

NAME

capng_apply – apply the stored capabilities settings

SYNOPSIS

```
#include <cap-ng.h>
```

```
int capng_apply(capng_select_t set);
```

DESCRIPTION

capng_apply will transfer the specified internal POSIX capabilities settings to the kernel. The options are CAPNG_SELECT_CAPS for the traditional capabilities, CAPNG_SELECT_BOUNDS for the bounding set, CAPNG_SELECT_BOTH if transferring both is desired, CAPNG_SELECT_AMBIENT if only operating on the ambient capabilities, or CAPNG_SELECT_ALL if applying all is desired.

RETURN VALUE

This returns 0 on success and a negative value on failure. The values are:

- 1 not initialized
- 2 CAPNG_SELECT_BOUNDS and failure to drop a bounding set capability
- 3 CAPNG_SELECT_BOUNDS and failure to re-read bounding set
- 4 CAPNG_SELECT_BOUNDS and process does not have CAP_SETPCAP
- 5 CAPNG_SELECT_CAPS and failure in capset syscall
- 6 CAPNG_SELECT_AMBIENT and process has no capabilities and failed clearing ambient capabilities
- 7 CAPNG_SELECT_AMBIENT and process has capabilities and failed clearing ambient capabilities
- 8 CAPNG_SELECT_AMBIENT and process has capabilities and failed setting an ambient capability
- 9 Unable to acquire process capabilities to check if CAP_SETPCAP is set.

NOTES

If you are doing multi-threaded programming, calling this function will only set capabilities on the calling thread. All other threads are unaffected. If you want to set overall capabilities for a multi-threaded process, you will need to do that before creating any threads. See the capset syscall for more information on this topic.

Also, bits in the bounding set can only be dropped. You cannot set them. After dropping bounding set capabilities, the bounding set is synchronized with the kernel to reflect the true state in the kernel.

SEE ALSO

capset(2), **capng_update(3)**, **capabilities(7)**

AUTHOR

Steve Grubb