DECnet–ULTRIX

digital

Use

Order Number: AA–PB61A–TE

DECnet–ULTRIX

Use

May 1990

This manual shows how you use the DECnet–ULTRIX user commands to perform file transfer and other user tasks.

Supersession/Update Information:	This is a new manual.
Operating System and Version:	ULTRIX V4.0
Software Version:	DECnet–ULTRIX V4.0

digital™

Order Number: AA–PB61A–TE

AA–PB61A–TE
May 1990

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may only be used or copied in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by Digital or its affiliated companies.

Restricted Rights: Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227–7013.

The following are trademarks of Digital Equipment Corporation:

DEC	PDP	VAX
DECnet	ULTRIX	VMS
DECUS	UNIBUS	digital™

UNIX is a registered trademark of AT&T in the USA and other countries.

This manual was produced by Networks and Communications Publications.

Contents

Preface

This manual contains both tutorial and reference material for DECnet–ULTRIX end users.

Intended Audience

This manual is for anyone who wants to use DECnet–ULTRIX to log on to remote DECnet nodes, exchange mail with users on remote DECnet nodes, and work with files on DECnet nodes. You should be familiar with the ULTRIX operating system and general file transfer principles. You should also know how to work on the DECnet nodes you plan to access via remote login.

Structure of This Manual

This manual contains six chapters and two appendixes:

- Chapter 1 introduces DECnet–ULTRIX product features and utilities.
- Chapter 2 describes how to use the **dlogin** command to log on to other DECnet systems.
- Chapter 3 describes how to use the ULTRIX **mail** utility to exchange mail with DECnet nodes.
- Chapter 4 explains how to specify remote files on a DECnet node.
- Chapter 5 describes how to display, copy, and delete files and directories on a remote DECnet node.
- Chapter 6 describes the DECnet–ULTRIX commands in detail. This reference chapter is also available on line using the **man** command.
- Appendix A lists the DECnet–ULTRIX error messages.
- Appendix B shows sample DECnet file specification formats.

Related Documents

To supplement the *DECnet–ULTRIX Use* manual, see the following DECnet–ULTRIX documents:

- *DECnet–ULTRIX Release Notes*

  This document contains miscellaneous information and updates not included in other books in the DECnet–ULTRIX documentation set.

- *DECnet–ULTRIX Installation*

  This manual describes procedures for installing, customizing, and testing a DECnet–ULTRIX node for proper operation. This manual also lists the DECnet–ULTRIX distribution files and the directory path names.

- *DECnet–ULTRIX DECnet–Internet Gateway Use and Management*

  This manual describes how to install, use, and manage the DECnet–Internet Gateway.

- *DECnet–ULTRIX Programming*

  This manual explains concepts and offers guidelines for application programming in the DECnet–ULTRIX programming environment. This manual also describes DECnet–ULTRIX system calls and subroutines, and shows DECnet–ULTRIX data structures and programming examples.

- *DECnet–ULTRIX Network Management*

  This manual describes procedures for managing the network, such as defining permanent and volatile databases, defining node identifications and addresses, defining lines and circuits, enabling event logging, displaying network counter information, operating and controlling a DECnet–ULTRIX node, and testing the network operation.

- *DECnet–ULTRIX NCP Command Reference*

  This manual describes the Network Control Program (**ncp**) commands for defining, monitoring, and testing your network.

To obtain a detailed description of the Digital Network Architecture, see *DECnet Digital Network Architecture (Phase IV), General Description.*

For a beginner's introduction to the ULTRIX operating system, see *The Little Gray Book: An ULTRIX Primer.*

Graphic Conventions

This manual uses the following conventions:

Convention	Meaning
special	In running text, ULTRIX commands, command options, user names, file names, and directory names appear in **special** type.
`example`	Indicates an example of system output or user input. System output is in black type; user input is in red type.
lowercase/UPPERCASE	Because the ULTRIX software is case sensitive, you must type all literal input in the case shown. In running text, UPPERCASE is also used for the names of all DECnet nodes, including DECnet–ULTRIX nodes. This convention follows DECnet protocol, which names and recognizes all nodes in UPPERCASE. However, node names are not case sensitive and need not be typed in the case shown.
italic	Indicates a variable, for which either you or the system specifies a value.
[ ]	On command syntax lines, square brackets indicate optional arguments. Do not type the brackets.
key	Indicates a key on your keyboard. CTRL/*key* represents a CONTROL key sequence, where you press the CONTROL key at the same time as the specified key. Note that keyboard keys are represented by this symbol, <*key*>, on line.
RET	Indicates the RETURN key.
%	The percent sign, the standard ULTRIX system prompt, is used in all examples in this manual to indicate an ULTRIX system.
$	The dollar sign, the standard VMS system prompt, is used in all examples in this manual to indicate a VMS system.

NOTE

All numbers are decimal unless otherwise noted.

Terminology

In this manual, "DECnet–RSX" stands for any of these DECnet products: DECnet-11M-PLUS, DECnet–Micro/RSX, DECnet-11S, DECnet-11M.

The following acronyms are used in this manual:

DDCMP	Digital Data Communications Message Protocol
DNA	Digital Network Architecture
FTP	File Transfer Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol

Chapter 1

Introduction to the DECnet–ULTRIX Product

The DECnet–ULTRIX product is software that runs on an ULTRIX system. With DECnet–ULTRIX software, an ULTRIX system can act as an end node on a DECnet network. The DECnet–ULTRIX product is an end-node implementation of Digital Network Architecture (DNA) Phase IV.

You can connect your DECnet-ULTRIX system to a DECnet network either through Digital Data Communications Message Protocol (DDCMP) point-to-point line or a DECnet Ethernet cable. A DECnet–ULTRIX node can access all other Phase III and Phase IV **Decnet nodes** on its network through direct communication with a Phase IV routing node. The routing node for a DECnet–ULTRIX system is either the adjacent node on its DDCMP point-to-point line or one of the nodes on the same Ethernet.

DECnet-ULTRIX software offers numerous features and capabilities:

- User commands that support standard ULTRIX conventions, including support of regular expressions, wildcards, and metacharacters. DECnet–ULTRIX software offers remote login to remote DECnet nodes; mail exchange; and file access and transfer between systems. The DECnet–ULTRIX user commands and utilities are documented in this manual.
- Software that supports the DECnet–Internet Gateway, which offers **bidirectional gateway** functions between DECnet and *Internet* systems, including file transfer, remote log-in, and mail.

  For example, an VMS user on a DECnet node and a UNIX user on an Internet node can display, copy, and delete each other's files; exchange mail; and log in to each other's systems. (The Gateway offers VMS users access to the powerful resources of the ULTRIX environment.) For more information about the DECnet–Internet Gateway, see the *DECnet–ULTRIX DECnet–Internet Gateway Use* manual.
- A programming interface that lets you write cooperating programs to exchange data over a DECnet network. For more information about DECnet–ULTRIX programming, see the *DECnet–ULTRIX Programming* manual.
- Network management features that let you configure a DECnet–ULTRIX node, test network performance, monitor network activity, and manage network components. For more information about network management, see the *DECnet–ULTRIX Network Management* manual and the *DECnet–ULTRIX NCP Command Reference*.

The following sections describe these features and capabilities.

1.1 DECnet–ULTRIX User Commands and Utilities

Table 1-1 shows the tasks DECnet–ULTRIX users can perform and describes the commands associated with those tasks.

Table 1–1: Overview of DECnet–ULTRIX User Commands and Utilities

Task	Command	Description
Remote login	**dlogin**	The **dlogin** command lets you log on to remote DECnet nodes so that you can access the resources of those systems. DECnet–ULTRIX software also lets remote DECnet users log on to your node. For more information, see Chapter 2 or 6.
Mail exchange	**mail**	The ULTRIX **mail** utility lets you exchange messages with other DECnet users. For more information, see Chapter 3 and *The Little Gray Book: An ULTRIX Primer.*
Directory listing	**dls**	The **dls** command lets you examine the contents of remote directories. DECnet–ULTRIX software also lets remote DECnet users display the contents of your directories. For more information, see Chapter 5 or 6.
File display	**dcat**	The **dcat** command displays the contents of one or more remote DECnet files. DECnet–ULTRIX software also lets remote DECnet users display the files on your system. For more information, see Chapter 5 or 6.
File transfer	**dcp**	The **dcp** command lets you copy files to and from remote DECnet systems. Remote DECnet users can initiate copy requests to your node. For more information, see Chapter 5 or 6.
File deletion	**drm**	The **drm** command lets you delete files from remote directories. Remote DECnet users can delete files from your local directories. For more information, see Chapter 5 or 6.

1.2 DECnet–Internet Gateway Features

The DECnet–Internet Gateway offers the following features:

Telnet

You can use Telnet commands through the Gateway. Telnet is the standard Internet application protocol for remote login. Using Telnet through the Gateway, Internet users can log on to remote DECnet nodes.

File Transfer Protocol

You can use File Transfer Protocol (**ftp**) through the Gateway. **ftp** is the primary Internet standard for file transfer. Using **ftp** through the Gateway, Internet users can access and manipulate files and directories on remote DECnet nodes.

Mail systems	You can use VMS, ULTRIX, and UNIX mail systems through the Gateway. Specify recipients according to the addressing syntax described in the *DECnet–ULTRIX DECnet–Internet Gateway Use* manual.
Common DECnet commands	You can use various common DECnet commands through the Gateway. See Chapter 3 for sample DECnet-VAX commands.

1.3 DECnet–ULTRIX Programming Interface

The DECnet–ULTRIX programming interface offers the following features:

Client-server communication	Sometimes called task-to-task communication, client-server communication lets DECnet–ULTRIX applications communicate with remote Phase III and Phase IV DECnet applications through a socket-level programming interface.
DECnet and TCP/IP coexistence	DECnet protocols and Transmission Control Protocol/Internet Protocol (TCP/IP) coexist and share system resources, including Ethernet and DDCMP hardware. You can easily modify most TCP/IP programs to use DECnet protocols, or DECnet programs to use TCP/IP protocols. You can use DECnet and TCP/IP simultaneously on an Ethernet, and you can alternate between the two protocols on DDCMP point-to-point lines.
File access	Programs on any other DECnet Phase III or Phase IV system can access DECnet–ULTRIX files and directories for sequential reading, writing, or deletion.

1.4 DECnet–ULTRIX Network Management Features

The DECnet–ULTRIX network management software offers the following features:

Node configuration	You can configure your DECnet–ULTRIX node to ensure that it runs smoothly on the network.
Network performance testing	You can test your DECnet–ULTRIX node's performance on the network.
Network Control Program (ncp)	You can manage the components of your network and test network performance. You can also display information on the condition, characteristics, and performance of network components.

