ELF1 1E Symbol Table Section

Young W. Lim

2022-07-22 Fri

Outline

Based on

"Study of ELF loading and relocs", 1999
`http://netwinder.osuosl.org/users/p/patb/public_html/elf_relocs.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`

TOC: Symbol table section

TOC: Symbol table section

Symbol table (1)

- An object file's symbol table holds information needed to *locate* and *relocate* a program's symbolic definitions and references
- A symbol table index is a subscript into this array.
- Index 0 both designates the first entry in the table and serves as the undefined symbol index

https://docs.oracle.com/cd/E19683-01/816-1386/6m7qcoblh/index.html

Symbol table (2)

- the first byte is index zero,
  holds a null character (\0)
- the last byte holds a null character (\0)
  ensuring null termination for all strings.
- A string with zero index specifies
  either no name or a null name,
  depending on the context.

`https://docs.oracle.com/cd/E19683-01/816-1386/6m7qcoblh/index.html`

Elf32_Sym structure type

```
typedef struct {
    Elf32_Word    st_name;
    Elf32_Addr    st_value;
    Elf32_Word    st_size;
    unsigned char st_info;
    unsigned char st_other;
    Elf32_Half    st_shndx;
} Elf32_Sym;
```

- st_name :
  An index into the object file's symbol string table

https://docs.oracle.com/cd/E19683-01/816-1386/6m7qcoblh/index.html

Elf32_Sym field types (1) st_name, st_value

- st_name
  - an index into the object file's symbol string table, which holds the character representations of the symbol names.
  - if the value is nonzero, the value represents a string table index that gives the symbol name.
  - otherwise, the symbol table entry has no name.
- st_value
  - the value of the associated symbol.
  - the value can be an absolute value or an address, depending on the context. See Symbol Values.

https://docs.oracle.com/cd/E19683-01/816-1386/6m7qcoblh/index.html

Elf32_Sym fields type (2) st_size, st_info

- st_size
  - Many symbols have associated sizes.
  - For example, a data object's size is the number of bytes that are contained in the object.
  - This member holds the value zero
    if the symbol has no size or an unknown size.
- st_info
  - The symbol's type and binding attributes.
  - A list of the values and meanings appears in Table

https://docs.oracle.com/cd/E19683-01/816-1386/6m7qcoblh/index.html

Elf32_Sym fiedls type (3) st_shndx

- st_shndx
  - every symbol table entry is defined in relation to some section
  - st_shndx member holds the relevant section header table index
- Some section indexes indicate special meanings
  - If this member contains SHN_XINDEX,
    then the actual section header index is
    too large to fit in this member.
  - The actual value is contained in the associated section
    of type SHT_SYMTAB_SHNDX

https://docs.oracle.com/cd/E19683-01/816-1386/6m7qcoblh/index.html

Elf32_Sym fiedls type (4) st_other

- st_other
  - A symbol's visibility
  - Other bits are set to zero, and have no defined meaning.
- symbol binding

STB_LOCAL	0
STB_GLOBAL	1
STB_WEAK	2
STB_LOOS	10
STB_HIOS	12
STB_LOPROC	13
STB_HIPROC	15

https://docs.oracle.com/cd/E19683-01/816-1386/6m7qcoblh/index.html

ELF Symbol binding (1)

- STB_LOCAL: Local symbol.
  - These symbols are not visible outside the object file containing their definition.
  - Local symbols of the same name can exist in multiple files without interfering with each other.
- STB_GLOBAL: Global symbols.
  - These symbols are visible to all object files being combined.
  - One file's definition of a global symbol satisfies another file's undefined reference to the same global symbol.
- STB_WEAK: Weak symbols.
  - These symbols resemble global symbols, but their definitions have lower precedence.

https://docs.oracle.com/cd/E19683-01/816-1386/6m7qcoblh/index.html

ELF Symbol binding (2)

- STB_LOOS - STB_HIOS
  - Values in this inclusive range are reserved for operating system-specific semantics.
- STB_LOPROC - STB_HIPROC
  - Values in this inclusive range are reserved for processor-specific semantics.

https://docs.oracle.com/cd/E19683-01/816-1386/6m7qcoblh/index.html

Global and weak symbols (1)

- When the link-editor combines several relocatable object files, it does not allow *multiple definitions* of `STB_GLOBAL` symbols with the same name.
- On the other hand, if a defined global symbol exists, the appearance of a weak symbol with the same name will not cause an error
- The link-editor honors the global definition and ignores the weak ones.

https://docs.oracle.com/cd/E19683-01/816-1386/6m7qcoblh/index.html

Global and weak symbols (2)

- Similarly, if a common symbol exists,
  the appearance of a weak symbol with the same name
  does not cause an error
- The link-editor uses the common definition
  and ignores the weak one.
- A common symbol has the `st_shndx` field holding `SHN_COMMON`

https://docs.oracle.com/cd/E19683-01/816-1386/6m7qcoblh/index.html

Global and weak symbols (3)

- When the link-editor searches archive libraries it extracts archive members that contain definitions of undefined or tentative global symbols.
- The member's definition can be either a global or a weak symbol.
- The link-editor, by default, does not extract archive members to resolve undefined weak symbols.
- Unresolved weak symbols have a zero value.
- The use of `-z weakextract` overrides this default behavior.
- It enables weak references to cause the extraction of archive members.

`https://docs.oracle.com/cd/E19683-01/816-1386/6m7qcoblh/index.html`

Section of type SHT_SYMTAB, SHT_DYNSYM (1)

sh_type = SHT_SYMTAB, SHT_DYNSYM

- identifies a symbol table
- typically a SHT_SYMTAB section
  provides symbols for link-editing
- as a complete symbol table, it can contain
  many symbols unnecessary for dynamic linking
- Consequently, an object file can also contain
  a SHT_DYNSYM section, which holds
  a minimal set of dynamic linking symbols,
  to save space

https://docs.oracle.com/cd/E19683-01/816-1386/6m7qcoblh/index.html

Section of type SHT_SYMTAB, SHT_DYNSYM (2)

sh_type = SHT_SYMTAB, SHT_DYNSYM

- sh_link
  - The section header index of the associated string table
- sh_info
  - One greater than the symbol table index of the last local symbol (binding STB_LOCAL) .

https://docs.oracle.com/cd/E19683-01/816-1386/6m7qcoblh/index.html

the section header index of the associated symbol table

`sh_type` =

- SHT_HASH
- SHT_REL, SHT_RELA
- SHT_GROUP
- in these sections, `sh_link` represents
  the section header index of the associated symbol table

https://docs.oracle.com/cd/E19683-01/816-1386/6m7qcoblh/index.html