NAME

swapctl – manage swap space

SYNOPSIS

#include <sys/stat.h>
#include <sys/swap.h>

int swapctl(int *cmd*, **void** ∗*arg***);**

DESCRIPTION

swapctl( ) adds, deletes, or returns information about swap resources. *cmd* specifies one of the following options contained in **<sys/swap.h>**:

```
SC_ADD          /* add a resource for swapping */
SC_LIST         /* list the resources for swapping */
SC_REMOVE       /* remove a resource for swapping */
SC_GETNSWP      /* return number of swap resources */
```

When **SC_ADD** or **SC_REMOVE** is specified, *arg* is a pointer to a **swapres** structure containing the following members:

```
char    *sr_name;       /* pathname of resource */
off_t   sr_start;       /* offset to start of swap area */
off_t   sr_length;      /* length of swap area */
```

sr_start and **sr_length** are specified in 512-byte blocks.

When **SC_LIST** is specified, *arg* is a pointer to a **swaptable** structure containing the following members:

```
int     swt_n;                  /* number of swapents following */
struct  swapent swt_ent[];      /* array of swt_n swapents */
```

A **swapent** structure contains the following members:

```
char    *ste_path;      /* name of the swap file */
off_t   ste_start;      /* starting block for swapping */
off_t   ste_length;     /* length of swap area */
long    ste_pages;      /* number of pages for swapping */
long    ste_free;       /* number of ste_pages free */
long    ste_flags;      /* ST_INDEL bit set if swap file */
                        /* is now being deleted */
```

SC_LIST causes **swapctl( )** to return at most **swt_n** entries. The return value of **swapctl( )** is the number actually returned. The **ST_INDEL** bit is turned on in **ste_flags** if the swap file is in the process of being deleted.

When **SC_GETNSWP** is specified, **swapctl( )** returns as its value the number of swap resources in use. *arg* is ignored for this operation.

The **SC_ADD** and **SC_REMOVE** functions will fail if calling process does not have appropriate privileges.

RETURN VALUES

Upon successful completion, the function **swapctl( )** returns a value of **0** for **SC_ADD** or **SC_REMOVE**, the number of **struct swapent** entries actually returned for **SC_LIST**, or the number of swap resources in use for **SC_GETNSWP**. Upon failure, the function **swapctl( )** returns a value of −**1** and sets **errno** to indicate an error.

ERRORS

Under the following conditions, the function **swapctl( )** fails and sets **errno** to:

EEXIST Part of the range specified by **sr_start** and **sr_length** is already being used for swapping on the specified resource (**SC_ADD**).

EFAULT *arg*, **sr_name**, or **ste_path** points to an illegal address.

EINVAL The specified function value is not valid, the path specified is not a swap resource (**SC_REMOVE**), part of the range specified by **sr_start** and **sr_length** lies outside the resource specified (**SC_ADD**), or the specified swap area is less than one page (**SC_ADD**).

EISDIR The path specified for **SC_ADD** is a directory.

ELOOP Too many symbolic links were encountered in translating the pathname provided to **SC_ADD** or **SC_REMOVE** .

ENAMETOOLONG The length of a component of the path specified for **SC_ADD** or **SC_REMOVE** exceeds **{NAME_MAX}** characters or the length of the path exceeds **{PATH_MAX}** characters and **{_POSIX_NO_TRUNC}** is in effect.

ENOENT The pathname specified for **SC_ADD** or **SC_REMOVE** does not exist.

ENOMEM An insufficient number of **struct swapent** structures were provided to **SC_LIST**, or there were insufficient system storage resources available during an **SC_ADD** or **SC_REMOVE**, or the system would not have enough swap space after an **SC_REMOVE**.

ENOSYS The pathname specified for **SC_ADD** or **SC_REMOVE** is not a file or block special device.

ENOTDIR Pathname provided to **SC_ADD** or **SC_REMOVE** contained a component in the path prefix that was not a directory.

EPERM The effective user of the calling process is not super-user.

EROFS The pathname specified for **SC_ADD** is a read-only file system.