

CS6570 Assignment 3a - Report

Rachit Tibrewal (CS16B022)
E Santhosh Kumar (CS16B107)

1 Handling Functions using PLT: Position Independent Code

Figures 1-5 are the screenshots for this section.

- There are 35 sections in total in the executable of driver code.
- Section [12] is the .plt section and it starts at location 0x08048480.
- Section [23] is the .got.plt section and it starts at location 0x0804a000.
- During the first call to function newnode(), the .got.plt entry for newnode contains 0x080484d6, which is the second instruction in newnode@plt.
- During the second call to function newnode(), the .got.plt entry for newnode contains 0xf7fd165f, which points to newnode's code.

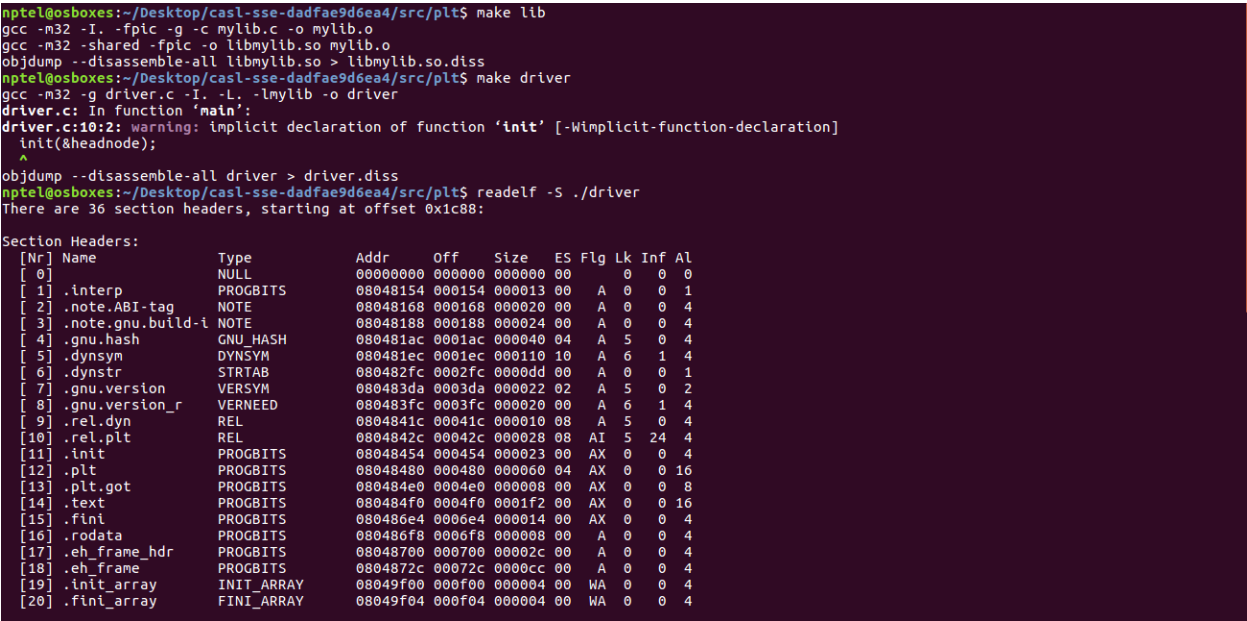


Figure 1:

