

SCO OpenServer Product Family Release 5

login(M)

login -- give access to the system

Syntax

login [**-cf**] [*name* [*env-var*]]

login [**-c**] [**-r** *remotehost remotename localname*] ...

Description

The **login** command is used at the beginning of each terminal session to identify users and allow them access to the system. It cannot be invoked except when a connection is first established, or after the previous user has logged out by sending an end-of-file (<Ctrl>d) to their initial shell.

login asks for a user name (if not supplied as an argument), and, if appropriate, the user's password and a dialup password. (For information on dialup passwords, refer to [passwd\(C\)](#)). Echoing is turned off (where possible) during the typing of the passwords, so it will not appear on the written record of the session.

If the user makes a mistake in the login procedure the user will receive the message ``Login incorrect" and a new login prompt will appear. The number of login attempts the user is allowed is configurable. If the user makes too many unsuccessful login attempts, the user or the terminal can be locked out.

If the login sequence is not completed successfully within a configurable period of time (for example, one minute), the user is returned to the ``login:" prompt or silently disconnected from a dial-in line.

The **-c** option must be specified to enable accounting for logins that use pseudo-ttys (over a network or on an [mscreen\(M\)](#)). It can also be used safely for ordinary logins.

The **-f** option enables user login directly without requesting for a password. For instance, **login -f name**.

The form of the command that uses the **-r** option is used for remote logins across a network. The remote login must supply parameters in the order indicated; these are the name of the remote host from which the login is being attempted, the user's name on the remote host, and the user's name on the local host (on which the login process is running). This form of the login command is intended for use by network software rather than users.

After a successful login, accounting files (*/etc/utmp* and */etc/wtmp*) are updated, the user is notified if they have mail, and the start-up shell files (*.profile* for the Bourne shell or *.login* for the C-shell) if any, are executed.

Login sets the user's supplemental groups list. If the file *.suppgroups* is in the user's home directory, the supplemental groups list is taken from this. The *.suppgroups* file contains a list of group names, one per line. Groups are verified before they are added to the supplemental group list.

To be able to use a group, a user must either be explicitly listed in that group in */etc/group*, or the group must have the group ID listed for the user in the */etc/passwd* file. If no *.suppgroups* file is found, the supplemental groups list is set from the */etc/group* file plus the login group ID.

If the hushlogin feature is enabled in */etc/default/login* and a file named *.hushlogin* exists in the user's home directory, **login** suppresses the printing of the last successful and last unsuccessful login times and the copyright messages. **login** also sets the environment variable **HUSHLOGIN** to **TRUE**, so the system and user initialization files are aware a hushlogin is taking place and can suppress output as appropriate (typically the message of the day, and the calling of [mail\(C\)](#) and [news\(C\)](#) are suppressed). The *.hushlogin* file itself does not need to contain anything; it only needs to exist.

login checks */etc/default/login* for the following definitions of the form **DEFINE=value**:

ALTSHELL

If **ALTSHELL** is set to **YES** or if it is not present in */etc/default/login*, then the **SHELL** environment variable is set to whatever shell is specified in the user's */etc/passwd* entry. If **ALTSHELL** is set to **NO**, then the **SHELL** environment variable is set only if the shell is defined in the */usr/lib/mkuser* directory (which is list of recognized shells).

CONSOLE

The **CONSOLE=device** entry means that root can only log in on the device listed. For example, **CONSOLE=/dev/tty01** restricts *root* logins to the first console multiscreen device.

ALLOWHUSH

The **ALLOWHUSH** entry is used to enable or disable the hushlogin feature on a system-wide basis. If **ALLOWHUSH=YES**, **login** checks for the existence of a *.hushlogin* file in the user's home directory. If the file exists, the environment variable **HUSHLOGIN** is set to **TRUE** and a quiet login takes place. If **ALLOWHUSH=NO** or **ALLOWHUSH=YES** and there is no *.hushlogin* file in the user's home directory, the environment variable **HUSHLOGIN** is set to **FALSE** and the normal login messages appear. If there is no **ALLOWHUSH** entry, the **HUSHLOGIN** environment variable is not set and the normal login messages appear.

IDLEWEEKS

If a password has expired, the user is prompted to choose a new one. If it has expired beyond **IDLEWEEKS**, the user is not allowed to log in, and must consult system administrator. This works in conjunction with [passwd\(C\)](#).

OVERRIDE

This allows root to log in on the console even if the Protected Password database entry for root is corrupted. **login** checks */etc/default/login* to see if there is an entry similar to the following, which identifies the tty to be used when doing an override login for root:

```
OVERRIDE=tty01
```

PASSREQ

If **PASSREQ=YES**, a password is required. Users who do not have a password will be forced to select one. **PASSREQ=NO** allows users to have accounts without passwords.

REUSEUID

The **REUSEUID** entry is used by [unretire\(ADM\)](#) and [rmuser\(ADM\)](#).

SUPATH

If a user's UID is 0 (that is, if this is the super user), the **PATH** variable is set to **SUPATH**, if **SUPATH** is specified in */etc/default/login*. It is not advisable for **SUPATH** to include the current directory symbol ``.``. Note that an empty directory (```:`" or ``:`" at the beginning or end) is equivalent to ``.``.

ULIMIT

This variable defines the maximum allowable file size. The default value used by the kernel is specified in the file [mtune\(F\)](#) as 2,097,151 blocks, or approximately 1GB. This value can be changed using [configure\(ADM\)](#); however, for login sessions, a lower value specified in */etc/default/login* overrides the kernel default value.

