PortServer II ®
Command Reference Manual

92000246C

http://www.dgii.com

Information in this document is subject to change without notice and does not represent a commitment on the part of Digi International.

Digi provides this document “as is”, without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of fitness or merchantability for a particular purpose. Digi may make improvements and/or changes in this manual or in the product(s) and/or the program(s) described in hsi manual at any time.

This product could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes may be incorporated in new editions of the publication.

Table of Contents

About This Manual

Purpose

The purpose of this reference manual is to provide descriptions of all PortServer II commands and command fields, which—along with the rest of the PortServer II library—should enable those responsible for setting up, maintaining, and using PortServer II to complete these tasks.

Audience

This manual is intended primarily for those who configure and administrator PortServer II, though some parts of the manual describe commands that users may execute as well.

Scope

This manual provides reference information on commands and command fields. It does not provide task-oriented information, which can be found in the other manuals in the PortServer II library.

chapter **1**

Introduction to PortServer II Commands

Introduction

This chapter provides information on using PortServer II commands. It discusses the following topics:

About the Command Line Interface

Introduction

This section discusses the PortServer II command line interface. It provides information on the following topics:

- The keys you use to navigate along the command line and edit commands
- PortServer II on-line help
- Tips on abbreviating PortServer II commands

Navigation and Editing Keys

Use the following keys to navigate along the command line and edit PortServer II commands:

Action	Keys
Move the cursor back one space	Ctrl b
Move the cursor forward one space	Ctrl f
Delete the character to the left of the cursor	Back space
Delete the character under the cursor	Delete
Delete the character to the left of the cursor	Ctrl h
Scrolls back through commands	Ctrl p
Scrolls forward through commands	Ctrl n
Executes the command typed on the command line	Enter

Online Help

On-line help is available for PortServer II commands. The following describes how to access help:

For information on...	Type
All PortServer II commands	? (with no additional parameters)
A specific command	The command and then ? **Example:** info ? **Example:** set user ?

Abbreviating Commands

All PortServer II commands can be abbreviated. You need only supply a sufficient number of command letters to uniquely identify the command.

Manual Organization and Conventions

Organization of Command Information

Commands are listed in alphabetical order. Each command description contains the following topics:

- Introduction, which describes the
  - — Purpose of the command
  - — Privileges required to execute the command
  - — Related information
- Command Syntax, which describes how you issue the command. Often Command Syntax is divided into separate discussions on how you use the command to accomplish a specific purpose. For example, the syntax discussion on the `set logins` command is divided into separate discussion on
  - — Using the command to display the logins table
  - — Using the command to configure login parameters
- Command Fields, which provides a description of each command field.
- Command Examples, which are examples of how the command is used.

In addition, when necessary, some command descriptions provide

- Additional information on the purpose of the command or some aspect of the command that cannot adequately be discussed elsewhere. The heading that identifies these discussions starts with the word "About." For example, the discussion on the `set route` command includes a topic called "About the Route Table."
- A description of the output that results from issuing the command. These descriptions are provided when the description of output fields is not the same as the description of command (input) fields. The `info` command is a good example.

Syntax Conventions

Presentation of command syntax in this manual follows these conventions:

- Brackets ([]) surround optional material.
- Braces ({}) surround entries that require you to chose one of several options, which are separated by the UNIX pipe (|).
- Non-italicized text indicates literal values, that is, fields or values that must be typed exactly as they appear. `Yes` and `no` options are examples of literals.
- Italicized text indicates that a type of information is required in that field. For example, *filename*, means that the name of a file is required in the field.

chapter **2**

PortServer II Commands

Introduction This chapter provides a description of each PortServer II command.

admin

Introduction

Purpose Use the `admin` command to temporarily access commands reserved for administrators (root) when you have logged in as a normal (non-root) user.

About the admin Command After issuing the `admin` command, PortServer II prompts you to supply the root password.

Required Privileges Normal users can issue the `admin` command.

Related Information For information on ending temporary root sessions, see the `exit` and `quit` commands.

Command Syntax

Syntax Here is how you issue the `admin` command:

```
admin
```

Command Example

Example In this example, the `admin` command initiates the following sequence

1. PortServer II displays a prompt requesting the root password.
2. The user types in the root password.
3. If the password is
   - Accepted, the PortServer II displays the root prompt and the user can issue commands reserved for administrators.
   - Not accepted, the PortServer II displays the following:
     `Incorrect password`

```
admin
```

boot

Introduction

Purpose

Use the `boot` command to do any of the following:

- Reboot PortServer II
- Restore the PortServer II configuration to factory defaults
- Load a new PortServer II OS into flash ROM from a TFTP host

Required Privileges

Administrator (root) privileges are require to use the `boot` command.

Related Information

See the `cpconf` for information on saving your current configuration to a host prior to restoring the configuration to defaults.

Warning!

Be very careful with the `boot load` command and option. If this operation fails and then you reboot your PortServer II, the unit may become inoperative. To ensure success, do the following when you intend to use the `load` option. (1) Attempt to boot from a remote firmware image before issuing the `boot load` command. See the `set config` command for more information. (2) After issuing the `boot load` command, ensure that you receive the message "`The image in flash now appears valid`." If you do **not** receive this message, do **not** reboot the PortServer II. Call technical support for instructions on what to do next.

Command Syntax

Reboot Syntax

Here is how you use the `boot` command to reboot PortServer II:

```
boot action=reset
```

Restore Configuration Defaults

Here is how you use the `boot` command to restore the PortServer II default configuration:

```
boot action=eewrite
```

Load New OS Syntax

Here is how you use the `boot` command to load a new OS into flash ROM from a TFTP host:

```
boot load=host-ipaddr:file
```

Command Fields

Field Descriptions

action=eewrite
resets the configuration to factory defaults stored in flash ROM. If you use this option, any configuration information previously entered will be lost.

action=reset
reboots PortServer II

load=*host-ipaddr*:*file*
is an IP address and file name that identifies a source host and file for the new PortServer II OS, which is then burned into flash ROM. To use this option, the host specified must be running TFTP.

Command Examples

Using Factory Defaults

In this example, the `boot` command reloads the OS stored in flash ROM and resets PortServer II configuration to factory defaults.

```
boot action=eewrite
```

Using the Current OS and Configuration

In this example, the `boot` command reboots the PortServer II and uses the current OS and configuration stored in flash ROM.

```
boot action=reset
```

Using a Boot Host

In this example, the `boot` command loads the OS stored on the host and file specified into PortServer II flash ROM. If you want to use this new OS, you must reboot PortServer II.

```
boot load=198.150.150.10:os-1
```

close

Introduction

Purpose Use the `close` command to close your own telnet sessions.

Required Privileges Normal users and administrators (root) can issue the `close` command.

Related Information None.

Command Syntax

Introduction Here is how you issue the `close` command:

Syntax

```
close {* | connection-number}
```

Command Fields

Field Descriptions

*****
specifies that all telnet sessions be closed

connection-number
identifies the session to close

Command Example

Example In this example, the `close` command closes session 2.

```
close 2
```

cpconf

Introduction

Purpose Use the `cpconf` command to do the following:

- Restore the configuration from a remote host or terminal
- Copy the configuration to a remote host or terminal
- Display the configuration on the administrative terminal

Required Privileges The `cpconf` command requires root privileges.

Related Information None.

Command Syntax

Syntax Here is how you issue the `cpconf` command:

`cpconf {fromhost` *`host file`*`|tohost` *`host file`* `|term}`

Command Fields

Field Descriptions

fromhost *host file*
copies the configuration to PortServer II from the host and file specified. When you use this field, remember to do the following:

- Identify the host by either its IP address or DNS name.
- Separate host and file fields by spaces.

tohost *host file*
copies the configuration to the host and file specified. When you use this field, remember to do the following:

- Identify the host by either its IP address or DNS name.
- Separate host and file fields by spaces.

Note: TFTP must be running on the host specified on the fromhost and tohost fields.

term
displays the configuration file on the administration terminal

Command Examples

Copying From a Host

In this example, the `cpconf` command copies the configuration from the host and file specified.

```
cpconf fromhost 190.150.150.10 ps-cnfg1
```

Copying To a Host

In this example, the `cpconf` command copies the configuration to the host and file specified.

```
cpconf tohost 190.150.150.10 ps-cnfg1
```

Copying To the Administrative Terminal

In this example, the `cpconf` command, displays the configuration on the administrative terminal.

```
cpconf term
```

exit

Introduction

Purpose Use the `exit` command to terminate

- Your current PortServer II session
- A temporary root session. If you are in a root session started with the `admin` command, `exit` returns you to a regular session.

Required Privileges Anyone can execute the `exit` command.

Related Information See the following:

- The `admin` command for information on starting a temporary root session
- The `quit` command for an alternate method of ending a session

Command Syntax

Syntax Here is how you issue the `exit` command:

```
exit
```

Command Example

Example In this example, the `exit` command ends the current session.

```
exit
```

info

Introduction

Purpose

Use the `info` command to

- Display PortServer II network statistics tables
- Clear network statistics tables

About Network Statistics Tables

The statistics in network statistics tables are those gathered since the tables were last cleared.

Required Privileges

Normal users can view network statistics tables. Administrator (root) privileges are required to clear them.

Related Information

None.

Command Syntax

Clear Syntax

Here is how you use the `info` command to clear network statistics tables:

```
info clear [table_name]
```

Display Syntax: Frame Relay Statistics

Here is how you use the `info` command to display the frame relay network statistics table:

```
info frame:range:dlci-range
```

Display Syntax: All Other Network Statistics

Here is how you use the `info` command to display statistics for IP, ICMP, ethernet, TCP, and UDP.

```
info table_name
```

Command Fields

Field Descriptions

clear | clear *table_name*
clears either (1) all network statistics tables (when no particular table is specified) (2) the specified table, which can be the IP, ICMP, ethernet, TCP, or UDP table

frame:*range*:*dlci-range*
displays information on the frame relay port or ports and DLCIs specified

table_name
is one of the following tables:

table_name	Contents
ip	IP statistics
icmp	ICMP statistics
network	Statistics collected on the ethernet interface
tcp	TCP statistics
udp	UDP statistics

Command Examples

Displaying the IP Table

In this example, the `info` command displays the IP table.

```
info ip
```

Displaying Frame Relay Statistics

In this example, the `info` command displays frame relay statistics for the ports and DLCIs specified.

```
info frame:4-5:17-26
```

Clear All Network Statistics Tables

In this example, the `info` command clears all network statistics tables.

```
info clear
```

Command Output: Frame Relay Fields

Introduction

This section describes the fields displayed when you issue the `info frame` command.

Frame Relay Field Descriptions

Link Index
the number of the table entry

frCircuitReceivedFrames \ frCircuitSentFrames
frames received and sent over this virtual circuit

frCircuitReceivedOctets \ frCircuitSentOctets
octets received and sent over this virtual circuit

Received Fragments \ Sent Fragments
fragments received and sent over this virtual circuit

Reassembled Frames
frames successfully re-assembled from fragments

Sent Fragmented
frames fragmented before sending

Reassemble Failures
failures to re-assemble fragments into complete frames

frCircuitReceivedBECNs \ FrCircuitReceivedFECNs
frames received with BECN (backward explicit congestion notification) and FECN (forward explicit congestion notification) messages

frErrType
type of error last seen on this interface

frErrFaults
times that traffic was stopped on this circuit due to LMI errors

frErrFaultTime
time at which an error was detected

receive errors
received frames with errors

undefined errors
detected errors not defined in the Frame Relay MIB (RFC 1315)

too long
frames received exceeding the maximum frame size on this circuit

too short
received packets smaller than the minimum frame relay packet size

bad DLCI
frames received with an invalid DLCI

unknown DLCI
received frames with a DLCI identifying an unconfigured PVC

undefined LMI error
received LMI packets not matching the LMI specification

LMI unknown IE
information elements in received LMI packets with an unrecognized type code

LMI bad sequence
LMI packets received with a bad sequence number

LMI unknown report
LMI reports received with an unrecognized type code

frErrData
portion of the frame that caused the error

status requests
LMI status requests sent

full status requests
full status requests sent

status responses
LMI status responses sent

full status responses
full status requests received

rcvd sequence number
last sequence number received

sent sequence number
last sequence number sent

updates or async status
updates or async status reports received. This value depends on the type of LMI used.

frames/octets sent within CIR
frames/octets sent within the Committed Information Rate

frames/octets sent beyond CIR
frames/octets sent in excess of the CIR

frames/octets buffered
frames/octets held for later transmission to avoid exceeding the CIR

frames/octets dropped
frames/octets discarded to avoid exceeding the CIR

Command Output: ICMP Fields

Introduction

This section describes the fields displayed when you issue the `info icmp` command.

ICMP Field Descriptions

icmpInMsgs
ICMP messages received, including those counted by `icmpInErrors`

icmpInEchos
ICMP Echo Request messages received

icmpInEchoRp
ICMP Echo Reply messages received

icmpInDstUnrec
ICMP Destination Unreachable messages received

icmpInRedirect
ICMP Redirect messages received

icmpInParmProb
ICMP Parameter Problem messages received

icmpInTimeExcd
ICMP Time Exceeded messages received

icmpInSrcQuenc
ICMP Source Quench messages received

icmpInTimest
ICMP Timestamp Request messages received

icmpInTimestRp
ICMP Timestamp Reply messages received

icmpInAdrMsk
ICMP Address Mask Request messages received

icmpInAdrMskRp
ICMP Address Mask Reply messages received

icmpInErrors
ICMP messages received with ICMP-specific errors (for example, bad ICMP checksums or length)

icmpOutMsgs
ICMP messages that PortServer II attempted to send, including those counted by `icmpOutErrors`

icmpOutEchoRp
ICMP Echo Reply messages sent

icmpOutEchos
ICMP Echo Request messages sent

icmpOutDstUnre
ICMP Destination Unreachable messages sent

icmpOutRedirec
ICMP Redirect messages sent

icmpOutParmPro
ICMP Parameter Problem messages sent

icmpOutTimeExc
ICMP Time Exceeded messages sent

icmpOutSrcQuen
ICMP Source Quench messages sent

icmpOutTimestR
ICMP Timestamp Reply messages sent

icmpOutTimest
ICMP Timestamp (request) messages sent

icmpOutAdrMskR
TICMP Address Mask Reply messages sent

icmpOutAdrMsk
ICMP Address Mask Request messages sent

Command Output: IP Statistics

Introduction

This section describes the fields displayed when you issue the `info ip` command.

