

Link 3A. Static Linking Overview

Young W. Lim

2024-11-15 Fri

1 Static Linking Overview

- Based on
- Creating Static Libraries
- Static Linking Examples
- `gcc -l, -L, and -Wl,option`

"Self-service Linux: Mastering the Art of Problem Determination",

Mark Wilding

"Computer Architecture: A Programmer's Perspective",

Bryant & O'Hallaron

I, the copyright holder of this work, hereby publish it under the following licenses: GNU head Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled GNU Free Documentation License.

CC BY SA This file is licensed under the Creative Commons Attribution ShareAlike 3.0 Unported License. In short: you are free to share and make derivative works of the file under the conditions that you appropriately attribute it, and that you distribute it only under a license compatible with this one.

Compiling 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`
- `-m32`
- `gcc-multilib`
- `g++-multilib`

Example 1 codes

main.c

```
#include <stdio.h>
#include "calc_mean.h"

int main(int argc, char* argv[]) {

    double m, v1 =5.2, v2 =7.9;

    m = mean(v1, v2);

    printf("The mean ");
    printf("of %3.2f ", v1);
    printf("and %3.2f ", v2);
    printf("is %3.2f\n", m);

    return 0;
}
```

calc_mean.h

```
double mean(double, double);
```

calc_mean.c

```
//#include <stdio.h>

double mean(double a, double b) {
    return (a+b) / 2;
}
```

<http://lsi.vc.ehu.eus/pablogn/docencia/ISO/Act5%20Libs/crealibdin.html>

Example 1 creating a static library libmean.a

- a **static library** is basically a set of object files that were copied into a single file
 - first, `calc_mean.c` is turned into an object file:
`gcc -c calc_mean.c -o calc_mean.o`
 - then, the archiver (**ar**) is invoked to produce a **static library** (named `libmean.a`) out of the object file `calc_mean.o`
`ar rcs libmean.a calc_mean.o`
 - a library name must start with **lib** and have the suffix **.a**.

<http://lsi.vc.ehu.eus/pablogn/docencia/ISO/Act5%20Libs/crealibdin.html>

Example 1 creating a shared library libmean.so

- an object file for shared libraries
must be position independent

- use the `-fPIC` option for a position independent code

```
gcc -c -fPIC calc_mean.c -o calc_mean.o
```

- create a shared library with the `-shared` option

```
gcc -shared -o libmean.so calc_mean.o
```

to follow linux naming convention

```
gcc -shared -Wl,-soname,libmean.so.1 -o libmean.so.1.0.1 calc_mean.o
```

- a library name must start with `lib`
and have the suffix `.so`.

<http://lsi.vc.ehu.es/pablogn/docencia/ISO/Act5%20Libs/crealibdin.html>

Example 1 (5-1)

- Linking against static library (`libmean.a`)

```
gcc -static main.c -o s_linked -L. -lmean
```

- the first three letters (the `lib`) must not be specified,
- as well as the suffix (`.a`)

- Executing the statically linked program

```
./s_linked
```

<http://lsi.vc.ehu.es/pablogn/docencia/ISO/Act5%20Libs/crealibdin.html>

Example 1 (5-2)

- Linking against shared library (`libmean.so`)

```
gcc main.c -o d_linked -L. -lmean
```

- e first three letters (the lib) must not be specified,
- as well as the suffix (`.so`)

- Executing the dynamically linked program

```
LD_LIBRARY_PATH=.  
./d_linked
```

<http://lsi.vc.ehu.eus/pablogn/docencia/ISO/Act5%20Libs/crealibdin.html>

Example | the archiver ar

- **ar** : a Linux tool to create, modify, and extract from archives
some options
 - **r** replace files existing inside the archive
 - **c** create an archive if not already present
 - **s** create an object-file index into the archive

<https://stackoverflow.com/questions/2734719/how-to-compile-a-static-library-in-linux>

Example I Position Independent Code (1)

- the shared library is set up so it can have its function entry points dynamically calculated when calling programs call functions in the library. The final function entry point addresses are determined at program runtime, not when the program is compiled. This is what allows the library itself to be shared by multiple programs at the same time.

<https://www.quora.com/What-are-the-advantages-and-disadvantages-of-adding-fPIC-opt>

Example | Position Independent Code (2)

- You need it or `-fpic` to generate position independent code, which is code usable in dynamic libraries without runtime relocations (which are expensive) or code usable in dynamic libraries, period.
- x86 dynamic libs can have position dependent code at the cost of runtime relocations. x86_64 dynamic libs won't even accept code that's not position independent.
- If you intend to build a dynamic lib, your c files should be compiled with `-fpic` (which generates better code than `-fPIC`, but isn't always possible depending on the size of your code), or, if the `-fpic` compilation fails, with `-fPIC`.