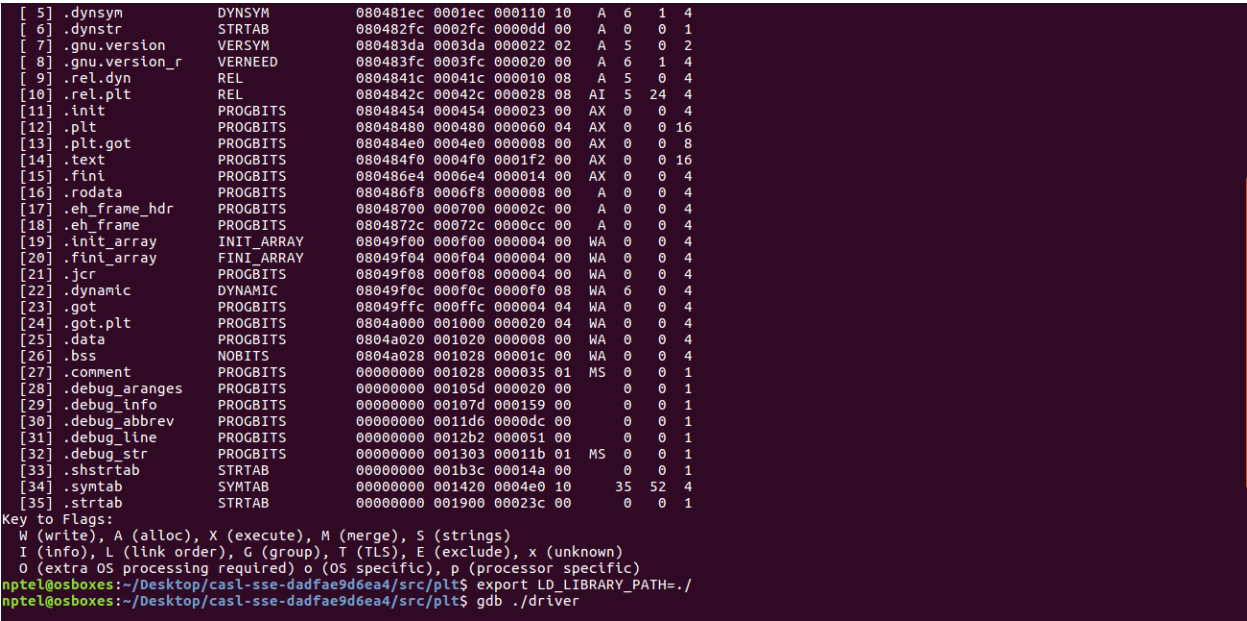


Figure 2:

```

0 (extra OS processing required) o (OS specific), p (processor specific)
nptel@osboxes:~/Desktop/casl-sse-dadfae9d6ea4/src/plt$ export LD_LIBRARY_PATH=./
nptel@osboxes:~/Desktop/casl-sse-dadfae9d6ea4/src/plt$ gdb ./driver
GNU gdb (Ubuntu 7.11.1-0ubuntu1-16.5) 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 "x86_64-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 ./driver...done.
(gdb) disassemble main
Dump of assembler code for function main:
0x080485eb <+0>:  lea    0x4(%esp),%ecx
0x080485ef <+4>:  and    $0xffffffff0,%esp
0x080485f2 <+7>:  pushl  -0x4(%ecx)
0x080485f5 <+10>: push    %ebp
0x080485f6 <+11>: mov     %esp,%ebp
0x080485f8 <+13>: push    %ecx
0x080485f9 <+14>: sub     $0x14,%esp
0x080485fc <+17>: sub     $0xc,%esp
0x080485ff <+20>: push    $0x804a028
0x08048604 <+25>: call    0x80484a0 <init@plt>
0x08048609 <+30>: add     $0x10,%esp
0x0804860c <+33>: call    0x80484d0 <newnode@plt>
0x08048611 <+38>: mov     %eax,-0x14(%ebp)
0x08048614 <+41>: mov     -0x14(%ebp),%eax
0x08048617 <+44>: movw    $0x41,%eax
0x0804861c <+49>: call    0x80484d0 <newnode@plt>
0x08048621 <+54>: mov     %eax,-0x10(%ebp)