1.5 Similarities Between DECnet–ULTRIX and ULTRIX

Both new and experienced ULTRIX users can easily become familiar with the DECnet–ULTRIX user environment because:

- DECnet–ULTRIX commands support standard ULTRIX conventions, including the use of regular expressions, wildcards, and metacharacters.

- You can **pipe** or **redirect** the DECnet–ULTRIX commands' input or output according to standard ULTRIX conventions.
- The DECnet–ULTRIX mail system utilizes the ULTRIX **mail** interface, requiring ULTRIX users to learn only a new syntax for creating mail recipients' addresses.
- DECnet–ULTRIX provides on-line manual pages (also called *man pages*) that follow standard ULTRIX man page conventions. The DECnet–ULTRIX manual pages, however, use a "dn" page identifier to distinguish them from ULTRIX manual pages. For example:

  To see the ULTRIX base system manual page for the **write** command, type:

  ```
  % man write RET
  ```

  To see the ULTRIX base system manual page for the **write** system call, type:

  ```
  % man 2 write RET
  ```

  To see the DECnet–ULTRIX manual page for the **write** system call, type:

  ```
  % man 2dn write RET
  ```

  The following section provides more information about on-line manual pages.

1.6 Displaying On-Line Manual Pages

On-line manual pages are pages from reference manuals that you can display on your screen. See the on-line manual pages for detailed descriptions of all the DECnet–ULTRIX commands. (Note that the DECnet–ULTRIX manual pages also appear in the DECnet–ULTRIX Command Summary (Chapter 6) of this manual.

To display an on-line manual page for a specific command, enter **man** and the command name. You can even display a reference page describing the **man** command itself. Enter:

```
% man man RET
```

The system displays:

```
                                                              man(1)

NAME
    man - displays manual pages on-line

SYNTAX
    man -k keyword...
    man -f page_title...
    man [-] [-t] [-s] [1...8] page_title...

DESCRIPTION
    The man command is a program that provides on-line displays
    of the reference pages.  Using options, you can direct the
    man command to display one line summaries of reference pages
    which contain a specific keyword, or you can use this com-
    mand to display one line summaries of specific reference
    pages.

OPTIONS
    -k                  display one line summaries of each
                        reference page that contains the speci-
                        fied keyword or keywords.
--More--(27%)
```

The –**More**– symbol indicates that there is more information available. Press the space bar or RET to see more, or type **q** (without pressing RET) to quit. The **(27%)** symbol indicates that 27% of the information available on the **man** command is displayed. The **man(1)** symbol at the top of the display is the page title.

NOTE

To use the **man** command, you must have the ULTRIX manual pages installed on your local system.

Chapter 2

Logging On to a Remote DECnet Node

Using the **dlogin** command, you can log on to a remote DECnet node and use programs running on that system. The remote DECnet node can be another DECnet–ULTRIX system or a different operating system running DECnet, such as a VMS system. (In this chapter, your target node is referred to as a **remote node**. Your target node, however, could just as easily be your **local node**.)

This chapter describes how you can use **dlogin** to establish and control a login session with a remote DECnet node. This chapter tells you how to:

- Log on to a remote DECnet node
- Log off a remote DECnet node
- Enter local command mode
- Display a list of **dlogin** commands
- Suspend the **dlogin** session
- Record, or log, the **dlogin** session
- Select a new escape character
- Use the escape character without entering local command mode

Be sure you understand how to use **dlogin** before you log on to a remote DECnet node. See the next section.

2.1 Understanding dlogin

This section describes the **dlogin** session you create when you log on to a remote DECnet node.

Your local node is your local DECnet–ULTRIX system. When you log on to this node, you start a **local login session**. During this local login session, you can use the **dlogin** command to log on to a remote DECnet node. A remote DECnet node is any node in the network that is running DECnet software. When you log on to a remote node, you start a remote login session. That session is called the **dlogin** session because you use the DECnet login command, **dlogin**, to initiate it.

During the **dlogin** session, you can run programs and access resources available on the remote system, according to whatever privileges the remote system grants you. You can also interrupt your **dlogin** session and execute commands on your local node. To do this, you use the **dlogin** escape character. The escape character can be any keyboard character; the default escape character is the **tilde** (~).

When you type the escape character and press RET, the escape character is not echoed to your screen or even read by the remote node. Instead, the local node, which controls your **dlogin** session, reads the escape character and interprets it as your signal to interrupt your activities on the remote node.

When you interrupt your **dlogin** session, you enter local command mode. Local command mode is simply an interface to your local node. The **dlogin** session is still active while you are in local command mode. Local command mode is characterized on screen by the **local command>** prompt.

At the **local command>** prompt, you can issue any ULTRIX or DECnet–ULTRIX command you want to execute on your local node. You can also issue the commands described in this chapter that control your **dlogin** session (for example, the commands that suspend your **dlogin** session or open a new **log file**).

The following sections show you how to begin, end, and control your **dlogin** session.

2.2 Logging On to a Remote DECnet Node

To log on to a remote DECnet node from your DECnet–ULTRIX system, type the **dlogin** command followed by the remote DECnet node name. The remote node prompts you for the login information it requires. Once you are logged on to the node, you can issue commands to that system. In the following example, user Irene logs on to the remote VMS node BACON and issues the VMS SHOW USERS command:

```
% dlogin bacon RET

Username:  IRENE RET
Password:  GOODNIGHT RET   (not echoed)

               Welcome to VAX/VMS V5.2 on node BACON

$ SHOW USERS RET
```

2.3 Logging Off a Remote Node

To log off the remote DECnet node, simply type the remote node's logout command. Logout commands vary; for example, the command is **logout** on DECnet–VAX and **bye** on DECnet–RSX. If you do not know the logout command for the remote operating system you are using, see that system's documentation.

In the following example, user Irene is shown logging off a DECnet–VAX node:

```
$ logout RET
IRENE   logged out at   6-JAN-1990   09:46:29.97
dlogin -- session terminated
%
```

Remember, when you log off the remote node, you end the **dlogin** session.

To end a **dlogin** session before you have logged on to the remote node, type the escape character (the tilde (~) is the default) and press RET. At the **local command>** prompt, press CTRL-D or type **exit** and press RET.

In this example, user Frank ends the **dlogin** session because he cannot remember the user name and **password** that have been assigned to him on the remote node BACON:

```
% dlogin bacon RET

Username: FRANK RET
Password: ORANGES RET  (not echoed)
User authorization failure
Username: ~ RET  (not echoed)

? for HELP in local command mode
local command> exit RET

dlogin -- session terminated
%
```

2.4 Entering Local Command Mode

Using the **dlogin** escape character, you can interrupt your login session on the remote node and enter local command mode.

At the **local command>** prompt, you can execute any ULTRIX or DECnet–ULTRIX command on your local node. You can also perform the following tasks, as described in this chapter:

- Display a list of **dlogin** commands
- Suspend the **dlogin** session
- Record the **dlogin** session

To enter the local command mode, type the escape character followed by RET at the remote system's prompt. (The tilde (~) is the default escape character.) The screen displays the **local command>** prompt. For example:

```
$ ~ RET  (not echoed)

? for HELP in local command mode
local command>
```

To return control to your **dlogin** session on the remote node, press RET at the **local command>** prompt, as follows:

```
local command> RET
$
```

You may have to press RET twice for the system prompt to appear.

2.5 Displaying a List of dlogin Commands

To display helpful information about the commands that control your **dlogin** session, type a question mark (?) at the **local command>** prompt. For example:

```
local command> ?
local command> ?              Displays this menu.
local command> >filename      Logs session to specified file.
local command> >>filename     Logs session, appending it to specified file.

local command> >              Closes the open log file.
local command> suspend        Suspends dlogin session.
local command> exit           Exits dlogin session.

local command> <CTRL-D>       Exits dlogin session.
local command> cmd            Invokes shell to execute a command.
local command>                (Blank line) - Resumes dlogin session.

[No log file active]
```

NOTE

In this display, *cmd* represents any ULTRIX command you can enter at the **local command>** prompt.

2.6 Suspending the dlogin Session

If you want to interrupt your **dlogin** session and return control (temporarily) to your local node, you can **suspend** your **dlogin** session. A suspended **dlogin** session is not active. As a result, the screen displays your local node's prompt, and you can execute commands at your local node.

To suspend a session, type the escape character followed by RET at the remote system's prompt. When the **local command>** prompt appears, type the **suspend** command followed by RET. For example:

```
$ ~ RET   (not echoed)

? for HELP in local command mode
local command> suspend RET

Stopped
%
```

At your local system's prompt, you can execute any commands or procedures that you usually execute on your local system.

When you are ready to resume the **dlogin** session, you simply issue a **fg** command to bring the session to the foreground and press RET at the **local command>** prompt. The following example shows a suspended session on node BACON being resumed:

```
% fg RET
dlogin bacon
local command> RET
$
```

2.7 Recording the dlogin Session

If you want to maintain a record of a **dlogin** session, you can open a log file. The log file contains both your input and the system's responses. The **dlogin** command allows you to have the session logged to a new log file or appended to an existing log file. The **dlogin** command also allows you to close the log file during the session.

2.7.1 Creating a Log File

You can open a new log file using either of the following methods:

- **When logging on to the remote DECnet node:** use the **-l** option on the **dlogin** command line and specify the name of the log file to which you want the session logged. For example, the following command line begins a **dlogin** session on node BACON and opens the file **logfile3**:

  ```
  % dlogin bacon -l logfile3 RET
  ```

- **During a session on the remote node:** enter local command mode with the escape character and, at the **local command>** prompt, issue the *>filename* command, where *filename* specifies the name of a log file. For example, the following command opens the file **logfile3**:

```
$ ~ RET   (not echoed)

? for HELP in local command mode
local command> >logfile3 RET
local command> RET
```

If you specify an existing log file when you use either of the two preceding commands, the system replaces the contents of that file with the new log file information.

If you begin your session using the **-l** option to open a log file and later, during the session, you open a second log file with the *>filename* command, the first log file is automatically closed.

2.7.2 Adding New Log Information to a Log File

You can add, or append, new log information to an existing log file. Enter the local command mode and issue the *>>filename* command, where *filename* specifies the name of the existing log file. For example, the following command appends a record of the current session to the file **logfile_1990**:

```
$ ~ RET   (not echoed)

? for HELP in local command mode
local command> >>logfile_1990 RET
local command> RET
```

2.7.3 Closing a Log File

Log files are closed automatically when you log off the remote node and end the **dlogin** session. To close a log file during a **dlogin** session, enter the local command mode and issue the **>** command. For example:

```
$ ~ RET   (not echoed)

? for HELP in local command mode
local command> > RET
local command> RET
```

2.8 Selecting a New dlogin Escape character

When you start a **dlogin** session, you can select a new escape character for that session. The character you select is the escape character only for the length of that **dlogin** session. To select a different escape character, use the **-e** option on the **dlogin** command line. For example, the following command starts a **dlogin** session and sets the escape character to the circumflex (^) character:

```
% dlogin bacon -e^ RET
```

2.9 Using the Escape Character Without Entering Local Command Mode

You might want to use the escape character as part of a mail message or a command line without signaling the **dlogin** session to enter local command mode.

