

Communicating with Users [8]

As a system administrator, you must communicate with your users frequently. Several methods of communication are available for you to use. The method to use in any specific instance generally is determined by the urgency of your message.

The following list describes the types of communication you will maintain with users, as well as the commands associated with that kind of communication:

<u>Type of communication</u>	<u>Command or file</u>
Issuing emergency messages only	<code>/etc/wall</code>
Issuing critical messages	<code>/etc/issue</code>
Issuing special messages (message of the day)	<code>/etc/motd</code>
Issuing normal (noncritical) communication to all users	<code>/usr/news</code>
Communicating with specific users	<code>write</code> and <code>mail</code>

This chapter describes when you should use each type of communication and gives examples of each.

8.1 Related user communication documentation

The following documentation contains detailed information covered in this chapter:

- *UNICOS User Commands Reference Manual*, Cray Research publication SR-2011: `mail(1)`, `news(1)`, `su(1)`, `wall(1)`, and `write(1)` man pages
- *UNICOS File Formats and Special Files Reference Manual*, Cray Research publication SR-2014: `issue(5)` and `motd(5)` man pages

8.2 Communicating with users

During the operation of a UNICOS system, it is frequently necessary for administrators to use the system to communicate information to its users. This

section discusses a number of UNICOS commands and tools that enable you to communicate with users:

- The `wall(8)` command
- The `/etc/motd` file
- The `/etc/issue` file
- The `/usr/news` directory
- The `write(1)` utility
- The `mail(1)` utility

8.2.1 The `wall` command

The `wall(8)` command broadcasts items of immediate concern to all users currently logged in to the system. Run the command by typing the following:

```
/etc/wall
```

The `wall` command responds by telling you to type your message and to press `CONTROL-d` when you are finished. To ensure that all users who are currently logged in see a message sent by `wall`, run the command while you have `root` privileges; otherwise, the message goes only to users who allow messages to be written to their terminals (see `mesg(1)`). Additionally, users who are not currently logged in will never see the message; `wall` is thus not a suitable method for communicating a message to all users who have accounts on the system.

The `wall` command is typically used to send the following messages:

- Warnings that the system will soon be brought down for scheduled downtime. Users who log in after the message is sent, however, miss the message and should be notified by the `/etc/issue` file (see `login(1)`).
- Warnings that the system must be brought down immediately to address a system emergency.
- Warnings that a particular file system has run out of disk space and that users should make an immediate effort to delete any unneeded files (see the description of the `-g` option on the `wall(8)` man page).

8.2.2 The `/etc/motd` file

The `/etc/motd` (message-of-the-day) file is displayed to users after they are logged in to the system. The `/etc/motd` file is an ordinary text file, and the administrator may place messages in it by using any UNICOS text editor.

Messages that should be placed in `/etc/motd` are those that are less immediate than those requiring the use of `wall(8)`, but they are important enough that users should be forced to see them. The administrator should remove messages from `/etc/motd` as soon as they are no longer needed. Suitable items for inclusion in this file include the following:

- Warnings to users to clean up unnecessary files on a particular file system or systems
- Brief explanations of recent problems that may have affected a number of users, often with a pointer to a news item containing a more detailed explanation

8.2.3 The `/etc/issue` file

The `/etc/issue` file is displayed while a user is logging in, before the user has successfully logged in to the system. It is an ordinary text file, and you may place messages in it by using any UNICOS text editor.

Messages placed in `/etc/issue` should be brief and so important that users may need the information to decide whether or not to log in to the system. Possible messages include the following:

- Warnings that the system will be brought down soon (so that users who do not see a `wall(8)` message are not surprised when the system is brought down shortly after they log in)
- Warnings that the system is being used for dedicated time and that not all users will be able to log in

8.2.4 The `/usr/news` directory

When users log in to the system they are alerted to the existence of any new files placed in the `/usr/news` directory. When a user then runs the `news(1)` utility, it displays any news files that have been created or modified since the last time the user ran `news`. The files placed in `/usr/news` are ordinary text files created with any UNICOS text editor, and they are usually assigned names that give a general idea as to their contents. For instance, a news file containing

information about a modification to a system library might be given the name `new.library`.

Because users are not notified of the existence of a new news file until the next time they log in, and because there is no guarantee that any given user will see the file (a user may choose to ignore the item by not running the `news` utility), `/usr/news` is appropriate for items that are not time-sensitive or items that are of interest to only some of the system's users. These categories include the following:

- Notices regarding recent system changes, such as a newly installed version of a command or library
- Explanations of imminent system reconfigurations or changes
- Explanations of recent system problems and their possible effects on users

It is a good idea to remove any old files in `/usr/news` periodically, not only to save disk space, but also to prevent new users on the system from having to read through a long list of out-of-date news items. The `/usr/news` file may be cleaned out regularly by `cron(8)`.

8.2.5 The `write` utility

The `write(1)` utility initiates immediate person-to-person communication with a logged-in user by opening that user's `tty` or `pty` for writing and copying each line of text you type to his or her screen. To write to a user with a login name of `dolores`, for example, you would issue the following command:

```
write dolores
```

If the user `dolores` happened to be logged in on more than one `tty` or `pty`, you could specify the connection:

```
write dolores tty001
```

If, in this example, the user `dolores` is currently logged in, a message appears on her screen indicating that you are writing to her. Typically, the user `dolores` replies by writing back to your account; each line of text she types appears on your screen.

Given the immediate nature of its communication, the `write` utility allows you to perform the following functions:

- Converse with a user

- Obtain information about what a user is doing
- Warn a specific user to stop what he or she is doing
- Instruct a specific user to clean up his or her directories

Because each typed line appears on the other user's terminal without regard for what that person may be typing at the moment, it is easy for the other user's messages to your terminal to appear to interfere with your typing. This problem is customarily solved by having the two users take turns typing, ending a message with an `o` on a line by itself (standing for "over," much as in a two-way radio conversation). To end such a session, either user then ends a message with an `oo` on a line by itself (for "over and out"). Thus, a typical "conversation" carried out by `write` might look like this (your input appears in **bold**):

```
# write dolores
Message from dolores (ttyp001) - Mon May 11 08:20:15 - ...
Yes
o
Please clean up your account, we're out of space.
o
All right, I will.
o
Thank you.
oo
<EOT>
```

Because many users either do not know of this etiquette when using `write`, or do not follow it, they think that `write` is difficult to use. In practice, it is used rather sparingly, mainly when more convenient forms of communication (such as simply calling the user on the telephone) are impossible. Taking steps to educate your user community in the proper use of the `write` utility will prove valuable when `write` is the appropriate communication method.

Note: On a UNICOS system or Cray ML-Safe configuration, for `write` to execute properly, the user's active security labels must be equal.

8.2.6 The mail utility

The mail(1) utility provides a way to leave messages for specific users, whether or not they are currently logged in to the system. The mail utility is used as follows:

```
mail ralph
```

Type in message

```
CONTROL-d
```

You may specify more than one account name, in which case copies of the message go to each user named. The next time users to whom you (or anyone else) have sent mail messages log in to the system, the system alerts them to the fact that they have mail messages waiting. The mail utility is thus particularly well suited for messages such as the following:

- Instructions to clean up directories
- Asking or responding to questions
- General communication

In theory, there is no guarantee that the recipient of a mail message will actually see the message, because the recipient may choose not to run the mail utility to read the message; however, in practice, most users read their mail when they log in.

Note: On a UNICOS system or Cray ML-Safe configuration, the recipient of a mail message might not be authorized to read mail at the classification with which it was sent.

For more information see mail(1) and mailx(1).