

NAME

auth_ttyok, **auth_hostok**, **auth_timeok** - functions for checking login class based login restrictions

LIBRARY

System Utilities Library (libutil, -lutil)

SYNOPSIS

```
#include <sys/types.h>
#include <time.h>
#include <login_cap.h>
```

int

```
auth_ttyok(login_cap_t *lc, const char *tty);
```

int

```
auth_hostok(login_cap_t *lc, const char *host, char const *ip);
```

int

```
auth_timeok(login_cap_t *lc, time_t t);
```

DESCRIPTION

This set of functions checks to see if login is allowed based on login class capability entries in the login database, `login.conf(5)`.

The **auth_ttyok()** function checks to see if the named tty is available to users of a specific class, and is either in the *ttys.allow* access list, and not in the *ttys.deny* access list. An empty *ttys.allow* list (or if no such capability exists for the given login class) logins via any tty device are allowed unless the *ttys.deny* list exists and is non-empty, and the device or its tty group (see `ttys(5)`) is not in the list. Access to ttys may be allowed or restricted specifically by tty device name, a device name which includes a wildcard (e.g. `ttyD*` or `cuaD*`), or may name a ttygroup, when `group=<name>` tags have been assigned in `/etc/ttys`. Matching of ttys and ttygroups is case sensitive. Passing a NULL or empty string as the *tty* parameter causes the function to return a non-zero value.

The **auth_hostok()** function checks for any host restrictions for remote logins. The function checks on both a host name and IP address (given in its text form, typically n.n.n.n) against the *host.allow* and *host.deny* login class capabilities. As with ttys and their groups, wildcards and character classes may be used in the host allow and deny capability records. The `fnmatch(3)` function is used for matching, and the matching on hostnames is case insensitive. Note that this function expects that the hostname is fully expanded (i.e., the local domain name added if necessary) and the IP address is in its canonical form. No hostname or address lookups are attempted.

It is possible to call this function with either the hostname or the IP address missing (i.e. NULL) and matching will be performed only on the basis of the parameter given. Passing NULL or empty strings in both parameters will result in a non-zero return value.

The **auth_timeok()** function checks to see that a given time value is within the *times.allow* login class capability and not within the *times.deny* access lists. An empty or non-existent *times.allow* list allows access at any time, except if a given time falls within a period in the *times.deny* list. The format of time period records contained in both *times.allow* and *times.deny* capability fields is explained in detail in the `login_times(3)` manual page.

RETURN VALUES

A non-zero return value from any of these functions indicates that login access is granted. A zero return value means either that the item being tested is not in the *allow* access list, or is within the *deny* access list.

SEE ALSO

`getcap(3)`, `login_cap(3)`, `login_class(3)`, `login_times(3)`, `login.conf(5)`, `termcap(5)`

HISTORY

The functions **auth_ttyok()**, **auth_hostok()** and **auth_timeok()** functions first appeared in FreeBSD 2.1.5.