UMASK

This is the default file creation mask (see [umask\(C\)](#)).

login initializes the user and group IDs and the working directory, then executes a command interpreter (usually [sh\(C\)](#)) according to specifications found in the */etc/passwd* file. Argument 0 of the command interpreter is a dash (-) followed by the last component of the interpreter's pathname. The basic environment (see [environ\(M\)](#)) is initialized to:

```
HOME= user-login-directory  
SHELL=last field of passwd entry  
MAIL=/usr/spool/mail/user-login-name
```

Possible **HUSHLOGIN=TRUE** or **FALSE**

Initially, **umask** is set to octal 022 by **login**.

Diagnostics

Not on system console

login is set up to allow root to log on to the console only, and the user is not on the system console.

Login incorrect

The **login** or dialup password is incorrect.

Unable to change directory to dir

login cannot change directories to the home directory as specified by */etc/passwd*.

No utmp entry. You must exec 'login' from the lowest level 'sh'.

init did not put an entry in *utmp*.

No Root Directory

The shell field starts with a ``*'', and the attempt to do a **chroot** to the home directory failed.

You don't have a password.

A password is required and it has not been set previously.

Protected Password information suddenly vanished

During the course of working with the Protected Password database information the pointer pointing to the static version of the information has suddenly disappeared.

Cannot execute passwd program

The password program cannot be executed for some reason.

Login aborted due to no password.

The password program has returned an error while setting a password, as when the key is pressed.

Can't rewrite Protected Password entry for user name,

Authentication error; see Account Administrator

The **login** program cannot update the Protected Password database entry.

Protected Password database problem

After updating Protected Password data, login reads the information again and the entry cannot be read. This can be caused by redundant database backup files and/or lockfiles; these may be distinguished by a -t suffix. See [tcbck\(ADM\)](#) for information on these files and how to remove them from the system.

Account is disabled but console login is allowed.

Account is disabled -- see Account Administrator.

If the account is locked, but root is logging in on the console (OVERRIDE tty), the first message is displayed; an ordinary user will see the second.

Account has been retired -- logins are no longer allowed.

The account is retired - see [unretire\(ADM\)](#) and [rmuser\(ADM\)](#) on how to unretire or remove an account.

Cannot set terminal mode.

The chmod of the tty failed.

Bad login user id.

No UID has been set. This can be due to a missing critical database file, such as */etc/auth/system/authorize*. Run [authck\(ADM\)](#) and check any error messages. This message will also be issued if login is run from an established login session rather than from [init\(M\)](#).

Wait for login retry.

"Wait for login exit." A login attempt has failed, and the system is configured to enforce a delay between login attempts.

user appears in */etc/passwd* but not in Protected Password database

If the user is in */etc/passwd* but not in the Protected Password database, there is no message printed, but login generates the audit record shown above.

Cannot obtain database information on this terminal

login cannot get information from */etc/auth/system/ttys* for the tty line.

Error in terminal setup.

Something is wrong with the terminal setup (for example, *stdin*, *stdout*, and *stderr* are the same thing.)

Cannot obtain settings for this terminal

The [ioctl\(S\)](#) on the tty device failed.

No login program on root

When attempting to perform a sublogin (using [chroot\(ADM\)](#) to change to a subtree for a restricted login), no login program was found.

Can't rewrite terminal control entry for tty,

Authentication error; see Account Administrator

The information for the login tty cannot be updated.

Terminal Control information suddenly vanished

During the course of working with the terminal database information the pointer pointing to the static version of the information suddenly disappeared.

Bad priority setting.

nice failed to set the nice value specified in the Protected Password entry for the user.

Bad supplemental group list.

The call to setgroups failed.

Bad group id.

The call to setgid failed.

Bad user id.

The call to setuid failed.

Unable to set kernel privileges.

The call to set the kernel privileges failed.

Login timed out

login received an ALARM signal. Note: login sets this itself, but it could conceivably come from somewhere else.

Terminal is disabled but root login is allowed.

Terminal is disabled -- see Account Administrator.

If the terminal is disabled and root attempts to login on the (OVERRIDE) tty the first message is displayed; the second message is displayed when any other user attempts to login on a disabled terminal.

The security databases are corrupt.

However, root login at terminal tty is allowed,

This is the message displayed when the OVERRIDE tty is used during a security problem.

Impossible to execute /bin/sh!

login cannot execute the shell program for doing an OVERRIDE.

Limitations

login cannot be executed from a shell.

If the Network Information Service (NIS) is disabled, accounts that are normally accessed from an NIS server will not be able to log in.

Environment variables such as HZ, PATH, and so forth should not be defined in /etc/default/login. Instead use /etc/initscript to set global variables.

Sublogins (indicated by a shell of ``*") are not supported and cause a warning.

Although **IDLEWEEKS** and **PASSREQ** are supported for compatibility with other UNIX systems, their use is not recommended. The proper way to set the behavior defined by these variables is by use of the Accounts Manager.

Files

/etc/utmp

information on current logins

/etc/wtmp

history of logins since last multiuser

/usr/spool/mail/name

mailbox for user *name*

/etc/motd

message of the day

/etc/default/login

default values for environment variables and login behavior

/etc/passwd

password file

/etc/profile

system profile for Bourne or Korn shell

\$HOME/.profile

personal profile for Bourne or Korn shell

\$HOME/.login

personal C shell login file

\$HOME/.cshrc

personal C shell initialization file

\$HOME/.suppgroups

supplemental groups file

\$HOME/.hushlogin

make login quieter

See also

[environ\(M\)](#), [getty\(M\)](#), [mail\(C\)](#), [newgrp\(C\)](#), [passwd\(C\)](#), [passwd\(F\)](#), [profile\(M\)](#), [rmuser\(ADM\)](#), [sh\(C\)](#), [sg\(C\)](#), [su\(C\)](#), [ulimit\(S\)](#), [umask\(C\)](#), [unretire\(ADM\)](#), [who\(C\)](#)

1 May 1995