```

Figure 3:

```

0x08048611 <+38>:  mov     %eax,-0x14(%ebp)
0x08048614 <+41>:  mov     -0x14(%ebp),%eax
0x08048617 <+44>:  movw    $0x41,%eax
0x0804861c <+49>:  call    0x80484d0 <newnode@plt>
0x08048621 <+54>:  mov     %eax,-0x10(%ebp)
0x08048624 <+57>:  mov     -0x10(%ebp),%eax
0x08048627 <+60>:  movw    $0x42,%eax
0x0804862c <+65>:  call    0x80484d0 <newnode@plt>
0x08048631 <+70>:  mov     %eax,-0xc(%ebp)
0x08048634 <+73>:  mov     -0xc(%ebp),%eax
0x08048637 <+76>:  movw    $0x43,%eax
0x0804863c <+81>:  call    0x8048490 <print@plt>
0x08048641 <+86>:  sub     $0xc,%esp
0x08048644 <+89>:  pushl   -0xc(%ebp)
0x08048647 <+92>:  call    0x80484c0 <delete@plt>
0x0804864c <+97>:  add     $0x10,%esp
0x0804864f <+100>: sub     $0xc,%esp
0x08048652 <+103>: pushl   -0x10(%ebp)
0x08048655 <+106>: call    0x80484c0 <delete@plt>
0x0804865a <+111>: add     $0x10,%esp
0x0804865d <+114>: sub     $0xc,%esp
0x08048660 <+117>: pushl   -0x14(%ebp)
0x08048663 <+120>: call    0x80484c0 <delete@plt>
---Type <return> to continue, or q <return> to quit---
0x08048668 <+125>: add     $0x10,%esp
0x0804866b <+128>: call    0x8048490 <print@plt>
0x08048670 <+133>: nop
0x08048671 <+134>: mov     -0x4(%ebp),%ecx
0x08048674 <+137>: leave
0x08048675 <+138>: lea     -0x4(%ecx),%esp
0x08048678 <+141>: ret
End of assembler dump.
(gdb) break 12
Breakpoint 1 at 0x804860c: file driver.c, line 12.
(gdb) break 15
Breakpoint 2 at 0x804861c: file driver.c, line 15.
(gdb) █

```

Figure 4:

```

(gdb) run
Starting program: /home/nptel/Desktop/casl-sse-dadfae9d6ea4/src/plt/driver

Breakpoint 1, main (argc=1, argv=0xffffd084) at driver.c:12
12      n1 = newnode();
(gdb) si
0x080484d0 in newnode@plt ()
(gdb) disassemble
Dump of assembler code for function newnode@plt:
=> 0x080484d0 <+0>:  jmp     *0x804a01c
0x080484d6 <+6>:  push    $0x20
0x080484db <+11>:  jmp     0x8048480
End of assembler dump.
(gdb) x/x 0x804a01c
0x804a01c:      0x080484d6
(gdb) si
0x080484d6 in newnode@plt ()
(gdb) si
0x080484db in newnode@plt ()
(gdb)
0x08048480 in ?? ()
(gdb)
0x08048486 in ?? ()
(gdb)
0xf7feef0 in ?? () from /lib/ld-linux.so.2
(gdb)
0xf7feef1 in ?? () from /lib/ld-linux.so.2
(gdb)
0xf7feef2 in ?? () from /lib/ld-linux.so.2
(gdb) finish
Run till exit from #0  0xf7feef2 in ?? () from /lib/ld-linux.so.2
0x08048611 in main (argc=1, argv=0xffffd084) at driver.c:12
12      n1 = newnode();
(gdb) x/x 0x804a01c
0x804a01c:      0xf7fd165f
(gdb) p newnode
$1 = {node_t *} 0xf7fd165f <newnode>

```

Figure 5:

2 Load Time Relocation

Figures 6-8 are the screenshots for this section.

- The libmylib.so.diss file contains 0x0 as address in places where you expect the mylib.int global vari-

able’s address to be.

- On inspecting the .rel.dyn section, we see that variable mylib.int occurs at 0x527 and 0x532.
- During runtime, we see that the loader has assigned address location 0xf7fd3014 to the mylib.int global variable.

