在bomb的反汇编代码内找到调用secret\_phase的函数居然是phase\_defused

Dump of assembler code for function phase\_defused:

0x0000000000401621 <+0>: sub $0x68,%rsp

0x0000000000401625 <+4>: mov $0x1,%edi

0x000000000040162a <+9>: callq 0x4013ae <send\_msg>

0x000000000040162f <+14>: cmpl $0x6,0x203556(%rip) # 0x604b8c <num\_input\_strings>

0x0000000000401636 <+21>: jne 0x4016a5 <phase\_defused+132>

0x0000000000401638 <+23>: lea 0x10(%rsp),%r8

0x000000000040163d <+28>: lea 0x8(%rsp),%rcx

0x0000000000401642 <+33>: lea 0xc(%rsp),%rdx

0x0000000000401647 <+38>: mov $0x4026ad,%esi

0x000000000040164c <+43>: mov $0x604c90,%edi

0x0000000000401651 <+48>: mov $0x0,%eax

0x0000000000401656 <+53>: callq 0x400b50 <\_\_isoc99\_sscanf@plt>

0x000000000040165b <+58>: cmp $0x3,%eax

0x000000000040165e <+61>: jne 0x401691 <phase\_defused+112>

0x0000000000401660 <+63>: mov $0x4026b6,%esi

x/s 0x4026b6

0x4026b6: "DrEvil"

所以phase\_defused应该是读入了字符，判断是否等于DrEvil，相等的话就进入隐藏关卡

于是我们在phase\_defused中寻找是否有读入的指令，发现上面相邻处有

0x0000000000401647 <+38>: mov $0x4026ad,%esi

(gdb) x/s 0x4026ad

0x4026ad: "%d %d %s"

0x0000000000401665 <+68>: lea 0x10(%rsp),%rdi

0x000000000040166a <+73>: callq 0x4012af <strings\_not\_equal>

0x000000000040166f <+78>: test %eax,%eax

0x0000000000401671 <+80>: jne 0x401691 <phase\_defused+112>

0x0000000000401673 <+82>: mov $0x402528,%edi

0x0000000000401678 <+87>: callq 0x400a90 <puts@plt>

---Type <return> to continue, or q <return> to quit---

0x000000000040167d <+92>: mov $0x402550,%edi

0x0000000000401682 <+97>: callq 0x400a90 <puts@plt>

0x0000000000401687 <+102>: mov $0x0,%eax

0x000000000040168c <+107>: callq 0x4011c5 <secret\_phase>

0x0000000000401691 <+112>: mov $0x402588,%edi

0x0000000000401696 <+117>: callq 0x400a90 <puts@plt>

0x000000000040169b <+122>: mov $0x4025b8,%edi

0x00000000004016a0 <+127>: callq 0x400a90 <puts@plt>

0x00000000004016a5 <+132>: add $0x68,%rsp

0x00000000004016a9 <+136>: retq

End of assembler dump.