Example I - 2A Swapping array elements

Young W. Lim

2019-03-02 Sat

Outline

1. Based on
2. example 2 : swapping array elements
   - source codes
   - Makefile
3. example 2 (swap) effects of compile and link options
   - relocatable object `swap.o`
   - `swap` in the executable object with no ld option
   - `swap` in the executable object with `-static` ld option
   - `swap` in the executable object with `-no-pie` ld option
   - `swap` in the shared object with no ld option

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

swap.c

```
/*::::: swap.c ::::::::::::::::::::::::::*/
extern int buf[];

int *p0 = &buf[0];
int *p1;

void swap()
{
  int tmp;

  p1 = &buf[1];

  tmp = *p0;
  *p0 = *p1;
  *p1 = tmp;

}
```

main.c

```
/*::::: main.c ::::::::::::::::::::::::::::::*/
void swap();

int buf[2]  = {1, 2};

int main()
{
  swap();

  return 0;
}
```

Makefile (1)

```
CF0 =
CF1 = -fPIC
CF2 = -fno-pic
CF3 = -fno-plt

LF0 =
LF1 = -static
LF2 = -no-pie

all : static dynamic cases so

cases : case0 case1 case2 case3 case4 case5 case6 case7 case8 case9 casea caseb

so : libswap.so libswap_pic.so libswap_nopic.so libswap_noplt.so

clean :
        rm *.o *.a *.so *.out
```

Makefile (2)

```
#-------------------------------------------------------------
static : swap.c main.c
        gcc -m32 -Wall -g -c swap.c
        ar rcs libswap.a swap.o

        gcc -m32 -Wall -g -c main.c
        gcc -m32 -static -o swap.out main.o ./libswap.a


dynamic : swap.c  main.c
        gcc -fPIC -m32 -Wall -g -c swap.c -o swap_pic.o
        gcc -shared -m32 -o libswap.so swap_pic.o

        gcc -m32 -Wall -g -c main.c
        gcc -m32 -o swap_dyn.out main.o ./libswap.so
```

Makefile (3)

```
#-------------------------------------------------------------
libswap.so : swap.c
	gcc $(CF0) -m32 -Wall -c swap.c
	ar rcs libswap.a swap.o

libswap_pic.so : swap.c
	gcc $(CF1) -m32 -Wall -c swap.c -o swap_pic.o
	ar rcs libswap_pic.a swap_pic.o

libswap_nopic.so : swap.c
	gcc $(CF2) -m32 -Wall -c swap.c -o swap_nopic.o
	ar rcs libswap_nopic.a swap_nopic.o

libswap_noplt.so : swap.c
	gcc $(CF3) -m32 -Wall -c swap.c -o swap_noplt.o
	ar rcs libswap_noplt.a swap_noplt.o
```

Makefile (4)

```
#-------------------------------------------------------------
libswap.a : swap.c
	gcc $(CF0) -m32 -Wall -c swap.c
	ar rcs libswap.a swap.o

libswap_pic.a : swap.c
	gcc $(CF1) -m32 -Wall -c swap.c -o swap_pic.o
	ar rcs libswap_pic.a swap_pic.o

libswap_nopic.a : swap.c
	gcc $(CF2) -m32 -Wall -c swap.c -o swap_nopic.o
	ar rcs libswap_nopic.a swap_nopic.o

libswap_noplt.a : swap.c
	gcc $(CF3) -m32 -Wall -c swap.c -o swap_noplt.o
	ar rcs libswap_noplt.a swap_noplt.o
```

Makefile (5)

```
#------------------------------------------------------------
case0 : main.c libswap.a
        gcc -m32 -Wall -c main.c
        gcc -m32 $(LF0)-o swap_0.out main.o ./libswap.a

case1 : main.c libswap_pic.a
        gcc -m32 -Wall -c main.c
        gcc -m32 $(LF0)-o swap_1_pic.out main.o ./libswap_pic.a

case2 : main.c libswap_nopic.a
        gcc -m32 -Wall -c main.c
        gcc -m32 $(LF0)-o swap_2_nopic.out main.o ./libswap_nopic.a

case3 : main.c libswap_noplt.a
        gcc -m32 -Wall -c main.c
        gcc -m32 $(LF0) -o swap_3_noplt.out main.o ./libswap_noplt.a
```

Makefile (6)

```
#-------------------------------------------------------------
case4 : main.c libswap.a
        gcc -m32 -Wall -c main.c
        gcc -m32 $(LF1) -o swap_4_static.out main.o ./libswap.a

case5 : main.c libswap_pic.a
        gcc -m32 -Wall -c main.c
        gcc -m32 $(LF1) -o swap_5_pic_static.out main.o ./libswap_pic.a

case6 : main.c libswap_nopic.a
        gcc -m32 -Wall -c main.c
        gcc -m32 $(LF1) -o swap_6_nopic_static.out main.o ./libswap_nopic.a

case7 : main.c libswap_noplt.a
        gcc -m32 -Wall -c main.c
        gcc -m32 $(LF1) -o swap_7_noplt_static.out main.o ./libswap_noplt.a
```

Makefile (7)

```
#-------------------------------------------------------------
case8 : main.c libswap.a
        gcc -m32 -Wall -c main.c
        gcc -m32 $(LF2) -o swap_8_nopie.out main.o ./libswap.a

case9 : main.c libswap_pic.a
        gcc -m32 -Wall -c main.c
        gcc -m32 $(LF2) -o swap_9_pic_nopie.out main.o ./libswap_pic.a

casea : main.c libswap_nopic.a
        gcc -m32 -Wall -c main.c
        gcc -m32 $(LF2) -o swap_a_nopic_nopie.out main.o ./libswap_nopic.a

caseb : main.c libswap_noplt.a
        gcc -m32 -Wall -c main.c
        gcc -m32 $(LF2) -o swap_b_noplt_nopie.out main.o ./libswap_noplt.a
```

objview.bash

```
#!/bin/bash

for i in $( ls func1*.o ); do
  objdump -drS $i > $i.txt
done

for i in $( ls *.out *.so ); do
  objdump -drS $i | sed  -n  '/<swao>:/, /^$/p' > $i.txt
done
```

analyzing commands

- $ readelf --segments swap_dyn.out
  $ objdump -d -s swap_dyn.out
  $ objdump -d -j .plt.got swap_dyn.out
  $ objdump -d -j .plt.got swap_dyn.out
  $ gdb ... disas, x/a 0x...., c
  $ cat /proc/<pid>/map

swap.o (1)

objdump -drS swap.o

```
00000000 <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
   0:   55                      push   %ebp
   1:   89 e5                   mov    %esp,%ebp
   3:   83 ec 10                sub    $0x10,%esp
   6:   e8 fc ff ff ff          call   7 <swap+0x7>
                        7: R_386_PC32   __x86.get_pc_thunk.ax
   b:   05 01 00 00 00          add    $0x1,%eax
                        c: R_386_GOTPC  _GLOBAL_OFFSET_TABLE_
```

swap.o (2)

objdump -drS swap.o

```
int tmp;
p1 = &buf[1];
10:   8b 90 00 00 00 00        mov    0x0(%eax),%edx
                      12: R_386_GOT32X         p1
16:   8b 88 00 00 00 00        mov    0x0(%eax),%ecx
                      18: R_386_GOT32X         buf
1c:   8d 49 04                 lea    0x4(%ecx),%ecx
1f:   89 0a                    mov    %ecx,(%edx)
tmp = *p0;
21:   8b 90 00 00 00 00        mov    0x0(%eax),%edx
                      23: R_386_GOTOFF         p0
27:   8b 12                    mov    (%edx),%edx
29:   89 55 fc                 mov    %edx,-0x4(%ebp)
```

swap.o (3)

objdump -drS swap.o

```
  *p0 = *p1;
  2c:   8b 90 00 00 00 00       mov    0x0(%eax),%edx
                        2e: R_386_GOT32X        p1
  32:   8b 0a                   mov    (%edx),%ecx
  34:   8b 90 00 00 00 00       mov    0x0(%eax),%edx
                        36: R_386_GOTOFF        p0
  3a:   8b 09                   mov    (%ecx),%ecx
  3c:   89 0a                   mov    %ecx,(%edx)
  *p1 = tmp;
  3e:   8b 80 00 00 00 00       mov    0x0(%eax),%eax
                        40: R_386_GOT32X        p1
  44:   8b 00                   mov    (%eax),%eax
  46:   8b 55 fc                mov    -0x4(%ebp),%edx
  49:   89 10                   mov    %edx,(%eax)
}
  4b:   90                      nop
  4c:   c9                      leave
  4d:   c3                      ret
```

swap_pic.o with -fPIC (1)

objdump -drS swap_pic.o

```
00000000 <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
   0:   55                      push   %ebp
   1:   89 e5                   mov    %esp,%ebp
   3:   83 ec 10                sub    $0x10,%esp
   6:   e8 fc ff ff ff          call   7 <swap+0x7>
                        7: R_386_PC32    __x86.get_pc_thunk.ax
   b:   05 01 00 00 00          add    $0x1,%eax
                        c: R_386_GOTPC  _GLOBAL_OFFSET_TABLE_
```

swap_pic.o with -fPIC (2)

objdump -drS swap_pic.o

```
int tmp;
p1 = &buf[1];
10:   8b 90 00 00 00 00       mov    0x0(%eax),%edx
                      12: R_386_GOT32X        p1
16:   8b 88 00 00 00 00       mov    0x0(%eax),%ecx
                      18: R_386_GOT32X        buf
1c:   8d 49 04                lea    0x4(%ecx),%ecx
1f:   89 0a                   mov    %ecx,(%edx)
tmp = *p0;
21:   8b 90 00 00 00 00       mov    0x0(%eax),%edx
                      23: R_386_GOT32X        p0
27:   8b 12                   mov    (%edx),%edx
29:   8b 12                   mov    (%edx),%edx
2b:   89 55 fc                mov    %edx,-0x4(%ebp)
```

swap_pic.o with -fPIC (3)

objdump -drS swap_pic.o

```
  *p0 = *p1;
  2e:   8b 90 00 00 00 00       mov    0x0(%eax),%edx
                        30: R_386_GOT32X        p1
  34:   8b 0a                   mov    (%edx),%ecx
  36:   8b 90 00 00 00 00       mov    0x0(%eax),%edx
                        38: R_386_GOT32X        p0
  3c:   8b 12                   mov    (%edx),%edx
  3e:   8b 09                   mov    (%ecx),%ecx
  40:   89 0a                   mov    %ecx,(%edx)
  *p1 = tmp;
  42:   8b 80 00 00 00 00       mov    0x0(%eax),%eax
                        44: R_386_GOT32X        p1
  48:   8b 00                   mov    (%eax),%eax
  4a:   8b 55 fc                mov    -0x4(%ebp),%edx
  4d:   89 10                   mov    %edx,(%eax)
}
  4f:   90                      nop
  50:   c9                      leave
  51:   c3                      ret
```

swap_nopic.o with -fno-pic (1)

objdump -drS swap_nopic.o

```
00000000 <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
   0:   55                      push   %ebp
   1:   89 e5                   mov    %esp,%ebp
   3:   83 ec 10                sub    $0x10,%esp
```

swap_nopic.o with -fno-pic (2)

objdump -drS swap_nopic.o

```
int tmp;
p1 = &buf[1];
 6:   c7 05 00 00 00 00 04    movl   $0x4,0x0
 d:   00 00 00
                      8: R_386_32     p1
                      c: R_386_32     buf
tmp = *p0;
10:   a1 00 00 00 00          mov    0x0,%eax
                      11: R_386_32    p0
15:   8b 00                   mov    (%eax),%eax
17:   89 45 fc                mov    %eax,-0x4(%ebp)
```

swap_nopic.o with -fno-pic (3)

objdump -drS swap_nopic.o

```
  *p0 = *p1;
  1a:   8b 15 00 00 00 00       mov    0x0,%edx
                        1c: R_386_32    p1
  20:   a1 00 00 00 00          mov    0x0,%eax
                        21: R_386_32    p0
  25:   8b 12                   mov    (%edx),%edx
  27:   89 10                   mov    %edx,(%eax)
  *p1 = tmp;
  29:   a1 00 00 00 00          mov    0x0,%eax
                        2a: R_386_32    p1
  2e:   8b 55 fc                mov    -0x4(%ebp),%edx
  31:   89 10                   mov    %edx,(%eax)
}
  33:   90                      nop
  34:   c9                      leave
  35:   c3                      ret
```

swap_noplt.o with -fno-plt (1)

objdump -drS swap_noplt.o

```
00000000 <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
   0:   55                      push   %ebp
   1:   89 e5                   mov    %esp,%ebp
   3:   83 ec 10                sub    $0x10,%esp
   6:   e8 fc ff ff ff          call   7 <swap+0x7>
                        7: R_386_PC32    __x86.get_pc_thunk.ax
   b:   05 01 00 00 00          add    $0x1,%eax
                        c: R_386_GOTPC  _GLOBAL_OFFSET_TABLE_
```

swap_noplt.o with -fno-plt (2)

objdump -drS swap_noplt.o

```
int tmp;
p1 = &buf[1];
10:   8b 90 00 00 00 00       mov    0x0(%eax),%edx
                      12: R_386_GOT32X        p1
16:   8b 88 00 00 00 00       mov    0x0(%eax),%ecx
                      18: R_386_GOT32X        buf
1c:   8d 49 04                lea    0x4(%ecx),%ecx
1f:   89 0a                   mov    %ecx,(%edx)
tmp = *p0;
21:   8b 90 00 00 00 00       mov    0x0(%eax),%edx
                      23: R_386_GOTOFF        p0
27:   8b 12                   mov    (%edx),%edx
29:   89 55 fc                mov    %edx,-0x4(%ebp)
```

swap_noplt.o with -fno-plt (3)

objdump -drS swap_noplt.o

```
  *p0 = *p1;
  2c:   8b 90 00 00 00 00       mov    0x0(%eax),%edx
                        2e: R_386_GOT32X        p1
  32:   8b 0a                   mov    (%edx),%ecx
  34:   8b 90 00 00 00 00       mov    0x0(%eax),%edx
                        36: R_386_GOTOFF        p0
  3a:   8b 09                   mov    (%ecx),%ecx
  3c:   89 0a                   mov    %ecx,(%edx)
  *p1 = tmp;
  3e:   8b 80 00 00 00 00       mov    0x0(%eax),%eax
                        40: R_386_GOT32X        p1
  44:   8b 00                   mov    (%eax),%eax
  46:   8b 55 fc                mov    -0x4(%ebp),%edx
  49:   89 10                   mov    %edx,(%eax)
}
  4b:   90                      nop
  4c:   c9                      leave
  4d:   c3                      ret
```

case 0: swap in swap_0.out (1)

objdump -d swap_0.out

```
0000052d <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
 52d:   55                      push   %ebp
 52e:   89 e5                   mov    %esp,%ebp
 530:   83 ec 10                sub    $0x10,%esp
 533:   e8 f1 ff ff ff          call   529 <__x86.get_pc_thunk.ax>
 538:   05 a4 1a 00 00          add    $0x1aa4,%eax
```

case 0: swap in swap_0.out (2)

objdump -d swap_0.out

```
 int tmp;
 p1 = &buf[1];
53d:   8d 90 3c 00 00 00       lea    0x3c(%eax),%edx
543:   8d 88 2c 00 00 00       lea    0x2c(%eax),%ecx
549:   8d 49 04                lea    0x4(%ecx),%ecx
54c:   89 0a                   mov    %ecx,(%edx)
 tmp = *p0;
54e:   8b 90 34 00 00 00       mov    0x34(%eax),%edx
554:   8b 12                   mov    (%edx),%edx
556:   89 55 fc                mov    %edx,-0x4(%ebp)
```

case 0: swap in swap_0.out (3)

objdump -dS swap_0.out

```
  *p0 = *p1;
 559:   8d 90 3c 00 00 00       lea    0x3c(%eax),%edx
 55f:   8b 0a                   mov    (%edx),%ecx
 561:   8b 90 34 00 00 00       mov    0x34(%eax),%edx
 567:   8b 09                   mov    (%ecx),%ecx
 569:   89 0a                   mov    %ecx,(%edx)
  *p1 = tmp;
 56b:   8d 80 3c 00 00 00       lea    0x3c(%eax),%eax
 571:   8b 00                   mov    (%eax),%eax
 573:   8b 55 fc                mov    -0x4(%ebp),%edx
 576:   89 10                   mov    %edx,(%eax)
}
 578:   90                      nop
 579:   c9                      leave
 57a:   c3                      ret
 57b:   66 90                   xchg   %ax,%ax
 57d:   66 90                   xchg   %ax,%ax
 57f:   90                      nop
```

case 1: swap in swap_1_pic.out (1)

objdump -dS swap_1_pic.out

```
0000052d <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
 52d:   55                      push   %ebp
 52e:   89 e5                   mov    %esp,%ebp
 530:   83 ec 10                sub    $0x10,%esp
 533:   e8 f1 ff ff ff          call   529 <__x86.get_pc_thunk.ax>
 538:   05 a4 1a 00 00          add    $0x1aa4,%eax
```

case 1: swap in swap_1_pic.out (2)

objdump -dS swap_1_pic.out

```
 int tmp;
 p1 = &buf[1];
53d:   8d 90 3c 00 00 00       lea    0x3c(%eax),%edx
543:   8d 88 2c 00 00 00       lea    0x2c(%eax),%ecx
549:   8d 49 04                lea    0x4(%ecx),%ecx
54c:   89 0a                   mov    %ecx,(%edx)
 tmp = *p0;
54e:   8d 90 34 00 00 00       lea    0x34(%eax),%edx
554:   8b 12                   mov    (%edx),%edx
556:   8b 12                   mov    (%edx),%edx
558:   89 55 fc                mov    %edx,-0x4(%ebp)
```

case 1: swap in swap_1_pic.out (3)

objdump -dS swap_1_pic.out

```
  *p0 = *p1;
 55b:   8d 90 3c 00 00 00       lea    0x3c(%eax),%edx
 561:   8b 0a                   mov    (%edx),%ecx
 563:   8d 90 34 00 00 00       lea    0x34(%eax),%edx
 569:   8b 12                   mov    (%edx),%edx
 56b:   8b 09                   mov    (%ecx),%ecx
 56d:   89 0a                   mov    %ecx,(%edx)
  *p1 = tmp;
 56f:   8d 80 3c 00 00 00       lea    0x3c(%eax),%eax
 575:   8b 00                   mov    (%eax),%eax
 577:   8b 55 fc                mov    -0x4(%ebp),%edx
 57a:   89 10                   mov    %edx,(%eax)
}
 57c:   90                      nop
 57d:   c9                      leave
 57e:   c3                      ret
 57f:   90                      nop
```

case 2: swap in swap_2_nopic.out (1)

objdump -dS swap_2_nopic.out

```
0000055d <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
 55d:   55                      push   %ebp
 55e:   89 e5                   mov    %esp,%ebp
 560:   83 ec 10                sub    $0x10,%esp
```

case 2: swap in swap_2_nopic.out (2)

objdump -dS swap_2_nopic.out

```
 int tmp;
 p1 = &buf[1];
563:   c7 05 18 20 00 00 0c    movl   $0x200c,0x2018
56a:   20 00 00
 tmp = *p0;
56d:   a1 10 20 00 00          mov    0x2010,%eax
572:   8b 00                   mov    (%eax),%eax
574:   89 45 fc                mov    %eax,-0x4(%ebp)
```

case 2: swap in swap_2_nopic.out (3)

objdump -dS swap_2_nopic.out

```
  *p0 = *p1;
 577:   8b 15 18 20 00 00       mov    0x2018,%edx
 57d:   a1 10 20 00 00          mov    0x2010,%eax
 582:   8b 12                   mov    (%edx),%edx
 584:   89 10                   mov    %edx,(%eax)
  *p1 = tmp;
 586:   a1 18 20 00 00          mov    0x2018,%eax
 58b:   8b 55 fc                mov    -0x4(%ebp),%edx
 58e:   89 10                   mov    %edx,(%eax)
}
 590:   90                      nop
 591:   c9                      leave
 592:   c3                      ret
 593:   66 90                   xchg   %ax,%ax
 595:   66 90                   xchg   %ax,%ax
 597:   66 90                   xchg   %ax,%ax
 599:   66 90                   xchg   %ax,%ax
 59b:   66 90                   xchg   %ax,%ax
 59d:   66 90                   xchg   %ax,%ax
 59f:   90                      nop
```

case 3: swap in swap_3_noplt.out (1)

objdump -dS swap_3_noplt.out

```
0000052d <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
 52d:   55                      push   %ebp
 52e:   89 e5                   mov    %esp,%ebp
 530:   83 ec 10                sub    $0x10,%esp
 533:   e8 f1 ff ff ff          call   529 <__x86.get_pc_thunk.ax>
 538:   05 a4 1a 00 00          add    $0x1aa4,%eax
```

case 3: swap in swap_3_noplt.out (2)

objdump -dS swap_3_noplt.out

```
 int tmp;
 p1 = &buf[1];
53d:   8d 90 3c 00 00 00       lea    0x3c(%eax),%edx
543:   8d 88 2c 00 00 00       lea    0x2c(%eax),%ecx
549:   8d 49 04                lea    0x4(%ecx),%ecx
54c:   89 0a                   mov    %ecx,(%edx)
 tmp = *p0;
54e:   8b 90 34 00 00 00       mov    0x34(%eax),%edx
554:   8b 12                   mov    (%edx),%edx
556:   89 55 fc                mov    %edx,-0x4(%ebp)
```

case 3: swap in swap_3_noplt.out (3)

objdump -dS swap_3_noplt.out

```
  *p0 = *p1;
 559:   8d 90 3c 00 00 00       lea    0x3c(%eax),%edx
 55f:   8b 0a                   mov    (%edx),%ecx
 561:   8b 90 34 00 00 00       mov    0x34(%eax),%edx
 567:   8b 09                   mov    (%ecx),%ecx
 569:   89 0a                   mov    %ecx,(%edx)
  *p1 = tmp;
 56b:   8d 80 3c 00 00 00       lea    0x3c(%eax),%eax
 571:   8b 00                   mov    (%eax),%eax
 573:   8b 55 fc                mov    -0x4(%ebp),%edx
 576:   89 10                   mov    %edx,(%eax)
}
 578:   90                      nop
 579:   c9                      leave
 57a:   c3                      ret
 57b:   66 90                   xchg   %ax,%ax
 57d:   66 90                   xchg   %ax,%ax
 57f:   90                      nop
```

case 4: swap in swap_4_static.out (1)

objdump -dS swap_4_static.out

```
080488d5 <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
 80488d5:       55                      push   %ebp
 80488d6:       89 e5                   mov    %esp,%ebp
 80488d8:       83 ec 10                sub    $0x10,%esp
 80488db:       e8 f1 ff ff ff          call   80488d1 <__x86.get_pc_thunk.ax>
 80488e0:       05 20 07 09 00          add    $0x90720,%eax
```

case 4: swap in swap_4_static.out (2)

objdump -dS swap_4_static.out

```
 int tmp;
 p1 = &buf[1];
80488e5:       c7 c2 c4 ac 0d 08       mov    $0x80dacc4,%edx
80488eb:       c7 c1 68 90 0d 08       mov    $0x80d9068,%ecx
80488f1:       8d 49 04                lea    0x4(%ecx),%ecx
80488f4:       89 0a                   mov    %ecx,(%edx)
 tmp = *p0;
80488f6:       8b 90 70 00 00 00       mov    0x70(%eax),%edx
80488fc:       8b 12                   mov    (%edx),%edx
80488fe:       89 55 fc                mov    %edx,-0x4(%ebp)
```

case 4: swap in swap_4_static.out (3)

objdump -dS swap_4_static.out

```
  *p0 = *p1;
 8048901:       c7 c2 c4 ac 0d 08       mov    $0x80dacc4,%edx
 8048907:       8b 0a                   mov    (%edx),%ecx
 8048909:       8b 90 70 00 00 00       mov    0x70(%eax),%edx
 804890f:       8b 09                   mov    (%ecx),%ecx
 8048911:       89 0a                   mov    %ecx,(%edx)
  *p1 = tmp;
 8048913:       c7 c0 c4 ac 0d 08       mov    $0x80dacc4,%eax
 8048919:       8b 00                   mov    (%eax),%eax
 804891b:       8b 55 fc                mov    -0x4(%ebp),%edx
 804891e:       89 10                   mov    %edx,(%eax)
}
 8048920:       90                      nop
 8048921:       c9                      leave
 8048922:       c3                      ret
 8048923:       66 90                   xchg   %ax,%ax
 ...
```

case 5: swap in swap_5_pic_static.out (1)

objdump -dS swap_5_pic_static.out

```
080488d5 <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
 80488d5:       55                      push   %ebp
 80488d6:       89 e5                   mov    %esp,%ebp
 80488d8:       83 ec 10                sub    $0x10,%esp
 80488db:       e8 f1 ff ff ff          call   80488d1 <__x86.get_pc_thunk.ax>
 80488e0:       05 20 07 09 00          add    $0x90720,%eax
```

case 5: swap in swap_5_pic_static.out (2)

objdump -dS swap_5_pic_static.out

```
 int tmp;
 p1 = &buf[1];
80488e5:        c7 c2 c4 ac 0d 08        mov    $0x80dacc4,%edx
80488eb:        c7 c1 68 90 0d 08        mov    $0x80d9068,%ecx
80488f1:        8d 49 04                 lea    0x4(%ecx),%ecx
80488f4:        89 0a                    mov    %ecx,(%edx)
 tmp = *p0;
80488f6:        c7 c2 70 90 0d 08        mov    $0x80d9070,%edx
80488fc:        8b 12                    mov    (%edx),%edx
80488fe:        8b 12                    mov    (%edx),%edx
8048900:        89 55 fc                 mov    %edx,-0x4(%ebp)
```

case 5: swap in swap_5_pic_static.out (3)

objdump -dS swap_5_pic_static.out

```
  *p0 = *p1;
 8048903:       c7 c2 c4 ac 0d 08       mov    $0x80dacc4,%edx
 8048909:       8b 0a                   mov    (%edx),%ecx
 804890b:       c7 c2 70 90 0d 08       mov    $0x80d9070,%edx
 8048911:       8b 12                   mov    (%edx),%edx
 8048913:       8b 09                   mov    (%ecx),%ecx
 8048915:       89 0a                   mov    %ecx,(%edx)
  *p1 = tmp;
 8048917:       c7 c0 c4 ac 0d 08       mov    $0x80dacc4,%eax
 804891d:       8b 00                   mov    (%eax),%eax
 804891f:       8b 55 fc                mov    -0x4(%ebp),%edx
 8048922:       89 10                   mov    %edx,(%eax)
}
 8048924:       90                      nop
 8048925:       c9                      leave
 8048926:       c3                      ret
 8048927:       66 90                   xchg   %ax,%ax
 ...
```

case 6: swap in swap_6_nopic_static.out (1)

objdump -dS swap_6_nopic_static.out

```
080488d5 <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
 80488d5:       55                      push   %ebp
 80488d6:       89 e5                   mov    %esp,%ebp
 80488d8:       83 ec 10                sub    $0x10,%esp
```

case 6: swap in swap_6_nopic_static.out (2)

objdump -dS swap_6_nopic_static.out

```
 int tmp;
 p1 = &buf[1];
80488db:        c7 05 c4 ac 0d 08 6c    movl   $0x80d906c,0x80dacc4
80488e2:        90 0d 08
 tmp = *p0;
80488e5:        a1 70 90 0d 08          mov    0x80d9070,%eax
80488ea:        8b 00                   mov    (%eax),%eax
80488ec:        89 45 fc                mov    %eax,-0x4(%ebp)
```

case 6: swap in swap_6_nopic_static.out (3)

objdump -dS swap_6_nopic_static.out

```
  *p0 = *p1;
 80488ef:       8b 15 c4 ac 0d 08       mov    0x80dacc4,%edx
 80488f5:       a1 70 90 0d 08          mov    0x80d9070,%eax
 80488fa:       8b 12                   mov    (%edx),%edx
 80488fc:       89 10                   mov    %edx,(%eax)
  *p1 = tmp;
 80488fe:       a1 c4 ac 0d 08          mov    0x80dacc4,%eax
 8048903:       8b 55 fc                mov    -0x4(%ebp),%edx
 8048906:       89 10                   mov    %edx,(%eax)
}
 8048908:       90                      nop
 8048909:       c9                      leave
 804890a:       c3                      ret
 804890b:       66 90                   xchg   %ax,%ax
 804890d:       66 90                   xchg   %ax,%ax
 804890f:       90                      nop
```

case 7: swap in swap_7_noplt_static.out (1)

objdump -dS swap_7_noplt_static.out

```
080488d5 <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
 80488d5:       55                      push   %ebp
 80488d6:       89 e5                   mov    %esp,%ebp
 80488d8:       83 ec 10                sub    $0x10,%esp
 80488db:       e8 f1 ff ff ff          call   80488d1 <__x86.get_pc_thunk.ax>
 80488e0:       05 20 07 09 00          add    $0x90720,%eax
```

case 7: swap in swap_7_noplt_static.out (2)

objdump -dS swap_7_noplt_static.out

```
 int tmp;
 p1 = &buf[1];
80488e5:       c7 c2 c4 ac 0d 08       mov    $0x80dacc4,%edx
80488eb:       c7 c1 68 90 0d 08       mov    $0x80d9068,%ecx
80488f1:       8d 49 04                lea    0x4(%ecx),%ecx
80488f4:       89 0a                   mov    %ecx,(%edx)
 tmp = *p0;
80488f6:       8b 90 70 00 00 00       mov    0x70(%eax),%edx
80488fc:       8b 12                   mov    (%edx),%edx
80488fe:       89 55 fc                mov    %edx,-0x4(%ebp)
```

case 7: swap in swap_7_noplt_static.out (3)

objdump -dS swap_7_noplt_static.out

```
  *p0 = *p1;
 8048901:       c7 c2 c4 ac 0d 08       mov    $0x80dacc4,%edx
 8048907:       8b 0a                   mov    (%edx),%ecx
 8048909:       8b 90 70 00 00 00       mov    0x70(%eax),%edx
 804890f:       8b 09                   mov    (%ecx),%ecx
 8048911:       89 0a                   mov    %ecx,(%edx)
  *p1 = tmp;
 8048913:       c7 c0 c4 ac 0d 08       mov    $0x80dacc4,%eax
 8048919:       8b 00                   mov    (%eax),%eax
 804891b:       8b 55 fc                mov    -0x4(%ebp),%edx
 804891e:       89 10                   mov    %edx,(%eax)
}
 8048920:       90                      nop
 8048921:       c9                      leave
 8048922:       c3                      ret
 8048923:       66 90                   xchg   %ax,%ax
 ...
```

case 8: swap in swap_8_nopie.out (1)

objdump -dS nets_8_nopie.out

```
08048426 <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
 8048426:       55                      push   %ebp
 8048427:       89 e5                   mov    %esp,%ebp
 8048429:       83 ec 10                sub    $0x10,%esp
 804842c:       e8 f1 ff ff ff          call   8048422 <__x86.get_pc_thunk.ax>
 8048431:       05 cf 1b 00 00          add    $0x1bcf,%eax
```

case 8: swap in swap_8_nopie.out (2)

objdump -dS nets_8_nopie.out

```
 int tmp;
 p1 = &buf[1];
8048436:       c7 c2 28 a0 04 08       mov    $0x804a028,%edx
804843c:       c7 c1 18 a0 04 08       mov    $0x804a018,%ecx
8048442:       8d 49 04                lea    0x4(%ecx),%ecx
8048445:       89 0a                   mov    %ecx,(%edx)
 tmp = *p0;
8048447:       8b 90 20 00 00 00       mov    0x20(%eax),%edx
804844d:       8b 12                   mov    (%edx),%edx
804844f:       89 55 fc                mov    %edx,-0x4(%ebp)
```

case 8: swap in swap_8_nopie.out (3)

objdump -dS nets_8_nopie.out

```
  *p0 = *p1;
 8048452:       c7 c2 28 a0 04 08       mov    $0x804a028,%edx
 8048458:       8b 0a                   mov    (%edx),%ecx
 804845a:       8b 90 20 00 00 00       mov    0x20(%eax),%edx
 8048460:       8b 09                   mov    (%ecx),%ecx
 8048462:       89 0a                   mov    %ecx,(%edx)
  *p1 = tmp;
 8048464:       c7 c0 28 a0 04 08       mov    $0x804a028,%eax
 804846a:       8b 00                   mov    (%eax),%eax
 804846c:       8b 55 fc                mov    -0x4(%ebp),%edx
 804846f:       89 10                   mov    %edx,(%eax)
}
 8048471:       90                      nop
 8048472:       c9                      leave
 8048473:       c3                      ret
 8048474:       66 90                   xchg   %ax,%ax
 ...
```

case 9: swap in swap_9_pic_nopie.out (1)

objdump -dS swap_9_pic_nopie.out

```
08048426 <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
 8048426:       55                      push   %ebp
 8048427:       89 e5                   mov    %esp,%ebp
 8048429:       83 ec 10                sub    $0x10,%esp
 804842c:       e8 f1 ff ff ff          call   8048422 <__x86.get_pc_thunk.ax>
 8048431:       05 cf 1b 00 00          add    $0x1bcf,%eax
```

case 9: swap in swap_9_pic_nopie.out (2)

objdump -dS swap_9_pic_nopie.out

```
 int tmp;
 p1 = &buf[1];
8048436:       c7 c2 28 a0 04 08       mov    $0x804a028,%edx
804843c:       c7 c1 18 a0 04 08       mov    $0x804a018,%ecx
8048442:       8d 49 04                lea    0x4(%ecx),%ecx
8048445:       89 0a                   mov    %ecx,(%edx)
 tmp = *p0;
8048447:       c7 c2 20 a0 04 08       mov    $0x804a020,%edx
804844d:       8b 12                   mov    (%edx),%edx
804844f:       8b 12                   mov    (%edx),%edx
8048451:       89 55 fc                mov    %edx,-0x4(%ebp)
```

case 9: swap in swap_9_pic_nopie.out (3)

objdump -dS swap_9_pic_nopie.out

```
  *p0 = *p1;
 8048454:       c7 c2 28 a0 04 08       mov    $0x804a028,%edx
 804845a:       8b 0a                   mov    (%edx),%ecx
 804845c:       c7 c2 20 a0 04 08       mov    $0x804a020,%edx
 8048462:       8b 12                   mov    (%edx),%edx
 8048464:       8b 09                   mov    (%ecx),%ecx
 8048466:       89 0a                   mov    %ecx,(%edx)
  *p1 = tmp;
 8048468:       c7 c0 28 a0 04 08       mov    $0x804a028,%eax
 804846e:       8b 00                   mov    (%eax),%eax
 8048470:       8b 55 fc                mov    -0x4(%ebp),%edx
 8048473:       89 10                   mov    %edx,(%eax)
}
 8048475:       90                      nop
 8048476:       c9                      leave
 8048477:       c3                      ret
 8048478:       66 90                   xchg   %ax,%ax
 ...
```

case 10: swap in swap_10_nopic_nopie.out (1)

objdump -dS swap_10_nopic_nopie.out

```
08048426 <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
 8048426:       55                      push   %ebp
 8048427:       89 e5                   mov    %esp,%ebp
 8048429:       83 ec 10                sub    $0x10,%esp
```

case 10: swap in swap_10_nopic_nopie.out (2)

objdump -dS swap_10_nopic_nopie.out

```
 int tmp;
 p1 = &buf[1];
804842c:       c7 05 28 a0 04 08 1c    movl   $0x804a01c,0x804a028
8048433:       a0 04 08
 tmp = *p0;
8048436:       a1 20 a0 04 08          mov    0x804a020,%eax
804843b:       8b 00                   mov    (%eax),%eax
804843d:       89 45 fc                mov    %eax,-0x4(%ebp)
```

case 10: swap in swap_10_nopic_nopie.out (3)

objdump -dS swap_10_nopic_nopie.out

```
  *p0 = *p1;
 8048440:       8b 15 28 a0 04 08       mov    0x804a028,%edx
 8048446:       a1 20 a0 04 08          mov    0x804a020,%eax
 804844b:       8b 12                   mov    (%edx),%edx
 804844d:       89 10                   mov    %edx,(%eax)
  *p1 = tmp;
 804844f:       a1 28 a0 04 08          mov    0x804a028,%eax
 8048454:       8b 55 fc                mov    -0x4(%ebp),%edx
 8048457:       89 10                   mov    %edx,(%eax)
}
 8048459:       90                      nop
 804845a:       c9                      leave
 804845b:       c3                      ret
 804845c:       66 90                   xchg   %ax,%ax
 804845e:       66 90                   xchg   %ax,%ax
```

case 11: swap in swap_11_noplt_nopie.out (1)

objdump -dS swap_11_noplt_nopie.out

```
08048426 <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
 8048426:       55                      push   %ebp
 8048427:       89 e5                   mov    %esp,%ebp
 8048429:       83 ec 10                sub    $0x10,%esp
 804842c:       e8 f1 ff ff ff          call   8048422 <__x86.get_pc_thunk.ax>
 8048431:       05 cf 1b 00 00          add    $0x1bcf,%eax
```

case 11: swap in swap_11_noplt_nopie.out (2)

objdump -dS swap_11_noplt_nopie.out

```
 int tmp;
 p1 = &buf[1];
8048436:       c7 c2 28 a0 04 08       mov    $0x804a028,%edx
804843c:       c7 c1 18 a0 04 08       mov    $0x804a018,%ecx
8048442:       8d 49 04                lea    0x4(%ecx),%ecx
8048445:       89 0a                   mov    %ecx,(%edx)
 tmp = *p0;
8048447:       8b 90 20 00 00 00       mov    0x20(%eax),%edx
804844d:       8b 12                   mov    (%edx),%edx
804844f:       89 55 fc                mov    %edx,-0x4(%ebp)
```

case 11: swap in swap_11_noplt_nopie.out (3)

objdump -dS swap_11_noplt_nopie.out

```
  *p0 = *p1;
 8048452:       c7 c2 28 a0 04 08       mov    $0x804a028,%edx
 8048458:       8b 0a                   mov    (%edx),%ecx
 804845a:       8b 90 20 00 00 00       mov    0x20(%eax),%edx
 8048460:       8b 09                   mov    (%ecx),%ecx
 8048462:       89 0a                   mov    %ecx,(%edx)
  *p1 = tmp;
 8048464:       c7 c0 28 a0 04 08       mov    $0x804a028,%eax
 804846a:       8b 00                   mov    (%eax),%eax
 804846c:       8b 55 fc                mov    -0x4(%ebp),%edx
 804846f:       89 10                   mov    %edx,(%eax)
}
 8048471:       90                      nop
 8048472:       c9                      leave
 8048473:       c3                      ret
 8048474:       66 90                   xchg   %ax,%ax
 ...
```

case 0: swap in $\texttt{swap}_0.out$

```
objdump -dS swap_0.out
```

case 0: swap in $swap_0.out$

```
objdump -dS swap_0.out
```

swap in libswap.so (1)

objdump -dS nets_8_nopie.out

```
0000047d <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
 47d:   55                      push   %ebp
 47e:   89 e5                   mov    %esp,%ebp
 480:   83 ec 10                sub    $0x10,%esp
 483:   e8 43 00 00 00          call   4cb <__x86.get_pc_thunk.ax>
 488:   05 78 1b 00 00          add    $0x1b78,%eax
```

swap in libswap.so (2)

objdump -dS nets_8_nopie.out

```
 int tmp;
 p1 = &buf[1];
48d:   8b 90 ec ff ff ff       mov    -0x14(%eax),%edx
493:   8b 88 f8 ff ff ff       mov    -0x8(%eax),%ecx
499:   8d 49 04                lea    0x4(%ecx),%ecx
49c:   89 0a                   mov    %ecx,(%edx)
 tmp = *p0;
49e:   8b 90 10 00 00 00       mov    0x10(%eax),%edx
4a4:   8b 12                   mov    (%edx),%edx
4a6:   89 55 fc                mov    %edx,-0x4(%ebp)
```

swap in libswap.so (3)

objdump -dS nets_8_nopie.out

```
  *p0 = *p1;
 4a9:   8b 90 ec ff ff ff       mov    -0x14(%eax),%edx
 4af:   8b 0a                   mov    (%edx),%ecx
 4b1:   8b 90 10 00 00 00       mov    0x10(%eax),%edx
 4b7:   8b 09                   mov    (%ecx),%ecx
 4b9:   89 0a                   mov    %ecx,(%edx)
  *p1 = tmp;
 4bb:   8b 80 ec ff ff ff       mov    -0x14(%eax),%eax
 4c1:   8b 00                   mov    (%eax),%eax
 4c3:   8b 55 fc                mov    -0x4(%ebp),%edx
 4c6:   89 10                   mov    %edx,(%eax)
}
 4c8:   90                      nop
 4c9:   c9                      leave
 4ca:   c3                      ret
```

swap in libswap_pic.so(1)

objdump -dS swap_9_pic_nopie.out

```
0000047d <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
 47d:   55                      push   %ebp
 47e:   89 e5                   mov    %esp,%ebp
 480:   83 ec 10                sub    $0x10,%esp
 483:   e8 47 00 00 00          call   4cf <__x86.get_pc_thunk.ax>
 488:   05 78 1b 00 00          add    $0x1b78,%eax
```

swap in libswap_pic.so(2)

objdump -dS swap_9_pic_nopie.out

```
 int tmp;
 p1 = &buf[1];
48d:   8b 90 ec ff ff ff       mov    -0x14(%eax),%edx
493:   8b 88 f8 ff ff ff       mov    -0x8(%eax),%ecx
499:   8d 49 04                lea    0x4(%ecx),%ecx
49c:   89 0a                   mov    %ecx,(%edx)
 tmp = *p0;
49e:   8b 90 e4 ff ff ff       mov    -0x1c(%eax),%edx
4a4:   8b 12                   mov    (%edx),%edx
4a6:   8b 12                   mov    (%edx),%edx
4a8:   89 55 fc                mov    %edx,-0x4(%ebp)
```

swap in libswap_pic.so(3)

objdump -dS swap_9_pic_nopie.out

```
  *p0 = *p1;
 4ab:   8b 90 ec ff ff ff       mov    -0x14(%eax),%edx
 4b1:   8b 0a                   mov    (%edx),%ecx
 4b3:   8b 90 e4 ff ff ff       mov    -0x1c(%eax),%edx
 4b9:   8b 12                   mov    (%edx),%edx
 4bb:   8b 09                   mov    (%ecx),%ecx
 4bd:   89 0a                   mov    %ecx,(%edx)
  *p1 = tmp;
 4bf:   8b 80 ec ff ff ff       mov    -0x14(%eax),%eax
 4c5:   8b 00                   mov    (%eax),%eax
 4c7:   8b 55 fc                mov    -0x4(%ebp),%edx
 4ca:   89 10                   mov    %edx,(%eax)
}
 4cc:   90                      nop
 4cd:   c9                      leave
 4ce:   c3                      ret
```

swap in libswap$_n$opic.so(1)

objdump -dS swap_10_nopic_nopie.out

```
0000049d <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
 49d:   55                      push   %ebp
 49e:   89 e5                   mov    %esp,%ebp
 4a0:   83 ec 10                sub    $0x10,%esp
```

swap in libswap$_n$opic.so(2)

objdump -dS swap_10_nopic_nopie.out

```
 int tmp;
 p1 = &buf[1];
4a3:   c7 05 00 00 00 00 04    movl   $0x4,0x0
4aa:   00 00 00
 tmp = *p0;
4ad:   a1 00 00 00 00          mov    0x0,%eax
4b2:   8b 00                   mov    (%eax),%eax
4b4:   89 45 fc                mov    %eax,-0x4(%ebp)
```

swap in libswap$_n$opic.so(3)

objdump -dS swap_10_nopic_nopie.out

```
  *p0 = *p1;
 4b7:   8b 15 00 00 00 00       mov    0x0,%edx
 4bd:   a1 00 00 00 00          mov    0x0,%eax
 4c2:   8b 12                   mov    (%edx),%edx
 4c4:   89 10                   mov    %edx,(%eax)
  *p1 = tmp;
 4c6:   a1 00 00 00 00          mov    0x0,%eax
 4cb:   8b 55 fc                mov    -0x4(%ebp),%edx
 4ce:   89 10                   mov    %edx,(%eax)
}
 4d0:   90                      nop
 4d1:   c9                      leave
 4d2:   c3                      ret
```

swap in libswap$_n$*oplt*.*so*(1)

objdump -dS swap_11_noplt_nopie.out

```
0000047d <swap>:
extern int buf[];
int *p0 = &buf[0];
int *p1;
void swap()
{
 47d:   55                      push   %ebp
 47e:   89 e5                   mov    %esp,%ebp
 480:   83 ec 10                sub    $0x10,%esp
 483:   e8 43 00 00 00          call   4cb <__x86.get_pc_thunk.ax>
 488:   05 78 1b 00 00          add    $0x1b78,%eax
```

swap in libswap$_n$*oplt*.so(2)

objdump -dS swap_11_noplt_nopie.out

```
 int tmp;
 p1 = &buf[1];
48d:   8b 90 ec ff ff ff       mov    -0x14(%eax),%edx
493:   8b 88 f8 ff ff ff       mov    -0x8(%eax),%ecx
499:   8d 49 04                lea    0x4(%ecx),%ecx
49c:   89 0a                   mov    %ecx,(%edx)
 tmp = *p0;
49e:   8b 90 10 00 00 00       mov    0x10(%eax),%edx
4a4:   8b 12                   mov    (%edx),%edx
4a6:   89 55 fc                mov    %edx,-0x4(%ebp)
```

swap in libswap$_n$*oplt*.so(3)

objdump -dS swap_11_noplt_nopie.out

```
  *p0 = *p1;
 4a9:   8b 90 ec ff ff ff       mov    -0x14(%eax),%edx
 4af:   8b 0a                   mov    (%edx),%ecx
 4b1:   8b 90 10 00 00 00       mov    0x10(%eax),%edx
 4b7:   8b 09                   mov    (%ecx),%ecx
 4b9:   89 0a                   mov    %ecx,(%edx)
  *p1 = tmp;
 4bb:   8b 80 ec ff ff ff       mov    -0x14(%eax),%eax
 4c1:   8b 00                   mov    (%eax),%eax
 4c3:   8b 55 fc                mov    -0x4(%ebp),%edx
 4c6:   89 10                   mov    %edx,(%eax)
}
 4c8:   90                      nop
 4c9:   c9                      leave
 4ca:   c3                      ret
```

case 0: swap in $swap_0.out$

```
objdump -dS swap_0.out
```

case 0: swap in $swap_0.out$

```
objdump -dS swap_0.out
```