IP Field Descriptions

ipInReceives
incoming datagrams, including any received in error

ipInHdrErrors
incoming datagrams discarded due to IP header errors. Causes include bad checksums, version number mismatches, other format errors, time-to-live values exceeded, and errors discovered in processing IP options. Correctly configured networks produce few such errors.

ipInAddrErrors
incoming datagrams discarded because the address in the IP header destination field was not valid for PortServer II's network. This includes addresses of unsupported classes (Class E, for example). Correctly configured networks produce few such errors.

ipInUnknownProtos
datagrams received successfully but discarded because of an unknown or unsupported protocol

ipInDiscards
good incoming datagrams discarded for lack of resources, such as buffer space, including those discarded while awaiting re-assembly

ipReasmOKs
IP datagrams successfully re-assembled

ipReasmFails
failures detected by the IP re-assembly algorithm. This is may not be a count of all discarded IP fragments because some algorithms (notably the algorithm in RFC 815) lose count by combining fragments as they are received.

ipForwDatagram
incoming datagrams destined for another subnetwork to which PortServer II's could not find a route

ipOutNoRoutes
outgoing datagrams discarded because no route could be found to their destination. This includes datagrams:

- Counted in `ipForwDatagrams`
- That a host could not route because default gateways are down

Correctly configured networks produce few such errors.

ipOutRequests
datagrams that local IP user protocols (including ICMP) supplied to IP for transmission, not including those counted in `ipForwDatagrams`

ipOutDiscards
good outgoing datagrams discarded for lack of resources, including those counted in `ipForwDatagrams`

ipFragCreates
datagram fragments PortServer II generated

ipFragOKs
datagrams successfully fragmented

Command Output: Network Statistics

Introduction

This section describes the fields displayed when you issue the `info network` command. This command reports activity on the ethernet interface.

Network Statistics Field Description

ifInOctets
octets received, including framing characters

ifInUcastPkts
subnetwork unicast packets delivered to higher-layer protocols

ifInNUcastPkts
non-unicast (for example, subnetwork-broadcast or subnetwork multicast) packets delivered to a higher-layer

ifInDiscards
inbound packets discarded, even though no error was detected that would prevent delivery to a higher-layer

ifInErrors
inbound packets with errors that prevent delivery to a higher-layer

ifUnknownProtos
inbound packets discarded because of unknown or unsupported protocols

ifOutOctets
Octets transmitted, including framing characters

ifOutUcastPkts
outbound packets using the subnetwork unicast address, including discards

ifOutNUcastPkts
outbound packets using a non-unicast (that is, a subnetwork broadcast or subnetwork multicast) address, including discards

ifOutDiscards
error-free outbound packets discarded, possibly to free buffer space

ifOutErrors
outbound packets not transmitted because of errors

In Total
frames received

In IP
IP protocol frames received

In ARP
ARP frames received

Out Total
frames sent by PortServer II

Out IP
IP frames sent

Out ARP
ARP frames sent

In Overruns
times the Ethernet controller was unable to place a received frame in memory

In Unaligned
misaligned frames received

In No Resource
incoming frames not processed due to lack of available buffers

In Collision
Ethernet collisions detected after a destination address was received

In Short Frame
short frames received

In Bad CRC
frames received with bad CRC

Out No Carrier
frames lost when lack of carrier was detected

Out Lost CTS
frames lost when `ClearToSend` was reset

Out DMA Underrun
frames lost because transmit buffers were not available

Out Deferred
transmissions deferred

Out Collisions
Ethernet collisions detected after starting a transmission

Command Output: TCP Statistics

Introduction

This section describes the fields displayed when you issue the `info TCP` command.

TCP Field Descriptions

tcpInSegs
segments received, including those received in error. This includes only segments received on currently established connections.

tcpInErrs
segments received in error (for example, bad TCP checksums)

tcpEstabResets
times that TCP connections made a direct transition to the CLOSED state from either the ESTABLISHED or CLOSE-WAIT states

tcpPassiveOpen
times that TCP connections made a direct transition to the SYN-RCVD state from the LISTEN state

tcpAttemptFail
times that TCP connections made a direct transition to the CLOSED state from either the SYN-SENT state or the SYN-RCVD state, plus the times TCP connections made a direct transition to the LISTEN state from the SYN-RCVD state

tcpOutSegs
segments sent, including those on current connections. This excludes those containing only retransmitted octets.

tcpRetransSegs
segments retransmitted, that is, the number of TCP segments transmitted containing one or more previously transmitted octets

tcpOutRsts
TCP segments sent containing the RST flag

tcpActiveOpens
times TCP connections made a direct transition to the SYN-SENT state from the CLOSED state

Command Output: UDP Statistics

Introduction

This section describes the fields displayed when you issue the `info UDP` command.

UDP Field Descriptions

udpInDatagrams
datagrams delivered to UDP users

udpInErrors
received UDP datagrams that could not be delivered for any reason other than the lack of an application at the destination port

udpNoPorts
received UDP datagrams for which there was no application at the destination port

udpOutDatagrams
UDP datagrams sent

kill

Introduction

Purpose Use the `kill` command to clear or reset a TTY session on a selected port.

Required Privileges The `kill` command requires root privileges.

Related Information None

Command Syntax

Syntax Here is how you issue the `kill` command:

`kill tty=`*`tty-number`*

Command Field

Field Description

tty
is the number of the port on which to clear a session

Command Examples

Killing a TTY Session In this example, the `kill` command clears TTY session 8:

```
kill tty=8
```

mode

Introduction

Purpose Use the `mode` command to change or display the operating parameters for a current telnet session.

Required Privileges Anyone can issue the `mode` command.

Related Information None.

Command Syntax

Change Syntax Here is the form of the `mode` command used for changing telnet operating parameters:

```
mode [bin={on|off}][crmod={on|off}][crlf={on|off}]
```

Display Syntax Here is the form of the `mode` command used for displaying the operating parameters of the current telnet session.

```
mode
```

Command Fields

Field Descriptions

bin

`on`
means that binary mode is `on`, that is, all transmitted and received characters are converted to binary during this telnet session

`off`
means that binary mode is `off` for this telnet session

The default is `off`.

crmod

`on`
means that line feed characters are added to received carriage return characters

`off`
means that line feed characters are **not** added to received carriage return characters

The default is `off`.

crlf

`on`
means that line feed characters are added to transmitted carriage return characters

`off`
means that line feed characters are **not** added to transmitted carriage return characters

The default is `off`.

Command Examples

Turning Binary Mode On

In this example, the `mode` command turns binary mode on.

```
mode binary=on
```

Adding Line Feed Characters

In this example, the `mode` command adds line feed characters to both transmitted and received carriage returns.

```
mode crmod=on crlf=on
```

Displaying Operating Parameters

In this example, the `mode` command displays information on each telnet session. This information includes

- The identity of the originating terminal
- The identity of the host on which the telnet session is running
- The state (on or off) of `mode` command parameters for the telnet session.

```
mode
```

newpass

Introduction

Purpose Use the `newpass` command to create or change

- Your own password (if you are logged in under your own name)
- The root password or another user's password (if you are logged in as root)

Required Privileges Anyone can change his or her own password. Root privileges are required to change someone else's password or the root password.

About the newpass Command When you enter the `newpass` command, PortServer II provides a series of prompts to guide you through the process of changing a password.

Related Information None.

Command Syntax

Syntax Here is the syntax for the `newpass` command:

`newpass [user=`*`username`*`]`

Command Field

Field Description

user
is the name of the user (configured with the `set user` command) whose password will be changed

Command Examples

Changing a Password In this example, the `newpass` command changes a user's password.

```
newpass
```

ping

Introduction

Purpose Use the `ping` command—which requests ICMP echo replies from a specified host or network device—to test if a host or other device is active and reachable.

Required Privileges Anyone can issue the `ping` command.

Related Information None.

Command Syntax

Syntax

```
ping [continuous][fill=char] {hostname | ip-addr}
[intv=msec] [loose_sroute=ip-addr,ip-addr...]
[npkts=num] [pksiz=bytes] [record_route] [verbose]
[strict_sroute=ip-addr,ip-addr...]
```

Command Fields

Field Descriptions

continuous
specifies that pings be sent continuously until stopped. (Press the interrupt keys to stop continuous pings. The default interrupt keys are `<Ctrl-C>`.)

fill
specifies characters to include in the data portion of the echo reply

intv
is the interval in milliseconds between pings

The range is -1 to 60,000, and the default is 1000 milliseconds (one second). -1 means that echoes will be continuously sent until the value in the `npkts` field is reached.

ip-addr | hostname
identifies the target device of the ping (ICMP echo request). Use one of the following to identify this device:

- An IP address
- A domain name

loose_sroute
specifies that the ping must pass through the routers indicated on its way to the target host. These routers are identified by their IP addresses.

npkts
is the number of packets to include with each ping

The range is 1 to 30,000, and the default is 1.

record_route
specifies that each router through which the ping passes record its IP addresses for inclusion in the echo reply

strict_sroute
specifies that the ping must pass through the routers indicated—and only those indicated—on its way to the target host. These routers are identified by their IP addresses.

verbose
specifies that returned echo replies include statistics associated with the ping, such as the roundtrip time and the number of packets transmitted and received

Command Examples

Ping with No fields

In this example, the `ping` command simply determines whether the specified host can be reached.

```
ping 199.150.150.10
```

Loose Source Routing

In this example, the `ping` command specifies loose source routing, which means that the ping must pass through the routers identified on the `loose_sroute` parameter. The ping may, however, pass through additional routers as well.

```
ping 199.150.150.10
loose_sroute=199.150.160.10,190.150.161.10
```

Strict Source Routing

In this example, the `ping` command specifies strict source routing, which means that the ping must pass through the routers identified on the `strict_sroute` field, and only those routers. If it cannot reach the destination along this path, the destination is regarded as unreachable.

```
ping 199.150.150.10
strict_sroute=199.150.160.10,190.150.161.10
```

quit

Introduction

Purpose Use the `quit` command to end

- Your current PortServer II session. If you are in a regular or root session, `quit` closes the session.
- A temporary root session. If you are in a root session started with the `admin` command, `quit` returns you to a regular session.

Required Privileges Anyone can issue the `quit` command.

Related Information `Admin` and `close` commands.

Command Syntax

Syntax Here is the syntax for the `quit` command:

```
quit
```

Command Example

Example In this example, the `quit` command ends either a regular session or a temporary root session.

```
quit
```

rlogin

Introduction

Purpose Use the `rlogin` command to log into a remote system from the PortServer II command line.

Required Privileges Anyone can execute the `rlogin` command.

Related Information None

Command Syntax

Syntax Here is the form of the `rlogin` command used to log into a remote host:

```
rlogin [esc=char] {hostname|host-ip-addr}
[user=user-name]
```

Command Fields

Field Descriptions

esc
is a different escape character than the ~ (tilde) character. This character is used for disconnecting from the remote host.

hostname
is the name of the host on which you want to log in

host-ip-addr
is the IP address of the host on which you want to log in

user
is the user name to use on the remote system. If you do not specify a name, your PortServer II name will be used.

Command Example

Using a Host Name In this example, the *rlogin* command establishes an rlogin session using a host name.

```
rlogin host1
```

Using an IP Address In this example, the *rlogin* command establishes an rlogin session using an IP address.

```
rlogin 192.192.150.28
```

Using a Host Name and User Name

In this example, the *rlogin* command establishes an rlogin session using a host name. The name that identifies the user on the host system is also supplied in the command.

```
rlogin host1 user=fred
```

send

Introduction

Purpose Use the `send` command to send a control command to a telnet peer.

Required Privileges Anyone can issue the `send` command.

Related Information telnet command.

Command Syntax

Syntax Here is the syntax of the `send` command:

```
send {ao|ayt|brk|ec|el|escape|ga|ip|nop|synch}
```

Command Fields

Field Descriptions

ao
sends the “abort output” signal, which discards output buffered on the peer

ayt
sends the “are you there” signal to test whether a host is still active

brk
sends the break signal to interrupt the executing application

ec
sends the “erase character” to delete the previous character

el
sends the “erase line” signal to delete the entire current line

escape
sends the “escape character”

ga
sends the “go ahead” signal

ip
sends the “interrupt process” signal to terminate the program running on the peer

nop
sends the “no option” signal to the peer

synch
sends the “synchronize process” signal to the peer

Command Examples

Send IP

In this example, the `send` command transmits an interrupt process signal.

```
send ip
```

Send AYT

In this example, the `send` command transmits an “are you there” signal.

```
send ayt
```

set altip

Introduction

Purpose

Use the `set altip` command to

- Configure a serial port or group of serial ports with an IP address
- Display current entries in the altip table
- Remove an entry from the altip table

About the set altip Command

PortServer II uses alternate IP addresses to route outbound calls to the correct serial port or group of ports. By associating ports with IP addresses, telnet users on the LAN can use IP addresses, rather than port numbers, to specify a port or range of ports in their telnet calls.

Up to 64 alternate IP address entries are permitted.

Required Privileges

Normal users can use the `set altip` command to view altip table entries. Root privileges are required to configure and remove altip table entries.

Related Information

None.

Command Syntax

Configuration Syntax

Here is the form of the `set altip` command used to configure altip entries:

`set altip group=`{*port#* | *group#*} `ip=`*ip-addr*

Display Entry Syntax

Here is the form of the `set altip` command used to display entries in the altip table:

`set altip [range=`*range*`]`

Remove Entry Syntax

Here is the form of the `set altip` command used to remove entries from the altip table.

`set altip`
`{rmrange range=`*ind-start-ind-end*`]| rmip=`*ip-addr*`}`

Command Fields

Field Descriptions

group
is a port or group of ports

ip
assigns an IP address to the ports or group of ports (hunt group) specified on the `group` field

range
specifies a range of index entries in the altip table

rmip
identifies an alternate IP address to remove

rmrange
removes the range of altip entries specified on the range field

Command Examples

Displaying the Altip Table

In this example, the `set altip` command displays the entire altip table.

```
set altip
```

Displaying Several Entries

In this example, the `set altip` command displays altip table entries 1 through 7.

```
set altip range=1-7
```

Configuring an Entry

In this example, the `set altip` command configures an alternate IP address for the ports specified on the `group` field.

```
set altip ip=198.150.150.10 group=65
```

Removing an IP Address from the Altip Table

In this example, the `set altip` command removes the specified IP address from the altip table.

```
set altip rmip=198.150.150.10
```

Removing a Range of Entries in the Altip Table

In this example, the `set altip` command removes altip table entries 7 through 14.

```
set altip rmrange range=7-14
```

set arp

Introduction

Purpose Use the `set arp` command to

- Manually configure an entry in the Address Resolution Protocol (ARP) Table
- Display the contents of the ARP table
- Remove an entry from the ARP table

About the ARP Table The ARP table contains the ethernet-to-IP address mappings of other devices on the local subnetwork. PortServer II requires these mappings to communicate with these devices. The ARP protocol updates this table automatically, so manual modification is usually not required.

Required Privileges Anyone can view the ARP table. Root privileges are required to configure or remove entries.

Related Information None.

Command Syntax

Configuration Syntax Here is the form of the `set arp` command used to configure entries in the arp table.

`set arp ether=`*`etaddr`* `ip=`*`ipaddr`* `[tim2liv=`*`time`*`]`

Display Syntax Here is the form of the `set arp` command used to display the contents of the arp table.

`set arp [range=`*`range`*`]`

Remove Entry Syntax Here is the form of the `set arp` command used to remove entries from the arp table.

`set arp range=`*`range`* `rmarp=on`

Command Fields

Field Descriptions

ether
specifies the ethernet address of a device

ip
specifies the IP address of a host or device

range
specifies a range of table entries, which are identified by the index field in the ARP table

rmarp
`on`
means remove ARP entries specified on the `range` field

tim2liv
specifies the time, in seconds, to keep an entry in the ARP Table

The range is 0 to 999 seconds. The default is 0, which means the entry will never time out.

