Revision History

<table>
<thead>
<tr>
<th>Date</th>
<th>Revision</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7/ 21/ 00</td>
<td>A</td>
<td>Initial production release</td>
</tr>
</tbody>
</table>

Created 2000 and Protected Under the U.S. Copyright Act of 1976.
Copyright ©2000, SYSTECH Corporation
All Rights Reserved
This document is subject to change without notice.
# Table of Contents

SECTION ONE: USER GUIDE ............................................................................................................. 11

CHAPTER 1: INTRODUCTION ........................................................................................................... 13
WHAT IS A SYSTECH PORT SERVER? .......................................................................................... 13
USING A SYSTECH PORT SERVER ............................................................................................. 13
ORGANIZATION OF THIS MANUAL .............................................................................................. 14

CHAPTER 2: QUICK START .............................................................................................................. 17
QUICK SYSTEM CONFIGURATION ............................................................................................... 17
Assigning an IP Address .................................................................................................................. 17
QUICK CONFIGURATION REFERENCE ......................................................................................... 21

CHAPTER 3: SETTING UP COMMON CONFIGURATIONS ................................................................. 23
TERMINAL DEDICATED TO A HOST (DIRECT CONNECTION) ....................................................... 23
  Configuration Options and Descriptions ......................................................................................... 24
TERMINAL FOR RLOGIN/TELNET USING MULTIPLE SESSIONS ........................................... 25
  Configuration Options and Descriptions ......................................................................................... 26
DIAL-IN MODEM FOR TELNET/RLOGIN OR PPP/SLIP ............................................................ 27
DIAL-OUT MODEM (RTELNET) ...................................................................................................... 28
  Equipment required ......................................................................................................................... 28
  Configuring and installing the modem ............................................................................................. 28
  Accessing the modem through a host ............................................................................................. 29
    Using the telnet command ............................................................................................................ 29
    Using a reverse telnet connection ................................................................................................. 29
POOL OF DIAL-OUT MODEMS ..................................................................................................... 30
  Equipment required ......................................................................................................................... 30
  Configuring the modem .................................................................................................................... 30
  Accessing the modem through a host ............................................................................................. 31
    Using the telnet command ............................................................................................................ 31
    Using reverse telnet ...................................................................................................................... 31
DIAL-ON-DEMAND TO AN ISP .................................................................................................... 32
  Setting up system and dial on demand services ............................................................................ 33
PPP DIAL-ON-DEMAND BETWEEN TWO SYSTECH PORT SERVERS ........................................... 36
  Setting up system and dial on demand services ............................................................................ 36
  Bi-Directional Dial on Demand ..................................................................................................... 39
  Troubleshooting your dial-on-demand connection ........................................................................ 39
ISDN TERMINAL ADAPTER ON AN ASYNCH PORT ................................................................. 40
  Equipment required ......................................................................................................................... 40
  Setting up the ISDN terminal adapter ............................................................................................. 40
  Accessing the terminal adapter through a remote host ................................................................. 41
**ISDN TERMINAL ADAPTER ON A SYNCHRONOUS PORT**

**Equipment required**

**ISDN Terminal Adapter Setup**

Set up system and dial-on-demand configurations

**CONFIGURING THE WAN PORT**

**Setup 1**

On the remote port server cuda2

On the local port server cuda1

**Setup 2**

**PRINTING TO A SERIAL OR PARALLEL PORT**

Connecting a printer to the port server

**Parallel printer**

**Serial printer**

**Print Options**

Selecting a Print Option

Printing using LPD

**On the port server:**

**On the host system:**

Specifying port server print queue names

Specifying default data file format

Managing the print queue on the port server

Sample configurations

**UNIX environment (AIX 4)**

**UNIX environment (SunOS or DEC)**

**Windows NT 3.51 environment**

**Windows NT 4.0 environment**

**Windows 95 environment**

**Printing using reverse telnet (rtn)**

On the port server:

On UNIX host systems (except AIX):

On AIX host systems

**Printing to the serial port(s) with PPOOL**

On the port server:

On the host system:

**Troubleshooting the serial printer**

**UPDATING PORT SERVER SOFTWARE**

**COPYING CONFIGURATIONS**

**Requirements**

Uploading a configuration to the server

Downloading a configuration from the server

**ASSIGNING AN IP ADDRESS WITH A UNIX BOOTP SERVER**

**ASSIGNING AN IP ADDRESS WITH WINDOWS NT DHCP SERVER**

**CHAPTER 4: CONFIGURING PORTS**

**CONFIGURATION METHODS**

Web Browser

Command Line Interface

**SERIAL PORT CHARACTERISTICS**

Setting/Modifying Individual Serial Port Characteristics

Predefined Serial Port Configurations

Modifiable Serial Port Configuration

**WAN PORT CHARACTERISTICS**

**SERVICE CONFIGURATIONS**

**RTELNET Service**

Default Service Configuration
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLIP Service Configuration</td>
<td>79</td>
</tr>
<tr>
<td>PPP Service Configuration</td>
<td>80</td>
</tr>
<tr>
<td>Frame Relay Service Configuration</td>
<td>81</td>
</tr>
<tr>
<td>User-defined Custom Service Configuration</td>
<td>82</td>
</tr>
<tr>
<td>PARALLEL PORT</td>
<td>83</td>
</tr>
</tbody>
</table>

**SECTION TWO: REFERENCE**

**CHAPTER 5: CONFIGURATION COMMANDS**

- **CONFIGURATION OVERVIEW** ........................................ 87
- **COMMAND OVERVIEW** ............................................. 88
  - Configuration Commands ....................................... 88
  - Saving the Configuration Settings .......................... 89
- **COMMAND USAGE** .................................................. 89
  - Command Syntax .................................................. 89
  - Keyword Summary .................................................. 91
  - Description and Examples ...................................... 99
  - BANNER ............................................................... 100
  - COMMUNITY ......................................................... 101
  - CUSTOM .............................................................. 102
    - Configuration Values ......................................... 103
      - authentication level ....................................... 103
      - capabilities level ........................................ 104
      - terminal type ................................................ 104
      - restart delay ................................................ 105
      - wait for keyboard hit ...................................... 105
      - display banner ............................................... 105
      - 1-key menus .................................................. 105
      - session ....................................................... 106
    - Defining Capabilities Access Levels ....................... 107
      - Level 1: user .................................................. 107
      - Level 2: net ................................................... 108
      - Level 3: to admin ........................................... 108
      - Level 4: admin ............................................... 108
      - Level 5: expert ............................................... 109
  - DEFAULT ............................................................ 111
  - DIAL ON DEMAND ..................................................... 112
  - DLCI ........................................................................ 114
  - DOMAIN NAME ........................................................ 115
  - FRAME_RELAY .......................................................... 116
  - HOST_NAME_LIST ..................................................... 117
  - IGNORED_DCD .......................................................... 118
  - INTERFACE .............................................................. 119
  - IP_ADDRESS ............................................................. 121
  - MAC ....................................................................... 122
  - MASK .................................................................... 123
  - MODEL .................................................................... 124
  - MODEM .................................................................... 125
  - MPOOL ................................................................... 126
  - NAME_SERVER_LIST .................................................. 129
  - NETWORK ADDRESS TRANSLATION (NAT) ......................... 130
  - OUTPUT ................................................................ 133
  - OWNER ................................................................... 135
  - PARITY .................................................................. 136
  - PASSWORDS ................................................................ 137
CHAPTER 6: OPERATIONAL COMMANDS........................................... 175

Operational Command Summary.............................................. 175
admin............................................................................... 179
apmem ............................................................................ 180
arp.................................................................................... 181
boot................................................................................... 182
drop................................................................................... 184
dump................................................................................... 185
dumpf............................................................................... 186
expert............................................................................... 187
flash_compact................................................................. 188
flash_log......................................................................... 189
flash_save....................................................................... 190
flash_update................................................................. 192
fstats.............................................................................. 193
gdbserv......................................................................... 194
hangup............................................................................. 195
help.................................................................................. 196
http_stats...................................................................... 197
ifconfig.......................................................................... 198
kill.................................................................................... 201
oommem......................................................................... 202
ookstream....................................................................... 203
pc...................................................................................... 205
pm..................................................................................... 206
npool............................................................................... 207
netstat............................................................................ 208
ping.................................................................................. 210
# Table of Contents

## CHAPTER 9: SNMP TCP DROP CAPABILITY
- The Management Information Base (MIB) .................................. 281
- Understanding the TCPConnTable ............................................... 281
- Issuing the TCP Drop Command .................................................. 283

## CHAPTER 10: TROUBLESHOOTING TOOLS
- Data Capture ................................................................................. 285

## CHAPTER 11: FIRMWARE MONITOR
- Entering the Firmware Monitor ..................................................... 288
- Command Menu ............................................................................ 288
- Diagnostic Menu ........................................................................... 290

## APPENDICES
- Appendix A: Open Networks ....................................................... 291
- Appendix B: Routing ................................................................. 295
- Appendix C: SLIP and PPP ............................................................. 299
Chapter 1: Introduction

What is a Systech Port server?

A Systech port server is a communications server that connects to 10 Mbps Ethernet and provides multiple EIA-232 serial ports. Some port server models also provide a unidirectional parallel printer port and support for a high-speed synchronous modem or other device (EIA-232, V.35, EIA-530, or EIA-422) (optional). The port server uses the TCP/IP family of protocols to provide terminal connectivity and communications services in “open systems” environments.

As system administrator, you will have responsibility for setting up and configuring the port server to meet your usage requirements. The port server has been designed to make your job easy. The hardware is simple to install, and a concise command set ensures that configuration is just as simple. In addition to configuration commands, the communications server provides commands for monitoring and managing your network activity and for diagnosing and troubleshooting system problems.

Using a Systech Port Server

Using the port server requires three steps:

1. Hardware installation
2. Configuration
3. Operation

Hardware installation is specific to the port server family you are installing and is documented in the Hardware Manual for your port server, available on the CD that was provided with your port server or on the Systech Web site (www.systech.com).

There are three ways to configure and use your port server: NativeCOM, a web browser, or the command line interface.

- **NativeCOM**: If you are using Windows, you can use NativeCOM, a software package provided free of charge by Systech. The NativeCOM Networked COM ports software makes remote serial communication ports (e.g. EIA-232 ports on a terminal server) available to PC programs as local COM ports. Applications from a PC running Microsoft Windows 95, Windows 98, or Windows NT can access and use the remote serial ports. The configuration is handled by NativeCOM. You generally do not need to configure the ports directly, with a few exceptions such as changing the electrical
interface or to configure a port that is not under NativeCOM control. For further information on configuring and using Networked COM ports, see the NativeCOM manual. The NativeCOM software and manual are available on the CD you received with your port server and on the Systech web site (www.systech.com).

- **Web Browser**: If you are not using Windows or do not wish to use NativeCOM, you can configure your port server directly using your web browser. To use the browser, point it at the IP address of your Systech port server. Information on assigning an IP address to your port server and on the configuration and operation options for your port server is provided in this Administrator’s Guide.

- **Command line interface**: You can also configure your port server directly using the command line interface, rather than a browser, by connecting a terminal to one of the serial ports or by connecting to the port server via telnet. Information on configuration and operation using the command line interface is available in this Administrator’s Guide.

## Organization of this manual

The Systech Port Server Administrator’s Guide is divided into two main sections: User Guide and Reference. The main sections are followed by three appendices.

**Section One: User Guide**

- **Chapter 1: Introduction**
  Overview of a port server. Organization of this manual.

- **Chapter 2: Quick Start**
  Quickly setup and use your port server, almost right from the box. Shows default configuration settings.

- **Chapter 3: Setting Up Common Configurations**
  Describes steps for setting up various configurations you might use, including terminal dedicated to a host, dial on demand routing, dial-in and dial-out modems and printer pools. Also describes updating the port server software and copying configurations.

- **Chapter 4: Configuring Ports**
  Describes port characteristics and services and provides information on modifying port configuration.

**Section Two: Reference**

- **Chapter 5: Configuration Commands**
  Describes the basic commands for configuring the port server: add, delete, set, and show.
Chapter 6: Operational Commands
Describes commands you will use for operating and troubleshooting the port server.

Chapter 7: Host System Utilities
Describes the installation and use of three host utilities that provide host access to the port server and its ports: the rtn utility that allows UNIX software to access serial devices attached to port server ports; the r4000 utility that allows the management of port server flash memory from the host system; and the dtelnetd utility that allows the port server to initiate telnet sessions with a specified PTY on a UNIX host.

Chapter 8: Writing Chat Scripts
Describes guidelines and syntax for writing chat scripts so that the port server can communicate with modems during a dial-on-demand session.

Chapter 9: SNMP TCP Drop Capability
Describes the capability of the port server to drop TCP connections using a remote SNMP manager node.

Chapter 10: Troubleshooting Tools
Describes the use of tools for data capture.

Chapter 11: Firmware Monitor
Describes the port server firmware monitor which allows you to suspend normal operation and perform other operations such as making a bootp request, restoring configuration settings, updating software, and troubleshooting.

Appendices

Appendix A: Open Networks
Describes open systems, including networks and TCP/IP.

Appendix B: Routing
Describes routers and gateways, using routing tables, and routing information protocol.

Appendix C: SLIP and PPP
Describes using the Serial Line Interface Protocol (SLIP) and Point-to-Point Protocol (PPP) methods of transmitting IP datagrams. Also describes the Password Authentication Protocol (PAP) and Challenge-Handshake Authentication Protocol (CHAP).
Chapter 2: Quick Start

This chapter summarizes steps you can follow to quickly set up and use your port server after the hardware installation (documented in the Hardware Manual provided with your port server). It describes the default settings which allow you to begin using the server almost as soon as you remove it from the box.

Chapter 3 contains detailed information about setting up more advanced services, including:

- Dial-out modems
- Printers
- IP routing functions
- PPP or SLIP services
- High-speed WAN connections

Quick System Configuration

When you power up the port server, the status LED flashes on and off as the port server runs its self-test. When the self-test completes, the LED turns green.

Assigning an IP Address

Before you can use the port server as a communication server, you need to assign an IP address to it. Assign the IP address using one of five methods:

- install and use the NativeCOM software, setting the IP address as part of the configuration process (NativeCOM software and manual are available on the CD that was provided with your port server or from the Systech Web site (www.systech.com).
- directly assign the address on the port server via terminal
- assign the IP address using ARP on a host
- assign the IP address using a BOOTP server
- allow a DHCP server to assign the IP address

If you previously set an IP address and now wish to set a new IP address on a different network and if you are not using NativeCOM, you need to delete the previous IP address before assigning a new one. You can delete the IP address while still connected to the old network using the browser interface or by telneting to the port server. If the unit is already connected to the new network, you can attach a terminal and change the IP address using the set ip command or you can go into the firmware monitor and restore the default configuration.
Assigning the IP Address via Terminal

1. If you connected a terminal, the “Welcome” banner and a password prompt will appear on your terminal. The factory-supplied user password is “user”; the administrative password is “admin”. Enter administrative mode, so you can assign an IP address to the port server:

```
Welcome to the port server Remote Communications Server
Password:user
>>admin
Password:admin
admin>>
```

2. To assign the ip address, use the `set ip` command. For example, if the port server’s IP address is 192.111.222.33, type:

```
admin>> set ip 192.111.222.33
```

3. If your network is sub-netted, you also need to enter a subnet mask. For example, on a class B network with 8-bit subnet numbers, you would type:

```
admin>> set mask 255.255.255.0
```

4. Save the address in non-volatile memory using the `save` command:

```
admin>> save
```

The IP address will be retained during reboot.

5. Reset the port server, either by cycling power or by entering “expert” mode and issuing the `reboot` command:

```
admin>> expert
expert>> reboot
```

6. When the port server has rebooted, log in again by repeating step 1 above. You can establish telnet or rlogin sessions with other hosts on your network via any of the serial ports on the port server using commands available in “user” mode or you can continue customizing the port server for your application using commands available in “admin” mode.

ARP

1. Record the MAC address for use with the `arp` command. The MAC address is printed on a label on the outside case of the port server. It is a 12 digit number, beginning with 00:80:44, printed in bar code and in digits.

2. Boot the port server.

3. Enter an `arp` command on a host on your network. Use the appropriate `arp` command to associate an IP address with a MAC address. Usually, the command will be:

```
arp -s {IP} {MAC}
(on AIX: arp -s ether {IP} {MAC} )
```

where: IP is the IP address you want to assign to the port server and MAC is the MAC address on the port server. For UNIX versions, enter the MAC address as shown, including the colons. For the Windows version, enter the MAC address replacing the colons with hyphens (e.g., 00-80-44 etc.).
For Windows, if more than one network adapter is configured, the interface address must be used with the `arp -s` command. If the interface address is not used, the `arp` command cannot distinguish which interface is being referenced, causing the command to be discarded without executing, with no error message returned. With more than one network adapter, enter:

```
arp -s {IP} {MAC} {interface address}
```

**Example:** `arp -s 209.75.217.100 00-80-44-0c-08-33 209.75.217.240`

where 209.75.217.240 is the address of the network adapter to which the port server is connected.

4. From the same host, telnet to the port server via the IP address specified in the `arp` command.

   On Windows:
   
   select: `start>run`
   enter: `telnet {IP}`

   On UNIX, enter:
   
   `% telnet {IP}`

   where `{IP}` is the IP address.

5. The port server will see this access to its MAC address and adopt the given IP address as its own.

Using the above method, the IP address will need to be reassigned whenever the port server is rebooted. To assign a permanent IP address, telnet to the port server (see step 4 above) and assign an IP address as described in the following steps.

1. The “Welcome” banner and a password prompt will be displayed. The factory-supplied user password is “user”; the administrative password is “admin”. Enter administrative mode, so you can assign an IP address to the port server:

   ```
   Welcome to the port server/3000 Remote Communications Server
   Password:user
   >>admin
   Password:admin
   admin>>
   ```

2. To assign the IP address, use the `set ip` command. For example, if the port server’s IP address is 192.111.222.33, type:

   ```
   admin>> set ip 192.111.222.33
   ```

3. If your network is sub-netted, you also need to enter a subnet mask. For example, on a class B network with 8-bit subnet numbers, you would type:

   ```
   admin>> set mask 255.255.255.0
   ```

4. Save the address in non-volatile memory using the `save` command:

   ```
   admin>> save
   ```

   The IP address will be retained during reboot.
5. Reset the port server either by cycling power or by entering “expert” mode and issuing the `reboot` command:

   admin>> expert
   expert>> reboot

6. When the port server has rebooted, log in again by repeating step 1 above. You can establish telnet or rlogin sessions with other hosts on your network via any of the serial ports on the port server using commands available “user” mode. Or you can continue customizing the port server for your application using commands available in “admin” mode.

**BOOTP**

1. Record the MAC address. The MAC address is printed on a label on the outside of the port server case. It is a 12 digit number, beginning with 00:80:44, printed in bar code and in digits.

2. Configure the BOOTP server with the MAC address of the port server and an IP address by modifying the “bootptab” file. The MAC address should be entered as a string of 12 digits, without spaces or hyphens.

   For further information, see the section, Assigning an IP address with a UNIX BOOTP server, page 35.

   The BOOTP server will reassign the same IP address to the port server each time the port server is rebooted.

3. Boot the port server on the same network as your BOOTP server.

**DHCP**

1. Record the MAC address. The MAC address is printed on a label on the outside of the port server case. It is a 12 digit number, beginning with 00:80:44, printed in bar code and in digits.

2. Boot the port server on the same network as the DHCP server.

3. The DHCP server will assign an IP address to the port server. Query the DHCP server to determine the IP address that was assigned. Look up the IP address using the MAC address.

   The port server will request a lease of 28 days. It will not accept a lease of less than 10 minutes. If the DHCP server does not specify a time for renegotiation, the port server will begin renegotiation when the lease is half way to expiration. If renegotiation cannot be completed for a new lease before the current lease expires, the port server will release its IP address and will have no IP address. As long as the port server does not have an IP address, it will continue to negotiate for one.

   **NOTE**: If the port server encounters both BOOTP and DHCP servers on the same network, it will attempt to get the IP address from the DHCP server first, but may experience problems. When both servers are present, it is recommended that you assign the IP address with DHCP, rather than configuring BOOTP to assign the IP address.
Quick Configuration Reference

After assigning an IP address to the port server, you can configure it for your applications. You can configure it by using a terminal attached to one of its serial ports, by connecting to it via telnet, or by using a web browser.

Table 2-1 summarizes some of the commonly changed configuration items. See the chapter, Setting Up Common Configurations, page 23, for extended examples of configurations for various common applications. See the chapter, Configuration Commands, page 87, for a comprehensive reference for configuration commands.

Table 2-1. Port server Quick Configuration Summary

<table>
<thead>
<tr>
<th>Option</th>
<th>Factory setting</th>
<th>Port server Shell Command</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Passwords</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>User login</td>
<td>user</td>
<td>set user {password}</td>
</tr>
<tr>
<td>Administrative user</td>
<td>admin</td>
<td>set admin {password}</td>
</tr>
<tr>
<td><strong>Networking</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Internet address</td>
<td>0.0.0.0</td>
<td>set ip {IP Address}</td>
</tr>
<tr>
<td>Subnetwork mask</td>
<td>0.0.0.0</td>
<td>set mask {mask value}</td>
</tr>
<tr>
<td>Initial route list</td>
<td></td>
<td>add route {dest} {gateway} {flags} {hops}</td>
</tr>
<tr>
<td>Host name list</td>
<td></td>
<td>add host {name} {P address}</td>
</tr>
<tr>
<td>Domain name list</td>
<td>yourdomain.com</td>
<td>set domain {name}</td>
</tr>
<tr>
<td>DNS server list</td>
<td></td>
<td>add nameserver {IP address}</td>
</tr>
<tr>
<td><strong>Asynch serial ports</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service definition Speed</td>
<td>default 9600</td>
<td>set {Cn} {port}</td>
</tr>
<tr>
<td>Character length</td>
<td>8</td>
<td>set speed {port} {bit rate}</td>
</tr>
<tr>
<td>Stop bits</td>
<td>1</td>
<td>set size {port} {5</td>
</tr>
<tr>
<td>Parity</td>
<td>none</td>
<td>set stop {port} {1</td>
</tr>
<tr>
<td>Input flow control</td>
<td>xoff</td>
<td>set parity {port} {even</td>
</tr>
<tr>
<td>Output flow control</td>
<td>xon</td>
<td>set input {port} {none</td>
</tr>
<tr>
<td>Inactivity timeout</td>
<td>6000</td>
<td>set output {port} {none</td>
</tr>
<tr>
<td>switch character</td>
<td>required</td>
<td>set timeout {port} {secs}</td>
</tr>
<tr>
<td>dcd modem signal</td>
<td></td>
<td>set switch {port} {char</td>
</tr>
<tr>
<td><strong>Inactivity timeout</strong></td>
<td></td>
<td>set ignoredcd {port} {yes</td>
</tr>
<tr>
<td><strong>WAN port</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Physical interface</td>
<td>rs232</td>
<td>(NOTE: Only some port server models have WAN ports)</td>
</tr>
<tr>
<td>Clocking</td>
<td>internal 9600</td>
<td>set wan interface {rs232</td>
</tr>
<tr>
<td>Inactivity timeout</td>
<td>6000</td>
<td>set wan clock {external</td>
</tr>
<tr>
<td></td>
<td></td>
<td>set timeout {port} {secs}</td>
</tr>
</tbody>
</table>
Chapter 3: Setting Up Common Configurations

This chapter includes several extended examples, or application notes, describing common port server setups:

- **Terminal dedicated to a host (direct connection)** See page 23
- **Terminal for rlogin/ telnet using multiple sessions.** See page 25
- **Dial-in modem for telnet/ rlogin or ppp/ slip** See page 27
- **Dial-out modem (rtelnet)** See page 28
- **Pool of dial-out modems** See page 30
- **Dial-on-Demand to an ISP** See page 32
- **PPP Dial-on-Demand between two Systech Port Servers** See page 36
- **ISDN terminal adapter on an asynch port** See page 40
- **ISDN terminal adapter on a synchronous port** See page 42
- **Configuring the WAN port** See page 46
- **Printing to a serial or parallel port** See page 51

It also describes the procedures for the following:

- **Updating Port Server Software** See page 63
- **Copying Configurations** See page 63
- **Assigning an IP address with a UNIX BOOTP server** See page 65

**Terminal dedicated to a host (direct connection)**

The port server can be configured to automatically connect a terminal to a host located anywhere on the network. This technique could be used in applications that require a user’s terminal to communicate with one and only one host computer, perhaps running a specialized application program. This configuration closely models a terminal that is directly connected to a host, via a conventional serial interface. The following example shows how this can be done for the terminal connected to port 1.
A custom service configuration must be defined to support a direct connect. At the admin access level enter the following:

>add custom C1

The custom configuration that you define should be similar to this:

---

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration name</td>
<td>Define as C0 through C33. Must be different for each unique configuration.</td>
</tr>
<tr>
<td>Authentication</td>
<td>Selecting none allows user to go directly to host system. Selecting user or admin requires a password before logging onto the host.</td>
</tr>
<tr>
<td>Capabilities</td>
<td>Selecting user allows telnet and rlogin access.</td>
</tr>
<tr>
<td>Terminal</td>
<td>Enter type appropriate for the user.</td>
</tr>
<tr>
<td>Restart Delay</td>
<td>Specifies 1-second delay between ending a session and starting a new one.</td>
</tr>
<tr>
<td>Wait for keyboard hit</td>
<td>Login banner will not display when the terminal is first powered on or connected.</td>
</tr>
<tr>
<td>Display banner</td>
<td>One-key menus are not used, since you want the user to automatically connect to the remote host, bypassing the port server shell.</td>
</tr>
<tr>
<td>Sessions</td>
<td>Defines the command the port server will execute to connect to the host: telnet or rlogin. If the host name has been added to the hosts list or if you are using DNS, you can use a host name instead of the IP address. The command shown in the example will establish a telnet session with the host whose IP address is 198.130.71.67.</td>
</tr>
</tbody>
</table>
---

The IP address set in the telnet command (198.130.71.67 in the example) should be that of the target host system. The table below describes the service configuration options in detail. Once a service configuration has been defined, it must then be assigned to a port or range of ports. Enter the following to assign configuration C1 to port 1:

>set C1 1

---

Command for session 1: telnet 198.130.71.67

---
Terminal for rlogin/telnet using multiple sessions.

The port server supports up to four sessions for each connected terminal. These sessions can be individually connected to any host on the network. The following example establishes three sessions for the terminal connected to port 1:

- Session 1 is a logon shell on the port server
- Session 2 is a remote logon to host 198.130.71.12
- Session 3 is a telnet session to host 198.130.71.18.

You must define a custom service configuration to support a connection with multiple sessions. Each port may have up to four sessions defined, allowing the user to enter the BREAK key to switch between the sessions. To define, enter the following at the admin level prompt:

```
>add custom C1
```

The custom configuration that you define should be similar to this:

```
Configuration name     C1
Authentication         user
Capabilities user
Terminal vt100
Restart Delay 1
Wait for keyboard hit no
Display banner yes
One-key menus None
Sessions
Command for session 1: shell
Command for session 2: rlogin 198.130.71.12
Command for session 3: telnet 198.130.71.18
```

Once you define a service configuration, you must assign it to a port or range of ports. Enter the following:

```
>set C1 1
```

The terminal type determines how the screen is updated between session switches. Terminal modes that support multiple display pages (e.g., Wyse-50+; Wyse-60) will repaint the screen, restoring the state of the session at the previous switch. Most standard terminal modes (e.g., vt200; ansi) do not support multiple display pages and will only clear the screen on a session switch. If the selected terminal type does not support screen clearing, no screen update will occur on a session switch.
## Configuration Options and Descriptions

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration name (C0-C33)</td>
<td>Define as C0 through C33. Must be different for each unique configuration.</td>
</tr>
<tr>
<td>Authentication (user)</td>
<td>Requires operator to enter user password before an active session begins.</td>
</tr>
<tr>
<td>Capabilities (user)</td>
<td>Selecting user allows telnet and rlogin access.</td>
</tr>
<tr>
<td>Terminal (vt100)</td>
<td>Enter type appropriate for the user.</td>
</tr>
<tr>
<td>Restart Delay (1)</td>
<td>Specifies 1-second delay between ending a session and starting a new one.</td>
</tr>
<tr>
<td>Display Banner (yes)</td>
<td>Identifies port server unit to which operator is logging on.</td>
</tr>
<tr>
<td>One-key Menus (none)</td>
<td>No one-key menus are configured in this example. One-key menus could be defined and would be available to the user in the shell session (session 1) as shortcuts for commonly used commands.</td>
</tr>
<tr>
<td>Sessions</td>
<td>Defines what command the port server will execute to connect to the host: telnet, shell or rlogin. If the host or nameserver has been defined, you can use a host name instead of the IP address.</td>
</tr>
</tbody>
</table>

When you switch from one session to another and then return to the original session, the screen will either re-display the original screen or clear the screen. Whether the original session screen is re-displayed or the screen is cleared depends on the terminal type defined in your custom configuration. Only terminal types that support multiple page display can re-display the original screen. The terminal type must support a number of multiple pages equal to the number of sessions required. That is, if a user is switching between 4 sessions (the maximum number of sessions the port server can support), the terminal must provide multiple page display support for at least 4 pages. Of the currently supported terminal types, only the Wyse 50+ terminal supports sufficient multiple display pages to support 4 sessions and, therefore, allows the original screen to be redisplayed after a switch when 4 sessions are configured. The Wyse 60 terminal supports up to 2 multiple display pages and so will re-display the original session if the user has only 2 sessions, but will clear the screen if the user has 3 or 4 sessions. All other supported terminal types will clear the screen when the user returns to the original session. (For a list of supported terminal types, see terminal types, page 104.)
Dial-in modem for telnet/rlogin or ppp/slip

These directions show how to configure a modem on a serial port to allow users to dial-in and automatically start a slip, ppp, rlogin or telnet session. In this example, we will set up a hayes compatible modem on serial port 1.

1. First, configure the modem. Connect the modem to port 1. On the port server, set up the following to allow you to communicate with the modem:

   set rtelnet 1
   set speed 1 115200

   Now connect to the modem with the following command:

   telnet cuda1 9001

   Assuming that cuda1 is the name of the port server you are using, this will connect you to the modem attached to port 1.

2. Configure the modem for auto answer. For the hayes compatible modem, for example, you would issue these commands to the modem to configure it:

   at&f Recall factory configuration
   at&c1 Track presence of carrier detect signal
   at&d2 Monitor DTR signal
   ats0=1 Answer after 1 ring
   at&k3 Enable RTS/CTS local flow control*
   at&r0 CTS tracks RTS
   at&s0 Assert DSR signal always
   at&w0 Write storable params in profile 0
   at&y0 Specify profile 0 as power-up configuration

   *This parameter may vary on your modem. Motorola Lifestyle modems use \Q3 for this mode.

3. Now set up the port for dial-in access. Configure port 1 as follows:

   set speed 1 115200
   set size 1 8
   set parity 1 none
   set input 1 rtscts
   set output 1 rtscts

4. Define a custom service configuration C10 using the add custom C10 command:

   Configuration name C10
   Authentication none
   Capabilities user+net
   Terminal vt100
   Restart Delay 10
   Wait for keyboard hit no
   Display banner no
   One-key menus None
Sessions
Command for session 1: (one of the following)
- ppp 198.130.71.15:198.130.71.213 (displayed when you enter `show ip` command)
- slip 198.130.71.213:198.130.71.15 (address assigned to the dial-in application)
- telnet <host name> or <IP address>
- rlogin <host name> or <IP address>

The commands shown will automatically start a service on the dial-in port as soon as the modem receives a call and connects to the remote (calling) modem.

5. Assign the custom service configuration C10, to port 1.
   ```
   set C10 1
   ```

6. Save the configuration changes to flash.
   ```
   save
   ```

**Dial-out modem (rtelnet)**

The following example provides a quick step-by-step description of how to add a dial-out modem to the port server, including:

- Equipment required
- Configuring and installing the modem
- Attaching the modem to a serial port
- Accessing the modem via a host

**Equipment required**

The following equipment is required before you can install and configure the dial-out modem:

- port server
- Hayes-compatible modem (for this example)
- Cable and connectors
- Host reverse telnet (rtelnet) program (optional)

**Configuring and installing the modem**

In this example, a modem is being attached to port 1 of a port server with a hostname cudasrv.

1. Connect the modem to port 1.

2. Setup reverse telnet on port 1 by typing the following command from the admin shell:
   ```
   set rtelnet 1
   ```
This will set up the port server to respond on tcp port 9001 to connect the telnet service to serial port 1. The tcp port number will change depending on the serial port to which you are connecting the modem. Each serial port has a tcp port associated with it. Serial ports on various models can range from port 0 through port 16 and are associated with tcp ports 9000 through 9016, respectively.

3. Set the baud rate of port 1 by typing:

   set speed 1 115200

   This sets the baud rate to 115200. Change this based on the type of modem you have. (Some modems may not be able to handle high DTE baud rates.)

4. Save your configuration by typing:

   save

Accessing the modem through a host

There are two ways to access the modem through a host:

- Using the `telnet` command
- Using a reverse telnet connection and an application program (such as terminal emulator, tip, cu)

Using the `telnet` command

The quickest and easiest way to access the modem is by using the `telnet` command. However, this does not allow a device name to be associated with the modem, which might be required by some applications (e.g. terminal programs).

To access the modem via `telnet`, type:

   telnet cudasrv 9001

`cudasrv` is the host name of the port server and 9001 is the tcp port that the modem is attached to. The IP address of the port server could be substituted for the host name.

You should now be able to type 'AT' commands to communicate with the modem. For example, to dial a number type:

   ATDT555-5555

Using a reverse telnet connection

To show an example of how to access the modem via a reverse telnet connection and an application program we will use the 'tip' program and reverse telnet running on a Solaris system (see The `rtn` Utility, Chapter 7, page 263 for a full description of the `rtn` program).

The following steps show how this is accomplished.

1. Run a reverse telnet program on the host. Reverse telnet programs for many versions of Unix are included with the port server. To run the Solaris version of reverse telnet, from the directory where `rtn.soli` is located, type:
rtn.soli -d /dev/modem1 -h cudasrv -p 9001

The "/dev/modem1" parameter in the above command is the name of the device that you want associated with the modem. You may name this anything you like.

2. Edit /etc/remote and add the following lines:

```
modem1:\
:dv=/dev/modem1:br#38400:pa=none:el=^C^S^Q^D:ie=%$:oe=^D:
```

`modem1` is the name of the device to be associated with the modem on the port server. Refer to the man pages of your system for an explanation of the other parameters.

3. Enter the following command:

```
tip modem1
```

This will connect you to the modem and you should now be able to type AT commands.

---

### Pool of dial-out modems

This section describes adding a pool of dial-out modems to the port server, including:

- Equipment required
- Configuring the modems
- Attaching the modems to serial ports
- Accessing the modems through a host

---

#### Equipment required

Make sure you have the following equipment:

- port server
- Hayes-compatible modems (used in this example)
- Cables and connectors
- Host reverse telnet (rtelnet) program (optional)

#### Configuring the modem

In this example, modems are being attached to ports 1-3 of an port server with the hostname cudasrv. These ports will be added to mpool 1.

1. Connect the modems to ports 1 through 3.
2. Add the modems attached to ports 1 through 3 to mpool 1 by typing the following command from the admin shell:

```
set mpool 1-3 1
```

The parameter '1-3' specifies the ports to add to mpool 1, which is specified by the parameter '1'. These values may be different from the actual values that you are using, depending on what ports your modems are attached to and what mpool you would like to use.
3. Set the baud rate of ports 1-3 by typing:

```
set speed 1-3 115200
```

This sets the baud rate to 115200 on ports 1-3. You might want to change this, based on
the type of the modem you have. (NOTE: Some modems may not be able to handle high
DTE baud rates.)

4. Save your configuration by typing:

```
save
```

Accessing the modem through a host

There are two ways to access modems through a host:

- Using the telnet command
- Using a reverse telnet connection and an application program (such as terminal
eulator, tip, cu)

**Using the telnet command**

The quickest and easiest way to access a modem in the pool is with the `telnet` command.
However, this does not allow a device name to be associated with the modem, which might
be required by some applications (such as terminal programs).

To access the modem through telnet, type:

```
telnet cuorasrv 9101
```

cudasrv is the host name of the port server and 9101 is the tcp port to which the modem pool
is attached. The '91' in '9101' specifies an mpool and the '01' is the number of the mpool.

You should now be attached to the modem located on port 1. You can communicate with the
modem by typing 'AT' commands. For example, to dial a number type:

```
ATDT555-5555
```

**NOTE:** To communicate to the modems attached to ports 2 and 3 issue
additional telnet commands without terminating your session to port 1.

**Using reverse telnet**

To access the modem in the pool with a reverse telnet connection and an application
program, use the 'tip' program and reverse telnet running on a Solaris system. Do the
following:

1. Run a reverse telnet program on the host. Reverse telnet programs for many versions of
Unix are included with the port server. To run the Solaris version of reverse telnet from
the directory where rtn.soli is located, type:

```
rtn.soli -d /dev/modem1 -h cuorasrv -p 9101
```
The "/dev/modem1" parameter in the above command is the name of the device that you want associated with the modem. You may name this anything you like.

2. Edit /etc/remote and add the following lines:

```
modem1: 
   :dv=/dev/modem1:br#38400:pa=none:el=^C^S^Q^D:ie=%$:oe=^D:
```

'modem1' is the name of the device associated with the modem on the port server. Please refer to the man pages of your system for an explanation of the other parameters.

3. Enter the following command:

```
tip modem1
```

You should now be attached to the modem located on port 1. You can communicate with the modem by typing 'AT' commands. For example, to dial a number type:

```
ATDT555-5555
```

**NOTE**: To communicate to the modems attached to ports 2 and 3, issue additional rtn and tip commands.

---

**Dial-on-Demand to an ISP**

The port server can be used as a dialup router to provide on-demand network connectivity using point-to-point protocol (PPP). You can configure it to dial an Internet Service Provider (ISP) when it receives a packet destined for the Internet, and to disconnect the dialed line if it is idle for a specified amount of time.
In the above figure, whenever a machine on the local network needs to access data or systems on the Internet, the local port server dials the ISP, establishes the connection, and keeps it alive as long as there is network activity between the two sites. This is transparent to the user.

**Setting up system and dial on demand services**

This section describes how to configure the port server using the dial on demand service. DOD provides "on-demand" network connectivity via PPP. DOD dials the ISP when a packet arrives for the Internet. You will configure the local port server (cuda1).

**On the Local port server**

1. Configure the local port server (cuda1) port 1 for required serial port operation.
   ```
   set speed 1 115200
   set size 1 8
   set parity 1 none
   set input 1 rtscts
   set output 1 rtscts
   set timeout 1 30
   delete def 1
   ```

2. Define the modem
Add the modem description and the corresponding chat script to the setup database. The chat script here describes the modem-specific conversation which must take place between the port server and the modem in order to establish communication to any remote system. This could include initializing the modem and the dial string. (For more information on chat scripts, see [Writing Chat Scripts page 277](#)). For example:

```
[admin:1/7]>> add modem modem1
Enter the modem chat script : "" ATDT\T "CONNECT"
```

sets the modem chat script to do the following:

- Expect nothing.
- Send ATDT and the telephone number.
- Expect the response “CONNECT” when the connection is established.

The "\T" specifies: insert the phone number for the defined system here (see next section on setting up the system).

Enter the modem initialization string: AT

Now apply modem1 (the modem type you just defined) to a specific port that is to be used for dial-on-demand. Use the set command.

```
[admin:1/8]>> set modem 1 modem1
```

to set the modem type on port 1 to modem1.

3. Set up the system

Set up information about the remote system that you are dialing into. This would include the name of the system, the ports to use to dial to the remote system, the backoff time for dialout failures, and the login chat script.

Example:

```
[admin:1/9]>> add system isp1
Enter the ports to communicate through : 1
Enter the phone number to dial : 3485839
Enter backoff time (sec) (default 0) : 0
Enter the login chat script : ""
```

The example sets up to connect to the ISP, isp1, using port 1, with the telephone number, 348-5839, and no login chat script (indicated by entering two double quote characters, with no space between them).

The login chat script describes the system-specific conversation that must take place between the port server and the remote system to establish the ppp service, once the modem has made the connection to the system. In most cases, with an ISP, this is unnecessary. The port server can immediately begin ppp and ppp will provide authentication.

4. Set up the interface and add route to route packets to and from the remote system through the interface:

```
[admin:1/10]>> set dod 4
Enter system name : isp1
Enter remote IP address
(Enter 0 to automatically negotiate) : 0
```
Enter local IP address
(Enter 0 to automatically negotiate) : 0
Enter netmask in IP address format
(Enter 0 to automatically determine) : 0
Enter an interface name (e.g. 'ppp0') : ppp_ispl
Use Van Jacobsen header compression (y/n) : y

What kind of authentication should be used?

1 - Authenticate remote host using CHAP.
2 - Authenticate remote host using PAP.
3 - port server must provide authentication to remote host.
4 - No authentication is required by either side.

Please select a number 1 - 4: 3

Enter PPP Authentication ID to use for authentication (or 'default') : default

Remote IP address : Negotiated
Local IP Address : Negotiated
Subnet Mask : Negotiated
Interface Name : ppp_ispl
Van Jacobsen header compression : on
Authentication Type : port server must provide authentication to remote host.

PPP Authentication ID : default

Are these values correct (y/n) : y

Dial-on-demand interface dod4 added to the system.

Would you like to add a route for this interface? y

Enter the destination network (or 'default') : default

The requested route was added.

5. **Set up the authentication ID and password:**

   add pau *username password

   This sets the default username to **username** and its associated password to **password**. You must supply your username and password used to access the ISP. Add the asterisk (*) to the front of the username to make it the default.

6. **Use the save command to retain the setup for the next time you reboot:**

   [admin:1/12]>> save

Set up the other clients on the local network to use cuda1 as their default gateway.

Start some activity to the remote network to test the connection. On the local port server, enter the following command:

   [admin:1/13]>> ping -c 60 198.130.71.20

The ping command will start dialing and establish a connection to the remote site.
PPP Dial-on-Demand between two Systech Port Servers

The port server provides on-demand network connectivity using point-to-point protocol (PPP). You can configure it to dial a remote system when a packet arrives for a specified remote site, and to disconnect the dialed line if it is idle for a specified amount of time.

In the above figure, whenever a machine on the local network needs to access data or systems on the remote network, the local port server dials the remote port server, establishes the connection and keeps it alive as long as there is network activity between the two sites. This is transparent to the user.

Setting up system and dial on demand services

This section describes how to configure the port server using the dial on demand service. DOD provides "on-demand" network connectivity via PPP. DOD dials a remote system when a packet arrives for a pre-configured remote site. You will configure the local port server (cuda1) and the remote port server (cuda2).
On the Local port server

5. Configure the local port server (cuda1) port 1 for required serial port operation.

   set speed 1 115200
   set size 1 8
   set parity 1 none
   set input 1 rtscts
   set output 1 rtscts
   set timeout 1 30
   set def 1

6. Define the modem

   Add the modem description and the corresponding chat script to the setup database. The chat script here describes the modem-specific conversation which must take place between the port server and the modem in order to establish communication to any remote system. This could include initializing the modem and the dial string. (For more information on chat scripts, see Writing Chat Scripts, page 277.) For example:

   [admin:1/7]>> add modem modem1
   Enter the modem chat script : "" ATDT\T "CONNECT"

   sets the modem chat script to do the following:
   
   • Expect nothing.
   • Send ATDT and the telephone number.

   The "\ T" specifies: insert the phone number for the defined system here (see next section on setting up the system).

   Enter the modem initialization string: AT

   Now apply modem1 (the modem type you just defined) to a specific port that is to be used for dial-on-demand. Use the set command.

   [admin:1/8]>> set modem 1 modem1

   to set the modem type on port 1 to modem1.

7. Set up the system

   Set up information about the remote system that you are dialing into. This would include the name of the system, the ports to use to dial to the remote system, the backoff time for dialout failures, and the login chat script.

   The login chat script describes the system-specific conversation that must take place between the port server and the remote system to establish the ppp service, once the modem has made the connection to the system.

   Example:

   [admin:1/9]>> add system remote71
   Enter the ports to communicate through : 1
   Enter the phone number to dial : 3485839
   Enter backoff time (sec) (default 0) : 0
   Enter the login chat script : Password: user >> "ppp 199.171.148.1:198.130.71.2"
The example sets up to connect to the remote system, remote71, using port 1, with the telephone number, 348-5839, and the login chat script as follows:

```
expect Password:
send user
expect >> (note: this is the prompt)
send "ppp 199.171.148.1:198.130.71.2"
```

**NOTE:** The `ppp` command is the command that is run on the remote port server to start ppp on the port that answers.

8. Set up the interface and add route to route packets to and from the remote system through the interface:

```
[admin:1/10]>> set dod 4
Enter system name : remote71
Enter remote IP address
(Enter 0 to automatically negotiate) : 198.130.71.2
Enter local IP address
(Enter 0 to automatically negotiate) : 199.171.148.1
Enter netmask in IP address format
(Enter 0 to automatically determine) : 255.255.255.0
Enter an interface name (e.g. 'ppp0') : ppp_remote71
Use Van Jacobsen header compression (y/n) : y
What kind of authentication should be used?
  1 - Authenticate remote host using CHAP.
  2 - Authenticate remote host using PAP.
  3 - port server must provide authentication to remote host.
  4 - No authentication is required by either side.
Please select a number 1 - 4: 4
Remote IP address : 198.130.71.2
Local IP Address : 199.171.148.1
Subnet Mask : 255.255.255.0
Interface Name : ppp_remote71
Van Jacobsen header compression : on
Authentication Type : No authentication is required by either side
Are these values correct (y/n) : y
Dial-on-demand interface dod4 added to the system.
Would you like to add a route for this interface? y
Enter the destination network (or 'default'): 198.130.71.0
The requested route was added.
```

7. Use the `save` command to retain the setup for the next time you reboot:
```
[admin:1/12]>> save
```

**On the Remote port server**

1. Configure the remote port server (`cuda2`) port 2 for the required serial port operation.
```
set speed 2 115200
set size 2 8
set parity 2 none
set input 2 rtscts
```
set output 2 rtscts
set def 2

Start some activity to the remote network to test the connection. On the local port server, enter the following command:

[admin:1/13]>> ping -c 60 198.130.71.20

The ping command will start dialing and establish a connection to the remote site.

To complete the setup, you need to setup proper routes on the local and remote systems. (For information on setting up routing, see ROUTING, page 151 and Appendix B.)

**Bi-Directional Dial on Demand**

The remote office and corporate headquarters may need a configuration where either side can initiate the dial-on-demand connection. This is done by setting up both a dial-on-demand service and the default service on the same port on both port server units.

A dial-on-demand from the remote office will connect with the shell (default service) at the local office. Similarly, a dial-on-demand from the local office will connect with the shell at the remote office.

At the shell prompt, the login chat script initiates the PPP connection for the other end.

When setting up Bi-Directional Dial-on-Demand, you must configure the backoff time when adding the remote systems (for adding remote systems, see the SYSTEM command, page 163). The backoff time determines how long the port server waits before retrying a failed dialout attempt. For uni-directional dial-on-demand configurations, this value should generally be 0 to specify no backoff.

For bi-directional configurations, one of the port servers must be configured to backoff in order to avoid the deadlock which would occur if both units dialed each other simultaneously (and continued to redial each other at regular intervals). Typically, one unit should be configured with a backoff time of 0, while the other unit should be configured with a backoff time approximately equal to the amount of time it takes for a DOD connection to be established. For asynchronous modems, this is in the neighborhood of about 30 seconds. For ISDN modems, this value should be in the neighborhood of about 5 seconds. Use a backoff time which is appropriate for your configuration.

**Troubleshooting your dial-on-demand connection**

Syslog can provide useful information for troubleshooting your dial-on-demand connection. Configure syslog to save data in a large buffer for easier debugging:

[admin:1/14]>> syslog config
Specify a remote loghost: <cr>
defaulting to <your host>
Do you wish to log messages to the local circular buffer?: yes
How many entries would you like in the circular buffer?(Max=3000): 100
Depending on what the syslog display shows, check your modem and the remote modem speed, your login chat script, and the remote and local routes.

ISDN terminal adapter on an asynch port

In the previous examples we used analog modems to dial in, dial out, or make an “on demand” connection. The same can be accomplished with an ISDN terminal adapter with an asynch communication port. This section describes how to set up a Motorola BitSurfr™ in place of an analog modem, including:

- Equipment required
- Attaching the terminal adapter to an async port
- Setting up the ISDN terminal adapter
- Configuring the async port
- Communicating with the terminal adapter using a telnet connection
- Communicating with the terminal adapter using a reverse telnet connection

Equipment required

Make sure you have the following equipment:

- port server (hostname of cuda1 in this example)
- Hayes-compatible ISDN terminal adapter (Motorola BitSurfr™)
- RJ-45 straight through cable and connector
- Remote host reverse telnet (rtelnet) program (optional)

Setting up the ISDN terminal adapter

To set up a terminal adapter, do the following:

1. Connect the terminal adapter to port 1 using a modem cable.
2. Set the network switch and global parameters to achieve proper ISDN operation. Use the following commands (Refer to ISDN documentation for your specific terminal adapter commands):
   
   - `AT>F` Set Switch and Global parameters to factory default
   - `AT!C0=0` Switch type - AT&T 5ESS
   - `AT!C1=0` Switch version - AT&T Point-to-Point
   - `AT!D3=255` Auto data TEI
AT!C6=x...x  Data SPID - Refer to ISDN service provider documentation
AT!N1=x...x  Directory number - Refer to ISDN service provider documentation
AT*ID3=255   Auto Voice TEI
AT>W>Z       Save switch parameters and restart the network

**NOTE:** Network switch and Global parameters need to be entered only once - during initial terminal adapter powerup.

3. Once the network settings are configured properly, restore the factory default configuration (which is usually sufficient) with the following commands to the terminal adapter:

   AT&F0     Restore factory default profile 0
   AT&D2     No DTR signal causes BitSurfr to hangup, used for port timeout
   AT&C1     DCD on when BitSurfr establishes a connection
   AT%A4=0   Line channel speed 64 Kbps
   AT&W0     Save active profile to stored profile 0
   AT>Z      Restore switch options and restart the network

   **NOTE:** Factory default configuration is used as a basis for further settings and therefore does not need to be reentered in order to change to another terminal adapter mode.

4. After setting the terminal adapter to the factory default condition issue the following commands to enter the appropriate async mode.

   **V.120 Asynchronous Mode**
   No additional commands necessary

   **Asynchronous Inverse Multiplexing Mode**
   \[AT%A2=3\]
   \[AT@B0=2\]

**Accessing the terminal adapter through a remote host**

You may now access the terminal adapter as you did a modem in the previous examples of a dial-in modem, dial-out modem or dial-on-demand routing.
ISDN terminal adapter on a synchronous port

(NOTE: only some port server models have a synchronous WAN port. See Error! Reference source not found., page Error! Bookmark not defined.)

The following example provides a quick step-by-step description of how to add an ISDN terminal adapter to the sync port of the port server, including:

- Equipment required
- Setting up the ISDN terminal adapter
- Attaching the terminal adapter to a sync port
- Configuring the sync port
- Setting up dial on demand services

Equipment required

Make sure you have the following equipment:

- Local port server (hostname of cuda1 in this example)
- Remote port server (hostname of cuda2 in this example)
- Hayes-compatible ISDN terminal adapter (Motorola BitSurfr)
- Cable and connectors
- Local host (hostname of apollo in this example)
- Remote host (hostname of dino in this example)

The following diagram illustrates using the port server sync port and ISDN terminal adapters to route packets between isolated networks
ISDN Terminal Adapter Setup

We will first set up the terminal adapter in asynch mode.

Attach the terminal adapter to port 1 on the local port server unit (cuda1). On the port server, configure the following:

```
set rtelnet 1
set speed 1 115200
```

Now connect to the terminal adapter using the following command:

```
telnet cuda1 9001
```

You may now enter commands to configure the terminal adapter. To achieve proper ISDN operation, the Network switch and Global parameters must be set by the following commands to the terminal adapter: (Refer to ISDN documentation for your specific terminal adapter commands)

- **AT>F** Set Switch and Global parameters to factory default
- **AT!C0=0** Switch type - AT&T 5ESS
- **AT!C1=0** Switch version - AT&T Point-to-Point
AT!D 3=255   Auto data TEI
AT!C 6=x...x   Data SPID - Refer to ISDN service provider documentation
AT!N 1=x...x   Directory number - Refer to ISDN service provider documentation
AT!*D 3=255   Auto Voice TEI
AT>W>Z   Save switch parameters and restart the network

NOTE: Network switch and Global parameters need to be entered only once - during initial terminal adapter powerup.

Once the network settings are configured properly, you can restore the factory default configuration (which is usually sufficient) with the following commands to the terminal adapter:

AT&F 0   Restore factory default profile 0
AT&D 2   No DTR signal causes BitSurfr to hangup, used for port timeout
AT&C 1   DCD on when BitSurfr establishes a connection
AT%A 4=0   Line channel speed 64 Kbps
AT&M 2   Synchronous Dial Mode - used at ISDN terminal adapter 1
AT&Z 0=x...x   Store Dial number in register number 0
AT&W 1   Save active profile to stored profile 1
AT>Z   Restore switch options and restart the network

NOTE: Factory default configuration is used as a basis for further settings and therefore does not need to be reentered in order to change to another terminal adapter mode.

After setting the terminal adapter to the factory default condition, issue the appropriate following commands to enter the desired sync mode.

To enter Clear Channel Synchronous Mode

AT%A 2=0

To enter Synchronous Bonding Mode (enter both commands)

AT%A 2=3
AT@B 0=2

Ensure terminal adapter parameters are correct and return to the admin shell. ISDN terminal adapter 1 has been set up to automatically dial the stored number in register 0 when DTR is enabled (which occurs when the DOD connection is made e.g., pinging a remote host). ISDN terminal adapter 2 attached at the stored phone number location can be configured to answer the call by using the same command format but using AT&M 1 in place of AT&M 2.

AT&M 1   Synchronous Pause Mode - used at ISDN terminal adapter 2

Once you have typed these commands and entered synchronous mode, move the terminal adapter over to be attached to the WAN port on the port server.
Set up system and dial-on-demand configurations

We will set up a dial on demand configuration on the local port server (cuda1) and an automatic ppp session on the remote port server (cuda2).

1. Set up cuda1 WAN port for RS232 interface and internal clock by the following commands.

   set wan interface RS232
   set wan clock internal 128000
   set timeout wan 30

   This sets the baud rate to 128000. (ISDN terminal adapters support sync baud rates of 300-128000 bps.)

   **NOTE:** If you change the WAN interface or clock, save and reboot the system for the WAN to work with the following commands.

   save
   expert
   reboot

2. Set up cuda2 WAN port for RS232 interface, internal clock and set ppp custom configuration with the following commands. Note that this example is configuring an RCS/4000 with the WAN port numbered as 33. If you are configuring an RCS/5000 or RCS/6000, the WAN port is numbered 18.

   set wan interface RS232
   set wan clock internal 128000
   set ppp C2 33

   Enter remote IP address : 199.171.148.1
   Enter local IP address : 198.130.71.2
   Enter netmask in IP address format : 255.255.255.0
   Enter authentication type (chap|pap|none): none
   Use Van Jacobsen header compression (y/n): n
   Are these values correct (y/n): y

   **NOTE:** If you change the WAN interface or clock, save and reboot the system for the WAN to work with the following commands:

   save
   expert
   reboot

3. Connect the terminal adapters to the WAN ports using the RS-232 straight-through cables.

4. Add system sync1 to cuda1's database:

   add system sync1
   Enter the ports to communicate through : 33
   Enter the phone number to dial : ""
   Enter the login chat script : ""

5. Set cuda1's dial on demand to the selected port:

   set dod 1
   Enter system name : sync1
   Enter remote IP address : 198.130.71.2
   Enter local IP address : 199.171.148.1
6. Configure cuda1's routing table to add a route through the ppp connection.
   
   add route 198.130.71.0 199.171.148.1 net 0

7. Configure apollo's routing table to add a route through the ppp connection:
   
   route add 198.130.71.0 199.171.148.11 1

   **NOTE:** This command may vary depending on the OS running on apollo.

8. To start the dial-on-demand connection, ping the remote host (dino):
   
   ping 198.130.71.20

9. This will start dialing and establish a connection to the remote host through the remote port server. Upon connection the terminal adapter LEDs should display as follows, from left to right:
   
   - **VS:** Steady green
   - **DS:** Steady green
   - **CS:** Steady green
   - **TR:** Steady green
   - **RD:** Steady green
   - **TD:** Steady green

10. To complete the setup, add the proper routes on the remote systems.

11. Configure cuda2's routing table to add a route to cuda1 through ppp.
    
    add route 199.171.148.0 198.130.71.2 net 0

12. Configure dino's routing table to add a route to 148 through cuda2:
    
    route add 199.171.148.0 198.130.71.21 1

---

**Configuring the WAN port**

(**NOTE:** only some port server models have a synchronous WAN port. See your Hardware Manual to determine whether your port server provides a WAN port.)

The following example shows how to configure the WAN ports on 2 port servers with a pair of CSU/DSUs to set up remote routing over the WAN. The WAN ports in the example are being configured on the RCS/4000 (port number 33). If you are configuring a WAN port on an RCS/5000 or RCS/6000, the WAN port is port number 18.

Before you configure the WAN port, ask the following questions:

- **Who does the clocking?**

  Generally, a CSU/DSU, ISDN terminal adapter or an external FRAD provides the clocking, so the clock should be set as external when such a device is being used. If two port servers are directly connected by a WAN port cable and no external device is involved then internal clocking is used.
• **If clocking is done by the port server, what speed is required?**

The maximum speed for the WAN port is 2,048,000 bits/second. Unless an unacceptable error rate occurs at the speed due to the length of the cable or electrical interference, the speed should be set to 2,048,000.

• **IP address to set up your ppp interface?**

Since this is a point-to-point connection, you need to know your remote IP address and your local IP address for this interface. The remote IP address on one side will be the local IP address on the other.

• **Type of physical interface?**

The type of physical interface is determined by the CSU/DSU or cable used to connect to the port server. Check the specifications for the CSU/DSU or cable to determine the appropriate settings. The possibilities are:

<table>
<thead>
<tr>
<th>RS-232</th>
<th>EIA-530</th>
</tr>
</thead>
<tbody>
<tr>
<td>RS-422</td>
<td>V.35</td>
</tr>
</tbody>
</table>
Setup 1

Using the port server sync port to route packets between isolated networks as shown in the following WAN diagram:

Consider cuda2 (198.130.71.21) and cuda1 (199.171.148.11) are two port servers on two different networks. The two port servers are connected by means of a CSU (could be a modem eliminator or sync modems) which does the clocking.

**On the remote port server cuda2:**

Set up the WAN port for EIA-530 interface and external clock. If you change the WAN interface or clock, save and reboot the system.

```
[admin:1/2]>> set wan interface e530
[admin:1/3]>> set wan clock external
```

Setup ppp on the WAN port (make sure C9 - custom configuration - is not being used by any other ports; if so use any other custom configuration not being used). Select an IP address such that the local address is an address on the network.
Example:

Since cuda2 is on the 198.130.71.0 network the local address is 198.130.71.2, and since cuda1 is on the 199.171.148.0 network the IP address chosen is 199.171.148.1. This allows hosts on 198.130.71.0 net to access hosts on 199.171.148.1 network by using the ppp interface on cuda2.

As you will see, when configuring cuda1’s sync port the remote and local IP addresses will be the opposite of cuda2.

```
[admin:1/4]>> set ppp C9 33
Enter remote IP address : 199.171.148.1
Enter local IP address  : 198.130.71.2
Enter netmask in IP address format : 255.255.255.0
Enter authentication type (chap|pap|none): none
Use Van Jacobsen header compression (y/n) : n
Remote IP Address       : 199.171.148.1
Local IP Address        : 198.130.71.2
Subnet Mask             : 255.255.255.0
Authentication Type     : none
Van Jacobsen header compression : off
Are these values correct (y/n): y
```

On entering yes, ppp is started on the WAN interface. To set up routes for 199.171.148.0 network through the local ppp address:

```
[admin:1/5]>> add route 199.171.148.0 198.130.71.2 net 0
```

If you want to save the setup in the database for use during the next reboot, enter:

```
[admin:1/6]>> save
```

On the local port server cuda1:

Set up the WAN port for EIA-530 interface and external clock:

```
[admin:1/2]>> set wan interface e530
[admin:1/3]>> set wan clock external
```

Set up ppp on the WAN port (make sure C9 - custom configuration - is not being used by any other ports; if so, use any other custom configuration not being used):

```
[admin:1/4]>> set ppp C9 33
Enter remote IP address : 198.130.71.2
Enter local IP address  : 199.171.148.1
Enter netmask in IP address format : 255.255.255.0
Enter authentication type (chap|pap|none): none
Use Van Jacobsen header compression (y/n): n
Remote IP Address       : 198.130.71.2
Local IP Address        : 199.171.148.1
Subnet Mask             : 255.255.255.0
Authentication Type     : none
Van Jacobsen header compression : off
Are these values correct (y/n): y
```
To set up routes for 198.130.71.0 network through the local ppp address:

```
[admin:1/5]>> add route 198.130.71.0 199.171.148.1 net 0
[admin:1/6]>> save
[admin:1/7]>> expert
[expert:1/8]>> reboot
```

All other hosts on the 198.130.71.0 network should be setup to route to 198.130.71.2 to access any hosts on 199.171.148.0 network. Example:

```
dino# route add 199.171.148.0 198.130.71.2 1
add net 199.171.148.0: gateway 198.130.71.2
```

Similarly, all hosts on the 199.171.148.0 network should be set up to route to 199.171.148.1 to access any hosts on 198.130.71.0 network.

```
apollo> route add 198.130.71.0 199.171.148.1 1
add net 198.130.71.0: gateway 199.171.148.1
```

Setup 2

Consider the same diagram, but instead of a CSU we use a cable (pinout shown in the diagram below) to connect the two port server sync ports. Most of the setup will be the same, except the configuration part related to clock:

- The clock has to be internal, i.e., supplied by the port server. The port server functions reliably up to 2,048 Mbits/s.
- On both the port servers enter the following different command from the above example for an internal clock of 56K baud:

```
[admin:1/2]>> set wan clock internal 56000
```
The rest of the setup is the same.

2 -------------------- 3
14 ----------------------- 16

3 -------------------- 2
16 ----------------------- 14

4 -------------------- 5
19 ----------------------- 13

5 -------------------- 4
13 ----------------------- 19

6 -------------------- 20, 8
22 ----------------------- 23, 10

7

8, 20 -------------------- 6
23, 10 ------------------ 22

15, 24 -------------------- 17
11, 12 ------------------ 9

17 -------------------- 24, 15
9 ---------------------- 11, 12

EIA-530 Synch Back-to-back Cable

Printing to a serial or parallel port

Connecting a printer to the port server

*Parallel printer*

1. Connect printer to the parallel printer port on the port server using a standard printer cable.

2. Enter system administration mode, then expert mode:

   1/1>> admin
   Password: admin
   [admin:1/1]>> expert

3. Issue printer test command to check the connection (the parallel port is port 17 for RCS/ 5000 or RCS/ 6000 or port 32 for the RCS/ 4000.):

   [expert:1/6]>> prtr_test /dev/tty/32

This should print 10 lines of output.
Serial printer

1. Connect printer to a serial port on the port server (assume port 4 for this example).

2. Enter system administration mode, then expert mode:
   
   1/1>> admin  
   Password: admin  
   [admin:1/1]>> expert

3. Configure the serial port characteristics to match the printer. The default port server port characteristics are:
   
   - 9600 baud
   - 8 bits/ character
   - 1 stop bit
   - no parity
   - xon/ xoff flow control

   To change these characteristics, use the following commands:

   set speed 4 19200
   set size 4 7
   set parity 4 even
   set stop 4 2
   set output 4 rtscts

   Save any changes to the configuration:

   save

4. Make sure rtelnet is set on the serial port and that the default service configuration has been deleted.

   set rt 4
   delete def 4

   Use show port 4 to verify that the configuration is set to rtelnet and no default is set.

5. Issue the printer test command to check the connection:

   [expert:1/6]>> prtr_test /dev/tty/4

   This should print 10 lines of output. See the section, Troubleshooting the serial printer, page 52 if the output does not appear.

Print Options

The port server supports two different protocols for sending output to a printer:

- LPD

   The LPD protocol is the UNIX Line Printer Daemon protocol and is supported by all UNIX systems and Windows/NT.
• Reverse Telnet

Reverse Telnet is a protocol that will allow a port on the port server to appear as a local device connected to a host on the network.

Selecting a Print Option

The following table compares the two print options: reverse telnet and the UNIX line printer daemon (lpd).

<table>
<thead>
<tr>
<th>Reverse telnet</th>
<th>lpd</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setup requires more steps.</td>
<td>Minimal setup on port server unless using printer pool or serial ports; minimal setup on hosts.</td>
</tr>
<tr>
<td>Less flexibility with multiple hosts. Denies connection if a shared printer is busy (or all printers in a pool are busy).</td>
<td>Manages sharing of port or printer pool so that multiple hosts can queue print jobs.</td>
</tr>
<tr>
<td>Allows host to enforce security.</td>
<td>Allows any host on the network to connect and print.</td>
</tr>
<tr>
<td>Host maintains accounting information.</td>
<td>Maintains no accounting information.</td>
</tr>
</tbody>
</table>

**Use this option if ...**

- You need security and accounting.
- You want the port server to act as a print server.

Printing using LPD

The port server can serve as a print server on a network. Print jobs can be sent to the port server from any host on the network that supports the LPD protocol. This includes all UNIX systems and Windows/NT.

**On the port server:**

No setup is required to use a printer connected to the parallel port. For a printer connected to a serial port, the default service on the port must be deleted and the serial port must be configured to match the printer configuration (baud rate, flow control, ...). A printer pool may also be used with lpd. Setting up a printer pool is described below.

**NOTE:** If you do not delete the default service on a serial port, the default service will control the port and lpd will be unable to access it. Use:

```
  delete default {portnum}
```

to configure a serial port for use with lpd.
On the host system:

The host system must be configured to use a remote printer. The steps involved vary depending upon the specific host system. However, in all cases the following information is required:

- System name or IP address of the port server
- Remote print queue name.

Use the print queue name for two purposes:

- Selection of ports for the print job
- Specification of default treatment of the data file.

For examples, see the Sample configurations page 55 at the end of this section.

Specifying port server print queue names

Use any name in either upper or lower case letters followed by the port number. The first letter of the printer name selects the default treatment of the data file. ‘f’ means that the data will be formatted text. Anything else means binary data. Serial ports are numbers 0–15. The parallel port is number 17 or 32, depending on the port server model.

The following special names are recognized:

- **ppoolN** Use any port from printer pool N
- **fppoolN** Use any port from printer pool N
- **lpt** Parallel port
- **f1pt** Parallel port
- **printer** Parallel port
- **fprinter** Parallel port

Any other name ending in a sequence of digits is taken to mean the port whose number is specified by the digits.

Example

To specify a queue name selecting serial port 3, you could use any of the following:

- **port3**
- **/dev/tty/3**
- **3**
- **f0003**
- **my_favorite_port_is_3**
Specifying default data file format

A print job consists of control information and the data to print. The control information specifies how to print the data, including page parameters, fonts and data format.

The port server supports printing of formatted or unformatted data. Unformatted data includes binary and PostScript.

The lpd protocol allows the host to send either the data or the control information first.

If the port server receives the control information before the data file, it uses that information to format the data file.

If it receives the data file before the control information, it prints the file without the control information formatting and uses default file formatting. The host can control the default formatting with the queue name it uses (see the next section).

If you are using the port server in a UNIX environment, the data file is usually sent before the control information. If you are printing both formatted and unformatted data, specify two queue names: one to support formatted data and the other to support unformatted data.

If you are using the port server in a Windows NT environment, the control information is sent before the data file. Specify one queue name for your print jobs.

For examples, see the Sample configurations page 55 at the end of this section.

Managing the print queue on the port server

Once you have set up lpd, you can perform additional tasks on the port server, using the following commands:

```
lpq      Display the print queue
lpqm n or lprm all  Delete job(s) from print queue
```

**NOTE:** You should delete print jobs from the host. The host may resend print jobs if you delete them from the port server.

You may repeat these commands as many times as necessary.

Sample configurations

This section describes setting up a remote printer on an RCS/ 4000 with lpd in both the UNIX (AIX 4 or SunOS or DEC) and Windows NT and 95 environments. If you are setting up an RCS/ 5000 or RCS/ 6000, the parallel port is port 17, rather than port 32.

**UNIX environment (AIX 4)**

To set up a remote printer with an AIX driver to print formatted text, do the following:

1. Log onto your system as root.
2. Type `smit printer` to start the System Management Interface Tool (SMIT).
3. From the SMIT main menu, select **Print Spooling**.
4. From the Print Spooling menu, select **Add a Print Queue**.
5. From the Add a Print Queue menu, select **Remote — Printer Attached to Remote Host**.
6. From the **Remote Printer Attached to Remote Host** menu, select standard processing.
7. From the **Add a Standard Remote Print Queue** dialog, fill in the following and click **OK**.
   
   Name of QUEUE to Add: **linda**
   
   HOSTNAME of remote server: **port server-host**
   
   Name of QUEUE on remote server: **fport32**
   
   where:
   
   linda: print queue name on host
   
   port server-host: name or IP address of the port server
   
   fport32: send formatted job to parallel port
   
8. The COMMAND STATUS dialog displays with the message: Added print queue ‘linda’
9. Press **F10** to exit the SMIT program.

**UNIX environment (SunOS or DEC)**

To set up a remote printer for Sun OS or DEC Alpha:

1. Add an entry for the printer to the **/etc/printcap** file.
2. Create the spool directory.
3. Specify maximum size for file to be printed.

If you are printing both formatted and unformatted (binary or PostScript) data, set up a printer queue name for each.

**Example— unformatted text**

```
tom_lpt  tom lpd cuda test printer:\
   :rm=cuda29:rp=printer:\
   :sd=/var/spool/lpd.tom_lpt:\
   :mx#0:
```

where:

- **tom_lpt**: queue name for Sun OS
- **rm=cuda29**: remote machine (name or IP address of the port server) in / etc/ hosts file
- **rp=printer\**: queue name for cuda 29 machine
- **sd=/var/spool/lpd.tom_lpt\**: set directory name for spooling
- **mx#0**: no maximum size for file to be printed
Example—formatted text

tom_fmt   tom lpd cuda test fprinter:\
:rm=cuda29:rp=fprinter:\
:sd=/var/spool/lpd.tom_fmt:\
:mx#0:

where:

tom_fmt          queue name for Sun OS
rm=cuda29:       remote machine (name or IP address of the port server) in
                 / etc/ hosts file
rp=fprinter:\   queue name for cuda 29 machine
sd=/var/spool/lpd.tom_fmt:\ `set directory name for spooling
mx#0:            no maximum size for file to be printed

Windows NT 3.51 environment

Before setting up a remote printer in the Microsoft Windows NT 3.51 environment, make
sure you have installed Microsoft TCP/ IP printing support.

To set up a remote printer in the Windows NT 3.51 environment, do the following:

1. From the Windows NT Program Manager dialog box, open the Main group icon.

2. From the Main dialog box, open the Print Manager icon.

3. From the Print Manager dialog box, select the Printer pull-down menu.

4. From the Printer pull-down menu, select Create Printer.

5. From the Create Printer dialog box:
   Enter a printer name
   Select a printer driver
   Enter a description
   Select “Other” for Print To.

6. From the Print Destinations pop-up dialog box, select LPR Port.
   Click OK.

   NOTE: If the LPR Port does not appear on the dialog box, make sure you have installed
   Microsoft TCP/IP printing support.

7. From the LPR Compatible Printer pop-up dialog box, enter:
   Name or IP address of the port server.
   Name of the printer on that machine (port n - where n is the port server port number, e.g.
   port6).
   Click OK.
**Windows NT 4.0 environment**

Before setting up a remote printer in the Microsoft Windows NT 4.0 environment, make sure you have installed Microsoft TCP/IP printing support.

To set up a remote printer in the Windows NT 4.0 environment, do the following:

1. From the Windows NT 4.0 desktop, click **Start** and select **Settings** and **Printers** from the pop-up menus.
2. From the **Printers** dialog box, double click on **Add Printer**.
3. From the **Add a Printer Wizard** dialog box, leave **My Computer** selected. Click **Next**.
4. From the **Add Port** dialog box, select **LPR Port** and click **New Port**.
   
   **NOTE:** If LPR Port option does not exist it means Microsoft TCP/IP Printing is not installed. To install, click **Start**, **Control Panel**, select **Network and Services**, highlight **Server/workstation** and click **Add**. From the list choose Microsoft TCP/IP Printing. This will require your installation CD.

5. From the **New Port** dialog box, enter:
   
   - Name or IP address of the port server
   - Name of the printer on that machine (portn, where n is the port server port number, e.g., port6).
   
   Click **Close**.

6. From the **Add a Printer** dialog box, select the port you added and click **Next**.
7. From the remaining dialog boxes, select or enter the following:
   
   - Printer driver
   - Printer name
   - Printer Shared or Not Shared
   - Print test page.
   
   Click **Next** after each entry.

**Windows 95 environment**

Before setting up a remote printer in the Microsoft Windows 95 environment, make sure you have installed it first in the Windows NT environment. An NT server is required.

To set up a remote printer in the Windows 95 environment, do the following:

1. From the Windows 95 desktop, click **Start** and select **Settings** and **Printers** from the pop-up menus.
2. Double click on **Add Printer** and click **Next** to open the **Add Printer Wizard** dialog box.
3. Select **Network Printer** and click **Next**.
4. From the **Network path or queue name** dialog box, enter a UNC pathname to the network printer, as follows:

```
\\servername\printername
```

or click **Browse** to select a server and then a printer. Click OK and click **Next**.

5. From the **Enter Printer Name** dialog box, enter a printer name and click **Next** and click **Finish**.

### Printing using reverse telnet (rtn)

To print using reverse telnet:

**On the port server:**

1. Enter system administration mode:
   
   `> admin`
   
   Password: admin

2. Set up rtelnet on the printer port and write down the tcp port number:
   
   `[admin:1/1]>> set rtelnet 32`
   
   `port32 : tcp ports 9032 and 8032`

3. Save the configuration:
   
   `[admin:1/2]>> save`

**On UNIX host systems (except AIX):**

To print from a remote host, use the supplied **rtn** application for that host. The following example is for a Motorola SVR4 system:

1. Start the rtn application, replacing the IP `198.130.71.113` with your port server IP address, and the tcp port # from above:
   
   `$rtn -d/dev/remote_lp -h198.130.71.113 -p9032 &`

2. You can test the connection by directing a file to the device:
   
   `$ cat file_name > /dev/remote_lp`

3. Refer to your system administration guide for adding a printer.

**On AIX host systems**

Setting up an rtn-based printer under AIX is a three step process: 1) If you have not already done so, set up an port server parent device; 2) Add an port server printer device; and 3) Add an port server print queue for the printer device.

1. To set up an port server parent device:
   
   1.1. Log onto your system as root.
1.2. Type `smit port server`.

1.3. Choose **Add port server to configuration**.

1.4. From the **Add port server to Configuration** menu, enter the port server type and host name or IP address:

```
port server Type: port server 8-port terminal server
IP Address or Host Name: [leroy]
```

1.5. Click on **OK** to add the port server.

2. To add an port server printer device on AIX:

2.1. Type `smit printer` to start the System Management Interface Tool (SMIT).

2.2. Click on **Printer/Plotter Devices** and **Add a Printer/Plotter**.

2.3. From the **Add a Printer/Plotter** menu, select one of the following.

   For serial printers, select the appropriate serial printer.
   
   For parallel printers, select the **rtn port server Parallel Port** option.

2.4. Select the appropriate interface (RS-232 or parallel.rtn).

2.5. From the **Parent Adapter** menu, click on **port server0 Available leroy port server Terminal Server (8-port)**.

2.6. Press **F4** to list the port numbers or enter the port server TCP port number and click on **OK** to add port server parallel printer.

3. To add a print queue for the port server:

3.1. Type `smit printer` to start the System Management Interface Tool (SMIT).

3.2. From the **Printer/Plotter** menu, click on **Print Spooling**.

3.3. From the **Print Spooling** menu, click on **Add a Print Queue**.

3.4. From the **Add a Print Queue** menu:

   For serial printers, select the **local Printer Attached to Local Host** attachment type.
   
   For parallel printers, select the **port server4000 Printer Attached to port server Parallel port** attachment type.

   Then select the Printer Type.

3.5. Select the appropriate port server printer device and supply the necessary queue information to add the printer queue.
Printing to the serial port(s) with PPOOL

The following example configures three serial printers attached to port 3, 4, & 6 to a printer pool (ppool 1) on the port server.

On the port server:

1. Enter system administration mode:
   
   1/1>> admin  
   Password: admin

2. Set up serial port 3, 4, & 6 on printer pool 1:
   
   [admin:1/1]>> set ppool 3-4,6 1  
   port3 added to ppool1  
   port4 added to ppool1  
   port6 added to ppool1

3. To determine what ports are in ppool 1:
   
   [admin:1/2]>> show ppool 1  
   ppool1  
   -------  
   port3  
   port4  
   port6

4. Get the tcp port number for ppool 1: (For a discussion of tcp port numbers, see the section, RTELNET, page [153])

   [admin:1/3]>> show rtelnet  
   pool: ppool1  tcp port: 9106/8106

5. Save configuration:

   [admin:1/2]>> save

On the host system:

On the remote host to print from, use the supplied rtn application for your appropriate system. The following example is for a Motorola SVR4 system:

1. Start the rtn application replacing the IP 198.130.71.113 with your port server IP address, and the tcp port # from above:

   $ rtn -d/dev/remote_lp -h198.130.71.113 -p9106 &

   NOTE: This connects to the first available printer in printer pool 1.

2. Test the connection by directing a file to the device:

   $ cat file_name > /dev/remote_lp

3. Refer to your system administration guide for adding a printer.
Troubleshooting the serial printer

The configuration of a serial printer must match the port configuration on the port server. Mismatches in this area are the most common cause of problems. Make sure the baud rate, character size, parity, and flow control settings all match.

No output

The most common reasons for no printer output are configuration errors or a cabling problem. To check for configuration problems, type:

```
show port n
```

The settings for port number n will be displayed. Check that the baud rate, parity, etc. match the settings for the printer. The configuration section of the output should not show default. If default shows in the configuration, you did not delete the default configuration from the port and the printer will not work correctly. To remove the default configuration from the port, enter:

```
delete def n
```

where n is the number of the port.

If the configuration settings appear to be correct, the problem may be with the cable. Try a null modem connector between the cable and the printer to see if this corrects the problem.

Lost data

Losing data that is sent to the printer is usually caused by the lack of flow control or a mismatch in the flow control settings between the printer and the port server. Be sure that the printer flow control is enabled (either RTS/CTS or XON/XOFF) and that the same type of flow control is enabled on the port server port.

Output suddenly stops

If your printer prints some output (maybe a page or so) then suddenly stops, your printer may be using the DTR signal for flow control. Most DTE serial cables map the DTR signal to the DCD input signal on the port server serial port connector. If the DCD signal is dropped, the port server interprets this as a hangup and closes the connection, terminating the print job.

This problem is best corrected by configuring the printer to not use DTR for flow control. If this is not possible, a cabling change can be made to have DCD wired high. This may mean that the port server will not be able to determine when the printer has been turned off. In this case, the port server cannot report the correct printer status and may try indefinitely to print on this port.
Updating Port Server Software

To upgrade port server software, do the following:

1. Make sure you have the following:
   - port server connected to a TFTP server via LAN or WAN
   - IP address of TFTP server
   - Upgrade file from Systech (assume it is called port server4000_new.bdnl) located on server in the default TFTP directory
   - Server running TFTP

2. Log onto the port server using either a direct terminal connection or Telnet. The screen displays the login banner and a password prompt.

3. Enter the password. The default password is user.

4. Enter admin to enter the administration mode.

5. Enter the password for the administrative mode. The default is admin.

6. Enter expert to enter the expert mode.

7. Enter flash_update. The screen displays the prompt:
   filename of download image:

8. Enter the name of the upgrade file such as port server4000_new.bdnl. The screen displays the prompt:
   server name or IP address (return for default):

9. Enter the IP address of the server. The screen displays a series of dots as the file loads, then the prompt:
   do you wish to reboot now? (y/n)

10. Enter y and press Enter to reboot the port server so that the new software takes effect.

   NOTE: The server may restrict access via TFTP to a particular directory such as /tftpboot. If this is the case, then the upgrade file from Systech must be copied into that directory. The filename given in response to the prompt “filename of download image:” is relative to the default tftp directory, so “/ tftpboot/” should not be entered as part of the name.

Copying Configurations

This section describes copying configurations from the port server to a server (uploading) and from the server to the port server (downloading). For more information, see the \texttt{flash\_save} and \texttt{flash\_update} commands, page \pageref{flash_update}.
Requirements

Before copying configurations, make sure you have the following:

- port server configured as desired
- IP address obtained from a BOOTP server instead of with set IP command (Enter show ip to verify this)
- Server running TFTP server
- IP address of server
- BOOTP server on the network configured for all the units to be configured

Uploading a configuration to the server

To save (upload) a configuration to the server, do the following:

1. Prepare the server to receive a file. This step is necessary if the TFTP server is configured for security. (For the Solaris version of TFTPD, this is the -s option.) Do the following:
   - cd to the default TFTP directory: "cd /tftpboot"
   - create an empty file to receive the config "touch my_config.bdnl"
   - make the file writeable by everyone "chmod 666 my_config.bdnl"

2. Log in to the port server using a direct terminal connection or telnet. The screen displays the login banner and a password prompt.

3. Enter the password. The default password is user.

4. Enter admin to enter the administration mode.

5. Enter the password for the administration mode. The default password is admin.

6. Enter expert to enter the expert mode.

7. Enter flash_save. The screen displays the prompt:
   - Selection:

8. Enter c and press Enter. The screen displays the prompt:
   - file name:

9. Enter a name for the configuration, for example my_config.bdnl. The screen displays the prompt:
   - server name or IP address (return for default):

10. Enter the IP address of the server, for example 198.130.71.12. The screen displays some dots followed by the message:
    - save complete.
Downloading a configuration from the server

To download a configuration from the server to the port server, do the following:

1. Log in to the port server using a direct terminal connection or telnet. The screen displays the login banner and a password prompt.
2. Enter the password. The default password is **user**.
3. Enter **admin** to enter the administration mode.
4. Enter the password for the administration mode. The default password is **admin**.
5. Enter **expert** to enter the expert mode.
6. Enter **flash_update**. The screen displays the prompt:
   
   filename of download image:

7. Enter the name of the configuration, for example **my_config.bdnl**. The screen displays the prompt

   server name or IP address (return for default):

8. Enter the IP address of the server, for example **198.130.71.12**. The screen displays some dots and status messages followed by the prompt

   do you wish to reboot now? (y/n)

9. Enter **y** and press **Enter** for the new configuration to take effect immediately.

Repeat Steps 1–9 for each port server receiving a copy of the configuration.

Assigning an IP address with a UNIX BOOTP server

BOOTP provides an address discovery feature; however, before it can "discover" the port server's IP address through a BOOTP request, you must configure BOOTP.

1. Verify that both the BOOTP daemon program, bootpd, and the database file, /etc/bootptab, reside on the boot host. If a bootpd executable file is not resident on your boot host, copy the bootpd sources from a free distribution package (such as GNU), compile it, and install it on your host in the /etc directory.

2. Enable BOOTP by:

   2.1. Entering a ps command to find out if bootpd is running. For example:

       # ps -axc | grep bootp

   On some systems, the command is:

       # ps -ef | grep bootp

   2.2. Host operating systems based on the 4.3 BSD UNIX operating system (such as ULTRIX and SunOS) require an entry in the /etc/inetd.conf file for BOOTP. Often these entries exist, but are rendered ineffective by a pound sign (#) at the beginning of the entry. If this is the case, remove the #. Some example entries are:

   For ULTRIX:
bootp dgram udp wait /usr/etc/bootpd bootpd -i

For SunOS:

bootps dgram udp wait root /usr/etc/bootpd bootpd

2.3. Ensure that TCP/IP ports are reserved for the bootpd server and client processes in the /etc/services file.

The usual entries in /etc/services are:

bootps 67/udp
bootpc 68/udp

Remove any pound signs (#) at the beginning of these entries.

2.4. If you make any changes in the /etc/inetd.conf or /etc/services file, restart the inetd daemon.

To restart the daemon, find the inetd process ID and send a hangup signal as shown in the following example. This causes the daemon to re-read the configuration file and use the new information.

# ps -axc | grep inetd
17601 ? I 0:12 inetd
# kill -HUP 17601

On some systems, the command to find the process ID is

# ps -ef | grep inetd

3. Next, you must add information about the port server to the /etc/bootptab database file. Bootptab files vary for different systems; however, as an example, one version of a bootptab file is shown below. Note that abbreviations, definitions, and other information you will need to modify the file are listed at the beginning of the file.

FILE: /etc/bootptab.3.1.0

# ns — IEN-116 name servers
# rl — resource location protocol servers
# sm — subnet mask
# tc — template host (points to similar host entry)
# to — time offset (seconds)
# ts — time servers

# Be careful to include backslashes where needed. Weird
# (bad) things can happen when a backslash is omitted.
#
# First, we define a global entry which specifies the stuff
# every host uses.

quicc00:\
:ht=1:ha=123456789044:ip=199.171.148.29:hn:\
:sm=255.255.255.0:\
:hd=/tftpboot:bf=patxinu.bdnl:bs=auto:\
:ds=199.171.148.12 199.171.148.6:\
:ns=0x80020b4d 0x80020ffd:\
:ts=199.171.148.12:\
:to=0:

The general rules for bootptab file entries are:

- A colon (:) usually indicates the end of a field, and a backslash (\) indicates that the
  entry is continued on the next line.
- Spaces are not permitted between the characters on a line.
- Fields consist of a tag followed by a variable. Each tag identifies a separate parameter.
- The Ethernet address has no internal punctuation, such as periods or dashes.

The typical file structure is one or more template entries, containing information common to
all port servers or a group of port servers, followed by individual entries, each containing
information about a specific port server. For example:

# Template entry
template.name:\
 tag=value:/
 .:

 tag=value:

# Entry for port server4000
hostname:\
 port server=template.name:\
 tag=value:/
 .:

 tag=value:

The tags required for each template file created for an individual port server depend on the
intended use/application of that particular port server. For example, if you are installing an
port server that will not access a gateway, the gateway address(es) tag, gw, need not be used.
As mentioned previously, bootptab files vary for different host systems. A sample bootptab template file for an port server identified as port server29, formatted for the SunOS, follows:

```
port server29:
    :ht=1:ha=0080440c028f:ip=199.171.148.148:hn:
    :sm=255.255.255.0:
    :ds=199.171.148.12 199.171.148.6:
    :ns=0x80020b4d 0x80020ffd:
    :ts=199.171.148.12:
    :to=0:
```

### Assigning an IP Address with Windows NT DHCP Server

The BOOTP/ DHCP tags (from RFC 2132) which are used by the port server are as follows:

<table>
<thead>
<tr>
<th>No.</th>
<th>Tag</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Subnet Mask</td>
</tr>
<tr>
<td>2</td>
<td>Time Offset</td>
</tr>
<tr>
<td>3</td>
<td>Router</td>
</tr>
<tr>
<td>4</td>
<td>Time Server</td>
</tr>
<tr>
<td>6</td>
<td>Domain Name Server</td>
</tr>
<tr>
<td>15</td>
<td>Domain Name</td>
</tr>
<tr>
<td>51</td>
<td>IP Address Lease Time</td>
</tr>
<tr>
<td>54</td>
<td>Server Identifier</td>
</tr>
</tbody>
</table>
Chapter 4: Configuring Ports

The port server provides either 2, 4, 8, or 16 serial ports, depending on the model. The port server commands specify individual serial ports by number. The RJ-45 or DB-9 connector for each serial port on the port server is labeled with the number of the port. On RCS/4000 models, the port numbers begin with 0; on all other models, the first serial port is port 1 (the modem on the POS/2489 is port 0). For hardware diagrams and pinout information, see the hardware manual for the model of port server that you are using. Some models of the port server also provide a synchronous WAN port, a modem port, and/or a parallel port.

Serial port characteristics and services are configurable. The serial port configuration specifies the serial port characteristics; the service configuration specifies the configured activities for the serial port.

If you are using NativeCOM, the configuration is under the control of the host computer and you do not generally need to configure the port server directly, with a few exceptions (see NativeCOM, page 13). If you are not using NativeCOM, the port server can be configured using a web browser or using the command line interface.

The serial port characteristics that can be changed are:

- speed
- character size
- stop bits
- parity
- timeout interval
- ignore/require DCD modem signal
- input and output flow control
- physical interface type (e.g., RS-232, RS-485) - on some models only
- modem initialization string and chat script
- designated character for switching between sessions
- two strings for labeling the port owner and the port description
- power (some models)
- modem signal mappings (some models)

The WAN port characteristics that can be changed are:

- physical interface type (e.g., EIA-232, EIA-422)
- type of clock (e.g., external, internal)

The services that can be configured are:
Configuration Methods

If you use NativeCOM, the host computer handles the port configuration and you do not generally need to configure the ports manually, with a few exceptions (see NativeCOM, page 13). If you are not using NativeCOM, there are two ways to manually configure the serial ports on your port server: using your web browser or using the command line interface. Both web browser and command line interface information is provided in this chapter.

Web Browser

You can configure most of the port characteristics and services using the port server web browser interface. To configure the port server using the web browser, point your browser at the IP address of the port server.

Command Line Interface

You can configure your port server directly using the command line interface, rather than a browser. To configure the port server using the command line interface, connect to the port server via telnet or connect a terminal to a serial port on the port server.

Before you can telnet to a port server, the port server must be assigned an IP address. For information on assigning an IP address to a port server, see Assigning an IP Address, page 17.

To access the command line via a terminal, connect the terminal to a serial port. The login screen displays a Welcome banner and a password prompt. The factory-supplied user password is user;
Welcome to the RCS/4000 Remote Communications Server
Password:user

You will see the system prompt: (>>). Most of the commands that you use to configure ports are administrative commands, requiring you to be in administrative mode. To enter administrative mode, type the admin command. The factory-supplied administrative password is admin.

>>admin
Password: admin
admin>>

**Serial Port Characteristics**

The port server is delivered with default characteristic settings. Serial port characteristics that can be changed are:

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>speed</td>
<td>9600</td>
</tr>
<tr>
<td>character size</td>
<td>8</td>
</tr>
<tr>
<td>stop bits</td>
<td>1</td>
</tr>
<tr>
<td>parity</td>
<td>none</td>
</tr>
<tr>
<td>timeout interval</td>
<td>0</td>
</tr>
<tr>
<td>ignore/require DCD modem signal</td>
<td>require (no)</td>
</tr>
<tr>
<td>input flow control</td>
<td>xoff</td>
</tr>
<tr>
<td>output flow control</td>
<td>xon</td>
</tr>
<tr>
<td>physical interface type</td>
<td></td>
</tr>
<tr>
<td>most ports</td>
<td></td>
</tr>
<tr>
<td>ports 4, 5, 6, 7 on RCS/4000 models with optional RS-422/485 mode</td>
<td>RS-232</td>
</tr>
<tr>
<td></td>
<td>RS-485</td>
</tr>
<tr>
<td>Modem configuration (initialization string and chat script)</td>
<td>generic</td>
</tr>
<tr>
<td>designated character for switching between sessions</td>
<td>BREAK</td>
</tr>
<tr>
<td>port owner (user-defined string)</td>
<td>blank</td>
</tr>
<tr>
<td>port description (user-defined string)</td>
<td>blank</td>
</tr>
<tr>
<td>power (POS/2000 only)</td>
<td>none</td>
</tr>
<tr>
<td>pin1 definition (RCS/4000 &amp; RCS/5000 only)</td>
<td>DCD</td>
</tr>
</tbody>
</table>

Serial port characteristics (with the exception of the physical interface) can be configured in one of three ways:
• **Individual port characteristic settings**: You can set individual port characteristics, one component (e.g., speed, parity, size) at a time. Either the web browser interface or the command line interface can be used to set individual port characteristics.

• **Predefined serial port configurations**: You can configure all the port settings in one command, using one of the twelve predefined serial port configurations that are available. You can apply the configuration to one or more ports but cannot modify the configuration. The predefined serial port configurations are available only through the command line interface.

• **Modifiable serial port configurations**: You can configure all the port settings in one command, using one of the three modifiable serial port configurations that are available. You can define the configuration with a combination of settings, save it, and apply it to one or more ports whenever the defined configuration is needed. Modifiable serial port configurations are available only through the command line interface; however, you can copy the configuration on one serial port to another serial port using the web browser.

**Note:** The physical interface on the RCS/4000 is not configured using software. It is configured by opening the port server and changing the jumper settings. (See the RCS/4000 Hardware Manual for instructions.)

**Setting/Modifying Individual Serial Port Characteristics**

Serial port characteristics can be set individually using the **Port Settings** options displayed by the web browser or using the **set** command through the command line interface. Table 4-1 shows the individual characteristics that can be set and describes the options for each.

The **set** command is used in conjunction with various keywords and parameters. Table 4-1 shows the keywords and their associated parameters that can be used with the **set** command to modify serial port characteristics. Table 4-1 also shows the page number of the complete description of the keyword in the reference section.

The format of the **set** commands is:

```
set keyword portnumber(s) parameter i
```

For example, the command to set the speed to 2400 for port 3 is:

```
set speed 3 2400
```

and, the command to set the parity to none on port 3, port 4, port 5, and port 6 is:

```
set parity 3-6 none
```

**Table 4-1. Options, Keywords and Parameters**

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>keyword</th>
<th>parameters (select one)</th>
<th>Description</th>
</tr>
</thead>
</table>

72   Serial Port Characteristics
| **physical interface type** | Interface (page 121) | rs232  
rs422  
rs485  
cloop | sets the port to the specified electrical interface. (Must be set in hardware for RCS/4000.) (cloop available only on POS/2000.) |
|---------------------------|----------------------|-----------------------------------------------|
| **ignore/ require DCD modem signal** | Ignoredcd (page 118) | yes  
n | yes configures the port to ignore DCD on session startup and to not care about the loss of DCD. no configures the port to require the DCD signal to start and maintain the service. |
| **input flow control** | Input (page 119) | none  
xoff  
rtscs  
dtrdsr | sets software and/or hardware input flow control for the port |
| **Modem configuration (initialization string and chat script)** | Modem (page 126) | modem name | applies a user-defined modem definition, consisting of a modem chat script and an initialization string, to a port(s). A modem definition is created using the add modem command. A modem definition, named generic, is factory enabled. |
| **output flow control** | Output (page 133) | none  
xon  
xany  
rtscs  
dtrdsr | sets software and/or hardware output flow control for the port |
| **port owner (user-defined string)** | Owner (page 135) | character string | used by host reverse telnet software (such as NativeCOM) to reserve a port. The string may be any value, but is generally set to the hostname and host port name. |
| **parity** | Parity (page 136) | even  
odd  
none | sets port to even, odd, or no parity |
<p>| <strong>port description (user-defined string)</strong> | Portdesc (page 140) | character string | enters a user-specified string that allows the user to identify the port. The string may be any value, up to 256 characters. |</p>
<table>
<thead>
<tr>
<th>Serial Port Characteristics</th>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>power</strong> <em>(POS/2000 only)</em></td>
<td><strong>power</strong> <em>(page 141)</em></td>
<td>none, 5, 12</td>
</tr>
<tr>
<td><strong>character size</strong></td>
<td><strong>size</strong> <em>(page 155)</em></td>
<td>5, 6, 7, 8</td>
</tr>
<tr>
<td><strong>speed</strong></td>
<td><strong>speed</strong> <em>(page 160)</em></td>
<td>50; 75; 110; 134; 150; 200; 300; 600; 1200; 1800; 2400; 4800; 9600; 14400; 19200; 28800; 38400; 57600; 115200</td>
</tr>
<tr>
<td><strong>stop bits</strong></td>
<td><strong>stop</strong> <em>(page 161)</em></td>
<td>1, 2</td>
</tr>
<tr>
<td><strong>designated character for switching between sessions</strong></td>
<td><strong>switch</strong> <em>(page 162)</em></td>
<td>ASCII code</td>
</tr>
<tr>
<td><strong>timeout interval</strong></td>
<td><strong>timeout</strong> <em>(page 167)</em></td>
<td>number of seconds</td>
</tr>
</tbody>
</table>

With any of the above keywords, you have the option of specifying in the command—using the i option—that the configuration settings take effect immediately. However, whether you specify that the settings are to take effect immediately or not, you must save the settings *(see the save command, page 232)* for them to be maintained following reboot. If you do not use the i option, the settings will take effect the next time a service or operation starts on the specified port(s).

You can display the current settings using the show command with a keyword and port number(s). For example:

    show speed 3
will display the speed setting for port 3, and
    show input all
will display the input flow control setting for each serial port.

Predefined Serial Port Configurations

There are 12 predefined port configurations that you can apply to one or more ports. If there is a predefined configuration that meets your needs, you can apply it with one command, changing all the settings at once, rather than setting each port characteristic individually. The predefined configurations are described in Table 4-2. You can apply any configurations to any serial port. You can apply the same predefined configuration to more than one port.

Use the set port command (see port, page 139) to apply a configuration to one or more ports. Because the 12 configurations cannot be modified in any way, there is no add port command associated with them.

Table 4-2. Predefined Serial Port Configurations

<table>
<thead>
<tr>
<th>Name</th>
<th>Speed</th>
<th>Character Size</th>
<th>Stopbits</th>
<th>Parity</th>
<th>Input Processing</th>
<th>Output Processing</th>
</tr>
</thead>
<tbody>
<tr>
<td>9600n</td>
<td>9600</td>
<td>8</td>
<td>1</td>
<td>none</td>
<td>xoff</td>
<td>xon</td>
</tr>
<tr>
<td>9600e</td>
<td>9600</td>
<td>7</td>
<td>1</td>
<td>even</td>
<td>xoff</td>
<td>xon</td>
</tr>
<tr>
<td>9600o</td>
<td>9600</td>
<td>7</td>
<td>1</td>
<td>odd</td>
<td>xoff</td>
<td>xon</td>
</tr>
<tr>
<td>14400n</td>
<td>14400</td>
<td>8</td>
<td>1</td>
<td>none</td>
<td>xoff</td>
<td>xon</td>
</tr>
<tr>
<td>14400e</td>
<td>14400</td>
<td>7</td>
<td>1</td>
<td>even</td>
<td>xoff</td>
<td>xon</td>
</tr>
<tr>
<td>14400o</td>
<td>14400</td>
<td>7</td>
<td>1</td>
<td>odd</td>
<td>xoff</td>
<td>xon</td>
</tr>
<tr>
<td>28800n</td>
<td>28800</td>
<td>8</td>
<td>1</td>
<td>none</td>
<td>xoff</td>
<td>xon</td>
</tr>
<tr>
<td>28800e</td>
<td>28800</td>
<td>7</td>
<td>1</td>
<td>even</td>
<td>xoff</td>
<td>xon</td>
</tr>
<tr>
<td>28800o</td>
<td>28800</td>
<td>7</td>
<td>1</td>
<td>odd</td>
<td>xoff</td>
<td>xon</td>
</tr>
<tr>
<td>57600n</td>
<td>57600</td>
<td>8</td>
<td>1</td>
<td>none</td>
<td>xoff</td>
<td>xon</td>
</tr>
<tr>
<td>57600e</td>
<td>57600</td>
<td>7</td>
<td>1</td>
<td>even</td>
<td>xoff</td>
<td>xon</td>
</tr>
<tr>
<td>57600o</td>
<td>57600</td>
<td>7</td>
<td>1</td>
<td>odd</td>
<td>xoff</td>
<td>xon</td>
</tr>
</tbody>
</table>

To apply the predefined configuration called 9600n on all the serial ports, enter:
    set port 9600n all
If you then wish to change the port server configuration by applying the 9600e serial port configuration set to port 2, enter:
    set port 9600e 2
The above command changes the port configuration that was set on port 2 using 9600n to the settings defined in 9600e.

If you want the port configuration to take effect immediately, use the \texttt{i} option:

\begin{verbatim}
set port 9600e 2 i
\end{verbatim}

Save the configuration (see the \texttt{save} command, page 232) if you want the configuration to be maintained following reboot.

There is no \texttt{delete} command associated with serial port configurations. After you have applied a configuration to a port or ports and issued a \texttt{save} command, the configured settings remain in effect until the settings are changed, either by applying a different pre-defined configuration or by changing individual settings.

Use the \texttt{show port} command to display current settings.

For example, to view the settings for the configuration named 9600e, enter:

\begin{verbatim}
show port 9600e
\end{verbatim}

If you want to view the configuration of port 2, enter:

\begin{verbatim}
show port 2
\end{verbatim}

\section*{Modifiable Serial Port Configuration}

There are three modifiable port configuration sets, named config1, config2, and config3. You can modify each of these sets to meet the needs of your network environment, then apply each set to a different port or ports to set the port characteristics to the settings defined in the configuration. The \texttt{add port} command (see \texttt{port}, page 139) is used to modify one of the configurations; the \texttt{set port} command is used to apply a configuration to one or more ports.

For example, to modify port configuration number 2, enter:

\begin{verbatim}
add port config2
\end{verbatim}

The current port settings for modifiable port configuration 2 will be displayed.

You will be asked if you want to modify the configuration. Respond either \texttt{y} (yes) or \texttt{n} (no). Responding with \texttt{no} aborts the \texttt{add port} command. Responding with \texttt{yes} allows you to change each of the port characteristics listed. After changing each of the characteristics, the new configuration is displayed and you will be asked if the values are correct. Respond \texttt{yes} here to accept the new values; respond \texttt{no} to abort the \texttt{add port} command.

After modifying configuration 2, you can apply it using the \texttt{set port} command:

\begin{verbatim}
set port config2 1,2,4
\end{verbatim}

The above command applies the settings in configuration 2 to ports 1, 2, and 4.

If you want the port configuration to take effect immediately, use the \texttt{i} option:

\begin{verbatim}
set port config2 1,2,4 i
\end{verbatim}

Save the configuration (see the \texttt{save} command, page 232) if you want the configuration to be maintained following reboot.
There is no **delete** command associated with serial port configurations. After you have applied a configuration to a port or ports and issued a **save** command, the configured settings remain in effect until the settings are changed, either by applying a different pre-defined configuration or by changing individual settings.

Use the **show port** command to display current settings.

For example, to view the current settings for configuration number 2, enter:

```
show port config2
```

If you want to view the configuration of ports 1, 2, and 4, enter:

```
show port 1,2,4
```

---

**WAN Port Characteristics**

*(NOTE: only some port server models have a synchronous WAN port. See the Hardware Manual for your port server to determine whether your port server has a WAN port.)*

The port server provides one synchronous serial port, typically used for WAN connections. The following characteristics can be set for the WAN port, using either a web browser or the command line interface:

- physical interface (default EIA-232)
- type of clock

The **set wan** command is used to change the characteristics of the WAN port (see **WAN** page 173). For example, to change the physical interface setting from the default EIA-232 to E530, enter:

```
set wan interface e530
```

The following services can be configured on the WAN port:

- PPP
- Frame Relay (optional)

The WAN port on the standard port server can only be configured for PPP. To configure Frame Relay on the WAN port, you need to purchase and install additional, optional software. Once you install the optional Frame Relay package, the WAN port can only be configured for Frame Relay, not for PPP.

The following section describes defining a service configuration and enabling it on a port. To enable a service configuration on the WAN port, specify port 33 for the RCS/4000 or port 18 for the RCS/5000 and RCS/6000.
Service Configurations

The services that can be configured are:

- r.telnet (reverse telnet)
- default (login)
- SLIP
- PPP
- Frame Relay (WAN port only)
- a user-defined custom configuration, defining the following session characteristics:
  - password authentication
  - commands/capabilities to which users will have access
  - terminal type
  - whether the login banner is displayed
  - restart delay (in seconds)
  - whether a keyboard hit will be required to initiate a user session
  - up to nine 1-key menu selections
  - up to four separate user sessions (telnet, rlogin, or command shell sessions) that you can switch between

R telnet can be configured directly on a serial port. Services other than r telnet must be defined as a custom configuration and enabled on the port. The port server has allocated 34 custom service configuration, C0 to C33, that can be defined as one of the following:

- PPP
- SLIP
- Frame Relay
- a user configuration—a combination of session parameters that define up to four sessions.

In addition to C0 through C33, there is a service configuration named default. Default (described in a later section) comes factory enabled on all the port server ports.

Only one custom service configuration can be enabled on a port at one time. If you delete the service configuration from a port, there will be no service configuration enabled and only r.telnet or LPD will function on the port. R telnet can be configured on a port simultaneously with the default or a custom configuration. When r telnet and a default/custom configuration are enabled simultaneously, the default/custom configuration waits for a DCD signal and r telnet waits for a connection request from a host. Whichever service receives the expected event first takes control of the port. When it has completed its activity, it releases the port and returns to the wait condition. Note that if you set the port to ignore the DCD signal (using the set ignoredcd command), the default/custom configuration will not wait for the DCD signal to become active and will maintain constant control of the port, not allowing r telnet to function on the port.

**NOTE:** LPD is a service that is also available on the serial ports. No specific configuration is required to use LPD. However, when using LPD, no other services should be configured on the port. Remove any default/custom configuration that is enabled on a port before attempting to send data through the port using LPD.
RTELNET Service

Rtelnet must be enabled on a serial port in order for the port to accept communication that is
initiated by a remote host, such as sending data via a dial-out modem. Rtelnet can also be
used to send data to a printer on a serial port or the parallel port. You can configure a port
using the rtelnet selection on the browser interface or the set rtelnet command via the
command line interface.

You can configure specific ports to accept connections via the network to a TCP port number
using the set rtelnet command (see rtelnet page 153). You can remove rtelnet using the
delete rtelnet command.

For example, to set ports 1 and 2 to accept rtelnet connections, enter:

    set rtelnet 1,2

This configures the port server to accept telnet connections for port 1 on TCP ports 9001 and
9002 and to accept raw TCP mode connections on TCP ports 8001 and 8002.

Default Service Configuration

The default service configuration is enabled on all ports by factory default. The default
service configuration defines the following:

- One user session, configured with access to the user and to_admin capabilities
  (described later in this chapter).
- No 1-key menu selections configured.
- User authentication (password) required.
- The login banner is displayed when the user shell is displayed.
- A terminal type of vt100.

When you enable another service configuration (C0 through C33), the default configuration
is replaced. You can re-enable the default configuration at any time using the set default
command (see default page 111). You can remove the default configuration from a port
using the delete default portnumber command.

To enable the default configuration on port 7, enter:

    set default 7

SLIP Service Configuration

A SLIP configuration on a serial port allows dial-in auto-SLIP or dial-out SLIP capability on
the port. The set slip command (see slip page 156) defines that the custom service
configuration (C0 - C33) specified in the command be configured as SLIP and that the SLIP
configuration be enabled on the port specified. You can remove a SLIP service configuration
from a port using the delete Cn portnumber command. (There is no SLIP option on the web
browser interface.)

SLIP can be enabled on any one of the serial ports, but not on the WAN port. A custom SLIP
service configuration can be enabled on only one port at a time. To run multiple SLIP
connections, you must create a SLIP configuration—with different IP addresses—for each connection. SLIP cannot be combined with other custom service configuration options such as multiple user sessions, 1-key menus, and capabilities access (described later in this chapter).

To define custom service configuration C13 as SLIP and enable it on port 3, enter:

```
set slip C13 3
```

After entering the `set slip` command, you will be prompted for the remote IP address and the local IP address. Your SLIP configuration settings will be displayed and you will be asked to confirm the values. Respond `y` (yes) to confirm the values or `n` (no) to abort the command. A "yes" response enables the SLIP configuration you defined as custom service configuration C13 on port 3.

For a SLIP custom service configuration to take effect, you must save it (see the `save` command, page 232), then reboot your port server.

The `show` command will display both a summary describing the custom configuration(s) specified in the command and a list of the port(s) on which the configuration is enabled. For example:

```
show C13
```

### PPP Service Configuration

A PPP configuration on a serial port allows dial-in auto-PPP or dial-out PPP capability on the port. The web browser interface allows you to configure PPP on a port. The command line interface provides the `set ppp` command for PPP configuration.

The `set ppp` command (see `ppp`, page 144) defines that the custom service configuration (C0 - C33) specified in the command be configured as PPP and that the PPP configuration be enabled on the port specified. You can remove a PPP service configuration from a port using the `delete Cn portnumber` command.

PPP can be enabled on any one of the serial ports or on the WAN port. A custom PPP service configuration can be enabled on only one port at a time. To run multiple PPP connections, you must create a custom configuration—with different IP addresses—for each connection. PPP cannot be combined with other custom service configuration options such as multiple user sessions, 1-key menus, and capabilities access (described later in this chapter).

To define custom service configuration C6 as PPP and enable it on port 3, enter:

```
set ppp C6 3
```

After entering the `set ppp` command, you will be prompted for the PPP interface settings. Your PPP configuration settings will be displayed and you will be asked to confirm the values. Respond `y` (yes) to confirm the values or `n` (no) to abort the command. A "yes" response enables the PPP configuration you defined as custom service configuration C6 on port 3.

If you want the PPP custom service configuration to be maintained following reboot, you must save it (see the `save` command, page 232).
The `show` command will display both a summary describing the custom configuration(s) specified in the command and a list of the port(s) on which the configuration is enabled. For example:

```
show C6
```

### Frame Relay Service Configuration

Frame relay service is not included in the standard port server. Frame relay service is an optional extra software feature. Frame Relay service can only be configured on the WAN port. Not all models of the port server include a WAN port. See the Hardware Manual for your port server to determine whether it has a WAN port.

A Frame Relay configuration on the WAN port allows frame relay connections on the port. The web browser interface provides a frame relay option. The command line interface provides the `set frame_relay` command.

The `set frame_relay` command (see `frame_relay`, page 116) defines that the custom service configuration (C0 - C33) specified in the command be configured as frame relay and that the frame relay configuration be enabled on the port specified. Frame Relay can only be enabled on the WAN port. You can remove a frame relay service configuration using the `delete Cn portnumber` command.

Configuring frame relay is a two-step process:

1. **Add a DLCI number**: Before you can define a custom service configuration as Frame Relay, you must have at least one Data Link Connection Identifier (DLCI) number. A DLCI number identifies a connection, also called a permanent virtual circuit (PVC), available for sending and receiving data during your frame relay session. The DLCI number is provided by your frame relay service provider. The `add dlci` command (See `dlci`, page 114) provides the port server with the information needed to correctly route information for a given DLCI number. To add a PVC for DLCI number 16, enter:

   ```
   add dlci 16
   ```

   You will be prompted for the information required, e.g. remote and local IP addresses.

2. **Define and enable the custom service configuration**: After adding the DLCI number, you can define a custom service configuration (see `frame_relay`, page 116). To define custom service configuration C6 as Frame Relay and enable it on the WAN port (port 18 on the RCS/5000 and RCS/6000 or port 33 on the RCS/4000), enter:

   ```
   set frame_relay C6 33
   ```

   After entering the `set frame_relay` command, you will be prompted to enter the type of Link Management Interface (LMI) you have leased: ANNEX_D, LMI, or ANNEX_A. Additional prompts will appear. In most cases, the default will be the preferred choice. Only change the default to meet a specific need. After responding to the prompts, you will be asked if you are sure you want to enable the configuration on the WAN port.

   If you want the frame relay custom service configuration to be maintained following reboot, you must save it (see the `save` command, page 232).
The **show** command will display both a summary describing the custom configuration(s) specified in the command and a list of the port(s) on which the configuration is enabled. For example:

    show C6

You can remove a PVC using the **delete dlcid dlcinumber** command.

**User-defined Custom Service Configuration**

Service configurations offer you the ability to define a custom configuration, specifying the characteristics of user sessions, and enable the defined custom configuration on one or more serial ports. Custom service configurations can be used to enable various user applications such as telnet or rlogin. The web browser interface allows you to define custom configurations. The command line interface provides the **set custom** command.

The **set custom** command (see [custom](#) page 102) both defines the custom service configuration (C0 - C33) specified in the command and enables it on the port specified. You can remove a custom service configuration from a port using the **delete Cn portnumber** command. A user custom configuration can be enabled on one or more serial ports.

To define user session service configuration C23 on port 3 and port 4, enter:

    set custom C23 3,4

You will be prompted for values for the configuration, including:

- Password authentication
- Commands/ capabilities to which users will have access
- The type of terminal being used
- Whether or not the login banner will be displayed on user sessions
- A restart delay (in seconds)
- Whether or not a keyboard hit will be required to initiate a user session
- Up to nine 1-key menu selections
- Up to four separate user sessions

Detailed descriptions of the possible configuration values are provided in the command reference chapter (see [CUSTOM](#) page 102).

The **show** command will display both a summary describing the custom configuration specified in the command and a list of the port(s) on which the configuration is enabled. For example:

    show C6

You can delete a user service configuration using the delete command. For example, to delete configuration C6 from port 4, enter:

    delete C6 4
Parallel Port

The parallel port is specified as port 32 for the RCS/4000 and port 17 for the RCS/5000 and RCS/6000. The parallel port is compatible with common PC parallel printer ports. The parallel port has no default configuration set on it, allowing LPD printing without modification. You can also enable rtelnet on the parallel port to allow access to the port from a host system (see RTELNET, page 153) using the rtn utility (see The rtn Utility, page 263). The web browser interface provides an option to configure rtelnet on the parallel port.
Chapter 5: Configuration Commands

Configuration Overview

When the port server is booted, the configuration information that defines the operation of the port server during the session (the active configuration) is loaded into memory from the current configuration database stored in flash memory. During the session, you can change any configuration setting in the active configuration. However, changes to the active configuration are not automatically written to the current configuration database. If you wish the changes to be stored in the current configuration database so that the changed settings will be loaded during the next restart/reboot, you must issue a `save` command.

When the port server is delivered, the current configuration database contains factory default settings. In addition, a factory default configuration database (a copy of the current configuration database) is stored, along with the current configuration database, in flash memory. The factory default configuration database is a backup configuration which can be restored into the current configuration database if necessary. If the port server encounters an error condition while loading information from the current configuration database during restart/reboot, it will load information from the factory default configuration database instead, in order to complete the boot. The LED on the front of the port server will flash alternately yellow and green to indicate that a problem occurred and the factory default configuration was loaded.

The configuration settings can be changed using one of several methods: 1) using the port server command interface via a telnet connection or a terminal connected to the port server; 2) using a web browser to connect to the IP address of the port server; or 3) the NativeCOM Windows host software package will change the configuration implicitly for you.

1. Command Interface

   The port server command interface can be accessed by connecting a terminal to the port server or by connecting to the port server via telnet. When you log onto the port server to change the configuration settings, the changes are made in the active configuration. You must save (see the `save` command, page 232) the active configuration into the current configuration database for the changes to be available after reboot/restart.

   The Description and Examples section below describes the use of the commands available through the port server command interface.
2. Web Browser

You can make configuration changes to the port server using any web browser that supports tables. Point the browser at the IP address of the port server. When you use the web browser to change the configuration settings, the changes are made in the active configuration. You must save (see the `save` command, page 232) the active configuration into the current configuration database for the changes to be available after reboot/restart.

3. NATIVEcom

Systech NATIVEcom is Windows 95/98/NT software that dynamically configures ports on the port server and makes the remote serial communication ports on the port server available to PC programs as local COM ports. If you use NATIVEcom, you generally do not need to manually configure the port server. Exceptions might be to change the electrical interface on a port (RS232, RS422, RS485) or to configure a port that is not under NativeCOM control.

At any point, you can restore the factory default configuration settings. Use the `restore` command in the Firmware Monitor to copy the factory default configuration database into the current configuration database.

Command Overview

The port server configuration command set consists of four commands. The four commands are combined with keywords and specific command parameters to configure the server for your needs. You can save the new settings so that they will be available after you reboot/restart the system.

Configuration Commands

The configuration commands are:

- **add**: Adds or defines a configuration option.
- **delete**: Deletes or removes a configuration option.
- **set**: Sets or enables a configuration option.
- **show**: Displays a configuration option setting.

**NOTE**: In a few cases, the `add` and `set` commands are used to perform similar functions; refer to the detailed instructions for each configuration option to learn when to apply these exceptions.

When you set configuration options, the settings are changed in the active configuration. Some of the settings become effective immediately, while others only become effective upon restart or user login. The command descriptions section indicates when each command becomes effective (immediately, following restart, upon user login). However, none of the settings will be added to the current configuration database, to be available after reboot/restart, unless you save the new settings into the current configuration database.
Saving the Configuration Settings

Use the `save` command after completing your configuration changes to save the new settings to the current configuration database. Saving the active configuration options to the current configuration database means that the settings are maintained following system reboot/restart. To save the changes made to the active configuration, type:

```
save
```

The current configuration database can, of course, be changed at any time if you set new configuration options and save them.

Command Usage

The four configuration commands are used with keywords and parameters to configure the server. Table 5-1 provides a summary of the keywords and options that are available for each command. The “Description and Examples” section provides details of the usage of the commands with each keyword and its available options.

Command Syntax

The syntax for the configuration commands is:

```
command keyword parameter
```

where:

- **command** is one of the four configuration commands.
- **keyword** is one of the keywords that can be combined with the command.
- **parameter** is one or more parameters that are required by the keyword. Not all keywords require parameters.
The command syntax is described using the following conventions:

- Commands and keywords that must be entered exactly as they appear in the command syntax line are shown in **bold**.
- Keyword abbreviations are underscored; the command will be executed using either the full keyword or the abbreviation.
- Required entries are enclosed in braces (**{}**)
- Optional entries are enclosed in brackets (**[]**)
- Multiple possible entries—either required or optional—are separated by bars (**|**); select only one of the entries listed.
- Parameters that are meant to be replaced by a term specific to your system (e.g., the host name parameter indicating that you are to type the name of a specific host on your system) are shown in italics. If necessary, these selection variables are defined below the command syntax line.

For example, the description of the command that allows you to set the character size for a serial port or ports is:

```
set size { port number(s) } { 5 | 6 | 7 | 8 } [ i ]
```

- **port numbers**: Any valid port number(s) for your port server model. May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range.

In this example, you would type `set size` or `set si` as shown in bold. One or more port numbers and one (only 1) of the character size parameters listed (either 5 or 6 or 7 or 8) are mandatory, indicated by the braces. The "i" parameter is optional, indicated by the brackets. Thus, one command might be:

```
set size 3-5 8 i
```
### Keyword Summary

**Table 5-1. Configuration Command Summary**

<table>
<thead>
<tr>
<th>Command</th>
<th>Keyword</th>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>add</td>
<td>community</td>
<td></td>
<td>Adds an SNMP community profile to the SNMP community list.</td>
</tr>
<tr>
<td>add</td>
<td>custom</td>
<td></td>
<td>Defines a custom service configuration (numbered C0 through C33).</td>
</tr>
<tr>
<td>add</td>
<td>dlci</td>
<td></td>
<td>Provides the port server with the information needed to correctly route information for a given DLCI number.</td>
</tr>
<tr>
<td>add</td>
<td>host</td>
<td></td>
<td>Adds a host name to the bottom of the host name list.</td>
</tr>
<tr>
<td>add</td>
<td>modem</td>
<td></td>
<td>Defines a modem and its associated chat script.</td>
</tr>
<tr>
<td>add</td>
<td>nameserver</td>
<td></td>
<td>Adds a DNS server to the bottom of the name server list.</td>
</tr>
<tr>
<td>add</td>
<td>nat</td>
<td></td>
<td>Adds NAT functionality to an interface.</td>
</tr>
<tr>
<td>add</td>
<td>port</td>
<td>config</td>
<td>Adds a modifiable serial port configuration set. There are three modifiable configurations available, numbered config1 through config3.</td>
</tr>
<tr>
<td>add</td>
<td>pppadlist</td>
<td></td>
<td>Adds an IP address pair to the PPP address pool list specified.</td>
</tr>
<tr>
<td>add</td>
<td>pppauthlist</td>
<td></td>
<td>Adds a name/peer-ID and corresponding secret/password to the PPP authorization list database.</td>
</tr>
<tr>
<td>add</td>
<td>radius</td>
<td></td>
<td>Adds a new RADIUS server to the list of valid RADIUS servers.</td>
</tr>
<tr>
<td>add</td>
<td>route</td>
<td></td>
<td>Adds a route based on the destination host name or IP address, gateway host name or IP address, route flags, and jumps specified.</td>
</tr>
<tr>
<td>add</td>
<td>system</td>
<td></td>
<td>Defines a remote system and its associated port(s), telephone number, and login script.</td>
</tr>
<tr>
<td>add</td>
<td>trap</td>
<td></td>
<td>Adds a Network Management Station (NMS) to the SNMP trap list.</td>
</tr>
<tr>
<td>Command</td>
<td>Keyword</td>
<td>Options</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>---------------</td>
<td>---------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>add</td>
<td>VIEW</td>
<td></td>
<td>Adds MIB object IDs to an existing SNMP community’s view definition.</td>
</tr>
<tr>
<td>delete</td>
<td>Cn</td>
<td></td>
<td>Deletes the custom service configuration (numbered C1 through C33) from the port(s) specified.</td>
</tr>
<tr>
<td>delete</td>
<td>community</td>
<td></td>
<td>Deletes an existing community profile from the SNMP community list.</td>
</tr>
<tr>
<td>delete</td>
<td>default</td>
<td></td>
<td>Deletes the default service configuration from the port(s) specified.</td>
</tr>
<tr>
<td>delete</td>
<td>dial_on_deman</td>
<td></td>
<td>Deletes a defined dial_on_demand interface.</td>
</tr>
<tr>
<td>delete</td>
<td>dlci</td>
<td></td>
<td>Deletes a DLCI number.</td>
</tr>
<tr>
<td>delete</td>
<td>host</td>
<td></td>
<td>Deletes the specified host name from the host name list.</td>
</tr>
<tr>
<td>delete</td>
<td>modem</td>
<td></td>
<td>Deletes the configured modem.</td>
</tr>
<tr>
<td>delete</td>
<td>mpool</td>
<td></td>
<td>Deletes the specified ports from a specified modem pool. If all ports are deleted, the pool itself is deleted.</td>
</tr>
<tr>
<td>delete</td>
<td>nameserver</td>
<td></td>
<td>Deletes a DNS server from the name server list.</td>
</tr>
<tr>
<td>delete</td>
<td>nat</td>
<td></td>
<td>Removes NAT functionality from an interface.</td>
</tr>
<tr>
<td>delete</td>
<td>bpool</td>
<td></td>
<td>Deletes the specified ports from a specified printer pool. If all ports are deleted, the pool itself is deleted.</td>
</tr>
<tr>
<td>delete</td>
<td>bppadlist</td>
<td></td>
<td>Deletes an IP address pair from the PPP address pool list specified.</td>
</tr>
<tr>
<td>delete</td>
<td>bppauthlist</td>
<td></td>
<td>Deletes the specified name/peer from the PPP authorization list database.</td>
</tr>
<tr>
<td>delete</td>
<td>radius</td>
<td></td>
<td>Removes a RADIUS server from a server list.</td>
</tr>
<tr>
<td>Command</td>
<td>Keyword</td>
<td>Options</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>delete</td>
<td>route</td>
<td></td>
<td>Deletes a route or route destination. Include only the destination host name or IP address to delete a specific destination. Include a destination host name or IP address to delete a route. To delete all destinations associated with a gateway, include only the gateway host name or IP address in the command.</td>
</tr>
<tr>
<td>delete</td>
<td>rtelnet</td>
<td></td>
<td>Deletes rtelnet sessions running on the specified port or ports.</td>
</tr>
<tr>
<td>delete</td>
<td>system</td>
<td></td>
<td>Deletes the specified system definition.</td>
</tr>
<tr>
<td>delete</td>
<td>tcp service</td>
<td></td>
<td>Removes a TCP listener.</td>
</tr>
<tr>
<td>delete</td>
<td>trap</td>
<td></td>
<td>Removes an NMS from the SNMP trap list.</td>
</tr>
<tr>
<td>delete</td>
<td>view</td>
<td></td>
<td>Removes existing view definitions from an SNMP community profile.</td>
</tr>
<tr>
<td>set</td>
<td>admin</td>
<td></td>
<td>Sets the administrative password to the string specified.</td>
</tr>
<tr>
<td>set</td>
<td>advertise</td>
<td>on</td>
<td>Enables/ disables the RIP advertise feature.</td>
</tr>
<tr>
<td>set</td>
<td>banner</td>
<td>on</td>
<td>Customizes the login banner to the string specified.</td>
</tr>
<tr>
<td>set</td>
<td>Cn</td>
<td></td>
<td>Enables a custom service configuration (numbered C0 through C33) on the port(s) specified.</td>
</tr>
<tr>
<td>set</td>
<td>contact</td>
<td></td>
<td>Names the person responsible for the system (SNMP configuration).</td>
</tr>
<tr>
<td>set</td>
<td>custom Cn</td>
<td></td>
<td>In one step, defines a custom service configuration (numbered C0 through C33) and enables it on the port(s) specified.</td>
</tr>
<tr>
<td>set</td>
<td>default</td>
<td></td>
<td>Enables the default service configuration on the port(s) specified.</td>
</tr>
<tr>
<td>set</td>
<td>description</td>
<td></td>
<td>Describes your SNMP configuration with a string of up to 256 characters.</td>
</tr>
<tr>
<td>Command</td>
<td>Keyword</td>
<td>Options</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------</td>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>set</td>
<td>dial on deman</td>
<td>dod</td>
<td>Defines the parameters for making a PPP connection to a particular system. The number is an arbitrary identifier for a defined DOD interface.</td>
</tr>
<tr>
<td>set</td>
<td>domain</td>
<td></td>
<td>Defines the name of the domain host.</td>
</tr>
<tr>
<td>set</td>
<td>frame relay</td>
<td></td>
<td>Defines the custom service configuration specified (Cn) as frame relay and enables it on the specified port.</td>
</tr>
<tr>
<td>set</td>
<td>ignoredcd</td>
<td>i</td>
<td>Configures the port server to ignore the DCD modem signal</td>
</tr>
<tr>
<td>set</td>
<td>input</td>
<td>xoff</td>
<td>Sets method of software flow control to be used for controlling the connected device's data input on the port(s) specified.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rtscts</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>dtrdsr</td>
<td></td>
</tr>
<tr>
<td>set</td>
<td>ip</td>
<td>rs232</td>
<td>Sets the electrical interface to be used on the specified port(s).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>rs422</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>rs485</td>
<td></td>
</tr>
<tr>
<td>set</td>
<td>location</td>
<td></td>
<td>Defines the physical location of the system (SNMP configuration).</td>
</tr>
<tr>
<td>set</td>
<td>mask</td>
<td></td>
<td>Sets subnet mask value you specify (as the total number of bits in the network address).</td>
</tr>
<tr>
<td>set</td>
<td>modem</td>
<td></td>
<td>Assigns a modem definition to one or more port server ports.</td>
</tr>
<tr>
<td>set</td>
<td>mpool</td>
<td></td>
<td>Configures the specified ports into a specified modem pool.</td>
</tr>
<tr>
<td>set</td>
<td>output</td>
<td>none</td>
<td>Sets method of software flow control to be used for controlling the port server's data output on the port(s) specified.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>xon</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>xany</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>rtscts</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>dtrdsr</td>
<td></td>
</tr>
<tr>
<td>set</td>
<td>owner</td>
<td>string</td>
<td>Specifies a host which has configured this port for reverse telnet operation.</td>
</tr>
<tr>
<td>Command</td>
<td>Keyword</td>
<td>Options</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>set</td>
<td><strong>parity</strong></td>
<td>even odd none i</td>
<td>Sets parity on the port(s) specified. Sets parameter immediately.</td>
</tr>
<tr>
<td>set</td>
<td><strong>pin1</strong></td>
<td>dcd dsr</td>
<td>Configures pin1 as the DCD or DSR signal. Default: DCD</td>
</tr>
<tr>
<td>set</td>
<td><strong>port</strong></td>
<td>i</td>
<td>Enables the serial port configuration you specify (either the default, a modifiable configuration, or a predefined configuration) on the specified port(s). Sets parameter immediately.</td>
</tr>
<tr>
<td>set</td>
<td><strong>portdesc</strong></td>
<td>string</td>
<td>Sets a user specified description of the port.</td>
</tr>
<tr>
<td>set</td>
<td><strong>power</strong></td>
<td>5 12 none i</td>
<td>Sets power on the specified port. Sets parameter immediately.</td>
</tr>
<tr>
<td>set</td>
<td><strong>ppool</strong></td>
<td>Configures the specified ports into a specified printer pools.</td>
<td></td>
</tr>
<tr>
<td>set</td>
<td><strong>PPP</strong></td>
<td></td>
<td>In one step, defines a PPP custom service configuration (numbered C0 through C33) and enables it on the specified port(s). Both the custom service configuration number and the port(s) must be specified in the command.</td>
</tr>
<tr>
<td>set</td>
<td><strong>rip</strong></td>
<td>on off</td>
<td>Enables/ disables the RIP option.</td>
</tr>
<tr>
<td>set</td>
<td><strong>rtelnet</strong></td>
<td></td>
<td>Enables an rtelnet connection on the serial port specified in the command.</td>
</tr>
<tr>
<td>set</td>
<td><strong>size</strong></td>
<td>5 6 7 8 i</td>
<td>Sets character size on the port(s) specified. Sets parameter immediately.</td>
</tr>
<tr>
<td>Command</td>
<td>Keyword</td>
<td>Options</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>----------</td>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>set</td>
<td>slip</td>
<td></td>
<td>In one step, defines a SLIP custom service configuration (numbered C0 through C33) and enables it on the specified port(s). Note that both the custom service configuration number and the port(s) must be specified in the command.</td>
</tr>
<tr>
<td>set</td>
<td>snmp</td>
<td>on/off</td>
<td>Enables/ disables the SNMP agent.</td>
</tr>
<tr>
<td>set</td>
<td>speed</td>
<td>i</td>
<td>Sets speed (bit rate) on the port(s) specified.</td>
</tr>
<tr>
<td>set</td>
<td>stop</td>
<td>1</td>
<td>Sets stop bits on the port(s) specified. Sets parameter immediately.</td>
</tr>
<tr>
<td>set</td>
<td>switch</td>
<td>i</td>
<td>Defines a character that can be used to switch between sessions.</td>
</tr>
<tr>
<td>set</td>
<td>tcp_service</td>
<td></td>
<td>Registers a TCP listener for the specified port.</td>
</tr>
<tr>
<td>set</td>
<td>timeout</td>
<td>i</td>
<td>Sets the inactivity time-out on a port or ports. If no activity takes place during the specified time-out period, the specified port(s) will close the connection.</td>
</tr>
<tr>
<td>set</td>
<td>user</td>
<td></td>
<td>Sets the user password to the string specified.</td>
</tr>
<tr>
<td>set</td>
<td>wan</td>
<td>interface clock</td>
<td>The interface option sets the physical WAN interface to be RS-232, RS-422, EIA-530, or V.35, as specified. The clock option sets the clock to be external or internal as specified. Note if clock is set as internal, you must supply baud rate.</td>
</tr>
<tr>
<td>show</td>
<td>banner</td>
<td></td>
<td>Displays the login banner string and lists the service configurations on which it is enabled.</td>
</tr>
<tr>
<td>show</td>
<td>Cn</td>
<td></td>
<td>Displays information about the specified customized service configuration (C0 through C33), including user session list, 1-key menu list, capabilities access levels, etc. Using the C* keyword shows information about all the custom service configurations.</td>
</tr>
<tr>
<td>show</td>
<td>community</td>
<td></td>
<td>Displays a list of the communities defined to the port server</td>
</tr>
<tr>
<td>Command</td>
<td>Keyword</td>
<td>Options</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>---------------</td>
<td>-----------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>show</td>
<td>contact</td>
<td></td>
<td>Displays the specified SNMP contact name.</td>
</tr>
<tr>
<td>show</td>
<td>default</td>
<td></td>
<td>Displays a summary of the default service configuration settings and shows on which ports it is enabled.</td>
</tr>
<tr>
<td>show</td>
<td>description</td>
<td></td>
<td>Displays the specified SNMP description.</td>
</tr>
<tr>
<td>show</td>
<td>dial on demand</td>
<td>dod</td>
<td>Displays all dial_on_demand interfaces.</td>
</tr>
<tr>
<td>show</td>
<td>dleci</td>
<td></td>
<td>Displays a DLCI number.</td>
</tr>
<tr>
<td>show</td>
<td>domain</td>
<td></td>
<td>Displays the domain host name.</td>
</tr>
<tr>
<td>show</td>
<td>frame relay</td>
<td></td>
<td>Displays frame relay configurations.</td>
</tr>
<tr>
<td>show</td>
<td>ignoredcd</td>
<td></td>
<td>Displays the DCD setting for a serial port.</td>
</tr>
<tr>
<td>show</td>
<td>host</td>
<td></td>
<td>Displays the host name list.</td>
</tr>
<tr>
<td>show</td>
<td>input</td>
<td></td>
<td>Displays the input software flow control settings for the port server's serial ports.</td>
</tr>
<tr>
<td>show</td>
<td>ip</td>
<td></td>
<td>Displays the port server's IP address and indicates how it was assigned (using bootp or the set ip command).</td>
</tr>
<tr>
<td>show</td>
<td>location</td>
<td></td>
<td>Displays the physical location of the system (SNMP configuration).</td>
</tr>
<tr>
<td>show</td>
<td>mask</td>
<td></td>
<td>Displays the subnet mask.</td>
</tr>
<tr>
<td>show</td>
<td>mac</td>
<td></td>
<td>Displays the port server Ethernet MAC address.</td>
</tr>
<tr>
<td>show</td>
<td>model</td>
<td></td>
<td>Shows model number of port server.</td>
</tr>
<tr>
<td>show</td>
<td>modem</td>
<td>no name</td>
<td>Displays all defined modems. Displays a specified modem and chat script.</td>
</tr>
<tr>
<td>show</td>
<td>nameserver</td>
<td></td>
<td>Displays the DNS server list.</td>
</tr>
<tr>
<td>show</td>
<td>nat</td>
<td></td>
<td>Displays a list of defined NAT interfaces and statistics.</td>
</tr>
<tr>
<td>show</td>
<td>output</td>
<td></td>
<td>Displays the output software flow control settings for the port server's serial ports.</td>
</tr>
<tr>
<td>show</td>
<td>parity</td>
<td></td>
<td>Displays the parity specified for the port server's serial ports.</td>
</tr>
<tr>
<td>Command</td>
<td>Keyword</td>
<td>Options</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>show</td>
<td>pppadlist</td>
<td></td>
<td>Displays the IP address pairs in the PPP address pool database.</td>
</tr>
<tr>
<td>show</td>
<td>pppauthlist</td>
<td></td>
<td>Displays all PPP password authentication database entries.</td>
</tr>
<tr>
<td>show</td>
<td>port</td>
<td></td>
<td>Displays a serial port configuration (either modifiable or predefined) if the configuration “name” is specified; displays the serial configuration enabled on a port if the port number is specified; or displays the configurations of all serial ports if “all” is specified.</td>
</tr>
<tr>
<td>show</td>
<td>ppool</td>
<td></td>
<td>Displays what ports are configured in a specified printer pool.</td>
</tr>
<tr>
<td>show</td>
<td>radius</td>
<td></td>
<td>Displays the current ordered lists of RADIUS servers.</td>
</tr>
<tr>
<td>show</td>
<td>rip</td>
<td></td>
<td>Displays status of the RIP option</td>
</tr>
<tr>
<td>show</td>
<td>route</td>
<td></td>
<td>Displays a list of manually created routes.</td>
</tr>
<tr>
<td>show</td>
<td>rtelnet</td>
<td></td>
<td>Displays a list of those ports that were configured for an rtelnet connection using the set rtelnet command.</td>
</tr>
<tr>
<td>show</td>
<td>size</td>
<td></td>
<td>Displays character sizes specified for port server’s serial ports.</td>
</tr>
<tr>
<td>show</td>
<td>snmp</td>
<td></td>
<td>Displays the port server’s SNMP configuration.</td>
</tr>
<tr>
<td>show</td>
<td>speed</td>
<td></td>
<td>Displays the baud rates specified for the port server’s serial ports.</td>
</tr>
<tr>
<td>show</td>
<td>stop</td>
<td></td>
<td>Displays the stop bits specified for the port server’s serial ports.</td>
</tr>
<tr>
<td>show</td>
<td>switch</td>
<td></td>
<td>Displays the switch character settings for serial ports</td>
</tr>
<tr>
<td>show</td>
<td>system</td>
<td></td>
<td>Displays all details of specified system.</td>
</tr>
<tr>
<td>show</td>
<td>tcp_service</td>
<td></td>
<td>Displays all the current TCP listeners.</td>
</tr>
<tr>
<td>show</td>
<td>timeout</td>
<td></td>
<td>Displays active timeout settings.</td>
</tr>
<tr>
<td>show</td>
<td>trap</td>
<td></td>
<td>Displays the current contents of the SNMP trap list.</td>
</tr>
<tr>
<td>Command</td>
<td>Keyword</td>
<td>Options</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>show</td>
<td>version</td>
<td></td>
<td>Displays current version of onboard software and firmware.</td>
</tr>
<tr>
<td>show</td>
<td>wan</td>
<td></td>
<td>Displays the interface and clock settings specified for the WAN port.</td>
</tr>
</tbody>
</table>

**Description and Examples**

The following section contains a detailed description of each configuration command, listed in alphabetical order by keyword. Each description provides the syntax and usage for the keyword, including examples.

After typing each command, enter a carriage return to initiate the command.
BANNER

set banner { character string }
show banner

The port server provides a default login banner which is displayed as part of the user login screen. You can modify this banner to suit your needs. Using the set banner command, you can enter up to 80 alphanumeric characters to customize the banner. Once set, the new login banner takes effect at next user login; use the save command to save the banner string in the current configuration database so that it is retained following reboot.

The command for setting the login banner (setting it modifies the default banner to your specifications) is:

    set banner Hello

The port server's default configuration displays the login banner on the one user session enabled on each of the serial ports. By creating a custom service configuration, you can specify that the login banner is not displayed on user sessions. Refer to the section titled Customizing a Service Configuration in this chapter for detailed information about deleting or adding the login banner.

To display the current login banner and to see which service configurations are displaying it as part of their user sessions, enter:

    show banner

The newly defined banner will be displayed. Use the save command to retain the banner string following reboot.

**NOTE:** There is only one login banner. You cannot define different login banners for different service configurations. The same login banner character string (either a customized banner of your choosing or the default banner), will be displayed on all user sessions on which it is enabled.
COMMUNITY

add community
add community {name}
show community
show community [name(s)]
delete community

Use the add community command to add an SNMP community profile to the SNMP community list. This list defines all the communities for which the port server will service SNMP requests. Once a community profile has been added, you can configure its Management Information Base (MIB) access permissions by using the add view command (see the add view command, page 170). If you do not specify a community name, you will be prompted for one.

Use the show community command to display a list of the communities defined to the port server. Specifying a community name or names will provide more detailed information about the associated views and permissions.

Use the delete community command to delete an existing community profile from the SNMP community list. If you do not specify a community name, you will be prompted for one.

By default, the port server creates a community named public which allows read-only access to the entire Internet MIB.

The example below illustrates the commands used to add a new community name to the port server communities list, to show a summary of all communities, to show a detailed report for one community, and to delete a community name.

Example

```
[admin:1/2]>>add community new_community
SNMP community 'new_community' has been added.
[admin:1/3]>>show community
SNMP Communities
----------------
public
new_community
[admin:1/4]>>show community public
Community Name: public
      View Subtree     Access
     ---------------     ------
        mib            readonly
[admin:1/5]>>delete community new_community
SNMP community 'new_community' has been deleted.
```
**CUSTOM**

add custom \{ Cn \}
add custom task
set \{ Cn \} \{ port number(s) \}
set custom \{ Cn \} \{ port number(s) \}
delete \{ Cn \} \{ port number(s) \}
delete task \{ n \}
show \{ Cn \| C* \}

Cn: Custom service configuration C0 through C33 that you define and enable on a specific port or ports. Note that C* in the show command indicates "all."

port numbers: Any valid port number for your port server model. May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range.

The add custom Cn command is used to define customer service configurations. The add custom task command is used to define commands that will be executed when the port server is booted.

The port server has allocated 34 custom service configurations. Initially, each custom service is unconfigured. You use the add custom command to define and modify the custom service configurations to fit your particular needs.

If you are defining PPP, SLIP, or Frame Relay services, use the set ppp command, the set slip command, or the set frame_relay command (see PPP page 144 or SLIP page 156 or FRAME_RELAY page 116). If you are defining a user session service configuration, use the add custom or set custom commands described in this section.

You can define a user service configuration in two separate steps or in a single step. In the two-step process, you first define the configuration using the add custom command and then enable the configuration on one or more ports using the set Cn command. Or, you can both define and enable the configuration with a single command using the set custom command.

To define the user session service configuration C8, without enabling it on a port, enter:

```
add custom C8
```

In the second step, to enable the configuration C8 on ports 2 and 4, enter:

```
set C8 2,4
```

To define the user session service configuration C8 and enable it on ports 2 and 4, enter:

```
set custom C8 2,4
```

When you define the user service configuration, you will be prompted for several configuration values, described below. As soon as you have finished defining the configuration, it can be enabled on a port or on multiple ports. However, for any
configuration to be retained following a reboot of the port server, you must save it. (See the \texttt{save} command, page \pageref{save}).

You can define up to ten custom tasks that will be run when the port server is booted. Each task is a command that the port server can execute. The tasks will run only once. To define a custom task, enter:

\texttt{add custom task}

You will be prompted for the task number, which can be any number up to 10. You will then be prompted to enter the single-line command.

The \texttt{show} command identifies the custom service configuration specified in the command. To determine the parameters of custom service configuration C2, as well as the port(s) on which it is enabled, enter:

\texttt{show C2}

To determine the parameters of all custom service configurations and the ports on which they are enabled and to see all defined custom tasks, enter:

\texttt{show C*}

With the \texttt{delete} command, you can delete a custom service configuration from the port or ports on which it is enabled. For example, to delete custom configuration C33 from ports 2 through 4, enter:

\texttt{delete C33 2-4}

To delete a custom task, type:

\texttt{delete task \textit{n}}

where \textit{n} is the number of the task to be deleted.

\section*{Configuration Values}

\textbf{authentication level}

You have the option of selecting \texttt{none}, \texttt{user}, \texttt{admin}, or \texttt{radius}. This selection determines the type of password authentication that will be required to access the user sessions you will be configuring:

\begin{tabular}{|l|p{13cm}|}
\hline
\textbf{none}: & No password is required to access the functions provided by the service configuration. \\
\hline
\textbf{user}: & A user password is required to access the functions provided at the user access level. \\
\hline
\textbf{admin}: & An administrative password is required to access the functions provided at the administrative access level. \\
\hline
\textbf{radius}: & A user name and password subject to RADIUS authentication is required to access this service. \\
\hline
\end{tabular}
NOTE: If RADIUS authentication is selected, the RADIUS server may request a different service configuration than the one defined in the respective custom configuration. For more information, see the RADIUS configuration command, page 148.

**capabilities level**

The **capabilities level** selection allows you to define the set of port server commands or capabilities to which users will have access. Specific information about the commands included within each of the levels is provided below (see Defining Capabilities Access Levels, page 107).

**NOTE:** To some degree, the password authentication selection has an effect on the capabilities that may be accessed by users. For example, if you assign a capabilities level of **user**, but require **admin** password authentication, anyone who knows the administrative password can access the administrative commands/capabilities.

**terminal type**

Enter the type that is appropriate for your users. You may enter any terminal type; however, an application may not recognize each type. If an application does not recognize a specified terminal type, it may display an error message. The terminal type, whether supported by the port server or not, is passed to the host when you telnet to a host from a shell.

The port server allows you to configure multiple sessions. When switching between sessions, a supported terminal type must be specified in order for the display to function correctly. When you switch from one session to another and then return to the original session, the screen will either re-display the original screen or clear the screen. Whether the original session screen is re-displayed or the screen is cleared depends on the terminal type. Only terminal types that support multiple page display can re-display the original screen. The terminal type must support a number of multiple pages equal to the number of sessions required. That is, if a user is switching between 4 sessions (the maximum number of sessions the port server can support), the terminal must provide multiple page display support for at least 4 pages. Of the currently supported terminal types, only the Wyse 50+ terminal supports sufficient multiple display pages to support 4 sessions and, therefore, allows the original screen to be redisplayed after a switch when 4 sessions are configured. The Wyse 60 terminal supports up to 2 multiple display pages and so will redisplay the original session if the user has only 2 session, but will clear the screen if the user has 3 or 4 sessions. All other supported terminal types will clear the screen when the user returns to the original session.

The terminal types currently supported and the type name to enter are:

<table>
<thead>
<tr>
<th>Terminal type</th>
<th>Multiple page display?</th>
<th>Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANSI terminal</td>
<td>no</td>
<td>ansi</td>
</tr>
<tr>
<td>Dumb terminal</td>
<td>no</td>
<td>dumb or un</td>
</tr>
<tr>
<td>IBM 3151</td>
<td>no</td>
<td>ibm3151 or i3151</td>
</tr>
</tbody>
</table>
### Configuration Commands

#### IBM 3161/3163 terminal
- No
- `ibm3161` or `ibm3163` or `wy60-316X` or `wyse60-316X`

#### IBM PC PC/IX
- No
- `ibmpc`

#### Sun Workstation console
- No
- `sun`

#### Sun Workstation (w/o INS character)
- No
- `sun-e` or `sun-nic` or `sune`

#### Sun Workstation (with status line)
- No
- `sun-s`

#### Sun Workstation (with status hacked for EMACS)
- No
- `sun-e-s` or `sun-s-e`

#### Sun 48-line window
- No
- `sun-48`

#### Sun 34-line window
- No
- `sun-34`

#### Sun 24-line window
- No
- `sun-24`

#### Sun 17-line window
- No
- `sun-17`

#### Intelligent switch
- No
- `switch`

#### DEC vt100
- No
- `vt100` or `vt100-am` or `vt100am`

#### Wyse 50
- No
- `w50` or `wy50` or `wyse50` or `wyse-50`

#### Wyse 50+
- 6 pages
- `w50+` or `wy50+` or `wyse50+` or `wyse-50+`

#### Wyse 60
- 2 pages
- `wy60` or `wyse60`

#### X-Windows Emulation
- No
- `xterm` or `vs100`

### restart delay
Enter, in seconds, the amount of time you wish the port server to wait—upon terminating an application on a port—before attempting to restart.

### wait for keyboard hit
Indicate whether you will require a user to press a key to initiate a user session.

### display banner
Indicate whether you want the login banner to be displayed upon login to user sessions. Deleting the login banner will cause it not to be displayed on the user login screen for ports on which the custom service configuration is enabled.

**NOTE**: There is only one login banner. You cannot define different login banners for different service configurations. The same login banner character string (either a customized banner of your choosing—defined with the `set banner` command—or the default banner) will be displayed on all user sessions when enabled.

### 1-key menus
Allows you to define command/ key correspondences that will allow users to execute specific commands by pressing single keys. The number keys 1 through 9 are used for 1-key assignments, providing up to nine 1-key menu selections. By then configuring one user session (described below) as `shell -1`, the 1-key menu assignments will be displayed.
on the user's terminal by key and command description (the menu is displayed when the shell session is active).

To define 1-key menu options, enter the information requested by the prompts:

- An alias key (number key 1 through 9) for this 1-key menu option.
- The description (name) to be assigned to this 1-key menu option.
- The execution command associated with this 1-key menu option. Enter the command string exactly as you would at the command shell.

**session**

Define up to four sessions users may switch between. For example, you can define one session as telnet, another as rlogin, etc. The BREAK key and the optional user-defined switch key is used to switch between sessions.

**NOTE:** For the 1-key menu options to be displayed to users, you must define one of the user sessions as shell -1. By doing so, the 1-key menu options, commands available to users, and information about using the BREAK key to switch between sessions will be displayed on the users' terminals when the shell -1 session is active.

When prompted for the session command line, enter the execution command associated with the session number shown in the prompt. The following commands are currently operational on the port server:

- telnet
- rlogin
- shell
- Other applications you wish to make available to users.

**NOTE:** Access to administrative and diagnostic commands is available to the capabilities level set for the service configuration (see [Defining Capabilities Access Levels](#) on page 107).

After completing your session requirements, the console terminal will display the custom configuration you defined and ask if the information is correct. You cannot modify this configuration. If you respond **y** (yes), the custom service configuration will be accepted; if you respond **n** (no), the configuration will be cleared, in which case you will need to define the configuration again.

If you accept the configuration as shown, you can enable it on a port or on multiple ports at any time. To enable C6 on ports 2 and 3, enter:

```
set C6 2,3
```

If you choose to define the configuration and enable it on a port or on multiple ports in one step, use the following command:

```
set custom C6 2,3
```

You will be presented with the prompts described above for the **add custom** command, but **set custom** eliminates the need to enable the configuration on a port or ports in a separate step.
Use the show command to identify the configuration settings of any custom service configuration. Enter:

show C6

to display a summary of the custom service configuration specified in the command; this summary will include a listing of the port(s) on which it is enabled.

To delete a custom service configuration from the port or port(s) on which it is enabled, enter:

delete C6 2,3

**Defining Capabilities Access Levels**

The port server offers several levels of capabilities and, within the custom service configurations, you can define the level or levels to which your users have access. Capabilities access levels range from user capabilities—with limited access to the port server’s full scope of capabilities—to an expert administrator (expert) level which includes access to all the functions and capabilities provided by the port server. You can combine access levels to provide users with the functions they need without jeopardizing administrative security. The access levels are listed below, along with a brief description of the functions included in each level. This organization is also summarized in Table 6-3.

Each access level (with the exception of user) allows access to the capabilities/commands available to that particular level and to all levels below it. For example, providing access to the admin level provides access not only to the admin capabilities/commands, but also to the capabilities/commands available through the to_admin, net, and user levels.

Most of the capabilities are actually commands that can be accessed directly by users who have been given access to them. For example, users who have access to the capabilities provided by the user and net access levels would be able to invoke the hangup, help, stty, telnet, rlogin, ppp, and slip commands directly. Each of the commands is described briefly below.

**Level 1: user**

User access allows a virtual terminal application (e.g., telnet, rlogin) to be selected, terminal type to be set, and stty configurations determined. The user level provides access to:

- **1-key menus**: allows a user to take advantage of any 1-key menus that have been configured.
- **hangup**: attempts to generate a hangup on the current line, terminating all sessions on the line.
- **help or (?)**: displays list of available commands.
- **term**: without an argument, shows the setting of the TERM environment variable; with an argument, sets the environment variable to that value.
- **stty**: sets or displays terminal characteristics for a device.
telnet: establishes a telnet session to a remote host (used to initiate a telnet session that was not defined as one of the user sessions).

rlogin: establishes a remote login session to a remote host (used to initiate an rlogin session that was not defined as one of the user sessions).

Level 2: net
In addition to providing access to all the capabilities of the user access level, the net capabilities level provides access for invoking SLIP and/or PPP connections directly. Users can initiate a SLIP or PPP session from the command line if the function was not provided as a custom configuration.

Level 3: to_admin
In addition to providing access to all the capabilities of the user and net capabilities levels, the to_admin capabilities level allows—with the proper administrative password—access to the admin command, from which the full range of administrator capabilities (described in levels 4 and 5, below) can be accessed. The administrative password is required of users who are assigned the to_admin capabilities level.

Level 4: admin
The admin capabilities level allows access to all the capabilities of the user, net, and to_admin capabilities, as well as commands and utilities for managing the port server. In addition, the admin level provides access to the expert level. Capabilities provided by the admin level are:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>add</td>
<td>used with various keywords, adds or defines an port server configuration option (see Configuration Commands page 88).</td>
</tr>
<tr>
<td>delete</td>
<td>used with various keywords, deletes or removes an port server configuration option (see Configuration Commands page 88).</td>
</tr>
<tr>
<td>set</td>
<td>used with various keywords, sets or enables an port server configuration option (see Configuration Commands page 88).</td>
</tr>
<tr>
<td>show</td>
<td>used with various keywords, displays port server configuration option settings (see Configuration Commands page 88).</td>
</tr>
<tr>
<td>save</td>
<td>saves configuration settings so that they will be retained following reboot.</td>
</tr>
<tr>
<td>exit</td>
<td>exits from the administrative or expert capabilities access level back to the previous level; does not terminate all sessions on the line.</td>
</tr>
<tr>
<td>drop</td>
<td>ends user session(s) active on the port(s) specified in the command.</td>
</tr>
<tr>
<td>http_stats</td>
<td>displays statistics concerning the operation of the HTTP server.</td>
</tr>
<tr>
<td>lpq</td>
<td>displays the print queue</td>
</tr>
<tr>
<td>lprm</td>
<td>deletes a job or jobs from the print queue.</td>
</tr>
<tr>
<td>printenv</td>
<td>displays all currently set environment variables (provides the same function as setenv without arguments).</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>setenv</td>
<td>sets or displays environment variables.</td>
</tr>
<tr>
<td>unsetenv</td>
<td>unsets environment variables, affecting all sessions on a line.</td>
</tr>
<tr>
<td>upgrade</td>
<td>enables optional features of the port server.</td>
</tr>
<tr>
<td>syslog</td>
<td>displays messages logged by system software.</td>
</tr>
<tr>
<td>pppstats</td>
<td>displays PPP interface statistics.</td>
</tr>
<tr>
<td>netstat</td>
<td>displays the contents of network-related data structures.</td>
</tr>
<tr>
<td>arp</td>
<td>displays and modifies the IP-to-Ethernet address resolution table.</td>
</tr>
<tr>
<td>ping</td>
<td>checks network connectivity.</td>
</tr>
<tr>
<td>kill</td>
<td>sends a signal to a process identified by the process ID number.</td>
</tr>
<tr>
<td>tty</td>
<td>identifies the port on which the user is currently running.</td>
</tr>
<tr>
<td>who</td>
<td>shows all current tty activity.</td>
</tr>
<tr>
<td>expert</td>
<td>accesses the expert commands, described below.</td>
</tr>
</tbody>
</table>

**Level 5: expert**

The expert level provides access to the highest level of capabilities, some of which may be used to monitor and/or troubleshoot the port server, possibly at the direction of a technical support representative. The expert capabilities include the following commands:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>apmem</td>
<td>Displays the amount of heap memory currently allocated to the specified process.</td>
</tr>
<tr>
<td>boot</td>
<td>sets firmware options.</td>
</tr>
<tr>
<td>dump</td>
<td>send a memory dump to the specified server.</td>
</tr>
<tr>
<td>flash_compact</td>
<td>reclaims space from the area of flash memory in which product information and error records are stored.</td>
</tr>
<tr>
<td>flash_log</td>
<td>displays the message log that is recorded in flash memory on the port server.</td>
</tr>
<tr>
<td>flash_save</td>
<td>saves port server configuration or software to a server system.</td>
</tr>
<tr>
<td>flash_update</td>
<td>load new software or configuration information.</td>
</tr>
<tr>
<td>lookmem</td>
<td>shows information on memory heap allocation and total memory allocation.</td>
</tr>
<tr>
<td>lookstream</td>
<td>shows STREAMS statistics to use for debugging.</td>
</tr>
<tr>
<td>port_stat</td>
<td>displays a one page overview of the status of all the ports on the port server.</td>
</tr>
<tr>
<td>processes</td>
<td>shows the active processes with their process IDs and priorities.</td>
</tr>
<tr>
<td>ifconfig</td>
<td>assigns an address to a network interface, or configures interface.</td>
</tr>
</tbody>
</table>
parameters.

**reboot:** performs an immediate reboot of the port server.

**prtr_test:** sends a predefined ASCII file to the selected port.

**quit:** performs the same function as exit; returns to the previous capabilities access level.

**trace_clear** clears the trace buffer.

**trace_display** displays the trace buffer.

**trace_off** turns off software tracing.

**trace_on** turns on software tracing.

---

**Table 5-3. Access Capabilities Overview**

<table>
<thead>
<tr>
<th>user</th>
<th>net</th>
<th>to_admin</th>
<th>admin</th>
<th>expert</th>
</tr>
</thead>
<tbody>
<tr>
<td>hangup</td>
<td>ppp</td>
<td>admin</td>
<td>add, arp, delete</td>
<td>apmem</td>
</tr>
<tr>
<td>help or ?</td>
<td>slip</td>
<td></td>
<td>drop, exit, expert</td>
<td>boot, dump</td>
</tr>
<tr>
<td>1-key menus</td>
<td></td>
<td></td>
<td>http_stats, kill</td>
<td>flash_compact</td>
</tr>
<tr>
<td>rlogin</td>
<td></td>
<td></td>
<td>lpq</td>
<td>flash_log</td>
</tr>
<tr>
<td>stty</td>
<td></td>
<td></td>
<td>lprm</td>
<td>flash_save</td>
</tr>
<tr>
<td>telnet</td>
<td></td>
<td></td>
<td>netstat</td>
<td>flash_update</td>
</tr>
<tr>
<td>term</td>
<td></td>
<td></td>
<td>ping, pppstats</td>
<td>ifconfig</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>printenv, save</td>
<td>lookmem</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>set, setenv</td>
<td>lookstream</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>show, syslog, tty</td>
<td>monitor</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>unsetenv, upgrade</td>
<td>port_stat</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>who</td>
<td>processes</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>prtr_test</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>quit</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>reboot</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>trace_clear</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>trace_display</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>trace_on</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>trace_off</td>
</tr>
</tbody>
</table>
DEFAULT

set default { port number(s) }
default { port number(s) }

show default

port numbers: Any valid port number for your port server model. May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range.

The default service configuration defines the following:

• One user session, configured with access to the user and to_admin capabilities (described later in this chapter).
• No 1-key menu selections configured.
• User authentication (password) required.
• The login banner is displayed when the user shell is displayed.
• A terminal type of vt100.

Use the set default command to enable the default service configuration on ports being directly connected to terminals and dial-in modems. For example, to enable the default service configuration on ports 1 through 4, enter:

set default 1-4

To delete the default service configuration from ports 3 and 4, enter:

delete default 3-4

To display a summary of the default service configuration settings and the ports on which the default service is enabled, enter:

show default
DIAL_ON_DEMAND

set dial_on_demand { interface number }

show dial_on_demand

delete dial_on_demand

Interface number: a number between 0 and 32 which is an arbitrary identifier for a defined DOD interface. The number does not correspond to a serial port number.

Use the dial_on_demand command to program the port server to automatically dial and make a connection with a remote system. When you enable this feature, the port server dials a remote system when IP traffic arrives and hangs up when there is no more activity.

You activate the dial on demand feature by using an application which sends IP packets to hosts on the remote side of the dial on demand interface. Associated with each interface is a system definition which specifies a name, phone number, one or more ports, and a dial or login script. Associated with each port is a modem and its dial script.

To define the parameters for making a dial on demand PPP connection to a particular system:

1. Define the system using the add system command (see SYSTEM, page 163).
2. Enter:

   set dial_on_demand {0-31}

   The number (0-31) is an arbitrary identifier for a defined DOD interface. The number does not correspond to a serial port number.

   You will be prompted for the following information:

   • system name
   • IP addresses of the local and remote sides of the link
   • a name for the interface
   • whether to invoke security and compression
   • which packets to route over the interface

   Unless you have some preference regarding the local and remote IP addresses, you can greatly simplify the configuration of the interface by allowing PPP to automatically negotiate its addresses. If you enter the value 0 in place of an IP address, PPP will set up a "Dynamically Addressed" interface and will attempt to negotiate the address(es) with the remote host during link establishment. This feature is also useful for setting up dial on demand links to Internet providers which usually want to provide their own IP addresses.

   In general, you will almost always want to add a route for this interface in order to begin routing packets over the dial on demand link. The set dial_on_demand command gives you the option to add the route automatically after you have entered
the interface information. Just enter the IP address of the destination network, or "default" if this interface is to handle all default routes.

Example:

To add a dial_on_demand configuration:

```
[admin:1/14]>>set dial_on_demand 1
Enter system name : cuda
Enter remote IP address
(Enter 0 to automatically negotiate) : 0
Enter local IP address
(Enter 0 to automatically negotiate) : 0
Enter netmask in IP address format
(Enter 0 to automatically determine) : 0
Enter an interface name (e.g. 'ppp0') : ppp1
Use Van Jacobsen header compression (y/n) : y

What kind of authentication should be used?
1 - Authenticate remote host using CHAP.
2 - Authenticate remote host using PAP.
3 - port server must provide authentication to remote host.
4 - No authentication is required by either side.

Please select a number 1 - 4: 4
Remote IP address : Negotiated
Local IP Address : Negotiated
Subnet Mask : Negotiated
Interface Name : ppp1
Authentication Type : No authentication is required by either side

Are these values correct (y/n) : y
Dial-on-demand interface dod4 added to the system.
Would you like to add a route for this interface? y
Enter the desination network (or 'default'): default
The requested route was added.
```

To display a list of configured dial-on-demand interfaces:

```
[admin:1/15*]>>show dial_on_demand
Configured Dial-On-Demand Interfaces
------------------------------------
dod1 : ppp -n ppp1 -t -u cuda 0:0 0
```

To delete a dial-on-demand configuration:

```
[admin:1/16*]>>delete dial_on_demand 1
Dial-on-demand interface dod1 has been deleted.
```
DLCI

**add dlci** *(dlci number)*

**delete dlci** *(dlci number)*

**show dlci**

The **add dlci** command adds a Data Link Connection Identifier (DLCI). A DLCI number identifies a permanent virtual circuit (PVC). A PVC is a connection available for sending and receiving data during a frame relay session. The **add dlci** command provides the port server with the information needed to correctly route information for a given DLCI number.

Before you can define a custom service configuration as frame relay, you must have at least one DLCI number. The DLCI number is provided to you by your frame relay service provider. You will be given an individual DLCI number for each PVC that will be available during your frame relay session. The port server supports up to 30 PVC’s. In most cases, the DLCI number for your first PVC will be the lowest available number—16.

To add a PVC for DLCI number 16, enter:

```
add dlci 16
```

You will be prompted for the following information.

1. **Enter PVC name**
   Enter the name you assign to this PVC (e.g. Cleveland). Although a name is not required, it is recommended.

2. **Enter PVC local IP address**
   Enter an available IP address on the network to which the port server is connected. Any local route you configure will use this address with a hop count of 0 (for further information, see [ROUTING](#) page 151).

3. **Enter PVC remote IP address**
   Enter an available IP address for the remote connection.

4. **Enter PVC netmask in IP address format:**
   Enter the IP subnet mask that corresponds to the remote IP interface.
DOMAIN NAME

set domain { domain name }

show domain

The set domain command defines the domain host name. The name you set becomes effective immediately; use the save command to ensure that the domain name is retained following reboot.

To set the domain name, enter the following command, substituting the actual domain name of the host system where indicated in the command:

        set domain mycompany.com

To view the domain host name and the method by which it was derived (e.g., Configuration Database, BOOTP, DHCP, etc.), enter:

        show domain
FRAME_RELAY

(NOTE: Frame Relay is an optional, added software feature, available to be implemented on the WAN port. Only some port server models have WAN ports. See the Hardware Manual provided with your port server to determine if your port server has a WAN port.)

set frame_relay \{Cn\} \{wan\|portnumber\}

show frame_relay

    portnumber: The number of the WAN port on your port server. For the RCS/ 5000 and RCS/ 6000, it is port 18; for the RCS/ 4000, it is port 33.

The set frame_relay command defines the custom service configuration specified (Cn) as frame relay and enables it on the specified port. Before you can issue a set frame_relay command, you must have added a dlcI number using the set dlcI command (see DLCI, page 114).

To define custom service configuration C6 as frame relay and enable it on port 33, enter:

    set fr C6 33

You will be prompted to enter the type of Link Management Interface (LMI).

    Type of LMI:

Enter the type of Link Management Interface you have leased: ANNEX_D, LMI, or ANNEX_A.

Additional prompts will appear. In most cases, the default will be the preferred choice. Only change the default (shown in the brackets) to meet a specific need.

<table>
<thead>
<tr>
<th>Prompt</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>N391 count [6]</td>
<td>number of Link Integrity Verifications between Full Status Queries.</td>
</tr>
<tr>
<td>N392 count [3]</td>
<td>number of errors permissible in a N393 time period.</td>
</tr>
<tr>
<td>N393 count [4]</td>
<td>number of good status checks needed to bring the error count to 0.</td>
</tr>
<tr>
<td>T391 timer [10]</td>
<td>number of seconds between the sending of status queries.</td>
</tr>
</tbody>
</table>

You will be asked if you are sure you want to enable the configuration on the WAN port.

To see your frame relay configurations, enter:

    show fr
HOST NAME LIST

add host { host name } { IP address }

delete host { IP address | IP name | all }

show host

The host name list is an unordered list containing all the hosts to which the port server has access. Entries to this list take effect immediately, but must be saved to be retained following a reboot.

Because the host name list is an unordered list, you do not need to worry about the order in which the names appear—you can add and delete as desired without regard to where the names appear on the list.

- The **add** command adds a host to the bottom of the list.
- The **delete** command either deletes a specific host from the list or deletes all host names from the list (using the **all** parameter).
- The **show** command displays the entire list.

Both the host name and the host’s IP address are required in the command (in that order). To delete a host from the list, you need enter only one identifier—either the name or the IP address. Host names cannot begin with numeric characters.

The command for adding a host to the list is:

```
add host saturn 199.188.10.1
```

The command for deleting a specific host is:

```
delete host saturn
```

To delete all host names from the list, enter:

```
delete host all
```

To view the entire host list, enter:

```
show host
```
IGNOREDCD

set ignoredcd { port number(s) } { yes|no } [i]
show ignoredcd [ port number(s) ]

port numbers: Any valid port number for your port server model. May be entered as
individual numbers, as a range of port numbers, as “all”, or as a combination of
individual numbers and a range.

The port server normally requires the presence of the DCD modem signal on a serial port
before it will start the defined service on the port. And it will shutdown the service on the
port if it loses the DCD signal. You may override this by configuring the port to ignore
the DCD modem signal. Entering **yes** configures the port server to ignore DCD on session
startup and to not care about the loss of DCD. Entering **no** configures the port server to
require the DCD signal to start and maintain the service on the port.

To set all ports to require DCD, enter the following:

```
set ignoredcd all no
```

**NOTE**: Append the **i** option to the command if you want the port
configuration to take effect immediately. For example:

```
set ignoredcd all no i
```

Save the configuration (see the [save] command, page 232) if you want the configuration to
be maintained following reboot.

To determine the DCD setting for each port server serial port, enter:

```
show ignoredcd all
```
INPUT

set input { port number(s) } { none | xoff | rtscts | dtrdsr } [ i ]
show input { port number(s) | all }

port numbers: Any valid port number for your port server model. May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range.

Use the set input command to set software and/ or hardware flow control of the connected device’s data input to the port server. Input flow control is illustrated in figure 4-1. (Note that output flow control is discussed later in this section.)

Input software flow control can be set as none or xoff; input hardware flow can be set as none, rtscts (Ready-to-Send/ Clear-to-Send), or dtrdsr (Data Terminal Ready/ Data Set Ready).

To set input software flow control to xoff on all the ports, enter:
   set input all  xoff
To set input hardware flow control to rtscts on all ports, enter:
   set input all  rtscts
Alternately, you can set input software and hardware flow control in one command:
   set input all  xoff  rtscts

NOTE: Append the i option to the command if you want the port configuration to take affect immediately. For example:
   set input all  xoff  rtscts  i

Save the configuration (using the save command) if you want the configuration to be maintained following reboot.

To determine the input flow control setting of each port server serial port, enter:
   show input all

The input flow control setting of each port will be listed. To display input software flow control information for specific ports, append the port numbers to the show input command.

NOTE: Setting input hardware flow control automatically sets output hardware flow control as well.
controls connected device’s software flow control input

none | x-off
controls connected device’s hardware flow control input

none | rtscts | dtrdsr

set input command

Figure 5-1. Input Flow Control
INTERFACE

set interface { port number(s) } { rs232 | rs422 | rs485 | cloop } [ i ]
show interface { port number(s) }

port numbers: 1, 2, 3, 4. May be entered as individual numbers, as a range of port numbers, as “all”, or as a combination of individual numbers and a range.

The set interface command sets the port to the specified physical layer interface. The default is RS-232.

To set the interface for port 3 to RS-485, type:
  set interface 3 rs485

To have the interface configuration applied immediately, rather than when the RCS/3000 reboots, use the i option.
  set interface 3 rs485 i

To display the current interface configuration setting for port 3, type:
  show interface 3

If no port is specified in the show interface command, the interface configuration for all ports will be displayed.

Current loop mode is available only on ports 7 and 8 of the POS/ 2000. To set current loop mode on port 7, enter:
  set interface 7 cloop
IP ADDRESS

set ip { IP address }

show ip

Use the set ip command to set the IP address for the port server. A frame type of ethernet is automatically selected with the set ip command. The IP address you set will take effect after reboot. However, so that it can take effect after reboot, you must save (see the save command, page 232) the IP address into the current configuration database.

For example, to set the IP address of your port server to 199.171.148.1, enter:

    set ip 199.171.148.11

When a show command is associated with a set command, it shows the status of that particular set command. In this case, entering:

    show ip

displays both the current IP address and the method by which it was derived (Configuration Database, BOOTP, DHCP, etc.). If the IP address was derived from DHCP, the lease information is displayed.

You can clear the IP address by entering:

    set ip 0.0.0.0
MAC

show mac

The **show mac** command displays the Ethernet MAC address of the port server.
**MASK**

`set mask { value }`

`show mask`

The subnet mask number is listed as the total number of bits in the network address. The command for setting the subnet mask is:

```
set mask 255.255.255.0
```

When a `show` command is associated with a `set` command, it shows the status of that particular set command. In this case, entering:

```
show mask
```

displays both the subnet mask you previously set and the method by which it was derived (e.g., Configuration Database, BOOTP, DHCP, etc.).

Setting the subnet mask to 0.0.0.0 (default) instructs the port server to determine the subnet mask dynamically from the IP address. In general, IP addresses fall into the following classes with their associated subnet masks:

<table>
<thead>
<tr>
<th>Class</th>
<th>IP range</th>
<th>Subnet Mask</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1.x.x.x - 126.x.x.x</td>
<td>255.0.0.0</td>
</tr>
<tr>
<td>B</td>
<td>128.x.x.x - 191.x.x.x</td>
<td>255.255.0.0</td>
</tr>
<tr>
<td>C</td>
<td>192.x.x.x - 223.x.x.x</td>
<td>255.255.255.0</td>
</tr>
</tbody>
</table>
MODEL

show model
The show model command displays the model number of the port server.
MODEM

add modem { modem name }
set modem { port numbers } { modem name }
show modem [ modem name ]
delete modem { modem name }

port numbers: Any valid port number for your port server model. May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range.

The modem commands are used to configure modems for dial-on-demand connections. There are two steps to configuring the modem on ports:
1. Define a modem, assigning a name to the modem definition.
2. Apply the modem definition to one or more ports.

There is a pre-defined modem definition, called generic, set up in the port server or you can create your own modem definition. A modem definition consists of a modem chat script and an initialization string. The values for the pre-defined modem are:

<table>
<thead>
<tr>
<th>modem name</th>
<th>chat script</th>
<th>initialization string</th>
</tr>
</thead>
<tbody>
<tr>
<td>generic</td>
<td>&quot;&quot;&quot;ATDT\T</td>
<td>at</td>
</tr>
</tbody>
</table>

If the supplied modem definition does not meet your needs, you can create your own modem definition. The command to define a modem is:

```
add modem { modem name }
```

The `add modem` command defines a modem and its associated modem chat script and initialization string. Enter the name of a single modem (such as hayes, hayes1, courier) as the argument. The modem name may be any alphanumeric string, up to 15 characters in length.

After you enter the command, you are prompted for the modem chat script. Enter a script up to 256 characters (see Writing Chat Scripts, page 277).

Next, you are prompted for the modem initialization string. The string is normally sent out the port when the port server first attempts to start a service on the port. It is especially important to specify a modem initialization string for dial-in lines, since the string is used to train the modem baud rate.

Example:

```
[admin:1/1]>>add modem hayes1
Enter the modem chat script: """"AT"""" OK"""" ATDT\T"
Enter the modem init string: AT
[admin:1/2]>>
```

The command to apply the modem definition to a port is:
set modem \{port number(s)\} \{modem name\}

The set command assigns the modem name to one or more port server ports. If you omit
the modem name, the set modem command deletes any modem set to the specified port
or ports.

**Example:**

[admin:1/2]>>set modem 0,1 hayes1
Modem hayes1 has been applied to port 0.
Modem hayes1 has been applied to port 1.
[admin:1/3]>>

**Example:**

[admin:1/3]>>set modem 0
The modem applied to port 0 has been removed.
[admin:1/4]>>

To display the configured modem and its chat script, enter:

show modem hayes1

Entering show modem with no modem name lists all the defined modems.

**Example:**

[admin:1/4]>>show modem hayes1
Modem Name: hayes1
Dial Chat Script: " " "AT" "OK" "ATDT\T"
Modem init string: AT
[admin:1/5]>>

To delete the configured modem, enter:

[admin:1/5]>>delete modem hayes1
[admin:1/6]>>

Modem will not be deleted from port 0 or 1.
MPOOL

set mpool \{ port number(s) \} \{ pool number \}
delete mpool \{ port number(s) \} \{ pool number \}
show mpool \{ pool number \}

port numbers: Any valid port number for your port server model, including 17 or 32 (parallel port). May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range.
pool number: The number of the pool you wish to define; valid pool numbers are 0, 1, 2, 3, and 4.

The set mpool command configures ports into individual modem pools. These ports can then be accessed via an rtelnet connection to the pool's TCP port number. Modem pools 0 through 4 are mapped to TCP ports 9100 through 9104 (telnet mode) or 8100 through 8104 (raw TCP mode), respectively. To configure ports 1 and 2 and 4 through 7 into mpool 2, enter:

set mpool 1,2,4-7 2

You can then delete ports 2 and 4 from mpool 2 by entering:

delete mpool 2,4 2

To determine what ports are in mpool 2, enter:

show mpool 2

To determine which mpools currently contain ports, use the command

show rtelnet

which displays a listing of the TCP mappings for tty ports, as well as mpools. If an mpool has ports configured into it, the ports will be listed in the output.
NAME SERVER LIST

add nameserver { IP address }
delete nameserver { IP address | all }
show nameserver

The port server allows you to list up to four DNS servers, in order of precedence, with “1” being the primary server. Entries to the name server list take effect immediately, but must be saved to be retained following a reboot. You determine the order in which the servers appear on the list, and thus are accessed. The port server offers three commands for maintaining this list: add, delete, and show.

• The add command adds a name server to the bottom of the list.
• The delete command deletes either a specific name server or all name servers from the list.
• The show command displays the entire list so that you can see the ordering of your name servers.

NOTE: If you do not first set a domain name for a remote host (see DOMAIN NAME page 115), you must enter the remote host’s full IP address in any command used to access that host.

The command for adding the first name server to the list (it will appear in the primary position) is:

add nameserver 198.130.71.2

To add a second name server in the secondary position, repeat the command, entering the IP address of the second name server.

The command for deleting a name server is:

delete nameserver 198.130.71.2

To delete all name servers from the list, enter:

delete nameserver all

To view the entire name server list and the method by which it was derived (e.g., Configuration Database, BOOTP, DHCP, etc.), enter:

show nameserver
NETWORK ADDRESS TRANSLATION (NAT)

add nat
add nat { interface }
add nat { interface } { num entries }
add nat { interface } { num entries } { timeout }
show nat { all }
show nat { interface } [ debug ]
delete nat [ interface ]

Network Address Translation (NAT) is a port server feature which can be enabled on any port server interface in order to allow TCP and UDP packets from private networks to be forwarded to hosts on public networks (such as the Internet). It is difficult for hosts on private networks (i.e. internal networks that are not registered on the Internet) to communicate with hosts on the Internet because Internet routers do not know how to find private hosts.

It is possible to connect to the Internet without a registered public network address by dialing into an Internet Service Provider, but in this case the service provider assigns you a single public address which can only be used by one host on the private network. If the port server is the router providing the Internet link, this limitation can be overcome by enabling the port server to perform Network Address Translation. The NAT functionality of the port server changes the source IP Address-Port pair (or socket) of all outgoing TCP and UDP packets to appear as if they originated from the public network address of the port server.

The port server implements NAT by maintaining an internal table that maps source TCP/UDP sockets to local TCP/UDP ports on the port server. The local ports start at 10000 and extend to the maximum limit specified by the NAT configuration. When the port server receives a packet that is to be routed over the NAT interface, it saves the original socket information, modifies the packet's source IP address to equal the IP address of the public interface (specified by the service provider), and modifies the packet's source port to equal the mapped NAT port on the port server. When the port server receives an incoming packet from the public network, it reverses the process to return the packet to the correct private host.

Each TCP and UDP connection requires a new map entry in the NAT table. These mappings remain in the table until the table fills up. At this point, new entries begin to replace old entries, with the oldest (unused) entries being replaced first.

Network Address Translation is enabled on an interface by using the `add nat` command. The `add nat` command requires three parameters: interface name, number of entries, and timeout. The interface name is the name of the interface on which you wish to enable NAT (e.g., ppp0). The number of entries specifies the size of table and timeout specifies how long an unused NAT mapping must remain in the NAT table before it can be replaced by a new map.
Larger tables allow more outstanding connections and reduce the probability of losing older inactive connections. However, larger tables consume more memory and slow down packet forwarding. For smaller networks, typical table sizes range between 50 and 1000 entries. The timeout value is used to prevent older connections from being dropped too soon, but larger timeout values cause the table to reach capacity sooner. Typical timeout values range from 5 to 20 minutes.

Example

```
[admin:1/24*]>>add nat
Enter the following information to enable Network Address Translation (NAT) on an interface ('?' for help).

Interface Name : ppp0
Number of Translation Entries <default 50> : 
Timeout for Old Entries (sec) <default 300> : 

The following NAT interface entry will be added:

Interface Name : ppp0
Number of Translation Entries : 50
Timeout for Old Entries (sec) : 300

Are these value correct? y
NAT interface 'ppp0' has been added.
The NAT configuration has also been applied to the interface.
```

You can view the current NAT configuration with the `show nat` command. With the `all` argument, the command displays a list of all the NAT interfaces defined in the configuration database. If you specify a particular interface, the NAT statistics associated with the interface are also displayed. The optional `debug` argument can be included to display the actual contents of the NAT tables.

The statistics can be used to determine if your table size is too small. A large number of "Table Full Errors" indicates that the table can not allocate enough entries for outgoing connections. This condition can be solved by increasing the size of the table and/ or reducing the timeout value. A large number of "Invalid Incoming Port Maps" indicates that older entries are being replaced too quickly. This condition can be solved by increasing the size of the table and/ or increasing the timeout value.

Example

```
[admin:1/24*]>>show nat ppp0
[Actual NAT Configuration]

Interface Num Entries Timeout (sec)
--------- ----------- -------------
ppp0 50 300

NAT Statistics:

Translation Statistics

Successful Outgoing Translations : 104
Table Full Errors : 0
Successful Incoming Translations : 81
Invalid Incoming Port Maps : 0
```
NAT Entry Statistics

---------------------
Number of Table Entries Used : 4
Number of Table Entries Expired : 0
Number of Closed Connections Replaced : 5
Number of Unsupported Protocol Packets : 0

Hash Table Statistics

-------------------------------
Percent Hash Table Utilization : 12
Longest Hash Chain : 2

To remove NAT functionality from an interface:

   [admin:1/25*]>>delete nat ppp0
   The NAT interface 'ppp0' has been deleted.

   The NAT configuration has also been removed from the interface.
OUTPUT

set output { port number(s) } { none | xon | xany | rtscts | dtrdsr } [ i ]

show output { port number(s) | all }

port numbers: Any valid port number for your port server model. May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range.

Use the set output command to set software and/or hardware flow control of the port server's data output. Figure 6-2 illustrates output flow control.

Output software flow control can be set as none, xon, or xany; output hardware flow control can be set as none, rtscts (Ready-to-Send/ Clear-to-Send), or dtrdsr (Data Terminal Ready/ Data Set Ready).

NOTE: dtrdsr output flow control will only work if pin1 has been configured to be treated as DSR.

To set output software flow control to xany on ports 3 through 7, enter:

set output 3-7  xany

To set output hardware flow control to rtscts on ports 3 through 7, enter:

set output 3-7  rtscts

Alternately, you can set output software and hardware flow control in one command:

set output 3-7  xany  rtscts

NOTE: Append the i option to the command if you want the port configuration to take affect immediately. For example:

set output 3-7  xany  rtscts i

Save the configuration (using the save command) if you want the configuration to be maintained following reboot.

To determine the output flow control setting of ports 3 through 7, enter:

show output 3-7

The output setting of ports 3 through 7 will be listed. To display output flow control information for all the port server's serial ports, use the all parameter with the show output command.

NOTE: Setting output hardware flow control automatically sets input hardware flow control as well.
Port Server

```
output data
```

| Device

- `none | x-on | x-any`
  - controls connected device's software flow control output
- `none | rtscs | dtrdsr`
  - controls connected device's hardware flow control output

```
set output command
```

Figure 5-2. Output Flow Control


**OWNER**

```plaintext
set owner { port number (s) } { string }
```

```plaintext
show owner { port number (s) | all }
```

- **port numbers**: Any valid port number for your port server model. May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range.

The **owner** setting is used by host reverse telnet software (such as NativeCOM) to reserve a port. The port server does not make any use of this information. The string may be any value, but is generally set to the hostname and host port name.

To set the owner for port 1, enter:

```plaintext
set owner 1 nthost1
```

To delete the owner for port 1, enter:

```plaintext
set owner 1 ""
```

Save the configuration (using the `save` command) if you want the configuration to be maintained following reboot.

To determine the owner of each port, enter

```plaintext
show owner all
```

The owner of each port will be listed. To display owner information for specific ports, specify the port number(s) instead of all.
PARITY

set parity { port number(s) } { even | odd | none } [ i ]

show parity { port number(s) | all }

port numbers: Any valid port number for your port server model. May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range.

The port server offers parity setting options of even, odd, and none. To set ports 1, 2, and 3 to even parity, enter:

    set parity 1-3 even

**NOTE**: Append the i option to the command if you want the port configuration to take affect immediately. For example:

    set parity 1-3 even i

Save the configuration (using the `save` command) if you want the configuration to be maintained following reboot.

To determine the parity setting of each port server serial port, enter:

    show parity all

The parity setting of each port will be listed. To display parity information for specific ports, append the port numbers to the `show parity` command.
PASSWORDS

set user { password }
set admin { password }

As system administrator, you define the passwords required of both users and yourself. The password you set takes effect on next login, but must be saved to be retained following reboot.

All users on a particular port server must be assigned the same user password. To set the user password, enter the following command, replacing the word “password” with the desired user password:

    set user secret

You will set the administrative password in the same manner. Enter the following command, substituting the password you have selected where indicated:

    set admin secret2

**NOTE**: There is no show command associated with the passwords. As administrator, you have the option of changing the passwords, but not viewing either the user or administrative password once you have defined them.
PIN1

(NOTE: RCS/4000 and RCS/5000 only)

`set pin1 { port number(s) } { dcd | dsr } [ i ]`

`show pin1 { port number(s) | all }`

port numbers: Any valid port number for your port server model. May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range.

By default, pin1 of the port server serial port connector is treated as the incoming DCD (Data Carrier Detect) signal and there is no incoming DSR (Data Set Ready) signal. In some cases (such as for flow control or for reporting modem signal), the DSR signal is required. The pin1 setting allows you to select whether pin1 will be treated as DCD or DSR.

To set pin1 to DCD on port 1, enter:

`set pin1 1 dcd`

To set pin1 to be DSR on port 1, enter:

`set pin1 1 dsr`

If you want the port configuration to take effect immediately, add the i option. For example:

`set pin1 1 dsr i`

Save the configuration (using the `save` command) if you want the configuration to be maintained following reboot.

To determine the pin1 setting of each port, enter:

`show pin1 all`

The pin1 setting of each port will be listed. To display pin1 information for specific ports, specify the port number(s) instead of all.
PORT

add port { confign }

set port { confign } { port number(s) } [ i ]

show port { port number(s) | configuration name | all }

port numbers: Any valid port number for your port server model. May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range.

configuration name: The name of the serial port configuration you wish to view.

There are three modifiable port configuration sets, numbered 1, 2, and 3. You can modify each of these sets to meet the needs of your network environment, then enable each set on a different port or ports. The number of the port configuration set you wish to modify or enable replaces the \( n \) in the command syntax above.

For example, to modify port configuration number 2, enter:

add port config2

The current port settings for modifiable port configuration 2 will be displayed.

You will then be asked if you want to modify the configuration. Respond either \( y \) (yes) or \( n \) (no). A "no" response aborts the add port command. A "yes" response allows you to change each of the port characteristics listed. After changing each of the characteristics, the new configuration is displayed and you will be asked if the values are correct. A "yes" in this instance will accept the new values; a "no" will again abort the add port command.

Use the show port command to display any modifiable configuration set you defined with the add port command. For example, to view configuration number 2, enter:

show port config2

Once you have modified a port configuration set to meet your needs, you can enable it on a port or ports using the set port command. For example, to enable configuration number 2 on ports 1, 2, and 4, enter:

set port config2 1,2,4

**NOTE**: Append the \( i \) option to the command if you want the port configuration to take affect immediately. For example:

set port config2 1,2,4 i

Save the configuration (see the [save] command, page 232) if you want the configuration to be maintained following reboot.

If you want to view the configuration enabled on those ports, enter:

show port 1,2,4

To display the configurations of all the serial ports, enter:

show port all
PORTDESC

set portdesc { port number(s) } { string }
set pd { port numbers(s) } { string }
show portdesc { port number(s) | all }
show pd { port number(s) | all }

port numbers: Any valid port number for your port server model. May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range.

The portdesc setting is a user specified string to allow the user to identify the port. The port server does not make any use of this information. The string may be any value, up to 256 characters.

To set the portdesc for port 1, enter:

   set portdesc 1 "32 bit scanner"

To delete the portdesc for port 1, enter

   set portdesc 1 ""

Save the configuration (using the save command) if you want the configuration to be maintained following reboot.

To determine the portdesc of each port, enter:

   show portdesc all

The portdesc of each port will be listed. To display portdesc information for specific ports, specify the port number(s) instead of all.
POWER

(Note: Power configuration available only for POS/2000)

set power { port number(s) } {none | 5 | 12 } [ i ]
show power { port number(s) }

| port numbers: 1,2...8. May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range. |

The `set power` command configures one or more specified ports for power. There are two steps to assigning power to a port:

1. Configure the port for power using the `set power` command
2. Apply power to the port using the `power` command, page 216

The two steps can be combined in step 1 if the i option is used with the `set power` command.

The power configuration may be none (no power), 5 (+5 Volts), or 12 (+12 Volts). The default value is none. To configure port 7 for 5 volts, enter:

```
set power 7 5
```

You can now issue a `power` command to apply power to port 7.

Alternatively, you can both configure port 7 for power and turn on the power in one step using the i option. Enter:

```
set power 7 5 i
```

After entering this command, the power is turned on for port 7; it is not necessary to enter a `power` command.

When the power is turned on, there is a delay of approximately 1 second for each specified port sequentially before the power is actually enabled.

Save the configuration (using the `save` command) if you want the configuration to be maintained following reboot. If you save the configuration after issuing the `set power` command, either with or without the i option, the specified port(s) will have power turned on whenever the system reboots.

There are situations in which the power to a configured port is turned off:

1. The total peripheral power consumption exceeded the power limit (45 watts) allowed. A message is displayed:
   
   `Automatically turned off`

2. The peripheral power consumption on the port exceeded the port current limit (1 Amp). A message is displayed:

   `Over current protection tripped`
3. The power was manually turned off using the `power` command. A message is displayed:

```
Manually turned off
```

When the power has been turned off to a configured port, the `power` command can be used to restore the power to the port. Or power will be restored to the port if the POS/2000 is rebooted.

Changing the power configuration from one voltage to another cannot be done in a single step. If a port is currently configured for a voltage (e.g., 5) and you issue a `set power` command for the port for a different voltage (e.g., 12), a warning message will be displayed and you will be asked to confirm the power setting change. For example, if port 7 is currently configured for 5 volts and you enter:

```
set power 7 12 i
```

you will see the following message:

```
Port 7 is currently configured for 5 volts. Are you sure you want to change it to 12 volts (y/n)?
```

If you respond with `y`, the configuration will be changed but the power will be turned off and the following message displayed:

```
Power to port 7 has been turned off.
```

You can use the `power` command to restore the power to the port.

To display the power configuration for all ports, enter:

```
show power all
```

Configurations for all ports and power statistics for the POS/2000 overall will be displayed:

<table>
<thead>
<tr>
<th>Port</th>
<th>Config</th>
<th>Current Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>none</td>
<td>off</td>
</tr>
<tr>
<td>2</td>
<td>none</td>
<td>off</td>
</tr>
<tr>
<td>3</td>
<td>+5</td>
<td>on</td>
</tr>
<tr>
<td>4</td>
<td>+12</td>
<td>on</td>
</tr>
<tr>
<td>5</td>
<td>none</td>
<td>off</td>
</tr>
<tr>
<td>6</td>
<td>+5</td>
<td>OFF Automatically turned off.</td>
</tr>
<tr>
<td>7</td>
<td>+5</td>
<td>OFF Over current protection tripped.</td>
</tr>
<tr>
<td>8</td>
<td>+12</td>
<td>OFF Manually turned off.</td>
</tr>
</tbody>
</table>

+5 volt peripheral power: 1.21 watts
+12 volt peripheral power: 0.43 watts
Total peripheral power: 1.64
**PPOOL**

**set ppool** { port number(s) } { pool number }

**delete ppool** { port number(s) } { pool number }

**show ppool** { pool number }

- **port numbers:** Any valid port number for your port server model, including 32 (parallel port). May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range.

- **pool number:** The number of the pool you wish to define; valid pool numbers are 0, 1, 2, 3, and 4.

The set ppool command configures ports into individual printer pools. These ports can then be accessed via reverse telnet connection to the pool’s TCP port number. Printer pools 0 through 4 are mapped to TCP ports 9105 through 9109 (telnet mode) or 8105 through 8109 (raw TCP mode), respectively. To configure ports 1 and 2 and 4 through 7 into pool 2, enter:

```
set ppool 1,2,4-7  2
```

You can then delete ports 2 and 4 from pool 2 by entering:

```
delete ppool 2,4  2
```

To determine what ports are in pool 2, enter:

```
show ppool 2
```

**NOTE:** To determine which ppools currently contain ports, use the **show rtelnet** command. Enter:

```
show rtelnet
```

to display a listing of the TCP mappings for tty ports, as well as ppools. If a ppool has ports configured into it, the ports will be listed in the **show rtelnet** output.
PPP

**set ppp** \{ \textit{Cn} \} \{ \textit{port number} \}

**show** \{ \textit{Cn} \}

PPP (point-to-point protocol) is used as a remote connection for transmitting IP datagrams over serial point-to-point links (refer to Appendix C for a discussion of PPP). You can configure a serial port to use PPP for any data transmitted using the **set ppp** command as described in this section.

**NOTE**: You can also establish a direct PPP connection from the command line. If you are dialed in, know an IP address to use, and have that IP address configured on your personal computer, you can type **ppp** at the command prompt (see **ppp** page 219) to establish a connection.

The **set ppp** command defines that the custom service configuration (C0-C33) specified in the command be configured as PPP and that the PPP configuration be enabled on the port listed. You can enable PPP on any one of the serial ports (0 through 15).

A PPP service configuration can only be enabled on one port at a time. To run multiple PPP connections, you must create a different PPP configuration—with different IP addresses—for each connection.

The PPP configuration allows dial-in auto-PPP or dial-out PPP capability on the port associated with that particular custom service configuration. PPP cannot be combined with other custom service configuration options (such as multiple user sessions, 1-key menus, capabilities access).

To define custom service configuration C6 as PPP and enable it on port 2, enter:

```
set ppp C6  2
```

After entering the **set ppp** command, you will be prompted for the following information:

- Remote IP address
- Local IP address
- Subnet mask value in IP address format
- Interface Name
- Authentication type (PAP, CHAP, or none)
- If you want to use Van Jacobsen compression
- If you want to add a route for this interface

When prompted for the Remote IP address, Local IP address, and Subnet mask value, you will be given the option to enter 0 to automatically negotiate the values. If you do not need to define a specific IP address, enter 0 and the port server will negotiate an IP address with the remote host.

Interface name is a user-defined name under which this interface will be stored. The string can be up to 64 characters in length.
If you select either PAP or CHAP, you will be prompted for a peer id/name. You may enter a peer id/name, or select the default or none. In addition, when using PAP or CHAP, you will need to create a PPP authentication list. For information on creating an authentication list, see the `PPPAUTHLIST` command, page [147](#).

You will be prompted:

> Are these values correct (y/n) :

Respond `y` (yes) to confirm the values or `n` (no) to abort the command. A `yes` response enables the PPP configuration you defined as custom service configuration C6 on port 2.

The PPP implementation defined with the `set ppp` command does not use password key authentication, any authentication protocol, or IP address negotiation. The default ACCM is 0x00000000 and the default packet size is 1500. The maximum frame size is 1500.

For the PPP service configuration to take effect, you must save it (see the `save` command, page [232](#)), then reboot your port server.

Use the `show` command to determine information about a specific custom configuration or to display information about all the custom service configurations. For example, to view the configuration of custom service C6, enter:

```
show C6
```

To display information about all custom service configurations, enter:

```
show C*
```

The above show commands will display a summary describing the custom configuration(s) specified in the command and list the port(s) on which the configuration is enabled.

As with other custom configurations, use the `delete` command to delete a PPP custom service configuration from a port on which it has been enabled. For example, to delete the PPP service configuration that was configured as C6 and enabled on port 4, enter:

```
delete C6 4
```
PPPADLIST

add pppadlist { pool name } { remote IP address } { local IP address }

delete pppadlist { pool name | all }

show pppadlist

The add pppadlist command allows you to set up a pool of IP addresses that can be used for PPP connections. A PPP address pool can contain one or more entries.

The following command adds an entry to a PPP address pool named sales:

add pppadlist sales 199.171.148.1 198.130.71.2

If the pool named sales does not already exist, the above add pppadlist command will set up the pool with the specified IP address pair as the first entry. If the pool, sales, already exists, the specified IP address pair will be added as the last entry in the existing pool.

The end user can then use the ppp command to establish a PPP connection using the IP address pairs in the pool, sales. For example, if you set up a pool named sales, containing three different remote IP addresses, the ppp command to establish a PPP connection is:

ppp -P sales

The first available entry in the specified pool (sales) of IP address pairs will be used for the connection.

Use the delete pppadlist command to delete entries in a PPP pool or pools. You can specify either a pool name or the parameter, all. If a pool name is specified, the delete pppadlist command deletes the first entry in the named pool. If all is specified, all entries in all pools are removed. For example:

delete pppadlist sales

deletes the first entry in the PPP pool named sales.

To delete all entries in all PPP pools, type:

delete pppadlist all

To display the existing list of PPP pools, enter:

show pppadlist
PPPAUTHLIST

add pppauthlist { peer id | name } { password | secret }

delete pppauthlist { peer id | name } { password | secret }

show pppauthlist

peer id/name; password/secret: PPP authentication protocols PAP and CHAP use a peer ID (PAP) or name (CHAP), matched to a password (PAP) or secret (CHAP), to authenticate users.

You can use either the Password Authentication Protocol (PAP) or Challenge-Handshake Authentication Protocol (CHAP) with PPP custom service configurations to determine whether the remote end PPP connection is a legitimate user. A PPP connection involves "two ends" (peers), and each is treated by the other as a peer-user. A peer is recognized by an identification of some type and an associated password. For PAP, this peer pair is referred to as peer-ID/ password, and for CHAP, as name/ secret.

The add pppauthlist command allows you to define a database of peer id/ name and password/ secret pairs. Entering an asterisk (*) at the beginning of one peer id/ name indicates the default pair to be used if the user does not explicitly specify a peer id/ name. If you do not specify a default peer pair, PPP will use the system's host name as the default.

The command for adding entries to the PPP authentication database is:

    add pppauthlist { peer id | name } { password | secret }

    NOTE: To specify a default peer pair, type an asterisk (*) as a prefix to the peer id/ name portion of the peer pair. Do not include a space between the asterisk and the peer id/ name.

The command for deleting an entry from the authentication database is:

    delete pppauthlist { peer id | name } { password | secret }

To view the existing database of peer pairs, enter:

    show pppauthlist
RADIUS

add radius
add radius {auth|acct}
add radius {auth|acct} {hostname}
add radius {auth|acct} {hostname} {shared secret}
add radius {auth|acct} {hostname} {shared secret} {UDP Port}
add radius {auth|acct} {hostname} {shared secret} {UDP Port} {retransmit ms}
show radius
show radius {hostname}
delete radius
delete radius {auth|acct}
delete radius {auth|acct} {hostname}

Use the add radius command to add a new RADIUS server to the list of valid RADIUS servers. This list defines all the RADIUS servers that will be queried during a RADIUS authentication request. The list is ordered, so that the first server in the list will be the first one queried, and so on.

Two lists of RADIUS servers are maintained. The first list is for RADIUS Authentication servers and is denoted on the command-line using the keyword "auth". The second list is for RADIUS Accounting servers and is denoted using the keyword "acct". Leaving either of the lists empty will disable the corresponding RADIUS functionality.

Each RADIUS server has a set of parameters associated with it. The parameters, "hostname" and "Shared Secret", are required. "hostname" specifies the IP address (or resolvable hostname) of the new RADIUS server and "Shared Secret" corresponds to the encryption key to be used on that server.

The remaining parameters are "UDP Port" and "Retransmit Timeout (ms)". These parameters correspond to the RADIUS server's UDP port number and the time (in milliseconds) to wait for a RADIUS response. These two parameters both have default values which should be sufficient for most configurations.

Use the show radius command to view the current ordered lists of RADIUS servers defined on the port server. You can also specify an optional hostname to show more detailed information about a specific RADIUS server.

Use the delete radius command to remove a RADIUS server from one of the port server's server lists. If you do not specify the name of the server on the command line, you will be prompted for one.

When RADIUS functionality is enabled on a port (see CUSTOM, page 102), the port server will send authentication and accounting information to each server in the respective list until it receives a response or runs out of servers. It then starts again at the
beginning of the list, retrying each server four times before rejecting the authentication request and logging an error to syslog.

The RADIUS authentication server must be configured to return service-type information. (See your RADIUS Server documentation for more information.) The port server currently supports the following RADIUS service-types:

<table>
<thead>
<tr>
<th>RADIUS Service-Type</th>
<th>port server Session(s) Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Login-User&quot;</td>
<td>Telnet</td>
</tr>
<tr>
<td></td>
<td>Rlogin</td>
</tr>
<tr>
<td>&quot;Framed-User&quot;</td>
<td>PPP</td>
</tr>
<tr>
<td></td>
<td>SLIP</td>
</tr>
<tr>
<td>&quot;Administrative-User&quot;</td>
<td>port server Custom Configuration</td>
</tr>
</tbody>
</table>

The port server will interpret the service-type information returned by the authentication server and launch the appropriate session on the port. If the "Administrative-User" service is configured, the port server will launch the session specified by the custom configuration configured on that port. If no service-type is configured on the RADIUS server, the port server will refuse login and an authentication error will be noted in the syslog (see syslog page 243).

The port server sends the following subset of RADIUS authentication and accounting attributes to any given RADIUS server:

<table>
<thead>
<tr>
<th>RADIUS Attributes Supported</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>User-Name</td>
<td>Login-IP-Host</td>
</tr>
<tr>
<td>User-Password</td>
<td>Login-Service</td>
</tr>
<tr>
<td>NAS-IP-Address</td>
<td>Login-TCP-Port</td>
</tr>
<tr>
<td>NAS-Port-Type</td>
<td>Reply-Message</td>
</tr>
<tr>
<td>Service-Type</td>
<td>Acct-Status-Type</td>
</tr>
<tr>
<td>Framed-Protocol</td>
<td>Acct-Session-Time</td>
</tr>
<tr>
<td>Framed-IP-Address</td>
<td>Acct-Input-Octets</td>
</tr>
<tr>
<td>Framed-IP-Netmask</td>
<td>Acct-Output-Octets</td>
</tr>
<tr>
<td>Framed-MTU</td>
<td>Acct-Session-Id</td>
</tr>
<tr>
<td>Framed-Compression</td>
<td></td>
</tr>
</tbody>
</table>
Example

[admin:1/13]>>add radius
Enter the following information for the RADIUS server('?' for help):

Enter the type of RADIUS server to add.

1) RADIUS Authentication Server
2) RADIUS Accounting Server
Please select a number 1 - 2: 1

RADIUS Server Hostname : 199.171.30.212
Shared Secret : secret
UDP Port <default 1812> :
Retransmit Timeout (ms) <default 8000> :

The following RADIUS server entry will be added:

RADIUS Server Type : RADIUS Authentication Server
RADIUS Server Hostname : 199.171.30.212
Shared Secret : secret
UDP Port : 1812
Retransmit Timeout (ms) : 8000

Are these value correct? y

RADIUS server '199.171.30.212' has been added.

[admin:1/14]>>show radius

[RADIUS Authentication Servers]
RADIUS Server Shared Secret
------------- -------------
199.171.30.212 secret

[RADIUS Accounting Servers]
RADIUS Server Shared Secret
------------- -------------

[admin:1/15]>>delete radius
Enter the type of RADIUS server to delete.

1) RADIUS Authentication Server
2) RADIUS Accounting Server
Please select a number 1 - 2: 1

Enter the RADIUS server to delete: 199.171.30.212
The RADIUS server '199.171.30.212' has been deleted.

[admin:1/16]>>
ROUTING

set rip { on | off }
set advertise { on | off }
show rip

add route { destination host name | IP address } { gateway host name | IP address } { host | net } { jumps } { force }
add iroute { destination host name | IP address } { interface name } { host | net } { jumps } { force }

host | net: Host indicates that the destination IP address is a specific host; net indicates that the destination IP address is a machine on the network specified.
jumps: a number representing the number of jumps to reach the destination address.
force: a keyword that can be appended to the end of the command. When force is appended to the command line, you will not be prompted for confirmation if you are adding a route for an unavailable interface.

delete route { destination host name | IP address } { gateway host name | IP address } { host | net }
delete iroute { destination host name | IP address } { interface name } { host | net }

show route

The port server stores two types of routing information. The first type is stored in the Initial Routes table in the configuration database. The routes in this table are used to initialize the routes for each interface as it is configured into the system (e.g., the ethernet, PPP interfaces, Frame Relay, etc). Routes that exist in this table may or may not appear in the network route table, depending on the state of their respective interfaces. The Initial Routes table is configured using the `add/show/delete route/iroute` commands.

The second type of routing information is stored in the network route table, the actual routing table used by IP to route packets. Entries are added to the network route table through RIP, the configuration database, and by interfaces themselves.

You have two options for routing. You can use the Routing Information Protocol (RIP) on the port server to automatically route packets or you can define routing parameters manually. The routing commands take effect immediately; however, you must save the routing configuration you define for it to remain in effect following reboot (see the `save` command, page 232).

The rip option is enabled with the `set` command. You may enable RIP with or without the advertise feature. To enable RIP, enter:

    set rip on
If you want the advertise feature turned on, set advertise to on before you set rip on. If you set advertise on after you set rip on, the advertise feature will not be in effect until after you reboot. Turn on the advertise feature by entering:

```plaintext
set advertise on
```

To turn off the advertise feature, enter:

```plaintext
set advertise off
```

And to turn off RIP, enter:

```plaintext
set rip off
```

The `show` command, used with the keyword `rip`, shows the status of RIP on the port server. You will be informed whether RIP is enabled or disabled, and if enabled, whether or not you have set it to advertise. To display information about RIP, enter:

```plaintext
show rip
```

If you choose not to use RIP, you can manually define specific routing parameters. You can add and delete routes.

You can add routes using either the `add route` or the `add iroute` command. The `add route` command adds a route through a host name or IP address. The `add iroute` command adds a route through a specific interface and is useful when adding routes across local interfaces that share IP addresses. Both the commands update the Initial Routes tables stored in the configuration database and attempt to add the route to the current network route table.

You must enter either `host` or `net` in the command. `Host` indicates that the destination IP address is a specific host; `net` indicates that the destination IP address is a machine on the specified network. For example, to add a route:

```plaintext
add route 198.130.71.0 199.171.148.1 net 1
```

or

```plaintext
add iroute 198.130.71.0 ppp_wan net 1
```

To delete a route, you must enter both the destination and the gateway, specifying them as either a host name or an IP address. Or, using the `delete iroute` command, you can specify the gateway as an interface name. In addition, you must define whether the destination host is an actual host or a network by appending host or net to the command. For example:

```plaintext
delete route 198.130.71.0 199.171.148.1 net 1
```

or

```plaintext
delete iroute 198.130.71.0 ppp_wan net
```

To display a listing of the Initial Routes table stored in the database, enter:

```plaintext
show route
```

To display a listing of the currently active network route table, enter:

```plaintext
netstat -r
```
**RTELNET**

```plaintext
set rtelnet { port number(s) }
delete rtelnet { port number(s) }
show rtelnet
```

**port numbers**: Any valid port number for your port server model, including the parallel port (port 17 or 32, depending on the port server model). May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range.

With the `set rtelnet` command, you can configure specific ports to accept connections from the network to a TCP port number; for example, from an rtelnet process running on a host. This option would be used to support dial-out modems or printers.

To set ports 1 and 2 and 4 through 7 to accept rtelnet connections, enter:

```
set rtelnet 1,2,4-7
```

This configures the port server to accept telnet connections on TCP ports 9001, 9002, 9004-9007 and to accept raw TCP mode connections on TCP ports 8001, 8002, 8004-8007.

You can delete the rtelnet sessions running on ports 4 through 7 by entering:

```
delete rtelnet 4-7
```

The rtelnet sessions running on ports 1 and 2 will continue.

To determine on which ports you have enabled an rtelnet process with the `set rtelnet` command (versus using a custom service configuration—see Service Configurations, page 78), enter:

```
show rtelnet
```

**TCP port numbers**

For each physical port configured for rtelnet, the port server sets up two logical TCP port numbers. Logical port 90nn (where nn is the serial port configured - 00-15 or 32) is for standard reverse telnet connections. Logical port 80nn is for raw reverse telnet connections.

Similarly, TCP ports are set up for modem pools (mpool) and printer pools (ppool). Refer to the following table for a summary of the logical port numbers.
### Raw reverse telnet

Raw reverse telnet allows you to send and receive data through port server ports without using special reverse telnet programs, pseudo-ttys or port redirector programs.

You use raw reverse telnet by opening a Transmission Control Protocol (TCP) session and specifying a port number on the port server. Data is sent and received unchanged between the port and the TCP endpoint because there is no intervening protocol or processing.

Refer to Appendix A for further description of host-based applications to access these TCP connections.
SIZE

set size { port number(s) } { 5 | 6 | 7 | 8 } [ i ]

show size { port number(s) | all }

port numbers: Any valid port number for your port server model. May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range.

Character size options for the port server are 5, 6, 7, and 8. To set a character size of 8 for ports 2, 4, and 7, enter:

    set size 2,4,7 8

    NOTE: Append the i option to the command if you want the port configuration to take affect immediately. For example:

    set size 2,4,7 8 i

Save the configuration (using the save command) if you want the configuration to be maintained following reboot.

To determine the character size setting of each port server serial port, enter:

    show size all

The character size setting of each port will be listed. To display size information for specific ports, append the port numbers to the show size command.
**SLIP**

`set slip { Cn } { port number }`

`show { Cn }`

The `set slip` command defines that the custom service configuration (C0-C33) specified in the command be configured as SLIP and that the SLIP configuration be enabled on the port specified. SLIP can be enabled on any one of the serial ports (0 through 15).

A SLIP service configuration can be enabled on only one port at a time. To run multiple SLIP connections, you must create a different SLIP configuration—with different IP addresses—for each connection.

A SLIP configuration allows dial-in auto-SLIP or dial-out SLIP capability on the port associated with that particular service configuration. SLIP cannot be combined with other custom service configuration options (such as multiple user sessions, 1-key menus, and capabilities access).

To define service configuration C13 as SLIP and enable it on port 3, enter:

```
set slip C13 3
```

After entering the `set slip` command, you will be prompted for the following information needed to configure the service correctly:

- Remote IP address
- Local IP address

Your SLIP configuration settings will be displayed and you will be asked to confirm the values. Respond `y` (yes) to confirm the values, and `n` (no) to abort the command. A "yes" response enables the SLIP configuration you defined as service configuration C13 on the port specified in the command.

Use the `show` command to display information about a specific SLIP configuration or all the service configurations. For example, to view the configuration of the SLIP service C13, enter:

```
show C13
```

To display information about all service configurations, enter:

```
show C*
```

Both of the above commands will display a summary describing the configuration(s) specified in the command and list the port(s) on which the configuration is enabled.

**NOTE:** For a SLIP service configuration to take effect, you must save it (see the `save` command, page 232), then reboot your port server. All other service configurations take effect as soon as they are enabled on a port or ports.
As with other custom configurations, use the `delete` command to delete a SLIP service configuration. For example, to delete the SLIP service configuration that was configured as C13 and enabled on port 3, enter:

```
delete C13 3
```
The Simple Network Management Protocol (SNMP) provided with the port server monitors and controls TCP/IP-based networks. By allowing you to retrieve and alter networking information maintained by hosts and routers attached to a network, SNMP provides the ability to diagnose and correct network problems from remote hosts.

SNMP can be run without customization. However, it will be a more effective tool for your network if you customize it to your needs. To turn the SNMP agent on, enter:

```
set snmp on
```

To turn off SNMP, enter:

```
set snmp off
```

The port server SNMP agent can send SNMP "trap" messages to one or more SNMP Network Management Stations (NMS) when various events occur. To allow the port server to send SNMP traps, use the `traps` option when you turn SNMP on:

```
set snmp on traps
```

You can run the SNMP agent without sending traps with the command:

```
set snmp on notraps
```

When traps are enabled, the port server will transmit a trap message to the designated NMS whenever the SNMP agent starts. To have the port server send a trap message to your NMS each time it boots up, save a configuration with SNMP on and traps enabled:

```
set snmp on traps
save
```

Entering the `set snmp on` or `set snmp off` commands without specifying `traps` or `notraps` options will leave the current trap setting unchanged.

Use the `add trap` command (for details, see TRAP page 168) to set up one or more Network Management Stations to which the port server will send trap messages.

The `show snmp` command displays the current status of the port server SNMP agent. For example, after executing the `set` command shown above, the show command would display:
show snmp
SNMP is ON; traps enabled

SNMP defines three descriptive strings that an NMS may read from SNMP agents. The strings are:

<table>
<thead>
<tr>
<th>Descriptive String</th>
<th>SNMP variable name</th>
</tr>
</thead>
<tbody>
<tr>
<td>The “system description”</td>
<td>sysDescr.0</td>
</tr>
<tr>
<td>The “contact”</td>
<td>sysContact.0</td>
</tr>
<tr>
<td>The “location”</td>
<td>sysLocation.0</td>
</tr>
</tbody>
</table>

All three hold printable strings that an NMS can fetch and display to help network managers identify individual devices on the network.

By default, the system description string gives the port server model number and software version information. You can examine the system description string with the `show description` command (of course, if the SNMP agent is on, you can retrieve the same string from an NMS). For example:

```
show description
SNMP Description: RCS/41x6 FW_05A port server_4000_BDNL_05A
```

The description above gives the model number (e.g. RCS/ 4116), the firmware revision level (FW_05A) and the software identification and revision level (e.g. RCS_4000_BDNL, Version 05A).

We recommend that you leave the factory default description string in place, since it can supply important diagnostic information about your port server. If you wish, however, you can supply your own system description string with the `set description` command:

```
set description "Comm server number 1234"
```

You can restore the factory default description by setting an empty string:

```
set description ""
```

The "contact" and "location" strings are both empty by default. You can place any useful information you want to in these strings (up to 255 characters each). For example:

```
set contact "Jane Smith, extension 555"
set location "Third floor equipment room"
```

There are corresponding `show` commands for both "contact" and "location".

Use the `add community` command (for details, see [COMMUNITY] page101) to add SNMP community profiles to the port server. In order for a management station to perform SNMP operations on an port server, a proper community profile must exist which defines SNMP access permissions. After a community is created, its permissions can be customized using the `add view` command (see [VIEW] page170).
SPEED

set speed { port number(s) } { baud rate value } [ i ]

show speed { port number(s) | all }

- port numbers: Any valid port number for your port server model. May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range.

- bit rate value: The desired bit rate speed.

Set the bit rate speed of a port with the set speed command. To set the port speed of serial ports 6 and 7 to 19.2K bits per second, enter:

```
set speed 6,7  19200
```

The port server will accept the bit rate formatted only as shown in the command example. Bit rate values that can be specified for the port server are:

<table>
<thead>
<tr>
<th>Value</th>
<th>Value</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>600</td>
<td>19200</td>
</tr>
<tr>
<td>75</td>
<td>1200</td>
<td>28800</td>
</tr>
<tr>
<td>110</td>
<td>1800</td>
<td>38400</td>
</tr>
<tr>
<td>134</td>
<td>2400</td>
<td>57600</td>
</tr>
<tr>
<td>150</td>
<td>4800</td>
<td>115200</td>
</tr>
<tr>
<td>200</td>
<td>9600</td>
<td></td>
</tr>
<tr>
<td>300</td>
<td>14400</td>
<td></td>
</tr>
</tbody>
</table>

 NOTE: Append the i option to the command if you want the port configuration to take effect immediately. For example:

```
set speed 6,7  19200  i
```

Save the configuration (using the save command) if you want the configuration to be maintained following reboot.

To determine the bit rate speed setting of each port server serial port, enter:

```
show speed all
```

The speed of each port will be listed. To display bit rate information for specific ports, append the port numbers to the show speed command.
STOP

set stop \{ port number(s) \} \{ 1 | 2 \} [ i ]

show stop \{ port number(s) | all \}

port numbers: Any valid port number for your port server model. May be entered as individual numbers, as a range of port numbers, as “all,” or as a combination of individual numbers and a range.

Stop bit options are 1 and 2. To set 1 stop bit for all the serial ports, enter:

    set stop all 1

    NOTE: Append the i option to the command if you want the port configuration to take effect immediately. For example:

    set stop all 1 i

Save the configuration (using the save command) if you want the configuration to be maintained following reboot.

To determine the stop bits setting of each port server serial port, enter:

    show stop all

The stop bits setting of each port will be listed. To display stop bits information for specific ports, append the port numbers to the command.
SWITCH

**set switch** {port number(s)} {char | <undef>} [i]

**show switch** {port number(s)}

**port numbers:** Any valid port number for your port server model. May be entered as individual numbers, as a range of port numbers, as “all”, or as a combination of individual numbers and a range.

**char:** decimal representation of the key that causes a switch between multiple sessions on a port. This may be any ASCII character from 1-255 (decimal). Normally you would specify a control character such as CTRL-\ (28). To turn off switch character recognition (default), set the switch character to 0.

If you have multiple sessions defined on a port, the BREAK key is normally used to switch between sessions. If you define a switch character, this character will also switch between sessions. If you telnet to a host from one or more of your multiple sessions AND you have a switch character defined, then the BREAK key is no longer treated as a switch key but is translated into a telnet IAC BRK sequence to the remote host.

If you do not have multiple sessions defined on a port (using a custom configuration), this setting has no effect.

To set the switch character to ASCII 28 (CTRL-\ ) for all the serial ports, enter:

```
set switch all 28
```

**NOTE:** Append the i option to the command if you want the port configuration to take effect immediately. For example:

```
set switch all 28 i
```

Save the configuration (see the `save` command, page 232) if you want the configuration to be maintained following reboot.

To determine the switch character setting for each port server serial port, enter:

```
show switch all
```
SYSTEM

add system
show system
delete system

The system commands are used to configure the remote system for dial on demand.

To define a remote system and its associated port(s), telephone number, and login script, enter:

    add system zeus

The system name can be any alphanumeric string up to 15 characters in length.

You will be prompted to enter the following:

    Ports: port numbers used to reach the remote site.
    Phone number: up to 32 characters, may include hyphens (no spaces).
    Backoff Time: The backoff time, in seconds, to delay prior to retrying a failed
dialout attempt. The backoff time should be 0 unless you are setting up a bi-
directional dial-on-demand configuration.

    Login Chat Script: up to 256 characters, the login script describes what negotiating
needs to take place to log into the remote site (see Writing Chat Scripts page 277).

Example

    [admin:1/12]>>add system test1
    Enter the ports to communicate through : 0-6
    Enter the phone number to dial : 222-5555
    Enter the login Chat script: Password: user >> "ppp 191.150.100.12:199:170.31.253"
    System Name : test1
    Ports : 0-6
    Phone Number : 222-5555
    Enter backoff time (sec) (default 0) : 0
    Login Chat Script: Password: user >>"ppp 191.150.100.12:199:170.31.253"
    Are these values correct (y/n)? y
    System test1 added
    [admin:1/13]>>

To list all details of the specified system:

    show system {system name}

Example 1

    [admin:1/13]>>show system test1

    System Name : test1
    Ports : port0 port1 port2 port3 port4 port5 port6
    Phone Number : 222-5555
    Backoff Time (sec) : 0
    Login Chat Script : Password: user >>
"ppp 191.150.100.12:199:170.31.253"
[admin:1/14]>>

Example 2

Omitting the system name displays a complete listing of the configured systems. Only the phone numbers are listed:

    System Name: test0  Phone Number: 222-1995
    System Name: test1  Phone Number: 222-5555

[admin:1/15]>>

To delete a system definition, enter:

delte system test1
TCP_SERVICE

set tcp_service {TCP port} { "command line" }
delete tcp_service { TCP port }
show tcp_service

TCP port: Any valid TCP port number. Numbers of 15000 or above are recommended.

The set tcp_service command is designed for use with the developer's kit for developers writing their own onboard applications.

The set tcp_service command registers a TCP listener for the specified port. Using a TCP port number of 15000 or higher is recommended, to avoid conflict with any other services. A warning will be displayed if you select a port lower than 15000.

You must specify a command to be run when an incoming TCP connection request is detected on the specified port. The command is formatted as a normal shell command, defined by the developer of the application.

For example, to set a TCP listener on port 16000 to execute an application called my_app on connection, type:

    set tcp_service 16000 "my_app"

To remove the TCP listener, type:

    delete tcp_service 16000

To see the TCP services currently in place, type:

    show tcp_service
show term

The show term command displays a list of the currently supported terminal types. The list shows the name(s) of the terminal type (e.g., xterm, vt100, wy50) and a description of the terminal type supported. Use the command to find the name to enter when specifying your terminal type in the setenv command, the term command (see term page 254), or when defining a custom configuration.
TIMEOUT

set timeout {port numbers} {secs} [ I ]

show timeout [port numbers]

port numbers: Any valid port number for your port server model. May be entered as individual numbers, as a range of port numbers, as “all”, or as a combination of individual numbers and a range.

Use the timeout command for setting and showing timeouts.

Sets the inactivity time-out on a port or ports. If no activity takes place during the specified time-out period, the specified port(s) will close the connection. Ports are specified by port number. Time is specified in seconds.

To set a 60-second time-out for ports 1 and 5:

    set timeout 1,5 60

To show the active timeout settings for port 5:

    show timeout 5
TRAP

**add trap** [ **host** ] [ **community** ] [ **port** ]

**delete trap** { **host** | all }

**show traps**

Use the **add trap** command to add a Network Management Station (NMS) to the SNMP trap list. This is the list of NMS’s to which the port server will direct any SNMP trap messages it generates. When traps are enabled, the port server will transmit a trap message to the designated NMS whenever the SNMP agent starts.

You may include all necessary parameters with the **add trap** command or you may allow the port server to prompt you for any missing parameters. For help, type **?** at any prompt.

You may enter the **add trap** command in any of these forms:

<table>
<thead>
<tr>
<th>Command Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>add trap</strong></td>
<td>The port server prompts you for all parameters.</td>
</tr>
<tr>
<td><strong>add trap</strong> { <strong>host</strong> }</td>
<td>The port server prompts you for the community name string and the optional destination port number.</td>
</tr>
<tr>
<td><strong>add trap</strong> { <strong>host</strong> } { <strong>community</strong> }</td>
<td>The port server prompts for the optional destination port number.</td>
</tr>
<tr>
<td><strong>add trap</strong> { <strong>host</strong> } { <strong>community</strong> } { <strong>port</strong> }</td>
<td>The port server uses the parameters supplied and does not prompt for any additional entry.</td>
</tr>
</tbody>
</table>

The parameters are:

**host**  
Enter the name or IP address of a host to which the port server should send traps. You may enter the name of a host that is present in the host name list (see **HOST NAME LIST**) or a name that is resolvable through the Domain Name Service (DNS). If you are not using the host name list nor DNS, you may enter the IP address in "dotted decimal" form (e.g. 192.123.123.123).

**community**  
The port server includes a community name in each trap message it sends. The destination network management station (NMS) may use the community name to authenticate trap messages. The community name string may be up to 255 characters long. The community name must match a community name configured on the NMS to which the port server will send traps.

**port**  
By default, traps are sent to UDP port 162 on the destination host. The port server will direct trap messages to the default port unless you specify another port number. To direct trap messages to a non-standard port, enter a decimal number between 1 and 65535. You may also specify the default port with the word default. For example, to enter a complete **add trap** command at the shell prompt and avoid all interactive prompts from
the port server, type [admin:1/1]>>add trap nms-1 public default

If you want the port server to send trap messages to more than one NMS, use a separate add trap command to add each NMS to the trap list.

Use the delete trap command to remove an NMS from the SNMP trap list. You may specify the host name of the NMS or the port server will prompt you for the name. You may use all to completely clear the trap list (to delete all entries).

delete trap { host | all }

The show traps command displays the current contents of the SNMP trap list. For example, on an port server with several trap hosts entered, the display might look like this:

[admin:1/1]>>show traps
SNMP trap list
Destination host    Community name    UDP Port
nms-1               public           162
myhost              public           162
nonstandard         xyzzy           10000
192.111.222.111     public           162
**VIEW**

**add view** [community [mib_oid] {none | readonly | readwrite}]

**delete view** [community]

Use the **add view** command to add MIB object IDs to an existing SNMP community's view definition. The view commands are used to define which portions of the MIB are visible to a given community.

The **add view** command has the following forms:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>add view</strong></td>
<td>The port server prompts you for all necessary parameters.</td>
</tr>
<tr>
<td><strong>add view {community}</strong></td>
<td>The port server prompts you for MIB object and access information. You can add multiple MIB objects to the specified community view.</td>
</tr>
<tr>
<td>**add view {community}{mib_oid} {none</td>
<td>readonly</td>
</tr>
</tbody>
</table>

The MIB objects which exist in a view can correspond to either leaves (i.e. MIB variables) or subtree identifiers. If a subtree identifier is specified, then all objects underneath that subtree inherit the specified access permissions. If additional subtree objects are added which exist underneath an already defined subtree node, then a MIB variable's access is determined by the access of the closest parent.

When the port server displays a list of view objects, it sorts the list, placing the most specific object identifiers (i.e. those closest to the MIB leaves/variables) first.

**Example**

Suppose the following community is defined on the port server:

<table>
<thead>
<tr>
<th>Community Name: system_community</th>
<th>View Subtree</th>
<th>Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>sysContact</td>
<td>readwrite</td>
<td></td>
</tr>
<tr>
<td>system</td>
<td>readonly</td>
<td></td>
</tr>
<tr>
<td>mib</td>
<td>none</td>
<td></td>
</tr>
</tbody>
</table>

Since "mib" has its access set to "none", the entire Internet MIB has its access permissions disabled. However, the entry for the system object (which is a descendant of "mib" in the MIB management tree) overrides the "mib" setting and allows read-only access to the entire system subtree. Furthermore, the "sysContact" object (which is a child of "system") allows read-write access to this specific variable.

**NOTE:** Since "sysContact" is a child of "system", it appears before "system" in the view list. Likewise, "mib" appears at the bottom of the list, since it is a parent to all Internet MIB objects.
Use the **delete view** command to remove existing view definitions from an SNMP community profile. If you do not specify a community name, you will be prompted for one. The command displays a list of MIB objects which currently belong to the specified community view. You can then select the objects to remove.

**Example**

```bash
[admin:1/15]>> delete view system_community
Choose a MIB subtree OID to remove from this community's view (Choose '0' when finished):
  1 - sysContact             readwrite
  2 - system                 readonly
  3 - mib                     none
Please select a number 0 - 3: 1
Choose a MIB subtree OID to remove from this community's view (Choose '0' when finished):
  1 - system                 readonly
  2 - mib                     none
Please select a number 0 - 2: 1
Choose a MIB subtree OID to remove from this community's view (Choose '0' when finished):
  1 - mib                     none
Please select a number 0 - 1: 0
'system_community' has been updated.
[admin:1/16]>>
```

To display a community's view, use the **show community** command (see **COMMUNITY** page 101).
VERSION

show version

Use the show version command to show the current version of the onboard software and firmware.
WAN

(NOTE: only some port server models have a WAN port. See the Hardware Manual for your port server to determine if it has a WAN port.)

set wan { interface } { rs232 | rs422 | e530 | v.35 }

set wan { clock } { external | internal } [ baud rate ]

show wan

The set wan command allows you to specify the physical WAN interface and the type of clock to be used for the WAN (synchronous). The setting takes effect following reboot, so save your configuration (using the save command) after configuring the WAN port.

You can specify the interface as RS-232, RS-422, EIA-530, or v.35. For example, to set the WAN interface as RS-422, enter:

    set wan interface rs422

You can specify the clock as either external or internal. For example, to set the wan clock to external, enter:

    set wan clock external

To display the WAN port interface and clock settings, enter:

    show wan
Chapter 6: Operational Commands

Operational Command Summary

This chapter describes the port server operational commands. Table 6-1 summarizes the commands. Detailed descriptions, in alphabetical order, follow the summary table.

Table 6-1. Operational Commands Summary

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>TYPE OF COMMAND</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>add</td>
<td>configuration</td>
<td>See Chapter 5.</td>
</tr>
<tr>
<td>admin</td>
<td>administrative</td>
<td>Enables commands at administrative level.</td>
</tr>
<tr>
<td>apmem</td>
<td>diagnostic</td>
<td>Displays memory usage for a specific process.</td>
</tr>
<tr>
<td>arp</td>
<td>administrative</td>
<td>Displays and modifies Internet-to-Ethernet address translation table.</td>
</tr>
<tr>
<td>boot</td>
<td>administrative</td>
<td>Sets firmware options.</td>
</tr>
<tr>
<td>delete</td>
<td>configuration</td>
<td>See Chapter 5.</td>
</tr>
<tr>
<td>drop</td>
<td>administrative</td>
<td>Terminates all sessions on selected port.</td>
</tr>
<tr>
<td>dump</td>
<td>diagnostic</td>
<td>Dumps a copy of RAM to a host system for debug.</td>
</tr>
<tr>
<td>exit</td>
<td>administrative</td>
<td>Exits from administrative or expert capabilities access level to previous level.</td>
</tr>
<tr>
<td>expert</td>
<td>administrative</td>
<td>Enables commands available at the expert level.</td>
</tr>
<tr>
<td>flash_compac</td>
<td>administrative</td>
<td>Reclaims space from flash memory.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>TYPE OF COMMAND</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------</td>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td><code>flash_log</code></td>
<td>administrative</td>
<td>Displays the message log that is recorded in flash memory.</td>
</tr>
<tr>
<td><code>flash_save</code></td>
<td>administrative</td>
<td>Uploads the current configuration to a host system.</td>
</tr>
<tr>
<td><code>flash_update</code></td>
<td>administrative</td>
<td>Downloads a configuration from a host system.</td>
</tr>
<tr>
<td><code>frstats</code></td>
<td>administrative</td>
<td>Displays the status of the frame relay link.</td>
</tr>
<tr>
<td><code>hangup</code></td>
<td>operational</td>
<td>Generates hangup and terminates all sessions on the line.</td>
</tr>
<tr>
<td><code>help</code> or <code>?</code></td>
<td>operational/administrative</td>
<td>Displays a list of available commands.</td>
</tr>
<tr>
<td><code>http_stats</code></td>
<td>diagnostic</td>
<td>Displays statistics concerning the operation of the port server HTTP server.</td>
</tr>
<tr>
<td><code>ifconfig</code></td>
<td>administrative</td>
<td>Assigns an address to a network interface.</td>
</tr>
<tr>
<td><code>kill</code></td>
<td>administrative</td>
<td>Sends a signal to an identified process.</td>
</tr>
<tr>
<td><code>lookmem</code></td>
<td>diagnostic</td>
<td>Shows memory statistics for troubleshooting.</td>
</tr>
<tr>
<td><code>lookstream</code></td>
<td>diagnostic</td>
<td>Shows STREAMS statistics for troubleshooting.</td>
</tr>
<tr>
<td><code>lpq</code></td>
<td>administrative</td>
<td>Displays print queue.</td>
</tr>
<tr>
<td><code>lprm</code></td>
<td>administrative</td>
<td>Deletes a job or jobs from the print queue.</td>
</tr>
<tr>
<td><code>mpool</code></td>
<td>operational</td>
<td>Selects a modem from a shared pool of modems.</td>
</tr>
<tr>
<td><code>netstat</code></td>
<td>administrative</td>
<td>Displays contents of network-related data structures.</td>
</tr>
<tr>
<td><code>ping</code></td>
<td>administrative/diagnostic</td>
<td>Elicits an ICMP ECHO_RESPONSE from a host or gateway.</td>
</tr>
<tr>
<td><code>port_stat</code></td>
<td>administrative/diagnostic</td>
<td>Produces a constantly updated screen of status information about ports.</td>
</tr>
<tr>
<td><code>power</code></td>
<td>administrative</td>
<td>Applies power to specified ports(s).</td>
</tr>
<tr>
<td>COMMAND</td>
<td>TYPE OF COMMAND</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>---------</td>
<td>----------------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>ppp</strong></td>
<td>operational</td>
<td>Attaches serial lines as network interfaces, using point-to-point protocol.</td>
</tr>
<tr>
<td><strong>pppstats</strong></td>
<td>administrative/diagnostic</td>
<td>Retrieves and displays ppp interface statistics.</td>
</tr>
<tr>
<td><strong>printenv</strong></td>
<td>administrative</td>
<td>Displays all currently set environment variables.</td>
</tr>
<tr>
<td><strong>processes</strong></td>
<td>administrative/diagnostic</td>
<td>Displays process information or system statistics.</td>
</tr>
<tr>
<td><strong>prtr_test</strong></td>
<td>administrative/diagnostic</td>
<td>Sends output data to a port to test the interface.</td>
</tr>
<tr>
<td><strong>quit</strong></td>
<td>administrative</td>
<td>Takes you from expert to administrative level.</td>
</tr>
<tr>
<td><strong>reboot</strong></td>
<td>administrative</td>
<td>Reboots the port server.</td>
</tr>
<tr>
<td><strong>rlogin</strong></td>
<td>operational</td>
<td>Connects your terminal on the current local host system to the remote host system and establishes a remote login session.</td>
</tr>
<tr>
<td><strong>save</strong></td>
<td>administrative</td>
<td>Saves the port server configuration to flash RAM.</td>
</tr>
<tr>
<td><strong>set</strong></td>
<td>configuration</td>
<td>See Chapter 5 of this manual.</td>
</tr>
<tr>
<td><strong>setenv</strong></td>
<td>administrative</td>
<td>Displays or sets environment variables.</td>
</tr>
<tr>
<td><strong>show</strong></td>
<td>configuration</td>
<td>See Chapter 5 of this manual.</td>
</tr>
<tr>
<td><strong>slip</strong></td>
<td>operational</td>
<td>Uses Serial Line Interface Protocol to attach serial lines as network interfaces.</td>
</tr>
<tr>
<td><strong>stty</strong></td>
<td>operational</td>
<td>Sets or reports on terminal characteristics for the device that is its standard input.</td>
</tr>
<tr>
<td><strong>syslog</strong></td>
<td>administrative/diagnostic</td>
<td>Displays or manages the system log.</td>
</tr>
<tr>
<td><strong>telnet</strong></td>
<td>operational</td>
<td>Uses TELNET protocol to communicate with another host.</td>
</tr>
<tr>
<td><strong>term</strong></td>
<td>operational</td>
<td>Displays current terminal type.</td>
</tr>
<tr>
<td>COMMAND</td>
<td>TYPE OF COMMAND</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>-------------</td>
<td>---------------------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>trace clear</td>
<td>diagnostic</td>
<td>Clears trace buffer.</td>
</tr>
<tr>
<td>trace display</td>
<td>diagnostic</td>
<td>Displays trace buffer.</td>
</tr>
<tr>
<td>trace off</td>
<td>diagnostic</td>
<td>Turns off software tracing.</td>
</tr>
<tr>
<td>trace on</td>
<td>diagnostic</td>
<td>Turns on software tracing.</td>
</tr>
<tr>
<td>tty</td>
<td>administrative/diagnostic</td>
<td>Shows port on which user is running.</td>
</tr>
<tr>
<td>unsetenv</td>
<td>administrative</td>
<td>Unsets an environment variable.</td>
</tr>
<tr>
<td>upgrade</td>
<td>administrative</td>
<td>Enables licensed software options.</td>
</tr>
<tr>
<td>who</td>
<td>administrative</td>
<td>Shows all current user activity.</td>
</tr>
</tbody>
</table>
admin

Syntax

    admin

Description

The admin command is used to gain access to commands that require administrative mode. It will prompt the user for the admin password. Any previously set environment variables remain unchanged.
apmem

Syntax

    apmem \{process_id\}

Description

Displays the amount of heap memory currently allocated to the specified process. (Use the processes command to get the names of the current processes along with their process ids.)

This command is designed for use by development and support personnel for diagnostic purposes.

Example

    [expert:1/18]>>apmem 44
    Total SWKAI malloc‘ed for pid (44): 8600
arp

Syntax

arp {hostname}
arp {-a}
arp {-d} {hostname}
arp {-s} {hostname} {ether_addr} [temp] [pub] [trail]

Description

The arp program displays and modifies the Internet-to-Ethernet address translation table, which is maintained by the address resolution protocol. When hostname is the only argument, arp displays the current ARP entry for the specified host name. The host may be specified by name or by number using Internet dot notation.

Flags

<table>
<thead>
<tr>
<th>Flag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-a</td>
<td>Displays all current ARP entries.</td>
</tr>
<tr>
<td>-d</td>
<td>Deletes an ARP entry for the specified host.</td>
</tr>
<tr>
<td>-s</td>
<td>Creates ARP entry for specified host, using the specified Ethernet address.</td>
</tr>
<tr>
<td>ether_addr</td>
<td>Specifies the Ethernet address for the new ARP entry. The Ethernet address must be six colon-separated, two-digit hexadecimal numbers. The entry will be permanent unless the temp option is specified.</td>
</tr>
<tr>
<td>hostname</td>
<td>specifies the host. When no other flags are used, arp displays the current ARP entry for this host.</td>
</tr>
<tr>
<td>pub</td>
<td>&quot;Publishes&quot; the new entry; that is, this system will act as an ARP server, responding to requests for hostname even though the host address is not an address of the local host.</td>
</tr>
<tr>
<td>temp</td>
<td>Makes the specified Ethernet address temporary.</td>
</tr>
<tr>
<td>trail</td>
<td>Specifies that trailer encapsulations should be used with this host. Trailers are a link-dependent issue. Currently, no known LLI-compliant Ethernet driver supports trailers; it is unwise to advertise them unless it is certain that the link layer can handle them.</td>
</tr>
</tbody>
</table>

Example

[admin:1/30]>>arp -a
oz.abc.company.com (123.45.67.89) at 8:0:20:1f:34:8a
ra.xyz.company.com (189.67.45.23) at 8:0:20:11:d6:27
boot

Syntax

    boot

Description

The boot command is an expert level command used to set firmware options. The settings that are controlled by this command are:

- whether or not the port server will attempt to obtain an IP address from bootp/ dhcp/ rarp when it boots (factory default is yes)
- whether or not the port server will initiate an automatic memory dump when an error occur (factory default is no)
- whether or not the port server will execute an automatic reload of the OS at reset (factory default is yes)

When the command is entered,

    boot

a list of the current settings is displayed. For example:

    [expert:1/7]>>boot
    a. bootp/dhcp/rarp is enabled
    b. automatic memory dump on error to file file_bug.dump
    c. dump host is 199.171.31.12
    d. IP address for this unit comes from bootp/dhcp
    e. OS is automatically loaded after reset
    q. exit.

To change a setting, enter a letter (a, b, c, d, or e).

a

displays a list of options for changing the bootp/ dhcp/ rarp setting.

- 0 - enabled
- 1 - disabled with auto re-enable
- 2 - disabled

To select an option, enter a number (0, 1, or 2).

0  sets the option to enabled. When the port server is rebooted, it will attempt to get an IP address from bootp/ dhcp/ rarp.

1  sets the option to disabled with automatic re-enable. When the port server/ 400 is rebooted, it will not get an IP address from bootp/ dhcp/ rarp. However, when the reboot is completed, if no IP address is available to the port server, it will automatically change the setting to enable and reboot to attempt to obtain an IP address. When the set ip command is used to assign an IP address, it automatically sets this option.
sets the option to disabled.

b
toggles the automatic memory dump between enabled and disabled. When the setting is toggled from disabled to enabled, you are prompted to enter a file name to which the memory dump will be written. The IP address of the host on which the file will be written is set using option c. The IP address to be used by the port server when writing the dump is set using option d. When the automatic memory dump is disabled, the boot command does not display options c and d.

[expert:1/7]>>boot
  a. bootp/dhcp/rarp is enabled
  b. automatic memory dump on error is disabled
  c. OS is automatically loaded after reset
  q. exit.

c
sets the IP address of the host on which the dump file will be written. You will be prompted:

  tftp server for dump (<cr> to use bootp/dhcp info):

Type an IP address or press <enter>. If you press <enter> without typing an IP address, the port server will use the IP address that was set as the TFTP server option on the BOOTP or DHCP server.

d
sets the IP address used by the port server when writing the dump file. You will be prompted:

  ip address to use for dump (<cr> to use bootp/dhcp):

Type an IP address or press <enter>. If you press <enter> without typing an IP address, the port server will obtain an IP address from the BOOTP or DHCP server.

e
toggles the automatic reload of the OS after reset between enabled and disabled. When this setting is disabled, a firmware menu is displayed on the lowest port number (0 or 1) after reset.

To exit from the boot command menu, type q.
drop

Syntax

drop {port}

Description

The drop command terminates all sessions on the selected port. The port is returned to the login prompt.
dump

Syntax

dump [filename [server [starting_address ending_address]]]

Description

The dump command is a diagnostic command that will send a memory dump to the specified server.

No arguments implies an interactive mode. The default starting_address is the beginning of RAM. The default ending_address is the end of RAM.

The server parameter defaults to the system that responded to the bootp or dhcp request when the port server was last reset. This server must be providing tftp service.

Some servers enforce security measures on tftp transfers to the server. For example, the Solaris version of the tftp daemon has a -s option that prevents:

- access to any directory except /tftpboot
- the creation of any files
- the overwriting of any file that does not have world write permission

Given these restrictions it is necessary to create an empty file in the /tftpboot directory with world write access prior to attempting to save a file to that system. This is done as follows on the server:

```
rm /tftpboot/myfile.bdnl
touch /tftpboot/myfile.bdnl
chmod 0777 /tftpboot/myfile.bdnl
```
exit

Syntax

exit

Description
The exit command exits from the administrative or expert capabilities access level back to the previous shell. For example, if exit is invoked from expert mode, the user would be returned to admin mode; from admin, an exit would return the user to to_admin mode.
expert

Syntax

   expert

Description

The expert command allows you to use the following commands at the expert level:

?     flash_log     pppstats     syslog
add    hangup      printenv     telnet
admin  help        processes    term
apmem  ifconfig    prtr_test    trace_clear
arp    kill        quit         trace_display
boot   lookmem     reboot       trace_off
delete lookstream  rlogin       trace_on
drop   lpq         save         tty
dump   lprm        set          unsetenv
exit   mpool       setenv       upgrade
expert netstat     show         who
flash_save ping      slip
flash_update ppool    stty
flash_compact ppp
flash_compact

Syntax

    flash_compact [-e]

Description

The flash_compact command is an expert level command used to reclaim space from the area of flash memory in which product information and error records are stored. The command removes invalidated and, optionally, error records from flash memory. Use this command when an error message is displayed during boot indicating that flash memory is full.

Flags

    -e

    Causes both error records and invalidated records to be removed. If the -e flag is not used, only invalidated records are removed.
**flash_log**

**Syntax**

`flash_log`

**Description**

The `flash_log` command is an expert level command that displays the message log that is recorded in the flash memory of the port server. This message log is intended to help Technical Support in diagnosing problems.

Typically, the log will contain messages like the following:

```shell
[expert:1/2]>>flash_log
device header record
configuration record
mac address record
feature set record
log message: port server_4000_BDNL Version 05A built Dec 12 1996
18:02:00
```

Multiple log message records may be present and indicate the various versions of the onboard software that have been loaded into the port server.

In the event that an unrecoverable error is detected, a message such as the following may appear in the flash log:

```shell
error category: 0x97 number 0x0
081ed8a0 27000813 0802ee82 0812f568 00000000 00000003 00000002
0811c190 081065ef 081ed8d4 0802f8ea 081303b8 080160aa 081ed8e4 0802f90e
0811c190 00000003 081ed900 08026e9c 0811c190 081304d4 0813f2a4 0811c1b4
0811e95c 081ed95c 08024b08 0813f2a4 081ed954 081ed950 00000003 00000005
```

Such errors are generally recorded prior to the port server performing an automatic restart. This information should be forwarded to Technical Support along with the set of actions that lead to the error occurring so that Systech can diagnose and fix the problem.
flash_save

Syntax

    flash_save [[-section] filename [server]]

Description

Use the flash_save command to save port server configuration or software to a server system. The format of the file created on the server is compatible with the flash_update command so that data saved from one system can be loaded onto another. The command allows any of the port server flash memory sections to be saved. If no section is specified, the configuration database section is saved.

The “product_data” or “product/manufacturing data” section should not be copied from one system to another. This area contains the MAC address for the port server and must be unique for all systems.

The server parameter defaults to the system that responded to the bootp or dhcp request when the port server was last reset. This server must be providing tftp service.

Some servers enforce security measures on tftp transfers to the server. For example, the Solaris version of the tftp daemon has an “-s” option that prevents:

- creation of any files
- access to any directory except /tftpboot
- overwriting any file that does not have world write permission.

Because of these restrictions, you must create an empty file in the /tftpboot directory with world write access before attempting to save a file to that system. Do this by entering the following on the server.

    rm /tftpboot/myfile.bdnl
    touch /tftpboot/myfile.bdnl
    chmod 0777 /tftpboot/myfile.bdnl

    NOTE: If you plan to copy a configuration database from one port server to another, it is simpler if the port servers receive their IP addresses from a DHCP or BOOTP server. By using the DHCP/BOOTP provided IP address there is no need to save an IP address in the configuration database.

If you do not have a DHCP/BOOTP server for your network, you can still copy a configuration database by doing the following:

1. Save the configuration database from the original port server to the server.
2. Disconnect the original port server from the network.
3. Load the configuration to the new port server.
4. Reboot the new port server. This system will come up with the IP address of the original port server.
5. Set the IP address to the proper value for the new port server.
6. Save the configuration
7. Reboot the new port server. It will now have the proper IP address.
8. Reconnect the original port server to the network.

Flags
The `section` keyword must be one of the following:

<table>
<thead>
<tr>
<th>Flag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>config</td>
<td>configuration database (default)</td>
</tr>
<tr>
<td>config_def</td>
<td>factory default configuration database</td>
</tr>
<tr>
<td>firmware</td>
<td>software for initializing the port server</td>
</tr>
<tr>
<td>product_data</td>
<td>unique manufacturing data for this port server</td>
</tr>
<tr>
<td>onboard</td>
<td>operational software</td>
</tr>
</tbody>
</table>

Example

```
[expert:1/6]>>flash_save tmp.bdnl
accessing 199.171.31.12
..................................
save complete
```

Interactive example

```
[expert:1/7]>>flash_save
Select flash component to be saved
  c - current configuration database
  d - default configuration database
  f - firmware
  o - onboard system software
  p - product/manufacturing data
Selection: c
file name: tmp.bdnl
    default tftp server is 199.171.31.12
server name or IP address (return for default):
Saving current configuration as tmp.bdnl on default tftp server
accessing 199.171.31.12 ..................
save complete
```
**flash_update**

**Syntax**

    flash_update [filename [server]]

**Description**

Use the `flash_update` command to load new software or configuration information. If there are no parameters, an interactive mode is assumed.

The server parameter defaults to the system that responded to the bootp or dhcp request when the port server was last reset. This server must be providing tftp service.

**Flags**

- `?` display usage information

**Example**

```
[expert:1/8]>>flash_update
filename of download image: new_release.bdnl
default tftp server is 199.171.31.12
server name or IP address (return for default): accessing 199.171.31.12
...............................................
<duplicate lines removed from example>
...............................................
replacing onboard code
erase complete
update complete
843736 bytes recorded in flash
do you wish to reboot now? (y/n) y
```
frstats

Syntax

frstats [-r] [-l] {dlci number}

Description

Available only as part of the optional frame relay software. The frstats command displays the status of the frame relay link. Link status, the number of link queries made and responded to, and the frames transmitted and received are displayed. If a DLCI number is specified, PVC statistics for the specified DLCI will be displayed, including the number of frames and octets transmitted and received across the specified PVC and the number of frames received with congestion management information.

Flags

| L | lists all PVC’s that are available from the network, including active, inactive, and unconfigured on the port server. |
| R | resets the statistics. |
**gdbserv**

Only for use with the Developer's Kit. For information on its use, see the documentation provided with the Developer's Kit. Use of gdbserv without information and resources provided in the Developer's Kit can cause the port server to hang.
hangup

Syntax
   hangup

Description
The hangup command generates a hangup on the current line. This terminates all sessions on the line.
help

Syntax
   help

Description
The help command displays a list of available commands.
http_stats

Syntax

http_stats

Description

The http_stats command is an admin level command that displays some statistics concerning the operation of the port server HTTP server (available for configuration from a web browser). The command gives HTTP Request and Response statistics, as well as information extracted from the last HTTP packet received by the server.

Example

[admin:1/5]>>http_stats
port server HTTPD Server Statistics
--------------------------------
General Statistics
  Request Packets : 4
  Response Packets : 4
  Largest Response (bytes) : 1724
HTTP Request Statistics
  GET Requests : 3
  POST Requests : 0
  HEAD Requests : 0
HTTP Response Statistics
  Response 200 (OK) : 3
  Response 400 (Parse Error) : 0
  Response 401 (Unauthorized) : 1
  Response 403 (Forbidden) : 0
  Response 404 (Not Found) : 0
  Response 500 (Internal Error) : 0
  Response 501 (Unsupported) : 0
HTTP Error Statistics
  Invalid Packets : 0
  Authentication Errors : 0
  Network Errors : 0
  Internal Errors : 0
HTTP Last Request Packet Information
  Packet Type : POST
  URL : /post/1
  Authentication Information : Yes
  Content Type : application/x-www-form-urlencoded
  Content Length : 114
  Actual Entity Length : 114
ifconfig

Syntax

```
ifconfig {interface} [address_family] [address [dest_addr]] [up] [down] [netmask mask] [broadcast broadcast_address] [metric n] [trailers | -trailers] [onepacket size threshold | -onepacket] [arp | -arp]
```

```
ifconfig {interface} [protocol_family]
```

Description

The ifconfig program assigns an address to a network interface or configures network interface parameters. Ifconfig may be used at a later time to redefine an interface's address or other operating parameters.

Since an interface may receive transmissions in differing protocols, each of which may require separate naming schemes, you must specify the address_family, which may change the interpretation of the remaining arguments. Currently, only the Internet address family is supported. Thus, the only valid value for address_family is inet.

Ifconfig displays the current configuration for a network interface when no optional parameters are supplied.

Flags

<table>
<thead>
<tr>
<th>Flag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>address</td>
<td>Specifies the address of the local end of the interface.</td>
</tr>
<tr>
<td></td>
<td>For the Internet address family, the address is either a host name or an Internet address expressed in Internet standard dot notation.</td>
</tr>
<tr>
<td>address_family</td>
<td>Specifies the address family. Currently the only valid value is inet.</td>
</tr>
<tr>
<td>dest_address</td>
<td>Specifies the address of the remote end of the interface.</td>
</tr>
<tr>
<td>interface</td>
<td>Identifies the interface, in a string of the form name unit (for example, es0).</td>
</tr>
<tr>
<td>protocol_family</td>
<td>Specifies that ifconfig should report only the details specific to that protocol family.</td>
</tr>
</tbody>
</table>

Parameters

The following parameters may be set with ifconfig:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>arp</td>
<td>Enables the use of the Address Resolution Protocol in mapping between network level addresses and link level addresses (default).</td>
</tr>
<tr>
<td>-arp</td>
<td>Disables the use of the Address Resolution Protocol.</td>
</tr>
<tr>
<td>Broadcast</td>
<td>(Internet only) Specifies the address to use to represent broadcasts to the network. The default broadcast address is the address with a host part of all ones (1's).</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>debug</strong></td>
<td>Enables driver-dependent debugging code; usually, this turns on extra console error logging.</td>
</tr>
<tr>
<td><code>-debug</code></td>
<td>Disables driver-dependent debugging code.</td>
</tr>
<tr>
<td><strong>dest_address</strong></td>
<td>Specifies the address of the correspondent on the other end of a point-to-point link.</td>
</tr>
<tr>
<td><strong>down</strong></td>
<td>Marks an interface as down. When an interface is marked down, the system will not attempt to transmit messages through that interface. If possible, the interface will be reset to disable reception as well. This action does not automatically disable routes using the interface. Setting a dial-on-demand interface to down has the additional side effect of disabling the dial-on-demand functionality, as well as dropping any existing connections.</td>
</tr>
<tr>
<td><strong>metric n</strong></td>
<td>Sets the routing metric of the interface to n (default 0). The routing metric is used by the route daemon. Higher metrics make a route less favorable; metrics are counted as additional hops to the destination network or host.</td>
</tr>
<tr>
<td><strong>netmask mask</strong></td>
<td>(Internet only) Specifies how much of the address to reserve for subdividing networks into subnetworks. The mask includes the network and the subnet portion of the local address. The subnet part is taken from the host field of the address. The mask can be specified as a single hexadecimal number with a leading 0x, or with a dot-notation Internet address. The mask contains 1's for the bit positions in the 32-bit address which are to be used for the network and subnet parts, and 0's for the host part. The mask should contain at least the standard network portion, and the subnet field should be contiguous with the network portion.</td>
</tr>
<tr>
<td><strong>onepacket size threshold</strong></td>
<td>Enables one-packet mode (used for interfaces that can not handle back-to-back packets). The keyword onepacket must be followed by two numeric parameters: size specifies the small packet size in bytes; threshold specifies the small packet threshold. If small packet detection is not desired, these values should be zero.</td>
</tr>
<tr>
<td><code>-onepacket</code></td>
<td>Disables one-packet mode.</td>
</tr>
<tr>
<td><strong>trailers</strong></td>
<td>Requests the use of trailer link level encapsulation when sending (this is the default). If a network interface supports trailers, the system will, when possible, encapsulate outgoing messages in a manner which minimizes the number of memory-to-memory copy</td>
</tr>
</tbody>
</table>
operations performed by the receiver. On networks that support ARP, this flag indicates that the system should request that other systems use trailers when sending to this host. Similarly, trailer encapsulations will be sent to other hosts that have made such requests. Currently used by Internet protocols only.

-trailers

Disables the use of trailer link level encapsulation.

up

Marks an interface as up. This may be used to enable an interface after an ifconfig down. It happens automatically when setting the first address on an interface. If the interface was reset when previously marked down, the hardware will be reinitialized.

Example

[expert:1/13]>>ifconfig es0
es0:  flags=63<UP,BROADCAST,NOTRAILERS,RUNNING>
inet 192.35.53.234 netmask ffffff00 broadcast 192.35.53.255
**kill**

**Syntax**

```bash
kill [-l] [-a] [ [-signal] process_id]
```

**Description**

The `kill` command sends a signal to a process identified by the process ID number. If a signal number preceded by "-" is given as the first argument, that signal is sent instead of the default SIGQUIT (3). The signal names and their respective numbers are given by using the `-l` option.

**Flags**

- `-l`
  
  Displays a list of signal names and their values.
lookmem

Syntax

lookmem

Description

The lookmem command shows memory statistics that are often useful in debugging/troubleshooting.

This command is designed for use by development and support personnel for diagnostic purposes.

Example

```
[expert :1/13]> lookmem
ORIGINAL HEAP SIZE = 0x180000 bytes
WALK FREE LIST = 0xe74a4 bytes
```

where:

<table>
<thead>
<tr>
<th>ORIGINAL HEAP SIZE</th>
<th>Indicates the size of the total pre-allocated kernel heap.</th>
</tr>
</thead>
<tbody>
<tr>
<td>WALK FREE LIST</td>
<td>Indicates the amount of kernel memory currently available (computed by traversing the free list).</td>
</tr>
</tbody>
</table>
lookstream

Syntax

lookstream

Description

The lookstream command shows STREAMS statistics, a capability that is useful for debugging/troubleshooting.

This command is designed for use by development and support personnel for diagnostic purposes.

Sample Output

[expert:1/30]>> lookstream

<table>
<thead>
<tr>
<th>Item</th>
<th>Size</th>
<th>Total</th>
<th>Alloc</th>
<th>Free</th>
<th>Max</th>
<th>Allocs</th>
<th>Overflows</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buf16</td>
<td>16</td>
<td>16</td>
<td>6</td>
<td>10</td>
<td>9</td>
<td>131</td>
<td>0</td>
</tr>
<tr>
<td>Buf32</td>
<td>32</td>
<td>16</td>
<td>2</td>
<td>14</td>
<td>60</td>
<td>222</td>
<td>44</td>
</tr>
<tr>
<td>Buf64</td>
<td>64</td>
<td>64</td>
<td>27</td>
<td>37</td>
<td>73</td>
<td>783</td>
<td>17</td>
</tr>
<tr>
<td>Buf128</td>
<td>128</td>
<td>64</td>
<td>13</td>
<td>51</td>
<td>17</td>
<td>193</td>
<td>0</td>
</tr>
<tr>
<td>Buf256</td>
<td>256</td>
<td>16</td>
<td>12</td>
<td>4</td>
<td>20</td>
<td>207</td>
<td>20</td>
</tr>
<tr>
<td>Buf512</td>
<td>512</td>
<td>8</td>
<td>0</td>
<td>8</td>
<td>1</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Buf1024</td>
<td>1024</td>
<td>8</td>
<td>0</td>
<td>8</td>
<td>1</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Buf2048</td>
<td>2048</td>
<td>80</td>
<td>0</td>
<td>80</td>
<td>2</td>
<td>27</td>
<td>0</td>
</tr>
<tr>
<td>LrgBuf</td>
<td>2048</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Mblks</td>
<td>-</td>
<td>512</td>
<td>60</td>
<td>452</td>
<td>165</td>
<td>1682</td>
<td>0</td>
</tr>
<tr>
<td>Stacks</td>
<td>-</td>
<td>50</td>
<td>2</td>
<td>48</td>
<td>3</td>
<td>366</td>
<td>0</td>
</tr>
<tr>
<td>Mgs</td>
<td>-</td>
<td>10</td>
<td>10</td>
<td>0</td>
<td>11</td>
<td>508</td>
<td>78</td>
</tr>
<tr>
<td>Timers</td>
<td>-</td>
<td>100</td>
<td>6</td>
<td>94</td>
<td>8</td>
<td>310</td>
<td>0</td>
</tr>
</tbody>
</table>

Pre-allocated memory = 310648 (0x4bd78)
Currently allocated memory = 12488 (0x30c8)
Maximum allocated memory = 33100 (0x814c)

where:

<table>
<thead>
<tr>
<th>Item</th>
<th>Indicates a resource type.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size</td>
<td>Indicates the number of bytes in the resource.</td>
</tr>
<tr>
<td>Total</td>
<td>Indicates the number of pre-allocated resources of type Item.</td>
</tr>
<tr>
<td>Alloc</td>
<td>Indicates the number of these resources in use.</td>
</tr>
<tr>
<td>Free</td>
<td>Indicates the number of these resources currently available for use.</td>
</tr>
<tr>
<td>Max</td>
<td>Indicates the maximum number of these resources ever used at one time.</td>
</tr>
<tr>
<td>Allocs</td>
<td>Indicates the total number of these resources used for all time.</td>
</tr>
<tr>
<td><strong>Overflows</strong></td>
<td>Indicates the total number of allocations beyond those pre-allocated resources.</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>

Memory allocations are listed at the bottom.
**lpq**

**Syntax**

```
lpq
```

**Description**

Use the `lpq` command to display the print queue.

**Example**

```
[admin:1/1]>>lpq
```

The print queue displays:

```
job 1 NT4_JJ:ModemLog_MultiModem MT2834ZDX.txt - Notepad
  waiting for printer LPR3
  file size 7433 bytes
```

```
job 2 NT4_JJ:ModemLog_MultiModem MT2834ZDX.txt - Notepad
  waiting for printer LPR3
  file size 7433 bytes
```

```
job 3 NT4_JJ:ModemLog_MultiModem MT2834ZDX.txt - Notepad
  waiting for printer LPR3
  file size 7433 bytes
```
lprm

Syntax
lprm [n | all] [-h]

Description
Use the lprm command to delete a job or jobs from the print queue.

Flags

n Specifies job number or list of numbers. Job numbers are the local job numbers displayed with the lpq command.

all Specifies all jobs.

-h Displays usage message.

Example

[admin:1/1]>>lprm 1 Delete job number 1 from queue
[admin:1/1]>>lprm 1 3 Delete job numbers 1 and 3 from queue
[admin:1/1]>>lprm all Delete all jobs from queue.

NOTE: You should delete print jobs from the host. The host may resend print jobs if you delete them from the port server.
mpool

Syntax

    mpool [-v] [-l [-s stty_string]] (device1) [device2 ...]

Description

The mpool command selects a modem from a shared pool of modems. When users request the service, they are connected to whichever device is free. Mpool establishes a connection from standard input to one or more device ports. The names of the ports, and the default port behavior, is specified in the configuration database.

The mpool command may be issued from the shell command line or it may be configured to be automatically invoked when a user or system requests the service.

Flags

- **-l** Invokes the ldterm STREAMS module, which passes data to the user line by line rather than character by character. This means that the user sees only the end result of line operations, and not backspaces, overwrites, and other such line control operations as they occur. The default is to pass raw data character by character.

- **-s stty_string** Specifies a set of stty strings which determine the behavior of the port, such as baud rate and parity. For a complete list of these strings, refer to the stty man page later in this chapter. To list more than one stty string, enclose the list in double quotation marks. The -s option may not be used without the -l option.

- **-v** Enables verbose mode, which displays the name of the port which mpool selected for your use.

- **device1** [device2 ...] Lists one or more physical ports. When the user requests a modem (or other device) from the service, the STREAMS operating system attempts to open the ports one by one until an available port is successfully opened.

**NOTE:** Use the set rtelnet command to configure a port to use mpool automatically.
netstat

Syntax

    netstat [-Aainrs] [-f address_family] [-l interface] [-p protocol] [interval]

Description

The netstat command symbolically displays the contents of various network-related data structures. There are a number of display formats, depending on the information presented. The default display, for active endpoints, shows the local and remote addresses, send and receive queue sizes (in bytes), protocol, and, optionally, the internal state of the protocol.

Address formats are of the form host.port or network.port if an endpoint's address specifies a network but no specific host address. When known, the host address is displayed symbolically. If a symbolic name for an address is unknown, or if the -n option is specified, the address is printed in Internet dot format. Unspecified, or wildcard, addresses and ports appear as "*.*.*".

The interface display provides a table for cumulative statistics regarding packets transferred, errors, and collisions. The network address (currently Internet specific) of the interface and the maximum transmission unit (mtu) are also displayed.

The routing table display indicates the available routes and their status. Each route consists of a destination host or network and a gateway to use in forwarding packets. The flags field shows the state of the route (U if "up"), and whether the route is to a gateway (G). Direct routes are created for each interface attached to the local host. The refcnt field gives the current number of active uses of the route. Connection-oriented protocols normally hold on to a single route for the duration of a connection, while connectionless protocols obtain a route and then discard it. The use field provides a count of the number of packets sent using that route. The interface entry indicates the network interface utilized for the route.

When netstat is invoked with an interval argument, it displays a running count of statistics related to network interfaces. This display consists of a column summarizing information for the loopback interface only and a column summarizing information for all interfaces. The first line of each screen of information contains a summary since the system was last rebooted. Subsequent lines of output show values accumulated over the preceding interval.

Flags

<table>
<thead>
<tr>
<th>Flag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-A</td>
<td>Displays the address of any associated protocol control blocks; used for debugging/troubleshooting.</td>
</tr>
<tr>
<td>-a</td>
<td>Displays the state of all endpoints; normally endpoints used by server processes are not shown.</td>
</tr>
<tr>
<td>-f address_family</td>
<td>Limits statistics and control block displays to address_family. The</td>
</tr>
</tbody>
</table>
only address_family currently supported is inet.

-1 Displays the state of interfaces which have been auto-configured (interfaces statically configured into a system, but not located at boot time are not shown).

-1 interface Displays interface state for interface only.

-n Displays network addresses as numbers (normally, netstat interprets addresses and attempts to display them symbolically).

-p protocol Limits statistics and control block displays to protocol (e.g., tcp).

-r Displays the routing tables.

-s Displays per-protocol statistics.

Interval If an interval is specified, netstat will continuously display the information regarding packet traffic on the configured network interfaces, pausing interval seconds before refreshing the screen.
**ping**

**Syntax**

```
```

**Description**

The ping command uses the ICMP protocol's mandatory ECHO_REQUEST datagram to elicit an ICMP ECHO_RESPONSE from a host or gateway. ECHO_REQUEST datagrams (pings) have an IP and ICMP header, followed by a struct timeval and then an arbitrary number of "pad" bytes used to fill out the packet.

**Flags**

<table>
<thead>
<tr>
<th>Flag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-c count</code></td>
<td>Stops after sending (and receiving) count ECHO_RESPONSE packets.</td>
</tr>
<tr>
<td><code>-f</code></td>
<td>Floods ping. Outputs packets as fast as they come back, or one hundred times per second, whichever is more. For every ECHO_REQUEST sent, a period (.) is printed, while for every ECHO_REPLY received, a backspace is printed. This provides a rapid display of how many packets are being dropped. This can induce a high network load and should be used with caution.</td>
</tr>
<tr>
<td><code>-i wait</code></td>
<td>Waits wait seconds between sending each packet. The default is wait for 1 second between each packet. This option is incompatible with the <code>-f</code> option.</td>
</tr>
<tr>
<td><code>-l preload</code></td>
<td>If preload is specified, ping sends that many packets as fast as possible before falling into its normal mode of behavior.</td>
</tr>
<tr>
<td><code>-n</code></td>
<td>Specifies numeric output only. No attempt will be made to look up symbolic names for host addresses.</td>
</tr>
<tr>
<td><code>-p pattern</code></td>
<td>You may specify up to 16 &quot;pad&quot; bytes to fill out the packet you send. This is useful for diagnosing data-dependent problems in a network. For example, <code>-p ff</code> will cause the sent packet to be filled with all ones (1's).</td>
</tr>
<tr>
<td><code>-q</code></td>
<td>Specifies quiet output. Nothing is displayed except the summary lines at startup time and when finished.</td>
</tr>
<tr>
<td><code>-R</code></td>
<td>Records the route. Includes the RECORD_ROUTE option in the ECHO_REQUEST packet and displays the route buffer on returned packets. Note that the IP header is only large enough for nine such routes. Many hosts ignore or discard this option.</td>
</tr>
</tbody>
</table>
| `-r` | Bypasses the normal routing tables and sends directly to a host on an attached network. If the host is not on a directly attached network, an error is returned. This option can be used to ping a local host through an interface that has no route through it (for example, after the interface was
ICMP Packet Details

An IP header without options is 20 bytes. An ICMP ECHO_REQUEST packet contains an additional 8 bytes worth of ICMP header followed by an arbitrary amount of data. A `packetsize`, if given, indicates the size of this extra piece of data (the default is 56). Thus, the amount of data received inside of an IP packet of type ICMP ECHO_REPLY will always be 8 bytes more than the requested data space (the ICMP header).

If the data space is at least 8 bytes large, `ping` uses the first 8 bytes of this space to include a time stamp which it uses in computing round-trip times. If less than 8 bytes of pad are specified, no round-trip times are given.

Duplicate and Damaged Packets

The `ping` command will report duplicate and damaged packets. Duplicate packets should never occur and seem to be caused by inappropriate link-level retransmissions. Duplicates may occur in many situations and are rarely (if ever) a good sign, although the presence of low levels of duplicates may not always be cause for alarm.

Damaged packets are obviously serious cause for alarm and often indicate broken hardware or software somewhere in the `ping` packet's path (in the network or in the hosts).

Trying Different Data Patterns

The (inter) network layer should never treat packets differently depending on the data contained in the data portion. Unfortunately, data-dependent problems have been known to sneak into networks and remain undetected for long periods of time. In many cases, the particular pattern that will have problems is something that does not have sufficient
“transitions,” such as all ones (1’s) or all zeros (0’s), or a pattern right at the edge, such as almost all zeros. It is not necessarily enough to specify a data pattern of all zeros (for example) on the command line because the pattern that is of interest is at the data link level, and the relationship between what you type and what the controllers transmit can be complicated.

Consequently, if you have a data-dependent problem, you will probably have to do a lot of testing to find it. You may find a file that either cannot be sent across your network or that takes much longer to transfer than other similar length files. You can then examine this file for repeated patterns that you can test using the \(-p\) option of `ping`.

**TTL Details**

The TTL value of an IP packet represents the maximum number of IP routers that the packet can go through before being thrown away. In current practice, you can expect each router in the Internet to decrement the TTL field by exactly one. The TCP/IP specification states that the TTL field for TCP packets should be set to 60, but many systems use smaller values (4.3 BSD uses 30, 4.2 used 15).

The maximum possible value of this field is 255, and most UNIX systems set the TTL field of ICMP ECHO_REQUEST packets to 255. This is why you can ping some hosts, but not reach them with `telnet` or `ftp`.

In normal operation, `ping` prints the TTL value from the packet it receives. When a remote system receives a `ping` packet, it has three ways it can handle the TTL field in its response:

- Not change it; this is what Berkeley UNIX systems did before the 4.3 BSD-tahoe release. In this case, the TTL value in the received packet will be 255 minus the number of routers in the round-trip path.
- Set it to 255; this is what current Berkeley UNIX systems do. In this case, the TTL value in the received packet will be 255 minus the number of routers in the path from the remote system to the pinging host.
- Set it to another value. Some machines use the same value for ICMP packets that they use for TCP packets, e.g., either 30 or 60. Others may use completely wild values.

**NOTE:** Many hosts and gateways ignore the RECORD_ROUTE option. The maximum IP header length is too small for options like RECORD_ROUTE to be completely useful. Flood pinging is not recommended in general, and flood pinging the broadcast address should only be done under very controlled conditions.

**Example**

```
[admin:1/36]>> ping -c 3 prep.ai.mit.edu
PING prep.ai.mit.edu (18.71.0.38): 56 data bytes
64 bytes from AENEAS.MIT.EDU (18.71.0.38): icmp_seq=0 ttl=243 time=201 ms
64 bytes from AENEAS.MIT.EDU (18.71.0.38): icmp_seq=1 ttl=243 time=269 ms
64 bytes from AENEAS.MIT.EDU (18.71.0.38): icmp_seq=2 ttl=243 time=578 ms
--- prep.ai.mit.edu ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 201/349/578 ms
```
port_stat

Syntax

    port_stat [-c] [-r]

Description

The port_stat command is an expert level command that gives a one page overview of the status of all the ports on the port server. This page is updated every few seconds whenever the statistics change. The display is designed primarily for developers.

The port_stat command is used to produce a constantly updated screen of status information about each of the serial ports, the parallel port, and the WAN port.

In order to run this command, it is necessary to provide some information about the terminal where the output will be displayed. This information is provided via the environment variables TERM, LINES, and COLUMNS.

Example

The setting of this information is shown in the following example:

    [expert:1/10]>>setenv TERM xterm
    [expert:1/11]>>setenv LINES 36
    [expert:1/12]>>setenv COLUMNS 90

The terminal type specified for TERM can be xterm, wy50, vt100, ansi, and a few other less common types. Once the environment variables are set, the port_stat command is issued to display a screen of information about the status of the ports. The information can be displayed in one of two modes: cooked or raw. The default is cooked.

To display the port status in cooked mode, the command is:

    [expert:1/13]>>port_stat

The command clears the screen and displays the information. The information is updated once a second. A sample of the output for an RCS/4000 is shown below:

<table>
<thead>
<tr>
<th>DRCDDR</th>
<th>CTTSTI</th>
<th>FP</th>
<th>DSSRR</th>
<th>WO</th>
<th>OP</th>
<th>EX</th>
<th>NB</th>
<th>BL</th>
<th>^s</th>
<th>HC</th>
<th>HD</th>
<th>speed</th>
<th>input</th>
<th>output</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>DRCST-</td>
<td>OP</td>
<td>9600</td>
<td>35</td>
<td>4616</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>-R-ST-</td>
<td>WO</td>
<td>9600</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-R-ST-</td>
<td>WO</td>
<td>9600</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>-R-ST-</td>
<td>WO</td>
<td>9600</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>-R-ST-</td>
<td>WO</td>
<td>9600</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>-R-ST-</td>
<td>WO</td>
<td>9600</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>-R-ST-</td>
<td>WO</td>
<td>9600</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>-R-ST-</td>
<td>WO</td>
<td>9600</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>-R--T-</td>
<td>OP</td>
<td>64102</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>online</td>
<td>ok</td>
<td>NO PAPER</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The first column of the output is the port number. The number of ports on a port server varies from 2 to 16. The serial port number is shown by the port on your port server. The port numbering one some models begins with port 0; on other models, it begins with port 1. Some port server models have a WAN port and/ or a printer port. The printer port is port 17 or 32 and the WAN port is port 18 or 33. See the hardware manual provided with your port server to determine which ports are available on your port server.

The second column of output consists of six separate characters that show the setting of each of six modem signals (exception: different output is shown for the printer port, discussed below). A header identifies each setting. For example, the first character indicates the DCD signal, as shown by the vertical header:

```
D
C
D
```

Either a D or a hyphen (-) is shown under the header. D means DCD is present; - means DCD is not present. The second character shows the setting for RTS, identified by the vertical header. An R is shown if RTS is present; a - is shown if RTS is not present. The 6 modem signal values are:

<table>
<thead>
<tr>
<th>Modem Signal</th>
<th>Character</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCD</td>
<td>D</td>
</tr>
<tr>
<td>RTS</td>
<td>R</td>
</tr>
<tr>
<td>CTS</td>
<td>C</td>
</tr>
<tr>
<td>DSR</td>
<td>S</td>
</tr>
<tr>
<td>DTR</td>
<td>T</td>
</tr>
<tr>
<td>RI</td>
<td>I</td>
</tr>
</tbody>
</table>

The 6 modem signal values are:

- DCD (D) carrier detect
- RTS (R) request to send
- CTS (C) clear to send
- DSR (S) data set ready
- DTR (T) data terminal ready
- RI (I) ring indicator. (Not supported)

The next 8 columns of the display may contain a short string to provide noteworthy status information (exception: different output is shown for the printer port, discussed below). The column header shows the string for the column. Each line, representing a port, will be blank or will show the status code. NOTE: The first of these columns is labeled both FP and WO. One or the other may show in the column.

<table>
<thead>
<tr>
<th>FP</th>
<th>Failed Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>WO</td>
<td>a process is waiting for the port to open (waiting for DCD signal).</td>
</tr>
<tr>
<td>OP</td>
<td>a process has the port open.</td>
</tr>
<tr>
<td>EX</td>
<td>the port is opened in exclusive mode by a process.</td>
</tr>
<tr>
<td>NB</td>
<td>no receive buffer is available for the port. Data may have been lost</td>
</tr>
<tr>
<td>BL</td>
<td>port has issued flow control to block input.</td>
</tr>
<tr>
<td>^s</td>
<td>output has been stopped with a ctrl-s. The port is flow controlled off (XOFF) and transmission is stopped.</td>
</tr>
<tr>
<td>HC</td>
<td>the port is flow controlled off (CTS) and transmission is stopped.</td>
</tr>
</tbody>
</table>
In the output line for the printer port, there are three columns where the second and additional columns are shown for the other ports. Each of the three columns shows a status message. Each status message has two possible values, one for normal operation and one for an error condition. The status messages, in order, are:

<table>
<thead>
<tr>
<th>Normal</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. online</td>
<td>OFFLINE</td>
</tr>
<tr>
<td>2. ok</td>
<td>ERR</td>
</tr>
<tr>
<td>3. has paper</td>
<td>NO PAPER</td>
</tr>
</tbody>
</table>

The final three columns provide the speed of the port and the count of input and output characters on that port. The speed shown for the WAN port (port 33) is the actual speed.

If you use the `-r` option with the `port_stat` command, the display will be in raw mode, as follows:

<table>
<thead>
<tr>
<th></th>
<th>i</th>
<th>o</th>
<th>c</th>
<th>l</th>
<th>st</th>
<th>hs</th>
<th>input</th>
<th>output</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>DRCST-</td>
<td>2c13</td>
<td>1805</td>
<td>0000004bd</td>
<td>08ea</td>
<td>40014</td>
<td>0</td>
<td>577</td>
</tr>
<tr>
<td>1</td>
<td>--R--</td>
<td>0c10</td>
<td>0000</td>
<td>0000004bd</td>
<td>0000</td>
<td>40002</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>---S--</td>
<td>0c10</td>
<td>0000</td>
<td>0000004bd</td>
<td>0000</td>
<td>02000</td>
<td>0</td>
<td>36210310</td>
</tr>
<tr>
<td>3</td>
<td>---S--</td>
<td>0c10</td>
<td>0000</td>
<td>0000004bd</td>
<td>0000</td>
<td>02000</td>
<td>0</td>
<td>36233630</td>
</tr>
<tr>
<td>4</td>
<td>---S--</td>
<td>0c10</td>
<td>0000</td>
<td>0000004bd</td>
<td>0000</td>
<td>02000</td>
<td>0</td>
<td>36223026</td>
</tr>
<tr>
<td>5</td>
<td>---S--</td>
<td>0c10</td>
<td>0000</td>
<td>0000004bd</td>
<td>0000</td>
<td>02000</td>
<td>0</td>
<td>36248097</td>
</tr>
<tr>
<td>6</td>
<td>---S--</td>
<td>0c10</td>
<td>0000</td>
<td>0000004bd</td>
<td>0000</td>
<td>02000</td>
<td>0</td>
<td>36342266</td>
</tr>
<tr>
<td>7</td>
<td>---S--</td>
<td>0c10</td>
<td>0000</td>
<td>0000004bd</td>
<td>0000</td>
<td>02000</td>
<td>0</td>
<td>36319417</td>
</tr>
<tr>
<td>8</td>
<td>---S--</td>
<td>0c10</td>
<td>0000</td>
<td>0000004bd</td>
<td>0000</td>
<td>02000</td>
<td>0</td>
<td>36262645</td>
</tr>
<tr>
<td>9</td>
<td>---S--</td>
<td>0c10</td>
<td>0000</td>
<td>0000004bd</td>
<td>0000</td>
<td>02000</td>
<td>0</td>
<td>36244966</td>
</tr>
<tr>
<td>10</td>
<td>---S--</td>
<td>0c10</td>
<td>0000</td>
<td>0000004bd</td>
<td>0000</td>
<td>02000</td>
<td>0</td>
<td>36399535</td>
</tr>
<tr>
<td>11</td>
<td>---S--</td>
<td>0c10</td>
<td>0000</td>
<td>0000004bd</td>
<td>0000</td>
<td>02000</td>
<td>0</td>
<td>36341176</td>
</tr>
<tr>
<td>12</td>
<td>---S--</td>
<td>0c10</td>
<td>0000</td>
<td>0000004bd</td>
<td>0000</td>
<td>02000</td>
<td>0</td>
<td>36217612</td>
</tr>
<tr>
<td>13</td>
<td>---S--</td>
<td>0c10</td>
<td>0000</td>
<td>0000004bd</td>
<td>0000</td>
<td>02000</td>
<td>0</td>
<td>36257930</td>
</tr>
<tr>
<td>14</td>
<td>DRCST-</td>
<td>2c13</td>
<td>1805</td>
<td>0000004bd</td>
<td>086b</td>
<td>40034</td>
<td>38400</td>
<td>36186889</td>
</tr>
<tr>
<td>15</td>
<td>DRCST-</td>
<td>2c13</td>
<td>1805</td>
<td>0000004bd</td>
<td>086b</td>
<td>40034</td>
<td>38400</td>
<td>36262306</td>
</tr>
<tr>
<td>33</td>
<td>-RCST-</td>
<td>0000</td>
<td>0000</td>
<td>00000000</td>
<td>0000</td>
<td>00000</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>32</td>
<td>-RCST-</td>
<td>0000</td>
<td>0000</td>
<td>00000000</td>
<td>0000</td>
<td>00000</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Rather than the status codes shown in the "cooked" mode, six columns of "raw" information are displayed. This information is used by Technical Support representatives.
power

(Note: power can be assigned to a serial port only on a POS/2000)

Syntax

    power [ port(s) ] [ on | off ]

Description

The power command is an administrative level command that turns the power on or off to one or more specified ports. The power can only be turned on for a port that has already been configured for power using the set power command. Thus, there are two steps to assigning power to a port:

1. Configure the port for power (see the set POWER command, page 141).
2. Apply power to the port using the power command.

The two steps can be combined into step 1 if the i option is used with the set power command. If the i option is used with the set power command or if the POS is rebooted after issuing the set power command, power will be applied to the port without requiring the use of the power command.

To turn on the power to ports 3, 4, and 5, enter:

    power 3-5 on

If any port(s) specified is not configured for power, an error message will be displayed and the power will not be turned on for that port. For instance, in the above example, if port 4 has not been previously configured for power using the set power command, the message:

    power: port4 has no power configured

will be displayed and the power will be turned on for ports 3 and 5 only.

The power command is most useful for situations in which the power has been turned off to a port that is configured for power. Situations in which a port might be configured for power but the power is turned off on the port are:

1. The total peripheral power consumption exceeded the limit allowed (45 watts). A message is displayed:

    Automatically turned off

2. The peripheral power consumption on the port exceeded the port current limit (1 Amp). A message is displayed:

    Over current protection tripped

3. The power was manually turned off using the power command. A message is displayed:

    Manually turned off

When the power has been turned off to a port for one of the above situations, the power command can be used to restore the power to the port.
The `power` command can be used to turn the power off for a port. Turning the power off can be necessary to clear an error condition from an LED. To turn off the power to port 6, enter:

```
power 6 off
```

Turning the power off using the `power` command does not remove the power configuration for the port from the configuration database. If the POS is rebooted, the power configuration in the configuration database will turn the power on for the port.
**ppool**

**Syntax**

```
ppool [-v] [-l [-s stty_string]] {device1} [device2 ...]
```

**Description**

The ppool command selects a printer from a shared pool of printers. When users request the service, they are connected to whichever device is free. Ppool establishes a connection from standard input to one or more device ports. The names of the ports, and the default port behavior, are specified in the configuration database.

The ppool command may be issued from the shell command line or it may be configured to be automatically invoked when a user or system requests the service.

**Flags**

<table>
<thead>
<tr>
<th>Flag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-l</code></td>
<td>Invokes the ldterm STREAMS module, which passes data to the user line by line rather than character by character. This means that the user sees only the end result of line operations, and not backspaces, overwrites, and other such line control operations as they occur. The default is to pass raw data character by character.</td>
</tr>
<tr>
<td><code>-s stty_string</code></td>
<td>Specifies a set of stty strings which determine the behavior of the port, such as baud rate and parity. For a complete list of these strings, refer to the stty man page later in this chapter. To list more than one stty string, enclose the list in double quotation marks. The -s option may not be used without the -l option.</td>
</tr>
<tr>
<td><code>-v</code></td>
<td>Enables verbose mode, which displays the name of the port which ppool selected for your use.</td>
</tr>
</tbody>
</table>

**device1**

Lists one or more physical ports. When the user requests a modem (or other device) from the service, the STREAMS operating system attempts to open the ports one by one until an available port is successfully opened.

**NOTE:** Use the `set rtelnet` command to configure a port to use ppool automatically.
Use the `ppp` command to attach serial lines as network interfaces, using the Point-to-Point Protocol (PPP).

**NOTE:** You can also configure a serial port to use PPP for any data transmitted by defining and applying a custom service configuration (see [PPP](#) page [144]).

### Syntax

```
ppp [-i] [-V] [{ A | S | t }] [-f { rtscts | xonxoff | noflow }]
[-d debug_level] [-u uucp_system] [-n interface_name]
[-T tmout] [-m mru] [-a acc_map] [-p { pap | chap } -k key]
[-x { noproxy | proxy }] [-P pool name] [-r { 1172 | 1332 } [-s speed]]
[-D device_name] [[remote:] local [mask]]
```

Use a remote/ local address defined in the specified ppp address pool:

- `P` option: `[-P pool name]`

The pool should be set up previously using the `set pppadlist` command (see [PPPADLIST](#) page [146]).

Whether or not to install:

- `x` option: `[-x noproxy | proxy]`

### Parameters and flags

- **-a acc_map**
  
  Sets the asynchronous control character map in hexadecimal format. Default: 0xffffffff. (Neither a leading 0x nor a trailing `H` is required.)

- **-A**
  
  Denotes an outgoing dial-on-command link. Specify the remote system using the `-u` option.

- **-d debug_level**
  
  Sets the debugging level. Available values are:
  
  - **0**
    
    No debugging (the default)
  
  - **1**
    
    Log PAP/CHAP, LCP, and IPCP messages.
  
  - **2**
    
    Log all PPP messages.

- **-D device_name**
  
  Specifies the device to open for static (dedicated) links. Use the `-D` flag only in combination with the `-S` flag.

- **-f flow control**
  
  Sets the flow control level. Available values are:
  
  - **rtscts**
    
    Specifies flow control using hardware signals (the default).
  
  - **xonxoff**
    
    Specifies software flow control.
-noflow
  Specifies no flow control.

-i
  Turns off IP address negotiation. Default: IP address negotiation ON.

-k auth_key
  Specifies an authentication password key (used only when the remote end requires authentication and has no default value). Must be an ASCII string no longer than 64 bytes.

-m mru
  Specifies the MRU in decimal form. Default: 1500 bytes.

-p auth_prot
  Sets the authentication protocol. Possible values are PAP and CHAP. Default: no authentication protocol.

-r negn_rfc
  Specifies the IP address negotiation RFC. Possible values are 1172 and 1332. Default: RFC 1332.

-P pool name
  Specifies the pool name to be used. Pool name is a name that was defined using the pppadlist command.

-s speed
  Specifies the baud rate speed for the dedicated link. Any of the speeds defined in termios.h (such as 1200, 2400, 4800, 9600) are valid. Default: 9600 bps. Use this flag only in combination with the -S flag.

-S
  CAUTION: Use the -S flag for dedicated links only. If you use it with any other type of link, ppp terminates with an error message.

  Denotes a dedicated (static) link. If used in combination with the -D flag, the specified device_name is used. If -D is not specified, the standard input device is used for PPP communications.

-t
  Denotes an outgoing dial-on-demand link. This flag should not be used in combination with the -S flag; they are mutually exclusive.

-T tmout
  Specifies maximum number of timeouts without data allowed before closing the connection. Default: 0.

-u system
  Specifies the system to dial into. The -u flag is used in combination with -t or -A. Set up the system with the add system command.

-V
  Turns off Van Jacobsen TCP/IP header compression. Default: Van Jacobsen TCP/IP header compression is ON.

-mask
  Specifies the netmask in IP address format. Default: 255.255.255.0.

-[remote:]local
  Specifies the remote and local ends of the PPP connection. Each address must be a valid host name/IP address or 0 if the address is to be automatically negotiated by PPP.
Example

To start ppp from the shell on the current tty port:

[admin:1/38]>> ppp -m1500 -S -f noflow 199.171.31.199:199.171.31.82
255.255.255.0
pppstats

Syntax

pppstats

Description

The pppstats command retrieves and displays PPP interface statistics.

Sample Output

[admin:1/4]>>pppstats

ppp:
  0 outbound connection requests
  1 inbound connection requests
  1 connections established
  0 connections closed
  0 PAP authentication failures
  0 CHAP authentication failures
  65 packets sent
  2 received packets with bad FCS
  0 received packets with bad address field
  0 received packets with bad control field
  2 received packets with bad protocol field
  66 correct packets received
    2 packets with bad id field
    0 loopback packets
  0 state table errors

[admin:1/5]>>
printenv

Syntax
printenv

Description
The printenv command displays all currently set environment variables.

**NOTE:** Printenv is the same as setenv without any arguments.

Example

```
[admin:1/43]>>printenv
USER=nobody
TERM=xterm
```
processes

Syntax


Description

Use the processes command to display performance and processes related information. If no flags are specified, -t is assumed.

This command is designed for use by development and support personnel for diagnostic purposes.

Flags

<table>
<thead>
<tr>
<th>Flag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>c</td>
<td>Clears the process and I/O performance indicators</td>
</tr>
<tr>
<td>i</td>
<td>Displays idle statistics and reset</td>
</tr>
<tr>
<td>p</td>
<td>Displays performance indicators</td>
</tr>
<tr>
<td>s</td>
<td>Displays stack usage</td>
</tr>
<tr>
<td>sf</td>
<td>Displays stack usage of terminated tasks</td>
</tr>
<tr>
<td>t</td>
<td>Displays task table</td>
</tr>
<tr>
<td>u</td>
<td>Displays user/process table</td>
</tr>
</tbody>
</table>

Example

    [expert:1/34]>> processes

<table>
<thead>
<tr>
<th>pid</th>
<th>name</th>
<th>state</th>
<th>pri</th>
<th>message</th>
<th>mbox</th>
<th>sched</th>
<th>%ticks</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>prnull</td>
<td>ready</td>
<td>0</td>
<td>0xffffffff</td>
<td>0x00000000</td>
<td>88589</td>
<td>99.5</td>
</tr>
<tr>
<td>1</td>
<td>launcher</td>
<td>mbwait</td>
<td>50</td>
<td>0x00000000</td>
<td>0x081f6ebc</td>
<td>33</td>
<td>0.0</td>
</tr>
<tr>
<td>3</td>
<td>spawner</td>
<td>mbwait</td>
<td>50</td>
<td>0x00000000</td>
<td>0x081db2f4</td>
<td>305</td>
<td>0.0</td>
</tr>
<tr>
<td>4</td>
<td>swk_oc_thread</td>
<td>mbwait</td>
<td>89</td>
<td>0x00000000</td>
<td>0x081fccd4</td>
<td>2</td>
<td>0.0</td>
</tr>
<tr>
<td>5</td>
<td>sysdiagd</td>
<td>mbwait</td>
<td>74</td>
<td>0x00000000</td>
<td>0x081ddf8</td>
<td>11</td>
<td>0.0</td>
</tr>
<tr>
<td>11</td>
<td>launcher</td>
<td>mbwait</td>
<td>50</td>
<td>0x00000000</td>
<td>0x081dae58</td>
<td>3</td>
<td>0.0</td>
</tr>
<tr>
<td>47</td>
<td>swk_oc_thread</td>
<td>mbwait</td>
<td>89</td>
<td>0x00000000</td>
<td>0x081e4878</td>
<td>2</td>
<td>0.0</td>
</tr>
<tr>
<td>52</td>
<td>launcher</td>
<td>mbwait</td>
<td>50</td>
<td>0x00000000</td>
<td>0x081fa194</td>
<td>22</td>
<td>0.0</td>
</tr>
<tr>
<td>55</td>
<td>swk_oc_thread</td>
<td>mbwait</td>
<td>89</td>
<td>0x00000000</td>
<td>0x081ff6ec</td>
<td>2</td>
<td>0.0</td>
</tr>
<tr>
<td>60</td>
<td>init</td>
<td>mbwait</td>
<td>50</td>
<td>0x00000000</td>
<td>0x081d33a4</td>
<td>281</td>
<td>0.1</td>
</tr>
<tr>
<td>61</td>
<td>swk_main_task</td>
<td>mbwait</td>
<td>90</td>
<td>0x00000000</td>
<td>0x080fe8ac</td>
<td>96301</td>
<td>0.1</td>
</tr>
<tr>
<td>63</td>
<td>to_task</td>
<td>recv</td>
<td>100</td>
<td>0x00000000</td>
<td>0x00000000</td>
<td>19</td>
<td>0.0</td>
</tr>
</tbody>
</table>
[expert:1/35]>>processes -p

Performance Indicators for 8613 Seconds
TTY Input 11 bytes (0 bytes/second)
TTY Output 286 bytes (0 bytes/second)
HDLC Input 0 bytes (0 bytes/second)
HDLC Output 0 bytes (0 bytes/second)
ENET Input 1073756 bytes (124 bytes/second)
ENET Output 105557 bytes (12 bytes/second)
Printer Output 0 bytes (0 bytes/second)
TTY Overrun Errors 0
TTY Framing Errors 0
TTY Parity Errors 0
**prtr_test**

**Syntax**

```plaintext
prtr_test {device} [number_of_sets]
```

**Description**

The `prtr_test` command sends the specified number of sets of lines of output to the specified device. Each set is 10 lines. The device names are of the form:

```
/dev/tty/N
```

where N is the port number.

To test a serial printer connected to the first async port, use the command:

```
prtr_test /dev/tty/0 25
```

To test a parallel printer, enter:

```
[expert:1/38]>>prtr_test /dev/tty/32 100
```

The parallel port on an RCS/4000 is device `/dev/tty/32`. For an RCS/5000 or RCS/60000, it is device `/dev/tty/17`. 
quit

Syntax
  quit

Description
  Use the quit command to return from the expert level to the administrative level.
reboot

Syntax

reboot [show] [cancel] [# minutes to reboot]

Description

Use the reboot command to immediately reset the port server. The system will go through the normal power-on self-test and then return to normal operation.

If any modifications to the active configuration database were made since the last save, the reboot command will prompt for confirmation prior to rebooting the system. Any unsaved changes made to the active configuration database are lost upon reboot.

Any active user session is dropped.

If the port server encounters an error during the loading of the configuration information, it will use the factory default configuration instead of the current configuration database to complete the reboot. The LED will flash alternating yellow and green to indicate that the default configuration was used.

Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show</td>
<td>Shows whether the port server has been scheduled to reboot.</td>
</tr>
<tr>
<td>cancel</td>
<td>Cancels a scheduled reboot of the port server.</td>
</tr>
<tr>
<td># minutes to reboot</td>
<td>Specifies number of minutes before port server reboots.</td>
</tr>
<tr>
<td>no options</td>
<td>The port server reboots immediately.</td>
</tr>
</tbody>
</table>
rlogin

Syntax


Description

The `rlogin` command connects your terminal on the current local host system to the remote host system `hostname` and establishes a remote login session.

Flags

- **-ex**
  Replaces the default escape character with `x`. There is no space separating the `-e` option flag and the argument `x`.

- **-l username**
  Specify a different user name for the remote login. Defaults to the value of the USER environment variable which defaults to “nobody”.

- **-8**
  Allows an 8-bit input data path at all times; otherwise, parity bits are stripped except when the remote side's stop and start characters are other than CTRL-S/CTRL-Q.

- **-L**
  Allows the `rlogin` session to be run without any output post-processing (for example, `stty -opost`).

- **-T term**
  Specifies the terminal type which is to be passed to the remote `rlogin` service. Defaults to the value of the TERM environment variable, or network when none is specified.

The response you receive to executing `rlogin` will depend on the remote `rlogin` service. In some cases, you may be requested to enter a login (logname) and password or just a password. If you do not specify either a logname or username, and have not set the USER environment variable, you will most likely be prompted for a login and a password. If you specify either username or logname, you may still be prompted for a password.

Typically, when using `rlogin` to access a machine running the UNIX operating system, there are two situations in which you would not be required to give a password:

- When using `rlogin` to access an equivalent host. If machine A contains a file named `/etc/hosts.equiv` and machine B is listed in that file, machine B is an equivalent host to machine A and users on machine B may `rlogin` to machine A without giving a password. If usernames are paired with host names in the `/etc/hosts.equiv` file, then only username on machine B may `rlogin` to machine A without a password.

- When the username you specify and the local host are in a private equivalence list on `rhost` in the home directory of the logname you specify. The home directory of a logname (on machine A, for example) may contain a `.rhosts` file which contains a list of host names (including machine B, for example), with each host name optionally
paired with a username. If the listing for machine B in the .rhosts file has no paired
username, any user on machine B may rlogin to machine A as logname without being
required to give a password. If the listing for machine B is paired with a username,
only username may rlogin to machine A from machine B as logname without being
required to give a password.

Once logged in at the remote host system, all echoing takes place at the remote site so that
(except for delays) the rlogin is transparent. Flow control via Ctrl-S and Ctrl-Q and
flushing of input and output on interrupts are handled properly. A line of the form "~." disconnections from the remote host, where ~ is the escape character. A different escape
character may be specified with the -e option.
rtn

Rtn is a host utility that allows certain UNIX programs to access serial devices attached to port server ports. (See Host System Utilities page 263)
save

Syntax
save

Description
The save command saves all changes made to the active configuration into the current configuration database so that they will be in effect following a system reset.

If the port server is unable to load the current configuration database into the active configuration while booting, the factory default configuration database is loaded into the active configuration instead so that it can complete the boot. The LED on the front of the port server flashes alternately yellow and green to indicate that there was a problem and that the factory default configuration was loaded.

You can revert to the factory default configuration database at any time using the restore command in the firmware monitor. (see Appendix E, Firmware Monitor, page Error! Bookmark not defined.)

In admin or expert mode, the command prompt will indicate whether the active configuration has changed and needs to be saved. An asterisk (*) in the command prompt indicates that the configuration has changed. For example:

    [admin: 1/1 *]>>
 shows that the configuration has changed since the last save, as indicated by the asterisk (*). On the other hand,

    [admin: 1/1]>>
 shows that the configuration has not changed, since no asterisk (*) is displayed.
setenv

Syntax
  setenv [name [value]]

Description
The setenv command displays or sets environment variables:

- When used with no arguments, setenv displays the currently set environment variables.
- When used with the name argument but no value, it sets name to an empty (null) value.
- When used with both arguments, it binds name to the specified value.

Flags

<table>
<thead>
<tr>
<th>name</th>
<th>Specifies the name for an environment variable. By convention, the name should be in all upper-case letters.</th>
</tr>
</thead>
<tbody>
<tr>
<td>value</td>
<td>Specifies the value to bind to the name above. If it contains more than one word, the string must be in quotation marks.</td>
</tr>
</tbody>
</table>

Example

```
setenv TERM xterm
[admin:1/3]>>setenv
USER=nobody
TERM=xterm
```
**shell**

**Syntax**

```
shell [-l] [-n] [-N]
```

**Description**

**NOTE:** The **shell** command is only available for use when defining a custom configuration. (See [session](#) page [106](#))

The **shell** command defines a user session as a shell session, rather than a telnet or rlogin session. A shell session provides access to the command processor on the local port server, allowing the user access to the command capabilities of the port server. The user can be restricted to commands of a specified command level using the **capabilities level** option when defining the custom configuration.

**Flags**

- **-1** enables one-key command parsing
- **-n** enables "no restore" mode. When restore mode is enabled (default), the following termios parameters are restored after each command is executed by the shell.
  - **c_iflag:** BRKINT, ICRNL, IMAXBEL (set)
    - IGNBRK, PARMRK (cleared)
  - **c_oflag:** OPOST, ONLCR, TAB3 (set)
  - **c_lflag:** ISIG, ICANON, ECHO, ECHOE, ECHOCTL, ECHOKE, IEXTEN (set)
- **-N**
slip

Syntax

slip [+c | -c] [+e | -e] [+i | -i] [+m mtu | -m mtu] [+p | -p] [+v | -v] [-D devicename] [-s speed] [src-name] [dst-name]

Description

The slip command attaches serial lines as network interfaces, using the Serial Line Interface Protocol (SLIP).

NOTE: You can also configure a serial port to use SLIP for any data transmitted by defining and applying a custom service configuration (see SLIP, page 156).

Flags

<table>
<thead>
<tr>
<th>Flag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>+c</td>
<td>-c</td>
</tr>
<tr>
<td>+e</td>
<td>-e</td>
</tr>
<tr>
<td>+i</td>
<td>-i</td>
</tr>
<tr>
<td>+m mtu</td>
<td>-m mtu</td>
</tr>
</tbody>
</table>

NOTE: The mtu value for the TCP/IP packet header should be 40 plus some power of 2 (for example, 296 = 40 + 2^8).

<table>
<thead>
<tr>
<th>Flag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>+p</td>
<td>-p</td>
</tr>
<tr>
<td>+v</td>
<td>-v</td>
</tr>
<tr>
<td>-s</td>
<td></td>
</tr>
</tbody>
</table>
| [-D devicename] | Specifies the name of the serial line over which Serial Line IP (SLIP)
will run; if not specified, the current terminal is used.

**src-name** Specifies the IP address of the local end of the connection.

**dst-name** Specifies the IP address of the remote end of the connection.

**Example**

```
slip +e +m1500 200.200.200.1 200.200.200.2
slip +c -Dport4 +m1500 200.200.200.1 200.200.200.2
```
**stty**

**Syntax**

```bash
stty [-a | -e | -g] [options]
```

**Description**

The `stty` command sets or reports on terminal characteristics for the device that is its standard input. If no options or operands are specified, it reports the settings of a subset of characteristics as well as additional ones if they differ from their default values. Otherwise, it modifies the terminal state according to the specified arguments. Some combinations of arguments are mutually exclusive on some terminal types.

**Flags**

- **a**  Displays to standard output all current settings for the terminal.
- **e**  Displays to standard output in the traditional BSD all and everything formats all current settings for the terminal.
- **g**  Displays all current settings for the terminal to standard output in a form that may be used as an argument to a subsequent invocation of `stty` to restore the current terminal state.

**Options**

Sets terminal characteristics, as listed below.

**Control Modes**

Control mode flags affect hardware characteristics associated with the terminal. This corresponds to the `c_cflag` in the `termios` structure.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-] clocal</td>
<td>Assume a line without [with] modem control.</td>
</tr>
<tr>
<td>[-] cread</td>
<td>Enable [disable] the receiver.</td>
</tr>
<tr>
<td>[-] crtscts</td>
<td>Enable [disable] RTS/ CTS flow control</td>
</tr>
<tr>
<td>cs5  cs6  cs7  cs8</td>
<td>Select character size, if possible.</td>
</tr>
<tr>
<td>[-] cstopb</td>
<td>Use two [one] stop bits per character.</td>
</tr>
<tr>
<td>[-] hup</td>
<td>Same as [-] hupcl.</td>
</tr>
<tr>
<td>[-] hupcl</td>
<td>Stop [do not stop] asserting modem control on last close.</td>
</tr>
<tr>
<td>number</td>
<td>Set terminal baud rate to the number given, if possible. If the baud rate is set to 0, modem control is no longer asserted.</td>
</tr>
<tr>
<td>[-] parenb</td>
<td>Enable [disable] parity generation and detection.</td>
</tr>
</tbody>
</table>
Enable [disable] extended parity for mark and space purity.

Select odd [even] parity.

This sets both ispeed and ospeed to number. Using the option ispeed, if the input baud rate is set to 0, the input baud rate is set to the value of the output baud rate. Using ospeed to set the output baud rate to 0 causes modem control to no longer be asserted.

**Input Modes**

This corresponds to the c_iflag in the termios structure.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>brkint</td>
<td>Signal [do not signal] INTR on break.</td>
</tr>
<tr>
<td>icrnl</td>
<td>Map [do not map] CR to NL on input.</td>
</tr>
<tr>
<td>ignbrk</td>
<td>Ignore [do not ignore] break on input.</td>
</tr>
<tr>
<td>igncr</td>
<td>Ignore [do not ignore] CR on input.</td>
</tr>
<tr>
<td>ignpar</td>
<td>Ignore [do not ignore] parity errors.</td>
</tr>
<tr>
<td>imaxbel</td>
<td>The system imposes a limit of MAX_INPUT (currently 255) characters in the input queue. If imaxbel is set and the input queue limit has been reached, subsequent input causes the system to send an ASCII BEL character to the output queue (the terminal beeps at you). Otherwise, if imaxbel is unset and the input queue is full, the next input character causes the entire input and output queues to be discarded.</td>
</tr>
<tr>
<td>inclr</td>
<td>Map [do not map] NL to CR on input.</td>
</tr>
<tr>
<td>inpck</td>
<td>Enable [disable] input parity checking.</td>
</tr>
<tr>
<td>istrip</td>
<td>Strip [do not strip] input characters to 7 bits.</td>
</tr>
<tr>
<td>iuclc</td>
<td>Map [do not map] uppercase to lowercase on input.</td>
</tr>
<tr>
<td>ixany</td>
<td>Allow any character [allow only START] to restart output.</td>
</tr>
<tr>
<td>ixoff</td>
<td>Request that the system send [not send] START/STOP characters when the input queue is nearly empty/full.</td>
</tr>
<tr>
<td>ixon</td>
<td>Enable [disable] START/STOP output control. Output from the system is stopped when the system receives STOP and started when the system receives START; or if ixany is set, any character restarts output.</td>
</tr>
<tr>
<td>parmrk</td>
<td>Mark [do not mark] parity errors.</td>
</tr>
</tbody>
</table>

**Output Modes**

This corresponds to the c_oflag of the termios structure:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ocrnl</td>
<td>Map [do not map] CR to NL on output.</td>
</tr>
<tr>
<td>ofdel</td>
<td>Uses DEL [NUL] as the fill character used by ofill.</td>
</tr>
<tr>
<td>ofill</td>
<td>Transmits [does not transmit] fill characters to delay during mechanical or...</td>
</tr>
</tbody>
</table>
other movement when certain characters such as line feeds and vertical tabs are sent to the terminal. With "-" uses timed delay instead.

[-] olcuc  Map [do not map] lowercase to uppercase on output.
[-] onlcr  Map [do not map] NL to CR-NL on output.
[-] onlret  Maps NL to carriage return, moves pointer to column 0, and uses delays associated with CR [NL functions as line feed only and the column position remains unchanged].

**NOTE:** The column pointer is also set to 0 if the CR character is actually transmitted.

[-] onocr  Transmit [do not transmit] CR when at column 0 (initial position).
[-] opost  Post-process output [do not post-process output; ignore all other output modes].

**Local Modes**

Local mode flags (lflags) affect various characteristics of terminal processing.

[-] echo  Echo back [do not echo back] every character typed.
[-] echoctl  If echoctl is set, echo control characters as ^X. Otherwise, control characters echo as themselves.
[-] echoe  The ERASE character will, if possible [will not] visually erase the last character in the current line from the display.
[-] echok  Echo [do not echo] NL after KILL character.
[-] echoke  The KILL character will, if possible [will not] visually erase the current line from the display.
[-] echonl  Even if echo is disabled, echo [do not echo] NL.
[-] echoprt  For printing terminals; echo erased characters backwards with "\" and "/" [disable this feature].
[-] flusho  Indicates output is [is not] being discarded.
[-] icanon  Enable [disable] canonical input (ERASE and KILL processing).
[-] iexten  Enable [disable] any implementation-defined special control characters not currently controlled by icanon, isig, orixon.
[-] isig  Enable [disable] the checking of characters against special control characters INTR, QUIT, and SUSP.
[-] noflsh  Disable [enable] flush after INTR, QUIT, SUSP.
[-] pendin  Indicates input is [is not] pending after a switch from noncanonical to canonical mode and will be re-input when a read becomes pending or more input arrives.
[-] **to stop**  Send [do not send] SIGTTOU for background output. This causes background jobs to stop if they attempt terminal output.

[-] **xcase**  If both xcase and iconon are specified without the "-," an uppercase letter is accepted on input by preceding it with a "\" character, and is output the same way. In this mode, the following escape characters are accepted on input and created on output:

- `' `\`
- `| `\!
- `~ `\^`
- `{ `\`
- `}` `\)
- `\ `\`

For example, A is input as `\a`, `\n` is input as `\n`, and `\N` as `\\n`.

---

**Control Characters**

```
control-character string
```

Set control-character to string. If string is a single character, the control character is set to that character. If string is the two-character sequence "^~" or the string "undef," the control character is disabled (set to `_[POSIX_VDISABLE]`). Recognized control characters include eof, eol, eol12, erase, werase, intr, kill, quit, susp, start, stop, dsusp, discard, lnext, reprint, and switch.

---

**Combination Modes**

```
columns number
```

The terminal size is recorded as having *number* columns (cols number is an alias for columns).

[-] **crt**  Set [disable] all modes suitable for a CRT display device.

**dec**  Set modes suitable for users of Digital Equipment Corporation systems (ERASE, KILL, and INTR characters are set to ^?, ^U, and ^C, respectively; ixany is disabled, and crt is enabled).

**evenp or parity**  Enable parenb and cs7; disable parodd.

**everything**  Displays to standard output all current settings for the terminal (same as stty -s).

[-] **nl**  Enable [disable] icrnl. In addition, -nl unsets inlcr and ignor.

**oddp**  Enable parenb, cs7, and parodd.

**-parity, -evenp, -oddp**  Disable parenb and set cs8.
Operational Commands 241

[-] raw
If set, change the modes of the terminal so that no input or output processing is performed. If unset, change the modes of the terminal to some reasonable state that performs input and output processing. Since the terminal driver no longer has a single RAW bit, it is not possible to intuit what flags were set prior to setting raw. This means that setting raw may not restore all the settings that were previously in effect.

rows number
The terminal size is recorded as having number rows.
sane
Resets all modes to reasonable values for interactive terminal use.
saved setting
Set the current terminal characteristics to the saved settings produced by the -g option.
size
The size of the terminal is printed as two numbers on a single line, first rows, then columns.

Compatibility Modes
These modes remain for compatibility with the previous version of the stty command:
all
Reports all the terminal modes as with stty -a except that the control characters are printed in a columnar format.
brk value
Same as the control character eol.
cbreak
If set, enables brkint, ixon, imaxbel, opost, isig, iexten, and icanon. If unset, same as sane.
Cooked
Same as sane.
[-] crtbs
Same as echoe.
[-] crterase
Same as echoe.
[-] crtkill
Same as echoke.
[-] ctlecho
Same as echoctl.
[-] decctlq
The converse of ixany.
everything
Same as all.
flush value
Same as the control character discard.
[-] litout
The converse of opost.
[·] newcrt  
    Same as crt.

pass8  
The converse of parity.

[·] prterase  
    Same as echopr.

rprnt value  
    Same as the control character reprint.
syslog

Syntax

    syslog display [number of lines | cont]
    syslog msg [message_text]
    syslog config [show]

Description

The syslog facility provides a means for port server software to log errors to a local message buffer or to send log messages to a designated syslog server on the network. Syslog is used to report software oriented errors, configuration problems, command failures, and so forth.

To display the system log, type:

    syslog display

You can display the last 10 lines of the system log by typing:

    syslog display 10

You can display the system log in “continuous” mode, so that each new system log entry is displayed immediately, by typing:

    syslog display cont

To end “continuous” mode and return to the command prompt, type CTRL-C.

To write messages to the log, type:

    syslog msg this is the message

If you do not specify a message, a default message will be written to the log. The default is: ***This is the default message***.

You can configure the syslog by specifying the size of the circular log buffer (default is 50 entries) and, optionally, by specifying a remote host that is to receive any log messages.

To send log messages to a designated syslog server on the network, type:

    syslog config

to specify the hostname or IP, whether to use a local buffer, and number of entries.

To show the correct syslog configuration, type:

    syslog config show
Example

[expert:1/2]>> syslog config
Specify a remote loghost: asdf
Do you wish to log messages to the local circular buffer? y
How many entries would you like in the circular buffer? (Max=3000) 100
You have specified the following configuration:
Logging to host: asdf
Logging to circular buffer with 100 entries.
Is this correct? y
The syslog configuration has been updated.

Messages will default to asdf:/var/adm/messages.

Example

[expert:1/2]>> syslog msg This is my message
[expert:1/3]>> syslog display
<46>Feb 13 16:22:04 syslogd: restart
<38>Feb 13 16:22:04 pppd: open (ppcid) of /dev/ppp succeeded
<38>Feb 13 16:22:05 pppd: open PPP_syslog) of /dev/ppp succeeded
<38>Feb 13 16:22:05 pppd: restarted
<135>Feb 13 16:22:25 telnetd: telnetd begin
<135>Feb 13 16:22:57 : This is my message
[expert:1/4]>>
telnet

Syntax


Description

The telnet command is used to communicate with another host using the TELNET protocol. If telnet is invoked without the host argument, it will enter command mode as indicated by its prompt, telnet>. In command mode, telnet will accept and execute the commands listed below. If telnet is invoked with arguments, it will perform an open command with those arguments. For example:

telnet zeus

will open a telnet session to a host named zeus.

The source TCP/IP port number for a telnet connection made from a shell or configuration on an port server serial port has the form:

1ppss

where

- pp is the serial port that is running the telnet command
- ss is the session ID. The session ID is 0 if the port is not configured for multi-sessions or 0-3 if the port is configured for multi-sessions.

Example 1

Running the default configuration on port 1, you log into the port server and telnet to "remotehost". The source TCP/IP port number in the telnet connection will be 10100, unless it is already in use.

Example 2

Running a custom configuration with multiple sessions on port 2, the second session is a command to telnet to "remotehost". The source TCP/IP port number for the telnet session will be 10201, unless it is already in use.

Flags

-8

Uses an 8-bit data path and attempts to negotiate the BINARY option on both input and output.

-E

Stops any character from being recognized as an escape character.

-e char

Sets the initial TELNET escape character to escape_char.
-f  Forces the use of a constant port number, formed as described above. If the port number is in use, the connection is not made and an error is reported.

-L  Uses an 8-bit data path on output and negotiates the BINARY option on output.

-R  Force remote echo.

-r  Uses a user interface similar to rlogin. In this mode, the escape character is set to the "~" character, unless modified by the -e flag.

-q  Quiet mode.

-p  Don't push ldterm onto tty stream.

-t  Ignores telnet escape sequences

hostname  Indicates the host's official name—an alias or the Internet address of a remote host.

port  Indicates a port number (the address of an application). If a number is not specified, the default TELNET port is used.

Once connection has been opened, TELNET enters the input mode. TELNET will attempt to enable the TELNET LINEMODE option. If this fails, then TELNET will revert to one of two input modes: either the character-at-a-time mode or the old-line-by-line mode, depending on what the remote system supports.

When LINEMODE is enabled, character processing will be done on the local system while under the control of the remote system. When input editing or character echoing is to be disabled, the remote system will relay that information. The remote system will also relay changes that occur to any special characters on the remote system so they can take effect on the local system.

In the character-at-a-time mode, most entered text will be sent immediately to the remote host for processing.

In the old-line-by-line mode, all text will be echoed locally, but (normally) only completed lines will be sent to the remote host. The "local echo character" (initially [Ctrl-E]) may be used to enable and disable the local echo mode; normally this would be used only for entering passwords so that the password is not echoed.

If the LINEMODE option is enabled or if the localchars toggle is TRUE (the default value for the old-line-by-line mode; see below), the user's quit, intr, and flush characters will be trapped locally and sent as TELNET protocol sequences to the remote machine. If LINEMODE had been enabled at any earlier time, the user's susp and eof characters will also be sent as TELNET protocol sequences; quit will be sent as a TELNET\ABORT instead of BREAK. There are options (toggle autoflush and toggle autosync) that cause
this action to flush any subsequent output to the terminal (until the remote host acknowledges the TELNET sequence) and to flush previous terminal input (in the case of quit and intr).

While connected to a remote host, the telnet command mode may be entered by typing the TELNET escape character (initially "\[").

When in command mode, the normal terminal editing conventions are available.

The current terminal type configured on the port is sent to the host when you telnet to a host from a shell. The default is VT100. The terminal type can be changed by enabling a different service configuration on the port (see User-defined Custom Service Configuration, page 82).

TELNET Commands

The TELNET commands listed below are available. You need only type enough of each command to uniquely identify it (this is also true for arguments pertaining to the mode, set, toggle, unset, slc, environ, and display commands). You can type a telnet command when the telnet prompt (telnet>) is displayed.

Example

Before you can transfer files during a telnet session using file transfer software such as KERMIT or XMODEM, you need to set telnet to binary transfer. To set telnet to binary, use the following telnet commands.

1. To open the telnet connection with the host, zeus, at the port server prompt, enter:
   
   telnet zeus

2. After successfully logging onto zeus, enter telnet command mode (escape to telnet) by typing:
   
   Control-[

3. At the telnet prompt (telnet>), enter:
   
   set binary

  close

  Closes the TELNET session and returns to command mode.

  display argument...

  Displays some or all of the set and toggle values.

  logout

  Forcibly logs out the remote user and closes the connection.

  mode [type]

  Depending on the state of the TELNET session, the type argument is one of several available options. The remote host will be asked for permission to go into the requested mode. If the remote host is capable of entering that mode, the requested mode will be entered. The following type options are available for the telnet mode command

---
character
Disables the TELNET\ LINEMODE option, or if the remote side does not understand the option, enters the character-at-time mode.

Line
Enables the TELNET\ LINEMODE option, or if the remote side does not understand the option, attempts to enter the old-line-by-line mode.

[ - ] edit
Attempts to enable [ disable] the EDIT mode of the LINEMODE option. This requires that the LINEMODE option be enabled.

[ - ] softtabs
Attempts to enable [ disable] the SOFT_TAB mode of the LINEMODE option. This requires that the LINEMODE option be enabled.

[ - ] litecho
Attempts to enable [ disable] the LIT_ECHO mode of the LINEMODE option. This requires that the LINEMODE option be enabled.

?
Prints out help information for the mode command.

open host [ port]
Open a connection to the named host. If no port number is specified, telnet will attempt to contact a TELNET server at the default port. The host specification may be either a host name or an Internet address specified in dot notation. When connecting to a non-standard port, telnet will omit the automatic initiation of any TELNET options.

quit
Close any open TELNET session and exit telnet. When in command mode, and End-of-File (EOF) will also close a session and exit.

send arguments
Sends one or more special character sequences to the remote host. Following are the arguments which may be specified (more than one argument may be specified at a given time):

abort
Sends the TELNET\ ABORT (Abort processes) sequence.

ao
Sends the TELNET\ AO (Abort Output) sequence which should cause the remote system to flush all output from the remote system to the user's terminal.

ayt
Sends the TELNET\ AYT ("Are You There") sequence; the remote system may or may not choose to respond to this transmission.

brk
Sends the TELNET\ BRK (Break) sequence which may have significance to the remote system.

ec
Sends the TELNET\ EC (Erase Character) sequence, which should cause the remote system to erase the last character entered.
el  Sends the TELNET\ EL (Erase Line) sequence, which should cause the remote system to erase the line currently being entered.

eof  Sends the TELNET\ EOF (End of File) sequence.

eor  Sends the TELNET\ EOR (End of Record) sequence.

escape  Sends the current TELNET escape character (initially "^]").

ga  Sends the TELNET\ GA (Go Ahead) sequence, which probably has no significance to the remote system.

getstatus  If the remote side supports the TELNET\ STATUS command, getstatus will send a subnegotiation request that the server send its current option status.

Ip  Sends the TELNET\ IP (Interrupt Process) sequence, which should cause the remote system to abort the currently running process.

nop  Sends the TELNET\ NOP (No Operation) sequence.

susp  Sends the TELNET\ SUSP (Suspend Process) sequence).

synch  Sends the TELNET\ SYNCH sequence. This sequence causes the remote system to discard all previously typed (but not yet read) input. This sequence will be sent as TCP urgent data (and may not work if the remote system is a 4.2 BSD system; if it does not work, a lower-case "$" may be echoed on the terminal).

?  Prints out help information for the send command.

set argument value
unset argument value
The set command will set any one of a number of TELNET variables to a specific value or to TRUE. The special value, off, will turn off the function associated with this variable; this is equivalent to using the unset command. The unset command will disable (or set to FALSE) any of the specified functions. The values of variables may be interrogated with the aid of the display command. The variables which may be set or unset (but not toggled) are listed here. In addition, any of the variables for the toggle command may be explicitly enabled or disabled using the set and unset commands. Arguments for the telnet set/ unset commands are:

echo  This is the value (initially "^]") which, when in the old-line-by-line mode, will toggle between doing local echoing of entered characters (for normal processing) and suppressing echoing of entered characters (for example, for entering a password).
eof
If telnet is operating in LINEMODE or in the old-line-by-line mode, entering this character as the first character on a line will cause this character to be sent to the remote system. The initial value of the eof character is taken to be the terminal's eof character.

erase
If telnet is in localchars mode (see toggle localchars), and if telnet is operating in the character-at-a-time mode, then when this character is entered, a TELNET\ EC sequence (see send ec) will be sent to the remote system. The initial value for the erase character is taken to be the terminal's erase character.

escape
This is the TELNET escape character (initially" ^\") which causes entry into the TELNET command mode when connected to a remote system.

flushoutput
If telnet is in localchars mode (see toggle localchars) and the flushoutput character is entered, a TELNET\ AO sequence (see send ao) will be sent to the remote host. The initial value for the flush character is taken to be the terminal's flush character.

interrupt
If telnet is in localchars mode (see toggle localchars) and the interrupt character is entered, a TELNET\ IP sequence (see send ip) will be sent to the remote host. The initial value for the interrupt character is taken to be the terminal's intr character.

kill
If telnet is in localchars mode (see toggle localchars), and if telnet is operating in the character-at-a-time mode, then when this character is entered, a TELNET\ EL sequence (see send el) will be sent to the remote system. The initial value for the kill character is taken to be the terminal's kill character.

lnext
If telnet is operating in LINEMODE or in the old-line-by-line mode, this character is taken to be the terminal's lnext character. The initial value for the lnext character is taken to be the terminal's lnext character.

quit
If telnet is in localchars mode (see toggle localchars) and the quit character is entered, a TELNET\ BRK sequence (see send brk) will be sent to the remote host. The initial value for the quit character is taken to be the terminal's quit character.

reprint
If telnet is operating in LINEMODE or in the old-line-by-line mode, then this character is taken to be the terminal's reprint character. The initial value for the reprint character is taken to be the terminal's reprint character.
start
If the TELNET\ TOGGLE-FLOW-CONTROL option has been enabled, then this character is taken to be the terminal's start character. The initial value for the start character is taken to be the terminal's start character.

stop
If the TELNET\ TOGGLE-FLOW-CONTROL option has been enabled, then this character is taken to be the terminal's stop character. The initial value for the stop character is taken to be the terminal's stop character.

susp
If telnet is in localchars mode or if LINEMODE is enabled and the suspend character is entered, a TELNET\ SUSP sequence (see send susp) will be sent to the remote host. The initial value for the suspend character is taken to be the terminal's suspend character.

worderase
If telnet is operating in LINEMODE or in the old-line-by-line mode, then this character is taken to be the terminal's worderase character. The initial value for the worderase character is taken to be the terminal's worderase character.

? Displays the legal set and unset commands.

slc [state]
The slc command (Set Local Characters) is used to set (or change) the state of the special characters when the TELNET\ LINEMODE option has been enabled. The special characters are characters that get mapped to TELNET command sequences (such as ip or quit) or line-editing characters (such as erase and kill). By default, the local special characters are exported.
Arguments for the telnet slc command are:

export
Switches to the local defaults for the special characters. The local default characters are those of the local terminal at the time when telnet was started.

import
Switches to the remote defaults for the special characters. The remote default characters are those of the remote system at the time when the TELNET connection was established.

check
Verifies the current settings for the current special characters. The remote side is requested to send all the current special character settings; if there are any discrepancies with the local side, the local side will switch to the set of remote values.

? Prints out help information for the slc command.
**toggle arguments [...]**

Toggle various flags (between TRUE and FALSE) that control how TELNET responds to events. These flags may be set explicitly to TRUE or FALSE using the set and unset commands listed above. More than one argument may be specified. The state of these flags may be interrogated with the aid of the display command. The valid arguments for this command are:

**autoflush**
If autoflush and localchars are both TRUE, then when the ao or the quit characters are recognized (and transformed into TELNET sequences; see set command) TELNET will refuse to display any data on the user's terminal until the remote system acknowledges (via a TELNET TIMING MARK option) that it has processed those TELNET sequences. The initial value for this toggle is TRUE if the terminal user had not executed an stty noflsh; otherwise FALSE.

**autosynch**
If autosynch and localchars are both TRUE, then when either the intr or quit character is entered (see set command for descriptions of the intr and quit characters), the resulting TELNET sequence sent will be followed by the TELNET SYNCH sequence. This procedure should cause the system to begin throwing away all previously entered input until both of the TELNET sequences have been read and acted upon. The initial value of this toggle is FALSE.

**binary**
Enables or disables the TELNET BINARY option on both the input and output (see inbinary and outbinary).

**crlf**
If this toggle value is TRUE, carriage returns will be sent as <CR><LS>. If FALSE, carriage returns will be sent as <CR><NUL>. The initial value for this toggle is FALSE.

**crmod**
Toggles the carriage return mode. When this mode is enabled, most carriage return characters received from the remote host will be mapped into a carriage return followed by a line feed. This mode does not affect those characters entered by the user, but only those received from the remote host. This mode is not very useful unless the remote host only sends carriage return, but never any line feeds. The initial value for this toggle is FALSE.

**inbinary**
Enables or disables the TELNET BINARY option on input.
**localchars**
If this it TRUE, the flush, interrupt, quit, erase, and kill characters (see set command) are recognized locally and then transformed into appropriate TELNET control sequences (respectively ao, ip, brk, ec, and el; see send command). The initial value for this toggle is TRUE in old-line-by-line mode and FALSE in character-at-a-time mode. When the LINEMODE option is enabled, the value of localchars is ignored and assumed to always be TRUE. If LINEMODE has ever been enabled, then quit will be sent as abort; eof and suspend will be sent as eof and susp (see the send command).

**netdata**
Toggles the display of all network data (in hexadecimal format). The initial value for this toggle is FALSE.

**options**
Toggles the display of some internal telnet protocol processing which pertain to TELNET options. The initial value for this toggle is FALSE.

**outbinary**
Enables or disables the TELNET BINARY option on output.

**prettydump**
When the netdata toggle is enabled and if prettydump is enabled, the output from the netdata command will be reorganized into a more user-friendly format. Spaces will be put between each character in the output and the beginning of any TELNET escape sequence will be preceded by a "*" to aid in locating them.

**?**
Displays the legal toggle commands.

**status**
Shows the current status of telnet. This includes the peer to which one is connected, as well as the current mode.

**unset**
See set command.

**? [command]**
Gets help. When no command is specified, telnet prints a summary for the help command. If a command is specified, telnet prints the help information for just that command.

**NOTE:** On some remote systems, the echo command has to be turned off manually when in the old-line-by-line mode. When in the old-line-by-line mode or in LINEMODE, the terminal’s eof character is only recognized (and sent to the remote system) when it is the first character in a line.
term

Syntax

term [termtype]

Description

The `term` command sets the local terminal type variable for this port. This local variable is passed to the host during telnet or rlogin sessions.
trace_clear

Syntax

    trace_clear

Description

Use the trace_clear command to clear the trace buffer.

This command is designed for use by development and support personnel for diagnostic purposes.
trace_display

Syntax

    trace_display [-d] [-llines] [+/-time] [-ttask_id] [facility]

Description

Use the trace_display command to display the trace buffer.
This command is designed for use by development and support personnel for diagnostic purposes.

Flags

-d
    Displays time between entries

-llines
    Number of lines to be displayed at once

+time
    Starting time offset from beginning for display

-time
    Starting time offset from end for display

-ttask_id
    Task from task table (see the processes command, page 224)

facility
    Module you are tracing (see the trace_on command, page 258)
trace_off

Syntax

    trace_off [facility]

See the trace_on command for a list of facility options.

Description

Use the trace_off command to turn off software tracing.

This command is designed for use by development and support personnel for diagnostic purposes.
trace_on

Syntax

    trace_on [facility]

Description

Use the trace_on command to turn on software tracing for named modules (facilities).
This command should be used at the direction of Customer Support.
This command is designed for use by development and support personnel for diagnostic
purposes.

Facility options

    all
    async
    cmnerr
    debug1
    debug2
    debug3
    debug4
    ethernet
    interrupt
    pppd
    pppm
    signal_handler
    snmp
    swk_call
    swk_intr
    tcp/ip
    time
    wan
    xinu

Examples

    [admin:1/3]>>trace_on wan
    Enables the collection of trace data relating to the WAN port.

    [admin:1/3]>>trace_on
    Displays current settings.
tty

Syntax

tty

Description
The tty command indicates on which port the user is currently running. May only be
done on console port or any of the ports if directly connected. Will not show if telnet’d in.
unsetenv

Syntax

    unsetenv {name}

Description

The unsetenv command unsets an environment variable by removing its binding from the environment. Remember that all sessions on a line share a single environment.

Flags

name

    Specifies the environment variable to be unbound.
upgrade

Syntax

    upgrade [key_a] [key_b]

Description

Use the upgrade command to enable optional features of the port server, including:

- TTY interface
- WAN interface
- LAN interface.

To see which options are currently enabled, use the `show version` command.

NOTE: All features are enabled on the existing port server models, so there is no need to enter an upgrade command.

Flags

key_a and key_b are 5 character keys provided to upgrade the unit. Note that a key is unique to a given port server. If no parameters are given then the keys are requested interactively.

Example

    [expert:1/5]>> upgrade a3bze qr9be
who

Syntax

who

Description

The who command shows all current user and application activity. Output is generated showing the process ids, the process/application names that are active, and the physical port or IP address of the active connection.

Example:

<table>
<thead>
<tr>
<th>pid</th>
<th>ppid</th>
<th>pgrp</th>
<th>process_name</th>
<th>device(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>69</td>
<td>12</td>
<td>launcher</td>
<td>port0</td>
</tr>
<tr>
<td>7</td>
<td>12</td>
<td>12</td>
<td>shell</td>
<td>port0</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>12</td>
<td>who</td>
<td>port0</td>
</tr>
<tr>
<td>27</td>
<td>69</td>
<td>27</td>
<td>telnetd</td>
<td>(199.171.148.12)</td>
</tr>
<tr>
<td>21</td>
<td>27</td>
<td>21</td>
<td>launcher</td>
<td>(199.171.148.12)</td>
</tr>
<tr>
<td>15</td>
<td>21</td>
<td>21</td>
<td>shell</td>
<td>(199.171.148.12)</td>
</tr>
</tbody>
</table>

This example shows two users logged in. One user is logged in to the shell on port 0 and is currently executing the who command. (You might see an x in front of port, e.g., xport2, which means that port 2 was opened in exclusive mode.) The other user is connected to the port server via telnet from host 199.171.148.12 and is logged into the shell.
Three utilities that can be installed and used on the host systems are available to provide host access to the port server and its ports: the rtn utility that allows UNIX software to access serial devices attached to port server ports; the r4000 utility that allows the management of port server flash memory from the host system; and the dtelnetd utility that allows the port server to initiate telnet sessions with a specified PTY on a host.

The rtn Utility

The **rtn** (rtelnet) utility is a modified version of the Berkeley `telnetd` program. It creates a pseudo-tty (pty) device on the host that is connected via TELNET or raw TCP to an port server serial or parallel port. This allows UNIX programs such as `tip(1C)` and `uucp(1C)` to access serial devices attached to port server ports. **Rtn** works like a telnet daemon program (`telnetd`) except in a reverse fashion; instead of accepting connections from other network machines, **rtn** makes connections to the port server. **Rtn** creates a new file-like device (usually in `/dev`) that programs can open and use like a normal /dev/tty terminal.

**Syntax**

```
 rtn.{osname} {-h ip_address} {-d device_name} {-p tcp_port_number} [-v] [-o [-s]] [-u uid] [-f] [-D level {-f filename}]
```

Where `osname` is the extension given the rtn file for your operating system, e.g., `rtn.aix` or `rtn.sco` (see table of operation system file names below).

**Flags**

- `-d`
  
  Specifies the device name to use for the pty. Can be any valid path and filename, such as `/dev/port1`.

- `-h`
  
  Specifies the host name or the IP address of the port server to which the rtn connects.

- `-p`
  
  Specifies the tcp port on the port server to which the rtn connects.
-o  Causes rtn to open the master pty, slave pty (unless -s is used) and the network connection as soon as rtn is started. Useful if detection of opens and closes is not necessary. If the slave pty is closed and then immediately reopened, the second open will fail. In that case, use both the -o option and the -s option.

-s  Prevents the rtn from opening the slave side of the pty. Useful if detection of opens and closes is necessary. Must be used in conjunction with the -o option.

-u  Specifies the UID. Useful if ptys are used with tip or cu.

-F  Causes rtn to stay in the foreground. Use to override the default of rtn in the background.

-v  Displays version information.

-D  Turns on debugging. Use this option in combination with the -f option. Specify level of debugging to be recorded as follows:

   1  Fatal errors
   2  Recoverable errors
   3  Warning messages
   4  General informational and status messages
   5  Initialization data.

-f  Specifies the filename to which debugging information is recorded.

rtn default behavior

If you use only the required rtn options (-d, -h and -p), rtn operates as follows:
1. The master pty opens immediately.
2. The slave side of the pty does not open.
3. The network connection is made only after the slave pty is opened by a process.
4. When the slave side of the pty is closed, the network connection is lost. It is reestablished when the slave side of the pty is reopened.
Installing rtn

The **rtn** utility is provided in tar format on the CD which accompanied the port server. Extract the particular **rtn** executable for your system onto your system disk. For example, for a Solaris Intel system, extract **rtn.soli** onto your disk.

The port server supports at least the following operating systems:

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Utility Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIX 4.1</td>
<td>rtn.aix</td>
</tr>
<tr>
<td>Digital UNIX 4.0</td>
<td>rtn.osf</td>
</tr>
<tr>
<td>DGUX 3.10/ 3.11/ 4.11</td>
<td>rtn.dgux</td>
</tr>
<tr>
<td>HP UNIX</td>
<td>rtn.hpux</td>
</tr>
<tr>
<td>Motorola R40V4.4</td>
<td>rtn.mot</td>
</tr>
<tr>
<td>Motorola R32V3</td>
<td>rtn.r32</td>
</tr>
<tr>
<td>NCR MP-RAS 3.01.00</td>
<td>rtn.ncr</td>
</tr>
<tr>
<td>SCO OpenServer Release 5</td>
<td>rtn.sco</td>
</tr>
<tr>
<td>Solaris Intel 2.1/ 2.3/ 2.4/ 2.5</td>
<td>rtn.soli</td>
</tr>
<tr>
<td>Solaris Sparc 2.3 (SunOS 5.3)</td>
<td>rtn.sols</td>
</tr>
<tr>
<td>SunOS 4.1.3</td>
<td>rtn.sun</td>
</tr>
<tr>
<td>UnixWare 2.0</td>
<td>rtn.uw</td>
</tr>
</tbody>
</table>

If you are installing **rtn** to work with the AIX System Management Interface Tool (SMIT), use the AIX4 SMIT Interface on disk in SMIT installp format.

**Before Setting up a Device on a Serial Port**

When setting up a terminal or a printer on a serial port, the configuration for the port should be set to rtelnet. To determine if there is a default configuration, use the port server admin command:

```
show port portnumber
```

where `portnumber` is the number of the serial port on which the device is being set up. In this example, the device is being set up on port 0. The show command is:

```
show port 0
```

The current settings will be displayed as follows:

```plaintext
Port        : port0
Baud Rate   : 9600
Inactivity Timeout : 6000
Character Size  : 8
Stopbits     : 1
```
The rtn Utility

Parity : none
Switch character : <undef>
Ignore dcd : no
Input Processing : xoff
Output Processing : xon
Modem Name : generic
Configuration(s) : rtelnet

If the setting for configuration does not show **rtelnet**, use the arrow keys to move to the entry field for configuration and change the entry to **rtelnet**.

**Sample configuration with dial-out modem**

In this example, a dial-out modem is attached to port 3 of an port server with an IP address of 123.56.78.90:

1. Using the port server’s **admin** command set, configure port 3 as an rtn-accessible port using the **set rtelnet** command.
2. Start up the rtn program to create the pseudo-device:
   
   ```
   rtn -h 123.56.78.90 -d /dev/port3 -p 9003
   ```
3. To enable use of the modem via tip on a BSD host, create a new system definition in **/etc/remote**, specifying **/dev/port3** as the tip (dv) device. For example:

   ```
   port3: 
   :dv=/dev/port3:br#9600:pa=none:
   :el="C"S"Q"D:ie=%$:oe="D
   ```

   To enable the modem for cu connections, create an entry for the new pseudo device in the **/usr/lib/uucp/Devices** file. For example:

   ```
   Direct port3 - 9600 direct
   ```

**Sample configuration with printer port**

In this example, a parallel printer is attached to the printer port (port 32) of an RCS/4000 port server with an IP address of 123.56.78.90:

1. Using the port server’s **admin** command set, configure port 32 as an rtn-accessible port using the **set rtelnet** command.
2. Start up the rtn program to create the pseudo-device:

   ```
   rtn -h 123.56.78.90 -d /dev/port32 -p 9032 -o
   ```
3. To send a file directly to the printer:

   ```
   cat my_file > /dev/port32
   ```

   For an RCS/5000 or RCS/6000 port server, the printer port is port 17.

   For more information on setting up the port server side of the connection, see the **RTELNET** command, page 153

   For more information on using rtn for various applications, see the README file on the distribution media.
Raw reverse telnet

The raw reverse telnet feature allows programs to send and receive data through port server ports without using special reverse telnet programs, pseudo-ttys, or port redirector programs. A program need only open a TCP session to a specified protocol port number on the port server. Bytes written into the TCP endpoint will be transmitted, unchanged, on the corresponding port server serial port. Data received on the port server serial port will appear on the TCP endpoint unchanged. There is no intervening protocol nor processing. Raw reverse telnet is automatically enabled when you setup the rtelnet processing on a serial port or modem/printer pool.

Some sample programs are available through anonymous ftp at our WWW or FTP site or by calling customer support. The programs may be used to demonstrate and exercise the raw reverse telnet feature with a loopback cable connected to two port server serial ports. They also serve as examples implementing other programs to use raw reverse telnet. See the README file included with the sample programs for further details.

AIX

The following sections provide instructions and examples for setting up a terminal or a printer for AIX.

Setting up a Terminal

To setup a tty device (terminal) for login on AIX, you can use the `smit` program or you can enter a `mkdev` command on the command line.

smit

1. Type: `smit tty`
2. Select `Add a TTY`
3. Select: `tty rs232 Asynchronous Terminal`
4. Select: `port server0 Available port server0 port server/400 Terminal Server (16-Port)`
5. The following display will show the current settings. To change any settings, use the arrow keys to move to the desired field and type or select the appropriate value. Press `<ENTER>` after making all changes.

<table>
<thead>
<tr>
<th>Entry Fields</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>TTY type</td>
<td>tty</td>
</tr>
<tr>
<td>TTY interface</td>
<td>rs232</td>
</tr>
<tr>
<td>Description</td>
<td>Asynchronous Terminal</td>
</tr>
<tr>
<td>Parent adapter</td>
<td>port server0</td>
</tr>
<tr>
<td>* PORT number</td>
<td>[9000] +</td>
</tr>
<tr>
<td>Enable LOGIN</td>
<td>enable +</td>
</tr>
<tr>
<td>Keep network connection open</td>
<td>yes +</td>
</tr>
<tr>
<td>Keep pty slave side open</td>
<td>no +</td>
</tr>
<tr>
<td>Pty owner (UID)</td>
<td>[]</td>
</tr>
<tr>
<td>Debug level</td>
<td>0 +</td>
</tr>
<tr>
<td>Debug log file</td>
<td>[/var/adm/ras/rtnlog]</td>
</tr>
</tbody>
</table>
Drain timeout [sec]
Click '?' for TCP Port info (or press F1)
Click '?' for Device Characteristics (or press F1)

For example, in the above screen, you may need to type 9000 in the entry field for port number.

**mkdev**

Enter the following command:

```
mkdev -c tty -t 'tty' -s 'rs232' -p 'port server0' -w '9000' -a login='enable'
```

The command will create a device /dev/tty0 and run rtn.aix from /usr/sbin on port server port 9000. The login prompt will appear on the terminal that is plugged into port server port 0.

**Setting up a local serial printer**

To set up a local serial printer (in this example, an HP Laserjet), you can use the smit program or you can type a command at the command line.

**smit**

1. Type: `smit printer`
2. Select **Print Spooling**
3. Select: **Add a Print Queue**
4. Select: **local**
5. Select: **Hewlett-Packard**
6. Select: **hplj-4 Hewlett-Packard LaserJet 4.4M**
7. Select: **Add New Printer**
8. Select: **rs232**
9. Select: **port server0 Available port server0 port server Terminal Server (16 Port)**
10. Select: **Add a Print Queue**

11. The following display will show the current settings. To change any settings, use the arrow keys to move to the desired field and type or select the appropriate value. Press <ENTER> after making all changes.

<table>
<thead>
<tr>
<th>Description</th>
<th>Entry Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td>Names of NEW print queues to add</td>
<td></td>
</tr>
<tr>
<td>PCL</td>
<td>[]</td>
</tr>
<tr>
<td>PostScript</td>
<td>[HP1]</td>
</tr>
<tr>
<td>HP-GL/2</td>
<td>[]</td>
</tr>
</tbody>
</table>

| Printer connection characteristics | |
| PORT number | [9001] + |
| BAUD rate | [9600] + |
PARITY [none] +
BITS per character [8] +
Number of STOP BITS [1] +
FLOW CONTROL to be used [xon] +
Printer TIME OUT period (seconds) [600] +
STATE to be configured at boot time available +

command line

Enter the following command:

```
/usr/lib/lpd/pio/etc/piomkpq -A 'local' -p 'hplj-4' -v 'hplj-4' -s 'rs232' \\
-r 'port server0' -D ps -q 'hp13' -w '9001'
```

The command will start rtn.aix on port 9001.

To setup a parallel printer

Setting up a parallel printer is a two-step process: 1) set up a port server parallel port device; and 2) set up an AIX print queue for the port server parallel port device. In this example, the port server is an RCS/4000 and the parallel port device is an HP Laserjet.

Step 1: Set Up Device

1. Type: `smit printer`
2. Select: `Printer/Plotter Devices`
3. Select: `Add a Printer/Plotter`
4. Select: `rtn port server Parallel Port`
5. Select: `parallel.rtn`
6. Choose the appropriate port server parent adapter from the list provided.
7. The following display will show the current settings. To change any settings, use the arrow keys to move to the desired field and type or select the appropriate value. Press `<ENTER>` after making all changes.

   [Entry Fields]

   | Printer/Plotter type | rtn       |
   | Printer/Plotter interface | parallel.rtn |
   | Description | port server Parallel Port |
   | Parent adapter | port server0 |
   | * PORT number | [9032] + |
   | Keep network connection open | yes + |
   | Keep pty slave side open | no + |
   | Pty owner (UID) | [] |
   | Debug level | 0 + |
   | Debug log file | [/var/adm/ras/rtnlog] |
   | Click '?' for TCP Port info (or press F1) |
   | Click '?' for Device Characteristics (or press F1) |

8. The command run by `smit` will create a device / dev/ lp0 and run rtn.aix from /usr/sbin on port server port 9000.
Step 2: Set Up Print Queue

1. Type: `smit printer`

2. Select: **Print Spooling**

3. Select: **Add a Print Queue**

4. Select: `port server4000  Printer Attached to port server Parallel Port`

5. Select: **Hewlett-Packard**


7. The following screen will show the current settings. To change any settings, use the arrow keys to move to the desired field and type or select the appropriate value. Press `<ENTER>` after making all changes.

   Name of port server Parallel Port device [lp0]

   For example, you may need to enter: `lp0`

8. The next screen to be displayed will allow you to add a print queue for the selected device. To change any settings, use the arrow keys to move to the desired field and type or select the appropriate value. Press `<ENTER>` after making all changes.

   [Entry Fields]
   
   Description  
   Hewlett-Packard LaserJet>
   
   Printer name  
   lp0

   Names of NEW print queues to add

   PCL  
   [ ]
   PostScript  
   [hp32]
   HP-GL/2  
   [

9. You should now be able to send output to the printer using the command:

   `lpr -Php32 <filename>`
The r4000 Utility

The r4000 utility is a command line based utility that allows an administrator to manage the contents of the port server's flash memory from a host system. This utility allows you to:

- upload configuration files or software updates to the port server,
- download configuration files, operational software, syslog data and trace data from the port server,
- reboot the port server.

Currently supported hosts include:

- SunOS 4.1
- NT

Call customer support or visit our web site at http://www.systech.com to check for other versions.

Syntax

```
r4000 {port servername} [-U user_passwd] [-A admin_passwd] [-u bdnl_file]
   [-d def_bdnl] [-c curr_bdnl] [-s syslog_fn] [-f firmware] [-o onboard_os]
   [-p prod_data] [-R #minute]
```

where port servername can be either a name or an IP address.

If invoked without any optional arguments, r4000 requests the user and admin passwords and then displays a simple menu of options. Or the passwords and options may be entered at the command line, allowing the utility to be invoked from a script file or from a shell command window.

Flags

- **U user_passwd**
  User password for port server. If U is used without specifying a user_passwd, the default user password is used. If U is used, it MUST FOLLOW THE port server HOSTNAME.

- **A admin_passwd**
  Admin password for port server. If A is used without specifying an admin_passwd, the default admin password is used. If A is used, it MUST FOLLOW THE USER PASSWORD(-U) SWITCH.

- **u bdnl_file**
  bdnl_file is the name of a bdnl file to upload to an port server. This file may be a configuration file or operational software.

- **d def_bdnl**
  Download default configuration from port server and save it under def_bdnl filename.


-c curr_bdnl
Download current configuration from port server and save it under curr_bdnl filename.

-s syslog_fn
Download syslog data from port server and save it under syslog_fn filename.

-f firmware
Download flash firmware from port server and save it under firmware filename.

-o onboard_os
Download onboard operational software from port server and save it under onboard_os filename.

-p prod_data
Download flash product data from port server and save it under prod_data filename.

-R #minute
Reboot the port server in #minute minutes. If R is used without specifying the number of minutes, the port server will reboot immediately. WHEN USED, THIS OPTIONAL SWITCH MUST ALWAYS BE THE LAST ONE ON THE COMMAND LINE.

Examples
The following command downloads a configuration from the port server with the IP address of 198.130.71.21, using the default user and admin passwords.

r4000 198.130.71.21 -U -A -c savecurr.port server

The following command uploads a saved configuration to the port server that is named, port server6, using passwords that have been changed, and reboots the port server in 5 minutes.

r4000 port server6 -U userp -A adminp -u savecurr.port server -R 5

The following command uploads new software to the port server that is named, port server6, using passwords that have been changed, and reboots the port server immediately.

r4000 port server6 -U userp -A adminp -u port server4000_05a.bdnl -R
The Deterministic Telnet Server Utility

The Deterministic Telnet Server Utility (dtelnetd) is a modified version of the Berkeley telnetd server that functions similarly in most respects. However, dtelnetd allows you to specify which PTY device to use for an incoming telnet connection, rather than arbitrarily picking a PTY device as the BSD server does. This functionality is useful: 1) for host applications that require users to be logged in on the same PTY device during every session and 2) when it is desirable to associate remote physical port server ports with specific local PTY devices.

The dtelnetd utility works by establishing a mapping of TCP port numbers to PTY device names. It runs through the standard UNIX inetd service and therefore invokes dtelnetd servers on-demand (as opposed to RTN services which are always running). Since dtelnetd is less resource intensive than RTN, it is preferable in situations where the port server can initiate telnet sessions. For applications where the UNIX host must initiate sessions, you must use RTN.

Requirements and System Specific Information

In order to use the dtelnetd server, your host system must be configured to emulate BSD-style ptys which adhere to the BSD naming conventions (see step 2 under "Configuring dtelnetd" below). Many UNIX operating systems are already configured to emulate both STREAMs and BSD-style PTYs. Special cases that need to be configured for BSD emulation are listed here:

- Motorola System V/88 Release 4.0
  This SVR4 based OS has a STREAMS-based pty subsystem which can be configured to emulate BSD-style ptys (called BCS pseudo-terminals) by adding the following line to the /stand/system configuration file:
  
  `INCLUDE:BCSPTS`

  The system must be rebooted for changes to take effect.

Configuring dtelnetd

Configuring the dtelnetd server is a four-step process:

1. Add network service identifiers in /etc/services.
   Each pty slave mapping requires its own separate service definition in the /etc/services file. These definitions have the following format:

   `<service name> <TCP port>/tcp`

   An example of three definitions, along with suggested values, is:

   `dtd-9000 9000/tcp`
   `dtd-9001 9001/tcp`
   `dtd-9002 9002/tcp`
The service names will be used in step 2 to specify the port mapping.

2. Add definitions to inetd.conf to establish port to pty mapping.

On most systems, the inetd.conf file is found in /etc. However, some systems (notably Motorola SVR3.2 implementations) store inetd.conf in /usr/etc.

Each of the service identifiers (see step 1) requires a corresponding entry in the inetd.conf file in order to have inetd listen for connections on those ports. Use the format below that is appropriate for your system.

MOTOROLA SYSTEM V/88 R4.0
DIGITAL UNIX (Formerly DEC OSF/1)
AIX 4.x

\[
\text{<service name> stream tcp nowait root /usr/sbin/dtelnetd dtelnetd -p <pty slave>}
\]

MOTOROLA SYSTEM V/88 R3.2

\[
\text{<service name> stream tcp nowait root /usr/etc/inet/dtelnetd dtelnetd -p <pty slave>}
\]

NOVELL UnixWare 2.x

\[
\text{<service name> stream tcp nowait root /usr/sbin/in.dtelnetd in.dtelnetd -p <pty slave>}
\]

SCO SYSTEM V R3

\[
\text{<service name> stream tcp nowait NOLUID /etc/dtelnetd dtelnetd -p <pty slave>}
\]

where

\[
\text{<service name> corresponds to the service port definition from step 1 (above).}
\]

\[
\text{<pty slave> corresponds to the name of the slave-side pty node which will be mapped to this particular port. This name must be a valid BSD-style pty node of the following format:}
\]

\[
/\text{dev/tty[p-za-l][0-9a-f]}
\]

If your system is not listed above, you can generally look at the entry for the standard telnet service running on port 23. The inetd entry for dtelnetd will be the same as the telnet entry, except for the service name and the addition of the '-p' option.

The -p is necessary to provide the name of the slave device, but any other standard telnetd options may be added as necessary.

Any connections to the specified port will be made over the specified slave pty. The name of this pty can then be obtained using ttyname() or other equivalent methods once the user is logged in. The following examples of the inetd.conf entries use the services defined in step 1 (Motorola System V/88 R40):

\[
dtd-9000 stream tcp nowait root /usr/sbin/dtelnetd dtelnetd -p /dev/tty0
\]

\[
dtd-9001 stream tcp nowait root /usr/sbin/dtelnetd dtelnetd -p /dev/tty1
\]

\[
dtd-9002 stream tcp nowait root /usr/sbin/dtelnetd dtelnetd -p /dev/tty2
\]

3. Place the dtelnetd server binary in the appropriate directory.
This directory is the one specified in the `inetd.conf` file above. Find the appropriate directory for your system and place the file there.

4. Activate the new services through `inetd`.

Before the changes made to the above files become effective, `inetd` must be restarted. This is usually accomplished by sending a SIGHUP to the `inetd` processes (which will then be automatically respawned). This can be done as follows:

```
kill -1 <inetd pid>
```

### Using Command-Override

By default, the `dtelnetd` daemon will spawn the command, `login`, whenever it receives a new connection request. The default command can be overridden by using the `-c` option for command override. Add the `-c` option to the end of the `dtelnetd` configuration line in the `inetd.conf` file, as in the following example:

```
dtd-9000 stream tcp nowait root /usr/sbin/dtelnetd dtelnetd \     -p /dev/tty0 -c mylogin tty0 arg2 arg3
```

**NOTE**: The above example should be entered as a single line.

The entry will execute the command "mylogin tty0 arg2 arg3" instead of the standard `login` command. The `-c` option and its associated arguments **must** be the last option on the command line in `inetd`. In addition, `inetd` does not support quoting or other shell-related features. Some versions of `inetd` impose a limit on the number of arguments which may be supplied. See your operating system documentation for more information.

### Configuring the port server

To configure the port server to use the `dtelnetd` services you have configured, set up a custom configuration for each port that needs to be mapped to a pty port. The custom configuration options that are most pertinent are sessions and wait for keyboard hit.

#### Sessions

One session should be configured which contains the following command line:

```
telnet { host } { dtelnetd port }
```

where

- `host`: IP address or hostname of the `dtelnetd` host.
- `dtelnetd port`: TCP port of the PTY device you want to map to.

#### Wait for Keyboard Hit

This option determines whether the above telnet session will be started immediately or whether it will be delayed until a terminal attached to the port server serial port gets a keyboard hit. If possible, this option should be set to **yes** to prevent the telnet session from starting until a user initiates it from the terminal.
Chapter 8: Writing Chat Scripts

The port server Chat scripts describe the interaction between the port server and the modem or the remote system. Scripts used to interact between the port server and the modem are termed modem scripts; scripts interacting with the remote system are termed the login scripts.

**Fields**

port server Chat strings are composed of one or more paired fields, such as:

```
"Password:" "user:" ">>" "ppp -t"
```

The script is made up of one or more paired fields.

Each grouping (pair) consists of an Expect field and a Send field. Scripts then take the following form: [Expect field] [Send field] [Expect field] [Send field]

Expect fields consist of characters that the port server expects to receive from the modem (modem script) or remote system (login script). The port server will wait for the characters specified in the expect field until it receives them or until it times out, regardless of any other characters it may receive. The first field is always an expect field, even though the field may be null.

Send fields consist of strings transmitted from the port server to the modem (modem script) or remote system (login script).

In the above example, the first pair consists of "Password" (the expect field) and "user" (the send field). The second pair consists of ">>" (the expect field) and "ppp -t" (the send field).

**Punctuation**

Fields are separated by pairs of demarcation characters. These include the following:

- quotation marks " "
- backslashes \ \
- tabs \ t
- spaces

You can add any number of spaces between other demarcation characters to enhance readability.
The following modem scripts produce identical actions, one using quotes and the other using spaces to separate the fields. The null content of the lead Expect field means nothing is expected.

"AT" "OK" "ATDT\T" (readability spaces between quotes)

AT OK ATDT\T

Subfields

Expect fields only (not Send fields) may be subdivided into subfields. Expect subfields are used for anticipating communication difficulties. Subfields are structured as follows:

Expect[-Send]. The Expect subfield is expected by the port server from the modem or remote system. If, however, there is reason to believe the string may not be received or cannot be read, a Send subfield can be included. The Send subfield will be transmitted when the Expect field fails to arrive. For example, given the following Expect field:

login:--“\d"

the port server, failing to receive the login: string, sends a control d.

A carriage return is automatically transmitted by the port server following the Send subfield. This feature can be disabled with the \c special character (see list below).

Expect[-Send-Expect]. The Expect subfield is expected by the port server from the modem or remote system. If, however, the string is not received or cannot be read, the port server initiates a Send string (subfield), followed by the second Expect string (subfield).

Example 1

login:--login: (Expect field only)

Failing receipt of the expected login: subfield, the port server sends a null string followed by the automatic carriage return, then expects login: again.

Example 2

"password:" "user" ">>-user->>

In this login script, the port server expects to receive a password: prompt, then sends the "user" password. It then expects to receive the ">>" prompt. If it fails to receive the expected ">>", it sends the "user" password and waits for the ">>" prompt again.

Example 3

password:--password: user >>

The port server expect to receive a "password:" prompt. If it does not receive the prompt, it sends a null string, followed by the automatic carriage return. It then expects "password:" again. When it receives "password:", it sends "user" and then expects to receive a ">>" prompt.
Chat script characters

The following is a list of additional port server special chat script characters.

\b  Backspace character.
\c  Placed at the end of a field, suppressing the normal carriage return.
\d  One second delay.
\p  One-tenth second delay.
\k  Break character, sometimes used to cycle a modem’s speed.
\r  Carriage return.
\n  Send a new line.
\s  Space.
\t  Tab.
\T  Phone number to be copied from configuration database.
Chapter 9: SNMP TCP Drop Capability

As part of its SNMP support, the port server provides the capability of dropping TCP connections using a remote SNMP manager node. This capability is implemented through the "tcpConnTable" table which is part of the standard Internet MIB definition for SNMP managed devices.

The Management Information Base (MIB)

SNMP provides a method by which information relative to the operation of a device can be viewed and modified. This information typically consists of things like operational statistics, configuration information, and connection information. Device information is referenced through MIB object IDs, or variable names, which correspond to specific items of information. These object IDs are organized into a tree-based hierarchy (called the Management Information Base, or MIB), and are subdivided into categories such as "system", "tcp", "udp", "interfaces", etc.

Understanding the tcpConnTable

The section of the MIB which is pertinent to the operation of the TCP Drop command is called the "tcpConnTable" and exists in the MIB under the following pathname:

iso.org.dod.internet.mgmt.mib.tcp.tcpConnTable

This MIB subtree contains a series of objects which defines a table of information. This particular table reflects information which would normally be provided by the port server command: netstat -a -p tcp -n. Under SNMP, tables are implemented by sorting and grouping object IDs. The tcpConnTable contains a series of tcpConnEntry objects which contain the following information:
tcpConnEntry - Describes a TCP connection.
{
    tcpConnState - Integer describing the state of the connection.
tcpConnLocalAddress - Local IP address.
tcpConnLocalPort - Local TCP port.
tcpConnRemAddress - Remote IP address.
tcpConnRemPort - Remote TCP port.
}

To view the tcpConnTable, you must use some sort of SNMP manager to walk through the variables which comprise the table. Many SNMP managers will have a graphical interface which will display the tcpConnTable in a graphical, table-based format. If you are using a text-based utility, be aware that MIB objects are by field and not by entry (see example below).

Entries in the tcpConnTable are indexed as follows:

    mib.tcp.tcpConnTable.tcpConnEntry.field_name.a.b.c.d.e.f.g.h.i.j

where

field name: One of the above tcpConnEntry fields
a.b.c.d: Local IP address of the connection
e: Local TCP port of the connection
f.g.h.i: Remote IP address of the connection
j: Remote TCP port of the connection

Example

The following is sample output from a text-based SNMP "walk" command and the corresponding "netstat" output from the port server. These listings show information pertaining to two TCP connections: an established connection and a listening connection. Notice how the sorting of the SNMP walk command groups the objects by field, rather than by connection.

NETSTAT OUTPUT:

    [admin:1/8]>netstat -a -p tcp -n
    Active Internet connections (including servers)
    Proto Recv-Q Send-Q Local Address Foreign Address (state)
    tcp 0 0 199.171.31.229.23 199.171.31.156.4384 ESTABLISHED
    tcp 0 0 *.23 *.* LISTEN
SNMPWALK OUTPUT

tcp.tcpConnTable.tcpConnEntry.tcpConnState.0.0.0.23.0.0.0.0.0 = listen(2)
tcp.tcpConnTable.tcpConnEntry.tcpConnState.199.171.31.229.23.199.171.31.156.4384 = established(5)
tcp.tcpConnTable.tcpConnEntry.tcpConnLocalAddress.0.0.0.0.23.0.0.0.0.0 = IpAddress: 0.0.0.0
tcp.tcpConnTable.tcpConnEntry.tcpConnLocalAddress.199.171.31.229.23.199.171.31.156.4384 = IpAddress: 199.171.31.229
tcp.tcpConnTable.tcpConnEntry.tcpConnLocalPort.0.0.0.0.23.0.0.0.0.0 = 23
tcp.tcpConnTable.tcpConnEntry.tcpConnLocalPort.199.171.31.229.23.199.171.31.156.4384 = 23
tcp.tcpConnTable.tcpConnEntry.tcpConnRemAddress.0.0.0.0.23.0.0.0.0.0 = IpAddress: 0.0.0.0
tcp.tcpConnTable.tcpConnEntry.tcpConnRemAddress.199.171.31.229.23.199.171.31.156.4384 = IpAddress: 199.171.31.229
tcp.tcpConnTable.tcpConnEntry.tcpConnRemPort.0.0.0.0.23.0.0.0.0.0 = 0
tcp.tcpConnTable.tcpConnEntry.tcpConnRemPort.199.171.31.229.23.199.171.31.156.4384 = 4384

Issuing the TCP Drop Command

In order to drop a TCP connection using SNMP, you must first be sure that an SNMP community has been defined which allows write access to at least the tcpConnState variable. For more information about SNMP communities and access rights, see COMMUNITY, page 101 and VIEW, page 170.

To actually drop the connection, you must know the complete MIB object ID of the tcpConnState object pertaining to the connection you wish to drop. This can be found by using the above methods to walk through the tcpConnTable. Once the object has been found, you can drop the connection by issuing an SNMP set command and setting the value of the tcpConnState variable to "deleteTCB(12)" (this is usually done by just specifying the integer value 12). This will cause the port server to immediately delete the connection from its internal tables, effectively dropping the TCP connection.

Example

Using the configuration from the previous example, suppose that you want to drop the connection from IP address 199.171.31.156. The tcpConnState object which corresponds to this connection is:

tcp.tcpConnTable.tcpConnEntry.tcpConnState.199.171.31.229.23.199.171.31.156.4384

You would need to issue an SNMP set command on this object and assign it the value of "12" (i.e. deleteTCB).
Chapter 10: Troubleshooting Tools

Data Capture

This set of tools, designed primarily for developers, monitors and captures a variety of traffic on serial ports. The traffic monitored includes data, modem signals, and other significant events on the port (such as opening or closing).

At the heart of this tool is an extension to the trace mechanism that already exists on the port server, to include the ability to capture information about the serial ports. In addition to this, however, is the ability to enable, disable, control, monitor and record this data from the host system. There are mechanisms and a protocol for getting snapshots of activity on the board or for getting continuous updates of activity.

The following example illustrates the use of the data capturing facilities. For this example we will look at the events happening on a port that is configured for the default service when a terminal is connected to that port.

The first step is to enable the tracing on a particular port. For this example, we will examine port 2. Note that you must be in expert mode to issue the following commands.

```
[expert:1/6]>>trace_on port 2
[expert:1/7]>>trace_display
```

The trace_on command enables tracing. The trace_display command shows that no data has been collected so far. At this point we connect a terminal to port 2. Immediately after connecting the terminal to port 2 we issue the trace_display command and get the following output:

```
[expert:1/8]>>trace_display
14257788 port task 0 (prnull) [17] port 2 modem signal change: CTS current: CTS DSR RI
14257809 port task 0 (prnull) [19] port 2 cts change ignored
14308148 port task 0 (prnull) [17] port 2 modem signal change: DCD current: DCD CTS DSR RI
14308175 port task 0 (prnull) [1b] port 2 device event connect
14342074 port task 125 (swk_main_task) [4] port 2 transmit 'W' 'e' 'l' 'c' 'o' 'm' 'e' 
14342092 port task 125 (swk_main_task) [4] port 2 transmit 't' 'o' ' ' ' ' ' ' ' '
14353106 port task 0 (prnull) [4] port 2 transmit 'h' 'e' ' ' ' ' ' ' ' 'S' 'Y' 'S' 'T' 'E''
14353124 port task 0 (prnull) [4] port 2 transmit 'C' 'H ' ' ' ' ' ' ' 'R'
14353142 port task 0 (prnull) [17] port 2 modem signal change: CTS current: DCD DSR RI
14354519 port task 0 (prnull) [19] port 2 cts change ignored
14365740 port task 0 (prnull) [4] port 2 transmit 'R' 'e' 'm' 'o''
14365758 port task 0 (prnull) [4] port 2 transmit 't' 'e'' ' ' ' ' ' ' ' '
14378209 port task 0 (prnull) [4] port 2 transmit 't' 'r'' 'C'
1438227 port task 0 (prnull) [4] port 2 transmit 'C' 'S' '/' '4'
```

The trace_on command enables tracing. The trace_display command shows that no data has been collected so far. At this point we connect a terminal to port 2. Immediately after connecting the terminal to port 2 we issue the trace_display command and get the following output:
The first column is the time in microseconds since tracing was turned on. This example output occurred just over 14 seconds after the trace_on command was issued.

The next 6 columns are not of much interest to the port tracing command.

Column 8 is the number of the port. Multiple ports can be traced at one time, but be aware that the trace buffer only holds the last 2048 items and will fill very fast when multiple ports are being traced.

The remaining information on the line is dependent upon the particular event that occurred on the port. The following line indicates a change in modem signals on the port. In particular, the carrier detect (DCD) signal changed and the complete set of modem signals now being asserted for this port are DCD, CTS, DSR, and RI.

The DCD signal being asserted resulted in a special "event" called "connect". That event is noted by the following line:

The port was configured to display a banner as part of the login process. This banner starts with the word "Welcome". The output of this data is shown in groups of up to 4 characters as shown in the following line:

The complete list of port trace messages are as follows:
- transmit - shows the characters being transmitted.
- receive - shows the characters received.
- no new data timeout - no characters received within the timeout period.
- send xon
- send xoff
- send break
- no receive buffer - character received but no memory was available to store it.
- overrun error
- framing error
- parity error
- receive break start
- receive break end
- receive xon
- receive xoff
- receive suspend char
- receive interrupt char
- modem signal change
- modem signal on closed port
- cts change ignored - rts/ cts flow control not configured for port.
- set modem signals - output modem signals are being set.
- device event - one of: break, error, flush, hangup, intr, connect, start, stop, switch.
Chapter 11: Firmware Monitor

Entering the Firmware Monitor

When you power up the port server, the server goes immediately into operating mode. The reset switch allows you to take the unit out of operating mode and put it into the firmware monitor. Operation is suspended and all active sessions are ended when the server enters the firmware monitor. If you use the save command to save the configuration settings you previously defined, these settings will be maintained when you exit the firmware monitor and reenter operating mode. The firmware monitor allows you to load new software into the port server, perform maintenance or troubleshooting functions, make a bootp request, etc. as described below.

To enter the firmware monitor, press the reset switch. Some models have one reset switch that resets both the hardware and the software. Other models have two reset switches—one hardware reset and one software reset. See the Hardware Manual for your port server model to locate the reset switch(s). Access the reset switch by pressing a narrow non-metal item (e.g. a toothpick) through the hole in the port server's case until the switch itself is depressed. Information will be displayed on a monitor connected to the lowest port number, port 0 or port 1, depending on the model of port server you are using.

NOTE: Entering the firmware monitor suspends normal operation. All active sessions are ended.

NOTE: The terminal you connect to port 0 or 1 must be set for 9600 baud, 8 bits, no parity.

Command Menu

The initial menu that is displayed on the terminal connected to the lowest numbered port when you enter the firmware monitor is shown below. Note that the actual version level of the firmware installed on your port server will be displayed where indicated.
The functions you can access from the Command Menu are:

**b - bootp**: The bootp command allows you to determine if a server on the network recognizes this port server. This command will use bootp, rarp, and DHCP to try to identify the port server.

**i - IP addresses**: The IP addresses command allows you to set a temporary IP address for the port server and for the tftp server for debugging purposes. IP addresses set in the firmware monitor do not affect the IP address of the port server in normal operating mode.

**r - restore default configuration settings**: The restore command restores the port server default configuration settings.

**WARNING**: Do not use the restore command as a system reset. As all user configured settings are lost when this command is invoked, use restore only when recovery from an existing problem is not possible through the command set. In the unlikely event that the port server hangs or the established configuration proves to be operationally unacceptable, use restore to reset the unit’s default configuration settings.

**s - diagnostics**: This option accesses commands (described briefly in the following section) that can be initiated for troubleshooting the port server. These commands were designed for factory use. Do not use them unless directed to do so by a customer service representative.

**u - update operating software**: In the event that the port server operating software level is updated for units already in the field, you would use this command—following instructions—to update your unit's software level.

**NOTE**: Because the process of updating operating software erases the existing version, do not use the update command unless new software and update instructions are provided.

**v - version**: The version command displays the version levels of the firmware and operating system software running on your port server.

**q - quit**: The quit command closes the command menu and returns you to operating status. If you saved your configuration settings before entering the firmware monitor, and you did not use the restore function, all user-defined configuration settings are
maintained when the port server quits from the firmware monitor. If you either used restore, or you did not save your settings, all user-defined configurations settings are lost when you quit from the firmware monitor.

**Diagnostic Menu**

If you access the Diagnostic Menu, you will see the options shown below, which may aid in troubleshooting the port server. These tests are designed for factory use only; use them at the direction of a customer service representative.

```
SYSTECH Diagnostic Menu
  c - compact product data area
  d - dump memory to host
  f - firmware configuration table display
  h - hardware exerciser
  l - log of errors recorded
  m - modify menu options
  p - product data display
  r - create flash layout product data record
  u - update flash segments from host
  q - quit

Please choose an option:
```
Appendix A: Open Networks

Computer networks are collections of hardware and software that connect computers and allow them to share resources and devices.

Network Topologies

Networks can be arranged in different physical configurations or topologies. Although topologies can be discussed in terms of location (of the computers on the network) and connection (how the computers are connected), we are primarily concerned with location topologies.

Location topologies include point-to-point networks, local area networks (LANs), and wide area networks (WANs), depending on where the computers are located. Point-to-point networks connect two hosts; LANs and WANs connect two or more hosts and are differentiated primarily by the geographic distances separating the hosts.

TCP/IP products support LANs, WANs, and point-to-point networks.

Point-to-Point Networks

Point-to-point networks use a single communication channel to connect two computers. LANs and WANs can also be point-to-point networks because they can connect either two local hosts in close physical proximity via a cable or two distant hosts separated by thousands of miles via a phone line and modem.

Local Area Networks

LANs consist of multiple hosts located closely together, for example, on a college campus, in a hospital, or in a single building. Several types of cables (coaxial, twisted pair, optical fiber) can be used to form the physical link between the various terminals and hosts. LANs usually span a kilometer or less and typical Ethernet LANs operate at 10 Mbps.
Wide Area Networks

WANs connect computers that are located over a large geographic area. Telephone lines, radio transmission, and satellite broadcast can form the links across a WAN. Virtually any LAN can be configured to connect to a WAN. WAN data rates are usually between 9.6 Kbps and 2.048 Mbps.

Network Protocols

Organized functions, grouped in a layered hierarchy, interact to make networking possible. Each function is a component of the whole that interacts with other functions in a systematic manner. When communication takes place, information passes vertically across layer boundaries and laterally between peer layers, according to rules known to the two interacting layers. These rules, or protocols, define the interface between the two layers and permit hosts to communicate with each other.

Some protocols define the way data is transmitted, but do not have provisions for user-level applications. For others, however, a coordinated set (or suite) of protocols is specified, offering a range of services or communications for either a specific type of compatible computing machines, or for open systems.

The OSI Model

In 1977, the International Standards Organization created the Open Systems Interconnection (OSI) Model, a conceptual view of networking. It depicts the organization of network activity as seven “layers,” with the least complex layer being on the bottom and the most complex on the top:

- Application layer: allows a user to perform network services like file transfer and email, or use applications like telnet, rlogin, etc.
- Presentation layer: handles format and code conversion for the application program.
- Session layer: establishes and maintains sessions between application programs.
- Transport layer: handles communication across the network.
- Network layer: routes data across the network.
- Data-link layer: controls access to the network medium; the data-link layer comprises the Media Access Control (MAC) layer and the Logical Link Control (LLC) layer.
- Physical layer: the networking medium—cables, connectors, etc.

The Internet Protocol Suite: TCP/IP

The port server works with the Internet Protocol (IP) suite. The IP suite, commonly referred to as TCP/IP, is an open systems architecture that enables communication between various computers using various operating systems. These computers can be set up in different connection topologies and at different, wide-spread locations.

TCP/IP comprises several layers, named similarly to those in the OSI model (above). The layers range from the physical (hardware) layer, through several software layers, to the
user, or application, layer. Each succeeding layer provides increasingly complex services than the layer below it. The user layer is the most complex, and through it, with little or no networking experience, an end user can take advantage of the facilities provided by the underlying layers.

TCP/IP protocols allow two or more cooperating processes on different hosts to communicate. TCP/IP hosts do not have to use the same computer or operating system. Each host can use a unique operating system, and each host can be interconnected to differing hosts by TCP/IP networking software. Each layer in TCP/IP has a defined protocol that governs the functions within that particular layer. The protocol implemented in a layer is specific to that layer so that a layer on one host can communicate with the corresponding layer on another host.

This ability of a protocol to communicate with a corresponding layer on another host is called peer-to-peer communication. It allows the communication between hosts to appear to be carried on entirely at the user level.

**What TCP Does**

TCP defines data flow, acknowledges data, and retransmits lost or damaged data. The TCP on a sending host breaks up messages into IP datagrams; the TCP on the receiving host reassembles the datagrams into the proper order, and requests that missing or garbled pieces of datagrams be resent. A receiving host's TCP acknowledges every byte, rather than every datagram, that it receives. TCP allows multiple connections to be multiplexed through a single network attachment, and provides a reliable, byte-stream-oriented virtual circuit for application processes.

**User Protocols**

User layer protocols allow users to impose application-specific rules on the session and its data. When these protocols use TCP, users are assured that data is sent and received—without errors—by the underlying network layers. The application does not need to be concerned with the lower-level issues of routing or with device and medium drivers.
The port server provides the following user protocols:

- **File Transfer Protocol (FTP):** FTP allows users to send or retrieve files interactively, remove files, list directories, obtain the status of file transfer, and rename the files. FTP operates as a client/server mode, meaning that a client sends commands and interacts with the user, while a server receives and responds to the commands.

- **Virtual terminal:** The port server's virtual terminal application provides to the user a terminal emulation.

- **routed:** The routed application is the server mode of route, receiving and responding to route commands.

- **telnet:** Telnet is an interactive remote access terminal protocol which allows the user to log into and use a remote computer system on the network as if by direct connection to the remote host.

- **telnetd:** The telnetd application is the server mode of telnet, receiving and responding to telnet commands.

- **rlogin:** Reverse login (rlogin).

- **Simple Network Management Protocol (SNMP):** SNMP is used to monitor and control TCP/IP-based networks. System Administrators can use SNMP to retrieve and alter networking information maintained by hosts and routers attached to a network, and to diagnose and correct network problems from remote hosts.

### Host Name List

A host table is a manually configured list that relates the name or alias of a host with its IP address.

### Domain Name Server

Like the host name list, the Domain Name Service (DNS) associates a host name or alias with its IP address; however, the DNS also distributes the data and allows different administrators to assign host names in a coordinated fashion.

Computers recognize other devices by their addresses, while users recognize other devices by the names assigned to them. In a network, it is necessary to map the user-assigned names to their corresponding addresses.

Small networks often use a simple table to map names to addresses. But as the network grows, the table grows and becomes cumbersome. In addition, the table constantly changes; it becomes difficult to keep it up-to-date on every host. When networks become interconnected, this problem is compounded.

The DNS resolves the problem of maintaining a dynamic name/address cross reference by grouping the names of resources, such as host names, according to the type of organization the host represents (e.g., ucsd.edu and netcom.com), and by replacing the host file of all the names with a specialized distributed database query system. A few hosts actually store files of names and addresses, and they store only partial copies. The database system directs its queries until it finds the correct copy.
Appendix B: Routing

This chapter discusses routing to provide you the background necessary for determining your routing needs and for setting routing options on the port server.

Routers

Routers find the best route for data sent to them by the previous router or the end station of the LAN; but because routers are protocol dependent, they have to do more data processing than bridges and they do not have a bridge's ability to determine addresses.

Gateways

Gateways are combinations of hardware and software which connect two dissimilar LANs or operating environments.

A gateway serves as the interpreter for computers that communicate in different protocols. Since the function performed by a gateway is more complicated than that performed by a bridge, a gateway is usually slower and handles fewer devices. Because it works at the top three layers of the OSI model, it is the best choice for connecting systems with completely different architectures. You can also use a gateway to connect your LAN to public-access networks. After that, you can share files, send messages, and log onto a remote computer.

What is Routing?

Routing is the process of selecting a path over which to send packets. A router is a device that makes the routing decisions.

The Internet comprises multiple physical networks interconnected by computers called gateways. Hosts connect directly to one or more physical networks. Both hosts and gateways take part in IP datagram routing, and hosts with multiple network connections can act as gateways.

For the purposes of this discussion, two types of routing will be defined: direct and indirect. Direct routing transmits an IP datagram from one device directly to another. Direct routing, which is the basis of all Internet communication, uses an underlying physical transmission system. Indirect routing occurs when the destination is not on a directly attached network, requiring the sender to pass the IP datagram to a gateway for delivery.
Gateways are not required when IP datagrams are transmitted between two devices on a single physical network. In this case, the sender encapsulates the datagram in a physical frame and sends it directly to the destination address. A datagram can be sent directly if the network portion of the destination IP address matches the network portion of the sender’s IP address.

Indirect routing is more complex because the sender must identify a gateway to which the datagram can be sent. The gateway then forwards the datagram toward its destination network. Because Internet gateways form a cooperative, interconnected system, the gateways pass IP datagrams from gateway to gateway until they reach a gateway that can deliver the datagram directly.

**Routing Tables**

Routing algorithms define the routing process between gateways. A routing table, which contains information about possible destinations, resides on each gateway machine. To limit the size of routing tables—which could never include all possible destination addresses—routing tables use the network ID to make routing decisions. To keep routing tables small, and routing efficient, gateways use destination network addresses rather than destination host addresses. A gateway does not need to know the final destination address as long as it knows the interconnection structure of the networks and the destination network to which a datagram has been sent. When a datagram arrives at a gateway, the gateway extracts the network portion of the destination IP address and makes routing decisions based on the address.

Routing tables typically contain pairs of addresses. Each pair consists of an Internet network address and a corresponding gateway address to which datagrams destined for that particular network address are sent. All gateways listed in a machine’s routing table must lie on networks to which that machine connects directly, making it possible to reach them directly.

The size of routing tables remains independent of the number of hosts in the Internet. The table grows only when new networks are added.

**Bi-directional Communication Between Gateways**

It is necessary that the sending gateway communicate with the destination gateway, however, it is equally necessary for the destination gateway to be able to communicate with the sending gateway. Destination gateways must have a way to send reports of problems—throughput delays, nonoperational destination hosts, etc.—back to the original source, and traffic may follow an entirely different path in one direction than on the return trip.
Default Routes

A default route is a “fall-back” mechanism for instances when no appropriate destination network address appears in the routing table. If the IP routing software cannot identify the destination network in the routing table, it will send the IP datagram to the default. This is particularly useful for a site with a small set of local addresses and only one connection to the rest of the Internet. For example, default routes work well in host machines that attach to a single physical network and reach only one gateway leading to the remainder of the Internet.

Host-Specific Routes

Most IP routing software allows the system administrator to specify per-host routes, providing more control over network use and access. The ability to specify a special route for one individual machine is especially useful for debugging network connections or routing tables.

Routing Information Protocol (RIP)

The port server includes support for the Routing Information Protocol (RIP). Implemented by the routed application, RIP eliminates the need for a separate IP network router and provides remote WAN-connected users with transparent, equal access to locally attached network devices. RIP is used most commonly when all participating machines attach to a single local area network.

How RIP Works

RIP arranges for each gateway to periodically broadcast to its neighbors a message of its current routing database. The message lists each destination, along with the distance to that destination measured in gateway hops. RIP’s hop count is relative to the sender’s active routing table. That is, RIP messages do not list a gateway, nor do they list networks relative to a specific gateway.

RIP Message Format

All RIP routing messages consist of a fixed header followed by an optional list of reachable networks.

The Command field specifies the contents of the packet as a request for routing information (1) or a response (2). Normally, gateways periodically broadcast unsolicited response packets. The Version field contains the protocol version number and is used by the receiver to verify that it will interpret the message correctly.
The RIP message format is not limited to the Internet; it can be used with multiple network protocol families. Each network reported by RIP can have an address of up to 14 octets. Since Internet addresses need only 4, the remaining octets are unused. The Family of network “n” field identifies the protocol family under which the network address should be interpreted, using integers assigned to address families under 4.X BSD UNIX.

The Distance of network “n” field is the last field in each RIP routing update. This field contains an integer count of the distance—measured in network hops—of the specified network.
Appendix C: SLIP and PPP

The Serial Line Interface Protocol (SLIP and Point-to-Point Protocol) are methods of transmitting IP datagrams over serial point-to-point links. SLIP is usually used with Internet Protocol; PPP is used for other network-layer protocols. The port server supports both SLIP and PPP; however, there are differences between the protocols and advantages and disadvantages to using one or the other.

For more information on SLIP and PPP, see Service Configurations, page 78.

Serial Line Interface Protocol (SLIP)

Although SLIP is a widely used protocol, it is not an Internet standard. SLIP is a simple protocol, designed for host-to-host, host-to-router, router-to-router, or workstation-to-host communications over asynchronous or synchronous leased or dial-up serial lines.

Most common SLIP implementations accept IP datagrams up to 1,006 bytes in length. SLIP datagrams contain two special characters: ESC and END. The END character marks the beginning and end of the SLIP packet. The ESC character marks bytes that match the ESC or END characters.

The biggest advantage to using SLIP is the ease with which it can be implemented (the port server implementation requires no modification once a custom service configuration is defined as SLIP). In addition, it works well in many environments. Its simplicity, however, is also the source of potential problems with SLIP, because SLIP makes no provision for addressing, identifying protocols, and error detection and correction.

Addressing

SLIP provides no mechanism for hosts to communicate addressing information with each other. For packets to be routed properly, each computer in a SLIP connection must know the other’s IP address.

Identifying protocols

Because SLIP datagrams have no protocol-type field, only a single network-layer protocol may be used over a SLIP connection. This means that multiple protocols cannot share one SLIP line.
Error detection and correction

Because SLIP is used primarily over low-speed lines, transmission time is an important consideration. Retransmitting packets would be very costly, so SLIP relies on the IP application to detect—and if necessary correct—damaged packets.

SLIP does not include any packet compression mechanism, the addition of which would greatly increase performance.

Point-to-Point Protocol (PPP)

PPP provides a standard method of encapsulating network layer protocol information over serial, bit-oriented synchronous and asynchronous point-to-point links. PPP also defines an extensible Link Control Protocol, which allows negotiation of an Authentication Protocol for authenticating its peer before allowing network layer protocols to transmit over the link.

PPP retains compatibility with existing hardware and allows control characters such as XON/ XOFF to be transmitted without affecting the operation of the link.

Authentication is not, by default, mandatory. If authentication is desired, your implementation must specify the Authentication-Protocol Configuration Option during the link establishment phase. The authentication protocols are used primarily by hosts and routers that connect to a PPP network server via switched circuits or dial-up lines, however, they might also be applied to dedicated links as well. The server can use the connecting host's or router's identification in selecting options for network layer negotiations.

PPP provides:

- A datagram encapsulation mechanism for use on serial links, based on the ISO High-level Data-Link Control protocol (HDLC).
- A Link-Control Protocol (LCP) for establishing, configuring, authenticating, and testing the data-link connection.
- A family of Network-Control Protocols (NCPs) for establishing and configuring different network-layer protocols.

To establish a PPP line, both ends of the connection first send a series of LCP packets to configure and test the data link. If authentication is selected, the connection is authenticated (the port server uses either PAP or CHAP for authentication). Once authentication is complete, the link remains configured and functional until either explicit NCP or LCP commands close it (in response to a user), or until some external event occurs. Typical external events are an inactivity timer (refer to the port server set timeout command), hardware or software failure, loss of carrier, an authentication problems. The LCP transmits a sequence of terminate packets, then PPP notifies each network layer to take whatever action is required when the link terminates.

To improve performance, PPP uses Van Jacobson compression, which reduces the size of TCP/IP headers.
Password Authentication Protocol (PAP)

The Password Authentication Protocol (PAP) provides a simple method for the peer to establish its identity using a two-way handshake which is performed only during the link establishment phase. After the link is established, an ID/password pair is sent repeatedly from the peer to the authenticator until authentication is acknowledged or the connection is terminated.

As an authentication method, PAP does not provide extensive security for users. Passwords are sent over the circuit without protection from playback or repeated trial and error attempts. The peer is in control of the frequency and timing of the attempts.

PAP is most appropriate in situations where a plaintext password must be available to simulate a login at a remote host. In such cases, PAP provides a level of security similar to the usual user login at the remote host.

It is possible to limit the exposure of the plaintext password to transmission over the PPP link and avoid sending it over the entire network. When the remote host password is kept as a one-way transformed value, and the algorithm for the transform function is implemented in the local server, the plaintext password should be locally transformed before comparison with the transformed password from the remote host.

NOTE: Implementations which include a stronger—more secure—authentication method (CHAP, for example), must offer to negotiate that method prior to PAP.

Challenge-Handshake Authentication Protocol (CHAP)

The Challenge-Handshake Authentication Protocol (CHAP) periodically verifies the identity of the peer using a three-way handshake, which is performed when the link is first established and may be repeated anytime after that.

After the link is established, the authenticator sends a “challenge” message to the peer. The peer responds with a value calculated using a one-way hash function. The authenticator checks the response against its own calculation of the expected hash value. If the values match, authentication is acknowledged; otherwise the connection should be terminated.

CHAP provides protection against playback attack through the use of an incrementally changing identifier and a variable challenge value. The use of repeated challenges is intended to limit the time of exposure to any single attack. The authenticator controls the frequency and timing of the challenges.

This authentication method depends upon a secret known only to the authenticator and that peer. The secret is not sent over the link. This method is most likely used where the same secret is easily accessed from both ends of the link.
NOTE: CHAP requires that the secret be available in plaintext form. To avoid sending the secret over other links in the network it is recommended that the challenge and response values be examined at a central server, rather than each network access server. Otherwise, the secret should be sent to network access servers in a reversibly encrypted form.

The CHAP algorithm requires that the length of the secret be at least 1 octet, and that it be at least as large and unguessable as a well-chosen password. In addition, to ensure a sufficiently large range for the secret to provide protection against exhaustive search attacks, the secret should be at least the length of the hash value for the hashing algorithm chosen (16 octets for MD5).

NOTE: The one-way hash algorithm is chosen such that it is computationally infeasible to determine the secret from the known challenge and response values.

The challenge value should be both unique and unpredictable. Make each challenge value unique so that repetition of a challenge value in conjunction with the same secret would not permit an attacker to reply with a previously intercepted response. Since it is expected that the same secret may be used to authenticate with servers in diverse geographic regions, the challenge should possess global and temporal uniqueness. Each challenge value should also be unpredictable, lest an attacker trick a peer into responding to a predicted future challenge, and then use the response to masquerade as that peer to an authenticator. Although protocols such as CHAP are incapable of protecting against real-time active wiretapping attacks, generating unique unpredictable challenges can protect against a wide range of active attacks.

The Magic Number Configuration Option includes a discussion of sources of uniqueness and probability of divergence.
Index

1-key menus, 105

A
access capabilities table, 110
ACCM, for PPP, 145
active configuration, 87
add command, 88
address resolution protocol, ARP, 181
address translation
  Internet to private, 130
  table, internet to ethernet, 181
address, IP
  assigning, 17–21
  assigning permanent, 122
  obtaining from server, 182
address, MAC, 18–20, 190
admin command, 179
admin keyword, 137
admin, capabilities level, 108
administrative password, 18, 71
advertise keyword, 151
AIX host systems
  rtn utility, 267
  setting up remote printer, 55
  setting up rtn based printer, 59
apmem command, 180
arp command, 181
  assigning IP address with, 18
arp option, ifconfig command, 198
asynch port, setting up ISDN on, 40
Asynchronous Inverse Multiplexing Mode, 41
authentication
  levels, 103
  PPP, 147
  RADIUS, 148
auto answer, modem, 27

B
backoff time
  bi-directional dial-on-demand, 39
dial on demand, 163
backup configuration database, 87
banner keyword, 100
banner option, custom configuration, 105
bi-directional dial-on-demand, 105
bit rate speed, 160
BitSurfr, Motorola, setting up, 39
boot command, 182
BOOTP server
  assigning IP address with, 20
  database file, bootptab, 66–83
  enabling BOOTP, 65
  UNIX, assigning IP address with, 65–83
bootptab file, 20, 65, 66–83
BREAK key, 162
browser, configuring with, 14
browser, web, using to change configuration settings, 88
buffer, trace
  clear, 255
display, 256

C
capabilities access levels
  assigning to custom configurations, 104
  defining, 107
table, 110
Challenge-Handshake Authentication Protocol
  (CHAP), 147, 301–2
character size, 155
chat script
  characters, 279
  dial on demand, 163
  writing, 277–79
checking status of host or gateway, 210
COM ports, 13
command interface, RCS/3000, accessing, 87
command line interface, configuring with, 14, 70
command, custom, run at boot time, 102
commands
  configuration--overview, 88
  configuration--table, 91
  operational--table, 175
commands, telnet, 247–53
community keyword, 101
configuration
active, 87
changing, 87
copying, 63
loading new, 192
restoring defaults, 289
saving settings, 89
saving to a server, 190
uploading/downloading from host systems, 271
configuration commands
   summary, 88
   summary table, 91
   syntax, 89
configuration database
   backup, 87
copying to another system, 190
current, 87
factory default, 87
configuration methods, 13
configuration summary table, 21
configurations, network, 291
configurations, service
   custom, 102
default, 111
user sessions, 82
contact information, adding, 159
contact keyword, 158, 159
CSU/DSU
   network diagram, 48
current configuration database, 87
current loop, 121
current port, displaying info, 259
custom keyword, 102
custom service configurations
   PPP, 144
   slip, 156
custom task, 102
damaged packets, 211
Data Link Connection Identifier (DLCI), 81, 114
dtnetd utility
description, 273–75
description, compared to RTN utility, 273
dtdrsr, 119
dump command, 185
derror messages, sending to server, 243
derror records, removing from flash memory, 188
exit command, 186
expert capabilities level, 109
expert command, 187
firmware options, setting, 175, 182
firmware revision, displaying, 159
flash memory
   managing from a host system, 271
   reclaiming space, 175, 188

bi-directional, 39
bi-directional, 39
configuring modems for, 126
custom service configurations
   PPP, 144
   slip, 156
dial-out modem
   access, 29–30
delay to RCS/4000, serial port, 28–29
display command, 88
dci keyword, 114
dci keyword, 114
d深刻的 Telnet Server Utility
   compared to RTN utility, 273
description, 273–75
dtrsr, 119
drop command, 184
drop command, 184
dsl modem, configuring on serial port, 27–28
dump command
   dump command, 185
dump command, 185
ECHO_REQUEST, 210
ECHO_RESPONSE, 210
EIA-530
   setting WAN port to, 173
encryption key, RADIUS, 148
evironment variables
   displaying, 223
   setting, 233
   unsetting, 260
error messages, sending to server, 243
error records, removing from flash memory, 188
exit command, 186
expert capabilities level, 109
expert command, 187
firmware options, setting, 175, 182
firmware revision, displaying, 159
flash memory
   managing from a host system, 271
   reclaiming space, 175, 188

factory default configuration database, 87
file transfer
   using KERMIT or XMODEM, 247
File Transfer Protocol (FTP), definition, 294
firmware monitor, 303
   diagnostic menu, 290
   entering, 288
   startup menu, 288
firmware options, setting, 175, 182
firmware revision, displaying, 159
flash memory
   managing from a host system, 271
   reclaiming space, 175, 188
**Index**

**flash_compact command**, 188
**flash_log command**, 189
**flash_save command**, 190
**flash_update command**, 192
flow control
  input, 73
  input, 119
  output, 133
  ppp, 219
frame relay
  displaying status, 193
  service configuration, 81
frame size, PPP, 145
frame type, 122
**frame_relay keyword**, 116
**frstats command**, 193
FTP, definition, 294
G
gateways, definition, 295
H
hangup command, 195
hayes compatible modem, dial-in, configuring serial port, 27–28
heap memory, displaying, 180
heap size, 202
**help command**, 196
**host keyword**, 117
host name list, 117
host systems
  accessing serial devices attached to RCS/4000, 263
  AIX, setting up rtn based printer, 59
  configuring for a remote printer, 54
  managing flash memory from, 271
  UNIX, setting up rtn based printer, 59
  utilities, 263
host table, 294
hostname, RADIUS, 148
HTTP server, displaying statistics, 197
**http_stats command**, 197
I
IAC BRK, telnet, 162
ICMP packet, details, 211
ICMP protocol, 210
**ifconfig command**, 198
**ignorecd keyword**, 118
inactivity time-out, 74, 167
inetd service, UNIX, 273
inetd.conf file, 274
Initial Routes table, 151
input flow control, 73, 119
**input keyword**, 119
interface keyword, 121
interface option, wan keyword, 173
interface, network, assigning address and parameters, 198
Internet Protocol (IP), definition, 292
Internet Service Provider, communication via, 130
internet-to-ethernet address translation table, 181
Internet-to-private network address translation, 130
invalidated records, removing from flash memory, 188
IP address
  assigning, 17–21
  assigning permanent, 122
  obtaining from server, 182
**ip keyword**, 122
**iroute keyword**, 151
ISDN terminal adapter
  equipment required, 40
  network diagram, sync port, 42
  setting global parameters, 41
  setting up dial on demand, 45
  setting up, asynch port, 40
  setting up, synchronous port, 42
J
jumps, in routing, 151
K
KERMIT, 247
kernel heap, size, 202
kernel memory, displaying free, 202
**kill command**, 201
L
LAN
  definition, 291
ldterm STREAMS module, 207, 218
LED
  at power up, 17
  boot, 87
  configuration problem, 228
  modem connections, 46
  yellow and green alternating, 228
limit, power, 141
Link Management Interface (LMI), 116
LLC layer, 292
Local Area Networks (LAN), definition, 291
local COM ports, 13
location information, adding, 159
**location keyword**, 158, 159
log, messages
  displaying, 189
  sending to server, 243
logical port numbers, 153
login banner, 100, 105
login script, 163, 277
login, remote, 229
logout, telnet command, 247
**lookmem command**, 202
**lookstream command**, 203
loopback interface, displaying information, 208
lpd
  configuring ports for, 78
  using RCS/4000 as print server, 53
vs reverse telnet for printing, 53
lpq command, 55, 205
lprm command, 55, 206

M
MAC address, 18–20, 190
mac keyword, 123
MAC layer, 292
Management Information Base (MIB), 101, 281
mask keyword, 124
mask, subnet, when assigning IP address, 18
memory dump
  automatic, 182
  manual, 185
memory statistics, displaying, 202
memory, flash
  managing from a host system, 271
  reclaiming space, 175, 188
memory, heap, displaying, 180
menus, 1-key, 105
message log
  displaying, 189
  sending to server, 243
MIB, 281
MIB object ID, adding to an SNMP community, 170
MIB objects, 170
model keyword, 125
model number, 159
modem
  accessing, 29–30
  adding dial-in to RCS/4000, serial port, 27–28
  adding dial-out to RCS/4000, serial port, 28–29
  configuring, 28
  configuring for auto answer, 27
  defining for dial-on-demand, 34, 37
  installing, 28
  modem initialization string, 126
modem keyword, 126
modem name, 126
modem pool
  adding dial-out to RCS/4000, 30
  configuring ports into, 128
  connecting to, 207
modem scripts, 277
modem signals
  DCD, 118
Motorola BitSurfr, setting up, 40
mpool command, 207
mpool keyword, 128
multiple page display, 26, 104
multiple sessions
  in custom configurations, 106
  refreshing screen display, 104
  switching between, 74
  switching between, 162

N	nameserver keyword, 129
  nat keyword, 130
NativeCOM, 13
net capabilities level, 108
netmask, 220
netstat command, 208
network
  layers, 292
  point-to-point, 291
  private/public communication, 130
  protocols, 292
  topologies, 291
Network Address Translation (NAT), 130
network diagram
  CSU/DSU, between two networks, 48
  dial-on-demand, 32, 36
  ISDN, between two networks, 42
network interface
  assigning address and parameters, 198
  displaying statistics, 208
Network Management Station (NMS), 158, 168
network route table, 151
Networked COM ports, 13
NMS, 158, 168

O
onepacket mode, 199
Open Systems Interconnection (OSI) Model, 292
open, telnet command, 248
operational commands summary table, 175
OS, reload at reset, 182
output flow control, 133
output keyword, 133
owner keyword, 135

P
packet size for PPP, 145
packets, damaged and duplicate, 211
parallel printer
  connecting to RCS/4000, 51
parity keyword, 136
password
  administrative, 18, 71
  PPP, 147
  set, 137
  user, 18, 70
Password Authentication Protocol (PAP), 147, 301
peers, 147
permanent virtual circuit (PVC), 81
phone number, dial on demand, 163
physical interface type, WAN port, 47
ping command, 210
point-to-point networks, 291
Point-To-Point Protocol (PPP), 300
pool, IP addresses, 146
pool, modem
Index 307

adding dial-out to RCS/4000, 30
configuring ports into, 128
pool, printer
configuring ports into, 143
connecting to, 218
setting up, 61
port
current, displaying info, 259
displaying status, 213
reserve a, 73
reserve a, 135
source TCP/IP port number for telnet, 245
terminating sessions, 184
port configurations, serial
modifiable, 76
modifiable, 139
predefined, 75
port keyword, 139
port numbers
logical, 153
specifying with dtelnetd, 273
port, asynch
setting up ISDN on, 40
port, parallel
connecting printer to, 51
port, serial
applying power to, 216
characteristics and services, 69
configuring for dial-in modem, telnet or ppp, 27–28
configuring for power, 141
configuring for printer, 53
modifying characteristics, 71
modifying settings, 72
overview, 69
service configuration overview, 78
setting for RS-232, RS-422, RS-485, current loop, 73, 121
setting speed, 160
traffic monitoring, 285
port, synchronous
setting up ISDN, 42
specifying physical interface type, 173
port, WAN
characteristics and services, 77
clocking, 47
set up for remote routing, 46
specifying physical interface type, 173
port_stat command, 213
portdesc keyword, 140
power command, 216
power keyword, 141
power limit on serial ports, 141
power on, 17
ppool command, 218
ppool keyword, 143
PPP
address pool, 146
authentication database, 147
configuration, enabling, 144
definition, 300
interface statistics, displaying, 222
service configuration, 80
ppp command, 219
ppp keyword, 80, 144
pppadlist keyword, 146
pppauthlist keyword, 147
pppstats command, 222
print jobs
deleting, 55, 206
print queue
displaying, 55, 205
names, host system, 54
names, RCS/4000, 54
print server, using, RCS/4000 as, lpd, 53
printenv command, 223
printer
configuring the host for remote printing, 54
connecting to RCS/4000, 51
testing, 226
printer pool
configuring ports into, 143
connecting to, 218
setting up, 61
printer protocols, 52
printer test command, 52
printer, remote
setting up in AIX4, 55–56
setting up in UNIX, 56–57
setting up in Windows NT 3.51, 57
setting up in Windows NT 4.0, 58
setting up on Windows 95, 58
printer, serial
connecting to RCS/4000, 52
troubleshooting, 62
printing
reverse telnet vs lpd, 53
specifying data file format, 55
using reverse telnet (rtn), 59–60
process ids, displaying, 262
process, kill, 201
processes command, 224
processes related information, displaying, 224
product/manifesting data section, 190
profile, community, 101
prtr_test command, 226
pseudo-tty, 263
pty, 263
PVC, 114

Q
query host or gateway, 210
quit command, 227
quit, telnet command, 248

R
r4000 utility, 271
radius keyword, 148
RADIUS server, adding, 148
raw reverse telnet, 154, 267

Index 307
Point-to-Point Protocol (PPP)

RCS/4000
- enabling optional features, 261
- functions, 13
RCS/4000 software, upgrade, 63
**reboot command**, 228
- reboot from host system, 271
- records, error and invalidated, removing from flash memory, 188
- refreshing the screen display in multiple sessions, 26, 104
- reload OS at reset, 182
- remote host, connecting to, 245
- remote login, 229
- remote printer
  - setting up in AIX4, 55–56
  - setting up in UNIX, 56–57
  - setting up in Windows NT 3.51, 57
  - setting up in Windows NT 4.0, 58
  - setting up on Windows 95, 58
- remote routing over the WAN, 46
- remote system, defining for dial on demand, 163
- reset, 228
- restart, 228
- restart delay, 105
- restore command, 289
- reverse telnet
  - accessing modems in pool, 32
  - printing, 59–60
  - using to access dial-out modem, 30
  - vs lpd for printing, 53
- reverse telnet connections
  - logical port numbers, 153
- reverse telnet, raw, 154, 267
RIP
- definition, 297
- message format, 297
**rip keyword**, 151
**rlogin command**, 229
**route keyword**, 151
**router**
- definition, 295
- routing, 151
Routing Information Protocol (RIP), 151, 297
- routing table
  - adding routes to, 151
  - description, 296
  - displaying, 208
RS-232
- set port to, 121
- RS-232, setting WAN port to, 173
RS-422
- set port to, 121
RS-422, setting WAN port to, 173
RS-485
- set port to, 121
**rtelnet keyword**, 153
**rtelnet service**, 79
**rtn**
- printing, 59–60
- rtn (rtelnet) utility, 263–75
  - AIX, 267
- compared to dtelnetd utility, 273
- installing, 265
**rtn command**, 231
**rtsets**, 119, 133

S

**save command**, 232
- screen display, multiple sessions, 26, 104
- self-test, 17
- send, telnet command, 248
Serial Line Interface Protocol (SLIP), 299–300
- serial port
  - applying power to, 216
  - characteristics and services, 69
  - configuring for dial-in modem, 27–28
  - configuring for power, 141
  - configuring for printer, 52, 53
  - modifying characteristics, 71
  - modifying settings, 72
  - overview, 69
  - service configuration overview, 78
  - setting for RS-232, RS-422, RS-485, current loop, 73, 121
  - setting speed, 160
  - traffic monitoring, 285
- serial port configurations
  - modifiable, 76
  - modifiable, 139
  - predefined, 75
- serial printer
  - connecting to RCS/4000, 52
  - troubleshooting, 62
- server, BOOTP
  - assigning IP address with, 20
  - UNIX, assigning IP address with, 65–83
- server, DHCP
  - assigning IP address with, 20
- server, RADIUS, 148
- service configurations
  - default, 111
  - default, 79
  - defining capabilities access levels for, 107
  - frame relay, 81
  - ppp, 80
  - PPP, 144
  - slip, 79, 156
  - user sessions, 82
- sessions
  - hanging up, 195
  - multiple, in custom service configurations, 106
  - multiple, screen refreshing, 104
  - switching between, 74
  - switching between, 162
  - terminating on port, 184
- set command, 88
**setenv command**, 233
- settings, configuration
  - saving, 89
**shell command**, 234
Simple Network Management Protocol (SNMP), 158
size keyword, 155
SLIP, 299–300
SLIP configuration, 79, 156
slip keyword, 156
SNMP, 158
SNMP community list, 101
snmp keyword, 158
SNMP walk command, 282
software
saving to a server, 190
 updating, 192
 uploading updates from host system, 271
software identification and revision level, 159
software tracing
turning off, 257
turning on, 258
software, RCS/4000, upgrade, 63
speed keyword, 160
statistics
 device operation, 281
 frame relay, 193
 HTTP server, 197
 memory, 202
 network interface, 208
 packet loss, 211
 port status, 213
 PPP, 222
 processes, 224
 STREAMS, 203
 status checking, host or gateway, 210
 stop bit, 161
stop keyword, 161
STREAMS statistics, displaying, 203
stty command, 237
subnet mask number, 124
subnet mask table, 124
switch keyword, 162
synchronous port
 setting up ISDN, 42
 specifying physical interface type, 173
 synchronous serial port, 77
 syslog command, 243
system keyword, 163
system, remote, defining for dial on demand, 163

T
task, custom, 102
TCP connections, dropping via SNMP, 281
TCP listener, 165
TCP port numbers
dtelnet, 273
 reverse telnet, 153
TCP, definition, 293
TCP/IP, definition, 292
tcp_service keyword, 165
tcpConnTable, 281
telnet command, 245
telnet commands, 247–53
telnet connections
 enabling ports to accept, 153
source TCP/IP port number, 245
 specifying PTY for, 273
telnet, IAC BRK, 162
telnet, reverse (rtn)
 accessing modems in pool, 32
 logical port numbers, 153
 printing, 59–60
 using to access dial-out modem, 30
term command, 254
term keyword, 166
terminal
 assigning an IP address using, 18
 connecting to host via RCS/4000, 23
 displaying and setting characteristics, 237
 multiple connections via RCS/4000, 25
terminal type
 list of supported, 104
 setting for custom configurations, 104
 setting local variable, 254
terminal types, 166
timeout keyword, 167
tip(1C), 263
to_admin, capabilities level, 108
trace buffer
 clear, 255
 display, 256
trace messages, port, 286
trace_clear command, 255
trace_display command, 256
trace_off command, 257
trace_on command, 258
traffic, serial port
 monitoring, 285
 trailer encapsulations, 181
 trailer link level encapsulation
 option for ifconfig command, 199
 transfer, file
 using KERMIT or XMODEM, 247
 translation table, internet-to-ethernet addresses, 181
trap keyword, 168
traps option for snmp keyword, 158
troubleshooting
dial-on-demand, 40
 network hosts and gateways, 211
 serial printer, 62
TTL value, 212
tty command, 259

U
Unix host systems
 running reverse telnet on, 30
UNIX host systems
 enabling BOOTP, 65
 password requirements for rlogin, 229
 rtelnet (rtn) on, 263
 setting up remote printer, 56
 setting up rtn based printer, 59
 specifying data file format for printing, 55
 TTL field values (ping), 212
unsetenv command, 260
upgrade command, 261
upgrade RCS/4000 software, 63
upload configurations and software from host systems, 271
user access level, 107
user activity, displaying, 262
user keyword, 137
user password, 18, 70
uucp(1C), 263

V
V.120 Asynchronous Mode, 41
V.35
setting WAN port to, 173
Van Jacobsen compression, 35, 38, 45, 46, 49, 113, 114, 220
version keyword, 172
view keyword, 170
voltage, on serial ports, 141

W
walk command, SNMP, 282
WALK FREE LIST, 202
wan clock, setting, 173

wan keyword, 173
WAN port
characteristics and services, 77
clocking, 47
set up for remote routing, 46
specifying physical interface type, 173
WAN, definition, 292
web browser, configuring with, 14
web browser, using to change configuration settings, 88
who command, 262
Wide Area Networks (WAN), definition, 292
Windows 95
setting up remote printer, 58
Windows NT
setting up remote printer (3.51), 57
setting up remote printer (4.0), 58
specifying data file format for printing, 55
using lpd for printing, 52

X
xany, 133
XMODEM, 247
xoff, 119
xon, 133