Link Example 1.A Static Linking - Analysis

Young W. Lim

2019-01-21 Mon

Outline

1. Based on
2. example code
3. relocation information
4. relocation in noting.o module
   - relocation entries : r.offset
   - relocation entries : r.type, r.symbol
   - relocation results
   - reference address
   - runtime address

Based on

1. https://stac47.github.io/c/relocation/elf/tutorial/2018/03/01/understanding-relocation-elf.html

Compling 32-bit program on 64-bit gcc

- gcc -v
- gcc -m32 t.c
- sudo apt-get install gcc-multilib
- sudo apt-get install g++-multilib
- gcc-multilib
- g++-multilib
- gcc -m32
- objdump -m i386

n_main.c

- // nothing.h ------------------------------------
  void doAlmostNothing();
- // n_main.c ------------------------------------
  #include "nothing.h"

  int main(int argc, const char *argv[])
  {
    doAlmostNothing();
    return 0;
  }

https://stac47.github.io/c/relocation/elf/tutorial/2018/03/01/understanding-reloca

doNothingStatic, doNothing, doAlmostNothing

```
// nothing.c -------------------------------------
static void doNothingStatic() {
}

void doNothing() {
}

void doAlmostNothing() {
  doNothingStatic();
  doNothing();
}
```

https://stac47.github.io/c/relocation/elf/tutorial/2018/03/01/understanding-reloca

compiling commands

- ```
  gcc -g -m32 -Wall -c nothing.c -o nothing.o
  ar rcs libnothing.a nothing.o

  gcc -g -m32 -Wall -c nmain.c
  gcc -g -m32  nmain.o -L. -lnothing   -o nmain.out
  ```

https://stac47.github.io/c/relocation/elf/tutorial/2018/03/01/understanding-reloca
  ```

analyzing commands

- ```
  objdump -d  nothing.o ;   objdump -d  nmain.out
  objdump -dr nothing.o ;   objdump -dr nmain.out
  objdump -D  nothing.o ;   objdump -D  nmain.out
  ```
- ```
  readelf -r nothing.o
  readelf -r nmain.o
  readelf -r nmain.out
  ```
- ```
  readelf -S nmain.out   (--sections)
  readelf -s nmain.out   (--symbols)
  ```

https://stac47.github.io/c/relocation/elf/tutorial/2018/03/01/understanding-reloca
  ```

relocation information in nothing.o (1)

objdump -dr nothing.o

```
 3:   e8 fc ff ff ff          call   4 <doNothingStatic+0x4>
                      4: R_386_PC32   __x86.get_pc_thunk.ax
 8:   05 01 00 00 00          add    $0x1,%eax
                      9: R_386_GOTPC  _GLOBAL_OFFSET_TABLE_
13:   e8 fc ff ff ff          call   14 <doNothing+0x4>
                      14: R_386_PC32  __x86.get_pc_thunk.ax
18:   05 01 00 00 00          add    $0x1,%eax
                      19: R_386_GOTPC _GLOBAL_OFFSET_TABLE_
```

readelf -r nothing.o

```
 Offset     Info    Type            Sym.Value  Sym. Name
00000004  00001002 R_386_PC32        00000000   __x86.get_pc_thunk.ax
00000009  0000110a R_386_GOTPC       00000000   _GLOBAL_OFFSET_TABLE_
00000014  00001002 R_386_PC32        00000000   __x86.get_pc_thunk.ax
00000019  0000110a R_386_GOTPC       00000000   _GLOBAL_OFFSET_TABLE_
```

relocation information in nothing.o (2)

objdump -dr nothing.o

```
23:   e8 fc ff ff ff          call   24 <doAlmostNothing+0x4>
                      24: R_386_PC32  __x86.get_pc_thunk.ax
28:   05 01 00 00 00          add    $0x1,%eax
                      29: R_386_GOTPC _GLOBAL_OFFSET_TABLE_
32:   e8 fc ff ff ff          call   33 <doAlmostNothing+0x13>
                      33: R_386_PC32  doNothing
```