Command Examples

Displaying a Range of Entries

In this example, the `set arp` command displays a range of ARP table entries

```
set arp range=1-4
```

Displaying All Entries

In this example, the `set arp` command displays the entire ARP table.

```
set arp
```

Configuring an Entry

In this example, the `set arp` command configures an ARP entry.

```
set arp ip=198.150.150.10 ether=08:00:20:05:0b:da
tim2liv=900
```

Changing the tim2liv

In this example, the `set arp` command configures the period for which an ARP table entry should be maintained.

```
set arp range=1 tim2liv=120
```

Removing a Range of Entries

In this example, the `set arp` command removes a range of entries from the ARP table.

```
set arp range=1-7 rmarp=on
```

set auth

Introduction

Purpose

Use the `set auth` command to

- Configure access permissions to PortServer II serial ports for users making outbound calls
- Display outbound call permission levels to PortServer II serial ports
- Delete a range of entries from the auth table

About the set auth Command

The `set auth` command is a very powerful tool for limiting outbound call access to PortServer II ports. There are, however, a few rules you must understand in order to use this command to produce the configuration results you intend. Here are those rules:

- The default for a port is unrestricted access. This means that all IP addresses and RealPort drivers have unrestricted access to the port to make outbound calls unless you use the `set auth` command to change this.
- When you use the `set auth` command to require a login for a particular IP address (or range of addresses), all other IP addresses continue to have unrestricted access to the port.
- When you use the `set auth` command to grant unrestricted access to a particular IP address (or range of addresses), all other IP addresses are required to login.
- When you use the `set auth` command to specify an IP address and range of ports but no permission levels, the IP address will **not** be able use the port because it has been assigned neither login nor unrestricted access to the port.
- Use the `mask` field to extend the scope of the `set auth` command to a range of IP addresses. In each mask position that a binary 1 appears, the incoming address must match perfectly with the address specified on the `ip` field.
- The `range` field is sensitive to the context in which it is used. When you configure access permissions, it specifies a range of ports. When you display an entry or remove one (using the `rmauth` field) from the auth table, it refers to an auth table index number, which is the way an auth table entry is identified.

The auth table is limited to 20 entries.

Required Privileges

Anyone can use the `auth` command to display auth table entries. Root privileges are required to configure access permissions or to remove entries from the auth table.

Related Information

None.

Command Syntax

Configuration Syntax Here is the form of the `set auth` command used to configure auth table entries.

```
set auth [ip=ipaddress] [login=range] [mask=mask]
[range=range] [realport=range] [unrestricted=range]
[rmauth=ip-address|on]
```

Display Syntax Here is the form of the `set auth` command used to display auth table entries.

```
set auth [range=range]
```

Delete Syntax Here is the form of the `set auth` command used to delete entries from the auth table.

```
set auth range=range rmauth
```

Command Fields

Field Descriptions

ip
is the IP address of the device to which this `set auth` command applies

login
configures login requirements to the range of ports specified for the IP address specified. Users with other IP addresses continue to have unrestricted access to these ports.

mask
specifies an IP mask used to extend the scope of this `set auth` command to a range of IP addresses

See the examples that follow for more information on using the `ip` and `mask` fields together.

range
specifies one of the following:

- A range of ports to which this `set auth` command applies when you configure port access
- A range of auth table entries (identified by an index number) to which this `set auth` command applies when you use the `rmauth` option

realport
configures access for the RealPort drivers running on the devices identified by the `ip` and `mask` fields to the specified range of ports

The default is that RealPort can access a port. Unless you use the `set auth` command to configure a port to restrict Realport access, Re-

alPort drivers can access that port.

unrestricted
configures unrestricted access for the IP address specified to the range of ports specified. Users with other IP addresses must log in.

rmauth
`ip`
is an ip address to remove from the auth table

`on`
makes the command apply to the auth table entries defined on the `range` field

Command Examples

Display the Entire Auth Table

In this example, the `set auth` command displays the entire auth table.

```
set auth
```

Display Setting for a Range of Entries

In this example, the `set auth` command displays a range of auth table entries.

```
set auth range=1-8
```

Configure RealPort Access

In this example of a TCP/IP Class C network, the `set auth` command configures

- RealPort running on any host on network 199.150.150.0 with unrestricted access to ports 1 through 8
- Users with other IP addresses with log in access

```
set auth ip=199.150.150.10 mask=255.255.255.0
realport=1-8
```

Configure Unlimited Access to a Port

In this example, the `set auth` command configures

- Telnet users on host 199.150.150.16 to access port 1 without logging in
- All other users to log in

```
set auth ip=199.150.150.16 mask=0.0.0.0 unrestricted=1
```

Configuring Mixed Access

In this example, the `set auth` command configures

- The user at IP address 199.150.150.16 with log in access to ports 2 and 3 and unrestricted access to ports 4 and 5
- All other users for unrestricted access to ports 2 and 3 and log in access to ports 4 and 5

```
set auth range=2-4 ip=199.150.150.16 login=2-3
unrestricted=4-5
```

Removing an IP Address from the Auth Table

In this example, the `set auth` command removes an entry from the auth table by specifying an IP address.

```
set auth ip=199.150.150.16 rmauth=on
```

Removing an Entry from the Auth Table

In this example, the `set auth` command removes an entry by specifying a range of entries in the auth table.

```
set auth rmauth=on range=1-2
```

set chat

Introduction

Purpose

Use the `set chat` command to

- Configure entries in the chat table
- Display chat table entries
- Remove entries
- Rename entries

About the Set Chat Command

Chat table entries provide telephone number string translation and can be accessed by any script that you configure. The chat table holds a maximum of 12 entries.

Required Privileges

Anyone can display chat table entries. Root privileges are required to configure entries.

Related Information

See the `set script` command for information on creating scripts that use telephone string translation.

Command Syntax

Configuration Syntax

Here is the form of the `set chat` command used to configure chat table entries.

```
set chat [delay=string][name=chat-name] [pound=string]
[range=range] [retry=number] [star=string]
[wait=string]
```

Display Syntax

Here is the form of the `set chat` command used to display chat table entries.

```
set chat [range=range]
```

Remove Syntax

Here is the form of the `set chat` command used to remove a chat table entry:

```
set chat {rmchat=on range=range | rmchat=chatname}
```

Rename Syntax

Here is the form of the `set chat` command used to rename a chat table entry:

```
set chat name=name newname=new-name
```

Command Fields

Field Descriptions

delay
is a string of up to 24 characters to substitute into telephone numbers in place of the delay character

name
configures a name for the chat table entry

pound
is a string of up to 24 characters to substitute into telephone numbers in place of the # character

range
is one of the following:

- A range of ports to which the chat table entry will apply
- A range of chat table index numbers, which identify chat table entries

retry
is the number of times to retry a call. The range is 0 to 99 times.

rmchat
removes the chat table entry specified on the `range` or `name` field

star
is a string of up to 24 characters to substitute into telephone numbers in place of the * character

wait
is a string of up to 24 characters to substitute into telephone numbers in place of the wait character

Command Examples

Displaying the Entire Chat Table

In this example, the `set chat` command displays the entire chat table.

```
set chat
```

Configuring a Table Entry

In this example, the `set chat` command configures a new entry.

```
set chat name=chat1 star=4452624
```

Removing An Entry

In this example, the `set chat` command removes a chat table entry from the chat table.

```
set chat rmchat=chat1
```

Renaming a Chat Table Entry

In this example, the `set chat` command renames the chat table entry.

```
set chat name=chat1 newname=chat2
```

set config

Introduction

Purpose

Use the `set config` command to configure or display entries in the network parameters configuration table, which holds

- PortServer II boot parameters
- PortServer II's IP and ethernet addresses and subnet mask
- The TCP port number for RealPort
- Information on how PortServer II should handle ICMP redirect messages

Required Privileges

Anyone can use the `set config` command to display entries in the network configuration table. Root privileges are required to use this command to configure entries.

Related Information

None.

Command Syntax

Configuration Syntax

Here is the form of the `set config` command used to add and change entries in the network parameter configuration table.

```
set config [bootfile=file] [boothost=host-ipaddr]
[bootp={yes|no|smart}] [bootpserver=server]
[bootpgenericfile=file] [domain=domain]
[ether=ether-addr] [gateway=ip-addr] [ip=ip-addr]
[myname=name] [nameserv=ip-addr]
[ramsize=show][realport=tcp-port]
[redirect={listen|ignore}][romversion=show]
[submask=mask] [tftpboot={yes|no|smart}]
```

Display Syntax

Here is the form of the `set config` command used to display entries in the network parameter configuration table.

```
set config
```

Command Fields

Field Descriptions

bootfile

is the name of a boot file on a TFTP host. See the host's administrator to determine if the full path to the file must be specified to satisfy the TFTP implementation on the host.

boothost

is the IP address of a host from which PortServer II can boot using TFTP

bootp

`yes`
means boot from the bootp host identified on the `bootpserver` field

`smart`
means that if PortServer II cannot boot from the host identified on the `boothost` field, boot from the PortServer II's internal flash ROM instead.

`no`
means boot PortServer II from internal flash ROM

The default is `no`.

bootpgenericfile
is the name of and complete path to the boot file on a bootp host.

bootpserver
is the IP address of a host from which PortServer II can boot using bootp

domain
is the name of PortServer II's domain

ether
is PortServer II's ethernet address. Normally, you do not have to configure this address. Digi recommends that you do not change the ethernet address.

gateway
is the IP address of the default gateway

ip
is PortServer II's IP address. PortServer II can obtain this address from a RARP (Reverse Address Resolution Protocol) server if

- A RARP server is available on the LAN
- PortServer II's ethernet and IP address mappings have been entered on the RARP server

myname
is PortServer II's DNS name

nameserv
is the IP address of a name server in PortServer II's domain

ramsize=show
displays the amount of RAM the PortServer II is equipped with.

realport
is the TCP port number used for RealPort connections. This port number is used by RealPort to establish connections.

The default is 771.

redirect

`listen`
means PortServer II accepts ICMP routing redirect messages. Use this option, only if you have not configured PortServer II to forward RIP packets.

`ignore`
means PortServer II discards ICMP routing redirect messages

The default is `ignore`.

romversion=show
displays the version of the ROM code used by the PortServer II.

submask
is the subnet mask for PortServer II's subnetwork

tftpboot

`yes`
means always boot from the TFTP host identified on the `boothost` field

`smart`
means that if PortServer II cannot boot from the TFTP host identified on the `boothost` field, boot from the PortServer II's internal flash ROM instead.

`no`
means boot PortServer II from internal flash ROM

The default is `no`.

Command Examples

Displaying the Complete Table

In this example, the `set config` command displays the network parameter configuration table.

```
set config
```

Booting from a TFTP Server

In this example, the `set config` command configures PortServer II to boot from the TFTP server and file specified on the `boothost` and `bootfile` fields.

```
set config tftpboot= smart boothost=190.250.150.10
bootfile=bootfle1
```

Booting from a Bootp Server

In this example, the `set config` command configures PortServer II to boot from the bootp server and file specified on the `boothost` and `bootfile` fields.

```
set config bootp=yes bootpserver= 190.250.150.10
bootpgenericfile=bootfle1
```

set device

Introduction

Purpose

Use the `set device` command to

- Configure modems and other devices used for outgoing connections to use dialer scripts and chat table entries
- Configure a different baud rate (line speed) for modems and other devices used for outgoing connections than the rate defined on the `set line` command
- Remove an entry from the device table
- Display the contents of the device table

Required Privileges

Anyone can display the contents of the device table. Root privileges are required to configure devices.

Related Information

See the `set chat`, `set line`, and `set script` commands.

Command Syntax

Configuration Syntax

Here is the form of the `set device` command used to configure entries in the device table:

```
set device [baud={no|rate}]
[chat={no|index-num|chat-name}]
[dialer={no|index-num|script-name}]
name=name ports=range
[newname=newname] [p{1-9}]
```

Remove Syntax

Here is the form of the `set device` command used to remove an entry from the device table:

```
set device
rmdevice={on range=index-range|device=name}
```

Display Syntax

Here is the form of the `set device` command used to display entries from the device table:

```
set device [{range=range|name=name]
```

Command Fields

Field Descriptions

baud

no
means the baud rate specified on the `set line` command will be used

`rate`
is the baud rate (line speed) when this device is used. This field overrides the baud rate (for this device) defined on the `set line` command.

The range is 300 to 115,200 bps, and the default is `no`.

chat

no
means that a chat table entry is **not** associated with this device

index-num
is a chat table entry (index number) associated with this device

`chat-name`
is the name of a chat table entry

The default is `no`.

dialer

no
means that a dialer script is not associated with this device

index-num
is a script table entry (index number) associated with this device

`script-name`
is the name of a script

The default is `no`.

name

is a user-defined name for the device

newname

is a new name for a previously defined device

p{1-9}

are integers (1-9) that can be used in the variable fields of login or dialer scripts.

ports

is the port or range of ports available to this device

range

is a device table entry or range of entries (identified by their index numbers)

rmdevice=on
removes the device specifies on this field and on the `range` field

Command Examples

Displaying the Device Table

In this example, the `set device` command displays the entire device table.

```
set device
```

Displaying an Entry in the Device Table

In this example, the `set device` command displays a range of entries in the device table.

```
set device range=4-7
```

Removing an Entry from the Device Table

In this example, the `set device` command removes an entry from the device table.

```
set device rmdevice=on range=2
```

Configuring a Device

In this example, the `set device` command configures a device to use a dialer script and to override the baud rate specified on the `set line` command.

```
set device name=OutDev ports=3-5 dialer=modemscp
baud=19200
```

set filter

Introduction

Purpose

Use the `set filter` command to manage filters that control and record traffic over PPP, SLIP, and CSLIP connections. With the `set filter` command, you can

- Create filters
- Remove filters from the filters table
- Display entries in the filter table
- Display the contents of a filter

About Filters: An Overview

Use filters to trigger the following actions on PPP, SLIP, and CSLIP connections:

- Block or pass packets
- Bring up or reject connections
- Reset the idle timeout timer
- Send information to the log file

Rules for Creating Filters

Here are some rules for creating filters:

- The action a filter takes depends on the contents of the filter and on the type of filter it is defined as on the `set user` command. If the filter is referenced on the
  - — `passpacket` field, it will allow packets that meet filter criteria to pass through a serial port and block all others
  - — `bringup` field, it will bring up a connection when the port handles a packet that meets filter criteria
  - — `keepup` field, it will reset the timer defined on the `set user idletimeout` field when the port handles a packet that meets filter criteria
  - — `logpacket` field, it will send a message to the log file when the port handles a packet that meets filter criteria
- Filters are made up of 1 to 32 stanzas, each of which expresses filtering criteria.
- Filter criteria are called tokens. Examples of tokens include IP addresses, TCP or UDP port numbers, whether a packet is incoming or outgoing, and several others.
- Tokens must be separated by slashes (/).
- Stanzas are processed in order. That is, first S1 (stanza 1) is processed and then S2, and so on.
- As soon as a stanza's criteria is completely satisfied, filtering action occurs and subsequent stanzas are ignored. For example, if S1 specifies an IP address of 190.159.146.10 and an ICMP message type 7, a packet from that IP address carrying that ICMP message type will

trigger filtering action. Subsequent stanzas will not be processed. Consequently, you must specify and relationships (all criteria must be satisfied) in the same stanza and or relationships (any of the criterion must be satisfied) in different stanzas.

