

NAME

vslock, **vsunlock** - lock/unlock user space addresses in memory

SYNOPSIS

```
#include <sys/param.h>
#include <sys/proc.h>
#include <vm/vm.h>
#include <vm/vm_extern.h>
```

int

```
vslock(void *addr, size_t len);
```

void

```
vsunlock(void *addr, size_t len);
```

DESCRIPTION

The **vslock**() and **vsunlock**() functions respectively lock and unlock a range of addresses belonging to the currently running process into memory. The actual amount of memory locked is a multiple of the machine's page size. The starting page number is computed by truncating *addr* to the nearest preceding page boundary, and by rounding up *addr + len* to the next page boundary. The process context to use for this operation is taken from the global variable *curproc*.

RETURN VALUES

The **vslock**() function will return 0 on success, otherwise it will return one of the errors listed below.

ERRORS

The **vslock**() function will fail if:

- | | |
|----------|--|
| [EINVAL] | The <i>addr</i> and <i>len</i> parameters specify a memory range that wraps around the end of the machine address space. |
| [ENOMEM] | The size of the specified address range exceeds the system limit on locked memory. |
| [EFAULT] | Some portion of the indicated address range is not allocated. There was an error faulting/mapping a page. |