readelf -r nothing.o

```
 Offset     Info    Type            Sym.Value  Sym. Name
00000024  00001002 R_386_PC32        00000000   __x86.get_pc_thunk.ax
00000029  0000110a R_386_GOTPC       00000000   _GLOBAL_OFFSET_TABLE_
00000033  00001202 R_386_PC32        00000010   doNothing
```

relocation information in nmain.o

objdump -dr nmain.o

```
 f:   e8 fc ff ff ff          call   10 <main+0x10>
                      10: R_386_PC32  __x86.get_pc_thunk.ax
14:   05 01 00 00 00          add    $0x1,%eax
                      15: R_386_GOTPC _GLOBAL_OFFSET_TABLE_
1b:   e8 fc ff ff ff          call   1c <main+0x1c>
                      1c: R_386_PLT32 doAlmostNothing
```

readelf -r nmain.o

```
 Offset     Info    Type            Sym.Value  Sym. Name
00000010  00001002 R_386_PC32        00000000   __x86.get_pc_thunk.ax
00000015  0000110a R_386_GOTPC       00000000   _GLOBAL_OFFSET_TABLE_
0000001c  00001204 R_386_PLT32       00000000   doAlmostNothing
```

r.offset (1) in doNothingStatic

doNothingStatic calls __x86.get_pc_thunk.ax

- 00000000 <doNothingStatic>:
  ...
  3: e8 [fc ff ff ff] call 4 <doNothingStatic+0x4>
- symbol reference location (offset) 0x4 = <0x0 + 0x4>
- initial value (bias) -0x4 = 0xfffffffc

r.offset (2) in doNothing

doNothing calls __x86.get_pc_thunk.ax

- 00000010 <doNothing>:
  ...
  13: e8 [fc ff ff ff] call 14 <doNothing+0x4>
- symbol reference location (offset) 0x14 = <0x10 + 0x4>
- initial value (bias) -0x4 = 0xfffffffc

r.offset (3) in doAlmostNothing

doAlmostNothing calls __x86.get_pc_thunk.ax

- 00000020 <doAlmostNothing>:
  ...
  23: e8 [fc ff ff ff] call 24 <doAlmostNothing+0x4>
- symbol reference location (offset) 0x24 = <0x20 + 0x4>
- initial value (bias) -0x4 = 0xfffffffc

r.offset (4) in doAlmostNothing

doAlmostNothing calls doNothingStatic

- 00000020 <doAlmostNothing>:
  ...
  2d: e8 [ce ff ff ff] call 0 <doNothingStatic>
- symbol reference location (offset) 0x0 = <0x0 + 0x0>
- initial value (bias) -0x32 = 0xffffffce
  otherwise
  - 0x2e = <0x20 + 0xe>
  - -0x4 = 0xfffffffc (0x32 = 0x2e + 0x4)

r.offset (5) in doAlmostNothing

doAlmostNothing calls doNothing

- 00000020 <doAlmostNothing>:
  ...
  32: e8 [fc ff ff ff] call 33 <doAlmostNothing+0x13>
- symbol reference location (offset) 0x33 = <0x20+0x13>
- initial value (bias) -0x4 = 0xfffffffc

relocation types

- R_386_PC32 determine the destinance
  from this memory location to the "symbol",
  then add it to the value currently at this dword;
  deposit the result back into the dword
- R_386_GOTPC determine the distance
  from here to the GLOBAL_OFFSET_TABLE (&GOT[0]) and
  deposit the difference as a dword into this location
  (does not involve a symbol!)
  used in function prolog to calculate &GOT[0]

https://stac47.github.io/c/relocation/elf/tutorial/2018/03/01/understanding-reloca

relocation section .rel.text, .rel.eh_frame

```
young@USys1:~$ readelf -r nothing.o

Relocation section '.rel.text' at offset 0x288 contains 7 entries:
 Offset     Info    Type            Sym.Value  Sym. Name
00000004  00000b02 R_386_PC32        00000000   __x86.get_pc_thunk.ax
00000009  00000c0a R_386_GOTPC       00000000   _GLOBAL_OFFSET_TABLE_
00000014  00000b02 R_386_PC32        00000000   __x86.get_pc_thunk.ax
00000019  00000c0a R_386_GOTPC       00000000   _GLOBAL_OFFSET_TABLE_
00000024  00000b02 R_386_PC32        00000000   __x86.get_pc_thunk.ax
00000029  00000c0a R_386_GOTPC       00000000   _GLOBAL_OFFSET_TABLE_
00000033  00000d02 R_386_PC32        00000010   doNothing

Relocation section '.rel.eh_frame' at offset 0x2c0 contains 4 entries:
 Offset     Info    Type            Sym.Value  Sym. Name
00000020  00000202 R_386_PC32        00000000   .text
00000040  00000202 R_386_PC32        00000000   .text
00000060  00000202 R_386_PC32        00000000   .text
00000080  00000602 R_386_PC32        00000000   .text.__x86.get_pc_thu
```

r.type, r.symbol (1) in doNothingStatic

doNothingStatic calls __x86.get_pc_thunk.ax

```
00000000 <__x86.get_pc_thunk.ax>:
...
00000000 <doNothingStatic>:
...
   3:   e8 fc ff ff ff          call   4 <doNothingStatic+0x4>
                        4: R_386_PC32   __x86.get_pc_thunk.ax
```

relocation entry r

offset	0x4
bias	-0x4
symbol	__x86.get_pc_thunk.ax
type	R_386_PC32

r.type, r.symbol (2) in doNothing

doNothing calls __x86.get_pc_thunk.ax

```
00000000 <__x86.get_pc_thunk.ax>:
...
00000010 <doNothing>:
...
  13:   e8 fc ff ff ff          call   14 <doNothing+0x4>
                        14: R_386_PC32  __x86.get_pc_thunk.ax
```

relocation entry r

offset	0x14
bias	-0x4
symbol	__x86.get_pc_thunk.ax
type	R_386_PC32

r.type, r.symbol (3) in doAlmostNothing

doAlmostNothing calls __x86.get_pc_thunk.ax

```
00000020 <doAlmostNothing>:
  23:   e8 fc ff ff ff          call   24 <doAlmostNothing+0x4>
                        24: R_386_PC32  __x86.get_pc_thunk.ax
```

relocation entry r

offset	0x24
bias	-0x4
symbol	__x86.get_pc_thunk.ax
type	R_386_PC32

r.type, r.symbol (4) in doAlmostNothing

doAlmostNothing calls doNothingStatic

```
00000020 <doAlmostNothing>:
...
  2d:   e8 ce ff ff ff          call   0 <doNothingStatic>
```

relocation entry r

offset	0x0
bias	-0x32
symbol	doNothingStatic
type	

r.type, r.symbol (5) in doAlmostNothing

doAlmostNothing calls doNothing

```
00000020 <doAlmostNothing>:
...
  32:   e8 fc ff ff ff          call   33 <doAlmostNothing+0x13>
                        33: R_386_PC32  doNothing
```

relocation entry r

offset	0x33
bias	-0x4
symbol	doNothing
type	R_386_PC32

relocation results (1) in doNothingStatic

nothing.o

```
00000000 <__x86.get_pc_thunk.ax>:
...
00000000 <doNothingStatic>:
...
   3:   e8 fc ff ff ff          call   4 <doNothingStatic+0x4>
                        4: R_386_PC32   __x86.get_pc_thunk.ax
--- bias = -0x04, offset = 0x04 = <0x0 + 0x4> --------------------
    0x519 + 0x04 + 0x04 = 0x521 reference location
```

nmain.out

```
00000519 <__x86.get_pc_thunk.ax>:
...
0000051d <doNothingStatic>:
...
 520:   e8 f4 ff ff ff          call   519 <__x86.get_pc_thunk.ax>
--- 0x521 + 0x4 - 0xc = 0x519 (0xfffffff4 = -0xc) -----------------
```

relocation results (2) in doNothing

nothing.o

```
00000000 <__x86.get_pc_thunk.ax>:
...
00000010 <doNothing>:
...
  13:   e8 fc ff ff ff          call   14 <doNothing+0x4>
                        14: R_386_PC32  __x86.get_pc_thunk.ax
--- bias = -0x04, offset = 0x14 = <0x10 + 0x4> --------------------
    0x519 + 0x4 + 0x14 = 0x531 reference location
```

nmain.out

```
00000519 <__x86.get_pc_thunk.ax>:
...
0000052d <doNothing>:
...
 530:   e8 e4 ff ff ff          call   519 <__x86.get_pc_thunk.ax>
--- 0x531 + 0x4 - 0x1c = 0x519 (0xffffffe4 = - 0x1c) -------------
```

relocation results (3) in doAlmostNothing

nothing.o

```
00000020 <doAlmostNothing>:
  23:   e8 fc ff ff ff          call   24 <doAlmostNothing+0x4>
                        24: R_386_PC32  __x86.get_pc_thunk.ax
--- bias = -0x04, offset = 0x24 = <0x20 + 0x4> --------------------
    0x519 + 0x4 + 0x24 = 0x541 reference location
```

nmain.out

```
0000053d <doAlmostNothing>:
 540:   e8 d4 ff ff ff          call   519 <__x86.get_pc_thunk.ax>
--- 0x541 + 0x4 - 0x2c = 0x519 (-0x2c = 0xffffffd4) ---------------
```

relocation results (4) in doAlmostNothing

nothing.o

```
00000020 <doAlmostNothing>:
  23:   e8 fc ff ff ff          call   24 <doAlmostNothing+0x4>
                        24: R_386_PC32  __x86.get_pc_thunk.ax
  2d:   e8 ce ff ff ff          call   0 <doNothingStatic>
--- bias = -0x32, symbol offset = 0x0 = <0x0 + 0x0> ----------------
    0x519 + 0x32 + 0x0 = 0x54b reference location
```

nmain.out

```
0000053d <doAlmostNothing>:
 540:   e8 d4 ff ff ff          call   519 <__x86.get_pc_thunk.ax>
 54a:   e8 ce ff ff ff          call   51d <doNothingStatic>
--- 0x54b + 0x4 - 0x32 = 0x51d (-0x32 = 0xffffffce) ----------------
```

relocation results (5) in doAlmostNothing

nothing.o

```
00000020 <doAlmostNothing>:
  23:   e8 fc ff ff ff          call   24 <doAlmostNothing+0x4>
                        24: R_386_PC32  __x86.get_pc_thunk.ax
  2d:   e8 ce ff ff ff          call   0 <doNothingStatic>
  32:   e8 fc ff ff ff          call   33 <doAlmostNothing+0x13>
                        33: R_386_PC32  doNothing
--- bias = -0x4, offset = 0x33 = <0x20 + 0x13> --------------------
    0x519 + 0x4 + 0x33 = 0x550 reference location
```

nmain.out

```
0000053d <doAlmostNothing>:
 540:   e8 d4 ff ff ff          call   519 <__x86.get_pc_thunk.ax>
 54a:   e8 ce ff ff ff          call   51d <doNothingStatic>
 54f:   e8 d9 ff ff ff          call   52d <doNothing>
--- 0x550 + 0x4 - 0x27 = 0x52d (-0x27 = 0xffffffd9) ----------------
```

function addresses

.text section address

```
young@USys2:~/nmain$ readelf -S nmain.out

  [14] .text             PROGBITS        000003b0 0003b0 000212 00  AX  0   0 16
```

function addresses

```
young@USys2:~/nmain$ readelf -s nmain.out

    64: 000004ed    44 FUNC    GLOBAL DEFAULT   14 main

    65: 00000519     0 FUNC    GLOBAL HIDDEN    14 __x86.get_pc_thunk.ax

    36: 0000051d    16 FUNC    LOCAL  DEFAULT   14 doNothingStatic

    67: 0000052d    16 FUNC    GLOBAL DEFAULT   14 doNothing

    46: 0000053d    26 FUNC    GLOBAL DEFAULT   14 doAlmostNothing
```

function symbol relocations

function symbol values in nothing.o and nmain.out

- 00000000 <__x86.get_pc_thunk.ax> in .text.__x86.get_pc_thunk.ax of nothing.o
  00000519 <__x86.get_pc_thunk.ax> in .text of nmain.out
- 00000000 <doNothingStatic>: in .text of nothing.o
  0000051d <doNothingStatic>: in .text of nmain.out
- 00000010 <doNothing>: in .text of nothing.o
  0000052d <doNothing>: in .text of nmain.out
- 00000020 <doAlmostNothing>: in .text of nothing.o
  0000053d <doAlmostNothing>: in .text of nmain.out

https://stac47.github.io/c/relocation/elf/tutorial/2018/03/01/understanding-reloca

symbol reference (1) in doNothingStatic

__x86.get_pc_thunk.ax reference

- doNothingStatic calls __x86.get_pc_thunk.ax
- function symbol values in nmain.out

```
doNothingStatic       : 0x51d <---
__x86.get_pc_thunk.ax : 0x519
```

- in nothing.o

```
3: e8 fc ff ff ff          call   4 <doNothingStatic+0x4>
   0x519 + 0x4 + 0x4 = 0x521 = refaddr ... 0xfffffffc, 4
   0x51d + 0x4 = 0x521 = refaddr       ... <doNothingStatic+0x4>
```

- in nmain.out

```
520: e8 f4 ff ff ff          call   519 <__x86.get_pc_thunk.ax>
     0x519 - 0x521 - 4 = -0xc = *refaddr
```

symbol reference (2) in doNothing

__x86.get_pc_thunk.ax reference

- doNothing calls __x86.get_pc_thunk.ax
- function symbol values in nmain.out

```
doNothing              : 0x52d <---
__x86.get_pc_thunk.ax  : 0x519
```

- in nothing.o

```
13: e8 fc ff ff ff          call   14 <doNothing+0x4>
    0x519 + 0x4 + 0x14 = 0x531 = refaddr ... 0xfffffffc, 14
    0x52d + 0x4 = 0x531 = refaddr        ... <doNothing+0x4>
```

- in nmain.out

```
530: e8 e4 ff ff ff          call   519 <__x86.get_pc_thunk.ax>
     0x519 - 0x531 - 4 = -0x1c = *refaddr
```

symbol references (3) in doAlmostNothing

__x86.get_pc_thunk.ax reference

- doAlmostNothing calls __x86.get_pc_thunk.ax
- function symbol values in nmain.out

```
doAlmostNothing       : 0x53d <---
__x86.get_pc_thunk.ax : 0x519
```

- in nothing.o

```
23:  e8 fc ff ff ff          call   24 <doAlmostNothing+0x4>
     0x519 + 0x04 + 0x24 = 0x541 = refaddr ... 0xfffffffc, 24
     0x53d + 0x04 = 0x541 = refaddr        ... <doAlmostNothing+0x4>
```

- in nmain.out

```
540:  e8 d4 ff ff ff          call   519 <__x86.get_pc_thunk.ax>
      0x519 - 0x541 - 4 = -0x2c = *refaddr
```

symbol references (4) in doAlmostNothing

doNothingStatic reference

- doAlmostNothing calls doNothingStatic
- function symbol values in nmain.out

```
doAlmostNothing       : 0x53d
doNothingStatic       : 0x51d <---
```

- in nothing.o

```
2d:  e8 ce ff ff ff          call   0 <doNothingStatic>
     0x519 + 0x32 + 0x0 = 0x54b = refaddr ... 0xffffffce, 0
     0x51d + 0x2e = 0x54b = refaddr       ... <doNothingStatic>
```

- in nmain.out

```
54a:  e8 ce ff ff ff          call   51d <doNothingStatic>
      0x51d - 0x54b -4 = -0x32 = *refaddr
```

symbol references (5) in doAlmostNothing

doNothing reference

- doAlmostNothing calls doNothing
- function symbol values in nmain.out

```
doAlmostNothing        : 0x53d <---
doNothing              : 0x52d
```

- in nothing.o

```
32:   e8 fc ff ff ff          call   33 <doAlmostNothing+0x13>
      0x519 + 0x4 + 0x33 = 0x500 = refaddr ... 0xfffffffc, 33
      0x53d + 0x13 = 0x550 = refaddr       ... <doAlmostNothing+0x13>
```

- in nmain.out

```
54f:   e8 d9 ff ff ff          call   52d <doNothing>
       0x52d - 0x550 - 0x4 = -0x27 = *refaddr
```

main gdb disassemble results

main disassemble results

- (gdb) disas main

```
Dump of assembler code for function main:
   0x004004ed <+0>:   lea    0x4(%esp),%ecx
   0x004004f1 <+4>:   and    $0xfffffff0,%esp
   0x004004f4 <+7>:   pushl  -0x4(%ecx)
   0x004004f7 <+10>:  push   %ebp
   0x004004f8 <+11>:  mov    %esp,%ebp
   0x004004fa <+13>:  push   %ebx
   0x004004fb <+14>:  push   %ecx
   0x004004fc <+15>:  call   0x400519 <__x86.get_pc_thunk.ax>
   0x00400501 <+20>:  add    $0x1adb,%eax
=> 0x00400506 <+25>:  mov    %eax,%ebx
   0x00400508 <+27>:  call   0x40053d <doAlmostNothing>
   0x0040050d <+32>:  mov    $0x0,%eax
   0x00400512 <+37>:  pop    %ecx
   0x00400513 <+38>:  pop    %ebx
   0x00400514 <+39>:  pop    %ebp
   0x00400515 <+40>:  lea    -0x4(%ecx),%esp
   0x00400518 <+43>:  ret
End of assembler dump.
```

doAlmostNothing gdb disassemble results

doAlmostNothing disassemble results

- (gdb) disas doAlmostNothing
  Dump of assembler code for function doAlmostNothing:
  0x0040053d <+0>: push %ebp
  0x0040053e <+1>: mov %esp,%ebp
  0x00400540 <+3>: call 0x400519 <__x86.get_pc_thunk.ax>
  0x00400545 <+8>: add $0x1a97,%eax
  0x0040054a <+13>: call 0x40051d <doNothingStatic>
  0x0040054f <+18>: call 0x40052d <doNothing>
  0x00400554 <+23>: nop
  0x00400555 <+24>: pop %ebp
  0x00400556 <+25>: ret
  End of assembler dump.

doNothingStatic gdb disassemble results

main disassemble results

- (gdb) disas doNothingStatic
  Dump of assembler code for function doNothingStatic:
  0x0040051d <+0>: push %ebp
  0x0040051e <+1>: mov %esp,%ebp
  0x00400520 <+3>: call 0x400519 <__x86.get_pc_thunk.ax>
  0x00400525 <+8>: add $0x1ab7,%eax
  0x0040052a <+13>: nop
  0x0040052b <+14>: pop %ebp
  0x0040052c <+15>: ret
  End of assembler dump.

doNothing gdb disassemble results

doNothing disassemble results

- (gdb) disas doNothing
  Dump of assembler code for function doNothing:
  0x0040052d <+0>: push %ebp
  0x0040052e <+1>: mov %esp,%ebp
  0x00400530 <+3>: call 0x400519 <__x86.get_pc_thunk.ax>
  0x00400535 <+8>: add $0x1aa7,%eax
  0x0040053a <+13>: nop
  0x0040053b <+14>: pop %ebp
  0x0040053c <+15>: ret
  End of assembler dump.