- The exclamation mark (!) at the beginning of a stanza changes how the filter acts. When a packet is encountered that meets stanza criteria, the filter does **not** execute the filter function (for example, bringing up a connection) and it does **not** process any more stanzas.

About the Filter Table The filter table holds a maximum of 64 entries.

Required Privileges Root privileges are required to use the `set filter` command.

Related Information See the `set user` command for information on associating a filter with a particular user.

Command Syntax

Creation Syntax Use this form of the `set filter` command to create filters and add stanzas to them or to rename filters.

```
set filter name=name [newname=name]
[s#=token\token\token...]
```

Removal Syntax Use this form of the `set filter` command to remove a filter from the filters table.

```
set filter {rmfilter=on range=range|rmfilter=name}
```

Display Filter Table Entries Use this form of the `set filter` command to display entries in the filter table.

```
set filter [range=range]
```

Display Filter Stanzas Use this form of the `set filter` command to display all the stanzas of a filter.

```
set filter name=name show=on
```

Command Fields

Field Descriptions

name
is a name for the filter

newname
is a new name for a previously defined filter

range
is an entry or range of entries in the filters table

rmfilter

`on`
means that identified filters will be removed from the filter table

`name`
means that the filter identified by this name will be removed from the filter table

show

`on`
means that stanzas from the filter identified on the `name` field will be displayed

`off`
means that stanzas from the filter identified on the `name` field will **not** be displayed

The default is `off`.

s#=token/token/token...

#
is the number of a stanza, which can be from 1 to 32.
token/*token*/*token*...
are 1-32 tokens, which are the criteria by which filtering is accomplished. Separate tokens by a forward slash (/). Tokens can consist of any of the following:

- *`servicename`*, which means filter criterion is a name in the service table that identifies a particular process, such as telnet (see `set service`)
- *`hostname`*, which means filter criterion is the name of a host defined in the host table (see `set host`)
- *`protocol-number`*, which means filter criterion is the number in an IP packet that identifies the protocol to which IP should pass the packet. Use one of the following: 1 for ICMP, 2 for IGMP, 6 for TCP, and 17 for UDP.
- *`ip-addr`*, which means filter criterion is an IP address
- *`ip-mask`*, which is an IP mask that modifies the meaning of the `ip-addr` field
- *`port-num`*, which means filter criterion is a TCP or UDP port number
- *`port-num-port-num`*, which means filter criterion is a range of TCP or UDP port numbers
- `rcv`, which means filter criterion is incoming packets
- `send`, which means filter criterion is outgoing packets
- `dst`, which means filter criteria will be found in destination IP packet fields within the IP packet, such as destination IP

addresses, ports, and host names

- `src`, which means filter criteria will be found in source IP packet fields, such as IP addresses, ports, or host names
- `syn`, which means start filtering when the start of a TCP data stream is encountered. This option is always used with the `fin` option and is used to trigger logging (`logpacket` field on the `set user` command).
- `fin`, which means stop filtering when the end of a TCP data stream is encountered. This value is always used with the `syn` option and ends logging (`logpacket` field on the `set user` command.).
- `tcp`, which means filter criterion is TCP packets
- `udp`, which means filter criterion is UDP packets
- `icmp`, which means filter criterion is ICMP packets. Note: You can also specify a type of ICMP packet. Here is how: `s1=`*`type`*`/icmp`. *`type`* is the type of ICMP packet, which can be any of the following:

Message Type	Type Identifier
Echo reply	0
Destination unreachable	3
Source quench	4
Redirect	5
Echo request	8
Time exceeded for a datagram	11
Parameter problem on a datagram	12
Timestamp request	13
Timestamp reply	14
Address mask request	17
Address mask reply	18

- ! (exclamation), which means that when a packet is encountered that meets stanza criteria, the filter does **not** execute the filter function (for example, bringing up a connection) and it does **not** process any more stanzas

Command Examples

Displaying the Filter Table

In this example, the `set filter` command displays the filter table.

```
set filter
```

Displaying Filter Stanzas

In this example, the `set filter` command displays stanzas of a filter.

```
set filter name=filter1 show=on
```

Removing a Filter from the Filter Table

In this example, the s`et filter` command removes a filter from the filter table.

```
set filter rmfilter=filter1
```

Filtering on a Source IP Address

In this example, the `set filter` command creates a filter that uses a source IP address as the filter criterion.

```
set filter name=filter1 s1=src/199.86.8.3
```

Filtering on an ICMP Packet Type

In this example the `set filter` command creates a filter that uses an ICMP type 13 packet (destination unreachable) as filter criterion.

```
set filter name=filter1 s1=13/icmp
```

set flow

Introduction

Purpose Use the `set flow` command to configure or display flow control parameters for PortServer II's EIA-232 serial ports.

Required Privileges All users can use the `set flow` command to configure or display flow control parameters for the port they are using.

Root privileges are required to use the `set flow` command to configure or display flow control parameters for other ports.

Related Information See `set line` and `set ports`.

Command Syntax

Configuration Syntax Use this form of the `set flow` command to configure flow control attributes for ports.

```
set flow [aixon={on|off}] [aixoff={on|off}]
[altpin={on|off}][cts={on|off}] [dcd={on|off}]
[dsr={on|off}] [dtr={on|off}] [itoss={on|off}]
[ixany={on|off}] [ixoff={on|off}] [ixon={on|off}]
[range=range] [ri={on|off}] [rts={on|off}]
```

Display Syntax Use this form of the `set flow` command to display flow control attributes for ports.

```
set flow [range=range]
```

Command Fields

Field Descriptions

aixoff

`on`
means that the auxiliary flow control characters defined on the `set keys` command are used for input flow control

`off`
means that the auxiliary flow control characters defined on the `set keys` command are **not** used for input flow control

The default is `off`.

aixon

`on`
means that the auxiliary flow control characters defined on the `set keys` command are used for output flow control

`off`
means that the auxiliary flow control characters defined on the `set keys` command are **not** used for output flow control

The default is `off.`

altpin

`on`
means that the altpin option is used. This option swaps DCD with DSR so eight-pin RJ-45 connectors can be used with modems. Ports using this option must be equipped with altpin cables.

`off`
means that the altpin option is **not** used

The default is `off`.

cts

`on`
means CTS (clear to send) is used for output flow control

`off`
means CTS is **not** used for output flow control

The default is `off`.

dcd

`on`
means that DCD (data carrier detect) is used for output flow control

`off`
means that DCD is **not** used for output flow control

The default is `off`.

dsr

`on`
means that DSR (data set ready) is used for output flow control

`off`
means that DSR is **not** used for output flow control

The default is `off`.

dtr

`on`
means that DTR (data terminal ready) is used for input flow control

`off`
means that DTR is **not** used for input flow control

The default is `off`.

itoss

is used only with software flow control (XON\XOFF) and only if `ixany=on`

`on`
means that the character that resumes output is discarded

`off`
means that the character that resumes output is **not** discarded

The default is `off`.

ixany

is used only with software flow control

`on`
means any received character can restart PortServer II output when output has been stopped because of software flow control. Specify `on` only when PortServer II communicates with a device, such as printers and terminals that use software flow control (XON\XOFF).

`off`
means output will resume only when the XON character is received

The default is `off`.

ixoff

`on`
means that PortServer II will use input software flow control

`off`
means that PortServer II will **not** use input software flow control

The default is `on`.

ixon

`on`
means that PortServer II will use output software flow control

`off`
means that PortServer II will **not** use output software flow control

The default is `on`.

range

is a port or range of ports to which this `set flow` command applies

ri

`on`
means that RI (ring indicator) is used for output flow control

`off`
means that RI is **not** used for output flow control

The default is `off`.

rts

`on`
means that RTS (request to send) is used for output flow control

`off`
means that RTS is **not** used for output flow control

The default is `off`.

Command Examples

Displaying Flow Control Settings

In this example the `set flow` command displays the flow control parameters for a port.

```
set flow range=3
```

Configuring Flow Control Settings

In this example, the `set flow` command configures hardware flow control.

```
set flow range=3 cts=on rts=on ixoff=off ixon=off
```

set forwarding

Introduction

Purpose

Use the `set forwarding` command to

- Configure PortServer II to
  - — Function as an IP router using Routing Information Protocol (RIP) to dynamically maintain routes
  - — Perform proxy ARP services
  - — Handle various ICMP-related functions
- Display IP routing parameters

Required Privileges

Anyone can display IP routing parameters. Root privileges are required to configure IP routing.

Related Information

See the `set route` command for information on creating static routes.

Command Syntax

Configuration Syntax

Here is the form of the `set forwarding` command used to configure PortServer II for IP routing, proxy ARP, and various ICMP-related functions.

```
set forwarding [advertise=time]
[icmpdiscovery={on|off}]
[icmpsendredirects={on|off}]
[icmpmaskserver={on|off}] [igmp={on|off}]
[poisonreverse={on|off}] [proxyarp={on|off}]
[state={off|passive|active}]
[splithorizon={on|off}] [timeout=time]
```

Display Syntax

Here is the form of the `set forwarding` command used to display IP routing parameters.

```
set forwarding
```

Command Fields

Field Descriptions

advertise

is the interval at which PortServer II advertises its routes. This field is used only if `state=active`.

The range is 10 to 180 seconds, and the default is 30 seconds.

icmpdiscovery

`on`
means PortServer II sends and answers ICMP Router Discovery packets

`off`
means PortServer II does **not** send and answer ICMP Router Discovery packets

The default is `off`.

icmpmaskserver

`on`
meansPortServer II acts as an ICMP mask server I

`off`
means PortServer II does not act as an ICMP Mask Server

The default is `off.`

icmpsendredirects

`on`
means PortServer II sends ICMP redirect messages when it detects a host is using a nonoptimal route, such as when the host uses the PortServer II to route to a destination that can be reached more efficiently using another router or when the destination host can be reached directly (that is, without the services of any router).

`off`
means PortServer II does not send ICMP redirect messages

The default is `off`.

igmp

`on`
means that PortServer II announces itself as a router when it initializes. This means that PortServer II will be included in the IGMP router's group broadcasts.

`off`
means that PortServer II does not announce itself as a router when it initializes and will not be included in IGMP router's group broadcasts

The default is `off`.

poisonreverse

`on`
means that `poisonreverse` is `on`. When this option is `on`, learned routes **are** propagated over the same interface on which they are learned, but the destination specified in those routes are advertised as unreachable. The `splithorizon` option must be `on` if `poisonreverse` is `on`.

`off`
means that the `poisonreverse` option is `off`

The default is `off`.

proxyarp

`on`
means PortServer II provides proxy ARP services. Proxy ARP is a technique in which a router answers ARP requests intended for another system. By pretending to be the other system, the router accepts responsibility for forwarding packets to that system. Use proxy ARP to route packets to and from serial routes on the same IP subnetwork as PortServer II's ethernet interface.

`off`
means PortServer II does not support proxy ARP

The default is `off`.

splithorizon

`on`
means the `splithorizon` option is `on`. When this option is `on`, learned routes are **not** propagated from the interface on which they are learned. Use this option, only if `state=active`.

`off`
means the `splithorizon` option is `off`.

The default is `on`.

state

`off`
limits PortServer II routing to static routes defined in the route table. See `set route`.

`passive`
configures PortServer II to use the routing information protocol (RIP) to learn routes but not to propagate them

`active`
configures PortServer II to use RIP to both learn and propagate routing information

The default is `off`.

timeout

is the time in which an entry in the routing table must be updated. If an entry exceeds the value specified here, it will be discarded. This value must be at least six times the `advertise` value.

The range is 60 to 1080, and the default is 180 seconds.

Command Examples

Displaying the IP Routing Table

In this example, the `set forwarding` command displays the IP routing table.

```
set forwarding
```

Configuring Proxy ARP

In this example, the `set forwarding` command configures Proxy ARP

```
set forwarding proxyarp=on
```

Configuring RIP

In this example, the `set forwarding` command configures PortServer II to

- Listen for and advertise RIP routing information every 45 seconds
- Discard this route from the routing table if a routing update is not received within 270 seconds. This value is derived from the value on the `advertise` field. The `timeout` value must be **at least** 6 times the `advertise` value. Since no `timeout` is specified, the default (6 times the `advertise` value) is used.
- Implement split horizon

```
set forwarding state=active advertise=45
splithorizon=on
```

set framerelay

Introduction

Purpose Use the `set framerelay` command to

- Configure a port for frame relay
- Display the current frame relay configuration parameters

Required Privileges Root privileges are required to configure frame relay parameters. Anyone can display them.

Related Information `set frdlci` command.

Command Syntax

Configuration Syntax Here is the form of the `set framerelay` command used to configure frame relay on PortServer II.

```
set framerelay [becn={on|off}] [enable={on|off}]
[lmi=scheme] [lmir1fc={on|off}] [mtu=size]
[nN1=cycles] [nN2=error_threshold][nN3=count]
[nt1=time] [range=range]
```

Display Syntax Here is the form of the `set framerelay` command used to display frame relay parameters.

```
set framerelay [range=range]
```

Command Fields

Field Descriptions

becn

`on`
means PortServer II responds to backward explicit congestion notification (BECN) messages from the network by throttling back transmission

`off`
means PortServer II ignores BECN messages from the network

The default is `on`.

enabled

`on`
means frame relay is enabled

`off`
means frame relay is disabled

The default is `off`.

lmi
is the protocol for exchanging line management information between PortServer II and the network. Your choice must match the protocol used by the network.

Options are `none`, `lmirev1`, `annexa`, and `annexd`.

The default is `annexd`.

lmir1fc
`on`
means use Rev1 flow control instead of BECN

`off`
means do **not** use Rev1 flow control instead of BECN

The default is `off`.

mtu
is the maximum frame size (in bytes) to use on ports configured by this command.

The range is 64 to 8192. The default is 1600 bytes.

range
is one of the following:

- A range of ports to configure for frame relay
- A range of frame relay table entries (index numbers) to display

nN1
is the number of polling cycles between full status enquiries

The range is 1 to 255 cycles, and the default is 6 cycles.

nN2 and **nN3**
`nN2` is the error threshold and `nN3` is the monitored events count for ports configure with this command. These fields work together. When `nN2` errors occur in the course of handling `nN3` frames, the line is assumed down and diagnostic action is initiated.

For `nN2`, the range is 1 to 10, and the default is 3.

For `nN3`, the range is 1 to 10, and the default is 4.

nt1
is the time in seconds between LMI status requests

The range is 5 to 30, and the default is 10.

Command Examples

Configuring a Port for Frame Relay

In this example, the `set framerelay` command configures a port for frame relay.

```
set framerelay range=3 becn=on lmi=annexd mtu=1600
```

Displaying the Frame Relay Table

In this example, the `set framerelay` command displays the entire frame relay table.

```
set framerelay
```

Displaying an Entry in the Frame Relay Table

In this example, the `set framerelay` command displays an entry in the frame relay table.

```
set framerelay range=3
```

set frdlci

Introduction

Purpose Use the `set frdlci` command to

- Configure frame relay virtual circuits
- Display virtual circuit configuration parameters
- Delete virtual circuits

Required Privileges Root privileges are required to configure and delete virtual circuits. Anyone can display virtual circuit parameters.

Related Information `set framerelay` command.