Type the escape character once, followed by any character (even the escape character) except RET. The escape character appears on your screen only after you have typed another character. For example:

```
% dcp -r kimono::~uucp . RET
```

or

```
% cd ~~ RET  (one ~ is echoed)
```

Chapter 3

Sending Mail to Users on Remote DECnet Nodes

You can use the ULTRIX mail command to send messages to users on remote DECnet nodes. You use the recipient's DECnet mail address as the destination for the message. This chapter defines the DECnet mail address and shows sample mail messages.

For information on the conventions used by ULTRIX **mail**, see *The Little Gray Book: An ULTRIX Primer*.

3.1 The DECnet Mail Address

The **DECnet mail address** identifies the user who will receive the mail you are sending. The DECnet mail address consists of the user's DECnet node name, a double colon (::), and the user's DECnet **user name**. For example, the DECnet mail address for user **Jim** on node DAVIS is

```
davis::jim
```

You can also follow the popular Internet mail addressing syntax to identify mail recipients. Internet syntax is as follows:

username`@`*node*`.dnet`

where

username	is the name of the mail recipient.
node	is the name of the recipient's node.
.dnet	is the Internet pseudodomain name.

For example, the **Internet mail address** for user **Markie** on node TWICE is

```
markie@twice.dnet
```

The Internet addressing syntax is provided merely as a convenient interface for Internet users. DECnet–ULTRIX actually converts this Internet address to the DECnet syntax before sending the mail out.

3.2 Sending Mail to a Remote DECnet User

To send mail to a user on a remote DECnet node, type the DECnet–ULTRIX **mail** command and the user's DECnet (or Internet format) mail address. For example:

```
% mail davis::jim RET
```

3.2.1 Mailing a New Message

In this example, user **Helen** on a DECnet–ULTRIX node sends mail to user **Jim** on node DAVIS.

```
% mail davis::jim RET
Subject: New Position RET
 RET
Dear Jim, RET
 RET
You've been in Davis, California, for three years now. RET
If you are open to moving to Atlanta, we would like to RET
consider you for a position here.  Please let me know RET
by next week how you feel about this. RET
 RET
                    Sincerely, RET
 RET
                    Helen RET
 CTRL/D
Cc: helen RET
%
```

3.2.2 Mailing a File to a Remote DECnet User

In this example, user **Jim** on node DAVIS sends the existing file **resume** to user **Helen** on remote DECnet–ULTRIX node ATLANT. **Jim** does not send a copy of the message to anyone.

```
% mail atlant::helen RET
Subject: helen, here is my updated resume RET
~r resume RET
"resume" 23/1204. RET
 CTRL/D
Cc: RET
%
```

3.2.3 Special Considerations

Be aware that a VMS node will not accept a message from ULTRIX if the "TO:" field of the message exceeds 512 characters. To reduce the number of characters in this field, use a **sendmail** alias.

For more details about setting up an **alias**, see the description in **/usr/lib/aliases**.

Chapter 4

Specifying Files on Remote DECnet Nodes

Before you can view or work with remote files and directories, you must be able to specify those files and directories. This chapter explains how to specify remote files and directories using the following information:

- File specification formats for all the operating systems that run DECnet
- Wildcard characters for use in file specifications
- **Access-control information** required to access files on a remote node
- Short cuts for specifying the full access-control string in the file specification

The examples in this chapter illustrate three DECnet–ULTRIX commands, **dcat**, **dcp**, and **dls**. In these examples, **dcat** displays the contents of files, **dcp** copies files, and **dls** lists the contents of directories. For more information about these commands, see Chapters 5 and 6.

4.1 File Specification Formats

Using DECnet–ULTRIX commands, you can work with files that reside on remote ULTRIX and **non-ULTRIX DECnet nodes**. The format you use to specify the files will vary according to the type of system the file resides on.

Table 4–1 lists the file specification formats for files on ULTRIX and non-ULTRIX DECnet nodes. For more information on these file specifications, refer to Appendix B.

Table 4–1: File Specifications for DECnet Nodes

DECnet System	File Specification
DECnet–VAX	*node::'dev:[directory]filename.typ;ver'*
DECnet–RSX	*node::'dev:[ufd]filename.typ;ver'*
DECnet–IAS	*node::'dev:[ufd]filename.typ;ver'*
DECnet/E	*node::'dev:[ppn]filename.typ'*
DECnet-10	*node::'dev:[ufd]filename.ext[p,pn]<prot'*
DECnet-20	*node::'dev:<directory>filename.typ.gen;att'*
DECnet–RT	*node::'dev:filename.typ'*
DECnet–DOS	*node::'dev:\path\filename.typ'*
DECnet–ULTRIX	*node::'path/filename'*

In Table 4–1, all the information following the node name is enclosed in single quotation marks (' '). The quotation marks prevent the local shell (or command interpreter) from reading and interpreting certain special characters (or metacharacters) which are included in the specification. Only the remote system reads and interprets characters enclosed in quotes.

Metacharacters represent special commands or functions to the local shell. Here are the ULTRIX metacharacters:

Table 4–2: ULTRIX Metacharacters

Character	Meaning
< >	Angle brackets
&	Ampersand
*	Asterisk
\	Backslash
|	Bar
{ }	Braces
	Blank space
$	Dollar sign
!	Exclamation point
( )	Parentheses
?	Question mark
;	Semicolon
[ ]	Square brackets

You must do something to ensure that the metacharacters you include in file specifications are not interpreted by the local shell. You can (as shown in the table) choose to enclose all the information following the node name in single quotes as a standard part of your file specification. The advantage of this choice is that you do not have to remember which characters require special treatment. Of course, you can also choose to treat the special characters individually by enclosing them in single quotes or preceding each one with a backslash.

The following example shows several ways you can compose a command line that includes metacharacters (square brackets and an asterisk):

```
% dcat woods::'usrdsk2:[nature]trees.txt;*'
```

or

```
% dcat woods::usrdsk2:\[nature\]trees.txt;\*
```

or

```
% dcat woods::usrdsk2:'[nature]'trees.txt;'*'
```

4.2 Using Wildcard Characters

All Digital operating systems support the use of wildcard characters. You can use them in file specifications to refer to a group of files by a general name, rather than specifying each file individually. Table 4-3 describes the ULTRIX-specific wildcard characters.

Table 4–3: Wildcard Characters

Character	Meaning
*	Matches one or more characters, except a slash (/).
?	Matches a single character, except a slash (/).
[set]	Matches one character from a set. Set members can be enumerated (for example, [1234]) and/or contain ranges (for example, [1-4]) where the ASCII order is used.
{tokens}	Matches a string from a list of ASCII strings, separated by commas, which can be a portion of the filename. For example, {abc,def}* will match any **file name** that begins with abc or def. A token can contain other wildcard characters.

These wildcard characters are among the metacharacters listed in Table 4-2. If you include a wildcard in a file specification, enclose it in single quotes or precede the wildcard with a backslash (\). Otherwise, the local shell will read and interpret the wildcard character.

This example shows two ways you can compose a command line that includes the asterisk (*) wildcard character:

```
% dcat fact::'dev233:*.txt' RET
```

or

```
% dcat fact::dev233:\*.txt RET
```

Table 4–3 describes how the ULTRIX system interprets wildcard characters. See the documentation supplied with the remote system for information on how that system interprets wildcard characters.

4.3 Supplying Access-Control Information

DECnet nodes can use access-control information to screen connection requests from remote nodes. A node screens connection requests by checking this user-supplied information against information in local password or proxy files.

Access-control information consists of a **user name**, **password** (optional), and **account** (optional). You can specify this information in the following ways:

- Enter the access-control information manually.
- Use an alias.
- Use a proxy account.

The target system handles access-control information as follows:

- When you supply access-control information to a remote system, the remote system checks its password file to verify the user name and password you gave it.
- When you supply only a user name, the remote system checks its proxy file. If the user name is not defined in the proxy file, the remote system checks if the user name belongs to an account that has no password.
- When you omit all access-control information, the remote system checks to see if you are defined in its proxy file. If not, the system then tries to use its default access account.

NOTE

The ULTRIX system is case sensitive. Non-ULTRIX systems may pass access-control information in uppercase. Therefore, if you plan to pass access-control information from a non-ULTRIX to an ULTRIX system, consult the other system's documentation to see whether or not the system provides a **case-sensitive** means for passing this information. If it does not, add an uppercase account name and password to your ULTRIX system password file.

The following systems pass access-control information in uppercase:

- DECnet–IAS
- DECnet–RT
- DECnet-11M V4.0 or earlier
- DECnet-11M-PLUS V2.0 or earlier

4.3.1 Entering the Information Manually

You can manually specify access-control information following the node name in the file specification. Use this format:

node/username/password/account::file_information

where

node	is the name or address of the DECnet node.
username	is a string of up to 39 characters identifying the user's log-in name, which is verified by the remote node.
password	is a string of up to 39 characters needed for gaining access to the remote system.
account	is a string of up to 39 characters that is verified by the remote node's system account file. (The *account* field is ignored by most DECnet systems.)
file_information	is the rest of the file specification as specified in Table 4–1.

4.3.1.1 User Name and Password

When you include the user name and password in the file specification, the remote node can verify access clearance and then execute the command. This example does not include file information because ATLANT/jones indicates the login directory to be listed by the **dls** command:

```
% dls atlant/jones/mysecret:: RET

Directory ATLANT::/usr/users/jones/
MYFILE              bin                 printf.notes        filen
log                 log2
%
```

4.3.1.2 User Name Only (Password Required)

If you include only the user name in the file specification, DECnet–ULTRIX prompts you for the password. The advantage of letting the system prompt you for the password is that your password is not displayed on your screen as you type it, thus providing additional security.