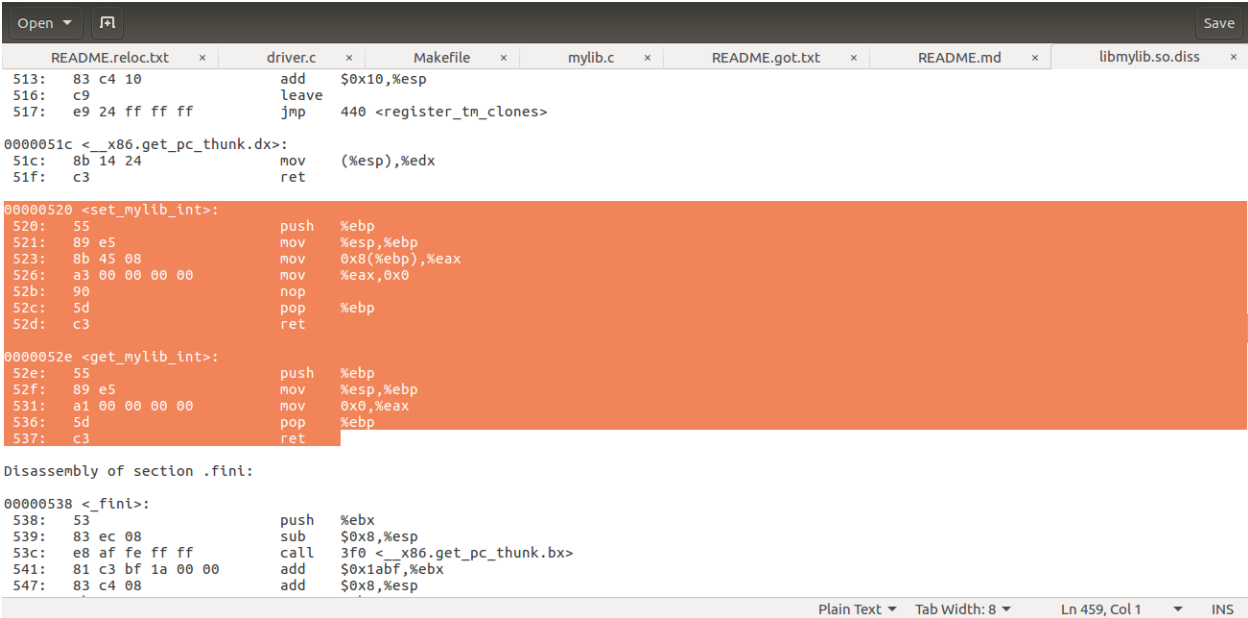


Figure 6:

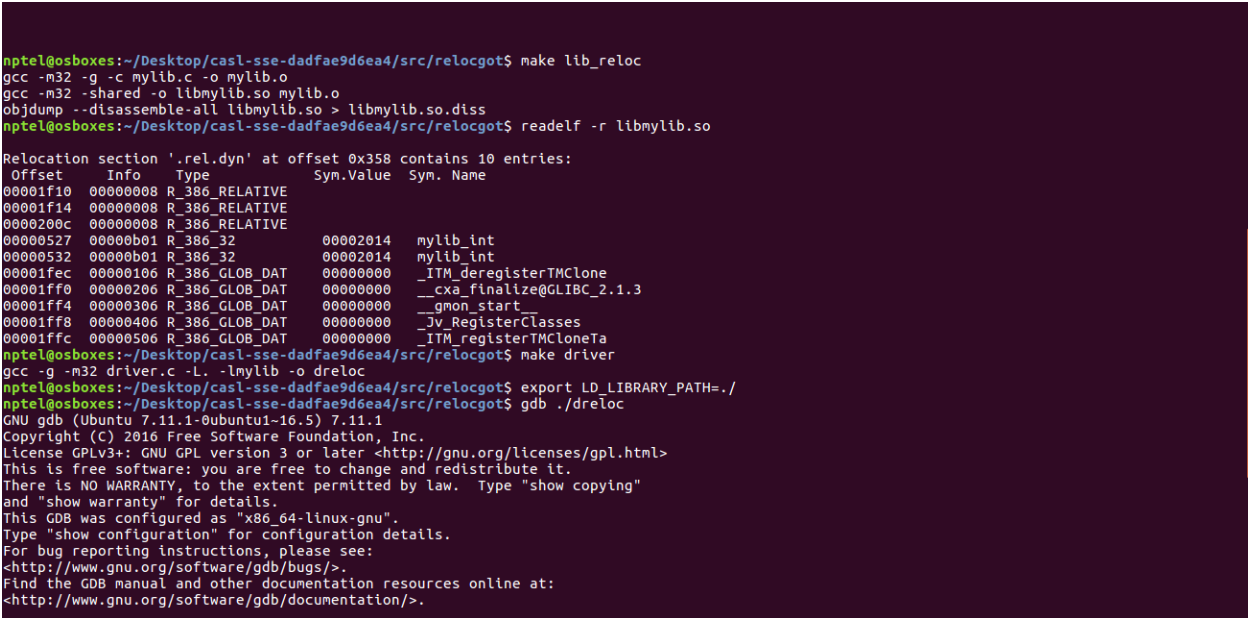


Figure 7:

3 Handling Data in PIC: Position Independent Code


Figures 9-13 are the screenshots for this section.

- The disassembled instructions for set_mylib.int in libmylib.pic.so.diss contains a call to the compiler specied function __x86.get_pc_thunk.ax . This function moves the address of the next instruction into eax.
 - The offset of GOT with respect to the current eip (here 0x528) is 0x1ad8.
 - The offset of mylib.int in the GOT is -0x14.
 - There are 38 section headers in the process address space. Entry [23] corresponds to the GOT table.
- During execution of the driver code,
- The address assigned to mylib.int by the loader is 0xf7fd3014.
 - GOT is present at location $0xf7fd1528 + 0x1ad8 = 0xf7fd3000$
 - The GOT entry for mylib.int is present at an offset of -0x14. This is $0xf7fd3000 - 0x14 = 0xf7fd2fec$. The value at 0xf7fd2fec is the pointer to mylib.int, 0xf7fd3014.
 - Global variable glob is present at location 0x804a024.

```
nptel@osboxes:~/Desktop/casl-sse-dadfae9d6ea4/src/relocgot$ make driver
gcc -g -m32 driver.c -L. -lmylib -o dreloc
nptel@osboxes:~/Desktop/casl-sse-dadfae9d6ea4/src/relocgot$ export LD_LIBRARY_PATH=./
nptel@osboxes:~/Desktop/casl-sse-dadfae9d6ea4/src/relocgot$ gdb ./dreloc
GNU gdb (Ubuntu 7.11.1-0ubuntu1-16.5) 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 "x86_64-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 ./dreloc...done.
(gdb) b main
Breakpoint 1 at 0x80485bc: file driver.c, line 11.
(gdb) r
Starting program: /home/nptel/Desktop/casl-sse-dadfae9d6ea4/src/relocgot/dreloc

Breakpoint 1, main () at driver.c:11
11      set_mylib_int(100);
(gdb) disass set_mylib_int
Dump of assembler code for function set_mylib_int:
0xf7fd1520 <+0>:  push    %ebp
0xf7fd1521 <+1>:  mov     %esp,%ebp
0xf7fd1523 <+3>:  mov     0x8(%ebp),%eax
0xf7fd1526 <+6>:  mov     %eax,0xf7fd3014
0xf7fd152b <+11>: nop
0xf7fd152c <+12>: pop     %ebp
0xf7fd152d <+13>: ret
End of assembler dump.
(gdb) █
```