Command Syntax

Configuration Syntax Here is the form of the `set frdlci` command used to configure a frame relay virtual circuit:

```
set frdlci [bcmax=bps] [bcmin=bps] [be=bps] [cir=bps]
dlci=dlci [enable={on|off}] [fallback=seconds]
port=port [protoencap={on|off}]
```

Display Syntax Here is the form of the `set frdlci` command used to display frame relay virtual circuit parameters:

```
set frdlci [{range=range | port=port}]
```

Delete Syntax Here is the form of the `set frdlci` command used to delete a frame relay virtual circuit.

```
set frdlci delete=on port=port dlci=dlci
```

Command Fields

Field Descriptions

bcmax
is the committed burst rate, which is the maximum amount of data the network agrees to transfer over this virtual circuit under normal conditions. Set this to the value provided by you network provider or to a value that is greater than or equal to the CIR.

The default is 56,000 bps.

bcmin
is the minimum transmission rate to which PortServer II will drop back when the network becomes congested

The default is half of `bcmax`.

be

is the excess burst size, which is the maximum transfer rate (bps) over the CIR for this virtual circuit. To maximize throughput, you can set this value to the desired transfer rate minus the CIR.

For example, on a 56 kbps line with one DLCI and a CIR of 28 kbps, you might set `be` to 28 kbps, enabling transmission at the line rate (56 kbps). This will, however, result in the discard eligibility bit being set in packets sent in excess of the CIR, meaning these packets are likely candidates for discard should the network become congested.

If you set this value to exceed the CIR, monitor the virtual circuit carefully to ensure that an unacceptable number of packets are not discarded, which will result in an excessive number of retransmissions.

The default is 0, meaning that the maximum transfer rate is the CIR.

cir

is the committed information rate (bps) for this virtual circuit. Use the rate ordered from your network service provider.

The default is 56,000 bps.

delete

`on`
means that the virtual circuits identified on the `port` and `DLCI` fields will be deleted

`off`
means that a delete operation will not occur

The default is `off`.

dlci

is the data link connection identifier, which is the means by which this virtual circuit is identified. The number you use here must be one of those supplied to you by your network service provider.

enable

`on`
enables this virtual circuit, permitting traffic to flow over it

`off`
disables this virtual circuit

The default is `on`.

fallback

is the period in seconds that PortServer II will use the fallback transmission rate (`bcmin`) when network congestion occurs.

The default is 10, and the range is 0 to 244 seconds (4 minutes).

port

is the port associated with this DLCI

protoencap

`on`
means use the encapsulation scheme defined in RFC 1490

`off`
means do not use RFC 1490 encapsulation

The default is `on`.

Command Examples

Displaying Virtual Circuit Parameters

In this example, the `set frdlci` command displays virtual circuit configuration parameters.

```
set frdlci port=8
```

Deleting a Virtual Circuit

In this example, the `set frdlci` command deletes the virtual circuit identified on the `port` and `dlci` fields.

```
set frdlci delete=on port=8 dlci=17
```

Configuring a Virtual Circuit

In this example, the `set frdlci` command configures a virtual circuit identified with a DLCI of 17 that

- Uses port 7
- Is enabled
- Has a normal transmission speed of 56,000 bps

```
set frdlci port=7 dlci=17 enable=on cir=56000
```

set host

Introduction

Purpose

Use the `set host` command to

- Configure the host table, which contains host name-to-IP address mappings
- Display entries in the host table
- Delete entries from the host table

Required Privileges

Root privileges are required to issue the `set host` command.

About the Host Table and DNS

PortServer II's IP component can use the host table and a DNS server to map host names to IP addresses. These mappings allow users to identify hosts by user-friendly names, instead of IP addresses.

This is a convenience only. If you do not configure the host table or configure DNS, users will have to identify hosts by IP addresses.

If the PortServer II can access a DNS server, there is no reason to configure the host table.

You can configure

- A host table and DNS
- Either the host table or DNS
- Neither the host table nor DNS

DNS Search Order

If you configure a host table and a DNS server, PortServer II will attempt to satisfy a request by first searching the host table and then the DNS server.

Related Information

See the `set config` command for information on configuring PortServer II to use a DNS server.

Command Syntax

Configuration Syntax

Here is the form of the `set host` command used to add (configure) entries in the host table.

```
set host name=host-name ip=ip-addr
```

Display Syntax

Here is the form of the `set host` command used to display host table entries.

```
set host [range=range]
```

Delete Syntax

Here is the form of the `set host` command used to delete entries from the host table.

```
set host {rmhost=on range=range | rmhost=host-name}
```

Command Fields

Field Descriptions

ip
is the IP address that is to be mapped to the name specified on the `name` field

name
is the name that is to be mapped to the IP address specified on the `ip` field

range
is one or a range of index numbers that identify entries in the host table

rmhost
`on`
specifies that the host table entry identified by a host name on the `name` field be removed from the table

`host-name`
specifies that the host table entry identified by this host name be removed from the table

Command Examples

Displaying the Host Table

In this example, the `set host` command displays the entire host table.

```
set host
```

Displaying an Entry in the Host Table

In this example, the `set host` command displays an entry in the host table.

```
set host range=4
```

Configuring a Name-to-IP Address Mapping

In this example, the `set host` command configures a mapping between a host name and an IP address.

```
set host ip=190.150.150.10 name=server1
```

set ippool

Introduction

Purpose Use the `set ippool` command to

- Create a pool of IP addresses
- Remove a pool of IP addresses

Required Privileges Root privileges are required to create IP address pools and remove addresses from the pool.

Related Information None.

Command Syntax

Configuration Syntax Here is the form of the `set ippool` command used to configure an IP address pool.

```
set ippool count=num-ip-addr ip=1st-ip-addr
```

Remove Syntax Here is the form of the `set ippool` command used to remove the IP address pool.

```
set ippool rmpool=yes
```

Configuration Fields

Field Descriptions

count
is the number of IP addresses in the pool

ip
is the first IP address in the pool

rmpool
`yes`
means remove the IP address pool from the configuration

`no`
means do **not** remove this IP address pool from the configuration

Command Examples

Configuring a Pool In this example, the `set ippool` command configures a pool of four IP addresses. These are 190.175.175.20, 190.175.175.21, 190.175.175.22, and 190.175.175.23.

```
set ippool ip=190.175.175.20 count=4
```

Removing a Pool In this example, the `set ippool` command removes an IP pool.

```
set ippool rmpool=yes
```

set keys

Introduction

Purpose

Use the `set keys` command to

- Change the key or key sequences used to generate certain characters and command functions
- Display current key mappings for these characters and functions

About the set keys Command

Use the carat character (^) to indicate that the `Ctrl` key should be held while pressing another key.

Required Privileges

Anyone can display or change key mappings.

Related Information

None.

Command Syntax

Configuration Syntax

Here is the form of the `set keys` command used to change the key sequences you use to generate certain characters and command functions.

```
set keys function=keys
```

Display Syntax

Here is the form of the `set keys` command used to display current key mappings.

```
set keys [range=range]
```

Command Fields

Field Descriptions

function
is one of the following characters or control functions:

`eof`
is the end of file character. The default is `^d`.

`erase`
is the erase command. The default is `^h`.

`intr`
is the interrupt command. The default is `^c`.

`kill`
is the kill character. The default is `^u`.

`tesc`
is the telnet escape character. The default is `^]` (Ctrl and right bracket)

`xon`
is the XON character. The default is `^q`.

`xoff`
is the XOFF character. The default is `^s`.

`xona`
is the auxiliary XON character. The default is `^q`.

`xoffa`
is the auxiliary XOFF character. The default is `^s`.

range
is an entry or range of entries in the key table.

Command Examples

Displaying the Key Table

In this example, the `set keys` command displays the key table.

```
set keys
```

Changing a Key

In this example, the `set keys` command changes the key that generates an end of file character (eof).

```
set keys eof=^h
```

set line

Introduction

Purpose Use the `set line` command to configure and display parameters associated with a serial line.

Required Privileges All users can display and configure attributes for the lines they are using. Root privileges are required to display or configure other lines, however.

Related Information See the `set ports` and `set flow` commands.

Command Syntax

Configuration Syntax Here is the form of the `set line` command used to configure serial line parameters.

```
set line [baud=bps] [break={ignore|send|escape}]
[csize={5|6|7|8}] [error={ignore|null|parmk}]
[inpck={on|off}] [istrip={on|off}] [onlcr={on|off}]
[otab={on|off}] [parity={o|e|n}] [range=range]
[stopb={1|2}]
```

Display Syntax Here is the form of the `set line` command used to display serial line parameters.

```
set line [range=range]
```

Command Fields

Field Descriptions

baud

is the line speed (bps) for this line. Use one of the following values (the default is `9600`):

110	300	2400	19200
134	600	3600	38400
150	1200	4800	57600
200	1800	9600	115200

break

`ignore`
means that the telnet break signal is ignored

`send`
means that PortServer II sends the telnet break signal on the serial line when the PortServer II receives a break signal

`escape`
means that PortServer II sends the escape sequence on the serial line when the PortServer II receives a break signal

The default is `ignore`.

csize

is the character size, which can be 5, 6, 7, or 8 bits. The default is 8.

error

determines how PortServer II handles parity errors on the line

`ignore`
means PortServer II ignores errors

`null`
means PortServer II changes the error character to a null character

`parmk`
means PortServer II “marks” the error with FF (16450 error byte)

`dos`
means that PortServer II marks the error with an error character

The default is `ignore`.

inpck

`on`
means input parity checking is turned on

`off`
means input error checking is turned off

The default is `off`.

istrip

`on`
means the high-order bit is stripped from each byte

`off`
means the high order bit is **not** stripped from each byte

The default is `off`.

onlcr

`on`
means that `new line` characters are mapped to `carriage return/line feed` characters

`off`
means that no mapping of `new line` characters occurs

The default is `off`.

otab

`on`
means that output tabs are converted to eight spaces

`off`
means that output tabs are **not** converted

The default is `off`.

parity

`o`
means odd parity is selected

`e`
means even parity is selected

`n`
means no parity is selected

The default is `n` (no parity).

range
is the port or range of ports to which this command applies

stopb
is the number of stop bits per character to use on this line. The value you use here must match the setting on the device connected to this port. Use 1 or 2 stop bits.

The default is 1 stop bit.

Command Examples

Displaying Serial Line Parameters

In this example, the `set line` command is used to display serial line parameters.

```
set line
```

Configuring Baud, Parity and Stop Bits

In this example, the `set line` command is used to configure the line's baud rate (line speed), parity, and the number of stop bits.

```
set line range=3-4 baud=150 parity=e stopb=2 csize=6
```

set logins

Introduction

Purpose

Use the `set logins` command to

- Configure the sequence of events that occurs when a user logs into a PortServer II port. This includes information the user supplies and PortServer II prompts and responses.
- Display current login settings

Required Privileges

Regular users can

- Display current login-related settings for the port they are using
- Change login-related settings for the port they are using for their current session
- Save configuration changes for the port for future sessions if the administrator (root) has previously specified `set logins write=on`

Root privileges are required to display information about other ports and to make configuration changes.

Related Information

`set port` and `set user` commands.

Command Syntax

Configuration Syntax

Here is the form of the `set logins` command used to configure login sequences:

```
set logins [cmdprompt=string] [logprompt=string]
[login={on|off}] [passwd={on|off}]
[passprompt=string] [range=range] [verbose={on|off}]
[write={on|off}]
```

Display Syntax

Here is the form of the `set logins` command used to display login sequences:

```
set logins [range=range]
```

Command Fields

Field Descriptions

cmdprompt

is the PortServer II prompt displayed to a regular user who has logged in. The maximum length is eight characters. Enclose this string in quotation marks if it includes spaces.

The default is `digi>` for normal users and `#>` for root users.

login

`on`
means that a user must log into the port.

`off`
means that a user is not required to log into the port

The default is `on` for inbound `dev` types (see `set ports`). This field is disabled when the port is configured as an auto port (see `set ports`).

logprompt

is the login prompt PortServer II displays. The maximum length is 10 characters. Enclose this string in quotation marks if it includes spaces.

The default is `login:`.

passprompt

is the password prompt PortServer II displays. The maximum length is 10 characters. Enclose this string in quotation marks if it includes spaces.

The default is `password:`.

passwd

`on`
means that users are required to supply a password to access PortServer II on the ports specified by the `range` field.

`off`
means that users do not supply a password to access PortServer II

The default is `on`. This field is disabled when the port is configured as an auto port (see `set ports`).

range

is the range of ports addressed by this `set logins` command

verbose

`on`
means that PortServer II displays connection status messages to users before the login prompt

`off`
means that PortServer II does **not** display connection status messages to users before the login prompt

The default is `off`.

write

`on`
means that configuration changes made by regular users can be saved and used for subsequent sessions by that user

`off`

means that configuration changes made by regular users are **not** saved

Command Examples

Displaying Login Information on a Port

In this example, the `set logins` command displays login-related information on the port the user is using:

```
set logins
```

Displaying Login Information on a Range of Ports

In this example, the `set logins` command displays login-related information on a range of ports:

```
set logins range=3-5
```

Configuring a Port for User Configuration

In this example, the `set logins` command configures a port so that users can save their login-related configuration changes and use them in future sessions:

```
set logins write=on
```

Configuring the Command Prompt

In this example, the `set logins` command configures the command prompt. Since there are spaces in the new command prompt, the entry is enclosed in quotation marks.

```
set logins cmdprompt=”Ent Cmd:”
```

set menu

Introduction

Purpose

Use the `set menu` command to

- Create menus for PortServer II users
- Display menu table entries
- Display lines of a menu
- Remove a menu from a port
- Remove a line from a menu

Required Privileges

Root privileges are required to configure menus and to perform any removal operations. Anyone can perform display operation.

Related Information

See the `menu` and `defaultaccess` fields on the `set user` command for information on setting up a user to use a menu.

Command Syntax

Creation Syntax

Use this form of the `set menu` command to create a menu.

`set menu [c#=`*`command`*`] [m#=`*`string`*`] [range=`*`range`*`] [t#=`*`string`*`]`

Display Menu Table Entries Syntax

Use this form of the `set menu` command to display the contents of the menu table:

`set menu [range=`*`range`*`]`

Display Lines of Menus

Use this form of the `set menu` command to display the contents of a menu:

`set menu range=`*`range`* `[show={on|off}]`

Remove Menu Syntax

Use this form of the `set menu` command to remove a menu from the menu table:

`set menu range=`*`range`* `rmmenu=on`

Remove Line Syntax

Use this form of the `set menu` command to remove a line from a menu:

`set menu range=`*`range`* `rmentry=`*`line-num`*

Command Fields

Field Descriptions

c#=*command*

`c`
means that this is a command that is executed when a user selects this menu line

`#`
is a line number. Lines appear in numeric order on the menu.

`command`
is any PortServer II command, but `telnet` and `rlogin` are the most common commands to use here

range
is a port or range of ports

rmentry
removes the specified line from the menu

rmmenu

`on`
means the menu will be removed from the ports specified on the range field

`off`
means the remove function is not active

The default is `off`.

m#=*string*

`m`
means that this is a text or informational line

`#`
is a line number for the menu. Lines appear in numeric order on the menu

`string`
is a text string. Enclose strings with spaces in quotation marks.

show=on
displays menu entries identified on the `range` field

t#=*string*

`t`
means that this is a title line

`#`
is a line number for the menu. Each menu can have two title lines (t1 and t2).

`string`
is a text string. Enclose strings with spaces in quotation marks.

