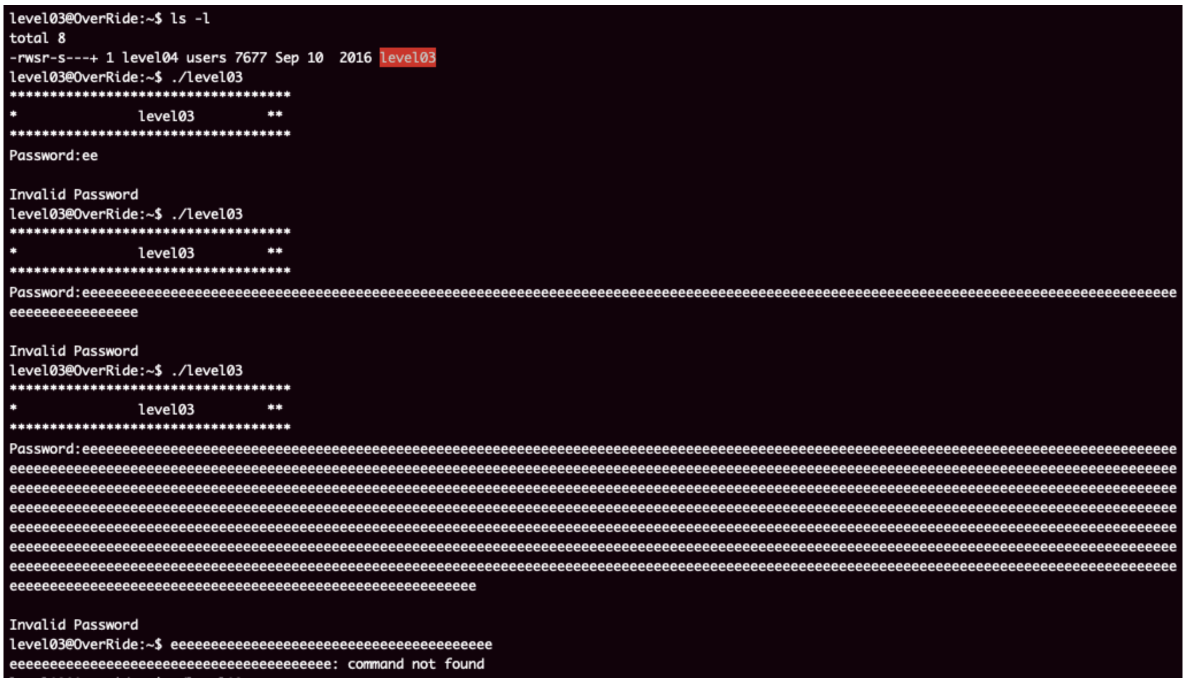
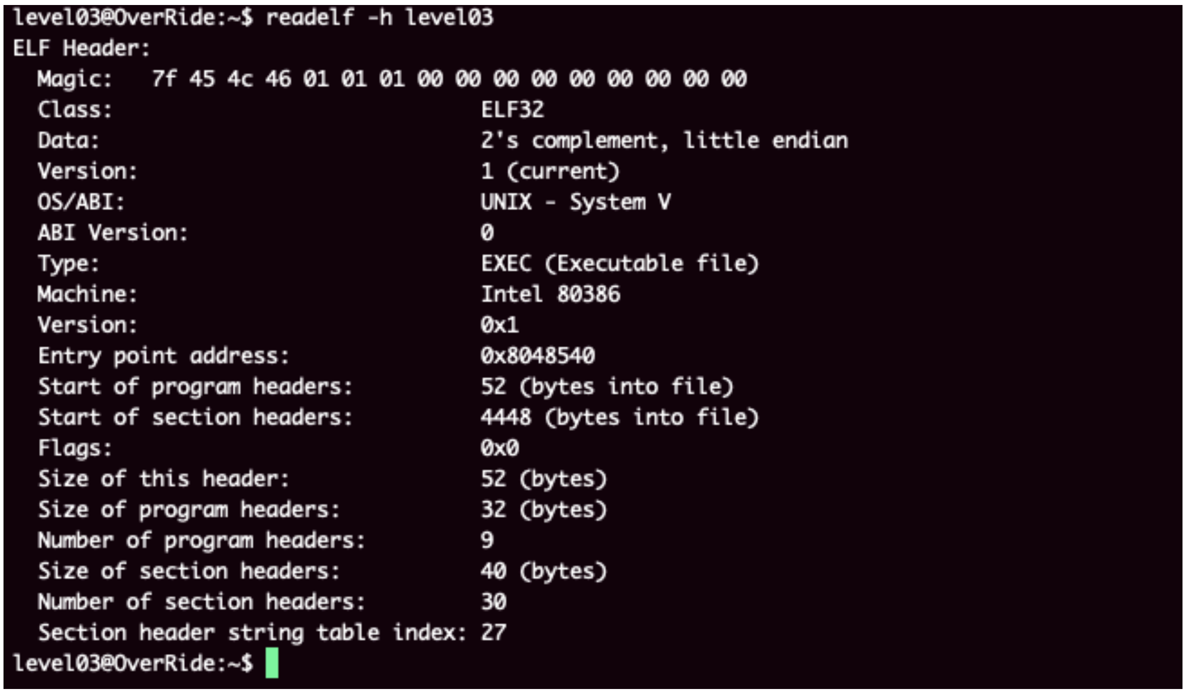
LEVEL03:



```
level03@OverRide:~$ ls -l
total 8
-rwsr-s---+ 1 level04 users 7677 Sep 10  2016 level03
level03@OverRide:~$ ./level03
***********************************
*               level03         **
***********************************
Password:ee

Invalid Password
level03@OverRide:~$ ./level03
***********************************
*               level03         **
***********************************
Password:eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeee

Invalid Password
level03@OverRide:~$ ./level03
***********************************
*               level03         **
***********************************
Password:eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee

Invalid Password
level03@OverRide:~$ eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee: command not found
```