Figure 8:

Open  Save

driver.c	x	Makefile	x	mylib.c	x	README.md	x	README.got.txt	x	libmylib_pic.so.diss	x
----------	---	----------	---	---------	---	-----------	---	----------------	---	----------------------	---

```
516: c9          leave
517: e9 24 ff ff  jmp    440 <register_tm_clones>

0000051c <__x86.get_pc_thunk.dx>:
51c: 8b 14 24     mov     (%esp),%edx
51f: c3          ret

00000520 <set_mylib_int>:
520: 55          push    %ebp
521: 89 e5       mov     %esp,%ebp
523: e8 2a 00 00  call   552 <__x86.get_pc_thunk.ax>
528: 05 d8 1a 00  add     $0x1ad8,%eax
52d: 8b 80 ec ff ff  mov     -0x14(%eax),%eax
533: 8b 55 08     mov     0x8(%ebp),%edx
536: 89 10       mov     %edx,(%eax)
538: 90          nop
539: 5d          pop     %ebp
53a: c3          ret

0000053b <get_mylib_int>:
53b: 55          push    %ebp
53c: 89 e5       mov     %esp,%ebp
53e: e8 0f 00 00  call   552 <__x86.get_pc_thunk.ax>
543: 05 bd 1a 00  add     $0x1abd,%eax
548: 8b 80 ec ff ff  mov     -0x14(%eax),%eax
54e: 8b 00       mov     (%eax),%eax
550: 5d          pop     %ebp
551: c3          ret

00000552 <__x86.get_pc_thunk.ax>:
552: 8b 04 24     mov     (%esp),%eax
555: c3          ret
```

Plain Text ▾ Tab Width: 8 ▾ Ln 457, Col 1 ▾ INS

Figure 9:

```
nptel@osboxes:~/Desktop/casl-sse-dadfae9d6ea4/src/relocgot$ make lib_pic
gcc -m32 -fpic -g -c mylib.c -o mylib.o
gcc -m32 -fpic -shared -o libmylib_pic.so mylib.o
objdump --disassemble-all libmylib_pic.so > libmylib_pic.so.diss
nptel@osboxes:~/Desktop/casl-sse-dadfae9d6ea4/src/relocgot$ make driver_pic
gcc -g -m32 driver.c -L. -lmylib_pic -o dpic
nptel@osboxes:~/Desktop/casl-sse-dadfae9d6ea4/src/relocgot$ readelf -S dpic
There are 36 section headers, starting at offset 0x1b10:

Section Headers:
[Nr] Name                Type              Addr      Off      Size    ES Flg Lk Inf Al
[ 0]                      NULL              00000000 000000 000000 00  0  0  0
[ 1] .interp                PROGBITS          00048154 000154 000013 00  A  0  0  1
[ 2] .note.ABI-tag          NOTE              00048168 000168 000020 00  A  0  0  4
[ 3] .note.gnu.build-id     NOTE              00048188 000188 000024 00  A  0  0  4
[ 4] .gnu.hash              GNU_HASH          000481ac 0001ac 00003c 04  A  5  0  4
[ 5] .dynsym                DYNSYM            000481e8 0001e8 0000f0 10  A  6  1  4
[ 6] .dynstr                STRTAB            000482d8 0002d8 0000e6 00  A  0  0  1
[ 7] .gnu.version            VERSYM            000483be 0003be 00001e 02  A  5  0  2
[ 8] .gnu.version_r          VERNEED           000483dc 0003dc 000020 00  A  6  1  4
[ 9] .rel.dyn               REL               000483fc 0003fc 000008 08  A  5  0  4
[10] .rel.plt               REL               00048404 000404 000020 08  AI 5 24  4
[11] .init                  PROGBITS          00048424 000424 000023 00  AX 0  0  4
[12] .plt                   PROGBITS          00048450 000450 000050 04  AX 0  0 16
[13] .plt.got               PROGBITS          000484a0 0004a0 000008 00  AX 0  0  8
[14] .text                  PROGBITS          000484b0 0004b0 0001a2 00  AX 0  0 16
[15] .fini                  PROGBITS          00048654 000654 000014 00  AX 0  0  4
[16] .rodata                PROGBITS          00048668 000668 000023 00  A  0  0  4
[17] .eh_frame_hdr          PROGBITS          0004868c 00068c 00002c 00  A  0  0  4
[18] .eh_frame              PROGBITS          000486b8 0006b8 0000cc 00  A  0  0  4
[19] .init_array            INIT_ARRAY         00049f00 000f00 000004 00  WA 0  0  4
[20] .fini_array            FINI_ARRAY         00049f04 000f04 000004 00  WA 0  0  4
[21] .jcr                   PROGBITS          00049f08 000f08 000004 00  WA 0  0  4
[22] .dynamic               DYNAMIC            00049f0c 000f0c 0000f0 08  WA 6  0  4
[23] .got                   PROGBITS          00049ffc 000ffc 000004 04  WA 0  0  4
[24] .got.plt               PROGBITS          0004a000 001000 00001c 04  WA 0  0  4
```

