]
└─$ ltrace ./nest
ptrace(0, 0, 0, 0)                                                    = 31
puts("Debugging detected! Exiting..."Debugging detected! Exiting...
)
exit(1 <no return ...>
+++ exited (status 1) +++
```

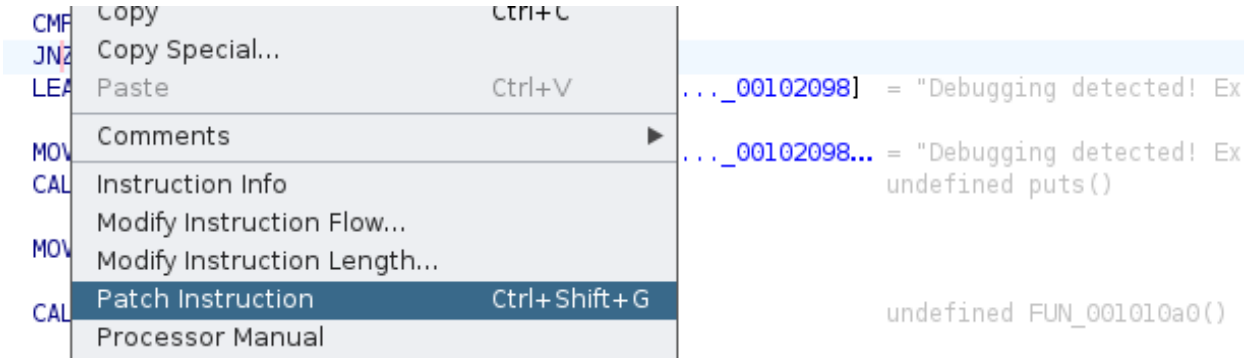
Ltrace ./nest гэхэд Ptrace debug хийх процессийг хориглож байгаа.

Ptrace-ийг болиулхийн тулд binary patch аргийг ашиглана.

```
                00 00
0010123e b8 00 00       MOV         EAX,0x0
                00 00
00101243 e8 48 fe       CALL        <EXTERNAL>::ptrace                          undefined ptrace()
                ff ff
00101248 48 83 f8 ff    CMP         RAX,-0x1
0010124c 75 19          JNZ         LAB_00101267
0010124e 48 8d 05       LEA         RAX,[s_Debugging_detected!_Exiting..._00102098]  = "Debugging detected! Ex
                43 0e 00 00
00101255 48 89 c7       MOV         RDI=>s_Debugging_detected!_Exiting..._00102098... = "Debugging detected! Ex
00101258 e8 d3 fd       CALL        <EXTERNAL>::puts                            undefined puts()
                ff ff
0010125d bf 01 00       MOV         EDI,0x1
                00 00
00101262 e8 39 fe       CALL        FUN_001010a0                                undefined FUN_001010a0()
                ff ff
```

Ptrace call дээр дарж дараах Instruction-уудаас JNZ -> JMP болгон солино.

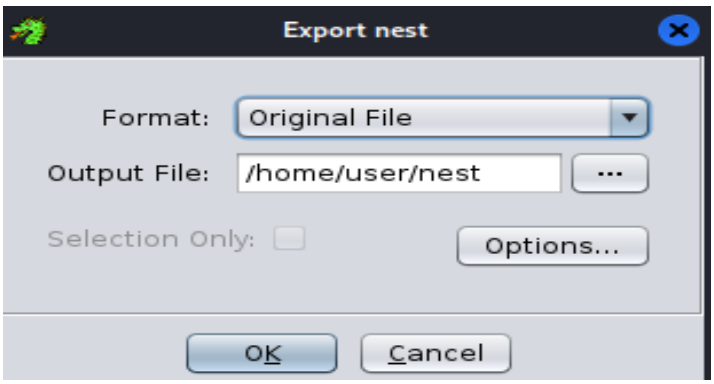
Ghidra дээр Patch хийх гэж байгаа instruction дээр дарж Patch instruction дээр дарна.



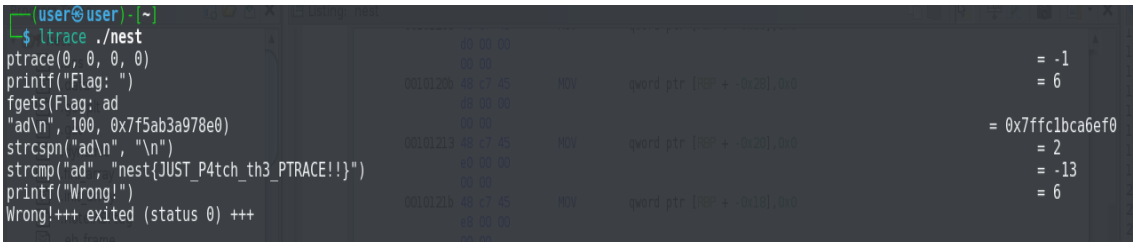
JNZ -> JMP болгоно.

```
                00 00
00101243 e8 48 fe       CALL        <EXTERNAL>::ptrace                          undefined ptrace()
                ff ff
00101248 48 83 f8 ff    CMP         RAX,-0x1
0010124c eb 19          JMP         LAB_00101267
0010124e 48 8d 05       LEA         RAX,[s_Debugging_detected!_Exiting..._00102098]  = "Debugging detected! Ex
                43 0e 00 00
00101255 48 89 c7       MOV         RDI=>s_Debugging_detected!_Exiting..._00102098... = "Debugging detected! Ex
00101258 e8 d3 fd       CALL        <EXTERNAL>::puts                            undefined puts()
                ff ff
0010125d bf 01 00       MOV         EDI,0x1
                00 00
00101262 e8 39 fe       CALL        FUN_001010a0                                undefined FUN_001010a0()
                ff ff
```

Patch хийсэн файлаа хадгалахдаа зүүн дээд буланд байгаа File -> Export Program -> Original file



Patch хийсэн файлаа ltrace ашилан ажилуулж тугаа авна.



Туг: **nest{JUST_P4tch_th3_PTRACE!!}**