

Let's look at some C code and its binary

0804840b <f0 804840b: 804840c: 804840e: 8048411: 8048414: 8048419: 8048421: 8048422: 8048423:</f0 	55 89 e5 83 ec 83 ec 68 d0 e8 c2 83 c4 90 c9	08 0c 84 fe			push mov sub sub push call add nop leave ret	ebp ebp,esp esp,0x8 esp,0xc 0x80484d0 80482e0 <printf@plt> esp,0x10</printf@plt>
08048424 <max 8048424:="" 804842b:="" 804842f:="" 8048431:="" 8048435:="" 8048436:="" 8048443:="" 8048444:="" 8048446:="" 804844c:<="" td=""><td>ain>: 8d 4d 83 e4 ff 71 55 89 e5 51 83 e4 b8 00 83 c4 59 5d 66 90 66 90 66 90</td><td>f0 fc 04 ff 00 04</td><td>ff</td><td>ff 00</td><td>lea and push push mov push sub call mov add pop pop lea ret xchg xchg</td><td>ecx,[esp+0x4] esp,0xfffffff0 DWORD PTR [ecx-0x4] ebp ebp,esp ecx esp,0x4 804840b <foo> eax,0x0 esp,0x4 ecx ebp esp,[ecx-0x4] ax,ax ax,ax ax,ax ax,ax</foo></td></max>	ain>: 8d 4d 83 e4 ff 71 55 89 e5 51 83 e4 b8 00 83 c4 59 5d 66 90 66 90 66 90	f0 fc 04 ff 00 04	ff	ff 00	lea and push push mov push sub call mov add pop pop lea ret xchg xchg	ecx,[esp+0x4] esp,0xfffffff0 DWORD PTR [ecx-0x4] ebp ebp,esp ecx esp,0x4 804840b <foo> eax,0x0 esp,0x4 ecx ebp esp,[ecx-0x4] ax,ax ax,ax ax,ax ax,ax</foo>

```
#include <stdio.h>
int foo(){
    printf("hello world!");
int main(int argc, char **argv){
    foo();
```

program cannot be placed at random locations in memory

Since function addresses and others

are hard-encoded in the binary, the

Let's look at some C code and its binary

```
#include <stdio.h>
int foo(){
    printf("hello world!");
}
int main(int argc, char **argv){
    foo();
}
```

Since function addresses and others are hard-encoded in the binary, the program cannot be placed at random locations in memory

```
0804840b <foo>:
804840b:
             55
                                       push
                                               ebp
804840c:
             89 e5
                                               ebp,esp
804840e:
             83 ec 08
                                               esp,0x8
                                       sub
8048411:
             83 ec 0c
                                               esp,0xc
                                       sub
             68 d0 84 04 08
8048414:
                                               0x80484d0
                                       push
8048419:
             e8 c2 fe ff ff
                                       call
                                               80482e0 <printf@plt>
804841e:
             83 c4 10
                                       add
                                               esp,0x10
8048421:
             90
                                       nop
8048422:
             c9
                                       leave
8048423:
             c3
                                       ret
08048424 <main>:
                                               ecx.[esp+0x4]
8048424:
             8d 4c 24 04
                                       lea
             83 e4 f0
                                               esp,0xfffffff0
8048428:
                                       and
             ff 71 fc
                                               DWORD PTR [ecx-0x4]
804842b:
                                       push
804842e:
             55
                                       push
                                               ebp
804842f:
             89 e5
                                               ebp,esp
                                       mov
8048431:
             51
                                       push
                                               ecx
8048432:
             83 ec 04
                                       sub
                                               esp,0x4
8048435:
             e8 d1 ff ff ff
                                       call
                                               804840b <foo>
804843a:
             b8 00 00 00 00
                                       mov
                                               eax,0x0
804843f:
             83 c4 04
                                       add
                                               esp,0x4
8048442:
             59
                                               ecx
                                       pop
8048443:
             5d
                                       pop
                                               ebp
                                               esp, [ecx-0x4]
8048444:
             8d 61 fc
                                       lea
8048447:
             c3
                                       ret
             66 90
8048448:
                                       xchq
                                               ax,ax
804844a:
             66 90
                                       xchq
                                               ax,ax
             66 90
804844c:
                                       xchq
                                               ax,ax
804844e:
             66 90
                                       xchq
                                               ax,ax
```

Naive Idea: load time linking

How about doing the linking when process executed, not at compile time

- → Determine where process will reside in memory and adjust all references within program
- How to relocate the program in memory during execution? (consider functions but also data pointers now)
- What if no contiguous free region fits program?
- How to avoid programs interfering with each others?