Command Example

Creating a Menu

In this example, the `set menu` command creates a menu with active fields that enable users to start telnet sessions to hosts named server1 and server2.

```
set menu range=4 t1="Welcome to the Communications
Server" t2="Make a Selection" m1="Telnet to Server1"
c1="telnet server1" m2="Telnet to Server2" c2="telnet
to server2"
```

Displaying the Menu Table

In this example, the `set menu` command displays the contents of the menu table.

```
set menu
```

Removing a Menu from a Port

In this example, the `set menu` command removes a menu from a port.

```
set menu range=4 rmmenu=on
```

set modem

Introduction

Purpose

Use the `set modem` command to

- Assign modem test and initialization scripts to ports
- Display the modem table
- Clear the association between ports and modem test and initialization scripts

Required Privileges

Normal users can use the `set modem` command to display the scripts associated with the port they are using. Administrator (root) privileges are required to use the command to display information on other ports and to configure an association between a port and test and initialization scripts.

Related Information

See the `set scripts` command for information on creating scripts.

Command Syntax

Configuration Syntax

Use this form of the `set modem` command to configure an association between a port and modem test and initialization scripts.

`set modem [init=`*`script`*`][range=`*`range`*`] [test=`*`script`*`]`

Display Syntax

Use this form of the `set modem` command to display modem table entries.

`set modem [range=`*`range`*`]`

Clear Syntax

Use this form of the `set modem` command to clear an association between a port and modem test and initialization scripts.

```
set modem [init=no] [test=no]
```

Command Fields

Field Descriptions

init

is one of the following:

- The name of an initialization script (created with the `set scripts` command)
- The index number of an initialization script in the scripts table
- `no`, which clears an association between a port and an initialization script

range

is the range of ports to which this command applies

test
is one of the following:

- The name of a test script (created with the `set scripts` command)
- The index number of a test script in the scripts table
- `no`, which clears an association between a port and a test script

Command Examples

Displaying the Current Port's Scripts

In this example, the `set modem` command displays the names of scripts associated with the user's port.

```
set modem
```

Displaying a Range of Ports' Scripts

In this example, the `set modem` command displays the names of scripts associated with a range of ports.

```
set modem range=1-16
```

Configuring a Port for Scripts

In this example the `set modem` command configures an association between a port and test and initialization scripts.

```
set modem test=test1 range=2 init=init1
```

Clearing a Port of Scripts

In this example, the `set modem` command clears an association between a port and test and initialization scripts.

```
set modem range=2 test=no init=no
```

set ports

Introduction

Purpose Use the `set ports` command to

- Configure the operating parameters of a port
- Display the port's operating parameters

Required Privileges Normal users can use the `set ports` command to display operating parameters for the port they are using. Administrator (root) privileges are required to use it to display parameters on other ports and to configure ports.

Related Information See `set line` and `set flow`.

Command Syntax

Configuration Syntax Here is the form of the `set ports` command to configure the operating parameters of a port.

```
set ports [auto={on|off}] [bin={on|off}]
[dest=ip-adr] [dev=device] [dport=tcp-port]
[edelay=milliseconds] [group=group] [range=range]
[sess=sessions] [termtype=type] [uid=id]
```

Display Syntax Here is the form of the `set ports` command to display operating parameters for a port.

```
set ports [range=range]
```

Command Fields

Field Descriptions

auto

`on`
means that all users of the port will bypass PortServer II's login and password sequence and be automatically connected to the destination defined on the `dest` field

`off`
means that port users will **not** be automatically connected to a destination.

The default is `off`.

bin

`on`
means that telnet users are provided with telnet binary connections

`off`
means that telnet users are provided with normal (ASCII) connections

The default is `off`.

dest
is the IP address of the destination system to which port users will be routed if `auto=on`

dev
is the device type, which defines the device connected to the port. Typically, you can use the following to define the devices listed:

- Most printers can use `dev=prn`.
- Most dumb terminals can use `dev=term`.
- Most incoming modem connections can use `dev=min`.
- Most outgoing modem connections can use `dev=mout`.
- Most bidirectional modem connections can use `dev=mio`.
- Most Realport connections can use `dev=rp`.
- Most reverse telnet connections can use `dev=prn`.

If the device you are configuring is not one of these listed or requires unusual flow control attributes, use the information in the following table to define a device type:

Device Type	Attributes
term	• PortServer II generates a login when it receives data. • PortServer II ignores loss of carrier (DCD low). • DTR and RTS are high when the connection is idle. • This type usually requires cable support for transmit, receive, and ground only, which means a 3-wire crossover cable will work. Six, eight, and ten wire crossover cables work as well. • Do **not** use `dev=term` for RealPort and reverse telnet connections.
prn	• PortServer II never generates a login. • PortServer II ignores carrier. • DTR and RTS are low when the connection is idle. • This type usually requires cable support for transmit, receive, and ground only, which means a 3-wire crossover cable will work. Six, eight, and ten wire crossover cables work as well. • Use `dev=prn` for reverse telnet connections.

min	• PortServer II generates a login when carrier is detected (DCD high). • PortServer II closes the port at carrier loss (DCD low). • DTR and RTS are high when the connection is idle. • This type requires a 10-pin straight-through cable or an altpin cable. • Do not use `dev=min` for RealPort and reverse telnet connections.
mout	• PortServer II never generates a login. • PortServer II closes the port at carrier loss (DCD low). • DTR and RTS are low when the connection is idle. • This type requires a 10-pin straight-through cable or an altpin cable. • `dev=mout` supports RealPort and reverse telnet.
mio	• PortServer II generates a login when carrier is detected (DCD high). • PortServer II closes the port at carrier loss (DCD low). • DTR and RTS are high when the connection is idle. • This type requires a 10-pin straight-through cable or an altpin cable. • `dev=mio` supports reverse telnet but does **not** support RealPort.
host	• PortServer II does not generate a login. • PortServer II opens the port at DCD high and closes the port at carrier loss (DCD low). • DTR and RTS are low when the connection is idle. • This type supports reverse telnet and RealPort. • This type requires a cable that supports carrier detect (DCD).

hdial	• PortServer II generates a login when carrier is detected (DCD high) and data is received. • PortServer II closes the port at carrier loss (DCD low). • DTR and RTS are low when the connection is idle. • This type does **not** support reverse telnet or RealPort. • This type requires 10-pin cables with DCD and DTR cross-connected or an altpin cable.
hio	• PortServer II generates a login when carrier is detected (DCD high) and data is received. • PortServer II closes the port at carrier loss (DCD low). • DTR and RTS are low when the connection is idle. • This type requires 10-pin cables with DCD and DTR cross-connected or an altpin cable.
rp	• PortServer II never generates a login. • PortServer II ignores carrier. • DTR and RTS are low when the connection is idle. • This type usually requires cable support for transmit, receive, and ground only, which means a 3-wire crossover cable will work. Six, eight, and ten wire crossover cables work as well. • Use `dev=rp` for RealPort connections.

The default is `term`.

Note: With mio, mout, min, host, and hdial device types, PortServer II lowers DTR at disconnect and holds it low for two seconds to ensure a clean disconnection.

dport

is the TCP port for users of autoconnect ports, which is one of the following:

- 23 for telnet
- 513 for rlogin
- Any other TCP port or a physical port on the PortServer II, identified by specifying `20` and then the port number. For example, to indicate an autoconnect telnet connection to port 12, specify `dport=2012`.
- 0, which means one of two things, depending on whether a specific user is assigned to this port on the `uid` field: (1) That rlogin

is used as the default if a specific user is assigned to this port (2) That telnet is used as the default if a specific user is **not** assigned to this port

The default is 0.

group
assigns a group number to this port, which means that this port is part of a hunt group. Outgoing calls specifying this hunt group can then use any available port in the group. Use numbers 65 to 99 to avoid conflicts with regular port numbers.

range
is the port or range of ports to which this command applies

sess
is the maximum number of sessions any user can run through this port

The range is 1-9, and the default is 4.

termtype
is the type of terminal assigned to the port. This information is used during multiscreen and multisession operations and is passed to the host during telnet negotiations. Use a terminal type that is valid with the host operating system.

uid
is an index number in the user table that identifies a particular user for this port. If you use this field, calls from others attempting to use this port will be rejected.

Command Examples

Displaying Attributes of the Current Port

In this example, the `set ports` command displays attributes for the port to which the user is connected.

```
set ports
```

Displaying Attributes for a Range of Ports

In this example, the `set ports` command displays attributes for a range of ports.

```
set ports range=7-8
```

Configuring an Autoconnect Port

In this example, the `set ports` command configures the port so that all incoming users are automatically connected via telnet to the host specified on the `dest` field. The port is also available for outgoing connections.

```
set ports range=5 auto=on dest=199.125.123.10 dev=mio
dport=23
```

set radius

Introduction

Purpose

Use the `set radius` command to

- Configure PortServer II to use one or more RADIUS (remote authentication dial-in user service) servers to authenticate and maintain user profiles on dial-in users
- Display current RADIUS configuration parameters

About RADIUS

When PortServer II uses a RADIUS server, it authenticates users by first searching its own user table and then, if the user is not found, searching the RADIUS server.

Required Privileges

Normal users can use the `set radius` command to display all RADIUS configuration parameters, except the RADIUS password. Administrator (root) privileges are required to display the password and configure PortServer II to use RADIUS servers.

Related Information

None.

Command Syntax

Configuration Syntax

Here is the form of the `set radius` command used to configure PortServer II to use RADIUS servers to authenticate dial-in users.

```
set radius [primary=ip-adr] [run={on|off}]
[secondary=ip-adr] [secret=password]
```

Display Syntax

Here is the form of the `set radius` command used to display RADIUS configuration status.

```
set radius
```

Command Fields

Field Descriptions

primary

is the IP address of the primary RADIUS server. This is the server that PortServer II queries first. If this server is down or busy, PortServer II queries the secondary server (if there is one).

run

`on`
enables RADIUS authentication

`off`
disables RADIUS authentication

The default is `off`.

secondary
is the IP address of a secondary RADIUS server

secret
is a password used for encryption of messages between the RADIUS server and PortServer II. The server and PortServer II must use the same password. The primary and the secondary servers are not required to use the same password. If they are different, however, you must issue two `set radius` commands, one to configure the primary RADIUS server and one to configure the secondary server. See the command examples for more information.

Command Examples

Displaying RADIUS Configuration Status

In this example, the `set radius` command displays the status of the current RADIUS configuration.

```
set radius
```

Configuring a Primary RADIUS Server

In this example the `set radius` command configures PortServer II to use a primary RADIUS server.

```
set radius run=on primary=199.150.150.10
secret=xyyzzz
```

Configuring Two RADIUS Servers

In this example, the first `set radius` command configures the primary RADIUS server. The second `set radius` command configures the secondary server. Two commands are required because the two servers use different passwords (`secret` field).

```
set radius run=on primary=199.150.150.10
secret=xyyzzz
```

```
set radius run=on secondary=199.150.150.22
secret=abbccc
```

set route

Introduction

Purpose

Use the `set route` command to

- Manually configure IP routes
- Display the contents of the route table

About the Route Table

The route table holds up to 50 entries.

Required Privileges

Normal users can display the contents of the route table. Root privileges are required to configure IP routes.

Related Information

See the `set forwarding` command for information on configuring PortServer II to use dynamic IP routes maintained by RIP.

Command Syntax

Configuration Syntax

Here is the form of the `set route` command used to manually configure IP routes:

```
set route {gateway=ip-adr | wanname=name} mask=mask
metric=hops net=net-adr range=range
[rmroute={on|off}]
```

Display Syntax

Here is the form of the `set route` command used to display the route table:

```
set route
```

Command Fields

Field Descriptions

gateway
is the IP address of the router that is the next hop to the destination network defined on the `net` field. Use this field if this router is on the LAN.

wanname
is the name, defined on a `set user` command, of a WAN connection that PortServer II can use to reach the next hop to the destination defined on the `net` field.

mask
is the subnet mask used by the destination network

metric
is the number of routers through which a datagram must pass before reaching the destination network defined on the `net` field

net
is the IP network address of the destination network

range
is the entry or range of entries in the route table that will be removed when the `rmroute` field is executed

rmroute=on
means that the route table entry or entries defined on the `range` field will be removed

The default is `off`.

Command Examples

Displaying the Route Table

In this example, the `set route` command displays the entire route table.

```
set route
```

Displaying a Range of Route Table Entries

In this example, the `set route` command displays a range of entries in the route table.

```
set route range=3-5
```

Removing an Entry in the Route Table

In this example, the `set route` command removes an entry from the route table.

```
set route rmroute=on range=2
```

Configuring a Route over a WAN Connection

In this example, the `set route` command configures a route that uses a WAN connection through a serial port.

```
set route net=199.150.144.8 mask=255.255.255.0
metric=3 wanname=user999
```

set script

Introduction

Purpose Use the `set script` command to

- Define a modem or login script
- Display entries in the script table
- Display all stanzas of a script
- Delete a script from the script table

Required Privileges Anyone can display entries in the scripts table. Root privileges are required to configure scripts and display script stanzas.

Related Information See the `set user, set device, set chat, and set modem` commands.

Command Syntax

Configuration Syntax Here is the form of the `set script` command used to configure or edit a modem or login script:

```
set script name=name range=range
s{1-24}=stanza-content
```

Display Entries Syntax Here is the form of the `set script` command used to display entries in the script table:

```
set script [range=range]
```

Display Stanzas Syntax Here is the form of the `set script` command used to display all the stanzas of a script:

```
set script name=name show=on
```

Delete a Script Here is the form of the `set script` command used to delete a script from a script table.

```
set script {rmscript=on name=name | rmscript=name}
```

Command Fields

Field Descriptions

name
is the name of the script

range
is one of the following:

- A range of ports to which this script applies (for configuration)
- An index number in the script table (for display)

rmscript
removes the script specified

s {1-24}=*stanza-content*
is the number of a script stanza (1 through 24) and the contents of the stanza. The contents can include any of the following commands:

Command	Description
A*np*	Sets • Character size to *n*, which can be either 7 or 8. • Parity to *p*, which can be one of the following values: 0=no parity, 1=odd 2=even 3=mark **Example:** s1=A70
B*n*	Transmits a break signal *n* milliseconds long. If *n* is not specified, the length is 250 milliseconds. **Example:** s7=B100
C*n*	Sets carrier loss detection. If *n*= • 0, carrier loss is not detected • 1, the modem hangs up if the port loses DCD **Example:** S2=C1
D+*m*	Raises a modem signal. If *m* is • 1, DTR is raised • 2, RTS is raised
D-*m*	Lowers a modem signal. If *m* is • 1, DTR is dropped • 2, RTS is dropped

E{*string*}

Writes the string either to

- A user terminal (if running interactively)
- To a trace buffer (if running in the background)

This string can include any of the escape commands listed in *Script Escape Commands*, which follows this discussion.

Example: S10=”E{Please Log In}”

F*n*

Pauses for *n* seconds and flushes input data. The default is 0.