We read once on stdin

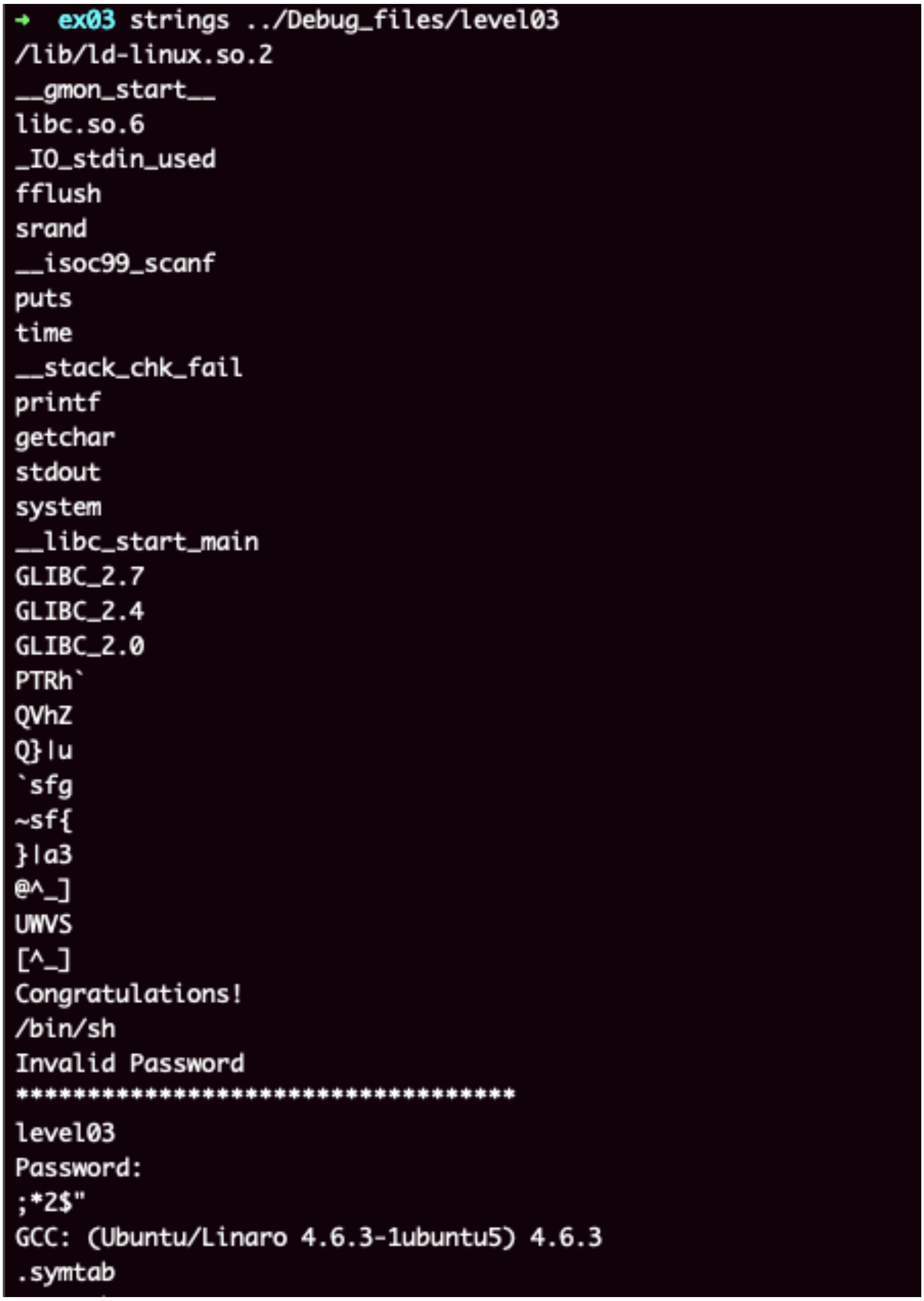
Readelf -h:



```
level03@OverRide:~$ readelf -h level03
ELF Header:
  Magic:   7f 45 4c 46 01 01 01 00 00 00 00 00 00 00 00 00
  Class:                             ELF32
  Data:                              2's complement, little endian
  Version:                           1 (current)
  OS/ABI:                            UNIX - System V
  ABI Version:                       0
  Type:                              EXEC (Executable file)
  Machine:                           Intel 80386
  Version:                           0x1
  Entry point address:               0x8048540
  Start of program headers:          52 (bytes into file)
  Start of section headers:          4448 (bytes into file)
  Flags:                             0x0
  Size of this header:               52 (bytes)
  Size of program headers:           32 (bytes)
  Number of program headers:         9
  Size of section headers:           40 (bytes)
  Number of section headers:         30
  Section header string table index: 27
level03@OverRide:~$
```

