NAME

statgrab − get system statistics

SYNOPSIS

```
#include <statgrab.h>

int sg_init(void);

int sg_drop_privileges(void);

sg_error sg_get_error(void);

const char *sg_get_error_arg(void);

intsg_get_error_errno(void);

const char *sg_str_error(sg_error code);

sg_cpu_stats *sg_get_cpu_stats(void);

sg_cpu_stats *sg_get_cpu_stats_diff(void);

sg_cpu_percents *sg_get_cpu_percents(void);

sg_disk_io_stats *sg_get_disk_io_stats(int *entries);

sg_disk_io_stats *sg_get_disk_io_stats_diff(int *entries);

sg_fs_stats *sg_get_fs_stats(void);

sg_host_info *sg_get_host_info(void);

sg_load_stats *sg_get_load_stats(void);

sg_mem_stats *sg_get_mem_stats(void);

sg_swap_stats *sg_get_swap_stats(void);

sg_network_io_stats *sg_get_network_io_stats(int *entries);

sg_network_io_stats *sg_get_network_io_stats_diff(int *entries);

sg_network_iface_stats *sg_get_network_iface_stats(int *entries);

sg_page_stats *sg_get_page_stats(void);

sg_page_stats *sg_get_page_stats_diff(void);

sg_process_count *sg_get_process_stats(void);

sg_user_stats *sg_get_user_stats(void);
```

DESCRIPTION

The statgrab library provides a cross−platform interface to getting system statistics. Each of the function calls returns a structure containing statistics. See the manual page for each individual function for more details on usage.

sg_init must be the first function you call before you start to use libstatgrab; it performs all the one−time initialisation operations that need setuid/setgid privileges. For instance, on *BSD it opens a descriptor to be able to read kernel structures later on, and on Solaris it reads the device mappings that in some cases are only accessible by root (machines with a /dev/osa). Once this has run, the other libstatgrab functions no longer need elevated privileges. It is therefore a good idea to call sg_drop_privileges, which discards setuid and setgid privileges, immediately after you call sg_init, unless your application has another reason for needing setuid or setgid privileges.

sg_init and sg_drop_privileges return 0 on success, and non−zero on failure.

There are three functions relating to error reporting in libstatgrab. The first, sg_get_error returns an sg_error code which relates to the last error generated by libstatgrab. This can be converted to a string by calling sg_str_error giving the sg_error code as an argument. An optional argument may be set when the error was generated. This can be accessed by calling sg_get_error_arg; NULL will be returned if no argument has been set. Some errors will also record the value of the system errno variable when the error occurred; this can be retrieved by calling sg_get_error_errno, which will return 0 if no valid errno has been recorded.

It is the intended practice that whenever a libstatgrab function is called and subsequently fails that an appropriate error will be set.

The library was originally written to support the i−scream central monitoring system, but has since become a standalone package. It has been ported to work on Linux, NetBSD, FreeBSD, OpenBSD, DragonFly BSD, Solaris, HP−UX and Cygwin.

SEE ALSO

sg_get_cpu_percents(3) **sg_get_disk_io_stats**(3) **sg_get_fs_stats**(3) **sg_get_host_info**(3) **sg_get_load_stats**(3) **sg_get_mem_stats**(3) **sg_get_network_io_stats**(3) **sg_get_network_iface_stats**(3) **sg_get_page_stats**(3) **sg_get_process_stats**(3) **sg_get_user_stats**(3)

WEBSITE

http://www.i−scream.org/libstatgrab/