Example: s1=F10

G*s*

Immediately does one of the following, depending on the value of *s.* If *s* is

- The number of a stanza, control is passed to that stanza
- + (plus), the script is exited with a success message from E string
- - (minus) the script is exited with a failure message from E string

Example: s2=G7

H*s*

Sets the carrier lost (hang-up) recovery to stanza *s,* which is the number identifying another stanza or one of the following:

- + (plus), which means Exit, indicating success
- - (minus), which means Exit, indicating a general failure
- * (star), which means indicate that the remote system is busy
- = (equal), which means indicate that the remote system is down

Example: s2=H+

M{*string*}

Writes *string* to a modem

Example: s2=M{at&f\c}

This string can include any of the escape commands listed in *Script Escape Commands*, which follows this discussion.

N*b*	Changes the baud rate. The range is 50 to 115,200. Rates under 110 bps should be used only on expansion ports. **Example:** s4=N19200
P*n*	Pauses for *n* seconds. If you do not specify a value for *n*, the default is 1 second. **Example:** s5=P2
Q*n*	Sets software flow control. If *n* is • 0, flow control is disabled • 1, flow control is enabled **Example:** s5=Q0
S*n*	Defines the time to wait (timeout), in seconds, for a modem signal or input data **Example:** s2=S5
T*s*	Defines the timeout recovery state. If the time-out is exceeded, control is passed to this stanza. **Example:** s2=T8
U*n*	Immediately executes the text of stanza *n*, as if it were inserted to replace this command. You can nest this command, up to a maximum of 10. **Example:** s2=U4
W+*m*	Waits for a modem signal to go high. If *m* is • 1, wait for DCD to go high • 2, wait for CTS to go high **Example:** s6=W+1
W-*m*	Waits for a modem signal to go low. If *m* is • 1, wait for DCD to go low • 2, wait for CTS to go low **Example:** s6=W-1

[*string*]*s*	Defines the *string* and the stanza to jump to when the *string* is received on a communications line. This string can include any of the escape commands listed in *Script Escape Commands*, which follows this discussion. **Example:** s7=[abort]s22

Script Escape Commands

Introduction

This section describes the escape command you can use in `E`, `M`, and `[]` command strings.

Escape Command Description

Escape Command	Description
^c	This is the character transmitted by an ASCII keyboard when the `CTRL` key is held down and the `c` key is pressed.
\b	Backspace
\f	Form feed
\t	Tab
\n	New line
\r	Return
\\	Backslash
nnn	Octal byte value *nnn*
\x*hh*	Hexadecimal byte value *hh*

%n	Is a variable, where n is • A telephone number whose value comes from the n*n* field on the set user command • one of the following special characters: `*` (star), which generates a tone equivalent to dialing * on a touch-tone phone # (pound), which generates a tone equivalent to dialing # on a touch-tone phone =, which causes a pause of 2 seconds `w,` which causes a wait for a secondary dial tone `-` (minus), which is completely ignored and not passed to the modem.
%p	Is a variable, where *p* is an integer from 1 to 9. For login scripts, the value of *p* comes from the p*n* field on the set user command. For dialer scripts, parameters come from the p*n* field of the set device command.

Command Examples

Displaying the Script Table

In this example, the `set script` command displays the entire script table.

```
set script
```

Displaying Entries in the Script Table

In this example, the `set script` command displays an entry in the scrip table.

```
set script range=4
```

Displaying all Stanzas in a Script

In this example, the `set script` command displays all stanzas of the specified script:

```
set script name=testmodem show=on
```

Configuring a Login Script

In this example, the `set script` command defines a login script.

```
set script name=loginscript
s1=”P2[Login:]2 S10 T4”
s2=”P1 M{%1\r} S1 [sword:]3 T4”
s3=”M{%2\r} G5”
s4=”E{login failed} G-”
s5=”E{login complete} G+”
```

set service

Introduction

Purpose

Use the `set service` command to

- Configure (associate) names with TCP and UDP service ports for use in filters
- Remove entries from the service table
- Display entries in the service table

Service Numbers

The following are the service numbers (TCP and UDP ports) to which you can assign names:

Service	Port Number
FTP	21
NNTP	119
RIP	520
Login	513
Shell	514
SMTP	25
Telnet	23, 2001 to 2099
TFTP	69

Required Privileges

Normal users can display service table entries. Root privileges are required to configure entries.

Related Information

See the `set filter` command for information on configuring filters.

Command Syntax

Configuration Syntax

Use this form of the `set service` command to associate names with TCP service ports:

`set service name=`*`name`* `port={udp:`*`port`*`|TCP:`*`port`*`}`

Removal Syntax

Use this form of the `set service` command to remove an entry from the service table:

`set service {rmservice=`*`name`*`|rmservice=on range=`*`range`*`}`

Display Syntax

Use this form of the `set service` command to display entries in the service table:

`set service [range=`*`range`*`]`

Command Fields

Field Description

name
is the name to assign the service

port
is the TCP or UDP port number for the service

range
is a range of entries in the service table, which is used to identify entries to display or delete

{rmservice=*name*| rmservice=on}

name
is the name of a service to be removed from the service table

on
means remove the service (or services) from the service table identified on the range field

Command Examples

Displaying the Service Table

In this example, the set service command displays the entire service table.

```
set service
```

Displaying an Entry in the Service Table

In this example, the set service command displays a range of entries in the service table.

```
set service range=2-4
```

Configuring an Entry in the Service Table

In this example, the set service command configures a name for telnet.

```
set service name=telnet port=tcp:23
```

Removing an Entry from the Service Table

In this example, the set service command removes the telnet entry from the service table.

```
set service name=telnet rmservice=on
```

set terms

Introduction

Purpose

Use the `set terms` command to

- Define terminal types and the escape sequence a terminal uses when initiating and maintaining multiple sessions
- Display entries in the term table
- Remove entries from the term table

About the set terms Command

Here is some information on the `set terms` command:

- The `set terms` command configures PortServer II to handle terminals that are **not**
  - — Connected to PortServer II over the ethernet
  - — Using PPP, SLIP, CSLIP, or frame relay connections
- If users are to use the `Ctrl` key in a key sequence, use a carat character (^) in place of the `Ctrl` key when you configure the sequence.

Required Privileges

Normal users can display entries in the term table. Administrator (root) privileges are required to configure terminals and remove entries from the term table.

Command Syntax

Configuration Syntax

Here is the form of the `set terms` command used to configure terminals:

```
set terms [clrseq=escape-seq] [npages=pages]
[swtseq=SessNumSequence] termtype=type
```

Display Syntax

Here is the form of the `set terms` command used to display entries in the term table:

```
set terms [range=range]
```

Removal Syntax

Here is the form of the `set terms` command used to remove entries from the term table:

```
set terms {rmterm=termtype | rmterm=on range=range}
```

Command Fields

Field Descriptions

clrseq
is the escape sequence that clears the terminal's current screen. This should be the sequence specified by your terminal's manufacturer.

npages
is the number of sessions available to this terminal type. This should be the same as the number of pages of screen memory available on the terminal.

The range is 1-9.

swtseq=*SessNumSequence*
is a number that identifies the session and the escape sequence used to access that session. This should be the sequence specified by your terminal's manufacturer.

Note: There are no spaces between the number identifying the session and the key sequence used to access that session.

range
is the range of term table entries to display or remove

rmterm
termtype
is the name of the terminal supplied on the `termtype` field. This terminal type will be removed from the term table when the command with this option is executed.

`on`
means that the term table entries identified on the `range` field will be deleted when the command is executed

termtype
is a name for the terminal type. This name must match the name

- Specified on the `termtype` field of the `set ports` command
- Used by hosts on your network for this type of terminal

PortServer II provides two default terminal types, `wy60` and `wy60-e`. Use the `set terms` command to display parameters associated with these types of terminals.

Command Examples

Displaying the Entire Term Table

In this example, the `set terms` command displays the entire term table.

```
set terms
```

Displaying a Range of Entries in the Term Table

In this example, the `set terms` command displays a range of entries in the term table:

```
set terms range=4-6
```

Removing an Entry from the Term Table

In this example, the `set terms` command removes an entry from the term table.

```
set terms rmterm=on range=3
```

Configuring a Terminal Type

In this example, the `set terms` command configures a terminal type.

```
set terms termtype=VT100 npages=4 clrseq=^! swtseq=1^]
swtseq=2^[ swtseq=3^} swtseq=4^{
```

set time

Introduction

Purpose Use the `set time` command to set and display the time and date PortServer II keeps.

Required Privileges Regular users can display the time and date. Root privileges are required to set them.

Related Information None.

Command Syntax

Syntax Here is how to use the `set time` command to set or display the time and date.

```
set time [date=mn.day.yr] [time=hr.mn.sec]
```

Command Fields

Field Descriptions

date
is the month (expressed numerically), day, and year (use only two digits for the year), separated by periods

time
is the hour (24-hour clock), minute, and second, separated by periods

Command Examples

Displaying the Time In this example, the `set time` command displays the current time and date:

```
set time
```

Setting the Time In this example, the `set time` command sets the time and date.

```
set time time=17.05 date=12.25.97
```

set trace

Introduction

Purpose Use the `set trace` command to
- Configure PortServer II for tracing
- Display tracing information

Required Privileges Root privileges are required to execute the `set trace` command.

Related Information None.

Command Syntax

Configuration Syntax Use this form of the `set trace` command to configure tracing:

```
set trace [loghost=ip-addr]
[mask=type:severity] [mode={historical | concurrent]}
[state={on|off|dump}] [syslog={on|off}]
```

Display Syntax Use this form of the `set trace` command to display the status of tracing information:

```
set trace
```

Command Fields

Field Descriptions

loghost
is the IP address of a host to which trace messages should be sent. This host must be running the syslog daemon.

mask=*type*:*severity*
is the type and nature of event that should be traced

type
is one of the following:

Type	Traces events associated with...
arp	Address Resolution Protocol
cache	Routing cache
dialer	Dial-out ports
dns	Domain Name System
ether	Ethernet
framerelay	Frame relay
fwdr	Routing (forwarded IP packets)
icmp	Internet Control Message Protocol

inetd	Internet daemon (based on received packets)
ip	Internet Protocol
netd	Net daemon
ppp	Point-to-Point Protocol
radius	RADIUS
realp	RealPort
rlogin	rlogin
routed	Route daemon
serial	Serial ports
snmp	Simple Network Management Protocol
tcp	Transmission Control Protocol
telnet	Telnet
udp	User Datagram Protocol
user	Users
wan	Wide-area network connections
*	All entities listed in this table

severity
is one of the following severity levels:

Severity	Meaning
critical (the default)	This means that tracing is done on only the most severe events. This level produces the least amount of trace data.
warning	This means tracing is done on critical events and on less severe events as well. This level produces more trace data than `critical`, but less than `info`.
info	This means tracing is done on many events. It produces more trace data than previous levels.
debug	Is the level to use for debugging. Do not use this level for anything but debugging.
ignore	Turns off all trace messages.

mode

`historical`
means that all trace messages stored in the buffer may be displayed by issuing the following command: `set trace state=dump`

`concurrent`
means that all trace messages are printed to the administrative terminal when `state=on`

state

`on`
means that all messages in the trace buffer are displayed. Once they are displayed, the state remains `on`.

`off`
means that tracing off

`dump`
means that all messages in the trace buffer are displayed. Once they are displayed, the state returns to `off`.

The default is `off`.

syslog

`on`
means that trace messages are sent to the host identified on the `log-host` field

`off`
means that trace messages are not sent to a host

The default is `off`.

Command Examples

Displaying Trace Settings

In this example, the `set trace` command displays current trace settings.

```
set trace
```

Configuring Trace Levels

In this example, the `set trace` command configures tracing for ARP events.

```
set trace mask=arp:warning mode=historical
state=dump
```

set user

Introduction

Purpose

Use the `set user` command to

- Display configuration attributes stored in the user table, such as whether a user must supply a password
- Configure a range of parameters associated with users, such as whether the user automatically connects to a host or is required to supply a password
- Remove a user from the user table

About the User Table

The user table holds up to 64 entries. If you need to configure additional users, use a RADIUS server. See the `set radius` command.

Required Privileges

All `set user` command functions require root privileges.

Related Information

For more information on...	See...
Filters	The `set filter` command
Scripts	The `set script` command
Using a RADIUS server	The `set radius` command

Command Syntax

Configuration Syntax Here is the form of the `set user` command used to configure user attributes:

```
set user [accesstime=time] [addrcompress={on|off}]
[asyncmap=mask] [autoconnect={on|off}]
[autohost=ip-addr] [autoport=tcp-port]
[autoservice={telnet|rlogin|raw}] [bringup=filter]
[chapid=id] [chapkey=key] [commandline={on|off}]
[compression={vj|none}] [defaultaccess=service]
[device=device-name] [dialout={on|off}]
[downdly=seconds] [frdlci=dlci] [frport=port]
[idletimeout=time] [ipaddr=ip-addr] [ipmask=mask]
[keepup=filter] [localbusydly=seconds]
[localipadr=ip-addr] [loginscript=script]
[logpacket=filter] [maxports=number]
[menu={off|index-num}] [mtu=bytes]
[n1, n2=phone-number] [name=name]
[netrouting={off|send|rec|both}]
[netservice={on|off}] [network] [newname=string]
[outgoing={on|off}] [p1,p2...=script-parm] [papid=id]
[pappasswd=password] [passive={on|off}]
[passpacket=filter] [password={on|off}] [ports=ports]
[pppauth={none|pap|chap|both}]
[protocol={frame|ppp|slip}] [protocompress={on|off}]
[range=range] [rmtbusydly=seconds]
[sessiontimeout=seconds] [vjslots=number]
```

Display Syntax Here is the form of the `set user` command used to display entries from the user table.

```
{set user {[name=name]|[range=range]} |
set user name=name network}
```

Remove Entry Syntax Here is the form of the `set user` command used to remove an entry from the user table.

```
set user [range=range] [rmuser={on|name}]
```

Command Fields

Field Descriptions

accesstime

is the period in which the user can access PortServer II. Use the `accesstime` field to restrict the user's access to the time specified.

Use the following keywords to specify day (or days) and hours:

Period	Keyword
Working week (Monday-Friday)	wk
Sunday	su
Monday	mo
Tuesday	tu
Wednesday	we
Thursday	th
Friday	fr
Saturday	sa

Specify hour ranges in the form: `hr:min-hr:min` or `hr-hr`. Use spaces to separate keywords and then enclose the entire string in quotation marks. Here are some examples:

Examples	Provides access...
accesstime=wk9:00-17:00	Monday through Friday from 9:00 a.m. until 5:00 p.m.
accesstime="wk9:00-17:00 su0-23"	Monday through Friday from 9:00 a.m. until 5:00 p.m. and all day Sunday
accesstime="su mo fr"	All day Sunday, Monday, and Friday

addrcompress

`on`
means PortServer II attempts to negotiate address compression on PPP connections

`off`
means PortServer II will **not** attempt to negotiate address compression

The default is `on`.

asyncmap

is a mask for PPP connections that defines which of the 32 asynchronous control characters to transpose. These characters, in the range 0x00 to 0x1f are used by some devices to implement software flow

control. These devices may misinterpret PPP transmission of control characters and close the link. This mask tells PPP which characters to transpose.

The default is FFFF, which means transpose all 32 control characters. Any combination is valid. The following are the most likely masks that you will want to use:

- FFFFFFFF, which means transpose all control characters
- 00000000, which means transpose none
- 000A0000, which means transpose Ctrl-Q and Ctrl-S

autoconnect

`on`

means that a telnet or rlogin user will be automatically connected to another system without accessing the PortServer II command line once the user has satisfied login and password requirements. If you specify `yes`, specify the `autohost` and `autoport` or `autoservice` fields.

