NAME

CREATE_SERVICE - casper service declaration macro

LIBRARY

Casper Library (libcasper, -lcasper)

SYNOPSIS

```
#include <sys/nv.h>
#include <libcasper.h>
#include <libcasper_service.h>

typedef int service_limit_func_t(const nvlist_t *, const nvlist_t *);

typedef int service_command_func_t(const char *, const nvlist_t *, nvlist_t *,
    nvlist_t *);

CREATE_SERVICE(name, limit_func, command_func, flags);
```

DESCRIPTION

The **CREATE_SERVICE** macro is used to create a new casper service. The *name* is a string containing the service name, which will be used in the cap_service_open(3), function to identify it.

The *limit_func* is a function of type service_limit_func_t where the first argument of the function contains an nvlist(9), old service limits and the second argument contains the new limits. If the service was not limited then the old limits will be set to NULL. This function must not allow the extension of service limits. The *command_func* is a function of type service_command_func_t where the first argument is the name of the command that should be executed. The first nvlist(9) contains the current limits and the second contains an nvlist(9) with the current request. The last argument contains a return value nvlist(9) which contains the response from casper.

The *flags* argument defines the limits of the service. The supported flags are:

CASPER_SERVICE_STDIO
The casper service has access to the stdio descriptors from the process it was spawned from.

CASPER_SERVICE_FD
The casper service has access to all of the descriptors, besides the stdio descriptors, from the process it was spawned from.

CASPER_SERVICE_NO_UNIQ_LIMITS

The whole casper communication is using an nvlist(9) with the NV_FLAG_NO_UNIQUE flag.

SEE ALSO

cap_enter(2), libcasper(3), capsicum(4), nv(9)

HISTORY

The **libcasper** library first appeared in FreeBSD 10.3.

AUTHORS

The **libcasper** library was implemented by Pawel Jakub Dawidek <*pawel@dawidek.net*> under sponsorship from the FreeBSD Foundation. The **libcasper** new architecture was implemented by Mariusz Zaborski <*oshogbo@FreeBSD.org*>