In the following example, a file in account **don** on node RED is copied to account **sue** on node BLUE. The system prompts for passwords to both accounts:

```
% dcp red/don::'dsk3:[don]num.dat' blue/sue::'u$2:[sue.data]02.dat' RET
Password for red/don:: ? subway RET  (not echoed)
Password for blue/sue:: ? underground RET  (not echoed)
%
```

4.3.1.3 User Name Only (Password Not Required)

A password is not required along with the user name if the account does not have a password or if a proxy account exists. However, to avoid being prompted for a password, type a slash (/) immediately after the user name. This indicates that you are not specifying a password, so the system does not prompt you for one. For example:

```
% dls atlant/public/:: RET

Directory ATLANT::/usr2/users/public/
README              bin                 printf.notes        filen
logfile1            logfile2
%
```

If you omit the slash from the command line and the system prompts you for a password, simply press RET at the password prompt. For example:

```
% dls atlant/public:: RET
Password for ATLANT/public:: ? RET

Directory ATLANT::/usr2/users/public/
README              bin                 printf.notes        filen
logfile1            logfile2
%
```

4.3.2 Using an Alias

As a shortcut to typing the node ID and access-control information, you can specify an "alias" node name. An alias node name is an alphanumeric string of one or more characters that you type in place of a node ID and any access-control information defined for it.

For example, you could have an alias "boo" that stands for user **tom** with password **secrets** on node BOSTON. With an alias like "boo," you do not have to specify any access-control information on the command line.

To use an alias in a command line, type the alias followed by the double colon (::) and the file specification. For example:

```
% dcat boo::'userdsk:[tom]memo.txt;3' RET
```

You can define an alias node name for any node on your network by creating a **.nodes** file in your home directory. Use the following format for your entries in the **.nodes** file:

alias=node-id[/login-name[/password]]

You cannot use spaces or tab characters in any of the fields. The following example shows sample **.nodes** file entries:

```
b=boston
w=boston/root/xyzkoroijt
boo=boston/tom/secrets
```

NOTE

Set up the protections on your **.nodes** file so that only the owner can read the file or write to it. Do this to prevent unauthorized access to your passwords.

4.3.3 Using Proxy Access

Using proxy access is another way to access a remote node without supplying access-control information. Although DECnet–ULTRIX systems support proxy access, not all DECnet systems do. Check with the manager of any non-ULTRIX system to find out if it does.

Before you can use proxy access, the system manager for the remote node must set up a proxy account for you. If you are going to have access to more than one proxy account from the same node and log-in name, indicate which proxy account is the default.

To use your default proxy account, enter the command without any access-control information. For example:

```
% dcat kokomo::farms.dat RET
```

To use an account other than your default proxy account, append the account login name to the node followed by a slash (/). The slash indicates that you are not supplying a password. For example:

```
% dcat kokomo/henry/::farms.dat RET
```

The *DECnet–ULTRIX Network Management* manual contains a full discussion of proxy access and instructions for defining proxy file entries.

Chapter 5

Working with Files on Remote DECnet Nodes

This chapter tells you how to:

- View directories on remote DECnet nodes
- Display and concatenate files from remote DECnet nodes
- Copy files to and from remote DECnet nodes
- Convert file names during file transfer
- Delete files on remote DECnet nodes

This chapter introduces the four DECnet–ULTRIX commands (**dls**, **dcat**, **dcp**, and **drm**) that you use to work with files on remote DECnet systems. Each command is illustrated with examples. For more information about each command, see Chapter 6 or the on-line manual pages.

Note that you can pipe or redirect input and outputfrom the **dls**, **dcat**, and **dcp** commands, according to standard ULTRIX conventions.

NOTE

Each sample command line in this chapter shows access-control information. Whenever you use the commands described in this chapter, you have to specify access-control information, unless you meet one of these criteria:

- You have defined an alias for the remote node.
- You have a proxy account on the remote node.

See Section 4.3 for a discussion about access-control information.

5.1 Viewing Remote Directories

The **dls** command displays the directories of a remote ULTRIX or non-ULTRIX node. The command displays the output on your terminal screen by default.

The syntax for the **dls** command is as follows:

dls [*options...*] *filespec* RET

For a complete description of the **dls** command, see Chapter 6.

5.1.1 On ULTRIX Remote Nodes

The following example displays **brightstar's** home directory on the remote DECnet–ULTRIX node BRAGG. The password is **starlet**.

The command line includes two options, **-a** and **-l**. The **-a** option causes all the files in the directory to be listed, including those whose names begin with a period. The **-l** option produces the long format, which includes the directory's **protection level**, creation date (or last modified date, for ULTRIX systems), size, and owner.

```
% dls -a -l bragg/brightstar/starlet:: RET

Directory WBRAGG::/usr/users/brightstar/
.                drwxrwxr-x   03-AUG-89 12:54:14      512   brightstar
..               drwxr-xr-x   29-JUL-89 15:09:39     2048   root
.cshrc           -rwxr-xr-x   05-MAY-89 19:03:53      281   brightstar
.forward         -rw-r--r--   05-MAY-89 20:12:49       16   brightstar
.login           -rwxr-xr-x   05-MAY-89 19:03:54      234   brightstar
.profile         -rwxr-xr-x   05-MAY-89 19:03:53      138   brightstar
log              -rw-r--r--   03-AUG-89 12:43:09      311   brightstar
text             -rw-r--r--   02-AUG-89 10:25:30      442   brightstar

8 files in 1 directory
%
```

5.1.2 On Non-ULTRIX Remote Nodes

The following example lists the files in directory [MANGO] on disk DRA2: on the DECnet–VAX node DAVIS. The access-control information includes the user name **s_wolf** and the password **quirk**.

```
% dls davis/s_wolf/quirk::'dra2:[mango]' RET

Directory DAVIS::DRA2:[MANGO]
FRUITY.TXT;2          FRUITS.LIS;6              TROPICAL.EXE;1
%
```

5.2 Displaying and Concatenating Remote Files

The **dcat** command displays the contents of a remote file and (by default) directs the output to your terminal screen, and also concatenates the contents of more than one file, following standard ULTRIX conventions.

The syntax for the **dcat** command is as follows:

dcat [*options...*] *filespec...* RET

For a complete description of the **dcat** command, see Chapter 6.

5.2.1 Displaying Remote Files on the Screen

You can use the **dcat** command to display remote files on your screen. In the following example, the command displays the contents of the file FRUITS.LIS in directory [MANGO] on disk DRA2: on the remote DECnet–VAX node DAVIS. The access-control information includes the user name **s_wolf** and the password **quirk**.

```
% dcat davis/s_wolf/quirk::'dra2:[mango]fruits.lis' RET
```

5.2.2 Concatenating Remote Files

You can also use the **dcat** command to concatenate multiple files. In the following example, all the files with the extension .TXT from the directory [ADOBE] on the remote DECnet–VAX node NAVAHO are concatenated and redirected into the local file **textfiles**. The access-control information includes the user name SANDY and the password BEACH.

```
% dcat navaho/sandy/beach::'[adobe]*.txt' > textfiles RET
```

5.3 Copying Files Between Systems

The **dcp** command lets you copy ASCII text and binary image files to and from remote DECnet nodes. You can copy files:

- From local to remote node
- From remote to local node
- From remote to remote node

The syntax for the **dcp** command is as follows:

dcp [*options...*] *input output* RET

For a complete description of the **dcp** command, see Chapter 6.

If you copy non-ULTRIX files to an ULTRIX system, you lose the non-ULTRIX attributes associated with those files.

5.3.1 From Local to Remote Node

In the following example, the command copies the local file **renee** to a new file RENEE.LIS on the remote VMS node WOODS in the directory [PETS.RACCOON]. The access-control information includes the user name TOMAS and the password TOM.

```
% dcp renee woods/tomas/tom::'[pets.raccoon]renee.lis' RET
```

5.3.2 From Remote to Local Node

The following command copies the file FARM.LIS from the remote DECnet–VAX node DAVIS to the file **farm** on the local DECnet–ULTRIX node. The access-control information includes the user name EVELYN and the password SECRET.

```
% dcp davis/evelyn/secret::'dra2:[mango]farm.lis' farm RET
```

The following command uses the **-i** option to copy, in image file mode, all the data files with the extension .DAT from the directory [HERO.HELIX.DATA] of HERO's account on the remote VMS node ONYX to the local directory **/usr/src/data**. The access-control information includes the user name HERO and the password MAGIC.

```
% dcp -i onyx/hero/magic::'[hero.helix.data]*.dat' /usr/src/data RET
```

The following command uses the **-r** option to copy all of the files in **~uucp** on the remote ULTRIX node KIMONO to the local directory. The access-control information includes the user name **larry** and the password **newcar**.

```
% dcp -r kimono/larry/newcar::~uucp . RET
```

5.3.3 From Remote to Remote Node

The following command copies the file [DON]NUM.TXT to the file [SUE.DATA]02.TXT. There is no access-control information in this example, because the aliases **red** and **blue** replace the full access-control information string. (For more information on using an alias, see Section 4.3.2.)

```
% dcp red::'user$55:[don]num.txt' blue::'user$22:[sue.data]02.txt' RET
```

5.4 Converting File Names During File Transfer

Non-ULTRIX DECnet systems do not follow the ULTRIX file-naming scheme. Therefore, when you copy a file from non-ULTRIX remote nodes to ULTRIX nodes, you may need to convert the file name. You can do this by using the **dcp -c** command option. Another way to solve this problem is to explicitly specify, on the command line, a name for the destination file.

By default, when you transfer a file, the file name changes (uppercase characters convert to lowercase, and the version number disappears) to match ULTRIX file-naming conventions. In the following example, the remote VMS file COSTS.TXT;3 is copied to the current directory on the local ULTRIX node. The access-control information is user name ANNA and password TOURIST:

```
% dcp venice/anna/tourist::'toni:[boats]costs.txt;3' . RET
```

This file appears as **costs.txt** in the local directory.

In the following example, the file name is not automatically converted during transfer because an output file name, **costs**, is specified in the command line:

```
% dcp venice/anna/tourist::'toni:[boats]costs.txt;3' costs RET
```

5.4.1 dcp -c Option Flags

The **-c** option flags control the format of the converted file output. By default, whenever you use the **dcp** command, the **ultrix** flags are in effect: **lower**, **nodollar**, **nosemicolon**, and **noversion**.

The value for each flag is defined as follows:

ultrix (default)	sets all **-c** flags (**lower**, **nodollar**, **nosemicolon**, and **noversion**)
lower	converts uppercase to lowercase
nodollar	converts '$' to underscore '_'
nosemicolon	converts ';' to '.'
noversion	strips off version numbers

Each flag also has a corresponding negative value:

none	clears all **-c** flags
nolower	does not convert uppercase to lowercase

dollar	does not convert '$' to underscore '_'
semicolon	does not convert ';' to '.'
version	does not strip off version numbers

NOTE

The **nosemicolon** flag is redundant when paired with the **noversion** flag. For example, if you specify **nosemicolon** and **noversion** when you transfer the file COSTS.TXT;3 to your local system, the semicolon is changed to a period, but that period is stripped off along with the version number.

5.4.2 Setting -c Option Flags

Use the **-c** option flags to change the way file names are converted during file transfer. Follow these guidelines:

- You can replace the **ultrix** default flags by adding the **setenv DCP_CNAMES** command to your **.login** file. For example:

```
setenv DCP_CNAMES nodollar,noversion
```

NOTE

Because the ULTRIX system is case sensitive, you must enter **DCP_CNAMES** in uppercase.

- You can add to the default (or **DCP_CNAMES**) flags by including **-c** option flags in the command line. For example:

```
% dcp -cnolower woods::'[pets.racoon]renee.lis;22' . RET
```

Note that the alias **woods** is used in this example in place of a complete access-control information string.

5.5 Deleting Remote Files

The **drm** command lets you delete a single file, multiple files, or an entire directory of files from a remote DECnet node. Of course, you need the appropriate access rights to delete any remote files and directories you specify.

The syntax for the **drm** command is as follows:

drm [*options...*] *filespec* RET

For a complete description of the **drm** command, see Chapter 6.

5.5.1 A Single File

In the following example, the **drm** command deletes the file FARM.LIS from the remote DECnet–VAX node DAVIS. The access-control information includes the user name S_WOLF and the password QUIRK:

```
% drm davis/s_wolf/quirk::'dra2:[mango]farm.lis' RET
```

5.5.2 Multiple Files

The **-r** option (recursive delete) for **drm** deletes all of the files and subdirectories from the directory **/usr/keith** on the remote DECnet–ULTRIX node IAMOK. The access-control information includes the user name **keith** and the password **partridge**. For example:

```
% drm -r iamok/keith/partridge::/usr/keith RET
```

5.5.3 All Files of a Single File Type

The following example deletes all the files with the .RNO extension from user WHITE'S account on the remote DECnet–VAX node ONYX. Because the files are in the home directory, no directory is specified.

```
% drm onyx/white::'*.rno' RET
Password for onyx/white:: ? snow RET  (not echoed)
%
```

Chapter 6

DECnet–ULTRIX Command Summary

This section describes each DECnet–ULTRIX user command in detail. The commands appear in alphabetical order and follow the graphic conventions set down in the Preface. Table 6–1 summarizes the functions of the DECnet–ULTRIX user commands.

Table 6–1: DECnet–ULTRIX Command Functions

Command	Function
dcat	Types the contents of remote files.
dcp	Copies files between DECnet systems.
dlogin	Provides a virtual terminal connection to remote DECnet nodes.
dls	Lists the contents of a remote directory.
drm	Deletes remote files.
mail	Sends messages and files to remote DECnet users.

Note that the command descriptions do not discuss error messages. For a complete list of all possible error messages and a description of each message, see Appendix A.

The DECnet–ULTRIX command descriptions also appear on-line in **dcat(1dn)**, **dcp(1dn)**, **dlogin(1dn)**, **dls(1dn)**, and **drm(1dn)**. In addition, all error messages are described in **errors(1dn)**.

dcat(1dn)

NAME

dcat — type the contents of remote files

SYNTAX

dcat [**-v**] *filespec...*

where

-v	logs the names of the files being typed to standard error.
filespec	is a file specification for one or more remote files. The format for a file specification varies with each Digital operating system. You can specify wildcard characters. If you want the target node instead of the local shell to interpret a string of wildcard characters, you must enclose the string in quotation marks. See the *DECnet–ULTRIX Use* manual for file specifications and wildcard characters.

DESCRIPTION

The **dcat** command reads remote files and displays them on the standard output. The command displays the files in the order that you list them.

EXAMPLES

The following example displays, to standard output, the contents of the file FRUITS.LIS in the directory [MANGO] on disk DRA2: on the remote DECnet–VAX node DAVIS. Note that no access control information is given, indicating one of these possibilities: the file is world-readable, you defined an alias for the remote node, or you have a proxy account on the remote node.

```
% dcat davis::'dra2:[mango]fruits.lis'
```

The following command redirects all the files with the extension .TXT from the directory [ADOBE] on the remote DECnet–VAX node NAVAHO into the local file **lefile.txt**. The string **/adobe/secret** is the access-control information that NAVAHO uses to verify remote access.

```
% dcat navaho/adobe/secret::'[adobe]*.txt' > lefile.txt
```

SEE ALSO

errors(1dn)

dcp(1dn)

NAME

dcp — copy files between DECnet nodes

SYNTAX

dcp [*options...*] *input output*

where

-A appends the input file or files to a specified output file. Note that the output file must already exist; this option does not create an output file.

-P prints the files at the default printer on the remote system.

-S submits remote output files for execution. On ULTRIX systems, the **-S** option submits output files to the shell and creates a log file in the log-in directory that has the name *filename*.log, where *filename* is the name of the specified output file.

-a copies files in ASCII record mode. ASCII mode transfers perform the necessary format conversions between heterogeneous systems. ASCII mode is the default when you copy to and from non-ULTRIX systems.

-c converts the input file name from a non-ULTRIX system to a name with an ULTRIX format. This conversion happens by default whenever you use **dcp** to copy a file from a non-ULTRIX system (specifically, a VMS, RSX, or MS-DOS system) to an ULTRIX system.

Using the -c option, you can also specify one or more of the following flags to customize how file names are converted:

ultrix (default)	sets all **-c** flags (**lower**, **nodollar**, **nosemicolon**, and **noversion**)
lower	converts uppercase to lowercase
nodollar	converts $ to underscore _
nosemicolon	converts ; to .
noversion	strips off version numbers
none	clears all **-c** flags
nolower	does not convert uppercase to lowercase
dollar	does not convert $ to underscore _
semicolon	does not convert ; to .
version	does not strip off version numbers

The **-c** option has no effect unless the output file is in a local directory; you cannot use this option when copying files to remote directories.

-i copies files in image mode. This option is useful for copying nonprintable data files between homogeneous systems. Image mode transfers are generally faster than ASCII mode transfers but do not perform data format conversions between heterogeneous systems. Image mode is the default when you copy between ULTRIX systems.

-r copies all of the files in a directory. Also copies all subdirectories. The input and output names you specify must be directory names. Note that the top directory to which you are copying must already exist; **dcp -r** does not create it. However, this option does create all the subdirectories if they do not already exist on the node to which you are copying files. This option is valid only between DECnet–ULTRIX systems.

-v logs the names of the files being copied to standard error.

input is one or more input files or directory specifications. The format for an input file specification varies with the operating system on which the file is located. If there are multiple input files, separate each with a blank space.

You can specify a dash (–) in place of an input file specification, directing **dcp** to read from standard input until it reaches end-of-file (EOF).

You can specify wildcard characters. If you want the target node, instead of the local shell, to interpret a string of wildcard characters, you must enclose the string in quotation marks. See the *DECnet–ULTRIX Use* manual for more information.

output is the output file or directory to which **dcp** copies the input files. The format for an output file or directory specification varies with the operating system on which the output file is created. See the *DECnet–ULTRIX Use* manual for a description of all DECnet file specifications and wildcard characters.

When you copy input files to a directory, the output files retain the input file names and syntax unless you use the **-c** option. However, if you are copying a non-ULTRIX file to an ULTRIX system, the file name is automatically converted to match ULTRIX file-naming conventions.

You can use a dash (–) in place of the output file specification to direct the files to standard output.

DESCRIPTION

The **dcp** command copies files between DECnet nodes. You can copy both ASCII text and binary image files. Note that non-ULTRIX files with additional attributes lose those attributes when copied to an ULTRIX system.

When you use **dcp** to copy a file to another DECnet–ULTRIX system, you need not specify a mode of transfer because image mode is the default transfer mode between ULTRIX systems. For non-ULTRIX systems, you need to specify a mode of transfer only for image files.

File protection modes are preserved when you copy files between DECnet–ULTRIX systems. With non-DECnet–ULTRIX systems, the output file protection modes are defined by the remote system's file protection defaults.

When you copy files from a non-ULTRIX system to an ULTRIX directory without using the file-name conversion option, the output file name retains the format of the input file name. (By default, however, a file copied from a VMS system is automatically converted to ULTRIX file-naming conventions.) In other cases, you can use the **-c** option to convert file names.

RESTRICTION

You cannot use the **-c** option to convert file names when copying files between ULTRIX systems.

EXAMPLES

The following command copies the local file **farm.3** to the directory [MANGO] on device DRA2: on the remote DECnet–VAX node DAVIS; the command names the new file FARM.LIS. The access control information is **/mango/fruity**.

```
% dcp farm.3 davis/mango/fruity::'dra2:[mango]farm.lis'
```

The following command copies the file FARM.LIS from the remote DECnet–VAX node DAVIS to the local DECnet–ULTRIX node. By default, the file name will be converted to match ULTRIX file-naming conventions (lowercase, without dollar signs, semicolons, or version numbers).

```
% dcp davis::'dra2:[mango]farm.lis' .
```

The following command also copies the file FARM.LIS from the remote DECnet–VAX node DAVIS to the local DECnet–ULTRIX node. However, the output file name is not converted to lowercase because the user specifies the **-cnolower** flag.

```
% dcp -cnolower davis::'dra2:[mango]farm.lis' .
```

SEE ALSO

errors(1dn)

dlogin(1dn)

NAME

dlogin — log on to remote DECnet nodes

SYNTAX

dlogin *node* [**-e***c*] [**-l** *logfile*]

where

node	is the DECnet node name or DECnet node address of the remote node to which you are connecting.
-e*c*	specifies an escape character for interrupting your remote session and temporarily returning control to your local node. The variable *c* is the character you define; any character is valid. The default escape character is the tilde (~).
-l *logfile*	logs your **dlogin** session to the file specified by the variable *logfile*.

DESCRIPTION

The **dlogin** utility lets you log on to remote DECnet nodes and access the resources of these operating systems.

This utility uses a protocol called the Digital Network Architecture (DNA) command terminal protocol (**CTERM**). With **dlogin**, you can connect to any DECnet node that supports CTERM. DECnet–ULTRIX nodes support both incoming and outgoing virtual terminal connections.

The **dlogin** utility lets you execute commands on your local node after you have started a remote session on another node. Whenever you want, you can switch back and forth between your local session and your **dlogin** session with the **dlogin** escape character.

To temporarily return control to your local node, at the remote system's prompt, type the **dlogin** escape character followed by RET. The default escape character is the tilde (~). You get a **local command>** prompt. At the **local command>** prompt, you can execute commands on your local node. To resume your **dlogin** session on the remote node, press RET at the **local command>** prompt.

To use the escape character without entering **local command mode**, type the escape character once, followed by any character except RET. You can also type the escape character twice, and it echoes to your screen once. For example, to send the **cd** ~ command to the remote system, type **cd** ~ ~ and press RET.

The **dlogin** utility offers help. For help on special commands that control your **dlogin** session, type a question mark (?) at the **local command>** prompt. The **dlogin** menu is displayed.

To end your **dlogin** session, type the remote node's logout command. To end a **dlogin** session after you have connected to a remote node but before you have logged on, type ~ RET and then issue the **exit** command.

RESTRICTION

CTERM is not supported on VMS versions before V4.0 or on DECnet–11M–PLUS versions before V3.0.

EXAMPLE

In the following example, user Irene logs on to the remote VMS node BACON and issues the SHOW USERS command:

```
% dlogin bacon

Username: IRENE
Password: GOODNIGHT     (not echoed)

                Welcome to VAX/VMS V5.2 on node BACON

$ SHOW USERS
```

SEE ALSO

errors(1dn)

dls(1dn)

NAME

dls — list the contents of a remote directory

SYNTAX

dls [**-1**] [**-C**] [**-a**] [**-l**] *filespec*

where

-1	formats the listing in a single-column format, which is the default when the standard output is not a terminal. The **-1** option is ignored if you use it with the **-l** option.
-C	formats the listing in a multicolumn format, which is the default when the standard output is a terminal. The **-C** option is ignored if you use it with the **-l** option.
-a	lists all the files in a remote ULTRIX directory, including the names that begin with a period (.). If you omit the **-a** option, file names that begin with a period are not listed.
-l	produces a list in long format. For each file specification, the **-l** option lists the file name, the protection settings, the creation date (or last modified date for ULTRIX systems), and the size in bytes.
filespec	is a file specification for a remote directory or file. The format for a file specification varies with each operating system. You can specify wildcard characters. If you want the target node instead of the local shell to interpret a string of wildcard characters, you must enclose the string in quotation marks. See the *DECnet–ULTRIX Use* manual for more information about file specifications and wildcard characters.

DESCRIPTION

The **dls** command lists all the file names in the specified remote directory or lists individual files. If you do not specify a directory or file, **dls** uses the directory name indicated by the access-control information.

All the file names in the directory are listed, unless you specify individual files. For each file name, **dls** repeats the name and any other information you request, for example, protection settings or creation dates.

Output from **dls -l** has designated characters that represent the protection setting. Displays from ULTRIX systems have 10 such characters, and **dls -l** displays from non-ULTRIX DECnet systems have 12 of these characters; for example:

```
DECnet-ULTRIX Node      DECnet-VAX Node
-rwxrw-r--              ---rwxrwxrwx
```

For ULTRIX systems, the first character can be any of the following:

d indicates a directory

b indicates a block-type special file

c indicates a character-type special file

s indicates a socket

- indicates a regular file

The next nine characters represent the protection levels for the file's owner, group, and other (world), in that order, consisting of three characters each. Within each level, the three modes are represented by the characters **r**, **w**, and **x**, which are defined as follows (for a directory, execute permission implies permission to search the directory):

r indicates read permission

w indicates write permission

x indicates execute permission

– indicates that no permission has been set

The group-execute permission character is **s** if a file has the set-group-id bit set. Likewise, the user-execute permission character is **s** if a file has the set-user-id bit set.

The last character of an ULTRIX protection setting (normally **x** or **–**) is **t** if the sticky bit of the file mode is on. See the description of **chmod(1)** in the *ULTRIX Reference Pages* for the meaning of this mode. Some non-ULTRIX systems use different mapping schemes for file protection. These systems map their file protection schemes into the syntax used by **dls**. For example, an MS-DOS system does not have the concept of protections.

EXAMPLES

In the following example, the user specifies the remote DECnet–ULTRIX node ATLANT, the remote user name **jones**, and the password **mysecret**. Because this information is included in the file specification, the remote node can verify access:

```
% dls atlant/jones/mysecret:: RET
```

Using the long format, the following command lists the file name TEST.DAT, its protection level, creation date (or last modified date for ULTRIX systems), size, and owner. TEST.DAT is located on the remote DECnet–RSX node NAVAHO; note that the standard ULTRIX use of **>** *filename* redirects the information to the file **info.tes** on the local node:

```
% dls -l navaho::'[312,42]test.dat' > info.tes
```

SEE ALSO

errors(1dn)

drm(1dn)

NAME

drm — delete remote files

SYNTAX

drm [**-r**] *filespec*

where

-r deletes all of the files from a directory and the directory itself.

filespec is a complete file specification for a remote file or directory. The format for a file specification varies with each Digital operating system.

You can specify wildcard characters. If you want the target node instead of the local shell to interpret a string of wildcard characters, you must enclose the string in quotation marks. See the *DECnet–ULTRIX Use* manual for more information about file specifications and wildcard characters.

DESCRIPTION

The **drm** command deletes one or more files from a remote system. The command can delete entire ULTRIX directories, but not non-ULTRIX directories.

RESTRICTION

If you specify the name of a directory, the **-r** option deletes all of the files in that directory, all of the files in all of the subdirectories, and both the specified directory and all existing subdirectories. This option is valid only between DECnet–ULTRIX systems.

EXAMPLES

This command deletes the file FARM.LIS from remote DECnet–VAX node DAVIS:

```
% drm davis::'dra2:[mango]farm.lis'
```

This command uses the **-r** option to delete all of the files in the directory **olddata** on the remote DECnet–ULTRIX node ATHENS. Note that if this command did not include a directory name, **drm** would use the access-control information **/george/seablue** and the effect would be to delete all of the files and all of the directories in account **george**:

```
% drm -r athens/george/seablue::olddata
```

SEE ALSO

errors(1dn)

mail

NAME

mail — send mail to DECnet users and receive mail from them

SYNTAX

mail *nodename*::*username*

mail *username*@*nodename*.dnet

where

nodename	is the name of the remote node where the user to whom you are sending mail resides.
username	is the name of the user to whom you are sending mail.
.dnet	is the Internet psuedodomain name.

DESCRIPTION

This command summary is not available on line. For on-line help, refer to **mail(1)** and **mailaddr(7)**.

The **mail** command lets you send and receive mail to and from remote DECnet users. All of the flags, commands, and rules associated with ULTRIX **mail** are valid.

You can choose DECnet addressing syntax (*nodename*::*username*) or Internet addressing syntax (*username*@*nodename*.dnet). Note that the Internet addressing syntax is provided merely as a convenient interface for Internet users. DECnet–ULTRIX actually converts this Internet address syntax to the DECnet syntax before sending the mail out.

SEE ALSO

mail(1), **mailaddr(7)**

Appendix A

Error Messages

This appendix describes all the possible error messages from **dcat**, **dcp**, **dlogin**, **dls**, and **drm** commands. Each error falls into one of the following categories:

- Connect errors
- File Access errors

Note that DECnet–ULTRIX error messages can also include ULTRIX errors and that descriptions of DECnet–ULTRIX error messages also appear on-line in **errors(1dn)**.

A.1 Connect Errors

Connect errors document failures to establish a DECnet network connection by either the local or the remote system.

Connect failed, Connection rejected by object

The network connection was rejected by the remote object.

Connect failed, Insufficient network resources

The network connection was rejected because of insufficient network resources on either the local or the remote node.

Connect failed, Unrecognized node name

The network connection could not be established because the local node does not know about the remote node.

Connect failed, Remote node shut down

The network connection could not be established because the remote node was shut down.

Connect failed, Unrecognized object

The remote node did not recognize the object. For more information, contact the remote node system manager.

Connect failed, Invalid object name format

The remote node did not understand the object name format of the connect request.

Connect failed, Object too busy

The network connection was rejected by the network partner because the remote object was too busy handling other clients.

Connect failed, Invalid node name format

The network connection could not be established because the format of the node name was incorrect. A node name is invalid if it contains illegal characters or is too long (node names can be up to 6 alphanumeric characters in length).

Connect failed, Local node is not on

The network connection could not be established because the network on the local node shut down.

Connect failed, Access control rejected

The network connection was rejected because the network partner could not successfully validate the access-control information it received. For example, you gave no proxy access and no default access exists.

Connect failed, No response from object

The network connection could not be established because the remote object did not respond. The remote object either responded too slowly or terminated abnormally.

Connect failed, Node unreachable

The network connection could not be established because no path currently exists to the remote node.

A.2 File-Access Errors

File-access errors document all error conditions other than DECnet connect errors.

Bad format DAP message received

An incompatibility in the Data Access Protocol (**DAP** version numbers or lower DECnet layers resulted in the losing and/or corrupting of the (DAP) message.

DAP error code *(macro:micro)*

where

macro	is the DAP macro error code
macro	is the DAP micro error code

This message appears when no other specific error message can be provided. For an explanation, see the DAP error code in **/usr/lib/dnet_shared/dap.errors**.

DAP message received out of order

An incompatibility in the DAP version numbers or lower DECnet layers resulted in losing and/or corrupting the DAP message.

Data type not supported

You attempted to copy a file whose data type is not supported by either the local or the remote system.

Device is write locked

You attempted to write to a file on a device that was write protected.

Device not in system

The device you specified is not known to the remote system.

Directory is full

The output file you specified cannot be created because there is no room available in the specified directory.

Error in file name

The file name you specified does not conform to the syntax of the remote system. See Appendix B for a definition of all DECnet file-name specifications.

File is locked

The file you are attempting to access is locked by the remote file system. This error can be caused by a remote system disallowing concurrent reading and writing of a file or by two users simultaneously attempting to write to the same file.

File organization not supported

You attempted to copy a file whose file organization was not sequential (DECnet–ULTRIX systems support only sequential files).

Link to partner broken

The DECnet network connection to the remote system was broken. This error can result when communication is no longer possible with the remote node.

Network operation failed

An operation to the network failed at the remote system.

Node name format error

The node name you specified was invalid; that is, the name contained illegal characters or was too long. See Appendix B for a definition of all DECnet node-name specifications.

No such device

The device you specified does not exist on the remote system.

No such file

The file you specified does not exist on the remote system.

Operation not supported locally

You attempted to perform an operation that is not supported by your local DECnet–ULTRIX system. For example, you cannot use **dls** to list a local directory.

Record attributes not supported

You attempted to copy a file whose record attributes are not supported by either the local or the remote system.

Record format not supported

You attempted to copy a file whose record format is not supported by either the local or the remote system.

Record too big for user's buffer

You attempted to copy a file that contained a record that is larger than **dcp**'s or the remote **fal**'s internal buffer. This error is frequently the result of an attempt to transfer a non-**ASCII** file in ASCII mode (which is the default transfer mode to non-ULTRIX systems).

Unspecified access error

An error occurred at the remote system in accessing a file. For an explanation, see the DAP error code in **/usr/lib/dnet_shared/dap.errors**.

Unsupported operation

The remote system does not support the operation you requested.

Appendix B

DECnet File Specifications

This appendix defines the syntax of file specifications for these DECnet systems:

- DECnet–ULTRIX
- DECnet–VAX
- DECnet–RSX
- DECnet–IAS
- DECnet/E
- DECnet-10
- DECnet-20
- DECnet–RT
- DECnet–DOS

The examples on the following pages show the different file specifications.

B.1 DECnet–ULTRIX File Specification

A DECnet–ULTRIX file specification has the following format:

node::path/filename

where

node	is the name or address of a DECnet–ULTRIX node.
path	is a list of directories, separated by slashes, that lead to the file name. – If *path* starts with a slash, this slash is the first character of the file specification, indicating that the path starts from the root file system. – If *path* starts with ~user, then ~user translates into the home directory of user on the remote ULTRIX system. – If the file specification does not begin with a slash, and no ~user part is specified, the file specification is relative to the home directory of the account through which access was granted.
filename	is an alphanumeric string of up to 255 characters that identifies a file.
*, ?, [ ], { }	A DECnet–ULTRIX file specification can also include any of these wildcard characters.
where	
*	matches zero or more characters anywhere within a file name.
?	matches exactly one character.
[set]	matches exactly one character from a set. Set members can be enumerated (for example, [1234]) and/or contain ranges (for example, [0-4]) where the ASCII order is used.
{tokens}	matches one string from a list of ASCII strings, separated by commas, which can be a portion of the file name. For example, {abc,def}* will match any file name that begins with abc or def. A token can contain other wildcard characters.

EXAMPLES

1. The following examples are valid ULTRIX file specifications:

```
nashua
Mail/inbox/34
/usr/etc/fal
~austin/balloons
*
program.[ch]
record-{beatles,turtles,coasters}=top[123]0.19??
```

2. The following command copies the file **spring** from one DECnet–ULTRIX node to another. The receiving node is JUNEAU. Notice that you need not enclose the path and file names in quotation marks when you copy files between DECnet–ULTRIX nodes.

```
% dcp spring juneau::~jones/flowers RET
```

3. This example also copies the file **flowers** from one DECnet–ULTRIX node to another, but includes access-control information **/jones/market** with the remote node name QUINCY. Because the full path name and file name are not spelled out, the home directory for account **jones** is used.

```
% dcp flowers quincy/jones/market:: RET
```

4. This command copies the file FLOWERS.TXT from a DECnet–VAX node to the file **flowers** on the DECnet–ULTRIX node BOSTON. Note that the access-control information (**"james haymarket"**) and the file name are inside quotation marks so that DECnet–VAX passes the information to BOSTON as typed.

```
$ COPY FLOWERS.TXT BOSTON"james haymarket"::"flowers" RET
```

B.2 DECnet–VAX File Specification

A DECnet–VAX file specification has the following format:

node::*dev*:[*directory*]*filename.typ;ver*

where

node	is the name or address of a DECnet–VAX node.
dev	is a device on the VMS system, such as DRB2: or USER$22:.
directory	is a directory name, such as [HART.MEMOS] or [.MEMOS].
filename	is a string that names the file. On VMS V3.x systems, *filename* is an alphanumeric string of up to 9 characters. On VMS V4.*x* and V5.0 systems, *filename* is a string of up to 39 characters. Valid characters are alphanumerics, the underscore (_), the dollar sign ($), and the dash (–).
typ	is a character string that identifies the file type. On VMS V3.*x* systems, *typ* is an alphanumeric string of up to 3 characters. On VMS V4.*x* systems, *typ* is a string of up to 39 characters. Valid characters are alphanumerics, the period (.), the dollar sign ($), and the dash (–).
ver	is the file version number. The version number is a decimal number between 1 and 32767. Multiple versions of a file can exist; the latest version of a file is the one with the highest version number.

EXAMPLES

1. The following command copies the file **rivers** from a DECnet–ULTRIX node to the DECnet–VAX node VENICE. The new copy is placed in the directory [OSCAR.WATER] and is named RIVERS.TXT. Notice that the VMS file specification is enclosed within quotation marks so that VENICE, instead of the local shell, interprets this specification.

   ```
   % dcp rivers venice::'dba0:[oscar.water]rivers.txt' RET
   ```

2. This example also copies the file **rivers** from a DECnet–ULTRIX node to the DECnet–VAX node VENICE, but includes access-control information **/oscar/boats** with the node name. Also notice that because the file name is not specified, the file is copied to Oscar's default directory.

   ```
   % dcp rivers venice/oscar/boats:: RET
   ```

3. This DECnet–VAX command copies the file FLOWERS.TXT from a VMS node to the file **flowers** on the DECnet–ULTRIX node UTICA. Note that the information within quotation marks is in lowercase so that DECnet passes this information to UTICA in lowercase.

   ```
   $ COPY FLOWERS.TXT UTICA"mark upstate"::"flowers" RET
   ```

B.3 DECnet–RSX and DECnet–IAS File Specifications

DECnet–RSX and DECnet–IAS file specifications have the following format:

node::*dev*:[*ufd*]*filename.typ;ver*

node	is the name or address of a DECnet–RSX or DECnet–IAS node.
dev	is a device on the remote system, such as DB1:.
ufd	is the user file directory, which is an octal user identification code (uic), such as [312,42] or a named directory. The uic can range from [1,1] to [377,377]. RSX-11M-PLUS and Micro/RSX systems also support named directories such as [SNOW] and [SMITH].
filename	is an alphanumeric string of up to 9 characters that names the file.
typ	is an alphanumeric string of up to 3 characters that identifies the file type.
ver	is the file version number. The version number is a decimal number between 1 and 32767. Multiple versions of a file can exist; the latest version of a file is the one with the highest version number.

EXAMPLES

1. This command copies the file **pacific** from a DECnet–ULTRIX node to the DECnet–RSX node RAPA. The DECnet–RSX file specification is enclosed within quotation marks so that RAPA, instead of the local shell, interprets this specification.

   ```
   % dcp pacific rapa::'db10:[50,377]pacific.txt' RET
   ```

2. This example also copies the file **pacific** from a DECnet–ULTRIX node to the DECnet–RSX node RAPA, but includes access-control information **/50,377/island** with the node name. Because the file name is not specified, the file is copied to the default directory for user 50,377.

   ```
   % dcp pacific rapa/50,377/island:: RET
   ```

3. This command copies the file FLOWERS.TXT from a DECnet–RSX node to **/usr/tmp/flowers** on the DECnet–ULTRIX node UTICA. Note that the information within quotation marks is typed in lowercase so that DECnet passes this information to UTICA in lowercase.

   ```
   $ NFT UTICA"mark research"::"/usr/tmp/flowers" = FLOWERS.TXT RET
   ```

B.4 DECnet/E File Specification

A DECnet/E file specification has the following format:

node::*dev*:[*ppn*]*filename.typ*

where

node	is the name or address of a DECnet/E node. You can append optional access-control information to the node name.
dev	is a device on the RSTS system, such as DK1:.
ppn	is a project programmer number, such as [314,42].
filename	is an alphanumeric string of up to 6 characters that names the file.
typ	is an alphanumeric string of up to 3 characters that identifies the file type.

EXAMPLES

1. The following command copies the file **flowers** from a DECnet–ULTRIX node to the DECnet/E node DAVIS. The DECnet/E file specification is enclosed within quotation marks so that DAVIS, instead of the local shell, interprets this specification.

```
% dcp flowers davis::'dm0:[50,25]test1.txt' RET
```

2. This example also copies the file **flowers** from a DECnet–ULTRIX node to the DECnet/E node DAVIS, but includes access-control information **/50,25/test** with the node name. Because the output file specification is not specified, the file is copied to the default directory for user 50,25.

```
% dcp flowers davis/50,25/test:: RET
```

3. This DECnet/E command copies the file TEST1.TXT from a DECnet/E node to **/users/jones/flowers** on the DECnet–ULTRIX node BOSTON. Note that the information within quotation marks is in lowercase type so that DECnet passes this information to BOSTON in lowercase.

```
$ COPY TEST1.TXT BOSTON"jones secret"::"~jones/flowers" RET
```

B.5 DECnet-10 File Specification

The file specification for a TOPS-10 operating system has the following format:

node::*dev*:[*ufd*] *filename.ext*[*p,pn*]<*prot*>

A complete file specification in an ANF-10 network has the following format:

node dev:*filename.ext*[*p,pn*]<*prot*>

or

logical name:*filename.ext*[*p,pn*]<*prot*>)

where

node	is the name or address of a DECnet-10 node. You can append optional access-control information to the node name.
node_dev	is a node name in the ANF-10 network, such as 1024_DSK0:.
dev	is a device on the TOPS-10 system, such as DSKC:.
ufd	is the user file directory.
filename	is an alphanumeric string of up to 6 characters that names the file.
ext	is an alphanumeric string of up to 3 characters, which is the file-name extension.
p,pn	is a project programmer number, such as [27,5117].
prot	is the file-access protection, which consists of up to 3 octal digits.

EXAMPLES

1. The following command copies the file **flowers** from a DECnet–ULTRIX node to a DECnet-10 node. The DECnet-10 file specification must be enclosed within quotation marks.

```
% dcp flowers atlant::'dba0:flowers.txt[52,879]' RET
```

2. This example also copies the file **flowers** from a DECnet–ULTRIX node to a DECnet-10 node, but includes access-control information **/52,879/secret** with the node name. Because the file name is not specified, the file is copied to the default directory for user 52,879.

```
% dcp flowers davis/52,879/secret:: RET
```

B.6 DECnet-20 File Specification

The file specification for a TOPS-20 operating system has the following format:

node::dev:<directory>filename.typ.gen;att

where

node	is the name or address of a DECnet-20 node.
dev	is a device on the TOPS-20 system, usually a file structure, such as SNARK:.
directory	is a directory name, such as <LONDONTOWN>.
filename	is an alphanumeric string of up to 39 characters that names the file.
typ	is an alphanumeric string of up to 39 characters that identifies the file type.
gen	is a generation or file version number.
att	is a file-access attribute.

EXAMPLES

1. The following command copies the file **snark** from a DECnet–ULTRIX node to the DECnet-20 node LONDON. The DECnet-20 file specification is enclosed within quotation marks so that LONDON, instead of the local shell, interprets this specification.

```
% dcp snark london::'dba0:<judy>garden.dat' RET
```

2. This example also copies the file **snark** from a DECnet–ULTRIX node to the DECnet-20 node LONDON, but includes access-control information **/judy/secret** with the node name. Because the output file is not specified, the file is copied to Judy's default directory.

```
% dcp snark london/judy/secret:: RET
```

B.7 DECnet-RT File Specification

An RT-11 file specification has the following format:

node::dev:filename.typ

where

node	is the name or address of a DECnet-RT node. You can append optional access-control information to the node name.
dev	is a device name on an RT-11 system, such as RK0:.
filename	is an alphanumeric string of up to 6 characters that names the file.
typ	is an alphanumeric string of up to 3 characters that identifies the file type.

EXAMPLES

1. The following command copies the file **daisy** from a DECnet–ULTRIX node to the DECnet-RT node RAGES. The DECnet-RT file specification is enclosed within quotation marks so that RAGES, instead of the local shell, interprets this specification.

```
% dcp daisy rages::rk0:test1.txt RET
```

2. This command also copies the file **daisy** from a DECnet–ULTRIX node to the DECnet-RT node RAGES, but specifies the access-control information **/sam/sam** for Sam's account.

```
% dcp daisy rages/sam/sam::test1.txt RET
```

B.8 DECnet–DOS File Specifications

A DECnet–DOS file specification has the following format:

node::dev:path\filename.typ

where

node	is the name or address of a DECnet–DOS node.
dev	is the drive name.
path	is the path name.
filename	is an alphanumeric file name of up to 8 characters.
typ	is an alphabetic file extension of up to 3 characters.

EXAMPLE

The following command copies the file **bear** from a DECnet–ULTRIX node to the DECnet–DOS node OZARK. The DECnet–DOS file specification is enclosed within quotation marks so that OZARK, instead of the local shell, interprets this specification.

```
% dcp bear ozark::'a:test1.txt' RET
```

Glossary

This section defines the terms and concepts you must understand to read this book. This book assumes you are familiar with ULTRIX.

access-control information
Information used by the system to control access to system resources. Access control is the process of screening inbound connect requests and verifying them against a local system account file. Access control is optional. Access-control information consists of a user name, password, and account.

account
The allocation of system resources to each user. A user must have an account to use the system. Each user has a separate account, identified by a special account number and password.

alias
A short, meaningful name that replaces all, or some, of the access-control information you supply to a node.

ASCII file
A file in ASCII (American Standard Code for Information Exchange) format.

bidirectional gateway
The DECnet–Internet Gateway, which is network software that acts as a bridge between Internet and DECnet systems. The Gateway can support communication in two directions: from DECnet to Internet and from Internet to DECnet.

bidirectional gateway functions
The functions available through the DECnet–Internet Gateway, namely, remote login, mail exchange, and file access and transfer.

binary file
A file in binary (image mode) format.

case-sensitive
Refers to the system's ability to distinguish between uppercase (A-Z) and lowercase (a-z) letters.

connect errors
A type of error message which indicates that either the local or remote system failed to establish a DECnet network connection.

CTERM (Command Terminal Protocol)
The Digital Network Architecture (DNA) command terminal protocol. With **dlogin**, you can connect to any DECnet node that supports CTERM.

DAP (Data Access Protocol)
In the network application layer of the Digital Network Architecture (DNA), the protocol used for remote file access and transfer.

dcat (DECnet-ULTRIX concatenate)
The command that displays the contents of remote files.

dcp (DECnet-ULTRIX copy)
The command that copies files between DECnet systems.

dcp -c option flags
Flags you can use with the **dcp -c** option, namely, **ultrix, none, lower, nolower, nodollar, dollar, nosemicolon, semicolon, noversion,** and **version.** These flags can be combined to control how the **-c** option converts the file name.

DECnet-Internet Gateway
Digital Equipment Corporation's bidirectional gateway network software that acts as a bridge between Internet and DECnet systems. The Gateway is capable of supporting communication in two directions, from DECnet to Internet and from Internet to DECnet.

DECnet mail address
A syntax for addressing mail to remote DECnet users, namely, *username::node.*

DECnet node
Any node running DECnet software so that it can communicate with other nodes in the DECnet.

directory
A group of files stored on a disk. A user file directory is a file that briefly catalogs a set of files stored on tape or disk. The directory may include information such as the name, type, and version number of each file.

dlogin
The command that provides DECnet–ULTRIX end users with a virtual terminal connection to remote DECnet nodes.

dlogin escape character
The character you use to signal your local node to interrupt your **dlogin** session and return control to your local node. By default, the tilde (~) is the escape character.

dlogin session
The login session you start on a remote DECnet node using the **dlogin** command. The **dlogin** session lasts until you log off the remote DECnet node.

dls (DECnet-ULTRIX list)
The command that lists the contents of a remote DECnet directory.

dnet

The artificial domain name you include in the mail recipient's address when you use the optional Internet addressing syntax to send mail to remote DECnet users. Also called the "Internet psuedodomain name."

drm (DECnet-ULTRIX remove)

The command that removes, or deletes, remote DECnet files and directories.

fg (DECnet-ULTRIX foreground)

The command that resumes a suspended **dlogin** session. Enter this command at the **local command>** prompt.

file-access errors

A type of error message that indicates all error conditions other than DECnet connect errors.

file name

The title assigned to identify a specific file.

file specification

The unique identification of a file that gives its physical location and an indication of its contents. Different systems require different file specifications; refer to Appendix B for a discussion of the file specifications required by the DECnet systems supported by DECnet–ULTRIX.

help

You can display a list of **dlogin** commands by typing a question mark (?) at the **local command>** prompt.

Internet

A collection of packet-switching networks interconnected by gateways, along with protocols that allow them to function logically as a single,large,virtual network.

Internet mail address

An optional syntax for addressing mail to remote DECnet users, namely, *username@node*.dnet. This syntax is provided as a convenient interface for users familiar with the Internet syntax; DECnet–ULTRIX converts this address to the standard DECnet mail address before sending the mail out.

Internet psuedodomain name

The artificial domain name (.dnet) you include in the mail recipient's address when you use the optional Internet addressing syntax to send mail to remote DECnet users. See Section 3.1.

local command mode

An interface to your local node that you can access during a **dlogin** session. Use the escape character to enter local command mode. In local command mode, you can issue any ULTRIX or DECnet–ULTRIX command you want to execute on your local node. You can also issue DECnet–ULTRIX commands to control your **dlogin** session.

local login session

The login session you start on your local node when you log on directly. You can use the **dlogin** escape character to interrupt your **dlogin** session at any time to return to your local login session.

local node

The node you can log on to directly.

log file

A file containing a record of your input and the system's responses during a **dlogin** session.

mail

The ULTRIX command that lets you exchange electronic mail messages with other DECnet users.

man pages (manual pages)

Actual pages from reference manuals that you can display on your screen. The DECnet-ULTRIX man pages are printed in the DECnet–ULTRIX Command Summary (Chapter 6) of this manual.

man

The ULTRIX command that displays on-line manual pages on your screen. Manual pages are also called "command reference pages."

metacharacters

A group of keyboard characters (not including letters or digits) that have special meaning either to the shell or to the ULTRIX system. To use a metacharacter without its special meaning, either enclose it within quotation marks or precede it with a backslash.

node

An individual computer system in a network that can communicate with other computer systems in the network. Also called "host" and "system."

non-ULTRIX DECnet node

A DECnet node that runs an operating system other than ULTRIX (for example, VMS).

password

A combination of characters that verifies your identity to the computer.

path

The list of directories between the root directory and another directory. Also called "directory path."

piping

The process of sending the output from one command directly to another for use as the later command's input. You use the vertical bar character (|) as a pipe between commands. Although piping is not discussed in this manual, DECnet–ULTRIX supports piping according to standard ULTRIX conventions.

protection levels

The settings in each file that indicate who may and may not access the file. The settings are *read*, *write*, and *execute* privileges, and the groups are *owner*, *group*, and *world*.

proxy access

Proxy access allows you to gain access to a remote node without supplying access-control information. Proxy access uses proxy accounts, which system managers can establish.

redirection

The process of writing output from a command to a file using the right angle bracket (>), or of reading input for a command from a file using the left angle bracket (<). Although redirection is not discussed in this manual, DECnet–ULTRIX supports redirection according to standard ULTRIX conventions.

remote login session

The login session you start when you log on to a remote node. When you use the **dlogin** command to initiate the remote login session, the session is also called the **dlogin** session.

remote node

Any node in the network that is not the user's local node; any node that the user cannot log on to directly.

tilde (~)

The keyboard character that is the default **dlogin** escape character. The tilde is also the ULTRIX symbol for your home directory.

ULTRIX shell

The command interpreter for Digital Equipment Corporation. The ULTRIX product is a licensed derivative of UNIX software.

user name

The name a user types on a terminal to log on to the system.

wildcard character

A symbol, such as an asterisk or a percent sign, used within or in place of a file name, file type, directory, or version number in a file specification to indicate "all" for the given field.

world-readable file

A file all users can access, including both local and remote system operators, system managers, and users.

Index

HOW TO ORDER ADDITIONAL DOCUMENTATION

DIRECT TELEPHONE ORDERS

In Continental USA
call 800–DIGITAL

In Canada
call 800–267–6215

In New Hampshire
Alaska or Hawaii
call 603–884–6660

In Puerto Rico
call 809–754–7575

ELECTRONIC ORDERS (U.S. ONLY)

Dial 800–DEC–DEMO with any VT100 or VT200
compatible terminal and a 1200 baud modem.
If you need assistance, call 1–800–DIGITAL.

DIRECT MAIL ORDERS (U.S. and Puerto Rico*)

DIGITAL EQUIPMENT CORPORATION
P.O. Box CS2008
Nashua, New Hampshire 03061

DIRECT MAIL ORDERS (Canada)

DIGITAL EQUIPMENT OF CANADA LTD.
940 Belfast Road
Ottawa, Ontario, Canada K1G 4C2
Attn: A&SG Business Manager

INTERNATIONAL

DIGITAL
EQUIPMENT CORPORATION
A&SG Business Manager
c/o Digital's local subsidiary
or approved distributor

Internal orders should be placed through the Software Distribution Center (SDC),
Digital Equipment Corporation, Westminster, Massachusetts 01473

*Any prepaid order from Puerto Rico must be placed
with the Local Digital Subsidiary:
809–754–7575 x2012

DECnet–ULTRIX
Use
AA–PB61A–TE

READER'S COMMENTS

What do you think of this manual? Your comments and suggestions will help us to improve the quality and usefulness of our publications.

Please rate this manual:

	Poor				Excellent
Accuracy	1	2	3	4	5
Readability	1	2	3	4	5
Examples	1	2	3	4	5
Organization	1	2	3	4	5
Completeness	1	2	3	4	5

Did you find errors in this manual? If so, please specify the error(s) and page number(s).

_____

_____

_____

_____

General comments:

_____

_____

_____

_____

Suggestions for improvement:

_____

_____

_____

_____

Name _____ Date _____

Title _____ Department _____

Company _____ Street _____

City _____ State/Country _____ Zip _____

DO NOT CUT – FOLD HERE AND TAPE

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY LABEL
FIRST CLASS PERMIT NO. 33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

digital™

Networks and
Communications Publications
550 King Street
Littleton, MA 01460–1289

DO NOT CUT – FOLD HERE

CUT ON THIS LINE