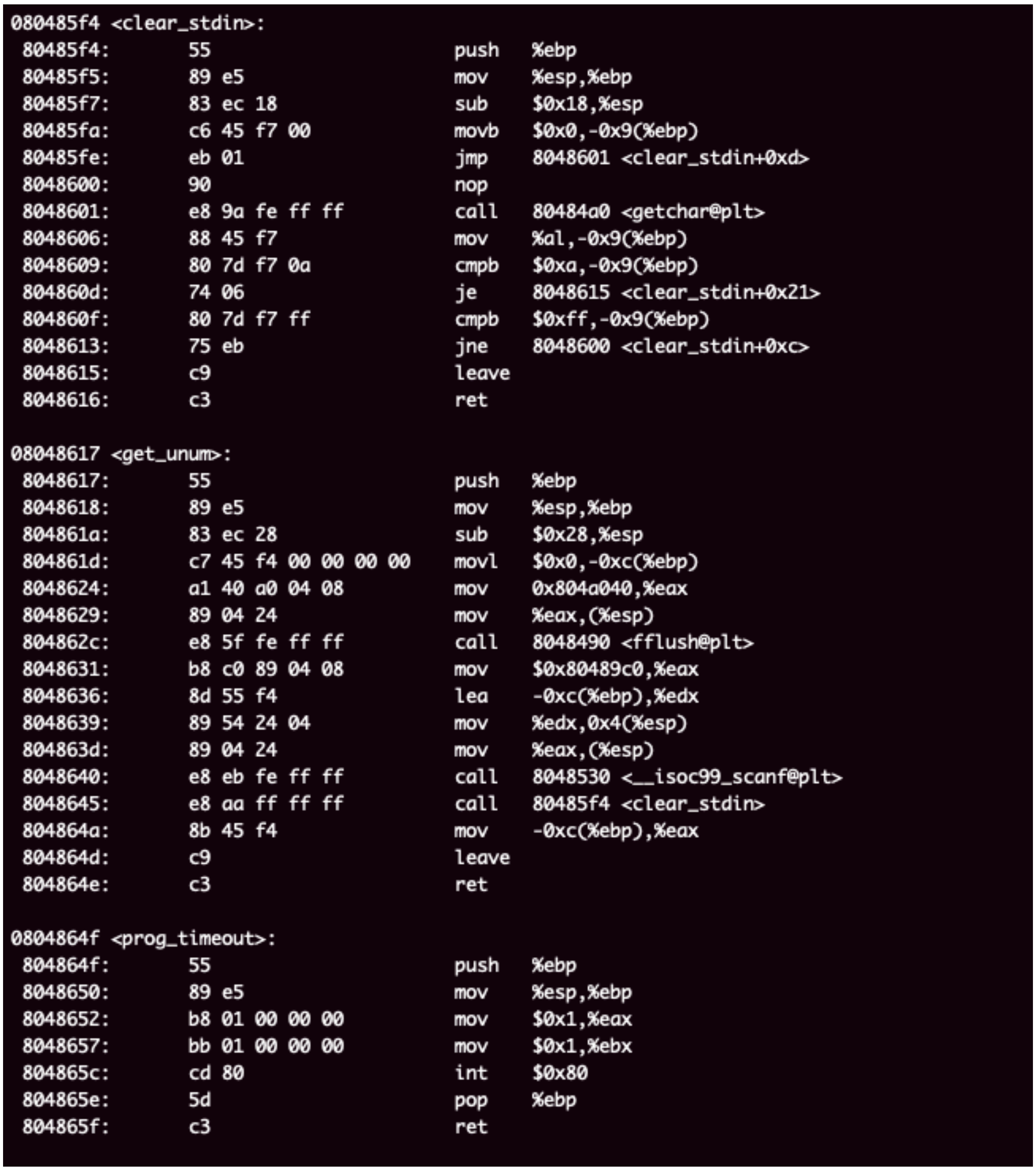
The binary is in 32bits

Strings:

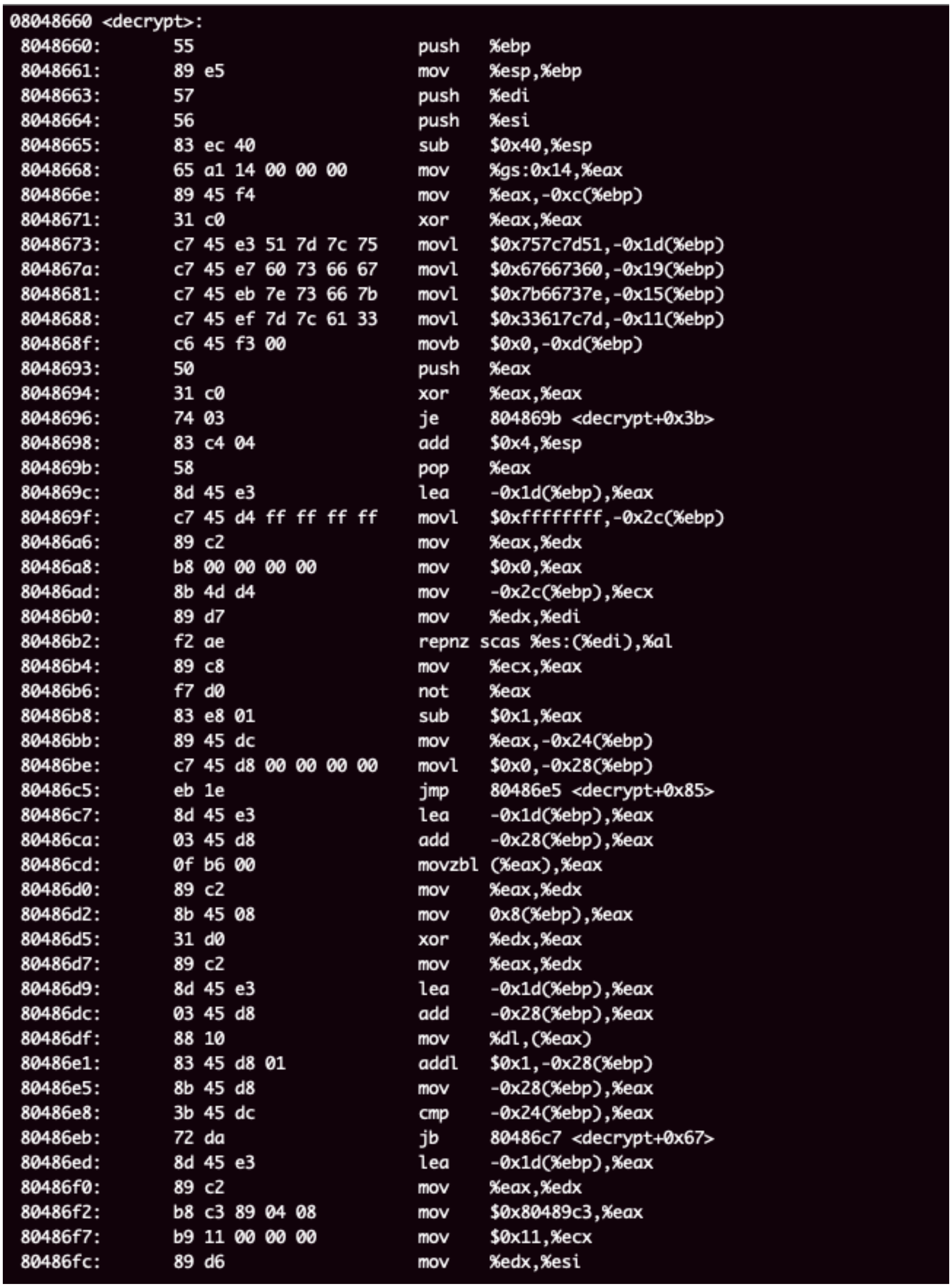


```
➜  ex03 strings ../Debug_files/level03
/lib/ld-linux.so.2
__gmon_start__
libc.so.6
_IO_stdin_used
fflush
srand
__isoc99_scanf
puts
time
__stack_chk_fail
printf
getchar
stdout
system
__libc_start_main
GLIBC_2.7
GLIBC_2.4
GLIBC_2.0
PTRh`
QVhZ
Q}|u
`sfg
~sf{
}|a3
@^_]
UWVS
[^_]
Congratulations!
/bin/sh
Invalid Password
***********************************
level03
Password:
;*2$"
GCC: (Ubuntu/Linaro 4.6.3-1ubuntu5) 4.6.3
.symtab
```

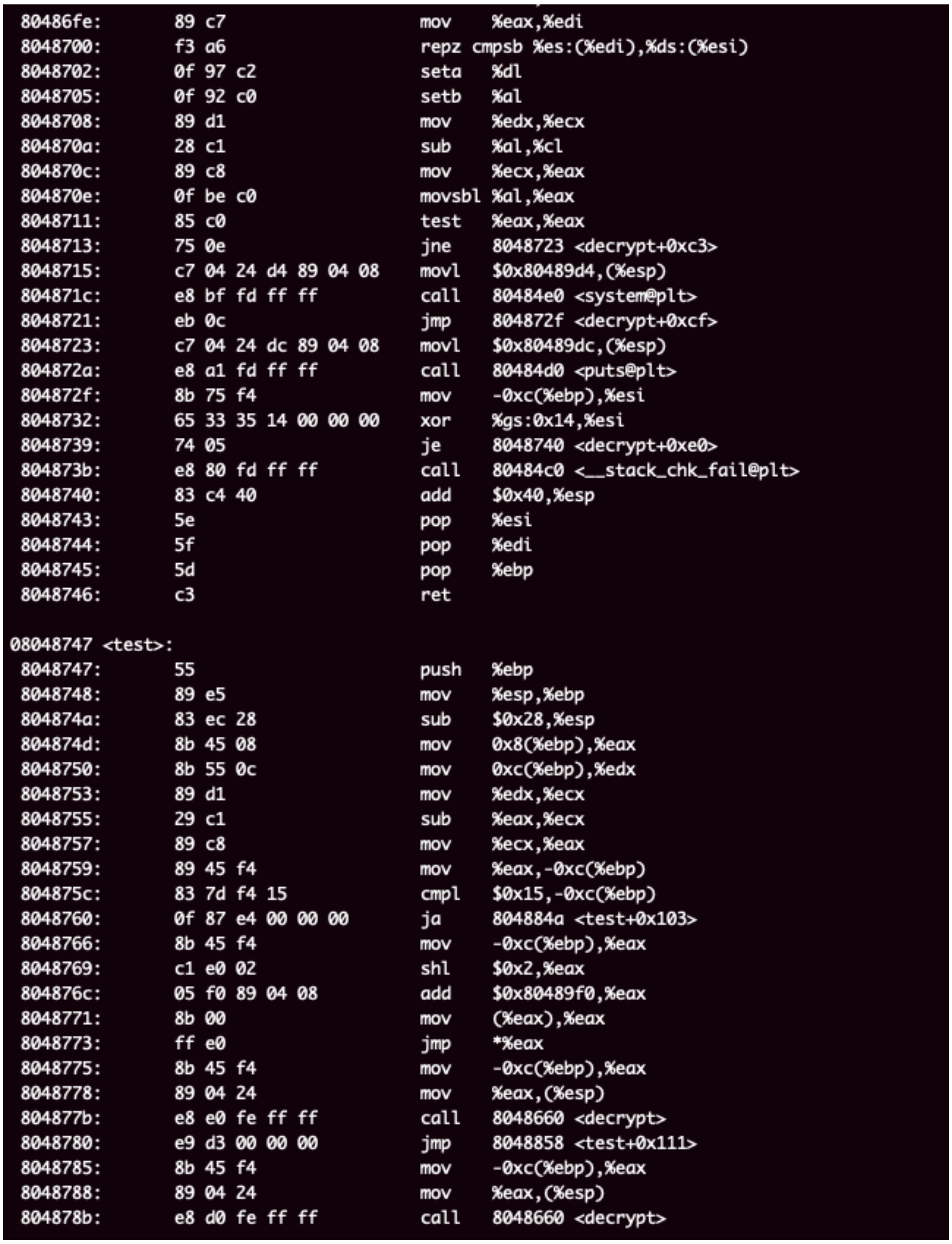
Objdump -d:



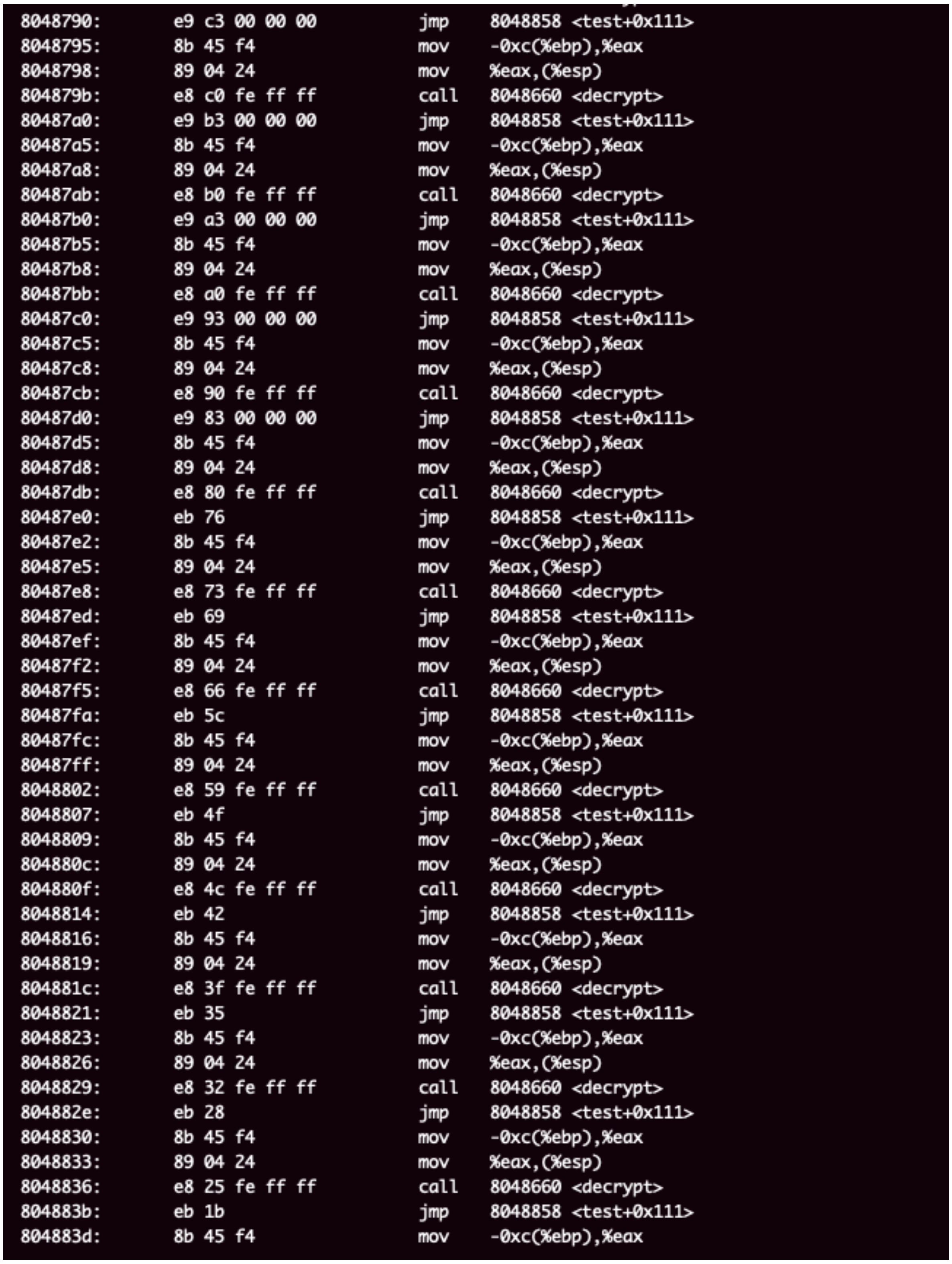
```
080485f4 <clear_stdin>:
 80485f4:       55                      push   %ebp
 80485f5:       89 e5                   mov    %esp,%ebp
 80485f7:       83 ec 18                sub    $0x18,%esp
 80485fa:       c6 45 f7 00             movb   $0x0,-0x9(%ebp)
 80485fe:       eb 01                   jmp    8048601 <clear_stdin+0xd>
 8048600:       90                      nop
 8048601:       e8 9a fe ff ff          call   80484a0 <getchar@plt>
 8048606:       88 45 f7                mov    %al,-0x9(%ebp)
 8048609:       80 7d f7 0a             cmpb   $0xa,-0x9(%ebp)
 804860d:       74 06                   je     8048615 <clear_stdin+0x21>
 804860f:       80 7d f7 ff             cmpb   $0xff,-0x9(%ebp)
 8048613:       75 eb                   jne    8048600 <clear_stdin+0xc>
 8048615:       c9                      leave
 8048616:       c3                      ret

08048617 <get_unum>:
 8048617:       55                      push   %ebp
 8048618:       89 e5                   mov    %esp,%ebp
 804861a:       83 ec 28                sub    $0x28,%esp
 804861d:       c7 45 f4 00 00 00 00    movl   $0x0,-0xc(%ebp)
 8048624:       a1 40 a0 04 08          mov    0x804a040,%eax
 8048629:       89 04 24                mov    %eax,(%esp)
 804862c:       e8 5f fe ff ff          call   8048490 <fflush@plt>
 8048631:       b8 c0 89 04 08          mov    $0x80489c0,%eax
 8048636:       8d 55 f4                lea    -0xc(%ebp),%edx
 8048639:       89 54 24 04             mov    %edx,0x4(%esp)
 804863d:       89 04 24                mov    %eax,(%esp)
 8048640:       e8 eb fe ff ff          call   8048530 <__isoc99_scanf@plt>
 8048645:       e8 aa ff ff ff          call   80485f4 <clear_stdin>
 804864a:       8b 45 f4                mov    -0xc(%ebp),%eax
 804864d:       c9                      leave
 804864e:       c3                      ret

0804864f <prog_timeout>:
 804864f:       55                      push   %ebp
 8048650:       89 e5                   mov    %esp,%ebp
 8048652:       b8 01 00 00 00          mov    $0x1,%eax
 8048657:       bb 01 00 00 00          mov    $0x1,%ebx
 804865c:       cd 80                   int    $0x80
 804865e:       5d                      pop    %ebp
 804865f:       c3                      ret
```



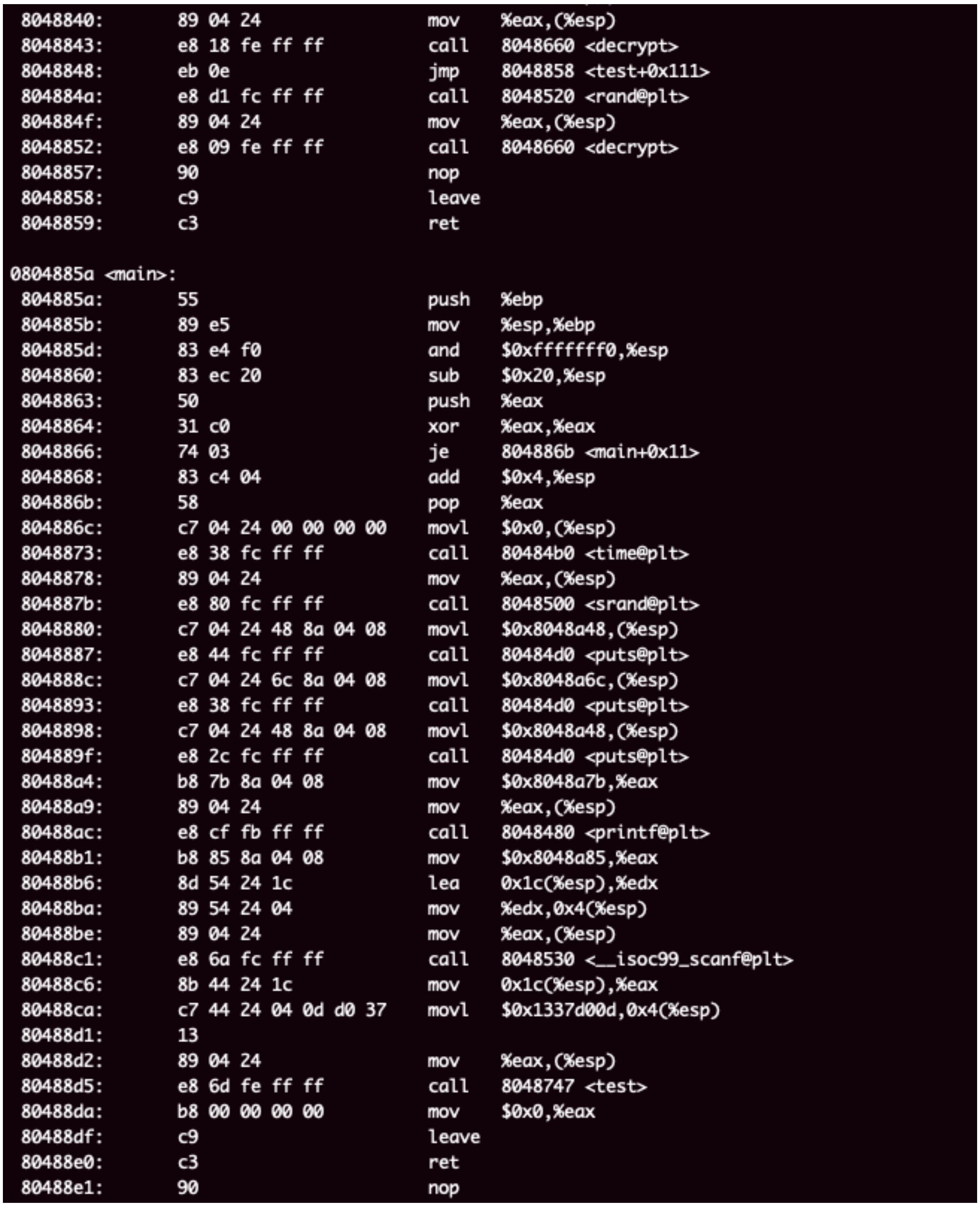
```
08048660 <decrypt>:
 8048660:       55                      push   %ebp
 8048661:       89 e5                   mov    %esp,%ebp
 8048663:       57                      push   %edi
 8048664:       56                      push   %esi
 8048665:       83 ec 40                sub    $0x40,%esp
 8048668:       65 a1 14 00 00 00       mov    %gs:0x14,%eax
 804866e:       89 45 f4                mov    %eax,-0xc(%ebp)
 8048671:       31 c0                   xor    %eax,%eax
 8048673:       c7 45 e3 51 7d 7c 75    movl   $0x757c7d51,-0x1d(%ebp)
 804867a:       c7 45 e7 60 73 66 67    movl   $0x67667360,-0x19(%ebp)
 8048681:       c7 45 eb 7e 73 66 7b    movl   $0x7b66737e,-0x15(%ebp)
 8048688:       c7 45 ef 7d 7c 61 33    movl   $0x33617c7d,-0x11(%ebp)
 804868f:       c6 45 f3 00             movb   $0x0,-0xd(%ebp)
 8048693:       50                      push   %eax
 8048694:       31 c0                   xor    %eax,%eax
 8048696:       74 03                   je     804869b <decrypt+0x3b>
 8048698:       83 c4 04                add    $0x4,%esp
 804869b:       58                      pop    %eax
 804869c:       8d 45 e3                lea    -0x1d(%ebp),%eax
 804869f:       c7 45 d4 ff ff ff ff    movl   $0xffffffff,-0x2c(%ebp)
 80486a6:       89 c2                   mov    %eax,%edx
 80486a8:       b8 00 00 00 00          mov    $0x0,%eax
 80486ad:       8b 4d d4                mov    -0x2c(%ebp),%ecx
 80486b0:       89 d7                   mov    %edx,%edi
 80486b2:       f2 ae                   repnz scas %es:(%edi),%al
 80486b4:       89 c8                   mov    %ecx,%eax
 80486b6:       f7 d0                   not    %eax
 80486b8:       83 e8 01                sub    $0x1,%eax
 80486bb:       89 45 dc                mov    %eax,-0x24(%ebp)
 80486be:       c7 45 d8 00 00 00 00    movl   $0x0,-0x28(%ebp)
 80486c5:       eb 1e                   jmp    80486e5 <decrypt+0x85>
 80486c7:       8d 45 e3                lea    -0x1d(%ebp),%eax
 80486ca:       03 45 d8                add    -0x28(%ebp),%eax
 80486cd:       0f b6 00                movzbl (%eax),%eax
 80486d0:       89 c2                   mov    %eax,%edx
 80486d2:       8b 45 08                mov    0x8(%ebp),%eax
 80486d5:       31 d0                   xor    %edx,%eax
 80486d7:       89 c2                   mov    %eax,%edx
 80486d9:       8d 45 e3                lea    -0x1d(%ebp),%eax
 80486dc:       03 45 d8                add    -0x28(%ebp),%eax
 80486df:       88 10                   mov    %dl,(%eax)
 80486e1:       83 45 d8 01             addl   $0x1,-0x28(%ebp)
 80486e5:       8b 45 d8                mov    -0x28(%ebp),%eax
 80486e8:       3b 45 dc                cmp    -0x24(%ebp),%eax
 80486eb:       72 da                   jb     80486c7 <decrypt+0x67>
 80486ed:       8d 45 e3                lea    -0x1d(%ebp),%eax
 80486f0:       89 c2                   mov    %eax,%edx
 80486f2:       b8 c3 89 04 08          mov    $0x80489c3,%eax
 80486f7:       b9 11 00 00 00          mov    $0x11,%ecx
 80486fc:       89 d6                   mov    %edx,%esi
```



```
 80486fe:       89 c7                   mov    %eax,%edi
 8048700:       f3 a6                   repz cmpsb %es:(%edi),%ds:(%esi)
 8048702:       0f 97 c2                seta   %dl
 8048705:       0f 92 c0                setb   %al
 8048708:       89 d1                   mov    %edx,%ecx
 804870a:       28 c1                   sub    %al,%cl
 804870c:       89 c8                   mov    %ecx,%eax
 804870e:       0f be c0                movsbl %al,%eax
 8048711:       85 c0                   test   %eax,%eax
 8048713:       75 0e                   jne    8048723 <decrypt+0xc3>
 8048715:       c7 04 24 d4 89 04 08    movl   $0x80489d4,(%esp)
 804871c:       e8 bf fd ff ff          call   80484e0 <system@plt>
 8048721:       eb 0c                   jmp    804872f <decrypt+0xcf>
 8048723:       c7 04 24 dc 89 04 08    movl   $0x80489dc,(%esp)
 804872a:       e8 a1 fd ff ff          call   80484d0 <puts@plt>
 804872f:       8b 75 f4                mov    -0xc(%ebp),%esi
 8048732:       65 33 35 14 00 00 00    xor    %gs:0x14,%esi
 8048739:       74 05                   je     8048740 <decrypt+0xe0>
 804873b:       e8 80 fd ff ff          call   80484c0 <__stack_chk_fail@plt>
 8048740:       83 c4 40                add    $0x40,%esp
 8048743:       5e                      pop    %esi
 8048744:       5f                      pop    %edi
 8048745:       5d                      pop    %ebp
 8048746:       c3                      ret

08048747 <test>:
 8048747:       55                      push   %ebp
 8048748:       89 e5                   mov    %esp,%ebp
 804874a:       83 ec 28                sub    $0x28,%esp
 804874d:       8b 45 08                mov    0x8(%ebp),%eax
 8048750:       8b 55 0c                mov    0xc(%ebp),%edx
 8048753:       89 d1                   mov    %edx,%ecx
 8048755:       29 c1                   sub    %eax,%ecx
 8048757:       89 c8                   mov    %ecx,%eax
 8048759:       89 45 f4                mov    %eax,-0xc(%ebp)
 804875c:       83 7d f4 15             cmpl   $0x15,-0xc(%ebp)
 8048760:       0f 87 e4 00 00 00       ja     804884a <test+0x103>
 8048766:       8b 45 f4                mov    -0xc(%ebp),%eax
 8048769:       c1 e0 02                shl    $0x2,%eax
 804876c:       05 f0 89 04 08          add    $0x80489f0,%eax
 8048771:       8b 00                   mov    (%eax),%eax
 8048773:       ff e0                   jmp    *%eax
 8048775:       8b 45 f4                mov    -0xc(%ebp),%eax
 8048778:       89 04 24                mov    %eax,(%esp)
 804877b:       e8 e0 fe ff ff          call   8048660 <decrypt>
 8048780:       e9 d3 00 00 00          jmp    8048858 <test+0x111>
 8048785:       8b 45 f4                mov    -0xc(%ebp),%eax
 8048788:       89 04 24                mov    %eax,(%esp)
 804878b:       e8 d0 fe ff ff          call   8048660 <decrypt>
```



```
8048790:	e9 c3 00 00 00       	jmp    8048858 <test+0x111>
8048795:	8b 45 f4             	mov    -0xc(%ebp),%eax
8048798:	89 04 24             	mov    %eax,(%esp)
804879b:	e8 c0 fe ff ff       	call   8048660 <decrypt>
80487a0:	e9 b3 00 00 00       	jmp    8048858 <test+0x111>
80487a5:	8b 45 f4             	mov    -0xc(%ebp),%eax
80487a8:	89 04 24             	mov    %eax,(%esp)
80487ab:	e8 b0 fe ff ff       	call   8048660 <decrypt>
80487b0:	e9 a3 00 00 00       	jmp    8048858 <test+0x111>
80487b5:	8b 45 f4             	mov    -0xc(%ebp),%eax
80487b8:	89 04 24             	mov    %eax,(%esp)
80487bb:	e8 a0 fe ff ff       	call   8048660 <decrypt>
80487c0:	e9 93 00 00 00       	jmp    8048858 <test+0x111>
80487c5:	8b 45 f4             	mov    -0xc(%ebp),%eax
80487c8:	89 04 24             	mov    %eax,(%esp)
80487cb:	e8 90 fe ff ff       	call   8048660 <decrypt>
80487d0:	e9 83 00 00 00       	jmp    8048858 <test+0x111>
80487d5:	8b 45 f4             	mov    -0xc(%ebp),%eax
80487d8:	89 04 24             	mov    %eax,(%esp)
80487db:	e8 80 fe ff ff       	call   8048660 <decrypt>
80487e0:	eb 76                	jmp    8048858 <test+0x111>
80487e2:	8b 45 f4             	mov    -0xc(%ebp),%eax
80487e5:	89 04 24             	mov    %eax,(%esp)
80487e8:	e8 73 fe ff ff       	call   8048660 <decrypt>
80487ed:	eb 69                	jmp    8048858 <test+0x111>
80487ef:	8b 45 f4             	mov    -0xc(%ebp),%eax
80487f2:	89 04 24             	mov    %eax,(%esp)
80487f5:	e8 66 fe ff ff       	call   8048660 <decrypt>
80487fa:	eb 5c                	jmp    8048858 <test+0x111>
80487fc:	8b 45 f4             	mov    -0xc(%ebp),%eax
80487ff:	89 04 24             	mov    %eax,(%esp)
8048802:	e8 59 fe ff ff       	call   8048660 <decrypt>
8048807:	eb 4f                	jmp    8048858 <test+0x111>
8048809:	8b 45 f4             	mov    -0xc(%ebp),%eax
804880c:	89 04 24             	mov    %eax,(%esp)
804880f:	e8 4c fe ff ff       	call   8048660 <decrypt>
8048814:	eb 42                	jmp    8048858 <test+0x111>
8048816:	8b 45 f4             	mov    -0xc(%ebp),%eax
8048819:	89 04 24             	mov    %eax,(%esp)
804881c:	e8 3f fe ff ff       	call   8048660 <decrypt>
8048821:	eb 35                	jmp    8048858 <test+0x111>
8048823:	8b 45 f4             	mov    -0xc(%ebp),%eax
8048826:	89 04 24             	mov    %eax,(%esp)
8048829:	e8 32 fe ff ff       	call   8048660 <decrypt>
804882e:	eb 28                	jmp    8048858 <test+0x111>
8048830:	8b 45 f4             	mov    -0xc(%ebp),%eax
8048833:	89 04 24             	mov    %eax,(%esp)
8048836:	e8 25 fe ff ff       	call   8048660 <decrypt>
804883b:	eb 1b                	jmp    8048858 <test+0x111>
804883d:	8b 45 f4             	mov    -0xc(%ebp),%eax
```



```
 8048840:	89 04 24             	mov    %eax,(%esp)
 8048843:	e8 18 fe ff ff       	call   8048660 <decrypt>
 8048848:	eb 0e                	jmp    8048858 <test+0x111>
 804884a:	e8 d1 fc ff ff       	call   8048520 <rand@plt>
 804884f:	89 04 24             	mov    %eax,(%esp)
 8048852:	e8 09 fe ff ff       	call   8048660 <decrypt>
 8048857:	90                   	nop
 8048858:	c9                   	leave
 8048859:	c3                   	ret

0804885a <main>:
 804885a:	55                   	push   %ebp
 804885b:	89 e5                	mov    %esp,%ebp
 804885d:	83 e4 f0             	and    $0xfffffff0,%esp
 8048860:	83 ec 20             	sub    $0x20,%esp
 8048863:	50                   	push   %eax
 8048864:	31 c0                	xor    %eax,%eax
 8048866:	74 03                	je     804886b <main+0x11>
 8048868:	83 c4 04             	add    $0x4,%esp
 804886b:	58                   	pop    %eax
 804886c:	c7 04 24 00 00 00 00 	movl   $0x0,(%esp)
 8048873:	e8 38 fc ff ff       	call   80484b0 <time@plt>
 8048878:	89 04 24             	mov    %eax,(%esp)
 804887b:	e8 80 fc ff ff       	call   8048500 <srand@plt>
 8048880:	c7 04 24 48 8a 04 08 	movl   $0x8048a48,(%esp)
 8048887:	e8 44 fc ff ff       	call   80484d0 <puts@plt>
 804888c:	c7 04 24 6c 8a 04 08 	movl   $0x8048a6c,(%esp)
 8048893:	e8 38 fc ff ff       	call   80484d0 <puts@plt>
 8048898:	c7 04 24 48 8a 04 08 	movl   $0x8048a48,(%esp)
 804889f:	e8 2c fc ff ff       	call   80484d0 <puts@plt>
 80488a4:	b8 7b 8a 04 08       	mov    $0x8048a7b,%eax
 80488a9:	89 04 24             	mov    %eax,(%esp)
 80488ac:	e8 cf fb ff ff       	call   8048480 <printf@plt>
 80488b1:	b8 85 8a 04 08       	mov    $0x8048a85,%eax
 80488b6:	8d 54 24 1c          	lea    0x1c(%esp),%edx
 80488ba:	89 54 24 04          	mov    %edx,0x4(%esp)
 80488be:	89 04 24             	mov    %eax,(%esp)
 80488c1:	e8 6a fc ff ff       	call   8048530 <__isoc99_scanf@plt>
 80488c6:	8b 44 24 1c          	mov    0x1c(%esp),%eax
 80488ca:	c7 44 24 04 0d d0 37 	movl   $0x1337d00d,0x4(%esp)
 80488d1:	13
 80488d2:	89 04 24             	mov    %eax,(%esp)
 80488d5:	e8 6d fe ff ff       	call   8048747 <test>
 80488da:	b8 00 00 00 00       	mov    $0x0,%eax
 80488df:	c9                   	leave
 80488e0:	c3                   	ret
 80488e1:	90                   	nop
```

Difference scanf and __isoc99_scanf:
	https://stackoverflow.com/questions/16376341/isoc99-scanf-and-scanf

Scanf parse the chain received on stdin to get a decimal format. (>> equivalent of atoi(stdin) that stores the result in the given parameter).
So we send a 'password' composed by digit characters.
We can see in the <test> function that <decrypt> is called

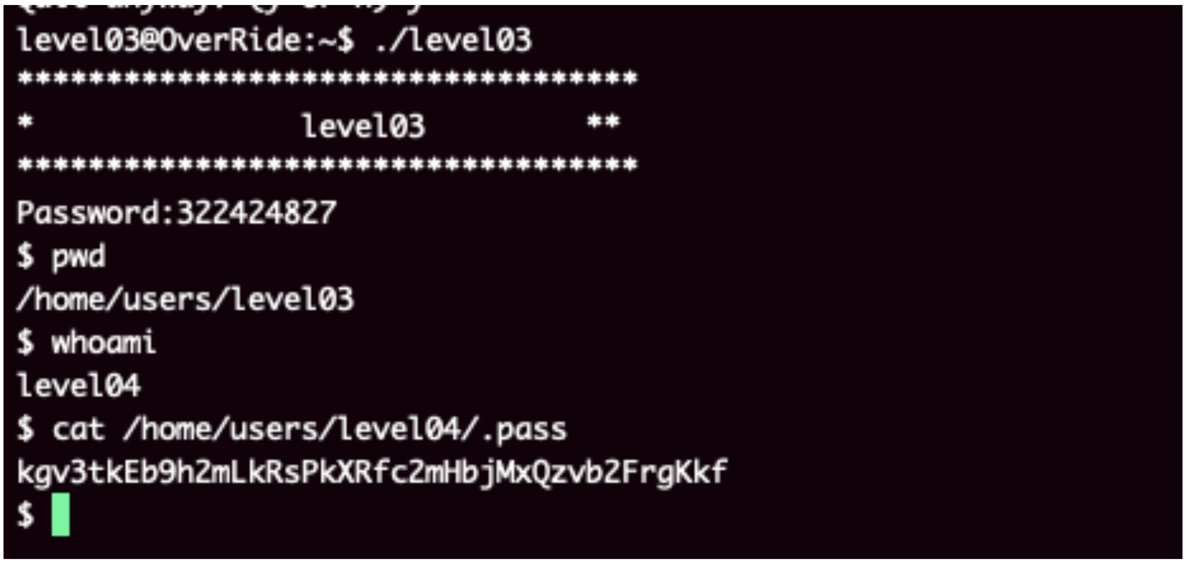
under certain conditions :

```
      difference = 0x1337d00d - password
      if (difference <= 0x15 && difference != 0, 0xA, 0xB, 0xC,
0xD, 0xE, 0xF)
            decrypt(difference)
      else
            decrypt(rand())
```

It lets us those possibilities of password:

- 0x1337D00C	0x1
- 0x1337D00B	0x2
- 0x1337D00A	0x3
- 0x1337D009	0x4
- 0x1337D008	0x5
- 0x1337D007	0x6
- 0x1337D006	0x7
- 0x1337D005	0x8
- 0x1337D004	0x9
- 0x1337CFFD	0x10
- 0x1337CFFC	0x11
- 0x1337CFFB	0x12
- 0x1337CFFA	0x13
- 0x1337CFF9	0x14
- 0x1337CFF8	0x15

We can either try all those pass, or think a better way:
we know that the difference will be used as an octet to xor the key (represented with 4 int or 16 octet) with it, and check if the result gives « Congratulations »
To know what is the octet that, xored with the first octet of the key, gives ‘0x43’, we do ‘0x43’ xor ‘0x51’ (first octet of the key) which gives 0x12.
So the difference must be 0x12. The password in hexa that gives 0x12 when substracted from 0x1337d00d if 0x1337CFFB.
Which gives in decimal : ***322424827.***



```
level03@OverRide:~$ ./level03
***********************************
*               level03         **
***********************************
Password:322424827
$ pwd
/home/users/level03
$ whoami
level04
$ cat /home/users/level04/.pass
kgv3tkEb9h2mLkRsPkXRfc2mHbjMxQzvb2FrgKkf
$ 
```

Flag: kgv3tkEb9h2mLkRsPkXRfc2mHbjMxQzvb2FrgKkf