<https://www.quora.com/What-are-the-advantages-and-disadvantages-of-adding-fPIC-opt>

Example II (1)

- `gcc -c -o out.o out.c`
- `-c` means to create an intermediary object file, rather than an executable.
- `ar rcs libout.a out.o`
- This creates the static library. `r` means to insert with replacement, `c` means to create a new archive, and `s` means to write an index. As always, see the man page for more info.

<https://stackoverflow.com/questions/2734719/how-to-compile-a-static-library-in-linux>

Example II (2)

```
TARGET = prog
$(TARGET): main.o lib.a gcc $^ -o $@
main.o: main.c gcc -c $< -o $@
lib.a: lib1.o lib2.o ar rcs $@ $^
lib1.o: lib1.c lib1.h gcc -c -o $@ $<
lib2.o: lib2.c lib2.h gcc -c -o $@ $<
clean: rm -f *.o *.a $(TARGET)
```

<https://stackoverflow.com/questions/2734719/how-to-compile-a-static-library-in-linux>

Example II (3)

explaining the makefile:

target: prerequisites - the rule head \$@ - means the target \$^ - means all prerequisites \$< - means just the first prerequisite

<https://stackoverflow.com/questions/2734719/how-to-compile-a-static-library-in-linux>

Example II (5)

main.c

```
#include <stdio.h>

#include "lib.h"

int main ( void )
{
    fun1(10);
    fun2(10);
    return 0;
}
```

lib.h

```
#ifndef LIB_H_INCLUDED
#define LIB_H_INCLUDED

#include "lib1.h"
#include "lib2.h"

#endif
```

<https://stackoverflow.com/questions/2734719/how-to-compile-a-static-library-in-linux>

Example II (6)

lib1.c

```
#include "lib1.h"

#include <stdio.h>

void fun1 ( int x )
{
    printf("%i\n",x);
}
```

lib1.h

```
#ifndef LIB1_H_INCLUDED
#define LIB1_H_INCLUDED

#ifdef __cplusplus
    extern "C" {
#endif

void fun1 ( int x );

#ifdef __cplusplus
    }
#endif

#endif /* LIB1_H_INCLUDED */
```

<https://stackoverflow.com/questions/2734719/how-to-compile-a-static-library-in-linux>

Example II (7)

lib2.c

```
#include "lib2.h"

#include <stdio.h>

void fun2 ( int x )
{
    printf("%i\n",2*x);
}
```

lib2.h

```
#ifndef LIB2_H_INCLUDED
#define LIB2_H_INCLUDED

#ifdef __cplusplus
    extern "C" {
#endif

void fun2 ( int x );

#ifdef __cplusplus
    }
#endif

#endif /* LIB2_H_INCLUDED */
```

<https://stackoverflow.com/questions/2734719/how-to-compile-a-static-library-in-linux>

Static Linking Examples

- 1 Example Program
- 2 Static Linking Procedures
- 3 Static Linking Commands

Example Program

main.c

```
void swap();

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

int main()
{
    int tmp;

    swap();

    return 0;
}
```

swap.c

```
extern int buf[];

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

void swap()
{
    int tmp;

    p1 = &buf[1];

    tmp = *p0;
    *p0 = *p1;
    *p1 = tmp;
}
```

Static Linking Procedures

cpp	preprocessor	main.c	main.i
cc1	compiler	main.i	main.s
as	assemble	main.s	main.o
cpp	preprocessor	swap.c	swap.i
cc1	compiler	swap.i	swap.s
as	assemble	swap.s	swap.o
ld	linker	main.o	p
		swap.o	

Static Linking Commands

```
cpp main.c main.i
cc1 main.i -O2 -m32 -o main.s
as --32 -o main.o main.s
```

```
cpp swap.c swap.i
cc1 swap.i -O2 -m32 -o swap.s
as --32 -o swap.o swap.s
```

```
gcc -o p main.o swap.o
```

```
alias cc1=/usr/lib/gcc/x86_64-linux-gnu/5/cc1
```

```
ld -m elf_i386 -o p main.o swap.o -lc --entry_main --> not working
```

```
readelf -a main.o
readelf -a swap.o
```

Intermediate and Assembly Files (main.i, main.s)

- 1 main.i
- 2 main.s

```
# 1 "main.c"
# 1 "<built-in>"
# 1 "<command-line>"
# 1 "/usr/include/stdc-predef.h" 1 3 4
# 1 "<command-line>" 2
# 1 "main.c"
void swap();

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

int main()
{
    int tmp;

    swap();

    return 0;
}
```


main.s (1)

```
.file      "main.i"
.section   .text.unlikely,"ax",@progbits
.LCOLDB0:
.section   .text.startup,"ax",@progbits
.LHOTB0:
.p2align  4,,15
.globl    main
.type     main, @function

main:
.LFB0:
.cfi_startproc
leal     4(%esp), %ecx
.cfi_def_cfa 1, 0
andl    $-16, %esp
pushl   -4(%ecx)
pushl   %ebp
.cfi_escape 0x10,0x5,0x2,0x75,0
movl    %esp, %ebp
pushl   %ecx
.cfi_escape 0xf,0x3,0x75,0x7c,0x6
```

main.s (2)

```
    subl     $4, %esp
    call    swap
    addl     $4, %esp
    xorl     %eax, %eax
    popl     %ecx
    .cfi_restore 1
    .cfi_def_cfa 1, 0
    popl     %ebp
    .cfi_restore 5
    leal    -4(%ecx), %esp
    .cfi_def_cfa 4, 4
    ret
    .cfi_endproc

.LFE0:
    .size    main, .-main
    .section .text.unlikely

.LCOLDE0:
    .section .text.startup
```

```
.LHOTE0:  
    .globl      buf  
    .data  
    .align 4  
    .type      buf, @object  
    .size      buf, 8  
  
buf:  
    .long      1  
    .long      2  
    .ident     "GCC: (Ubuntu 5.4.0-6ubuntu1~16.04.4) 5.4.0 20160609"  
    .section   .note.GNU-stack,"",@progbits
```

Intermediate and Assembly Files (swap.i, swap.s)

- 1 swap.i
- 2 swap.s

```
# 1 "swap.c"
# 1 "<built-in>"
# 1 "<command-line>"
# 1 "/usr/include/stdc-predef.h" 1 3 4
# 1 "<command-line>" 2
# 1 "swap.c"
extern int buf[];

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

void swap()
{
    int tmp;

    p1 = &buf[1];

    tmp = *p0;
    *p0 = *p1;
    *p1 = tmp;
}
```

swap.s (1)

```
.file "swap.i"
.section .text.unlikely,"ax",@progbits
.LCOLDB0:
.text
.LHOTB0:
.p2align 4,,15
.globl swap
.type swap, @function

swap:
.LFB0:
.cfi_startproc
movl    p0, %eax
movl    buf+4, %ecx
movl    $buf+4, p1
movl    (%eax), %edx
movl    %ecx, (%eax)
movl    %edx, buf+4
ret
.cfi_endproc
.LFE0:
.size   swap, .-swap
.section .text.unlikely
```

```
.LCOLDE0:  
    .text  
.LHOTE0:  
    .comm    p1,4,4  
    .globl  p0  
    .data  
    .align  4  
    .type   p0, @object  
    .size   p0, 4  
  
p0:  
    .long   buf  
    .ident  "GCC: (Ubuntu 5.4.0-6ubuntu1~16.04.4) 5.4.0 20160609"  
    .section      .note.GNU-stack,"",@progbits
```

Relocatable Object Files (main.o)

- 1 main.o ELF Header
- 2 main.o Section Header
- 3 main.o Section
- 4 main.o Symbol Table

main.o (1) ELF Header

ELF Header:

```
Magic: 7f 45 4c 46 01 01 01 00 00 00 00 00 00 00 00 00
Class: ELF32
Data: 2's complement, little endian
Version: 1 (current)
OS/ABI: UNIX - System V
ABI Version: 0
Type: REL (Relocatable file)
Machine: Intel 80386
Version: 0x1
Entry point address: 0x0
Start of program headers: 0 (bytes into file)
Start of section headers: 584 (bytes into file)
Flags: 0x0
Size of this header: 52 (bytes)
Size of program headers: 0 (bytes)
Number of program headers: 0
Size of section headers: 40 (bytes)
Number of section headers: 14
Section header string table index: 11
```

main.o (2) Section Header

Section Headers:

[Nr]	Name	Type	Addr	Off	Size	ES	Flg	Lk	Inf	Al
[0]		NULL	00000000	000000	000000	00		0	0	0
[1]	.text	PROGBITS	00000000	000034	000000	00	AX	0	0	1
[2]	.data	PROGBITS	00000000	000034	000008	00	WA	0	0	4
[3]	.bss	NOBITS	00000000	00003c	000000	00	WA	0	0	1
[4]	.text.unlikely	PROGBITS	00000000	00003c	000000	00	AX	0	0	1
[5]	.text.startup	PROGBITS	00000000	000040	000021	00	AX	0	0	16
[6]	.rel.text.startup	REL	00000000	0001c4	000008	08	I	12	5	4
[7]	.comment	PROGBITS	00000000	000061	000035	01	MS	0	0	1
[8]	.note.GNU-stack	PROGBITS	00000000	000096	000000	00		0	0	1
[9]	.eh_frame	PROGBITS	00000000	000098	000044	00	A	0	0	4
[10]	.rel.eh_frame	REL	00000000	0001cc	000008	08	I	12	9	4
[11]	.shstrtab	STRTAB	00000000	0001d4	000074	00		0	0	1
[12]	.symtab	SYMTAB	00000000	0000dc	0000d0	10		13	10	4
[13]	.strtab	STRTAB	00000000	0001ac	000016	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)

0 (extra OS processing required) o (OS specific), p (processor specific)

main.o (3) Sections

There are no section groups in this file.

There are no program headers in this file.

Relocation section '.rel.text.startup' at offset 0x1c4 contains 1 entries:

Offset	Info	Type	Sym.Value	Sym. Name
00000012	00000b02	R_386_PC32	00000000	swap

Relocation section '.rel.eh_frame' at offset 0x1cc contains 1 entries:

Offset	Info	Type	Sym.Value	Sym. Name
00000020	00000602	R_386_PC32	00000000	.text.startup

The decoding of unwind sections for machine type Intel 80386 is not currently supported.

main.o (4) Symbol Table

Symbol table '.symtab' contains 13 entries:

Num:	Value	Size	Type	Bind	Vis	Ndx	Name
0:	00000000	0	NOTYPE	LOCAL	DEFAULT	UND	
1:	00000000	0	FILE	LOCAL	DEFAULT	ABS	main.i
2:	00000000	0	SECTION	LOCAL	DEFAULT	1	
3:	00000000	0	SECTION	LOCAL	DEFAULT	2	
4:	00000000	0	SECTION	LOCAL	DEFAULT	3	
5:	00000000	0	SECTION	LOCAL	DEFAULT	4	
6:	00000000	0	SECTION	LOCAL	DEFAULT	5	
7:	00000000	0	SECTION	LOCAL	DEFAULT	8	
8:	00000000	0	SECTION	LOCAL	DEFAULT	9	
9:	00000000	0	SECTION	LOCAL	DEFAULT	7	
10:	00000000	33	FUNC	GLOBAL	DEFAULT	5	main
11:	00000000	0	NOTYPE	GLOBAL	DEFAULT	UND	swap
12:	00000000	8	OBJECT	GLOBAL	DEFAULT	2	buf

No version information found in this file.

Relocatable Object Files (swap.o)

- 1 swap.o ELF Header
- 2 swap.o Section Header
- 3 swap.o Section
- 4 swap.o Symbol Table

swap.o (1) ELF Header

ELF Header:

```
Magic: 7f 45 4c 46 01 01 01 00 00 00 00 00 00 00 00 00
Class: ELF32
Data: 2's complement, little endian
Version: 1 (current)
OS/ABI: UNIX - System V
ABI Version: 0
Type: REL (Relocatable file)
Machine: Intel 80386
Version: 0x1
Entry point address: 0x0
Start of program headers: 0 (bytes into file)
Start of section headers: 596 (bytes into file)
Flags: 0x0
Size of this header: 52 (bytes)
Size of program headers: 0 (bytes)
Number of program headers: 0
Size of section headers: 40 (bytes)
Number of section headers: 14
Section header string table index: 11
```

swap.o (2) Section Header

Section Headers:

[Nr]	Name	Type	Addr	Off	Size	ES	Flg	Lk	Inf	Al
[0]		NULL	00000000	000000	000000	00		0	0	0
[1]	.text	PROGBITS	00000000	000040	000020	00	AX	0	0	16
[2]	.rel.text	REL	00000000	0001b0	000028	08	I 12		1	4
[3]	.data	PROGBITS	00000000	000060	000004	00	WA	0	0	4
[4]	.rel.data	REL	00000000	0001d8	000008	08	I 12		3	4
[5]	.bss	NOBITS	00000000	000064	000000	00	WA	0	0	1
[6]	.text.unlikely	PROGBITS	00000000	000064	000000	00	AX	0	0	1
[7]	.comment	PROGBITS	00000000	000064	000035	01	MS	0	0	1
[8]	.note.GNU-stack	PROGBITS	00000000	000099	000000	00		0	0	1
[9]	.eh_frame	PROGBITS	00000000	00009c	00002c	00	A	0	0	4
[10]	.rel.eh_frame	REL	00000000	0001e0	000008	08	I 12		9	4
[11]	.shstrtab	STRTAB	00000000	0001e8	00006a	00		0	0	1
[12]	.symtab	SYMTAB	00000000	0000c8	0000d0	10		13	9	4
[13]	.strtab	STRTAB	00000000	000198	000017	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)

0 (extra OS processing required) o (OS specific), p (processor specific)

swap.o (3) Sections

There are no section groups in this file.

There are no program headers in this file.

Relocation section '.rel.text' at offset 0x1b0 contains 5 entries:

Offset	Info	Type	Sym.Value	Sym. Name
00000001	00000a01	R_386_32	00000000	p0
00000007	00000b01	R_386_32	00000000	buf
0000000d	00000c01	R_386_32	00000004	p1
00000011	00000b01	R_386_32	00000000	buf
0000001b	00000b01	R_386_32	00000000	buf

Relocation section '.rel.data' at offset 0x1d8 contains 1 entries:

Offset	Info	Type	Sym.Value	Sym. Name
00000000	00000b01	R_386_32	00000000	buf

Relocation section '.rel.eh_frame' at offset 0x1e0 contains 1 entries:

Offset	Info	Type	Sym.Value	Sym. Name
00000020	00000202	R_386_PC32	00000000	.text

The decoding of unwind sections for machine type Intel 80386 is not currently supported.

swap.o (4) Symbol Table

Symbol table '.symtab' contains 13 entries:

Num:	Value	Size	Type	Bind	Vis	Ndx	Name
0:	00000000	0	NOTYPE	LOCAL	DEFAULT	UND	
1:	00000000	0	FILE	LOCAL	DEFAULT	ABS	swap.i
2:	00000000	0	SECTION	LOCAL	DEFAULT	1	
3:	00000000	0	SECTION	LOCAL	DEFAULT	3	
4:	00000000	0	SECTION	LOCAL	DEFAULT	5	
5:	00000000	0	SECTION	LOCAL	DEFAULT	6	
6:	00000000	0	SECTION	LOCAL	DEFAULT	8	
7:	00000000	0	SECTION	LOCAL	DEFAULT	9	
8:	00000000	0	SECTION	LOCAL	DEFAULT	7	
9:	00000000	32	FUNC	GLOBAL	DEFAULT	1	swap
10:	00000000	4	OBJECT	GLOBAL	DEFAULT	3	p0
11:	00000000	0	NOTYPE	GLOBAL	DEFAULT	UND	buf
12:	00000004	4	OBJECT	GLOBAL	DEFAULT	COM	p1

No version information found in this file.

Executable Object Files (p)

- 1 p ELF Header
- 2 p Section Header
- 3 p Program Header
- 4 p Section to Segment Mapping
- 5 p Dynamic Sections
- 6 p Misc Sections
- 7 p Symbol Table
- 8 p Others

p (1) ELF Header

ELF Header:

```
Magic: 7f 45 4c 46 01 01 01 00 00 00 00 00 00 00 00 00
Class: ELF32
Data: 2's complement, little endian
Version: 1 (current)
OS/ABI: UNIX - System V
ABI Version: 0
Type: EXEC (Executable file)
Machine: Intel 80386
Version: 0x1
Entry point address: 0x8048301
Start of program headers: 52 (bytes into file)
Start of section headers: 6180 (bytes into file)
Flags: 0x0
Size of this header: 52 (bytes)
Size of program headers: 32 (bytes)
Number of program headers: 9
Size of section headers: 40 (bytes)
Number of section headers: 31
Section header string table index: 28
```

p (2) Section Header (a)

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	08048154	000154	000013	00	A	0	0	1
[2]	.note.ABI-tag	NOTE	08048168	000168	000020	00	A	0	0	4
[3]	.note.gnu.build-id	NOTE	08048188	000188	000024	00	A	0	0	4
[4]	.gnu.hash	GNU_HASH	080481ac	0001ac	000020	04	A	5	0	4
[5]	.dynsym	DYNSYM	080481cc	0001cc	000040	10	A	6	1	4
[6]	.dynstr	STRTAB	0804820c	00020c	000045	00	A	0	0	1
[7]	.gnu.version	VERSYM	08048252	000252	000008	02	A	5	0	2
[8]	.gnu.version_r	VERNEED	0804825c	00025c	000020	00	A	6	1	4
[9]	.rel.dyn	REL	0804827c	00027c	000008	08	A	5	0	4
[10]	.rel.plt	REL	08048284	000284	000008	08	AI	5	24	4
[11]	.init	PROGBITS	0804828c	00028c	000023	00	AX	0	0	4
[12]	.plt	PROGBITS	080482b0	0002b0	000020	04	AX	0	0	16
[13]	.plt.got	PROGBITS	080482d0	0002d0	000008	00	AX	0	0	8
[14]	.text	PROGBITS	080482e0	0002e0	0001a2	00	AX	0	0	16
[15]	.fini	PROGBITS	08048484	000484	000014	00	AX	0	0	4
[16]	.rodata	PROGBITS	08048498	000498	000008	00	A	0	0	4
[17]	.eh_frame_hdr	PROGBITS	080484a0	0004a0	000034	00	A	0	0	4
[18]	.eh_frame	PROGBITS	080484d4	0004d4	0000e0	00	A	0	0	4
[19]	.init_array	INIT_ARRAY	08049f08	000f08	000004	00	WA	0	0	4
[20]	.fini_array	FINI_ARRAY	08049f0c	000f0c	000004	00	WA	0	0	4

p (3) Section Header (b)

[21]	.jcr	PROGBITS	08049f10	000f10	000004	00	WA	0	0	4
[22]	.dynamic	DYNAMIC	08049f14	000f14	0000e8	08	WA	6	0	4
[23]	.got	PROGBITS	08049ffc	000ffc	000004	04	WA	0	0	4
[24]	.got.plt	PROGBITS	0804a000	001000	000010	04	WA	0	0	4
[25]	.data	PROGBITS	0804a010	001010	000014	00	WA	0	0	4
[26]	.bss	NOBITS	0804a024	001024	000008	00	WA	0	0	4
[27]	.comment	PROGBITS	00000000	001024	000034	01	MS	0	0	1
[28]	.shstrtab	STRTAB	00000000	00171a	00010a	00		0	0	1
[29]	.symtab	SYMTAB	00000000	001058	000490	10		30	48	4
[30]	.strtab	STRTAB	00000000	0014e8	000232	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)

0 (extra OS processing required) o (OS specific), p (processor specific)

p (4) Program Header

Program Headers:

Type	Offset	VirtAddr	PhysAddr	FileSiz	MemSiz	Flg	Align
PHDR	0x000034	0x08048034	0x08048034	0x00120	0x00120	R E	0x4
INTERP	0x000154	0x08048154	0x08048154	0x00013	0x00013	R	0x1
[Requesting program interpreter: /lib/ld-linux.so.2]							
LOAD	0x000000	0x08048000	0x08048000	0x005b4	0x005b4	R E	0x1000
LOAD	0x000f08	0x08049f08	0x08049f08	0x0011c	0x00124	RW	0x1000
DYNAMIC	0x000f14	0x08049f14	0x08049f14	0x000e8	0x000e8	RW	0x4
NOTE	0x000168	0x08048168	0x08048168	0x00044	0x00044	R	0x4
GNU_EH_FRAME	0x0004a0	0x080484a0	0x080484a0	0x00034	0x00034	R	0x4
GNU_STACK	0x000000	0x00000000	0x00000000	0x00000	0x00000	RW	0x10
GNU_RELRO	0x000f08	0x08049f08	0x08049f08	0x000f8	0x000f8	R	0x1

p (5) Section to Segment Mapping

Section to Segment mapping:

Segment Sections...

00

01 .interp

02 .interp .note.ABI-tag .note.gnu.build-id .gnu.hash .dynsym .dynstr .gnu.v

03 .init_array .fini_array .jcr .dynamic .got .got.plt .data .bss

04 .dynamic

05 .note.ABI-tag .note.gnu.build-id

06 .eh_frame_hdr

07

08 .init_array .fini_array .jcr .dynamic .got

p (6) Dynamic Sections (a)

Dynamic section at offset 0xf14 contains 24 entries:

Tag	Type	Name/Value
0x00000001	(NEEDED)	Shared library: [libc.so.6]
0x0000000c	(INIT)	0x804828c
0x0000000d	(FINI)	0x8048484
0x00000019	(INIT_ARRAY)	0x8049f08
0x0000001b	(INIT_ARRAYSZ)	4 (bytes)
0x0000001a	(FINI_ARRAY)	0x8049f0c
0x0000001c	(FINI_ARRAYSZ)	4 (bytes)
0x6ffffef5	(GNU_HASH)	0x80481ac
0x00000005	(STRTAB)	0x804820c
0x00000006	(SYMTAB)	0x80481cc
0x0000000a	(STRSZ)	69 (bytes)
0x0000000b	(SYMENT)	16 (bytes)

p (7) Dynamic Sections (b)

Dynamic section at offset 0xf14 contains 24 entries:

Tag	Type	Name/Value
0x00000003	(PLTGOT)	0x804a000
0x00000002	(PLTRELSZ)	8 (bytes)
0x00000014	(PLTREL)	REL
0x00000017	(JMPREL)	0x8048284
0x00000011	(REL)	0x804827c
0x00000012	(RELSZ)	8 (bytes)
0x00000013	(RELENT)	8 (bytes)
0x6fffffff	(VERNEED)	0x804825c
0x6fffffff	(VERNEEDNUM)	1
0x6fffffff0	(VERSYM)	0x8048252
0x00000000	(NULL)	0x0

p (8) Misc Sections

Relocation section '.rel.dyn' at offset 0x27c contains 1 entries:

Offset	Info	Type	Sym.Value	Sym. Name
08049ffc	00000106	R_386_GLOB_DAT	00000000	__gmon_start__

Relocation section '.rel.plt' at offset 0x284 contains 1 entries:

Offset	Info	Type	Sym.Value	Sym. Name
0804a00c	00000207	R_386_JUMP_SLOT	00000000	__libc_start_main@GLIBC_2.0

The decoding of unwind sections for machine type Intel 80386 is not currently supported.

Symbol table '.dynsym' contains 4 entries:

Num:	Value	Size	Type	Bind	Vis	Ndx	Name
0:	00000000	0	NOTYPE	LOCAL	DEFAULT	UND	
1:	00000000	0	NOTYPE	WEAK	DEFAULT	UND	__gmon_start__
2:	00000000	0	FUNC	GLOBAL	DEFAULT	UND	__libc_start_main@GLIBC_2.0 (2)
3:	0804849c	4	OBJECT	GLOBAL	DEFAULT	16	_IO_stdin_used

p (9) Symbol Table (a)

Symbol table '.symtab' contains 73 entries:

Num:	Value	Size	Type	Bind	Vis	Ndx	Name
0:	00000000	0	NOTYPE	LOCAL	DEFAULT	UND	
1:	08048154	0	SECTION	LOCAL	DEFAULT	1	
2:	08048168	0	SECTION	LOCAL	DEFAULT	2	
3:	08048188	0	SECTION	LOCAL	DEFAULT	3	
4:	080481ac	0	SECTION	LOCAL	DEFAULT	4	
5:	080481cc	0	SECTION	LOCAL	DEFAULT	5	
6:	0804820c	0	SECTION	LOCAL	DEFAULT	6	
7:	08048252	0	SECTION	LOCAL	DEFAULT	7	
8:	0804825c	0	SECTION	LOCAL	DEFAULT	8	
9:	0804827c	0	SECTION	LOCAL	DEFAULT	9	
10:	08048284	0	SECTION	LOCAL	DEFAULT	10	
11:	0804828c	0	SECTION	LOCAL	DEFAULT	11	
12:	080482b0	0	SECTION	LOCAL	DEFAULT	12	
13:	080482d0	0	SECTION	LOCAL	DEFAULT	13	
14:	080482e0	0	SECTION	LOCAL	DEFAULT	14	
15:	08048484	0	SECTION	LOCAL	DEFAULT	15	
16:	08048498	0	SECTION	LOCAL	DEFAULT	16	
17:	080484a0	0	SECTION	LOCAL	DEFAULT	17	
18:	080484d4	0	SECTION	LOCAL	DEFAULT	18	
19:	08049f08	0	SECTION	LOCAL	DEFAULT	19	
20:	08049f0c	0	SECTION	LOCAL	DEFAULT	20	

p (10) Symbol Table (b)

21:	08049f10	0	SECTION	LOCAL	DEFAULT	21	
22:	08049f14	0	SECTION	LOCAL	DEFAULT	22	
23:	08049ffc	0	SECTION	LOCAL	DEFAULT	23	
24:	0804a000	0	SECTION	LOCAL	DEFAULT	24	
25:	0804a010	0	SECTION	LOCAL	DEFAULT	25	
26:	0804a024	0	SECTION	LOCAL	DEFAULT	26	
27:	00000000	0	SECTION	LOCAL	DEFAULT	27	
28:	00000000	0	FILE		DEFAULT	ABS	crtstuff.c
29:	08049f10	0	OBJECT	LOCAL	DEFAULT	21	__JCR_LIST__
30:	08048340	0	FUNC	LOCAL	DEFAULT	14	deregister_tm_clones
31:	08048370	0	FUNC	LOCAL	DEFAULT	14	register_tm_clones
32:	080483b0	0	FUNC	LOCAL	DEFAULT	14	__do_global_dtors_aux
33:	0804a024	1	OBJECT	LOCAL	DEFAULT	26	completed.7200
34:	08049f0c	0	OBJECT	LOCAL	DEFAULT	20	__do_global_dtors_aux_fin
35:	080483d0	0	FUNC	LOCAL	DEFAULT	14	frame_dummy
36:	08049f08	0	OBJECT	LOCAL	DEFAULT	19	__frame_dummy_init_array_
37:	00000000	0	FILE		DEFAULT	ABS	main.i
38:	00000000	0	FILE		DEFAULT	ABS	swap.i
39:	00000000	0	FILE		DEFAULT	ABS	crtstuff.c
40:	080485b0	0	OBJECT	LOCAL	DEFAULT	18	__FRAME_END__

p (11) Symbol Table (c)

50:	08048330	4	FUNC	GLOBAL	HIDDEN	14	__x86.get_pc_thunk.bx
51:	0804a010	0	NOTYPE	WEAK	DEFAULT	25	data_start
52:	0804a024	0	NOTYPE	GLOBAL	DEFAULT	25	_edata
53:	0804a020	4	OBJECT	GLOBAL	DEFAULT	25	p0
54:	08048484	0	FUNC	GLOBAL	DEFAULT	15	_fini
55:	0804a010	0	NOTYPE	GLOBAL	DEFAULT	25	__data_start
56:	00000000	0	NOTYPE	WEAK	DEFAULT	UND	__gmon_start__
57:	0804a014	0	OBJECT	GLOBAL	HIDDEN	25	__dso_handle
58:	0804849c	4	OBJECT	GLOBAL	DEFAULT	16	_IO_stdin_used
59:	00000000	0	FUNC	GLOBAL	DEFAULT	UND	__libc_start_main@@GLIBC_
60:	08048420	93	FUNC	GLOBAL	DEFAULT	14	__libc_csu_init
61:	0804a02c	0	NOTYPE	GLOBAL	DEFAULT	26	_end
62:	08048301	0	FUNC	GLOBAL	DEFAULT	14	_start
63:	08048498	4	OBJECT	GLOBAL	DEFAULT	16	_fp_hw
64:	0804a018	8	OBJECT	GLOBAL	DEFAULT	25	buf
65:	0804a024	0	NOTYPE	GLOBAL	DEFAULT	26	__bss_start
66:	080482e0	33	FUNC	GLOBAL	DEFAULT	14	main
67:	0804a028	4	OBJECT	GLOBAL	DEFAULT	26	p1
68:	00000000	0	NOTYPE	WEAK	DEFAULT	UND	_Jv_RegisterClasses
69:	0804a024	0	OBJECT	GLOBAL	HIDDEN	25	__TMC_END__
70:	00000000	0	NOTYPE	WEAK	DEFAULT	UND	_ITM_registerTMCloneTable
71:	08048400	32	FUNC	GLOBAL	DEFAULT	14	swap
72:	0804828c	0	FUNC	GLOBAL	DEFAULT	11	_init

p (12) Others

Histogram for '.gnu.hash' bucket list length (total of 2 buckets):

Length	Number	% of total	Coverage
0	1	(50.0%)	
1	1	(50.0%)	100.0%

Version symbols section '.gnu.version' contains 4 entries:

```
Addr: 0000000008048252  Offset: 0x000252  Link: 5 (.dynsym)
000:  0 (*local*)          0 (*local*)          2 (GLIBC_2.0)        1 (*global*)
```

Version needs section '.gnu.version_r' contains 1 entries:

```
Addr: 0x000000000804825c  Offset: 0x00025c  Link: 6 (.dynstr)
000000:  Version: 1  File: libc.so.6  Cnt: 1
0x0010:  Name: GLIBC_2.0  Flags: none  Version: 2
```

Displaying notes found at file offset 0x00000168 with length 0x00000020:

Owner	Data size	Description
GNU	0x00000010	NT_GNU_ABI_TAG (ABI version tag)
OS: Linux, ABI: 2.6.32		

Displaying notes found at file offset 0x00000188 with length 0x00000024:

Owner	Data size	Description
GNU	0x00000014	NT_GNU_BUILD_ID (unique build ID bitstring)
Build ID: 1574c15c3270ef63673147a7608a8d225665af50		

gcc -llibrary (1)

- `-llibrary`
`-l library`
 - Search the library named `library` when linking
 - the second alternative with the library `-l library` as a separate argument is only for POSIX compliance and is not recommended
 - The `-l` option is passed directly to the linker by GCC.
 - the linker searches a standard list of directories for the library `library`
 - the directories searched include several standard system directories plus any that you specify with `-L`

man gcc

gcc -llibrary (2)

- `-llibrary`
`-l library`
 - static libraries are archives of object files, and have file names like `liblibrary.a`
 - some targets also support shared libraries, which typically have names like `liblibrary.so`
 - if both static and shared libraries are found, the linker gives preference to linking with the shared library unless the `-static` option is used.

`man gcc`

gcc -llibrary (3)

- `-llibrary`
 - `-l library`
 - it makes a difference where in the command you write this option;
 - the linker searches and processes libraries and object files in the order they are specified.
 - Thus, `foo.o -lz bar.o` searches library `z` after file `foo.o` but before `bar.o`
 - if `bar.o` refers to functions in library `z`, those functions may not be loaded.

`man gcc`

- `-Ldir`
 - Add directory `dir` to the list of directories to be searched for `-l`.
- `-Lsearchdir`
 - `--library-path=searchdir`
 - add path `searchdir` to the list of paths that `ld` will search for archive libraries and `ld` control scripts.
 - you may use this option any number of times.
 - The directories are searched in the order in which they are specified on the command line.
 - Directories specified on the command line are searched before the default directories.

http://ftp.gnu.org/old-gnu/Manuals/ld-2.9.1/html_node/ld_3.html

- -Ldir
- -Lsearchdir
 - library-path=searchdir
 - All -L options apply to all -l options, regardless of the order in which the options appear.
 - the default set of paths searched (without using -L) depends on which emulation mode ld is using, and in some cases also on how it was configured.
 - The paths can also be specified in a link script with the SEARCH_DIR command.
 - Directories specified this way are searched at the point in which the linker script appears in the command line.

http://ftp.gnu.org/old-gnu/Manuals/ld-2.9.1/html_node/ld_3.html

- -Wl,/option/
 - pass option as an *option* to the linker.
 - if option contains *commas*, it is split into multiple options at the *commas*
 - use the *comma* (,) syntax to *pass* an argument to the option.
-Wl,-Map,output.map
passes -Map output.map to the linker.
 - When using the GNU linker, you can also get the same effect with =
-Wl,-Map=output.map

man gcc