Figure 10:

```
[16] .rodata          PROGBITS          00048668 000668 000023 00    A 0 0 4
[17] .eh_frame_hdr    PROGBITS          0004868c 00068c 00002c 00    A 0 0 4
[18] .eh_frame        PROGBITS          000486b8 0006b8 0000cc 00    A 0 0 4
[19] .init_array      INIT_ARRAY        00049f00 000f00 000004 00   WA 0 0 4
[20] .fini_array      FINI_ARRAY        00049f04 000f04 000004 00   WA 0 0 4
[21] .jcr             PROGBITS          00049f08 000f08 000004 00   WA 0 0 4
[22] .dynamic         DYNAMIC          00049f0c 000f0c 0000f0 08   WA 6 0 4
[23] .got             PROGBITS          00049ffc 000ffc 000004 04   WA 0 0 4
[24] .got.plt         PROGBITS          0004a000 001000 00001c 04   WA 0 0 4
[25] .data           PROGBITS          0004a01c 00101c 00000c 00   WA 0 0 4
[26] .bss            NOBITS           0004a028 001028 000004 00   WA 0 0 1
[27] .comment        PROGBITS          00000000 001028 000035 01   MS 0 0 1
[28] .debug_aranges  PROGBITS          00000000 00105d 000020 00    0 0 1
[29] .debug_info     PROGBITS          00000000 00107d 0000a0 00    0 0 1
[30] .debug_abbrev   PROGBITS          00000000 00111d 000053 00    0 0 1
[31] .debug_line     PROGBITS          00000000 001170 00003d 00    0 0 1
[32] .debug_str      PROGBITS          00000000 0011ad 0000f6 01   MS 0 0 1
[33] .shstrtab       STRTAB           00000000 0019c5 00014a 00    0 0 1
[34] .symtab         SYMTAB           00000000 0012a4 0004d0 10   35 52 4
[35] .strtab         STRTAB           00000000 001774 000251 00    0 0 1
Key to Flags:
W (write), A (alloc), X (execute), M (merge), S (strings)
I (info), L (link order), G (group), T (TLS), E (exclude), x (unknown)
O (extra OS processing required) o (OS specific), p (processor specific)
nptel@osboxes:~/Desktop/casl-sse-dadfae9d6ea4/src/relocgot$ export LD_LIBRARY_PATH=./
nptel@osboxes:~/Desktop/casl-sse-dadfae9d6ea4/src/relocgot$ gdb ./dpic
GNU gdb (Ubuntu 7.11.1-0ubuntu1-16.5) 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 "x86_64-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:
```