`off`

means the user will **not** be automatically connected to another system

The default is `off`.

autohost

is the IP address of a host to which this telnet or rlogin user should be automatically connected. Use this field only if you specify `autoconnect=yes`.

autoport

is the TCP port to use for the automatic connection. Use this field only if you specify `autoconnect=yes`.

If you specify `autoconnect` and do not specify a TCP port, the port will be determined by the `autoservice` field, or—if there is no `autoservice` field specified—the default, port 513, which is rlogin.

autoservice

is an alternate way to specify a TCP port for an autoconnect user (see the `autoport` field). Use this field only if you specify `autoconnect=yes`. Specify one of the following services: `telnet`, `rlogin`, or `raw (`which means that data will be passed between the serial port and the TCP stream without modification).

The default is the value of the `autoport` field.

bringup

is the name of a filter (defined on the `set filter` command) that PortServer II uses to initiate a remote connection to a PPP, SLIP, or CSLIP user. This filter must be created before you use this field.

chapid
is a character string that identifies the PPP user using CHAP authentication. This is equivalent to a user (or login) name. The string must be 16 or fewer characters and must be recognized by the peer.

chapkey
is a character string that authenticates the PPP user using CHAP authentication. This is equivalent to a password. The string must be 16 or fewer characters and must be recognized by the peer.

commandline

`on`
means that a telnet, rlogin, PPP, SLIP, or CSLIP user can access the PortServer II command line to issue commands

`off`
means that the user can **not** access the command line and can **not** issue commands

The default is `on`.

compression

`vj`
means that Van Jacobsen Header compression is used on PPP and SLIP connections

`none`
means that header compression is not used on SLIP and PPP connections

The default is `none`.

Note: The difference between a SLIP and a CSLIP connection is that CSLIP connections use Van Jacobsen Header compression. Consequently, when you specify `protocol=slip` and `compression=vj`, the connection becomes a CSLIP connection.

defaultaccess
restricts the service accessible to the user

`commandline`
means that the PortServer II command line is displayed to the user

`menu`
means that a menu is displayed to the user. If you specify this option, you must also specify a menu number on the `menu` field

`autoconnect`
means that PortServer II automatically connects the user to the destination specified on the `autohost` field

`netservice`
starts outgoing PPP, SLIP or CSLIP services, depending on which protocol is specified on the protocol field

`outgoing`
means that this user is limited to outgoing connections only

The default is `commandline`.

device
is the name of a device or a device pool (defined with the `set device` command) used for outgoing PPP, SLIP, or CSLIP connections

dialout
`on`
means that outgoing PPP, SLIP, CSLIP, or frame relay connections are enabled. A dialer script requires this field to be `on` to initiate outbound connections.

`off`
means that outgoing connections are **not** enabled

The default is `off`.

downdly
is the number of seconds the dialer script should delay before attempting to establish a PPP, SLIP, or CSLIP connection with a previously inaccessible host

The default is 0, which means do not delay in making the attempt to reconnect. The range is unlimited.

frdlci
is a DLCI (data link connection identifier) the virtual circuit that this frame relay user will use. This DLCI must have been previously defined on the `set frdlci` command.

frport
is the port on the PortServer II that this user accesses for frame relay connections. This port must be within the range of ports specified on the `ports` field.

idletimeout
is the maximum time in seconds that a PPP, SLIP, or CSLIP user's connection can be idle before the user is disconnected

The range is 0 to unlimited. The default is 0, which means that the user will never be disconnected for lack of connection activity.

ipaddr
is the remote PPP, SLIP, CSLIP, or frame relay user's IP address. Possible values are

- An IP address in dotted decimal format. SLIP, CSLIP, and frame relay users must be defined with a particular IP address.
- `negotiated or 0.0.0.0.`, which means that the peer pro-

vides an address.

- `ippool` or `255.255.255.254`, which means that PortServer II provides an address for the peer from its IP address pool.

ipmask
is the IP mask to apply to the address specified on the `ipaddr` field

keepup
is the name of a keepup filter, defined with the `set filter` command, that PortServer II uses to maintain PPP, SLIP, and CSLIP connections. A keepup filter is one in which the reception of certain types of packets are indications to PortServer II that the connection should be maintained.

localbusydly
is the number of seconds that PortServer II delays before retrying to establish a PPP, SLIP, or CSLIP connection that could not be made because local ports were unavailable.

The range is 0 to an unlimited number of seconds. The default is 0, which means there will be no delay.

localipadr
is the IP address of the local end of a PPP, SLIP, CSLIP, or frame relay link. If this is set to 0.0.0.0, the IP address for PortServer II's ethernet interface is used.

loginscript
is the name of a script, defined with the `set script` command, to use to log in to a remote system. This field is required for outbound PPP, SLIP, and CSLIP connections unless the remote system does not require a login and password.

logpacket
is the name of a filter designed to write to the log file whenever PortServer II handles a particular type of packet on PPP, SLIP, and CSLIP connections.

maxports
is the maximum number of ports that a telnet or rlogin user can be logged into at the same time

0 means that the user can be simultaneously logged into all ports specified on the `ports` field

menu
`index-num`
is the menu, identified by an index number, presented to a telnet or rlogin user

`off` and `0` (zero)
means that no menu is presented to the user

The default is `off`.

mtu
is the maximum transmission unit (frame size in bytes) to use for this PPP, SLIP, or CSLIP connections. For PPP connections, the MTU is negotiated, so enter 1500, the largest size PortServer II will permit the remote host to send.

For PPP, SLIP, and CSLIP users, the range is 296 to 1500 bytes, and the default is 1500 bytes.

n1,n2...
are phone numbers (up to 10) to dial to request a PPP, SLIP, or CSLIP outgoing connection, which dialer scripts reference. If you enter more than one number, when PortServer II encounters a busy signal, it tries these numbers in the order specified here.

You can enter this number as digits only, with dashes (-) separating digits, or with commas.

name
is the name that identifies this user

netrouting
specifies how RIP routing updates are handled on connections to this PPP, SLIP, CSLIP, or frame relay user. Use this field only if the user is an IP router.

`off`
means that this user is not included in RIP updates

`send`
means propagate RIP updates to this user, but do not accept RIP updates from this user

`receive`
means accept RIP updates from this user, but do not send RIP updates to this user

`both`
means RIP updates will be sent to and received from this user

The default is `off.`

netservice
`on`
allows PPP, SLIP, or CSLIP connections for the user

`off`
allows no PPP, SLIP, or CSLIP connection for the user

network
displays network-related parameters associated with the user specified on the `name` field

newname
is a new name for a previously defined user

outgoing
`on`
means that the user can initiate outgoing connections

`off`
means that the user can **not** initiate outgoing connections

p1, p2 ...
are integers (1-9) that can be used in the variable fields of login or dialer scripts

papid
is a character string that identifies the PPP user using PAP authentication. This is equivalent to a user (or login) name. The string must be 16 or fewer characters and must be recognized by the peer.

pappasswrd
is a character string that authenticates the PPP user using PAP authentication. This is equivalent to a password. The string must be 16 or fewer characters and must be recognized by the peer.

passive
`on`
means that PortServer II waits for the remote system to begin PPP negotiations

`off`
means that PortServer II may initiate PPP negotiations

The default is `off`.

Note: Do not set both sides of a PPP connection to `passive=on`.

passpacket
is the name of a filter designed to allow packets meeting filter criteria to pass through PortServer II serial ports on PPP, SLIP, and CSLIP connections

password
`on`
means a PortServer II password is required of this user

`off`
means a password is not required of this user

The default is `on`.

ports
is a port or range of ports that this user can access

pppauth
determines whether PPP authentication is required and, if so, what kind

`none`
means the remote user does not require PPP authentication

`chap`
means CHAP authentication is required

`pap`
means PAP authentication is required

`both`
means both CHAP and PAP authentication is required

The default is `both`.

protocol
is the protocol this user uses:

- `frame` means use frame relay
- `ppp` means use PPP
- `slip` means use SLIP (or CSLIP)

The default is `PPP`.

protocompress
`on`
means PortServer II attempts to negotiate protocol compression on PPP connections

`off`
means PortServer II will **not** negotiate protocol compression

The default is `on`.

range
identifies an entry or range of entries in the user table to display or remove

rmtbusydly
is the number of seconds that PortServer II delays before reattempting a connection to a remote system that was previously inaccessible

The range is 0 to an unlimited number of seconds. The default is 0, which means no delay.

sessiontimeout
is the maximum time in seconds that a user may be connected

The range is 0 to an unlimited number of seconds. The default is 0, which means that there is no limit.

vjslots

is the number of slots used for Van Jacobson header compression. The number of slots you configure should correspond to the expected maximum number of simultaneous connections using Van Jacobson header compression on this WAN interface. To avoid excessive processor usage, configure only the number you think you will need.

The default is 16 and the range is 0 to 256.

Command Examples

Displaying the Entire User Table

In this example, the `set user` command displays the entire user table.

```
set user
```

Displaying a Range of Entries in the User Table

In this example, the `set user` command displays a range of entries in the user table.

```
set user range=2-7
```

Removing a User from the User Table

In this example, the `set user` command removes a user from the user table.

```
set user rmuser=user4
```

Configuring an Autoconnect User

In this example, the `set user` command configures an autoconnect user.

```
set user name=user4 autoconnect=on
autohost=199.193.150.10 autoport=23
defaultaccess=autoconnect
```

Configuring a PPP User

In this example the `set user` command configures a remote PPP user.

```
set user name=user4 protocol=ppp addrcompress=on
pppauth=pap papid=user4-id pappasswrd=howdy
compression=vj defaultaccess=netservice
ippaddr=ip-pool netservice=on range=4-7
```

snmp

Introduction

Purpose Use the `snmp` command to configure, enable, and disable PortServer II's SNMP (Simple Network Management Protocol) agent.

Required Privileges The `snmp` command requires `root` privileges.

Related Information None.

Command Syntax

Syntax

```
snmp [auth_trap={off|on}] [contact=administrator]
[location=location-string] [name=name-string]
[run={off|on}] [trap_dest=ipaddress]
```

Command Fields

Field Descriptions

auth_trap

`on`
means the agent sends an authentication trap to the SNMP manager when an authentication error occurs

`off`
means the agent silently ignores SNMP requests that fail authentication

The default is `off`.

contact
is a text string that identifies a contact person (usually an administrator). The entry must be surrounded by quotation marks if there are spaces in the text.

location
is a text string that describes PortServer II's location. The entry must be surrounded by quotation marks if there are spaces in the text.

name
is a text string that identifies PortServer II. The entry must be surrounded by quotation marks if there are spaces in the text.

run

`on`
starts the SNMP daemon

`off`
means the SNMP daemon will not start

The default is `off`.

trap_dest
is the IP address of the system to which the agent should send traps

Command Examples

Displaying SNMP Configuration

In this example, the `snmp` command displays the SNMP configuration.

```
snmp
```

Configuring All Options

In this example, the `snmp` command configures SNMP.

```
snmp run=on auth_trap=on trap_dest=190.175.178.73
location=Manufacturing-1 name=PServer1
contact=”Gary Groven”
```

status

Introduction

Purpose Use the `status` command to display information about your current telnet session.

Required Privileges Anyone can execute the `status` command.

Related Information See the `close` command. Typically you use the `status` command to determine which telnet sessions to close.

Command Syntax

Syntax Here is how you issue the `status` command.

```
status
```

Command Example

Example In this example, the `status` command provides information on the user's current telnet session.

```
status
```

telnet

Introduction

Purpose Use the `telnet` command to establish a telnet session with a remote system.

Required Privileges Anyone can execute the `telnet` command.

Related Information None.

Command Syntax

Syntax Here is how you issue the `telnet` command.

```
telnet {hostname | host-ip-addr} [tcp-port]
```

Command Fields

Field Descriptions

hostname
is the name of the host to which you want a telnet session. DNS must be configured on the PortServer II to use this option.

host-ip-addr
is the IP address of the host to which you want a telnet session

tcp-port
is the TCP port assigned the telnet application on the remote system. The default is 23, the port typically used for telnet.

Command Example

Telnetting Using a Host Name In this example, the `telnet` command establishes a telnet session using a host name. The default TCP port (23) is used.

```
telnet host1
```

Telnetting Using an IP Address In this example, the `telnet` command establishes a telnet session using an IP address. The default TCP port (23) is used.

```
telnet 192.192.150.28
```

Telnetting to a PortServer II Port from the LAN In this example, a user on the LAN initiates a telnet connection to port 4 on a PortServer II named `host1`.

```
telnet host-1 2004
```

traceroute

Introduction

Purpose Use the *traceroute* command to display a list of routers through which an IP packet passes on its way to a particular destination.

Required Privileges Anyone can issue the traceroute command.

Related Information None.

Command Syntax

Syntax Here is the syntax for issuing the traceroute command.

```
traceroute ip-addr|name
```

Command Field

Field Description

ip-addr | name
is either the IP address or the DNS name of the host to which you want a route traced

Command Examples

Tracing a Route Using an IP Address In this example, the traceroute command traces a route to a host using the specified IP address.

```
traceroute 199.150.150.74
```

Tracing a Route Using a Name In this example, the traceroute command traces a route to a host using a host name.

```
traceroute poe
```

wan

Introduction

Purpose Use the `wan` command to

- Initiate and control PPP, SLIP, CSLIP, and frame relay connections
- Display the status of current connections

Required Privileges Anybody can issue the `wan` command to display the status of WAN connections. Root privileges are required to initiate or control WAN connections.

Related Information `set modem` command.

Command Syntax

Initiate and Control Syntax Use this form of the `wan` command to initiate and control WAN connections:

```
wan [close=filtername] [initmodem=range]
[start=filtername] [testmodem=range]
[verify={all|username}]
```

Display Syntax Use this form of the `wan` command to display the status of current WAN connections:

```
wan [range=range]
```

Command Fields

close
closes an outbound connection. The connection is identified by the name of the filter (see the `set filter` command) used to bring up the connection.

initmodem
executes the modem initialization script associated with the port or ports specified

range
is a port or range of ports

start
places the connection in the start-up condition. The connection is identified by the name of filter (see the `set filter` command) used to bring the connection up.

testmodem
executes the modem test script associated with the port or ports specified. See the discussion on the `set modem` command for information on test scripts.

verify
`all`
verifies that that all connections are associated with real users, that is, users that are defined in the configuration

`wanname`
verifies that the user has been defined in the configuration

Note: Only incorrectly configured WAN interfaces produce a message in response to this command. If WAN interfaces are configured correctly, no message is returned.

Command Examples

Closing a WAN Interface

In this example, the `wan` command closes a WAN connection.

```
wan close=filter-ppp01
```

Starting a WAN Interface

In this example, the `wan` command initiates a WAN connection.

```
wan start=filter-ppp01
```

Displaying WAN Status Information

In this example, the `wan` command displays the status of the connection on port 2.

```
wan range=2
```

who

Introduction

Purpose Use the `who` command to display a list of current PortServer II users.

Required Privileges Anyone can issue the `who` command.

Related Information None.

Command Syntax

Syntax Here is how you issue the `who` command.

```
who
```

Command Example

Example

```
who
```

Index