NAME

elf_memory - process an ELF or ar(1) archive mapped into memory

LIBRARY

ELF Access Library (libelf, -lelf)

SYNOPSIS

#include <libelf.h>

*Elf **
elf_memory(*char *image, size_t size*);

DESCRIPTION

Function **elf_memory**() is used to process an ELF file or ar(1) archive whose image is present in memory.

Argument *image* points to the start of the memory image of the file or archive. Argument *size* contains the size in bytes of the memory image.

The ELF descriptor is created for reading (i.e., analogous to the use of elf_begin(3) with a command argument value of ELF_C_READ).

RETURN VALUES

Function **elf_memory**() returns a pointer to a new ELF descriptor if successful, or NULL if an error occurred.

The return value may be queried for the file type using elf_kind(3).

EXAMPLES

To read parse an elf file, use:

```
int fd;
void *p;
struct stat sb;
Elf *e;
...
if ((fd = open("./elf-file", O_RDONLY)) < 0 ||
    fstat(fd, &sb) < 0 ||
    (p = mmap(NULL, sb.st_size, PROT_READ, MAP_PRIVATE, fd, (off_t) 0)) ==
    MAP_FAILED) {
```

```
        ... handle system error ...
}

if ((e = elf_memory(p, sb.st_size)) == NULL) {
        ... handle elf(3) error ...
}
... use ELF descriptor "e" here ...
```

ERRORS

Function **elf_memory**() can fail with the following errors:

[ELF_E_ARGUMENT] A NULL value was used for argument *image* or the value of argument *sz* was zero.

[ELF_E_HEADER] The header of the ELF object contained an unsupported value in its *e_ident[EI_CLASS]* field.

[ELF_E_HEADER] The header of the ELF object contained an unsupported value in its *e_ident[EI_DATA]* field.

[ELF_E_RESOURCE] An out of memory condition was detected.

[ELF_E_SEQUENCE] Function **elf_memory**() was called before a working version was set using elf_version(3).

[ELF_E_VERSION] The ELF object referenced by argument *image* was of an unsupported ELF version.

SEE ALSO

elf(3), elf_begin(3), elf_end(3), elf_errno(3), elf_kind(3), gelf(3)