Figure 11:

```
I (info), L (link order), G (group), T (TLS), E (exclude), x (unknown)
O (extra OS processing required) o (OS specific), p (processor specific)
nptel@osboxes:~/Desktop/casl-sse-dadfae9d6ea4/src/relocgot$ export LD_LIBRARY_PATH=./
nptel@osboxes:~/Desktop/casl-sse-dadfae9d6ea4/src/relocgot$ gdb ./dpic
GNU gdb (Ubuntu 7.11.1-0ubuntu1-16.5) 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 "x86_64-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 ./dpic...done.
(gdb) b set_mylib_int
Breakpoint 1 at 0x8048470
(gdb) r
Starting program: /home/nptel/Desktop/casl-sse-dadfae9d6ea4/src/relocgot/dpic

Breakpoint 1, set_mylib_int (x=100) at mylib.c:5
5      mylib_int = x;
(gdb) p/x &mylib_int
$1 = 0xf7fd3014
(gdb) disassemble
Dump of assembler code for function set_mylib_int:
   0xf7fd1520 <+0>: push    %ebp
   0xf7fd1521 <+1>: mov     %esp,%ebp
   0xf7fd1523 <+3>: call    0xf7fd1552 <_x86.get_pc_thunk.ax>
   0xf7fd1528 <+8>: add     $0x1ad8,%eax
=>  0xf7fd152d <+13>: mov     -0x14(%eax),%eax
   0xf7fd1533 <+19>: mov     0x8(%ebp),%edx
   0xf7fd1536 <+22>: mov     %edx,(%eax)
   0xf7fd1536 <+22>: mov     %edx,(%eax)
End of assembler dump.
(gdb) x/4x 0xf7fd2fec
0xf7fd2fec: 0xf7fd3014 0xf7e33a30 0x00000000 0x00000000
(gdb) p/x &glib
$2 = 0x804a024
(gdb) s
6      }
(gdb) s
main () at driver.c:12
12      printf("Value set in mylib is %ld\n", get_mylib_int());
(gdb) set *((int *) 0xf7fd2fec) = 0x804a024
(gdb) s
get_mylib_int () at mylib.c:10
10      return mylib_int;
(gdb) s
11      }
(gdb) s
Value set in mylib is 21845
main () at driver.c:13
13      }
(gdb) █
```

Figure 12:

```
Breakpoint 1, set_mylib_int (x=100) at mylib.c:5
5      mylib_int = x;
(gdb) p/x &mylib_int
$1 = 0xf7fd3014
(gdb) disassemble
Dump of assembler code for function set_mylib_int:
   0xf7fd1520 <+0>: push    %ebp
   0xf7fd1521 <+1>: mov     %esp,%ebp
   0xf7fd1523 <+3>: call    0xf7fd1552 <_x86.get_pc_thunk.ax>
   0xf7fd1528 <+8>: add     $0x1ad8,%eax
=>  0xf7fd152d <+13>: mov     -0x14(%eax),%eax
   0xf7fd1533 <+19>: mov     0x8(%ebp),%edx
   0xf7fd1536 <+22>: mov     %edx,(%eax)
   0xf7fd1538 <+24>: nop
   0xf7fd1539 <+25>: pop     %ebp
   0xf7fd153a <+26>: ret
End of assembler dump.
(gdb) x/4x 0xf7fd2fec
0xf7fd2fec: 0xf7fd3014 0xf7e33a30 0x00000000 0x00000000
(gdb) p/x &glib
$2 = 0x804a024
(gdb) s
6      }
(gdb) s
main () at driver.c:12
12      printf("Value set in mylib is %ld\n", get_mylib_int());
(gdb) set *((int *) 0xf7fd2fec) = 0x804a024
(gdb) s
get_mylib_int () at mylib.c:10
10      return mylib_int;
(gdb) s
11      }
(gdb) s
Value set in mylib is 21845
main () at driver.c:13
13      }
(gdb) █
```

Figure 13: