Command Reference

Digi Connect® Family

Digi Connect Family:
Digi Connect SP, Digi Connect Wi-SP, Digi Connect ME, Connect ME4,
Digi Connect Wi-ME, Digi Connect EM, Digi Connect Wi-EM,
Digi Connect ES Family (Digi Connect ES 4/8/16 devices)

ConnectPort® TS Products:
ConnectPort TS 8, ConnectPort TS 8 MEI

Digi Cellular Family:
Digi Connect WAN, Digi Connect WAN VPN,
Digi Connect WAN Sync, Digi Connect WAN IA, ConnectPort™ WAN VPN

Connect WAN Family:
Digi Connect® WAN, Digi Connect WAN VPN,
Digi Connect WAN IA, Digi Connect WAN Sync

ConnectPort® Family:
ConnectPort WAN VPN

ConnectPort® X Family:
ConnectPort X8

ConnectPort Display

www.digi.com

90000566_H

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 this 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.

Contents

Chapter 1 Introduction

This book describes the commands in the command-line interface for several Digi product families, listed below. This chapter provides the following:

- A quick reference showing the commands used to configure features or perform configuration tasks from the command line.
- Basic information that applies to all commands, including navigation and editing keys, displaying online help, abbreviating commands, syntax conventions, and entering special characters in string values.
- How to access the command line.
- How to configure an IP address for a Digi device from the command line, if an address has not already been assigned.
- Information about user models and user permissions in Digi Connect products, and how they affect the commands you can issue.

Digi Connect Products and Families to Which This Book Applies

This manual documents the command-line interface for the following Digi products:

- The **Digi Connect Family**, which includes these products:
  - Digi Connect SP
  - Digi Connect Wi-SP
  - Digi Connect ME
  - Digi Connect Wi-ME
  - Digi Connect EM
  - Digi Connect Wi-EM
  - Digi Connect ES Family (Digi Connect ES 4/8/16 devices)
  - Connectport TS 8
  - ConnectPort TS 8 MEI
- The **Digi Cellular Family**, which includes these products:
  - Digi Connect WAN
  - Digi Connect WAN VPN
  - ConnectPort WAN VPN
  - Digi Connect WAN IA
  - Digi Connect WAN Sync
- The **ConnectPort X Family**, which includes these products:
  - ConnectPort X8
- ConnectPort Display

Quick Reference for Configuring Features

The following table shows common features that can be configured from the command line, and the commands used to configure each feature. If you are viewing the PDF file of this document, click the commands in the “Commands” column to go to the command descriptions.

Feature/Task	Commands
Alarms	"set alarm" on page 70.
Autoconnection (automatically connect a user to a server or network device)	"set autoconnect" on page 81. "set serial" on page 189. "set tcpserial" on page 212.
Bisynchronous communications	"set bsc" on page 85
Configuration management/administration	Backup/restore a configuration from a TFTP server on the network: "backup" on page 18. Update firmware: "boot" on page 19. Reset configuration to factory defaults: "revert" on page 61; or boot action=factory (see "boot" on page 19). Reboot the device: "boot" on page 19.
Connectware Manager/Remote Management: Connectware Device Protocol configuration settings	"set mgmtconnection" on page 137. "set mgmtglobal" on page 140. "set mgmtnetwork" on page 143.
Custom menus	"set menu" on page 132.
DHCP (Dynamic Host Configuration Protocol)	To configure a DHCP server: "set dhcpserver" on page 96. To manage and show status of a DHCP server: "dhcpserver" on page 23.
Display current configuration settings in a device	"show" on page 249.
Dynamic DNS (DDNS)	"set ddns" on page 92.
Ekahau Client™ device-location software	"set ekahau" on page 104
Ethernet settings for wired devices	"set ethernet" on page 107.
General Purpose Input/Output (GPIO) pins	"set gpio" on page 113. "set alarm" on page 70.
Help on device commands	"help and ?" on page 40.
Host name for a device (Specify a name for the device)	"set host" on page 118.
Industrial Automation (IA)	• “set profile profile=ia” See "set profile" on page 172. • "set ia" on page 119. • For additional information on configuring Industrial Automation, see this web site: http://www.digi.com/support/ia
IP address settings	"set network" on page 149.

Feature/Task	Commands
IP Forwarding and Network Address Translation (NAT)	"set forwarding" on page 109. "set nat" on page 146.
IP pass-through	"set passthrough" on page 153
Mobile (Cellular) features:	
• Provisioning CDMA cellular modules	To display existing provisioning parameters: “display provisioning” -- see "display" on page 27 To provision the CDMA module: "provision" on page 53
• Mobile service provider and connection settings	The only mobile connection setting that can be set from the command line is the inactivity timeout, (“set pppoutbound (rx_idle_timeout=*timeout*.) The Inactivity timeout specifies the time, in seconds, after which if no data has received over the link, the mobile connection will be disconnected and re-established. "set pppoutbound" on page 166.
• SureLink™ Settings	"set surelink" on page 202.
Multiple Electrical Interface (MEI)	"set switches" on page 208
Point to Point Protocol (PPP)	"set pppoutbound" on page 166.
Port buffering	"display buffers" on page 37. "set buffer" on page 90.
Port profiles: sets of preconfigured serial-port settings for a particular use	"set profile" on page 172.
Python® program storage and execution on Digi devices	To learn about the Python programming language and writing programs: see the *Digi Python Programming Guide*. To configure Python programs to execute when the Digi device boots: "set python" on page 183. To manually execute a Python program from the command line: "python" on page 58.
RCI over Serial	"set rciserial" on page 184.
RealPort (COM port redirection) configuration	"set realport" on page 185. See also the *RealPort Installation Guide*.
Remote access through VNC (Virtual Network Computing) protocol	"set vncclient" on page 226.
Remote login (rlogin)	"rlogin" on page 66.
Reverting configuration settings	"revert" on page 61.
RTS Toggle	"set rtstoggle" on page 187.

Feature/Task	Commands
Serial port configuration	Serial port communication options: "set serial" on page 189. Port profiles: "set profile" on page 172. RCI serial mode: "set rciserial" on page 184. RTS Toggle: "set rtstoggle" on page 187. TCP serial connections: "set tcpserial" on page 212. UDP serial characteristics: "set udpserial" on page 216.
Security, users, user access permissions, and user groups	See "User Models and User Permissions in Digi Connect Products" on page 14 for a discussion of how users and access permissions are implemented in Digi Connect products.To create users and change user names: "set user" on page 220. To control access to inbound ports:"set service" on page 191. Enable/disable command-line access: "set term" on page 215. To issue new password to user: "newpass" on page 51. To set permissions associated with various services and commands: "set permissions" on page 157. To add or remove user groups, change group configuration attributes, or display group configuration attributes: "set group" on page 115. To suppress user login: "set login" on page 131.
Simple Network Management Protocol (SNMP)	To configure SNMP: "set snmp" on page 198. To enable/disable SNMP service: "set service" on page 191. To enable/disable SNMP alarm traps: "send" on page 67.
Set system information: assign system-identifying information to a device	"set system" on page 211.
Socket tunnel settings	"set socket_tunnel" on page 200.
Statistics for your Digi device	"info" on page 41.
Status of your Digi device	"display" on page 27. "status" on page 254. "who" on page 258.
Digi SureLink™	"set surelink" on page 202.
Telnet to network devices	"telnet" on page 255. "mode" on page 50. "send" on page 67.
Terminal Emulation for ConnectPort Display	"set putty" on page 175.
Video settings for ConnectPort Display	"set video" on page 225.
VPN (Virtual Private Network)	To configure VPN: Using the Web user interface is recommended. See "set vpn" on page 228. To manage VPN: "vpn" on page 256.

Feature/Task	Commands
Wireless devices	"set wlan" on page 241. "set ekahau" on page 104.

Access the Command Line

To configure devices using commands, you must first access the command line, and then log on as needed.

This procedure assumes that you have already configured the Digi device with an IP address.

1. To access the Command-Line Interface for the Digi device, enter the following command from a command prompt on another networked device, such as a server:

   ```
   #> telnet ip address
   ```

   where *ip address* is the Digi device's IP address. For example:

   ```
   #> telnet 192.3.23.5
   ```

2. If user authentication has been set up for the device, (that is, a user-name and password have been set up for the device), a login prompt is displayed. If you do not know the user name and password for the device, contact the system administrator who configured the device. The default username is “root” and the default password is “dbps.”

Configure an IP Address

If the device to which you will be issuing commands has not already been assigned an IP address, or if the IP address needs to be modified from its initial configuration, see the Digi product's *User's Guide* for details on configuring an IP address.

Basic Command Information

Navigation and Editing Keys

Use the keys listed in the table to navigate the command line and edit 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 or Ctrl+h
Delete the character under the cursor.	Delete
Scroll back through commands.	Ctrl+p
Scroll forward through commands.	Ctrl+n
Execute the command.	Enter

Displaying Online Help

Help is available for all commands. The table describes how to access it.

For information on...	Type
All commands	? (with no additional options)
A specific command	help [*command*] OR [*command*] ? **Example:** help info **Example:** info ? **Example:** set alarm ?

Abbreviating Commands

All commands can be abbreviated. Simply supply enough letters to uniquely identify the command.

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 vertical bar, |.
- Non-italicized text indicates literal values, that is, options 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 option. For example, *filename* means that the name of a file is required in the option.

Entering Special Characters in String Values

Several commands have options that are string values, for example the “set alarm” command’s “match” option and the “set autoconnect” command’s “connect_on_string” option.

Escape Sequences for Special Characters

Special characters can be entered in strings using the following escape sequences:

Escape Sequence	Processed as:
*	Match any character. This escape sequence is only available on the “set alarm match=string” option.
\a	Alert character.
\b	Backspace character.
\f	Form-feed character.
\n	New-line character.
\r	Carriage-return character.
\s	Acts as a separator between characters. This sequence allows you to enter a string such as “\xB8\s4” where you want the B8 translated as a hexadecimal character separate from the numeric character 4.
\t	Horizontal tab character.
\v	Vertical tab character.
\\	Backslash character (\ ).
\x*N*	A hexadecimal number, where *N* is up to 20 hexadecimal digits. For example: `\x10\x2`
N	An octal byte, where *N* is up to 3 octal digits. For example: `\2` or `\208`

Length Limitations on String Values

String values for certain command options have specific limitations on the maximum total string value including special characters, and the maximum parsed value (that is, the character-string length when any escape sequences in the string are processed). The option descriptions note these maximum lengths.

User Models and User Permissions in Digi Connect Products

The user model in a Digi Connect product influences the commands that users can issue. There are three user models implemented in the various Digi Connect and ConnectPort devices: one-user model, two-user model, and more than two-user model.

Identifying the User Model for Your Digi Connect Product

To determine which user model is implemented in your Digi Connect product, issue a “show user” or “set user” command (see "show" on page 249 and "set user" on page 220). In the command output, note how many user IDs are defined: one, two, or more than two. You can also issue a “set user ?” command and note the range for the “id=” option. If the “id=” option is not listed, there is one user. Otherwise, the range for user IDs is displayed.

One-user Model

In the one-user model, by default there is no login prompt, and the default name for user 1 is “root.”

To enable the login prompt, you must issue a “newpass” command with a password length of one or more characters (see "newpass" on page 51). Once a password is enabled, issuing a “newpass” command with a zero-length password will disable it.

- User 1 has a default name of “root.”
- User 1 has permissions that enables it to do all commands. Permissions cannot be altered.

Two-user Model

- User 1 has a default name of “root.” This user is also known as the administrative user.
- User 1 has default permissions that enables it to issue all commands.
- Permissions for User 1 can be changed to be less than the default root permissions.
- User 2 is undefined. That is, it does not exist by default, but it can be defined.
- When defined, User 2 has a limited set of permissions, defined by the “set permissions” command (see "set permissions" on page 157).
- Permissions for User 2 can be changed to be either greater than or less than its default.

More than Two-user model

User definitions are exactly the same as the two-user model, with the addition of user groups and more users. The “set group” command defines user groups (see "set group" on page 115).

Login Suppression Feature

The login prompt can be disabled by issuing the “set login” command. See "set login" on page 131.

Increasing Security for Digi Device Users

As needed, you can enforce additional security for device users. For example, you can use the autoconnect feature, where a user is automatically connected to another system without accessing the Digi device’s command line. See "set autoconnect" on page 81.

Chapter 2 Command Descriptions

This chapter provides a description of each command in the Digi Connect Family Command-Line Interface.

Verifying Device Support for Commands

To verify whether a Digi Connect device supports a particular command or command options, and to get the allowed ranges and limits for command options, you can enter several commands. For example:

- “help” displays all supported commands for a device.
- “?” displays all supported commands for a device.
- “set ?” displays the syntax and options for the “set” command. You can use this to determine whether the device includes a particular “set” command variant.
- “help set” displays syntax and options for the “set” command.
- “set serial ?” displays the syntax and options for the “set serial” command.
- “help set serial” displays the syntax and options for the “set serial” command.

Some options may become available in new firmware revisions or before new documentation is released.

Some commands relate only to particular features unique to specific Digi products. For example, the “set wlan” command applies only to wireless products. Other commands may have options that are specific to features that are not available on all devices. For example, the “display” command’s “mobile” option applies only to Digi Cellular Family products.

backup

Devices supported

This command is supported in all Digi Connect products.

Purpose

Save the device configuration to a TFTP server located on the network, or restores the configuration from a saved copy on the TFTP server.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions backup=execute” to use this command. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

```
backup [to=serveripaddress[:filename]|
   from=serveripaddress[:filename]|print]
```

Options

to=*serveripaddress*[*:filename*]
The IP address of the TFTP server to which the configuration will be saved, and the filename that the configuration will be saved as. If a filename is not specified, the **default filename of config.rci** is used.

from=*serveripaddress*[*:filename*]
The IP address of the TFTP server and the filename from which the configuration will be restored. If a filename is not specified, the **default filename of config.rci** is assumed.

print
Prints out the current device configuration.

Example

```
#> backup from=10.0.0.1:config.rci
```

See also

"set rciserial" on page 184. The “set rciserial” command allows a configuration file to be loaded over a serial port when the DSR input signal is high.

boot

Devices supported

This command is supported in all Digi Connect products.

Purpose

Reboots the device server, restores the device configuration to factory default settings, or loads new firmware files (both EOS and POST images) from a TFTP server.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions boot=execute” to use this command. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Reboot the device server

```
boot action=reset
```

Restore configuration defaults

```
boot action=factory
```

Load new firmware or POST file into flash ROM from a TFTP host

```
boot load=host ip address:load file
```

Options

action

The action to be performed.

factory

Resets the entire configuration to factory defaults, then reboots the device.

reset

Reboots the device.

load

The firmware to be loaded.

host ip address

The IP address of a host with new firmware or POST file, which is then burned into flash ROM. The host must be running a TFTP server.

load file

The name of a firmware file or POST file. The software automatically detects the type of file and performs the appropriate load operation.

Examples

Restore configuration defaults

This example reloads the firmware stored in flash ROM and resets the configuration to factory defaults then reboots the device.

```
#> boot action=factory
```

Reboot using the current firmware and configuration

This example reboots the device and uses the current firmware and configuration stored in flash ROM.

```
#> boot action=reset
```

Reboot using firmware from a boot host

This example loads the firmware stored on the TFTP host into flash ROM. A reboot is required to use the new firmware.

```
#> boot load=10.0.0.1:firmware.bin
```

See also

"revert" on page 61.

close

Devices supported

This command is supported in all Digi Connect products.

Purpose

Closes active connect, Rlogin, and Telnet sessions; that is, sessions opened by “connect,” “rlogin,” or “telnet” commands.

The “close” command is associated with the sessions displayed by the “status” command.

A “close” command issued without any options closes the current connection.

To issue the “close” command, you must escape the active session. Do this by pressing the escape key defined for your session type. The following table lists default escape keys.

Session Type	Default Escape Keys
Connect	Ctrl+[+Enter
Rlogin	~+Enter
Telnet	Ctrl+]+Enter

Syntax

```
close [{*|connection number}]
```

Options

***
Closes all active sessions.

connection number
Identifies the session to close by its session number.

Examples

Close a session identified by number

```
#> close 1
```

Close the current session

```
#> close
```

Close all active sessions

```
#> close *
```

See also

- "kill" on page 49. The kill command has a broader effect than close, and lets you kill connections from the global list. That is, it is not limited to sessions associated with the current connection.
- "status" on page 254 for information on displaying status information on active sessions.
- "connect" on page 22
- "rlogin" on page 66
- "telnet" on page 255

connect

Devices supported

This command is supported in all Digi Connect products.

Purpose

Used to make a connection, or establish a session, with a serial port.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions connect=execute” to use this command. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

There are several ways to create and manage connections:

Create a single connection

```
connect serial port
```

Create multiple connections

Issue multiple “connect” commands.

Temporarily suspend a connection

Escape the active session by pressing Ctrl [.

Temporarily suspend a connection and return to the command line

Press the escape character and then the Enter key.

Switch between active sessions (without first escaping to the command line)

Press the escape character and then the number of the session you wish to enter, for example, Esc+1.

Pressing the connect escape character twice causes the next session to appear, enabling you to easily page through sessions.

Options

serial port

The number of the port on which to establish a connection.

Example

Create a connection to port 1

```
#> connect 1
```

See also

- "close" on page 21 for information on ending a session.
- "reconnect" on page 60 for information on reestablishing a port connection.

dhcpserver

Devices supported

This command is supported in Digi Cellular Family products only.

Purpose

Used for managing and showing the status of a DHCP server, including managing the leases for IP addresses, restarting, running, and shutting down the DHCP server, and displaying DHCP server status information.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions connect=execute” to use this command. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

```
dhcpserver [deletelease={ip address|all}]
   restart
   run
   shutdown
   status
```

Options

deletelease={*ip address*|all}

Specifies how to handle IP address leases. You may remove leases from the DHCP Server while it is running.

ip address

Removes a specific lease from the DHCP server.

all

Removes all IP address leases from the DHCP server.

Removing a lease will cause the associated IP address to be returned immediately to the available address pool. Any IP address in this available address pool may be served in a new lease to a DHCP client.

If you stop or restart the DHCP server, or if you reboot the Digi Connect product, all knowledge of the IP address leases will be lost. All leased addresses, except for reservations, will be returned to the available address pool and may be served in a new lease to a DHCP client.

Static lease reservations will always be displayed in the lease list. These reservation leases may be removed, but a new lease will be created immediately. To disable or permanently remove a reservation, use the “set dhcpserver” command. See the examples for "set dhcpserver" on page 96.

restart

Restarts the DHCP server.

run

Runs (starts) the DHCP server if it not already started.

shutdown

Shuts down the DHCP server.

status

Displays DHCP server status information.

Example

Display DHCP server status

```
#> dhcpserver status

Device Networking Status:

  IP address           : 10.30.1.188
  Subnet mask          : 255.255.255.0
  Default gateway      : 10.6.6.6
  Static IP configured : yes
  Uptime               : 0 days + 21:00:44

DHCP server status: running

  Uptime               : 0 days + 21:00:36

Scopes configured in server:

Scope 1:
  Name                 : eth0
  IP address           : 10.30.1.188
  Subnet mask          : 255.255.255.0
  Starting IP address  : 10.30.1.190
  Ending IP address    : 10.30.1.198
  Routers              : 10.30.1.188
  DNS servers          : 209.183.48.10 209.183.48.11
  Lease duration       : 3600 (seconds)
  Offer delay          : 500 (milliseconds)
  Addr conflict detect : disabled
Address reservations:
    Reservation 1:
      IP address           : 10.30.1.135
      Client ID            : 00:40:9D:24:73:F8
      Lease duration       : 3600 (seconds)
    Reservation 2:
      IP address           : 10.30.1.192
      Client ID            : 02:40:9D:24:73:F8
      Lease duration       : using scope lease duration
    Reservation 3:
      IP address           : 10.30.1.195
      Client ID            : 00:09:26:19:51:05
      Lease duration       : using scope lease duration
    Reservation 4:
      IP address           : 10.30.1.196
      Client ID            : 00:09:26:19:51:06
      Lease duration       : using scope lease duration
    Reservation 5:
```

```
    IP address           : 10.30.1.197
    Client ID            : 00:09:26:19:51:07
    Lease duration       : using scope lease duration
Address exclusions:
  none configured
Lease Records:
                  Client ID         Lease Time in Seconds
  IP Address      (MAC Address)     Duration   Remaining  Lease Record Status
  --------------- ----------------- ---------- ---------- ------------------
  10.30.1.135     00:40:9D:24:73:F8 3600       1834       Reserved (active)
  10.30.1.192     02:40:9D:24:73:F8 3600       N/A        Reserved (inactive)
  10.30.1.195     00:09:26:19:51:05 3600       N/A        Reserved (inactive)
  10.30.1.196     00:09:26:19:51:06 3600       N/A        Reserved (inactive)
  10.30.1.197     00:09:26:19:51:07 3600       N/A        Reserved (inactive)
```

Delete a lease

```
dhcpserver deletelease=10.30.1.135
```

Delete all leases

```
dhcpserver deletelease=all
```

Lease status values

Following are descriptions of the lease status values. The amount of time that a lease table entry will remain in each state also is stated. Note that after a lease is deleted, the associated IP address is returned to the available address pool.

Assigned (active)
A lease is currently assigned and active for the given client. The client may renew the lease, in which case the lease remains in this state.

Assigned (expired)
A lease has expired and is no longer active for the given client. A lease in this state will remain for 4 hours, after which it is deleted. If the same client requests an IP address before the lease is deleted, it will be given the same IP address previously served to it.

Reserved (active)
A lease for an address reservation is currently active for the given client. A reservation lease will remain indefinitely, although the status may alternate between active and inactive.

Reserved (inactive)
A lease for an address reservation is currently inactive for the given client. A reservation lease will remain indefinitely, although the status may alternate between active and inactive.

Reserved (unavail)
A lease for an address reservation was offered to a client, but that client actively declined to use the IP address. Typically this is because the client determined that another host on the same subnetwork is already using that IP address. Upon receiving the client's decline message, the

DHCP server will mark the address as unavailable. The lease will remain in this state for 4 hours, after which it is reverts to the Reserved (inactive) status.

Offered (pre-lease)

A lease has been offered to the given client, but that client has not yet requested that the lease be acknowledged. It may be that the client also received an offer from another DHCP server, in which case this offer will expire in approximately 2 minutes. If the client requests this lease before that 2 minute interval elapses, this lease will change status to Assigned.

Released

A lease was previously assigned to the given client, but that client has proactively released it. A lease in this state will remain for 1 hour, after which it is deleted. If the same client requests an IP address before the lease is deleted, it will be given the same IP address previously served to it.

Unavailable Address

A lease was offered to a client, but that client actively declined to use the IP address. Typically this is because the client determined that another host on the same subnetwork is already using that IP address. Upon receiving the client's decline message, the DHCP server will mark the address as unavailable. The lease will remain in this state for 4 hours, after which it is deleted.

This status may also occur if the DHCP Server determines that the IP address is in use before it offers the address to a client. See the "set dhcpserver" command option "conflictdetect" option.

See also

- "set dhcpserver" on page 96.
- The Web user interface’s help text for Network Settings, which includes information on configuring DHCP server settings and managing DHCP servers.

display

Devices supported

This command is supported in all Digi Connect products.

Purpose

Displays status information for a Digi Connect device. The “display” command’s focus is on real-time information. In contrast, the “info” command displays statistical information about a device over time, while the “status” command displays the status of outgoing connections (connections made by “connect,” “rlogin,” or “telnet” commands).

Status information that can be displayed includes:

- General product information, including the product name, MAC address, boot, post, and firmware versions, memory usage, CPU utilization, and uptime, or the amount of time since the device was last booted.
- Access control status information.
- ARP table information.
- Contents of a port buffer (see also "display buffers" on page 37).
- Dynamic DNS (DDNS) service status information.
- GPIO signals.
- Memory usage information only.
- Mobile (cellular modem) status information.
- Network Address Table (NAT) status information.
- Active network device interfaces and their status.
- IP pass through status.
- Point-to-Point Protocol (PPP) status information, including SureLink statistics.
- Provisioning parameters in a Digi Cellular Family device’s CDMA cellular module.
- Route Table entries.
- Security Association (SA) database entries (can also be displayed with other Virtual Private Network (VPN) information).
- Security Policy Database (SPD) entries (can also be displayed with other Virtual Private Network (VPN) information)
- Serial modem signals (DTR, RTS, CTS, DSR, DCD).
- Socket status information.
- Multiple Electrical Interface (MEI) switch settings currently defined for ports, on devices supporting MEI.
- Current TCP and UDP session and listener information.
- Uptime information only.
- Version information for Boot, POST and EOS firmware, and Digi part numbers for those items.

- Virtual Private Network (VPN) information, including Security Association (SA) database entries and Security Policy Database (SPD) entries.
- Typical wireless LAN (WLAN) parameters for wireless devices.

100% CPU Utilization may indicate encryption key generation is in-progress

There may be instances when a “display device” command returns a CPU utilization of 100%. A CPU usage this high may indicate that encryption key generation is in-progress. On initial boot, the Digi device generates some encryption key material: an RSA key for SSL/TLS operations, and a DSA key for SSH operations. This key-generation process can take as long as 40 minutes to complete. Until the corresponding key is generated, the Digi device will be unable to initiate or accept that type of encrypted connection. It will also report itself as 100% busy but, since key generation takes place at a low priority, the device will still function normally. On subsequent reboots, the Digi device will use its existing keys and will not need to generate another unless a reset to factory defaults is done, which will cause a new key to be generated on the next reboot.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions display=execute” to use this command. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

```
display {accesscontrol|arp|buffers|ddns|device|gpio|memory|mobile|
   nat|netdevice|passthrough|pppstats|provisioning|route|sadb|
   serial|sockets|spd|switches|tcp|udp|uptime|version|vpn|wlan}
```

Options

accesscontrol
Displays access control status information.

arp
Displays ARP table entries.

buffers
Displays the contents of a port buffer. This option is covered in more detail in "display buffers" on page 37.

device
Displays general product information including product name, MAC address, boot, post, and firmware versions, memory usage, CPU utilization, and uptime. The information displayed by this option is the same as that displayed by the “info device” command (see "info" on page 41).

ddns
Displays status information for the Dynamic DNS (DDNS) service. See "set ddns" on page 92 for information on the DDNS service.

gpio
Displays GPIO signals.

memory
Displays general memory, network memory, and streams memory usage.

mobile
Displays mobile (cellular modem) status information. Applies to Digi Cellular Family products only. To display statistics associated with the SureLink feature, use the “pppstats” option.

nat
Displays Network Address Table (NAT) status information.

netdevice
Displays the active interfaces on the system, for example, PPP and Ethernet interfaces, and their status, such as “Closed” or “Connected.”

passthrough
Displays status of the IP pass-through mode, enabled by the “set passthrough” command. See "set passthrough" on page 153.

pppstats
Displays status and activity information for a Point-to-Point Protocol (PPP) link, including SureLink statistics. See "Information returned by “display pppstats”"" on page 32 for descriptions.

provisioning
Displays the current provisioning information in the Digi device’s CDMA cellular module.

Before using the “provision” command to provision the CDMA module, It is recommended that you use this option to determine which parameters are already set in the module. See "provision" on page 53.

Important Use of the “provision” and “display provisioning” commands requires that any existing PPP sessions be closed.

route
Displays Route Table entries.

sadb
Displays the contents of the Security Association (SA) database. The SA database lists the connections to VPN servers. Each entry identifies the subnets traffic is being routed between, and the security protocols chosen for the VPN tunnel. Applies to Digi Cellular Family products only.

serial
Displays serial modem signals (DTR, RTS, CTS, DSR, DCD).

sockets
Displays information about how socket resources are being used by the system.

spd
Displays Security Policy Database (SPD) entries defined for Virtual Private Network (VPN) tunnels.

switches
Displays Multiple Electrical Interface (MEI) switch settings currently currently defined for ports, on devices supporting MEI. (See "set switches" on page 208.)

tcp
Displays active TCP sessions and active TCP listeners. To display more TCP-related statistics, such as number of input and output bytes transmitted, issue an “info tcp” command (see "info" on page 41).

udp
Displays current UDP listeners.

To display more UDP-related statistics, such as number of input and output bytes transmitted, issue an “info udp” command (see "info" on page 41).

uptime
Displays amount of time since the device was booted.

version
Displays boot, POST and EOS firmware version information and Digi part numbers for those items.

vpn
Displays all VPN-related status information, including Security Association (SA) database entries and Security Policy Database (SPD) entries. Applies to Digi Cellular Family products only.

wlan
Displays typical wireless LAN (WLAN) parameters for Digi Connect wireless devices.

Example

Display device information

```
#> display device

Device Information:

Product              : Digi Connect ME
MAC Address          : 00:40:9D:24:8B:B3
Firmware Version     : 1.9.0 (Version 82000856_F5 09/16/2005)
Boot Version         : 0.0.0.1 (release_82000866_C)
Post Version         : 0.0.0.1 (release_82000867_B)
CPU Utilization      : 14 %
Uptime               : 4 hours, 51 minutes, 38 seconds
Total Memory         : 8388608
Free Memory          : 2798316
Used Memory          : 5590364
```

Display Virtual Private Network (VPN) status information

```
#> display vpn

 SADB Table:

 Source IP AddresDestination IP AProtect Mode  SPI      Hash  Enc   TTL-sec   T
TL-kb



 SPD Table:

 Idx, Selector(local ip:port,remote ip:port, protocol),LEAST PREFERRED SPD
ENTRY
 FIRST Protect Mode  Hash  Enc   Protect Mode  Hash
```

Output

This section describe and interprets selected information and statistics output by the “display” command.

Information returned by “display pppstats”

The “display pppstats” displays status and activity information for a PPP link, and SureLink statistics.

This information is specific to Digi Cellular Family products.

PPP status and activity information

In these status and activity values, a “session” is a PPP session. The session statistics are reset to zero at the start of a new PPP link. The “total” statistics are the accumulated totals for all sessions since the device booted.

state
The current state of the PPP link. “Active” indicates that a PPP link is up. “Inactive” means the PPP link is down. Inactive is indicated when the link is coming up, or going down.

ip address
The PPP WAN IP address of the Digi device. This is the IP address used to communicate over the cellular network. This IP address is assigned by the carrier most of the time, but can also be given to the network by the Digi device.

primary dns addr
secondary dns addr
These are addresses for the DNS nameservers, used for performing name lookups. These DNS addresses are assigned by the carrier most of the time, but can also be assigned by users.

tx bytes
rx bytes
The total number of bytes transmitted (tx) or received (rx) over the PPP link since the last reboot.

session tx bytes
session rx bytes
The number of bytes transmitted (tx) or received (rx) over the PPP link in the current PPP session.

reset status

These PPP status values describe why a PPP link was terminated.

idle resets

The number of resets because the idle timeout was reached/exceeded for transmitted and received data. These idle timeouts are set by the “set pppoutbound” command. Most of the time, no idle timeout is used on transmitted data.

no carrier resets

The number resets because the carrier was dropped for any reason.

no service resets

The number of resets because the data network was not available.

Note: For Digi Cellular Family products, “no carrier” and “no service” resets indicate problems with your cellular service. “No service” resets could be cased by low signal strength. Review the signal strength and reposition the antenna or Digi device as needed. An external high-gain or directional antenna may be needed. These resets can also be due to roaming issue. To check the signal strength and current carrier settings, issue a “display mobile” command.

admin resets

The number of resets done for administrative purposes, such as issuing “kill” commands, or disconnecting a PPP session in the Web user interface’s Connections page by clicking the **Disconnect** button.

non-admin resets

The number of LCP termination requests made from the network; that is, the network notifies the Digi device that the PPP link is being brought down.

surelink resets

The number of resets caused by SureLink bringing down the PPP link and reestablishing it. SureLink performs three tests to monitor the integrity of the PPP link: ping, DNS, and TCP connection testing. The “set surelink” command has options for setting how these tests are performed. (see "set surelink" on page 202). If SureLink is unable to complete these tests, it concludes that the link is broken, and reestablishes the connection.

lcp keepalive resets

The number of resets caused by the LCP keepalive tests. LCP keepalive tests are similar to the SureLink link integrity monitoring tests, and perform the equivalent of a ping test for the PPP link. If the cellular network does not ping back after the number of replies specified by the number of consecutive missed replaces on the “set pppoutbound” command option “lcp_ka_max_missed_replies,” the LCP keepalive feature drops the link.

last reset reason
The reason for the most recent reset of the PPP link.

idle
An idle reset brought down the link last.

lcp keepalive
An LCP keepalive reset brought down the link last.

surelink
Surelink tests failed and brought down the link.

no service
The modem received a “no service” indication on the monitoring channel, and brought down the link.

no carrier
The modem dropped the link (hard), and was responsible for the termination of the last link.

administrative
Someone issued a “kill” command, or disconnected the PPP link in the Web user interface’s Connections page.

non-administrative
The network initiated closure of the last link.

unknown
The link was brought down for unknown reasons. This status is also displayed if the PPP connection has not been brought down since the Digi device was last rebooted. Since it is not possible for "last reset reason" information to persist across resets, this unknown state indicates that it is not clear which event may have been responsible for a reset that occurred in a prior life of the device.

tx timer
The time, in seconds, after which if no data is transmitted, the PPP link is disconnected. Typically, this value is 0 (disabled).

rx timer
The time, in seconds, after which if no data is received, the PPP link is disconnected. An idle reset ends the PPP session and reestablishes it, to prevent the carrier network from dropping an inactive call. The default is 1440 seconds (24 minutes). This value is also known as the “Inactivity timeout” in the Web user interface’s “Mobile Settings.”

session time
The duration of the current PPP session. To display total system uptime, issue a “display uptime” command.

rx idle time
tx idle time
The amount of time since data was last received (rx)/or transmitted (tx) by the Digi device.

lcp echo requests
The number of Link Control Protocol (LCP) echo requests that have been sent after a “quiet” interval, in order to test the PPP link and/or keep it alive. For Digi Cellular products, LCP echo requests are typically not used.

SureLink Statistics

Digi SureLink™ provides an “always-on” mobile network connection to ensure that a Digi Cellular Family device is in a state where it can connect to the network.

The statistics displayed for Digi SureLink pertain to the periodic tests, known as Link Integrity Monitoring tests, that are run over the established PPP connection to ensure that end-to-end communication is possible. There are three Link Integrity Monitoring tests available: There are three tests available: Ping Test, TCP Connection Test, and DNS Lookup Test. For descriptions of these tests, see "set surelink" on page 202.

In these SureLink statistics, a “session” is a PPP session. The session statistics are reset to zero at the start of a new PPP connection. The “total” statistics are the accumulated totals for all sessions since the device booted. The “tests” are the SureLink Link Integrity Monitoring tests that have been configured to be run when the mobile network connection is established.

session successes
The number of times a configured test was attempted and succeeded in a PPP session.

session failures
The number of times a configured test was attempted but failed in a PPP session.

session consecutive failures
The number of consecutive failures for a test, with no success. When a test is successful, the consecutive failures counter is reset to zero. The consecutive failures counter indicates a device's “progress” toward the configured maximum number of consecutive failures, after which the PPP link is taken down (and restarted).

session bypasses
If a configuration parameter is bad, a test is bypassed rather than considered to have succeeded or failed. This means the test was not run. If the PPP connection goes down while a test is in progress, that test may be classified as bypassed, since it could not be run. (Note that the PPP link may come down for many reasons, independent of SureLink testing.)

total successes
The total number of times a configured test was attempted and succeeded since the Digi device was booted.

total failures
The total number of times a configured test was attempted but failed since the Digi device was booted.

total link down requests
The number of times the SureLink feature has failed consecutively the configured number of failures and, as a result, requested that PPP shut down and restart its connection. This statistic counts such occurrences during the current device boot. SureLink itself does do the PPP stop/start; it sends a message to PPP asking it to do so, owing to a Surelink test failure.

total bypasses
The total test bypasses (see “session bypasses”) since the Digi device was rebooted.

See also

- "info" on page 41.
- "show" on page 249 and "set wlan" on page 241. The "show wlan" command displays additional wireless LAN information, including wireless LAN settings configured by "set wlan" and evaluations of the settings.
- "status" on page 254.
- "set vpn" on page 228 for information on the settings related to the “sadb,” “spd,” and “vpn” options.

display buffers

Devices supported

This command is supported in all Digi Connect products except Digi Connect WAN and ConnectPort Display.

Purpose

Displays the contents of a port buffer, or transfers the contents of a port buffer to a server running Trivial File Transfer Protocol (TFTP). Port buffering is enabled by the “set buffer” command (see "set buffer" on page 90). Contents are displayed in log form.

Required permissions

For Digi Connect products with two or more users, permissions must be set to one of the following:

- For a user to display the contents of a port buffer for the line on which they are logged in: “set permissions buffers=r-self” or higher.
- For a user to display the contents of a port buffer for any line: “set permissions buffers=read” or higher.

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

```
display buffers [port=range] {[screen] [lines=number]
   [tail=number] | tftp=server:filename}
```

Options

port=*range*
The port or ports to which the command applies. Optional on a single-port device.

screen
Displays the port buffer contents on the screen when screen is specified.

lines=*number*
The number of lines of data to display at a time when the “screen” option is specified. Use 0 to indicate continuous flow.

tail=*number*
The total number of lines in the buffer to be displayed. The number is calculated from the end of the buffer counting back.

tftp=*server*:*filename*

server
The IP address or DNS name of a server running TFTP to which buffer information should be transferred.

filename
The name to use for the file that will be transferred to the TFTP server. If the “port” option specifies more than one port, one file will be transferred for each port. The filename for each port will be *filename_n*, where n is the port number.

Examples

Display port buffering information on the screen

```
#> display buffers port=2 screen lines=32 tail=30
```

Output buffering information to a TFTP server

```
#> display buffers port=2 tftp=192.168.1.1:port_ouput
```

Output multi-port buffering information to a TFTP server

```
#> display buffers port=2-3 tftp=192.168.1.1:port_ouput
```

Note that port 2 buffering information goes to file port_output_2 and port 3 buffering information goes to file port_output_3.

See also

- "set buffer" on page 90.

exit

Devices supported This command is supported in all Digi Connect products.

Purpose Terminates your current session.

Syntax

```
exit
```

Example

```
#> exit
```

See also "quit" on page 59. The “quit” and “exit” commands perform the same operation.

help and ?

Devices supported

This command is supported in all Digi Connect products.

Purpose

Displays help about a specific command.

Syntax

```
help [command]
```

OR

```
[command]?
```

Examples

```
#> help boot
#> boot?
#> help set serial
#> set serial?
```

See also

"Displaying Online Help" on page 12.

info

Devices supported

This command is supported in all Digi Connect products.

Purpose

Displays statistical information about a device.

The “info” command displays statistical information about a device over time. In contrast, the “display” command’s focus is on real-time information, while the “status” command displays the status of outgoing connections (connections made by “connect,” “rlogin,” or “telnet” commands).

Command options allow display of the following categories of statistics:

- Device statistics
- Ethernet statistics
- ICMP statistics
- IP statistics
- Mesh network statistics
- Serial statistics
- TCP statistics
- UDP statistics
- WLAN statistics (for wireless devices only)

The statistics in these tables are those gathered since the tables were last cleared. The statistics tables are cleared by rebooting the Digi device.

Status and statistics for mobile communications

To display information and statistics for mobile/cellular communications, issue a “display mobile” command. To display SureLink statistics, issue a “display pppstats” command. See "display" on page 27.

Syntax

```
info {device|ethernet|icmp|ip|mesh|serial|tcp|udp|wlan}
```

Options

For a description of the statistics displayed by all these options, see “Results” on the following page.

device
Displays statistics from the device table. This information includes device-model information, MAC address, current Boot and POST code, firmware, memory usage, utilization, and uptime. The information displayed by this option is the same as that displayed by the “display device” command (see "display" on page 27).

ethernet
Displays statistics from the Ethernet table.

icmp
Displays statistics from the ICMP table.

ip
Displays statistics from the IP table.

serial
Displays statistics from the serial table. For descriptions of these statistics, see "Output" on page 42.

tcp
Displays statistics from the TCP table.

udp
Displays statistics from the UDP table.

wlan
Displays statistics from the wireless Ethernet (wlan) table.

Output

Following are descriptions of the statistics displayed for each “info” command option.

The statistics displayed include data, event, and error counts. These statistics are useful in understanding how the device is operating and can be helpful in finding problems. In particular if an error counter is found to be increasing you may have a problem with the device.

To reset the statistics, reboot the device.

Device statistics

Device Information	Description
Product	The model of the Digi Connect device.
MAC Address	A unique network identifier. All network devices are required to have their own unique MAC address. The MAC address is on a sticker on your Digi Connect device. The number is displayed as 12 hexadecimal digits, usually starting with 00:40:9D.
Firmware Version	The current firmware version. This information may be used to help locate and download new firmware. Firmware updates may be downloaded from the Digi Support website.
Boot Version	The current boot version.
Post Version	The current POST version.
CPU Utilization	The amount of CPU resources being used by the Digi Connect device.
Uptime	The amount of time the Digi Connect device has been running since it was last powered on or rebooted.
Total Memory	The total amount of memory (RAM) available.
Free Memory	The amount of memory (RAM) currently not being used.
Used Memory	The amount of memory (RAM) currently in use.

Ethernet statistics

Statistic	Description
InBytes	Number of bytes received.
OutBytes	Number of bytes sent.
InUcastPkts	Number of Unicast packets received.
OutUcastPkts	Number of Unicast packets sent.
InNonUcastPkts	Number of non-Unicast packets received.
OutNonUcastPkts	Number of non-Unicast packets sent.
InDiscards	Number of incoming packets that were discarded.
OutDiscards	Number of outgoing packets that were discarded.
InErrors	Number of incoming packets that contained errors.
OutErrors	Number of outgoing packets that contained errors.
RxOveruns	Number of Rx overruns. Rx overruns are generally caused by the inability of the device to get sufficient bus bandwidth to offload the data.
TxResets	Number of times the transmitter has been reset.
InUnknownProtos	Number of incoming packets where the protocol was unknown.

ICMP statistics

Statistic	Description
InMessages	Number of incoming messages.
OutMessages	Number of outgoing messages.
InDestUnreachables	Number of incoming destination-unreachable messages received. A destination-unreachable message is sent to the originator when a datagram fails to reach its intended destination.
OutDestUnreachables	Number of destination-unreachable messages sent. A destination-unreachable message is sent to the originator when a datagram fails to reach its intended destination.
InErrors	Number of incoming received messages with errors.

IP statistics

Statistic	Description
InReceives	Number of datagrams received.
OutRequests	Number of datagrams given to IP to transmit.
InAddressErrors	Number of received datagrams discarded because they were for another host and could not be forwarded.
DatagramsForwarded	Number of received datagrams forwarded to another host.
InHeaderErrors	Number of received datagrams discarded because of invalid header information.
OutNoRoutes	Number of received datagrams discarded because no route to the destination IP address could be found.
InUnknownProtos	Number of received datagrams discarded because the specified protocol is not available.
OutDiscards	Number of outgoing datagrams that were discarded for miscellaneous reasons. This statistic is not used and is always zero.
InDiscards	Number of received datagrams discarded for miscellaneous reasons.
FragCreates	Number of outgoing datagram fragments created.
ReassembleOks	Number of received datagrams that were successfully reassembled from fragments.
FragOks	Number of outgoing datagrams that were fragmented.
FragFails	Number of outgoing datagram fragmentation attempts that failed. This statistic is not used and is always zero.
AclExamines	Number of received datagrams examined for access control filtering.
AclAccepts	Number of received datagrams accepted after being examined by access control filtering.
AclDiscards	Number of received datagrams discarded after being examined by access control filtering.
NatPrivateToPublic	Number of datagrams received from the private network, successfully translated by NAT, and returned to IP to be forwarded to the public network.
NatPublicToPrivate	Number of datagrams received from the public network, successfully translated by NAT, and returned to IP to be forwarded to the private network.

Serial statistics

Statistic	Description
rbytes	Total data in: the number of bytes received.
tbytes	Total data out: the number of bytes transmitted.
overrun errors	The number of times FIFO has overrun. The next data character arrived before the hardware could move the previous character.
overflow errors	The number of times the Received buffer has overrun. The receive buffer was full when additional data was received.
frame errors	The number of framing errors detected. The received data did not have a valid stop bit.
parity errors	The number of parity errors detected. The received data did not have the correct parity setting
breaks	The number of break signals detected.
signal change	For each signal (CTS, DSR, RI, DCD, RTS, DTR), the number of times the signal has changed states.

TCP statistics

Statistic	Description
InSegments	Number of segments received.
OutSegments	Number of segments sent.
InErrors	Number of segments received with errors.
RetransmitSegments	Number of segments retransmitted. Segments are retransmitted when the server doesn't respond to a packet sent by the client. This is to handle packets that might get lost or discarded somewhere in the network.
EstabResets	Number of established connections that have been reset.
OutResets	Number of outgoing connections that have been reset.
PassiveOpens	Number of passive opens. In a passive open, the Digi device server is listening for a connection request from a client.
ActiveOpens	Number of active opens. In an active open, the Digi device server is initiating a connection request with a server.
Established	Number of established connections.
Attempt Fails	Number of failed connection attempts.

UDP statistics

Statistic	Description
InDatagrams	Number of datagrams received.
OutDatagrams	Number of datagrams sent.
InErrors	Number of bad datagrams that were received. This number does not include the value contained by "No Ports"
NoPorts	Number of received datagrams that were discarded because the specified port was invalid.

Wireless (WLAN) statistics

The WLAN statistics may aid in troubleshooting network communication problems with your wireless network.

For additional wireless settings and an evaluation of the wireless settings, issue a "show wlan" command. See "show" on page 249.

Statistic	Description
TxFrames	Number of frames transmitted.
TxBroadcastFrames	Number of broadcast frames transmitted.
TxRtsFrames	Number of Request-to-Send (RTS) frames transmitted.
TxRetries	Number of times an outgoing frame is retransmitted because the acknowledgement for the frame was not received.
TxDroppedRetries	Number of outgoing frames that were dropped because the maximum number of retries were exceeded for the frame.
TxDroppedBroadcasts	Number of broadcast frames dropped because the acknowledgement for the frame was not received.
TxDroppedAssoc	Number of outgoing packets dropped because the device had not yet associated with a wireless network
RxFrames	Number of received frames.
RxBroadcastFrames	Number of received broadcast frames.
RxRtsFrames	Number of RTS frames received.
RxRetries	Number of incoming frames that have the retry bit set in their frame header. The retry bit indicates that the other side has attempted to transmit a given frame more than once.
RxDroppedNoBuffers	Number of received frames dropped due to no buffer.
RxDropInvalid	Number of incoming frames dropped because the frame appeared incorrect.
RxDropDuplicate	Number of incoming frames dropped because a given frame had already been received.
RxDropAge	Number of fragmented frames dropped because the fragment timed out before the rest of the frame sequence was received.
RxDropDecrypt	Number of frames dropped because they were not properly encrypted.
RxDropSize	Number of frames dropped because their frame size was too big

Examples

Display ICMP statistics

```
#> info icmp

ICMP statistics:

InMessages           : 14             OutMessages          : 0
InDestUnreachables   : 5              OutDestUnreachables  : 0
InErrors             : 0
```

See also

- "display" on page 27.
- "show" on page 249. The "show wlan" command displays settings and evaluation information that may be useful to view in addition to the wireless statistics displayed by this command.
- "status" on page 254

kill

Devices supported

This command is supported in all Digi Connect products.

Purpose

Use the kill command to kill connections. The kill command is associated with the connections displayed by the “who” command.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions kill=execute” to use this command. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

```
kill [range] [connection id]
```

Options

range
A range of connection IDs.

connection id
An ID for the connection.

Examples

Killing a session on a specific port

```
#> kill 1
```

Killing a session on a range of ports

```
#> kill 1-3
```

See also

- "close" on page 21, to close sessions created from the current connection.
- "status" on page 254, to display the list of current sessions.
- "who" on page 258, for information on determining active connections.

mode

Devices supported

This command is supported in Digi Connect Family products only and not in Digi Cellular Family products or ConnectPort Display.

Purpose

Changes or displays the operating options for a current Telnet session.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions telnet=execute” to display or set Telnet operating options. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Change Telnet options

```
mode [options]
```

Display Telnet options

```
mode
```

Options

options

The operating options for a current Telnet session, which are as follows:

binary={on|off}

Enables or disables Telnet binary mode is enabled or disabled. “binary=on” turns on binary mode, which means that all transmitted and received characters are converted to binary during this Telnet session. “binary=off” turns off binary mode off for this Telnet session. The default is off.

crmod={on|off}

Specifies whether line feeds are added to received carriage returns. “crmod=on” specifies that line feeds are added to received carriage returns. “crmod=off” specifies that line feeds are **not** added to received carriage returns. The default is off.

Examples

Turn on binary mode

```
#> mode binary=on
```

Add line feed characters

```
#> mode crmod=on
```

Display operating options

```
#> mode
```

See also

"telnet" on page 255.

newpass

Devices supported

This command is supported in all Digi Connect products.

Purpose

Creates or changes user passwords for the device.

In Digi devices with a single-user model, changing the “root” user password also changes the password for Advanced Digi Discovery Protocol (ADDP). In Digi devices with the multi-user model, changing the “root” password has no effect on ADDP. To change the ADDP password, you would specify “name=addp.”

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions newpass=rw-self” for a user to set their own password, and “set permissions newpass=rw” to set another user’s password. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

`newpass [id=`*`number`*`|name=`*`string`*`]`

Options

id=*number*
Specifies the ID of the user to be acted on.

name=*string*
Specifies the name of the user to be acted on.

Example

The “newpass” command initiates a dialog that changes the user’s password.

User changing their own password

```
#> newpass
```

Changing another user’s password

```
#> newpass name=jdoe
```

See also

- "User Models and User Permissions in Digi Connect Products" on page 14.
- "set user" on page 220 for information on configuring users.

ping

Devices supported

This command is supported in Connect Family products only and not in Digi Cellular Family products or ConnectPort Display.

Purpose

Tests whether a host or other device is active and reachable.

To interrupt the “ping” command, use Ctrl-C.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions ping=execute” for a user to use this command. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

```
ping ipaddress [options]
```

Options

ipaddress

Identifies the target of the “ping” command by its IP address.

options

The options associated with the “ping” command, which are:

count=0|*n*

The number of “ping” commands to be issued. 0 means ping until interrupted. The default is 0.

interval=*milliseconds*

The ping time in milliseconds. The default is 1000 milliseconds.

size=*bytes*

The number of bytes to send in each ping packet. The default is 56 bytes.

Examples

Specify a simple ping

The following command determines whether the specified host can be reached:

```
#> ping 199.150.150.10
```

provision

Devices supported

This command is supported in Digi Cellular Family products that have a CDMA (Code-Division Multiple Access) module.

Purpose

Provisions the CDMA module in a Digi Cellular Family device. Provisioning establishes configuration settings in the CDMA module for use in a mobile network. Examples of CDMA-based mobile service providers include Sprint, Verizon, Alltel, and Midwest. The CDMA module must be provisioned before you will be able to create a data connection to the mobile network.

Provisioning needs to be performed once only. It is not necessary for Digi Cellular Family devices that use GSM (Global System for Mobile Communication).

Provisioning is done either from the command line, using this command, or from the Web user interface, by launching the Mobile Device Provisioning Wizard from the Mobile Configuration page.

Provisioning types

There are several types of provisioning, each with different sets of parameters. Your mobile service provider can tell you which type is appropriate for your CDMA module.

- Simple IP only (SIPONLY).
- Mobile IP (MIP), which is a super set of SIPONLY.
- IP-Based Over-the-Air (IOTA).
- OTASP: Over-the-Air Service Provisioning (OTASP).

Check with your mobile service provider for provisioning parameters

The information that you need to specify during provisioning depends on your CDMA module and the settings that your mobile service provider has given you or already set up in your CDMA module.

Contact your mobile service provider for the most appropriate provisioning type and the required provisioning parameters. Have the ESN (Electronic Serial Number) for your Digi Cellular device ready to give to the provider. This number is located on the label on the bottom of the device.

Use “display provisioning” to get current provisioning parameters

You can query for the currently configured provisioning parameters in the CDMA cellular module by entering a “display provisioning” command.

Important: Close PPP sessions before issuing provisioning commands

The “provision” and “display provisioning” commands cannot be used while Point-to-Point Protocol (PPP) sessions are active.

To close any existing PPP sessions:

1. Disable the PPP interface by entering a “set pppoutbound” command with these options:

   `#> set pppoutbound port=`*`interface port number`* `state=disabled`

2. Next, identify the ID of the connection, by issuing a “who” command.
3. Once the session is identified, issue a “kill” command to end the PPP session.
4. After provisioning, you can enable the PPP interface again by entering another “set pppoutbound” command:

   `#> set ppp port=`*`interface port number`* `state=enabled.`

Syntax

Parameters may be required, optional, or preset and not to be changed depending on your mobile service provider and the information they have given you.

Some mobile services providers do not accept dashes in phone numbers. Enter phone numbers as numbers only with no dashes.

Display current provisioning parameters

```
display provisioning
```

Manually provision the module for a SIP-only network

```
provision type=siponly
   spc=service programming code (also known as master subsidy lock
   or MSL)
   mdn=mobile directory number
   min=mobile ID number
```

Manually provision the module for a MIP network

```
provision type=mip
   spc=service programming code (also known as master subsidy lock
   or MSL)
   mdn=mobile directory number
   min=mobile ID number
   nai=network access id
   aaass=AAA shared secret
   aaasstype={ascii|hex} default=ascii
   ha=home address
   priha=primary host agent IP address
   secha=secondary host agent IP address
   hass=host agent shared secret
   hasstype={ascii|hex} default=ascii
   haspi=index
   aaaspi=index
   rtun={0|1}
   profile=MIP profile number
```

Use IOTA to provision the module

```
provision type=iota
   spc=service programming code (MSL)(also known as master subsidy
   lock or MSL)
   mdn=mobile directory number
   min=mobile ID number
```

Use OTASP to provision the module

```
provision type=otasp
   otaspnumber=OTASP number - for example, *228 or *22899
```

Options

SIP-only provisioning parameters

type=siponly
Specifies that the CDMA module is being provisioned using the SIPONLY method.

spc=*service programming code* (MSL)
A six-digit number required to program CDMA module parameters. This code is also known as a master subsidy lock or MSL.

mdn=*mobile directory number*
The phone number of the CDMA module.

min=*mobile ID number*
How the CDMA module is identified in the cellular network. Depending on the cellular provider, this number may be the same as the mobile directory number.

MIP provisioning parameters

type=mip
Specifies that the CDMA module is being provisioned using the MIP method.

nai=*network access id*
Internet Authentication, Authorization and Accounting (AAA) protocols such as RADIUS or DIAMETER identify users with the Network Access Identifier (NAI). When used with Mobile IP and AAA, the NAI is composed of a username and a realm, separated with “@”. The username portion identifies the subscriber within the realm. The AAA nodes use the realm portion of the NAI to route AAA requests to the correct AAA server.

aaass=*AAA shared secret*
The shared secret used in authentication by Internet AAA protocols such as RADIUS or DIAMETER.

The format of the shared secret differs depending on whether it is entered in ASCII or hexadecimal, as specified by the “aaasstype” option.

If “aaasstype=ascii,” enter the shared secret as a string in quotation marks.

If “aaasstype=hex,” enter the shared secret as a hexadecimal number with no leading “0x” or trailing “h.”

aaasstype={ascii|hex} default=ascii
Specifies whether the AAA shared secret is specified in ASCII or hexadecimal form. This option affects how the shared-secret values are specified on the “aaass” option.

ha=*home address*
The home address for the CDMA module, specified as an IP address

priha=*primary host agent IP address*
The IP address of the primary host agent that provides mobile service for the CDMA module.

secha=*secondary host agent IP address*
The IP address of the secondary host agent that provides mobile service for the CDMA module.

hass=*host agent shared secret*
The shared secret used for authentication for the host agent.

The format of the shared secret differs depending on whether it is entered in ASCII or hexadecimal, as specified by the “hasstype” option.

If “hasstype=ascii,” enter the shared secret as a string in quotation marks.

If “hasstype=hex,” enter the shared secret as a hexadecimal number with no leading “0x” or trailing “h.”

hasstype={ascii|hex} default=ascii
Specifies whether the host agent shared secret is specified in ASCII or hexadecimal form. This option affects how the shared-secret values are specified on the “hass” option.

haspi=*index*
aaaspi=*index*
A Security Parameter Index (SPI) is an index identifying a security context between a pair of routers among the contexts available in the mobility security association. These are index options that set the security context between the host agent and AAA server.

rtun= {0|1}
Enables or disables use of reverse tunnelling.

profile=*MIP profile number*
Specifies which of several profiles, or configuration scenarios, that the cellular module will use when communicating with the cellular network. This is a numeric value; the values available depend on your cellular provider.

IOTA provisioning parameters

type=iota
Specifies that the CDMA module is being provisioned using the IOTA method.

spc=*service programming code* (MSL)
A six-digit number required to program CDMA module parameters.

mdn=*mobile directory number*
The phone number of the CDMA module.

min=*mobile ID number*
How the CDMA module is identified in the cellular network. Depending on the cellular provider, this number may be the same as the mobile directory number.

OTASP provisioning parameters

type=otasp
Specifies that the CDMA module is being provisioned using the OTASP method.

otaspnumber=*OTASP number* for example, *228
A phone number for initiating an OTASP provisioning session. This number typically begins with *228, for example *22899.

See also

- "display" on page 27. The “display provisioning” command displays the currently configured parameters in the CDMA cellular module.
- The *Digi Cellular Family User’s Guide’s* section on Mobile Device Provisioning. That discussion describes provisioning through a Wizard and the Web user interface, but the same concepts apply to command-line based provisioning.
- "set pppoutbound" on page 166
- "who" on page 258
- "kill" on page 49

python

Devices supported

This command is supported in Connectport X Family products only.

Purpose

Manually executes a Python program from the command line.

The “python” command is similar to a command executed on a PC. However, other than a program name and arguments for the program, the command takes no arguments itself, and is currently unable to spawn an interactive session.

Syntax

```
python [(TFTP server ip):]filename [program args...]
```

Options

[(TFTP server ip):]filename

The main file to be executed. This file can be either a file on the file system accessed through the Web UI, or a file accessible through a TFTP server on the network. This TFTP functionality reduces the number of times that you may need to place a program on the file system while developing and refining functionality. However, the TFTP behavior only works for the main program. Modules and packages must still be present on the file system to be used.

program args...

Arguments to be supplied to the program.

See also

- The *Digi Python Programming Guide* to learn more about the Python programming language as implemented in Digi products, and writing Python programs.
- "set python" on page 183 to manually execute a Python program.

quit

Devices supported This command is supported in all Digi Connect products.

Purpose Use the quit command to log out of the device.

Syntax

```
quit
```

Example

```
#> quit
```

See also "exit" on page 39. The “quit” and “exit” commands perform the same operation.

reconnect

Devices supported

This command is supported in all Digi Connect products.

Purpose

Reestablishes a previously established connection; that is, a connection opened by a “connect,” “rlogin,” or “telnet” command. The default operation of this command is to reconnect to the last active session.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions reconnect=execute” to use this command. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

```
reconnect [{serial port|p=serial port|s=session}]
```

Options

serial port

The serial port to which this command applies. Use this option to reconnect to a session opened by a connect command.

p=*serial port* | s=*session*

The serial port number or session number (displayed by the “status” command) to reconnect to.

Example

Reconnect to the last port used

```
#> reconnect
```

Reconnect to port 1

```
#> reconnect p=1
```

Reconnect to session 1

```
#> reconnect s=1
```

See also

- "connect" on page 22 for information on establishing a connection on a selected port.
- "close" on page 21 for information on ending a connection.
- "status" on page 254 for information on gathering status on current connections.
- "rlogin" on page 66.
- "telnet" on page 255.

revert

Devices supported

This command is supported in all Digi Connect products.

Purpose

Sets a particular group of a devices' settings to its default values.

The "revert" command keywords are used one at a time to revert one group of settings. You cannot enter several keywords on a single command to revert multiple settings.

If you enter "revert user," "revert group," or "revert permissions," a message is displayed indicating that those settings cannot be reverted individually, and instead must be reverted all together at the same time via the "revert auth" command. The "revert auth" command (revert authentication and authorization) reverts all users, all groups, and all permissions at the same time.

Required permissions

No "set permissions" option is required for all "revert" command variants except "revert all." The permissions used by the various "set" commands apply to the various "revert" command variants. "revert all" uses a different mechanism that bypasses the individual "set" commands, and therefore has its own permissions. To execute the "revert all" command, a user must have permissions set to "set permissions revert-all=execute". See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

```
revert [all|
  accesscontrol|
  alarm|
  auth|
  autoconnect [port=range]|
  bsc|
  buffer [port=range]|

  ddns|
  dhcpserver|
  ekahau
  forwarding|
  gpio|
  host|
  ia|
  idle|
  login|
  menu|
  mesh|
  mgmtconnection|
  mgmtglobal|
  mgmtnetwork|
  nat|
  network|
  passthrough|
  pmodem [port=range]|
  pppoutbound [port=range]|
  profile [port=range]|
```

```
putty|
python|
rciserial|
realport|
rtstoggle|
serial [port=range]|
service|
snmp|
socket_tunnel|
surelink|
switches|
system|
tcpserial [port=range]|
term [port=range]|
udpserial [port=range]|
user|
video|
vncclient|
vpn {all|global|tunnel|phase1|phase2}|
wireless]
```

Options

all
Reverts all settings *except* these*:*

- Network settings
- Security settings (passwords and suppress login)
- Host key settings.

accesscontrol
Reverts the access control settings configured by the “set accesscontrol” command.

alarm
Reverts the alarm settings configured by the “set alarm” command.

auth
Reverts the permission settings configured by the “set permissions” command, the user settings configured by the “set user” command, and group settings, configured by the “set group” command.

autoconnect [port=*range*]
Reverts the Autoconnect settings configured by the “set autoconnect” command.

bsc
Reverts the settings configured by the “set bsc” command.

buffer [port=*range*]
Reverts the port-buffering settings configured by the “set buffer” command.

ddns
Reverts the Dynamic DNS (DDNS) settings configured by the “set ddns” command.

dhcpserver
Reverts the DHCP server settings configured by the “set dhcpserver”

command.

ekahau
Reverts the Ekahau client settings configured by the “set ekahau” command. See "set ekahau" on page 104.

forwarding
Reverts the port-forwarding settings configured by the “set forwarding” command.

gpio
Reverts the GPIO settings configured by the “set gpio” command.

host
Reverts the host name set by the “set host” command.

ia
Reverts the Industrial Automation (IA) settings configured by the “set ia” command.

idle
Reverts the settings configured by the “set idle” command.

menu
Reverts the custom menu settings configured by the “set menu” command.

mesh
Reverts the Mesh network settings configured by the “set mesh” command.

login
Reverts the login settings configured by the “set login” command

mgmtconnection
Reverts the Connectware Manager connection settings configured by the “set mgmtconnection” command.

mgmtglobal
Reverts the Connectware Manager global settings configured by the “set mgmtglobal” command.

mgmtnetwork
Reverts the Connectware Manager network settings configured by the “set mgmtnetwork” command.

nat
Reverts the Network Address Translation (NAT) and port/protocol forwarding settings configured by the “set nat” command.

network
Reverts the network settings, configured by the “set network” command, and the wireless configuration settings, configured by the “set wlan” command.

passthrough
Reverts the IP pass-through settings configured by the “set passthrough” command.

pmodem [port=*range*]
Reverts the modem emulation settings, configured by the “set pmodem” command.

pppoutbound [port=*range*]
Reverts the Point-to-Point Protocol (PPP) outbound connection settings, configured by the “set pppoutbound” command.

profile [port=*range*]
Reverts the profile settings configured by the “set profile” command.

putty
Reverts the terminal emulation settings configured by the “set putty” command.

python
Reverts the Python program settings configured by the "set_python" command.

rciserial
Reverts the RCI serial settings configured by the “set rciserial” command.

realport
Reverts the Realport settings configured by the “set realport” command.

rtstoggle
Reverts the RTS toggle settings configured by the “set rtstoggle” command.

serial [port=*range*]
Reverts the serial settings configured by the “set serial” command.

service
Reverts the service settings configured by the “set service” command.

snmp
Reverts the SNMP settings configured by the “set snmp” command.

socket_tunnel
Reverts the socket tunnel settings configured by the “set socket_tunnel” command.

surelink
Reverts the Digi SureLink™ settings configured by the “set surelink” command.

switches
Reverts the Multiple Electrical Interface (MEI) switch settings configured by the "set switches" command.

system
Reverts the system settings configured by the “set system” command.

tcpserial [port=*range*]
Reverts the TCP serial settings configured by the “set tcpserial” command.

term [port=*range*]
Reverts the terminal connection settings configured by the “set term”

command.

udpserial [port=*range*]
Reverts the UDP serial settings configured by the “set udpserial” command.

user
Reverts the user settings configured by the “set user” command.

video
Reverts the video settings configured by the “set video” command.

vncclient [port=*range*]
Reverts the settings configured by the “set vncclient” command.

vpn {all|global|tunnel|phase1|phase2}
Reverts the Virtual Private Network (VPN) settings configured by the “set vpn” command. Keyword options allow for reverting all or selected VPN settings. See "set vpn" on page 228 for descriptions of the settings.

all
Reverts all VPN settings.

global
Reverts global VPN options.

tunnel
Reverts VPN tunnel settings.

phase1
Reverts Internet Key Exchange (IKE)/Internet Security Association and Key Management Protocol (ISAKMP) Security Association (SA) Phase 1 options.

phase2
Reverts IKE/ISAKMP SA Phase 2 options.

wireless
Reverts the wireless settings configured by the “set wlan” command.

Example

Reset a device’s serial setting

The device serial setting is reset to the default serial configuration.

```
#> revert serial
```

Reset a serial port to default settings

```
#> revert serial port=2
```

See also

- "boot" on page 19.
- The various “set” commands referenced in this description.
- "show" on page 249.

rlogin

Devices supported

This command is supported in Digi Connect Family and Digi Cellular Family products only. Not supported in ConnectPort Display.

Purpose

Performs a login to a remote system, also referred to as an rlogin.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions rlogin=execute” to use this command. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

```
rlogin [esc=(char)] [{user=user name|-l user name}]
   [ip address]
```

Options

esc

A different escape character than the ~ (tilde) character, which will be used for the current Rlogin session. This character is used for suspending a session from the remote host to return to the device server command line.

user=*user name* | -l *user name*

The user name to use on the remote system. If you do not specify a name, your device server user name will be used. The “-l *user-name*” option is for compatibility with the UNIX “rlogin” command.

ip address

The IP address of the system to which you are performing the remote login.

Examples

```
#> rlogin 10.0.0.1
```

See also

- "telnet" on page 255.
- "connect" on page 22.
- "status" on page 254.
- "close" on page 21.

send

Devices supported

This command is supported in Digi Connect Family and Digi Cellular Family products only. Not supported in ConnectPort Display.

Purpose

Sends a Telnet control command, or special-character sequences, when connected using the Telnet client.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions telnet=execute” to display or set Telnet operating options. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

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

Options

ao
Sends the “abort output” signal to discard 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 operation” signal to the peer.

synch
Sends the “synchronize process” signal to the peer.

Examples

Send an “interrupt process” signal

```
#> send ip
```

Send an “are you there” signal

```
#> send ayt
```

See also

See "telnet" on page 255 for information on establishing Telnet sessions.

set accesscontrol

Devices supported

This command is supported in Digi Cellular Family products only.

Purpose

Used to specify information that limits network access to this device, or display current access-control settings. For the Digi Connect WAN, the access-control settings also limit routing of packets through the device.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display the access control settings: “set permissions s-accesscontrol=read”
- For a user to display and set access control settings: “set permissions s-accesscontrol=rw”

Syntax

Configure access control settings

```
set accesscontrol [enabled={on|off}] [autoaddsubnets={on|off}]
   [addrip[1-64]=ipaddress] [subnip[1-32]=ipaddress]
   [subnmask[1-32]=mask]
```

Display current access-control settings

```
set accesscontrol
```

Options

enabled={on|off}

Used to enable access control. Care must be used with this command because improper settings can render this device inaccessible from the network. Specifically, setting this option to “on” with no “addrip” option values specified will disable all access.

on

Enables access control.

off

Disables access control.

autoaddsubnets={on|off}

Used to enable the automatic adding of subnets and subnet masks to this table. The IP subnets for the device server's network interfaces (Ethernet and PPP), may be automatically added to the table. This permits access by all IP sources on the device server's networks, without having to explicitly identify either the subnet IP addresses (and netmasks) or individual IP addresses.

on

Enables automatic adding of subnets and subnet masks.

off

Disables automatic adding of subnets and subnet masks.

addrip[1-64]=*ipaddress*

Used to specify up to 64 individual IP addresses that are allowed to access this device.

subnip[1-32]=*ipaddress*
Used to specify up to 32 subnet IP addresses. Any IP address in these subnets will be allowed to access this device server.

subnmask[1-32]=*mask*
Used to specify a subnet mask associated with one of the 32 subnet IP addresses.

Examples

Set access control settings

```
#> set accesscontrol enabled=on addrip1=143.191.1.228
```

Set access control for a specific subnet

This command will allow any IP address in the 143.191.2.0 subnet (netmask 255.255.255.0) to access this device server:

```
#> set accesscontrol enabled=on subnip1=143.191.2.0 subnmask1=255.255.255.0
```

Display access control settings

```
#> set accesscontrol
```

See also

- "revert" on page 61.
- "show" on page 249.

set alarm

Devices supported

This command is supported in the following products:

- Connect Family: Digi Connect EM, Digi Connect Wi-EM, Digi Connect ME, Digi Connect Wi-ME. Setting alarms in GPIO mode is not supported in the Digi Connect SP device.
- All Digi Cellular Family products. Setting alarms in GPIO mode is not supported.
- Not supported in ConnectPort Display.

Purpose

Configures device alarms and display current alarm settings. Device alarms are used to send emails or SNMP traps when certain device events occur. These events include changes in GPIO pin states; data patterns detected in the serial stream; and, for Digi Cellular Family products, the average signal strength falling below a specified level for a specified amount of time, and the amount of cellular traffic for a specified period of time, mobile temperature exceeding certain thresholds, and configuration changes to settings associated with the mobile device.

For Digi devices managed by Connectware Manager, the “cwm” option sends all alarms to a Connectware Manager server.

Up to 32 alarms can be configured in Digi Connect products.

To avoid false errors, configure alarms while alarms are disabled, by entering a “set alarm state=off” command, then enable alarms after they are fully configured by entering “set alarm state=on”.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions s-alarm=read” to display current alarm settings, and to “set permissions s-alarm=rw” to display alarm settings and configure alarms. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Configure alarms with general options (applies to all alarms)

```
set alarm [state={on|off}]
   [mailserverip=ipaddress]
   [from=string]
   [cwm={on|off}]
```

Configure alarms for a range (set multiple alarms)

```
set alarm range={1-32}
   [active={on|off}|to=string|cc=string|subject=string|
   priority={normal|high}|
   mode={match|gpio|rssi_gsm|cell_data|mobile_temp|
   rssi_1xrtt| rssi_1xevdo|ecio_1xrtt|ecio_1xevdo|
   1xevdo_unavail|mobile_unavail}|
   type={email|snmptrap|all}]
```

Configure alarms based on GPIO pin states, where *n* is the GPO pin number

```
set alarm range={1-32} mode=gpio
   [pins=list of pins|highpins=list_of highpins|
   lowpins=list of lowpins|pin{n}={high|low|ignore}|
   trigger_interval=seconds|reminder={on|off}|
   reminder_interval=seconds]
```

Configure alarms based on data pattern matching

```
set alarm mode=match
   match=string
```

Configure alarms based on mobile signal strength (GSM)

```
set alarm mode=rssi_gsm
   sig_strength_threshold=threshold
   time=time
   optimal_alarms_enabled={yes|no}
```

Configure alarms based on signal strength (CDMA)

```
set alarm mode={rssi_1xrtt|rssi_1xevdo|ecio_1xrtt|ecio_1xevdo}
   sig_strength_threshold=threshold
   time=time
   optimal_alarms_enabled={yes|no}
```

Configure alarms based on mobile temperature

```
set alarm mode=mobile_temp
   temperature=temperature
   optimal_alarms_enabled={yes|no}
```

Configure alarms based on mobile service unavailable

```
set alarm mode={1xevdo_unavail|mobile_unavail}
   time=time
   optimal_alarms_enabled={yes|no}
```

Configure alarms based on mobile configuration changes

```
set alarm mode=config_change
```

Set alarms based on cellular data traffic

```
set alarm mode=cell_data
   cell_data=byte count threshold
   cell_data_type={receive|transmit|total}
   time=max time
```

Display current alarm settings

```
set alarm [range={1-32}]
```

Options

General alarm options

state= {on|off}

Enables or disables all alarms.

on

Enables all alarms.

off

Disables all alarms. To avoid false errors, it is recommended that you configure alarms while alarms are disabled, and enable alarms after they are fully configured.

The default is “off.”

mailserverip=ipaddress

Used to configure IP address of the mail server to which alarm-triggered emails are sent.

from=*string*

The text to be included in the “from” field of an alarm-triggered email.

cwm={on|off}

Enables or disables sending of alarm notifications to the Connectware Manager server.

on

Send all alarm notifications to the Connectware Manager server. Turn this option on if your Digi Connect device is managed by Connectware Manager. Enabling this option is useful because it allows all alarms to be monitored from one location, the Connectware Manager. Enabling this option also allows Digi Connect devices to send alarms to clients that would otherwise be unreachable from the Digi Connect device, either because the Digi Connect device is behind a firewall or not on the same network as the alarm destination.

off

Disables sending of alarm notifications to the Connectware Manager server. Leave this option off if you do not manage your devices with Connectware Manager or if you wish to have alarms sent from the device, for example, because an SNMP trap destination is local to the device, not the Connectware Manager server.

For more information on Connectware Manager, see the *Connectware Manager Operator’s Guide,* and the Connectware Manager online help.

Options for setting multiple alarms with the “range” option

range= {1-32}
Specifies the alarm or range of alarms for which alarm options are set.

active={on|off}
Enables or disables an alarm.

on
Enables an alarm.

off
Enables an alarm.

The default is “off.”

cc=*string*
The text to be included in the “cc” field of an alarm triggered email.

mode={match|gpio|rssi_gsm|cell_data|rssi_1xrtt|rssi_1xevdo|ecio_1xrtt|ecio_1xevdo|mobile_temp|1xevdo_unavail|mobile_unavail|config_change}
e alarm mode, which determines what type of event will trigger an alarm. The default mode is “gpio,” unless the Digi product does not support GPIO pins, in which case, the default is “match.”

match
An alarm will be triggered when a pattern is found in the stream of serial data.

gpio
Transitions for GPIO pins will trigger alarms. See "GPIO pin state-based alarm options" on page 75 for more information.

rssi_gsm
Alarms are triggered when the average signal strength on a GSM device falls below a specified threshold for a specified amount of time. Optionally, a subsequent alarm is triggered when signal strength returns to the optimal state above the threshold.

cell_data
Alarms are triggered based on cellular data exchanged in an amount of time

rssi_1xrtt
Alarms are triggered when the average RSSI 1xRTT signal strength on a CDMA device falls below a specified threshold for a specified amount of time. Optionally, a subsequent alarm is triggered when signal strength returns to the optimal state above the threshold.

rssi_1xevdo
Alarms are triggered when the average RSSI 1xEVDO signal strength on a CDMA device falls below a specified threshold for a specified amount of time. Optionally, a subsequent alarm is triggered when signal strength returns to the optimal state above the threshold.

ecio_1xrtt
Alarms are triggered when the average Ec/Io 1xRTT signal strength on a CDMA device falls below a specified threshold for a specified amount of time. Optionally, a subsequent alarm is triggered when signal strength returns to the optimal state above the threshold.

ecio_1xevdo
Alarms are triggered when the average Ec/Io 1xEVDO signal strength on a CDMA device falls below a specified threshold for a specified amount of time. Optionally, a subsequent alarm is triggered when signal strength returns to the optimal state above the threshold.

mobile_temp
Alarms are triggered when the mobile temperature goes above a specified threshold. Optionally, a subsequent alarm is triggered when signal strength returns to a temperature below the threshold.

1xevdo_unavail
Alarms are triggered when 1xEVDO service is unavailable for a specified amount of time. Optionally, a subsequent alarm is triggered when service becomes available again.

mobile_unavail
Alarms are triggered when mobile service is unavailable (not registered on the home network) for a specified amount of time. Optionally, a subsequent alarm is triggered when service becomes available again.

config_change
Alarms are triggered when mobile configuration is changed.

priority={normal|high}
The priority of the triggered email.

normal
The email is sent with normal priority.

high
The email is sent with high priority.

The default is “normal.”

subject=*string*
If “type=email,” this option specifies the text to be included in the “subject” field of an alarm-triggered email. If “type=snmptrap,” this option specifies the text to be included in the “Serial Alarm Subject” field of an alarm-triggered SNMP trap.

to=*string*
The text to be included in the “to” field of an alarm-triggered email.

type={email|snmptrap|all}

Used to determine what kind of an alarm is sent: an e-mail alarm, an SNMP trap or both.

For SNMP traps to be sent, the IP address of the system to which traps are sent must be configured, by issuing a “set snmp” command with the “trapdestip” option. See "set snmp" on page 198.

email

An email alarm is sent.

snmptrap

An SNMP trap is sent. If snmptrap is specified, the “subject” text is sent with the alarm. The MIBs for these traps are DIGI-SERIAL-ALARM-TRAPS.mib, and DIGI-MOBILETRAPS.mib.

all

Both an email alarm and SNMP trap are sent.

The default is “email.”

GPIO pin state-based alarm options

In GPIO mode, alarms are triggered when there are transitions between states for GPIO pins. These options allow you set which GPIO pins’ transitions trigger alarms.

pins=*list of pins*

A list of GPIO pins that trigger alarms.

highpins=*list of highpins*

A list of GPIO pins that trigger alarms when a pin’s signal is high.

lowpins=*list of lowpins*

A list of GPIO pins that trigger alarms when a pin’s signal is low.

pin{*n*}={high|low|ignore}

An alternative way to specify the action of a given GPIO pin, where *n* is the pin number.

high

The pin will trigger an alarm when the pin’s signal is high.

low

The pin will trigger an alarm when the pin’s signal is low.

ignore

The pin will not trigger an alarm.

The default is “ignore.”

reminder={on|off}

The type of reminder sent.

on

An email or SNMP trap is sent periodically while the alarm-triggering event is active. The interval is based on the value of the “reminder_interval” option.

off

An email or SNMP trap is sent only when an alarm is triggered.

reminder_interval=*seconds*

The minimum reminder interval in seconds. Indicates how often an email or SNMP trap is sent when the “reminder” option is set to “on” and an alarm-triggering event is active.

trigger_interval=*seconds*

The minimum trigger interval in seconds. If the “reminder” option is set to “off,” this option indicates the minimum amount of time that is allowed between alarm-triggered emails or SNMP traps.

Data pattern matching-based alarm options

In data pattern match mode, an alarm will be triggered when a pattern is found in the stream of serial data. These options are used for setting alarms in data pattern match mode:

mode=match

Sets the alarm to match mode.

match=*string*

A string that triggers an alarm if the data pattern is found in the incoming serial stream. The maximum length of this string is 40 characters, including escape sequences for special characters. For more details on the escape sequences, see "Entering Special Characters in String Values" on page 13. The maximum parsed length of this string is 10 characters. That is, this string must reduce down to a 10-character string when the escape sequences are processed.

Signal-strength-based alarm options

Options for setting alarms based on signal strength are supported in Digi Cellular Family products only. Note that not all signal-strength options are available with all radios. To determine which signals can be monitored, use the “display mobile” command. If a signal is displayed, it can be monitored.

mode=rssi_gsm

Sets the alarm to signal-strength mode for a GSM radio.

mode=rssi_1xrtt

Sets the alarm to 1xRTT RSSI signal-strength mode for a CDMA radio.

mode=rssi_1xevdo

Sets the alarm to 1xEV-DO RSSI signal-strength mode for a CDMA radio.

mode=ecio_1xrtt

Sets the alarm to 1xRTT Ec/Io signal-strength mode for a CDMA radio.

mode=ecio_1xevdo
Sets the alarm to 1xEV-DO Ec/Io signal-strength mode for a CDMA radio.

signal_strength_threshold=*threshold*
The threshold average signal strength., This is measured in dBm for rssi_gsm, rssi_1xrtt, and rssi_1xevdo (typically -120 dBm to -40 dBm) and dB for ecio_1xrtt and ecio_1xevdo (typically -24 dB to -2 dB).

Note that “0 dB” reported by a “display mobile” command when there is no signal strength should properly be interpreted as the minimum value, which is -120 dBm or -24 dB.

time=*time*
The time in minutes that the average signal strength stays below the threshold specified on the “signal_strength_threshold” option.

optimal_alarms_enabled={yes|no}
If optimal alarms are enabled, an optimal alarm will also be sent when the signal strength returns to a value that is above the specified threshold. Default is no.

Cellular data traffic-based alarm options

These options for setting alarms based on cellular data traffic are supported in Digi Cellular Family products only.

mode=cell_data
Sets the alarm to cellular-data mode.

cell_data=*byte count threshold*
The number of bytes of cellular data to be counted before triggering an alarm.

cell_data_type={receive|transmit|total}
The type of cellular data to be counted.

receive
Data received by the Digi Cellular device.

transmit
Data transmitted by the Digi Cellular device.

total
The total data received and transmitted the Digi Cellular device.

time=*max time*
The time, in minutes, during which bytes of cellular data are counted.

Mobile temperature-based alarm options

These options for setting alarms based on the radio's reported temperature are supported in Digi Cellular Family products only. Note that temperature-based options are not available with all radios. To determine if your radio supports these alarms, use the "display mobile" command. If the temperature is displayed, it can be monitored.

mode=mobile_temp
Sets the alarm to mobile temperature mode.

optimal_alarms_enabled={yes|no}
If optimal alarms are enabled, an optimal alarm will also be sent when the temperature returns to a value that is below the specified threshold

Mobile service-based alarm options

These options for setting alarms based on the service state of the radio are supported in Digi Cellular Family products only. Note that some service-based options are not available with all radios. To determine if your radio supports these alarms, use the "display mobile" command. If the service is displayed, it can be monitored.

mode=1xevdo_unavail
Sets the alarm to monitor 1xEV-DO service.

mode=mobile_unavail
Sets the alarm to monitor registration status. Any registration status other than "Registered (Home Network)" is considered to be a mobile unavailable status.

time=*time*
The time in minutes that the specified service has been unavailable.

optimal_alarms_enabled={yes|no}
If optimal alarms are enabled, an optimal alarm will also be sent when the service being monitored is reestablished. Default is no.

Mobile-oriented configuration change alarm options

This option for setting alarms based on a change to any of the mobile-oriented configurations is supported in the Digi Cellular Family products only. The following configuration items are monitored: PPP settings (set pppoutbound) associated with the mobile radio; PPP Bridge settings (set passthrough); Mobile settings (set surelink); or mobile provisioning, manual or IOTA.

mode=config_change
Sets the alarm to monitor mobile configuration changes.

Examples

Set a GPIO alarm and send an email message or SNMP trap

This example shows how to set up a GPIO alarm to trigger when two GPIO pins go high, and sending an email message when they do. It also shows how to change from sending an email message when the alarm condition occurs to issuing an SNMP trap.

Turn off alarms and set global email properties (this is done to avoid false error conditions triggering alarms):

```
#> set alarm state=off mailserverip=10.0.0.1 from=myemail@digi.com
```

Set alarm #1 mode to GPIO mode:

```
#> set alarm range=1 mode=gpio
```

Set alarm #1 to designate which pins trigger alarm:

```
#> set alarm range=1 pin2=high pin3=high
```

```
#> set alarm range=1 highpins=2,3
```

Set alarm # 1 to send an email message when the alarm condition is met, and enable alarm #1:

```
#> set alarm range=1 active=on type=email to=destination@digi.com
subject=”Alarm 1 triggered”
```

Change alarm #1 to send an snmp trap:

```
#> set alarm range=1 highpins=2,3 type=snmptrap
```

Enable alarms:

```
#> set alarm state=on
```

Set up signal-strength and cellular-traffic alarms and send them to Connectware Manager

This example shows how to set up two alarm and have them be s to be sent to the Connectware Manager server. It configures two alarms: Alarm #6 for is based on a threshold signal strength value (rssi), and alarm #7 is based on cellular data traffic (cell_data).

Disable alarms during configuration:

```
#> set alarm state=off
```

Set alarm #6 to trigger when average GSM rssi drops below -80 dB for at least 20 minutes. Turn alarm #6 on.

```
#> set alarm range=6 active=on mode=rssi_gsm sig_strength_threshold=-80
time=20
```

Set alarm #7 to trigger when more than 10000 bytes are sent in a period of 5 minutes. Turn alarm #7 on.

```
#> set alarm range=7 active=on mode=cell_data cell_data=10000
cell_data_type=transmit time=5
```

Set alarm #8 to trigger when mobile radio temperature exceeds the default thresholds. Turn alarm #8 on.

```
#> set alarm range=8 active=on mode=mobile_temp
```

Set alarm #9 to trigger when 1xEV-DO service is unavailable for a period of 3 minutes and also trigger when service becomes available again. Turn alarm #9 on.

```
#> set alarm range=9 active=on mode=1xevdo_unavail time=3
optimal_alarms_enabled=yes
```

Set alarm #10 to trigger when the mobile configuration has been changed.

Turn alarm #10 on.

```
#> set alarm range=10 active=on mode=config_change
```

Set all alarms to be sent to Connectware Manager, and turn on alarms:

```
#> set alarm state=on cwm=on
```

See also

- "set gpio" on page 113. This command determines whether pins act as GPIO input, GPIO output, or standard serial.
- "set snmp" on page 198.
- "revert" on page 61
- "show" on page 249.

set autoconnect

Devices supported

This command is supported in Digi Connect Family and Digi Cellular Family products only. Not supported in ConnectPort Display.

Purpose

Used to establish an automatic connection (autoconnection) between the serial port and a remote network destination, and to display current autoconnect settings.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display autoconnect settings for the line on which they are logged in: “set permissions s-autoconnect=r-self”
- For a user to display autoconnect settings for any line: “set permissions s-autoconnect=read”
- For a user to display and set the autoconnect settings for the line on which they are logged in: “set permissions s-autoconnect=rw-self”
- For a user to display autoconnect settings for any line, and set the autoconnect settings for the line on which the user is logged in: “set permissions s-autoconnect=w-self-r”
- For a user to display and set the autoconnect settings on any line: “set permissions s-autoconnect=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Configure autoconnect

```
set autoconnect [port=range]
   [state={on|off}]
   [trigger={always|data|dcd|dsr}]
   [service={raw|rlogin|ssl|telnet}]
   [description={string}]
   [ipaddress=ipaddress]
   [ipport=ipport]
   [connect_on_string=string]
   [flush_string={on|off}]
   [keepalive={on|off}]
   [nodelay=on|off]
```

Display autoconnect settings

```
set autoconnect [port=range]
```

Options

port=*range*
Used to specify the serial port. Optional on a single-port device.

state={on|off}
Enables or disables the autoconnect feature.

on
Enables the autoconnect feature.

off
Disables the autoconnect feature.

The default is off.

If you are using the serial port for another purpose, it is recommended this value be set to “off.”

trigger={always|data|dcd|dsr|string}
Indicates which events from the serial port will trigger a network connection to occur.

always
The serial port will continually attempt to keep a connection to a remote network destination active.

data
The serial port will attempt a network connection whenever data arrives on the serial port.

dcd
The serial port will attempt a network connection whenever the serial port’s DCD signal goes high.

dsr
The serial port will attempt a network connection whenever the serial port’s DSR signal goes high.

string
A connection will be made upon detecting a particular sting, specified by the “connect_on_string” option, in the data from the serial port.

The default is “always.”

service={raw|rlogin|ssl|telnet}
The type of network connection that will be established.

raw
A connection without any special processing will occur.

rlogin
A remote login (rlogin) connection will occur.

ssl
A secure connection conforming to SSL (Secure Sockets Layer) Version 3 and Transport Layer Security (TLS) Version 1 will occur.

telnet
A connection with Telnet processing will occur.

The default is “raw.”

description=*string*
A name for descriptive purposes only.

ipaddress=*ipaddress*
The IP address of the network destination to which a connection will be made.

ipport=*ipport*
The TCP port of the network destination to which a connection will be made.

connect_on_string=*string*
When the value of the “trigger” option is string, this option specifies the string that must be found in the serial data in order for a connection to occur. The maximum length of this string is 32 characters, including escape sequences for special characters. For more details on the escape sequences, see "Entering Special Characters in String Values" on page 13. The maximum parsed length of this string is 32 characters. That is, this string must reduce down to a 32-character string when the escape sequences are processed.

flush_string={on|off}
Indicates whether the connect string, specified by the “connect_on_string” option, is flushed or sent over the newly established connection.

on
The connect string is flushed.

off
The connect string is sent over the newly established connection.

The default is on.

keepalive={on|off}
Indicates whether or not TCP keepalives will be sent for the specified range of clients. If set to on, keepalives will be sent, if it is off, keepalives will not be sent.

Configurable TCP keepalive parameters, for example, how many keepalives to send and when to send them are configured globally via the “set network” command (see "set network" on page 149).

nodelay={on|off}
Used to allow unacknowledged or smaller than maximum segment sized data to be sent.

Note: The “nodelay” option disables Nagle’s algorithm, which is on by default, for some TCP services. The purpose of Nagle's algorithm is to reduce the number of small packets sent. Briefly Nagle's algorithm says to hold on to outgoing data when there is either unacknowledged sent data or there is less than maximum segment size (typically around 1500 bytes for Ethernet) worth of data to be sent. It does a good job at keeping transmission efficient, but there are times where it is desirable to disable it.

Examples

Set autoconnect on with trigger

This example shows setting autoconnect to connect to the TCP port (2101) of the network IP destination when data arrives on the serial port.

```
#> set autoconnect state=on trigger=data ipaddress=10.0.0.1 ipport=2101
```

Allow outgoing data that is either unacknowledged or less than maximum segment size

```
#> set autoconnect port=1 nodelay=on
```

See also

- "revert" on page 61.
- "set network" on page 149.
- "set serial" on page 189.
- "set tcpserial" on page 212.
- "show" on page 249.

set bsc

Devices supported

This command supported in Digi Connect WAN Sync only.

Purpose

Configures the binary synchronous feature (also known as bisync or BSC) that provides bisync to IP protocol configuration. This feature allows you to attach a bisync terminal to the Digi device server's serial port. The Digi device server emulates the host by polling the terminal. The bisync data is forwarded to the host over the TCP/IP network.

The Digi device can be configured in either client or server mode for bisync communications.

Configuring the bisync feature involves:

- Enabling/disabling bisync communications and configuring serial settings.
- Configuring polling settings.
- Configuring the network service settings for bisync communications.

Setting any of the bisync options will apply those options immediately, with the current network connection. If the Digi device server is up and running, has a network connection to the host, and you change the polling address, the Digi unit will apply that to the next poll command it sends. However, if you change anything on the network options, the current network connection will be terminated, and a new connection will be established using the new configuration.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display bisync settings: "set permissions s-bsc=read"
- For a user to display and set bisync settings: "set permissions s-bsc=rw"

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Enable/disable bisync communications and configure serial settings

```
set bsc [state={disabled|enabled}]
   [serial_mode={bisync3270|bisync3275}]
   [baudrate={baud rate)|external_clock}]
```

Configure polling settings

```
set bsc [code_set={ascii|ebcdic}]
   [text_conversion={on|off}]
   [polling_address=0x followed by 2 hex digits]
   [selection_address=0x followed by 2 hex digits]
   [device_id=0x followed by 2 hex digits]
   [poll_interval=milliseconds]
   [rx_timeout=milliseconds]
   [tx_retries=0-255]
```

Configure bisync network services

```
set bsc [mode={client|server}]
   [server_hostname={name|ip address}] (used in client mode only)
   [server_port=tcp port] (used in client and server mode)
   [use_ssl={on|off}]
   [tpdu_header={on|off}]
   [power_on_message=(0x followed by up to 64 hex digits)]
   [no_response_message=(0x followed by up to 64 hex digits)]
```

Display current BSC settings

```
set bsc
```

Options

Options for enabling/disabling bisync communications and serial settings

state={disabled|enabled}

Enables or disables bisync processing for the Digi device server.

Important: Do not set “state=enabled” until all other bisync parameters are set. Enabling bisync processing causes the Digi device server to try to initiate the network connection as soon as it gets enabled.

serial_mode={bisync3270|bisync3275}

Sets the bisync protocol used by the bisync terminal connected to the serial port. The protocol specified on this option should match that of the attached bisync terminal. The two protocols are similar with their main difference being use of characters.

baudrate={*baud rate*)|external_clock}

The baud rate for the Digi device server. You can set the baud rate to a specific value or have the Digi device server get the baud rate from an external clock.

baud rate

Set the baud rate to match that of the bisync terminal connected to the serial port. The Digi device server supplies an external clock signal at the selected baud rate which can be used by the bisync terminal.

external_clock

If the bisync terminal supplies an external clock signal, use this setting to have the Digi device server determine the data rate.

Options for polling settings

code_set={ascii|ebcdic}

Specifies whether the bisync terminal uses the ASCII or EBCDIC code set. This option determines the control characters used for serial communication and the format of text data if the “text_conversion” option is “on.”

This option must match the setting used by the connected device. The default value is “ebcdic.”

text_conversion={on|off}

Specifies if data sent to and received from the bisync terminal should be converted to the opposite code set as selected by the “code_set” option.

If “text_conversion=on” and “code_set=ascii,” text data sent to the connected device is converted from EDCDIC to ASCII, and data received from the connected device is converted from ASCII to EBCDIC.

If “text_conversion=on” and “code_set=ebcdic,” text data sent to the connected device is converted from ASCII to EDCDIC, and data received from the connected device is converted from EBCDIC to ASCII.

If “text_conversion=off,” text data is not changed.

The default value is “on.”

polling_address=0x followed by 2 hex digits

Specifies the 8-bit control unit address used to poll the bisync terminal. Polling is used to receive data from the connected device. This address determines which device responds to polling when multiple devices are connected to the serial line.

This option must match the value expected on the serial line by the connected device. It should be in the code set specified by the “code_set” option. The default value is “0xC1,” which is used to poll control unit 1 in the EBCDIC code set.

selection_address=0x followed by 2 hex digits

Specifies the 8-bit control unit address used to select the bisync terminal. Selection is used to send data to the connected device. This address determines which device responds to selection when multiple devices are connected to the serial line.

This option must match the value expected on the serial line by the connected device. It should be in the code set specified by the “code_set” option. The default value is “0x61,” which is used to select control unit 1 in the EBCDIC code set.

device_id=0x followed by 2 hex digits

Specifies the 8-bit device address used to poll and select the bisync terminal. This address determines which sub device responds when multiple sub devices are controlled by a single control unit.

This option must match the value expected on the serial line by the bisync terminal. It should be in the code set selected by the “code_set” option. The default value is “0x40,” which is indicates device 0 in the EBCDIC code set.

poll_interval=*milliseconds*
Specifies the time interval between successive attempts to poll the bisync terminal. The default value is “1000,” or 1 second.

rx_timeout=*milliseconds*
Specifies the maximum time interval between sending data and receiving a response from the bisync terminal. If the expected response is not received within this interval, the data being sent is normally repeated to accommodate serial line transmission errors. The default value is “3000,” or 3 seconds.

tx_retries=0-255
Specifies the maximum number of additional attempts to send data when a response is not received from the bisync terminal. If the expected response is not received after all attempts, the data being sent is lost. The default value is “3”.

Options for configuring bisync network services

mode={client|server}
Specifies whether the Digi device server is in bisync server or client mode. This is the mode that will be used to establish a connection with the host.

client
In bisync client mode, the Digi device server will initiate a connection to the server hostname, which is either specified as a name or an IP address on the “server_hostname” option and using the TCP port specified by the “server_port” option.

server
In bisync server mode, the Digi device server will listen for a connection from the network, using the TCP port specified by the “server_port” option. Note that configuring server mode involves specifying the “server_port” option but not the “server_hostname” option.

server_hostname={*name*|*ip address*} (used in bisync client mode)
The server hostname to which the Digi device server connects in bisync client mode. This option can be specified as a name or an IP address.

server_port=*tcp port* (used in both bisync client and server mode)
For bisync client mode, this value is the TCP port that Digi device server uses to make a connection to the server.

For bisync server mode, this value is the TCP port on which the Digi device server listens for a connection from the network.

use_ssl={on|off}
Enables or disables encrypting the network connection using the Secure Sockets Layer (SSL) protocol. When enabled, the bisync data sent over the TCP/IP network is encrypted using SSL.

tpdu_header={on|off}
Enables or disables a Transport Protocol Data Unit (TPDU) header that is added to IP packets transmitted between the host and the Digi device server.

power_on_message=(0x followed by up to 64 hex digits)
The status message that is sent to the host the first time a connection is established after the Digi device server has been powered on or rebooted.

no_response_message=(0x followed by up to 64 hex digits)
The status message that is sent to the host when there has been no response from the bisync serial device. The message is sent after the receive has timed out for the number of retries specified on the “tx_retries” option.

Example

```
#> set bsc serial_mode=bisync3275 baudrate=2400
#> set bsc code_set=ascii text_con=on polling_addr=0xc1 selection_addr=0x61
device_id=0x40
#> set bsc poll_interval=1000 rx_timeout=3000 tx_retries=3
#> set bsc mode=client server_hostname=test.example.com server_port=8501
use_ssl=on
#> set bsc tpdu_header=on power_on_message=0x506F776572204F6e
#> set bsc no_response_message=0x4E6F2052657370
#>
#> set bsc state=enabled
```

See also

- "revert" on page 61.
- "show" on page 249.
- The online help for the Web user interface’s Bisync (BSC) Settings page.
- The *Cellular Family User’s Guide’s* section on configuring Bisync (BSC) settings.

set buffer

Devices supported

This command is supported in Digi Connect Family and Digi Cellular Family products only. Not supported in ConnectPort Display.

Purpose

Configures buffering settings on a port, or displays the port buffer configuration settings on all ports. The port buffering feature allows you to monitor incoming ASCII serial data in log form.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display the port buffering settings for the line on which they are logged in: “set permissions buffers=r-self”
- For a user to display the port buffering settings for any line: “set permissions buffers=read”
- For a user to display and set the port buffering settings for the line on which they are logged in: “set permissions buffers=rw-self”
- For a user to display the port buffering settings for any line, and set port buffering settings for the line on which the user is logged in: “set permissions buffers=w-self-r”
- For a user to display and set the port buffering settings on any line: “set permissions buffers=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Configure port buffering

```
set buffer [clear] [port=number] [size=number]
   [state={on|off|pause}]
```

Display port buffering settings

```
set buffer [port=port]
```

Options

clear

Clears the contents of the specified buffer.

port

The port or ports to which the command applies.

size

The size in kilobytes to configure the buffer. Settings are configurable in 2-kilobyte increments. The maximum size is 64 kilobytes. The default is 32 kilobytes.

state
The buffering state, which can be any of the following:

on
The data will be buffered.

off
The data will not be buffered and all data will be cleared from the buffer.

pause
The data will not be buffered, but data in the buffer will not be cleared.

Examples

Display port buffer configuration for all ports

```
#> set buffer
```

Configure buffers

In this example, the set buffer command sets the buffer state for port 1 to on mode and the buffer size to 64 kilobytes.

```
#> set buffer port=1 state=on size=64
```

See also

- "revert" on page 61.
- "show" on page 249.

set ddns

Devices supported

This command is supported in Digi Cellular Family products only.

Purpose

Configures a Dynamic DNS (DDNS) service. DDNS allows a user whose IP address is dynamically assigned to be located by a host or domain name.

A DDNS service provider typically supports the registration of only public IP addresses. When using such a service provider, if your Digi Cellular Family device has a private IP address (such as 192.168.x.x or 10.x.x.x), your update requests will be rejected.

Your Digi Cellular Family device monitors the IP address it is assigned. It will typically update the DDNS service or server automatically, but only when its IP address has changed from the IP address is previously registered with that service.

Important:

- Before using this command, you must contact your DDNS service provider and create an account with them. Currently, the only supported DDNS service provider is Dynamic DNS (DynDNS.com). The provider will give you account information such as username and password that you will enter on this command to register the IP address of your Digi Cellular Family product with their service, and update it as it changes.
- The DDNS service supports only *public* IP addresses. If you have a private IP address (such as 192.168.x.x or 10.x.x.x), your update requests will be rejected.
- DDNS service providers may consider frequent updates to be an abuse of their service. In such a circumstance, the service provider may act by blocking updates from the abusive host for some period of time, or until the customer contacts the provider. Please observe the requirements of the DDNS service provider to ensure compliance with possible abuse guidelines.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display Dynamic DNS settings: “set permissions s-ddnsupdater=read”
- For a user to display and set Dynamic DNS settings: “set permissions s-ddnsupdater=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Set basic DDNS settings

```
set ddns [service={disabled|dyndnsorg}]
   [action={updatenow|clearstatus}]
```

Set service settings for Dynamic DNS (service=dyndnsorg):

```
set ddns [ddconntype={standardhttp|alternatehttp|securehttp}]
   [ddsystem={dyndns|statdns|custom}]
   [ddusername=user name]
   [ddpassword=password for DynDNS.org account]
   [ddhostname=full host name]
   [ddwildcard={off|on|nochg}]
```

Display current DDNS settings

```
set ddns
```

Options

Basic DDNS settings

service={disabled|dyndnsorg}

Specifies the DDNS service used for handling dynamic DNS updates, or disables use of the DDNS service.

disabled

Turns off the Dynamic DDNS service. This the default setting for the Dynamic DNS feature. Use this option if you have configured the Dynamic DNS feature and you want to temporarily turn off use of the service for some reason.

dyndnsorg

Update the DDNS service at DynDNS.com. Currently, this is the only Dynamic DNS service supported. When you select a specific DDNS service provider, you must also provide the related account information for that service provider.

action={updatenow|clearstatus}

The action to be performed on the DDNS service. If “action” is specified, other command options are not used.

updatenow

Force a DDNS service update now.

clearstatus

Clears the last information returned from the Dynamic DNS service from your Digi Cellular device.

DynDNS.com service settings

These settings are specific to your account information with DynamuiDynDNS.org; please consult their website for more information on account terms and settings.

ddconntype={standardhttp|alternatehttp|securehttp}
The method to use to connect to the DynDNS.org server:

standardhttp
Connect to the Standard HTTP port (80).

alternatehttp
Connect to the Alternate HTTP port (8245).

securehttp
Connect to the Secure HTTPS port (443).

ddsystem={dyndns|statdns|custom}
The DynDNS.org system to use for the update.

dyndns
Update a Dynamic DNS host name.

statdns
Update a Static DNS host name.

custom
Update a Custom DNS host name.

ddusername=*user name*
The user name for the DynDNS.org account.

ddpassword=password
The password for DynDNS.org account.

ddhostname=full host name
The full host name to update for DynDNS.org account, for example, myhost.dyndns.net.

ddwildcard={off|on|nochg}
Enables/disables wildcards for this host.

According to wildcard documentation at DynDNS.org: “The wildcard aliases *.yourhost.ourdomain.tld to the same address as yourhost.ourdomain.tld.”

Using this option in the settings for your Digi Cellular Family device has the same effect as selecting the wildcard option on the DynDNS.org web site. To leave the wildcard option unchanged from the current selection on their web site, use the “no change” option (the “nochg” keyword) in the device settings. Note that DynDNS.org support for this option may vary according to the DynDNS system you are registered to use.

The available choices for this option are:

off
Disables wildcards.

on
Enables wildcards.

nochg
Specifies that there should be no change to service setting from the current selection for wildcards in the DDNS settings at the DynDNS.org Web site.

Examples

This example shows “set ddns” being used to display the current status of the Dynamic DNS service:

```
#> set ddns

 DDNS Service Update Configuration :

    service     : dyndnsorg

    ddconntype  : securehttp
    ddsystem    : statdns
    ddusername  : "test"
    ddpassword  : (Not shown for security reasons)
    ddhostname  : "test-static.dnsalias.com,test-static.dnsalias.org"
    ddwildcard  : nochg

 Current IP address: 166.213.228.220 (ppp1)

 Most recent DDNS service update status:

    Service              : DynDNS.org
    IP address reported  : 166.213.228.220
    Update status        : successful
    Result information   : [good] The update was successful.
    Raw result data      :
      good 166.213.228.220
      good 166.213.228.220

 Most recent DDNS service update log message:
    IP address for "ppp1" is now 166.213.228.220, but no DDNS update is needed
last reported IP address is unchanged).
```

See also

- "display" on page 27.
- "revert" on page 61.
- "show" on page 249.
- Dynamic DNS resources available from your service provider, such as glossary definitions, FAQs, knowledge base articles, and tips for managing your account.

set dhcpserver

Devices supported

This command is supported in Digi Cellular Family products only.

Purpose

Configures the DHCP server settings for the Digi Connect product. A DHCP server allows other devices or hosts on the same local network as the Digi Connect product to be assigned dynamic IP addresses. This DHCP server supports a single subnetwork scope.

The only DHCP server scope currently supported in Digi Connect firmware is for the “eth0” interface. That, the firmware serves IP addresses to DHCP clients on the Ethernet side of the Digi Connect product only, and not the cellular side, which is handled through Point-to-Point Protocol (PPP) instead.

The DHCP server operates only if the Digi Connect product is configured to use static IP address configuration. For information on how to configure static IP settings, see "set network" on page 149 and the help for the Web user interface’s Network Configuration settings.

Once configured, the DHCP server is managed through the “dhcpserver” command. See "dhcpserver" on page 23.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display DHCP server settings: “set permissions s-dhcpserver=read”
- For a user to display and set DHCP server settings: “set permissions s-dhcpserver=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Configure the DHCP server: basic settings

The “item” option must be specified on every instance of a “set dhcpserver” command. The “item” option specifies the configuration settings to which the command will be applied. To enable a scope, or a reservation or exclusion entry within a scope, all of its parameters must be specified and valid. A scope or entry can be enabled only if it is completely valid.

```
set dhcpserver item={scope|reservation|exclusion}
   [action={set|revert}]
```

Configure the DHCP server scope (“item=scope”)

```
set dhcpserver item=scope
   [action={set|revert}]
   [enabled={on|off}]
   [startip=ip address]
   [endip=ip address]
   [leasetime={time|infinite}] (in seconds, 0=default (86400)}
   [offerdelay=0-5000] (time in milliseconds, default 500}
   [conflictdetect={on|off}]
```

Configure the scope's address reservations (“item=reservation”)

In this type of configuration, the DHCP server assigns a particular IP address to a Digi device, rather than a random address from a pool.

```
set dhcpserver item=reservation
   [action={set|revert}]
   [range={1-16|all}]
   [enabled={on|off}]
   [ip=ip address]
   [clientid=client MAC address] {e.g., 00:40:9D:12:34:56}
   [leasetime={time|infinite}] {in seconds, 0=default (use scope's
   time)}
```

Configure the scope's address exclusions (“item=exclusion”)

```
set dhcpserver item=exclusion
   [action={set|revert}]
   [range={1-4|all}]
   [enabled={on|off}]
   [startip=ip address]
   [endip=ip address]
```

Display current DHCP server settings

```
set dhcpserver
```

Options

Options for configuring the DHCP server scope (“item=scope”)

item=scope

Specifies that the DHCP server configuration settings apply to the scope of IP addresses for a network. A scope is the full consecutive range of possible IP addresses for a network. A scope typically defines a single physical subnet on your network, to which DHCP services are offered. A scope is the primary way for the DHCP server to manage distribution and assignment of IP addresses and related configuration parameters to its clients on the network.

action={set|revert}

Specifies the action to be performed by the “set dhcpserver” command.

set

Sets the DHCP configuration settings to the specified values. This is the default setting that is used if this option is not specified.

revert

Resets the configuration settings for the specified item to default values.

enabled={on|off}
Enables the DHCP server feature on this Digi Connect product. Note that for the DHCP server to operate, the Digi Connect product must be configured to use a static IP address.

startip=*ip address*
The first IP address in the pool.

endip=*ip address*
The last address in the pool. The addresses in the range specified by “startip” and “endip” must be in the same subnet as the Digi Connect product.

leasetime={*time*|infinite}
The length, in seconds, of the leases for the scope being served by this DHCP server. Specifying a time of 0 means that the default of 86400 seconds (24 hours) will be used. A DHCP client may request a lease duration other than this setting, and the DHCP server will grant that request, if possible.

offerdelay=0-5000
The interval of time, in milliseconds, to delay before offering a lease to a new client. The range for this delay is 0 to 5000 milliseconds, and the default delay is 500 milliseconds. Use of this delay permits the Digi Connect product to reside on a network with other DHCP servers, yet not offer leases to new clients unless the other DHCP servers do not make such an offer. This provides a measure of protection against inadvertently connecting a Digi Connect product to a network that is running its own DHCP server, and thereby offering leases to clients in a manner inconsistent with that network.

conflictdetect={on|off}
When a DHCP client requests a new IP address lease, before offering an IP address to that client, use "ping" to test whether that IP address is already in use by another host on the network but is unknown to the DHCP Server. If an IP address is determined to be in use, it is marked as "Unavailable" for a period of time, and it will not be offered to any client while in this state.

Enabling this test adds approximately one second of delay before the IP address is offered to the client, since the "ping" test must not receive a valid reply for that test to successfully determine that the IP address is not already in use.

This option is off (disabled) by default. This option does not apply to Static Lease Reservations, since the "ping" test is not used for them.

Options for configuring the scope's address reservations ("item=reservation")

item=reservation
Specifies that the DHCP server configuration settings apply to the scope's address reservations. You may use a reservation to create a permanent address lease assignment by the DHCP server. Reservations assure that a specified hardware device on the subnet can always use the same IP address. Address lease reservations associate a specific IP address with a specific client's Ethernet MAC address.

action={set|revert}
Specifies the action to be performed by the "set dhcpserver" command.

set
Sets the DHCP configuration settings to the specified values. This is the default setting that is used if this option is not specified.

revert
Resets the configuration settings for the specified item to default values. This effectively removes the entry specified by the "range" option.

range={1-16|all}
Selects the specific entry to which the action is to be applied.

enabled={on|off}
Enables the DHCP server feature on this Digi Connect product. Note that for the DHCP server to operate, the Digi Connect product must be configured to use a static IP address.

ip=*ip address*
The IP address reserved for the client. This value must not be the same as the IP address of the DHCP Server itself.

clientid=*client MAC address*
The MAC address for the client, for example, 00:40:9D:12:34:56.

leasetime={*time*|infinite}
The length, in seconds, of the leases for the scope being served by this DHCP server. Specifying a time of 0 means that the default of using the scope's lease time will be used. A DHCP client may request a lease duration other than this setting, and the DHCP server will grant that request, if possible. Leaving this option blank causes the default value specified in the scope to be used. That default will change whenever the scope changes.

Options for configuring the scope's address exclusions (“item=exclusion”)

item=exclusion

Specifies that the DHCP server configuration settings apply to the scope’s address exclusions. An exclusion range is a limited sequence of IP addresses within a scope, excluded from DHCP service offerings. Exclusion ranges assure that any addresses in these ranges are not offered by the server to DHCP clients on the network. Note that the IP address of the DHCP server itself will not be given out to any DHCP clients, even if it is within the range specified on this command.

action={set|revert}

Specifies the action to be performed by the “set dhcpserver” command.

set

Sets the DHCP configuration settings to the specified values. This is the default setting that is used if this option is not specified.

revert

Resets the configuration settings for the specified item to default values. This effectively removes the entry specified by the “range” option.

range={1-4|all}

Selects the specific entry to which the action is to be applied.

enabled={on|off}

Enables the DHCP server feature on this Digi Connect product. Note that for the DHCP server to operate, the Digi Connect product must be configured to use a static IP address.

startip=*ip address*

The first address in the exclusion block.

endip=*ip address*

The last address in the exclusion block. An IP address exclusion range must specify valid IP addresses in the subnetwork of the DHCP server. The DHCP server will ignore an exclusion that does not meet this requirement.

Examples

Configure the IP address range for the DHCP server scope and enable it

```
set dhcpserver item=scope action=set enabled=on
startip=10.30.1.150 endip=10.30.1.199
```

Since the “leasetime” and “offerdelay” options are not specified, the default values for them are used, unless they were previously changed to another value by use of a “set dhcpserver item=scope” command (or using the web UI).

Add an IP address reservation for a client:

```
set dhcpserver item=reservation action=set range=1 enabled=on ip=10.30.1.195
clientid=00:09:26:19:51:05
```

Since the leasetime option is not specified, the DHCP server's scope lease time is used, unless a lease time was previously changed to a value by use of a “set dhcpserver item=reservation action=set range=1” command (or using the web UI).

Disable all reservations that were previously added:

```
set dhcpserver item=reservation action=set range=all enabled=off
```

Permanently remove a reservation that was previously added:

```
set dhcpserver item=reservation action=revert range=1
```

Any client that has the lease for the reserved IP address that is removed in this manner, will still keep its lease. However, without the reservation, future address leases to that client are not guaranteed to be for this same IP address, which is the purpose of a reservation.

Add an IP address exclusion range for the scope:

```
set dhcpserver item=exclusion action=set range=1 enabled=on
startip=10.30.1.170 endip=10.30.1.179
```

This exclusion instructs the DHCP server to not issue leases for the IP addresses from 10.30.1.170 to 10.30.1.179 inclusive. Note that reservation leases may be configured for any address in that range, and the DHCP server will permit a lease of such an address to the correct client only. That is, reservations override exclusions.

Display current DHCP Server settings

```
#> set dhcpserver

 DHCP Server Settings:

   server enabled       : on
   scope name           : eth0
   starting ip address  : 10.30.1.190
   ending ip address    : 10.30.1.198
   lease time           : 3600 (seconds)
   offer delay          : 500 (milliseconds)
   addr conflict detect : off

  Reservation Settings:

   idx enabled ip address      client id          lease time
   --- ------- --------------- -----------------  ----------
   1   on      10.30.1.135     00:40:9D:24:73:F8        3600
   2   on      10.30.1.195     00:09:26:19:51:05           0
   3   on      10.30.1.196     00:09:26:19:51:06           0
   4   on      10.30.1.197     00:09:26:19:51:07           0
   5   on      0.0.0.0         00:00:00:00:00:00           0
   6   on      0.0.0.0         00:00:00:00:00:00           0
   7   off     0.0.0.0         00:00:00:00:00:00           0
   8   off     0.0.0.0         00:00:00:00:00:00           0
   9   off     0.0.0.0         00:00:00:00:00:00           0
   10  off     0.0.0.0         00:00:00:00:00:00           0
   11  off     0.0.0.0         00:00:00:00:00:00           0
   12  off     0.0.0.0         00:00:00:00:00:00           0
   13  off     0.0.0.0         00:00:00:00:00:00           0
   14  off     0.0.0.0         00:00:00:00:00:00           0
   15  off     0.0.0.0         00:00:00:00:00:00           0
   16  off     0.0.0.0         00:00:00:00:00:00           0

   A reservation lease time of 0 means to use the scope's lease time.

  Exclusion Settings:

   idx enabled start address   end address
   --- ------- --------------- ---------------
   1   off     0.0.0.0         0.0.0.0
   2   off     0.0.0.0         0.0.0.0
   3   off     0.0.0.0         0.0.0.0
   4   off     0.0.0.0         0.0.0.0
```

See also

- "dhcpserver" on page 23.
- "revert" on page 61.
- "show" on page 249.
- In the Web user interface, the online help for Network Settings. “Configuring DHCP Sever Settings” provides more information on DHCP terminology and managing DHCP server operation.

set ekahau

Devices supported

This command is supported in Digi Connect Wi-SP, Digi Connect Wi-ME, and Digi Connect Wi-EM only.

Purpose

Configures Ekahau Client™ device-location software in a Digi Connect wireless device.

The Ekahau Client feature provides integrated support for Ekahau's Wi-Fi device-location solution on the Digi Connect Wi-ME, Digi Connect Wi-EM, and Digi Connect Wi-SP products. Ekahau offers a complete access point vendor-independent real-time location system for wireless LAN devices that is capable of pinpointing wireless LAN devices such as the Digi Connect products, laptops, PDAs, or other intelligent Wi-Fi enabled devices. The solution provides floor-, room- and door-level accuracy of up to 3.5 feet (1 m). The patented Ekahau positioning technology is based on simple signal-strength calibration maps, and enables customers to fully leverage an existing wireless LAN infrastructure without any need for proprietary hardware components.

Please visit www.ekahau.com for additional information, including free evaluation licenses for the Ekahau Positioning Engine and Ekahau Site Survey software products.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display Ekahau client settings: “set permissions s-ekahau=read”
- For a user to display and set Ekahau client settings: “set permissions s-ekahau=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Configure Ekahau settings

```
set ekahau [state={on|off}]
   [id=device id]
   [poll_rate=seconds]
   [protocol={tcp|udp}]
   [port=port]
   [server={hostname|ip address}]
   [password=string]
   [name=string]
```

Display current Ekahau settings

```
set ekahau
```

Options

state={on|off}
Enables or disables the Ekahau Client feature.

The “id,” “name,” and “server” values must be set before you can set “state” to “on.”

id=*device id*
A numeric identifier for the Digi Connect device, used internally by the Ekahau Positioning Engine for device tracking over time. This identifier should be unique for each Digi device being located on the network. It must be configured before the device will allow the “state” option to be set to “on.”

poll_rate=*seconds*
The time in seconds between each scan or wireless access points and communication with the server.

Once the Ekahau Client is enabled (“state=on”), every time the Digi Connect device scans the network, it is essentially disassociated with the access point (AP) providing its network connectivity. In addition, during the time, or scanning interval, set by the “poll_rate” option, it will not be receiving or transmitting wireless packets. This could lead to packet loss. Set the “poll_rate” as slow as acceptable in the application where the Digi Connect product is being used.

The default is five seconds.

protocol={tcp|udp}
Specifies whether to use TCP or UDP as the network transport. The default is “tcp.”

port=*port*
The network port to communicate on. In the default Ekahau configuration, port 8548 is used for TCP, and port 8549 for UDP. This setting must be configured before the device will allow “state” to be set to “on.”

server={*hostname*|*ip address*}
The hostname or IP address of the Ekahau Positioning Engine. The maximum length of this option is 50 characters. The default is 8548.

password=*password*
A password to authenticate with the server. The maximum length of this option is 50 characters. The default for Digi and the Ekahau Positioning Engine is 'Llama'.

name=*device name*
A descriptive name to identify the Digi Connect device to users. The maximum length of this option is 50 characters. This name must be configured before the device will allow “state” to be set to “on.”

Examples

Set identifiers

```
#> set ekahau id=1 server=myepe.domain.com name=”Tracked Device 1"
```

Enable Ekahau Client

```
#> set ekahau state=on
```

See also

- "revert" on page 61.
- "show" on page 249.
- For additional information, including free evaluation licenses for the Ekahau Positioning Engine and Ekahau Site Survey software products, see the Ekahau Web site at www.ekahau.com.

set ethernet

Devices supported

This command is supported in the following products:

- Connect Family: Digi Connect SP, Digi Connect EM, Connect ME.
- Digi Cellular Family: All Connect WAN products, but not ConnectPort WAN VPN.
- ConnectPort Display.

Purpose

Configures, adjusts, and displays Ethernet communications options.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions s-ethernet=read” to display Ethernet communications options, and “set permissions s-ethernet=rw” to display and configure Ethernet communications options. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Configure Ethernet communications options

```
set ethernet [duplex={half|full|auto}] [speed={10|100|auto}]
```

Display Ethernet communications options

```
set ethernet
```

Options

duplex

Determines the mode the Digi device uses to communicate on the Ethernet network. Specify one of the following:

half

The device communicates in half-duplex mode.

full

The device communicates in full-duplex mode.

auto

The device senses the mode used on the network and adjusts automatically.

The default is “half.” If one side of the Ethernet connection is using auto, the other side can set the duplex value to whatever is desired. If one side uses a fixed value (for example, half-duplex), the other side has to use the same.

speed

Configures the Ethernet speed the Digi device will use on the Ethernet network. Specify an appropriate setting for your Ethernet network, which can be one of the following:

10

The device operates at 10 megabits per second (Mbps) only.

100

The device operates at 100 Mbps only.

auto

The device senses the Ethernet speed of the network and adjusts automatically.

The default is “auto.” If one side of the Ethernet connection is using auto (negotiating), the other side can set the Ethernet speed to whatever value is desired. Or, if the other side is set for 100 Mbps, this side must use 100 Mbps.

Examples

Configure 100 Mbps Ethernet speed

```
#> set ethernet speed=100
```

See also

- "set network" on page 149 to configure network communications options.
- "revert" on page 61.
- "show" on page 249.

set forwarding

Devices supported

This command is supported in Digi Cellular Family products only.

Purpose

Configures IP routing, or forwarding of IP datagrams, between network interfaces. IP routing must be enabled to allow the Network Address Table (NAT) and port forwarding features to work properly.

The “set forwarding” command enables addition of static route entries to the IP routing table. Static routes instruct the device to route packets for a known destination host or network, to (through) a router or gateway different from the default gateway. They explicitly define the next “hop” from a router for a particular destination. This is sometimes necessary to communicate with hosts on a different subnet than the Digi device, or to provide a more direct or efficient route to such hosts than may be provided by using the default gateway or other routes.

Required privileges

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display the forwarding settings:
  “set permissions s-router=read”
- For a user to display and set forwarding settings:
  “set permissions s-router=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Set IP forwarding settings

The first option, “ipforwarding,” enables or disables IP forwarding. The remaining options pertain to static routes.

```
set forwarding [ipforwarding={on|off}]
   [staticrouteindex={1-8}]
   [action={add|change|delete}]
   [enabled={on|off}]
   [net=destination ip address]
   [mask=subnet mask]
   [gateway=ip address]
   [metric={1-16}]
   [interface=interface name]
```

The “staticrouteindex” and “action” options are required for all static route management commands. To add a new static route entry, all eight of the above static route options are required.

Note that this command cannot be used to add, change or delete a default gateway entry: the default gateway is managed internally by the device.

Change an existing static route table entry

To change an existing static route table entry, these options are required:

```
staticrouteindex={1-8}
action=change
[one or more additional options]
```

Delete a static route table entry

To delete a static route entry, these options are required:

```
staticrouteindex={1-8}
action=delete
```

Enable a static route table entry

To enable a static route entry, all of its options must be specified (on this or a previous command) and valid. That is, an entry can be enabled only if it is completely valid.

When a new entry is added or a change is made to a static route entry and that entry is enabled, the change is applied immediately to the IP routing table. If a static route entry is disabled or deleted, it is removed immediately from the IP routing table.

Display IP forwarding settings

```
show route
```

Display the current active IP routing table

```
display route
```

Options

ipforwarding={on|off}

Enables or disables IP forwarding.

on
Enables IP forwarding.

off
Disables IP forwarding.

staticrouteindex={1-8}

Specifies which of the 8 static route table entries is to be acted upon.

action={add|change|delete}

Specifies the action to be performed on the selected static route table entry.

add
Add a new entry.

change
Change one or more options of an existing entry.

delete
Delete an entry, resetting all its options to defaults (empty).

enabled={on|off}
Enables or disables a static route table entry.

on
Enables an entry. All its options must be specified and valid to enable an entry. The enabled entry is immediately added to the device's IP routing table.

off
Disables an entry. If the entry was previously enabled and added to the device's IP routing table, that entry is immediately removed from the IP routing table.

net=*destination ip address*
Specifies the IP address of destination network or host to which the static route applies. This static route table entry defines how packets will be routed by the device when they are sent to the destination network or host.

mask=*subnet mask*
The subnetwork mask to be used for the static route, which is used in conjunction with the destination IP address. A subnetwork mask of 255.255.255.255 indicates that the destination IP address is a specific host rather than a network.

gateway=*ip address*
The IP address of the gateway for this static route. When the device routes packets that are destined for the specified destination IP address (host or network), those packets are sent to this gateway as their first "hop."

metric={1-16}
Specifies the metric, or the "cost" to reach the destination. This is the "distance" in terms of number of "hops" for the routed packets to get to the destination.

interface=*interface name*
Specifies the name of the local network interface through which routed packets are sent for this static route. Examples are "eth0" and "ppp1." For devices that support PPP interfaces, the actual PPP interface name may vary: ppp1, ppp2, ppp4. The valid network interface names can be displayed using "display netdevice."

Examples

Enable IP forwarding

```
#> set forwarding ipforwarding=on
```

Add a static route entry

This command adds a static route for packets destined to hosts on the 10.10.0.0 network. These packets are sent through the eth0 interface to a local router 10.30.1.1. In this example, the device on which this static route is added has an IP address of 10.30.1.188, and the router 10.30.1.1 is on its local subnetwork.

```
set forwarding staticrouteindex=1 action=add enabled=on net=10.10.0.0
mask=255.255.0.0 gateway=10.30.1.1 metric=2 interface=eth0
```

See also

- "display" on page 27. The “display route” command displays current routing information.
- "set nat" on page 146.
- "revert" on page 61
- "show" on page 249.
- In the online help for the Web user interface, the topic “IP Forwarding Settings” under Network Configuration.

set gpio

Devices supported

This command is supported in the following products:

- Connect Family: Digi Connect EM, Digi Connect Wi-EM, Digi Connect ME, Digi Connect Wi-ME.
- Digi Cellular Family: Digi Connect WAN, and Digi Connect RG devices.

Not supported in ConnectPort Display.

Purpose

Used to:

- Configure General Purpose I/O (GPIO) pins. In normal operation, the GPIO pins are used for the serial CTS, DCD, DSR, DTR, and RTS pins. The set gpio command allows these GPIO pins to be used for different purposes.
- Display current GPIO pin settings.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions s-gpio=read” to display GPIO pin settings, and “set permissions s-ethernet=rw” to display and configure GPIO pins. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Configure GPIO pins

```
set gpio range={1-n} mode={serial|input|output}
```

Display current GPIO pin settings

```
set gpio [range={1-n}]
```

Options

range={1-*n*}

Used to specify the index of the GPIO pin to manipulate, where *n* is the maximum number of GPIO pins on the device.

mode={serial|input|output}

The mode of operation of the GPIO serial pin.

serial

Indicates normal serial operation.

input

Allows input of GPIO signals. This is used in conjunction with alarms to trigger emails or SNMP traps indicating a particular signal change.

output

Allows output of GPIO signals. Currently, output of GPIO signals is not supported in the command-line interface. The web user interface can be used to toggle the output of GPIO signals between high and low.

The default is “serial” for all pins.

Default serial signal settings for GPIO pins

The default serial signal settings for the GPIO pins on a Digi Connect device are as follows. Depending on the device, there are five or nine GPIO pins.

Pin Number	Default Serial Signal	Signal Direction
GPIO 1	DCD	Input
GPIO 2	CTS	Input
GPIO 3	DSR	Input
GPIO 4	RTS	Output
GPIO 5	DTR	Output
GPIO 6	TXD	Output
GPIO 7	RXD	Input
GPIO 8	TXD for port 2	Output
GPIO 9	RXD for port 2	Input

Examples

Changing the operation of the GPIO signal pins

The following command changes GPIO pins 1-5 to allow input of GPIO signals.

```
#> set gpio range=1-5 mode=input
```

See also

- "revert" on page 61.
- "send" on page 67, for details on setting up alarms that issue email messages or SNMP traps when GPIO pins change.
- "show" on page 249.

set group

Devices supported

This command is supported in Digi Cellular Family products Digi Connect WAN and Digi Connect RG only.

Purpose

Used to create and manage user groups. You can use “set group” to do the following:

- Add a group. A maximum of 32 groups can be defined.
- Remove groups.
- Change group configuration attributes.
- Display group configuration attributes.

In order to apply a common set of user settings to more than one user, it may be desirable to create a group with the required settings and then associate that group with multiple users. If a user is a member of one or more groups, the user's effective permissions are the maximum of the permissions of the user and all of the groups to which the user belongs.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions s-group=read” to display group configuration attributes, and “set permissions s-group=rw” to display and set group configuration attributes. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Add a group

```
set group add id=number newname=string
```

Remove a group

```
set group remove {id=range|name=string}
```

Change group configuration attributes

```
set group {id=range|name=string} [newname=string]
   [commandline={on|off}] [menu={none|index|name}]
   [defaultaccess={none|commandline|menu}]
```

Display group configuration attributes

```
set group {id=range|name=string}
```

Display group configuration attributes for all groups

```
set group
```

Options

add
Add a group. New groups are created with no permissions. A maximum of 32 groups can be defined.

remove
Remove groups.

id=*range*
Specifies the ID or range of IDs of the groups to be acted on.

name= *string*
Specifies the name of the group to be acted on.

newname=*string*
Specifies a new group name.

commandline={on|off}
Specifies whether the users in the group are allowed to access the command line of the device.

on
Users can access the command line interface.

off
Users can not access the command line interface.

The default is “on.”

menu={none|index|name}
Specifies whether the users in the group are allowed to access the custom menu interface of the device and defines the custom menu that the users will have displayed.

none
Users are not allowed to access the custom menu interface.

index
Users are allowed to access the custom menu interface and will be displayed the custom menu at the specified index

name
Users are allowed to access the custom menu interface and will be displayed the custom menu using the specified name.

The default is “none.”

defaultaccess={none|commandline|menu}
Specifies the default access method and interface that users in the group will be given upon logging into the device. Note that the specified interface must be enabled for the group and have a valid menu if specified.

none
The group has no default access to the device and the users are not allowed to access either the command line interface or the custom menu interface without explicitly specifying the access method.

commandline
The users will be displayed and given access to the command line interface assuming the group has command line access rights enabled.

menu
The users will be displayed and given access to the custom menu interface and be displayed the custom menu as specified by the “menu” option.

The default is “commandline.”

Default permissions

When a new group is created, it has no permissions.

Examples

Add a new group

```
#> set group add newname=gurus id=4
```

Remove group 7

```
#> set group remove id=7
```

Set a new group name

```
#> set group id=4 newname=gurus
```

Set a group with command line access rights

```
#> set group id=4 commandline=on defaultaccess=commandline
```

Set a group with custom menu access rights to access the “my_menu” custom menu

```
#> set group name=gurus menu=my_menu defaultaccess=menu
```

See also

- "User Models and User Permissions in Digi Connect Products" on page 14.
- "newpass" on page 51
- "revert" on page 61.
- "set menu" on page 132.
- "set permissions" on page 157.
- "set user" on page 220.
- "show" on page 249.

set host

Device support

This command is supported in all products.

Purpose

Configures a name for the device, also known as a host name, or displays the current host name for the device.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions s-host=read” to display the current host name, and “set permissions s-host=rw” to display and set the host name. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Configure a host name for the device

```
set host name=name
```

Display the current host name

```
set host
```

Options

name=*name*

The name for the device. The name can be up to 32 characters long, and can contain any alphanumeric characters, and can also include the underscore (_) and hyphen (-) characters.

See also

- "show" on page 249.

set ia

Device support

This command is supported in Digi Connect WAN IA, Digi Connect SP, Digi Connect Wi-SP, Digi Connect ME4, Digi Connect Wi-ME, Digi Connect EM, Digi Connect Wi-EM.

Purpose

Configures selected Digi devices to support special parsing and bridging of Industrial Automation (IA) protocols. For example, it enables the Digi device to function as a Modbus/TCP to serial Modbus Bridge. Command options allow for configuring:

- Protocol details of serial-port connected devices
- Protocol details of network-based masters
- Destination tables and route entries within the tables that define how protocol requests are forwarded to one of many slaves for a response.

Supported IA protocols include Modbus/RTU and Modbus/ASCII on the serial port or encapsulated within TCP/IP or UDP/IP, and Modbus/TCP transported by either TCP/IP or UDP/IP.

For more information on Industrial Automation, see the IA application help available at this URL:

http://www.digi.com/support/ia

Displaying current IA settings is best done by using the “show” command, as shown in the Syntax section below.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions s-ia=read” to display the current IA settings, and “set permissions s-ia=rw” to display and configure IA settings. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

There are several variants of syntax for the “set ia” command, depending on whether the command is being used to configure serial port-connected devices, network-based masters, or destination tables and route entries within destination tables. These syntax descriptions and their option descriptions are presented separately.

Factory defaults for Industrial Automation settings

The factory defaults for Industrial Automation settings are for a Modbus bridge with Modbus/RTU serial slaves attached. This configuration is also automatically created if you change the serial port profile to IA from another profile and there is no existing IA configuration. The default configuration is:

- The IP address is assigned by DHCP client (there is no fixed IP address).
- Internal DHCP server is disabled (off).
- serial “term” login is disabled (off) (you cannot log into the serial port).
- “set ia serial=1” assumes Modbus/RTU slaves are attached at 9600:8,N,1. The protocol selected on the serial port is matched by an

implied network master encapsulated on TCP and UDP port 2101. So by default, incoming Modbus/RTU in TCP or UDP port 2101 is enabled.

- “set ia master=1” enables incoming Modbus/TCP masters.
- “set ia master=2” enables incoming Modbus/UDP masters (this is Modbus/TCP format by UDP port 502).
- “set ia table” assumes slaves 1 to 32 are on the serial port. Slave address (or Unit ID or Bridge Index) zero will be treated as 1. Slave addresses 33 to 254 are assumed to be Modbus/TCP slaves on the local Ethernet subnet, with the slave address used as the last octet of the IP. For example, if the Digi device has the IP address 10.45.203.1, then slave 75 is assumed to be a Modbus/TCP server at 10.45.203.75 on the local Ethernet.

Configure serial-port connected devices (set ia serial)

```
set ia serial=range [serial options] [modbus options]

[serial options]:
  type={master|slave}
  table=1..8
  protocol={modbusrtu|modbusascii}
  messagetimeout=100-99999 ms
  slavetimeout=10-99999 ms
  chartimeout=3-99999 ms
  idletimeout={0=disabled|1-99999 seconds}
  priority={high|medium|low}

[modbus options]:
  errorresponse={on|off}
  broadcast={on|off|replace}
  fixedaddress={auto|1-255}
```

To set the baud rates for the port, see "set serial" on page 189.

To enable IA protocols, set the serial port profile to “ia.” see "set profile" on page 172.

Configure network-based masters (set ia master)

```
set ia master=range
  state={on|off}
  active={on|off}
  type={tcp|udp}
  ipport=ip port
  table=1..8
  protocol={modbusrtu|modbusascii|modbustcp}
  messagetimeout=100-99999 ms
  chartimeout=3-99999 ms
  idletimeout={0=disabled|1-99999 seconds}
  priority={high|medium|low}
```

Configure destination tables and route entries (set ia table)

```
set ia table=range [table options]
   [route=range [route options]]

[table options]:
   state={on|off}
   name=string
   addroute=route index
   removeroute=route index
   moveroute=from_route_index,to_route_index

[route options]:
   active={on|off}
   connect={active|passive}
   protaddr=protocol address range
   type={discard|ip|mapto|nopath|serial}
   protocol={modbusrtu|modbusascii|modbustcp}
   port=serial port
   transport={tcp|udp}
   ipaddress=ip address
   ipport=ip port
   replaceip={on|add|sub|off}
   mapto=protocol address
   slavetimeout=10-99999 ms
   chartimeout=3-99999 ms
   idletimeout={0=disabled, 1-99999 seconds}
   reconnecttimeout=0-99999 ms
```

Display current IA settings

To display current IA settings, the “show” command is recommended instead of a “set ia” command with no options:

```
show ia all
```

Options

Options for configuring serial-port connected devices (set ia serial)

serial=*range*

Specifies that the serial settings apply to the specified serial port or range of serial ports. The default is port 1, and on a single-port device, entering serial is the same as serial=1.

[*serial options*]

The serial settings, which include:

state={on|off}

Enables the IA serial settings. Setting to off risks the configuration being deleted.

active={on|off}

To temporarily stop processing the serial port, set “active” to “off.” The configuration will remain valid and saved.

type={master|slave}

Defines whether the serial device attached is acting as a master or a slave.

table=1..8 (applies to master only)
Defines which table is used to route messages to their destination. This option applies only to master-attached devices.

protocol={modbusrtu|modbusascii}
The protocol being used by the serial device. The serial protocol also affects the implied incoming network master on TCP and UDP ports 2101.

modbusrtu
Modbus/RTU – 8-bit binary per www.modbus-ida.org specification.

modbusasscii
Modbus/ASCII – 7-bit ASCII per www.modbus-ida.org specification.

messagetimeout=100-99999 ms (applies to master only)
When messages are received from remote clients, this option defines the time to allow the message to be answered. This includes both the queuing and slave response delays, and this should be set to slightly less than the timeout of the remote client. After this time, the Digi device assumes the remote client no longer wants a response. The range is 100 to 99999 milliseconds. The default is 2500 milliseconds.

slavetimeout=10-99999 ms (applies to slave only)
After all bytes of the message have been sent to the slave device, this is the time to wait for the first byte of a response. Note that the serial shift times are not included within this timeout. The range is 10 to 99999 milliseconds. The default is 1000 milliseconds.

chartimeout=3-99999 ms (applies to master or slave)
After a first byte is received, this is the time to wait for additional bytes to either consider the message complete or to be aborted. The default is protocol-specific, but usually 50 milliseconds.

idletimeout={0=disabled|1-99000 seconds}
The device aborts a connection on the implied incoming master sockets after the remote client has been idle for this time. The time is saved in seconds, and the best use for this timeout is to speed up fault recovery. For example, many wide-area networks can suffer shutdowns without the Digi device detecting it. Using the idle timeout speeds up detection of lost TCP connections. The range is 1 to 99999 seconds. The default is 5 minutes.

priority={high|medium|low}

Normally messages are processed in a fair round-robin scheme. This becomes unfair when one master acts as many – for example opening 16 TCP sockets to talk to 16 slaves contrasted to a second master using a single TCP socket to talk to 16 slaves. In this situation, the device assumes it has 17 masters and in effect the first master will have 16 requests answered for every one the second master succeeds in getting answered. This option can be used to adjust the handling of serial master requests. For example, set the serial master to High and the network masters to medium. The effectiveness of this option depends on the protocol behavior – so while some Modbus systems will find it useful, it has no effect on most Rockwell protocols.

high

A high-priority master can get up to 50 percent of the bandwidth – of course you cannot have too many high-priority masters. All high-priority masters with queued messages get one message serviced before any low or medium priority masters get any service.

medium

If a high-priority master has queued messages, then one medium-priority master gets one message serviced before all the high-priority masters are offered service again. If only medium-priority masters exist (which is the default setting), then all masters are serviced in a round-robin manner.

low

Low-priority masters only get service when no High- or Medium-priority master has messages to service.

The default is “medium.”

[*modbus options*]

The configuration options specific to the Modbus protocol, which include:

errorresponse={on|off}

Controls behavior for common run-time errors such as no response from the slave device. By default, “errorresponse=off” for serial Modbus protocols, since most masters assume no response when errors occur. Having this option off also actively filters out returning Modbus exception codes 0x0A and 0x0B from remote Modbus/TCP slaves.

broadcast={on|off|replace}

Specifies how to handle incoming requests with a slave address set to the broadcast value. For Modbus, this is 0. The default is to replace 0 with 1, which was selected to overcome the fact that many Modbus/TCP clients always send requests to unit ID zero (0) and do not want this treated as a broadcast.

on

Tells the Digi device to send requests as broadcast to the destination device(s) and not expect any response message.

off

Tells the Digi device to throw away the broadcast request.

replace

Changes a broadcast request to a normal request by replacing the unit ID 0 with a value of 1.

fixedaddress={auto|1-255}

Used to override the Modbus protocol address (unit ID) with a fixed address.

auto

When set to “auto,” the protocol address will not be overwritten.

1-255

Setting it to a fixed number from 1-255 forces this value to be used for all Modbus requests.

The default setting is “auto.”

Options for configuring network-based masters (set ia master)

master=*range*
Specifies the index of the network master to which the master options apply.

state={on|off}
Enables the IA network master settings.

active={on|off}
Enables or disables the network listener that accepts network connections.

type={tcp|udp}
Defines whether the incoming connection is TCP (connected) or UDP (unconnected). The default is “tcp.”

For cellular connections, using UDP/IP can cut 40% to 60% from your monthly bill.

ipport=*ip port*
Defines the UDP or TCP port on which to listen for protocol messages. Modbus/TCP defaults to TCP port 502.

table=1..8
Defines which table is used to route messages to their destination. This option applies only to master-attached devices.

protocol={modbusrtu|modbusascii|modbustcp}
The protocol used for the connection.

modbusrtu
Modbus/RTU – 8-bit binary per www.modbus-ida.org specification.

modbusascii
Modbus/ASCII – 7-bit ASCII per www.modbus-ida.org specification.

modbustcp
Modbus/TCP – or “Open Modbus” per www.modbus-ida.org specification. Can be enabled with UDP/IP as well as TCP/IP.

messagetimeout=100-99000 ms
When messages are received from remote clients, this is the time to allow the message to wait to be answered. This includes both the queuing and slave response delays, and this should be set to slightly less than the timeout of the remote client. After this time, the Digi device assumes the remote client no longer wants a response. The range is 100 to 99000 milliseconds. The default is 2500 milliseconds.

chartimeout=3-99000 ms
After a first byte is received, this is the time to wait for additional bytes to either consider the message complete or to be aborted. The default is protocol-specific, but usually 50 milliseconds.

idletimeout={0=disabled|1-99000 seconds}

The device aborts a connection after the remote client has been idle for this time. The time is saved in seconds, and the best use for this timeout is to speed up fault recovery. For example, many wide-area networks can suffer shutdowns without the Digi device detecting it. Using the idle timeout will speed up detection of lost TCP connections. The range is 1 to 99999 seconds. The default is 5 minutes.

priority={high|medium|low}

Normally messages are processed in a fair round-robin scheme. This becomes unfair when one master acts as many – for example opening 16 TCP sockets to talk to 16 slaves contrasted to a second master using a single TCP socket to talk to 16 slaves. In this situation, the device assumes it has 17 masters and in effect the first master will have 16 requests answered for every one the second master succeeds in getting answered. This option can be used to adjust the handling of serial master requests. For example, set the serial master to High and the network masters to medium. The effectiveness of this option depends on the protocol behavior – so while some Modbus systems will find it useful, it has no effect on most Rockwell protocols.

high

A high-priority master can get up to 50 percent of the bandwidth – of course you cannot have too many high-priority masters. All high-priority masters with queued messages get one message serviced before any low or medium priority masters get any service.

medium

If a high-priority master has queued messages, then one medium-priority master gets one message serviced before all the high-priority masters are offered service again. If only medium-priority masters exist (which is the default setting), then all masters are serviced in a round-robin manner.

low

Low-priority masters only get service when no High- or Medium-priority master has messages to service.

The default is “medium.”

Options for configuring destination tables and route entries (set ia table)

The destination table and routes are used by the incoming master connections to select which one of many potential slaves a request is to be answered by.

table=*range*
Selects one of eight possible tables in which to look up forwarding information.

[*table options*]
The configuration options specific to the destination table, which include:

state={on|off}
Enables the IA destination table settings.

name=*string*
A useful name for the destination table. Default names are “table1,” “table2,” etc. This option gives you the option to rename the table for convenience. Note that tables are still handled internally by number.

addroute=*route index*
Inserts a new route at this index of the table. If an existing route occupies this index, it is pushed up to a higher index.

removeroute=*route index*
Destroys the route at this index in the table.

moveroute=*from_route_index*,*to_route_index*
Moves the destination route from one route index to another.

route=*range*
Specifies the index of the route in this table to which the settings apply.

[*route options*]
The configuration options specific to the route table entries in the destination table, which include:

active={on|off}
Enables or disables the route in the table.

connect={active|passive}
Defines whether the Digi device attempts immediately to connect to a remote device (“active”), or waits and only connects on demand (“passive”). The default is “passive.”

protaddr=*protocol address range*
Defines the range of protocol address(es) that will be forwarded to this destination entry in the table. You can specify a single address or an inclusive range. The permitted values are defined by the protocol. The table is scanned from the first route index to the last, stopping at the first route with the appropriate protocol address. So duplicates or overlapping ranges can exist, but the route with the lowest index will be used.

type={discard|ip|mapto|nopath|serial}
Defines the type of destination for this route.

discard
Messages destined for this route entry are discarded without error.

ip
Messages destined for this route entry are forwarded to the entered IP address. If you enter the IP address as 0.0.0.0, the Digi device's IP address is used to fill in the IP address, and the "replaceip" function is applied. For example, if the IP is 0.0.0.0, the Digi device's IP address is 143.191.23.199, and the protocol address of the message is 45, then the remote IP address used will be 143.191.23.45.

mapto
Messages destined for this route entry are reevaluated as-if having the protocol address configured within this entry.

nopath
Messages destined for this route entry are returned to sender with a protocol-defined error message.

serial
Messages destined for this route entry are forwarded to a serial port.

protocol={modbusrtu|modbusascii|modbustcp}
The protocol used for the connection.

modbusrtu
Modbus/RTU – 8-bit binary per www.modbus-ida.org specification.

modbusascii
Modbus/ASCII – 7-bit ASCII per www.modbus-ida.org specification.

modbustcp
Modbus/TCP – or "Open Modbus" per www.modbus-ida.org specification.

Can be enabled with UDP/IP as well as TCP/IP.

port=*serial port*
Defines the serial port for the route table entry. This option applies only if the route "type=serial."

transport={tcp|udp}
Defines whether the outgoing connection is TCP (connected) or UDP (unconnected). The default is "tcp." This option applies only if the route "type=ip."

ipaddress=*ip address*
The destination IP address of the entry. This option applies only if the route "type=ip."

ipport=*ip port*
The UDP or TCP port on which to listen for protocol messages. Modbus/ TCP defaults to TCP port 502. This option applies only if the route “type=ip.”

replaceip={on|add|sub|off}
Specifies whether and how the last octet of the IP address is replaced. This option applies only if the route “type=ip.”

on
The protocol address is used to replace the last octet of the IP address. For example, if the table IP is 192.168.1.75 and the protocol address of this message is 23, the message will be forwarded to the remote IP 192.168.1.23.

add
sub
If the “add” or “sub” value is set, the protocol address is added or subtracted from the final octet of the IP address. In the above example, the result would be 192.168.1.98 or 192.168.1.52, respectively.

off
The last octet of the IP address is not replaced.

The default is “off.”

mapto=*protocol address*
Used for destination entries of type “mapto.” This option defines the protocol address for which to reevaluate this message.

slavetimeout=10-99999 ms
After all bytes of the message have been sent to the slave device, this is the time to wait for the first byte of a response. The range is 10 to 99999 milliseconds. The default is 1000 milliseconds.

chartimeout=3-99999 ms
After a first byte is received, this is the time to wait for additional bytes to either consider the message complete or to be aborted. The default is protocol-specific, but usually 50 milliseconds. In TCP/IP network context, this can be thought of as the re-fragment time between packets.

idletimeout={0=disabled, 1-99999 seconds}
The connection is closed after no new messages have been forwarded to the remote slave (server) for this idle time. The time is saved in seconds. The range is 0 to 99999 seconds, where 0 means never close this connection. The default is 5 minutes.

reconnecttimeout=0-99999 ms
If the connection to the remote node fails and “connect=active,” this time is used to delay attempts to reconnect. The default is 2500 milliseconds.

Examples

Serial port configuration settings

The following “set ia” commands show the default serial-port configuration settings for the Digi Connect WAN IA:

```
set ia serial=1 active=on type=slave protocol=modbusrtu table=1
set ia serial=1 messagetimeout=2500 slavetimeout=1000 chartimeout=20
set ia serial=1 priority=medium idletimeout=0
set ia serial=1 errorresponse=off broadcast=replace fixedaddress=0
set ia table=1 route=1 active=on type=serial protaddr=0-32 port=1
```

Note that future Digi Connect WAN IA firmware releases will expose these configuration settings in the Web user interface.

Enable a Modbus/RTU serial master

To enable a Modbus/RTU serial master instead, change the configuration settings by entering the following “set ia” commands via Telnet. These “set ia” commands change the serial device type from slave to master. In addition, to handle cellular latency, a timeout value of 31 seconds or longer is needed. In addition, the first route, which forwards requests to the serial port, can be turned off.

```
set ia serial=1 type=master messagetimeout=31000
set ia table=1 route=1 active=off
```

Additional examples

For more examples of the “set ia” command, see the application note “Remote Cellular TCP/IP Access to Modbus Ethernet and Serial Devices” (Part Number 90000773) available at available from digi.com Support page: http://www.digi.com/support/ia

See also

- "revert" on page 61. The “revert ia” command options revert any existing IA configuration settings.
- "set profile" on page 172. The “ia” port profile configures a serial port for controlling and monitoring various IA devices and PLCs.
- "set serial" on page 189.
- "show" on page 249 for displaying the current IA configuration settings.
- For more information on Industrial Automation, see the IA application help available at this URL: http://www.digi.com/support/ia

set login

Device support

This command is currently supported in all Digi Connect Family devices except Digi Connect ES, Cellular Family products, and ConnectPort Display.

Purpose

Suppresses the user login for a Digi Connect device.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display the login settings: “set permissions s-login=read”
- For a user to display and set login settings: “set permissions s-login=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

```
set login [suppress={on|off}]
```

Options

suppress={on|off}

Specifies whether the user login is suppressed for the Digi Connect device.

on

Suppress login. The Digi Connect device uses the one-user model, as described in "User Models and User Permissions in Digi Connect Products" on page 14.

off

Do not suppress login. The Digi Connect device uses the two-user model, as described in "User Models and User Permissions in Digi Connect Products" on page 14.

See also

- "revert" on page 61.
- "show" on page 249.
- "User Models and User Permissions in Digi Connect Products" on page 14.

set menu

Devices supported

This command is currently not supported in any Digi Connect Family devices.

Purpose

The “set menu” command is used to create and modify custom menus. For example, you can use custom menus to present a set of actions to users, or display a limited set of commands to users for security purposes.

There are several modes of using the “set menu” command:

- To update global settings for custom menus.
- To add or update custom menus.
- To add or update individual items in a custom menu.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions s-menu=read” to display custom menu settings, and “set permissions s-menu=rw” to display and set custom menu settings. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Updating Global Settings

```
set menu [quit_key=key] [quit_label=string] [previous_key=key]
   [previous_label=string] [presskey_label=string]
```

Adding/Updating Custom Menus

```
set menu range=1-32|name=string [newname=string]
   [title=string] [subtitle=string] [sortby={none|key|label}]
   [columns=1-5] [direction={horizontal|vertical}]
```

Adding/Updating Custom Menu Items

```
set menu range={1-32)|name=string} [item=1-32] [key=key]
   [label=string] [{command=string|submenu=string}]
```

Options

Options for Updating Global Settings

quit_key=*key*

The key and text displayed on the custom menu next to the “quit_label” that allows a user to quit the custom menu and close the associated connection. The menu is closed when this key is pressed by the user. The *key* is either 0-9, a-z, or A-Z. The default is “Q.” The *key* is case insensitive; therefore, the keys “A” and “a“ will act the same.

quit_label=*string*
The text displayed on the custom menu next to the “quit_key” that allows a user to quit and close the custom menu. The string is a short description. If the string contains spaces, enclose it in double quotes. The default is “Quit.”

previous_key=*key*
The key and text displayed on the custom menu next to the “previous_label” that allows a user to return to the previous menu when they have previously selected a submenu. This option is only shown and only valid when the displayed menu is an active submenu such that pressing this key will return the user to the original menu. The default is “R.“ The *key* is case insensitive; therefore, the keys “A” and “a“ will act the same.

previous_label=*string*
The text displayed on the custom menu next to the “previous_key” that allows a user to return to the previous menu when the current menu is a submenu that was displayed when issuing a submenu command on another menu. If the string contains spaces, enclose it in double quotes. The default is “Return to Previous Menu.”

presskey_label=*string*
The text displayed after a user has finished an issued command. After the user selects an option that executes a command, the command output will be displayed until the command completes. At this time, processing will be stopped and this text will be shown to inform the user to press any key in order to continue. After a key is pressed, the custom menu will once again be shown. If the string contains spaces, enclose it in double quotes. The default is “Press any key to continue...”

Options for Adding/Updating Custom Menus

range=1-32
The index of the custom menu to create or update. A maximum of 32 menus can be created, and these menus are assigned index values 1 to 32. The index is solely used as a way to identify a custom menu by direct index without having to type in a complete name. When creating a new menu, both the “range” and the “newname” options must be specified. When updating a custom menu, only the “range” or the “name” option must be specified, but not both.

name=*string*
A short descriptive string used to identify a menu. The string is used only to identify a menu when linking a user or other custom menu to this menu. This option can be used rather than the “range” option when updating a custom menu. If the string contains spaces, enclose it in double quotes.

newname=*string*
A short descriptive string used to identify a menu. This option is used to create a new menu or to change the name of an existing menu. When a new menu is being created, this option must be specified along with the "range" option.

When an existing menu name is being changed, this option must be specified along with either "range" or "name," where "name" is the current name of the menu.

If the string contains spaces, enclose it in double quotes.

title=*string*

subtitle=*string*
The title and optional subtitle for a particular menu. The title is displayed above the custom menu and the subtitle, if specified, is displayed immediately below it. These strings are shown to the user accessing the custom menu as a means to identify, explain, or describe a custom menu. Note that when creating a new custom menu, the "title" option is required. If the string contains spaces, enclose it in double quotes.

sortby={none|key|label}
The method by which menu items for the custom menu are ordered and displayed to the user.

none
Organizes the menu items and display them in the order in which they are defined.

key
Sorts the menu items by the keys assigned to the menu items. The sort is case-insensitive.

label
Sorts the menu items based on an alpha-numeric sort of the labels assigned to the menu label.

columns=1-5
The number of columns to display the menu items in to the user. This option is used in order to help avoid scrolling by the user. For instance, a custom menu with many entries may want to display the menu items over 3 columns whereas a custom menu with only a few menu items will appear better using single column. Note that when using multiple columns an attempt should be made to avoid long menu item labels to help avoid possible horizontal scrolling. The default is 1.

direction={horizontal|vertical}

The direction in which to display and arrange menu items. This option only applies when the value of the “columns” option is more than 1 since a single column has no sense of direction other than vertical.

horizontal

The items will be displayed in order left-to-right first, then vertical.

vertical

The items will be displayed top-to-bottom first, then left-to-right.

In other words, the following graphic on the left is done using horizontal direction while the one on the right uses vertical.

A	B	C	A	D	G
D	E	F	B	E	H
G	H	I	C	F	I

The default is “horizontal.”

Options for Adding/Updating Custom Menu Items

range=1-32

The index of the custom menu to add menu items to. Either the “range” or “name” option can be specified to add or update menu items for a particular menu.

name=*string*

The name of the custom menu to add or update menu items for. This option can be used rather than “range**.”** If the string contains spaces, enclose it in double quotes.

item=1-32

The index of the menu item to add or update. There are a maximum of 32 menu items for a particular custom menu and are indexed 1 to 32.

key=*key*

The key to assign to the menu item. This key is displayed on the custom menu next to the “label” value, and is used to select the particular menu item. The user presses this key to select the corresponding menu item. The key must be unique to the custom menu, so that no two menu items for the same custom menu share the same key. The keys assigned to the global settings for “quit_key” and “previous_key” are reserved, and may not be assigned to a menu item.

label=*string*

The text displayed on the custom menu next to the “key” value that describes the action that the menu item will take. When using multiple columns with a custom menu (specified by the “columns” option), it is in the best interest to keep these strings short in length to avoid scrolling. If the string contains spaces, enclose it in double quotes.

command=*string*
The command that is executed when this menu item is selected. This may be any valid command on the CLI (command line interface). The user accessing the custom menu must have the necessary permissions for the supplied command in order to properly execute the command. Note that this option may not be combined with the “submenu” option. If the string contains spaces, enclose it in double quotes.

submenu=*string*
The menu displayed to the user upon selecting the menu item. Submenus allow multi-level menus to exist, and grouping of information so that a user may access one or more submenus each with their own distinct menu items. The string is the menu name identifying the custom menu to link to. The menu must have already been created. Note that this option may not be combined with the “command” option. If the string contains spaces, enclose it in double quotes.

Examples

Updating Global Settings

```
#> set menu quit_key=Q quit_label=”Quit” previous_key=R previous_label=”Return
to Previous Menu” presskey_label=”Press any key to continue...”
```

Adding/Updating Custom Menus

Adding a new custom menu (implies that index 2 is not yet created):

```
#> set menu range=2 newname=my_menu title=”My Menu” sortby=key columns=2
direction=horizontal
```

Updating an existing menu with a new menu name:

```
#> set menu name=my_menu newname=admin_menu
```

Updating an existing menu with new settings:

```
#> set menu name=admin_menu title=”Administration Menu” subtitle=”Select an
Option”
```

Adding/Updating Custom Menu Items

Adding a menu item for a custom menus (implies that index 1 is not yet created):

```
#> set menu name=admin_menu item=1 key=1 label=”Connect Port 1”
command=”connect 1”
```

Updating a menu item to display another submenu:

```
#> set menu name=my_menu item=2 key=A label=”Go to Admin Menu”
submenu=”admin_menu”
```

See also

- "set user" on page 220.
- "revert" on page 61.
- "show" on page 249.

set mgmtconnection

Devices supported

This command is supported in all products except Digi Connect ES and ConnectPort Display.

Purpose

Configures or displays Connectware Manager server connection settings. The Connectware Manager server allows devices to be configured and managed from remote locations. These connection settings set up the connection to the Connectware Manager server so the Digi device knows how to connect to the server.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display the Connectware Device Protocol connection settings: “set permissions s-mgmtconnection=read”
- For a user to display and set Connectware Device Protocol connection settings: “set permissions s-mgmtconnection=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Configure Connectware Manager server connection settings

```
set mgmtconnection [svraddr[1-8]=string]
   [conntype={client|timed|serverinitiated}]
   [connenabled={on|off}]
   [timedperiod=period]
   [timedoffset={immediate|oneperiod|randomtime}]
   [lkaupdateenabled={on|off}]
   [clntreconntimeout={none|timeout]
```

Display Connectware Manager server connection settings

```
set mgmtconnection
```

Options

svraddr[1-8]=*string*

Used to specify one of eight possible Connectware Manager server addresses. When the device server attempts to connect to a Connectware Manager, it tries the server addresses in this list in the order 1-8.

conntype={client|timed|serverinitiated}

Used to specify the connection type.

client

This is a client connection.

timed

This is a timed connection.

serverinitiated

This is a server-initiated connection.

connenabled={on|off}

Used to specify whether or not this instance is enabled for use.

on
Enables this instance for use.

off
Disables this instance for use.

timedperiod=*period*
For a timed connection, this option is used to specify the time interval in minutes between the device server's attempts to connect to the Connectware Manager server. If a device server is already in a connection to a Connectware Manager when the time interval expires, it will not start a new connection at that time. Rather, the device server will start a new timed period timer, and it will again check whether it needs to connect to the Connectware Manager when that new timer expires.

timedoffset={immediate|oneperiod|randomtime}
For a timed connection, this option is used to specify when the first timed connection (to a Connectware Manager) should be attempted after the device server boots.

immediate
Attempt to connect immediately.

oneperiod
Wait one full timed period, then attempt to connect.

randomtime
Wait some random interval of time, between 0 and the full timed period, then attempt to connect.immediate

lkaupdateenabled={on|off}
In conjunction with a server-initiated connection, this option enables or disables a connection to a Connectware Manager server to inform that server of the IP address of the device server. This permits the Connectware Manager to connect back to the device server, or to dynamically update a DNS with the IP address of the device.

on
Enables "last known address" connections to the Connectware Manager.

off
Disables "last known address" connections to the Connectware Manager.

clntreconntimeout={none|*timeout*}

Specifies the retry timeout interval, in seconds, for a last-known-address (LKA) update, if the LKA update fails.If and LKA update fails, the interval configured by this option is used as the amount of time to wait before attempting another LKA update. This option is used for both client-initiated and server-initiated connections. The keyword “none” turns off the retry timeout interval feature.

Examples

Set values for the client connection

```
#> set mgmtconnection connenabled=on conntype=client clntreconnecttimeout=50
```

Display current connection settings

```
#> set mgmtconnection
```

See also

- "revert" on page 61.
- "show" on page 249.
- For more information on Connectware Manager, see the *Connectware Manager Operator’s Guide,* and the Connectware Manager online help.

set mgmtglobal

Devices supported

This command is supported in all products except Digi Connect ES and ConnectPort Display.

Purpose

The Connectware Manager server allows devices to be configured and managed from remote locations. This command is used to set or display the Connectware Manager global settings, or revert the device ID to factory settings.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display the Connectware Device Protocol global settings: “set permissions s-mgmtglobal=read”
- For a user to display and set Connectware Device Protocol global settings: “set permissions s-mgmtglobal=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Configure Connectware Manager global settings

```
set mgmtglobal [deviceid={hex string}]
   [rcicompressionenabled={on|off}]
   [tcpnodelayenabled={on|off}]
   [tcpkeepalivesenabled={on|off}]
   [connidletimeout={none|timeout value}]
```

Display Connectware Manager global settings

```
set mgmtglobal
```

Revert the Device ID to factory settings

```
set mgmtglobal revertdeviceid
```

Options

deviceid={*hex string*}

Used to specify the device ID. The device ID is 32 hexadecimal digits, preceded by the characters “0x.”

rcicompressionenabled={on|off}

Configures whether RCI command and response text is compressed, when both are passed between the Digi device and the Connectware Manager server. This compression primarily affects the size of the data passed when settings or state information are formatted as RCI and conveyed between device and server. Using compression on this RCI text can reduce the size of passed data, and, for cellular products, reduce the cost of reading and writing device settings.

When RCI compression is enabled, LIBZ compression is used on RCI command and response text when it is sent between device and server. The Connectware Device Protocol itself internally negotiates whether compression is applied. RCI compression is enabled, or “on” by default to reduce byte count and cost of sending data. As an example of savings, typical cellular router settings will compress to about 8% of its original size, which means that data can be sent in far fewer packets and less time, than when the uncompressed version of the same data is sent.

The default is “on.” The ability to turn off RCI compression off is provided for technical support/troubleshooting purposes; for example, if you want to eliminate the possibility that this compression is causing some sort of problem.

tcpnodelayenabled={on|off}

Configures whether use of the TCP NODELAY option is disabled by default for the Connectware Manager connection between device and server, when configuring the device's TCP socket endpoint for that connection.

The default is “off.” This default reduces the number of packets sent when the Connectware Manager connection is established between device and server. While there is a very slight penalty in terms of added latency, that penalty is very small compared to the relative high latencies for cellular network communications. Reducing the packet count reduces the number of bytes exchanged over the cellular connection, which saves money. The typical start-up data count is reduced from about 7KB to 4KB just by disabling TCP NODELAY.

The ability to turn on the TCP NODELAY option is provided for technical support/troubleshooting purposes.

tcpkeepalivesenabled={on|off}

Enables or disables sending of TCP keep-alive packets over the client-initiated connection to the Connectware Manager server, and whether the device waits before dropping the connection. The default is “on.”

TCP keep-alives are performed at the TCP protocol level. The application (Connectware Manager in this case) that is using that connection does not know anything about when the TCP keep-alives are sent or received. The TCP keep-alives simply serve to keep each end of the TCP connection aware that the connection is still viable, and intermediate network equipment (NATs in particular) is also made aware that the connection is still good.

connidletimeout={none|*timeout value*}

Enables or disables the idle timeout for the Connectware Manager connection between device and server. Specifying “none” disables the idle timeout. Specifying a timeout value enables the idle timeout, which means the connection will be dropped, or ended, after the amount of time specified. The default is “on.” The minimum value is 300 and the maximum 43200.

In contrast to TCP keep-alives, the timeout managed by the "connidletimeout" option is at the Connectware Manager application level. The "connidletimeout" option provides a way for the connection to the Connectware Manager server to be closed if no Connectware Manager protocol data is sent or received for some period of time. This capability is particularly useful for server-initiated connections. When a user at the server side requests that a connection be established to a device, that user needs to explicitly terminate the connection when they are done with the device. This timeout permits a way to configure the device such that a "forgetful user" does not inadvertently leave the connection in place, which could cost money on a cellular connection if Connectware or TCP keepalives are enabled and transferred needlessly between device and server.

revertdeviceid

Reverts the device ID to factory settings. If the device’s MAC address is GG:HH:JJ:KK:LL:MM, then the device ID is set to 0x0000000000000000GGHHJJ****KKLLMM.

Examples

Set the device id

```
#> set mgmtglobal deviceid=0x0123456789abcdef0123456789abcdef
```

See also

- "revert" on page 61.
- "show" on page 249.
- For more information on Connectware Manager, see the *Connectware Manager Operator’s Guide,* and the Connectware Manager online help.

set mgmtnetwork

Devices supported

This command is supported in all products except Digi Connect ES and ConnectPort Display.

Purpose

The Connectware Manager server allows devices to be configured and managed from remote locations. The “set mgmtnetwork” command configures the network settings for the Digi device’s connection to the Connectware Manager server so the device knows how to connect to the server.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display the Connectware Device Protocol network settings: “set permissions s-mgmtnetwork =read”
- For a user to display and set Connectware Device Protocol network settings: “set permissions s-mgmtnetwork =rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Configure Connectware Manager network settings

```
set mgmtnetwork
   [networktype={modemppp|ethernet|802.11}]
   [connectionmethod={auto|none|mt|mdh|proxy}]
   [proxyaddress=string]
   [proxyport=port]
   [proxylogin=string]
   [proxypassword=string]
   [proxypersistentconnection={on|off}]
   [mtrxkeepalive=time]
   [mttxkeepalive=time]
   [mtwaitcount=count]
   [mdhrxkeepalive=time]
   [mdhtxkeepalive=time]
   [mdhwaitcount=count]
```

Display Connectware Manager network settings

```
set mgmtnetwork
```

Options

[networktype={modemppp|ethernet|802.11}

The type of network to which this command applies.

modemppp

A modem PPP network.

ethernet

An Ethernet network.

802.11

An 802.11 network.

connectionmethod={auto|none|mt|mdh|proxy}
The Connectware Device Protocol firewall traversal method.

auto
Automatically detect the connection method.

none
No firewall; connect using TCP.

mt
Connect using TCP.

mdh
Connect using HTTP.

proxy
Connect using HTTP over proxy.

proxyaddress=*string*
The proxy host address when the connection method is “proxy.”

proxyport=*port*
The proxy host port when the connection method is “proxy.”

proxylogin=*string*
The login string when the connection method is “proxy.”

proxypassword=*string*
The proxy password when the connection method is “proxy.”

proxypersistentconnection={on|off}
Whether the device server should attempt to use HTTP persistent connections when the connection method is “proxy.” Not all HTTP proxies correctly handle HTTP persistent connections. The use of persistent connections can improve performance of the exchange of messages between the device server and Connectware Manager, when that connection is HTTP/proxy. The reason for this is that the same HTTP connection can be reused for multiple consecutive HTTP requests and replies, eliminating the overhead of establishing a new TCP connection for each individual HTTP request/reply, then closing that connection when the request is complete.

on
The device server should attempt to use HTTP persistent connections.

off
The device server should not attempt to use HTTP persistent connections.

mtrxkeepalive=*time*
The transmit keep alive time when connection method is “mt,” where *time* is the number of seconds to wait between sending keep-alive messages.

mttxkeepalive=*time*
The receive keep alive time when connection method is “mt,” where *time* is the number of seconds to wait for a keep-alive message from the Connectware Manager server before assuming the connection is lost.

mtwaitcount=*count*
Used to specify the wait count when the connection method is “mt,” where *count* is how many timeouts occur before the Digi device assumes the connection to the Connectware Manager server is lost and drops the connection.

mdhrxkeepalive=*time*
Used to specify the transmit keep alive time when the connection method is “mdh,” where *time* is the number of seconds to wait between sending keep-alive messages.

mdhtxkeepalive=*time*
Used to specify the receive keep alive time when the connection method is “mdh,” where *time* is the number of seconds to wait for a keep-alive message from the Connectware Manager server before assuming the connection is lost.

mdhwaitcount=*count*
Used to specify the wait count when the connection method is “mdh,” where *count* is how many timeouts occur before the Digi device assumes the connection to the Connectware Manager server is lost and drops the connection.

Examples

Set instance 1 for proxy connection

```
#> set mgmtnetwork connectiontype=modemppp connectionmethod=proxy
proxyaddress=”What goes here?” proxyport=40002 proxylogin=”johnsmith”
proxypassword=”testpass” proxypersistentconnection=off
```

Set instance 2 for mdh connection

```
#> set mgmtnetwork connectiontype=ethernet connectionmethod=mdh
mdhrxkeepalive=100 mdhtxkeepalive=110 mdkwaitcount=15
```

Display current Connectware Device Protocol network settings

```
#> set mgmtnetwork
```

See also

- "revert" on page 61.
- "show" on page 249.
- For more information on Connectware Manager, see the *Connectware Manager Operator’s Guide,* and the Connectware Manager online help.

set nat

Devices supported

This command is supported in Digi Cellular Family products only.

Purpose

Used to set or display Network Address Translation (NAT) and port/protocol forwarding settings.

Note that at this time, the only IP protocols for which protocol forwarding is supported are:

- Generic Routing Encapsulation (GRE, IP protocol 47)
- Encapsulating Security Payload (ESP, IP protocol 50, tunnel mode only).

Port forwarding is supported for the TCP and UDP protocols.

You can forward a single port or a range of ports. To forward a range of ports, specify the number of ports in the range using “pocount” option in the port forwarding entry.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display the NAT and port/protocol forwarding settings: “set permissions s-router=read”
- For a user to display and set the NAT and port/protocol forwarding settings: “set permissions s-router=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Set NAT and port/protocol forwarding settings

```
set nat [enabled={on|off}]
   [maxentries=64-1024]]
   [prenabled[1-4]={on|off}]
   [prnumber[1-4]={gre|esp}]
   [prtype[1-4]=type]
   [prip[1-4]=ipaddress]
   [poenabled[1-64]={on|off}]
   [poproto[1-64]={tcp|udp}]
   [pocount=[1-64]=number of ports in range, minimum 1]
   [poexternal[1-64]=number of ports in range, minimum 1]
   [pointernal[1-64]=number of ports in range, minimum 1]
   [poip[1-64]=ipaddress]
```

Display NAT and port/protocol forwarding settings

```
set nat
```

Options

enabled={on|off}
Enables or disables NAT. Note that IP forwarding must be enabled by the "set forwarding" command for NAT to work.

on
Enable NAT.

off
Disable NAT.

maxentries=64-1024
The maximum number of concurrent NAT table entries that the device will support. This setting effectively limits the number of concurrent NAT rules and sessions that are permitted before disallowing them for resource constraint purposes. The maximum entries can range from 64 through 1024. The default is 256.

prenabled[1-4]={on|off}
Enables one of the four protocol-forwarding entries.

on
Enable this protocol-forwarding entry.

off
Disable this protocol-forwarding entry.

prnumber[1-4]={gre|esp}
The IP protocol whose packets will be forwarded for this entry.

gre
Indicates that the Generic Routing Encapsulation (GRE) protocol will be forwarded.

esp
Indicates that the Encapsulating Security Payload (ESP) protocol will be forwarded.

At this time, GRE and ESP (tunnel mode only) are the only protocols supported by the protocol-forwarding feature.

prtype[1-4]=*type*
This option is deprecated and unused by the device.

prip[1-4]=*ipaddress*
The IP address to which GRE packets will be forwarded.

poenabled[1-64]={on|off}
Used to enable or disable one of the 64 port forwarding entries.

on
Enable this port forwarding entry.

off
Disable this port forwarding entry.

poproto[1-64]={tcp|udp}
The IP protocol associated with this port forwarding entry.

tcp
A TCP port is forwarded.

udp
A UDP port is forwarded.

pocount=[1-64]=*number of ports in range, minimum 1*
The number of consecutive ports in a port-forwarding range. This option allows you to forward more than one port in a single port-forwarding entry. When a range is configured, the first port in the range is specified, and the full range is indicated in the displayed entry information. The default is 1.

poexternal[1-64]=*number of ports in range, minimum 1*
The external (or public) port that will be forwarded for this entry.

pointernal[1-64]=*number of ports in range, minimum 1*
The internal (or private) port to which packets will be forwarded for this entry. This value is a port number on the host whose IP address is specified by the “poip” option value for this entry.

poip[1-64]=*ipaddress*
The IP address of the host to which packets will be forwarded for this entry.

Examples

Enable NAT and specify settings for port forwarding entry 1

This example command will enable the forwarding of TCP packets received at port 4009 of the public (PPP) interface of the device server, to TCP port 7008 of the host whose IP address is 143.191.1.228 on the Ethernet side of the device server.

```
#> set nat enabled=on poenabled1=on poproto1=tcp poexternal=4009
pointernal=7008 poip=143.191.1.228
```

Display NAT and port/protocol forwarding settings

```
#> set nat
```

See also

- "set forwarding" on page 109.
- "revert" on page 61.
- "show" on page 249.

set network

Devices supported

This command is supported in all Digi Connect products.

Purpose

Used to set general network configuration options and display current network configuration options, including options for IP address settings, TCP keep-alives, retransmissions of TCP packets, and ARP settings.

The “garp” and “rto_min.” options can be used for optimizing for latency at the network level.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions s-network=read” to display network configuration attributes, and “set permissions s-network=rw” to display and set network configuration attributes. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Set network configuration options

```
syntax: set network [ip address options]
   [TCP keepalive options]
   [TCP retransmit options]
   [ARP options]
```

Where:

```
[ip address options]:
   [ip=device ip address]
   [submask=subnet mask]
   [gateway=gateway ip address]
   [static={on|off}]
   [dhcp={on|off}]
   [autoip=(on|off)
   [dns1=primary dns server ip address]
   [dns2=secondary dns server ip address]

[TCP keepalive options]:
   [idle=10-86400] (seconds)
   [probe_count=5-30]
   [probe_interval=10-75] (seconds)

[TCP retransmit options]:
   [rto_min=30-1000] (milliseconds)
   [rto_max=1-240] (seconds)

[ARP options]:
   [arp_ttl=1-20] (minutes)
   [garp=30-3600] (seconds)
```

Display current network configuration options

```
set network
```

Options **[*ip address options*]**

Set IP address-related options for the Digi device, including:

ip=*device ip address*
Sets the device IP address when DHCP is off. This option is only applicable if the “static” option is set to “on.”

gateway=*gateway ip*
Sets the network gateway IP address.

submask=*device submask*
Sets the device submask address when DHCP is off. This option is only applicable if the “static” option is set to “on.”

The following three IP address options have a precedence. That is, if all three options are turned on, the order of precedence is: “static,” “dhcp,” “autoip.”

static={on|off}
When enabled, the device uses the specified IP address, gateway address, and submask. The default is off.

dhcp={on|off}
When enabled, the device attempts to use the DHCP protocol to find an IP address, gateway address, and submask. The default is “on.”

The "dhcp" option is enabled by default in almost all Digi Connect devices, except "static" is enabled by default for these: All Digi Connect WAN products except Digi Connect WAN IA (which is DHCP default) and ConnectPort WAN VPN.

autoip={on|off}
When enabled, the device attempts to use the Auto IP protocol to find an IP address, gateway address, and submask. The default is “on.”

dns1=*primary dns server ip address*
dns2=*secondary dns server ip address*
For DNS, these options specify the DNS nameservers to use. Name lookups will be performed using the nameserver specified on “dns1” first, and if that fails, the nameserver specified on “dns2” will be used.

[*TCP keepalive options*]

Are options that configure how TCP keep-alive probes are sent.

The keep-alive options (“idle,” “probe_count,” “probe_interval”) should be configured for various services that are configured by “set service keepalive={on|off},” or clients such as autoconnect (“set autoconnect keepalive={on|off}”).

idle=10-86400
The amount of time, in seconds, to wait while not receiving TCP packets before sending out a keep-alive probe.

probe_count=5-30
The number of TCP keep-alive probes (specially formatted TCP frames) to send out before closing the TCP connection.

probe_interval=10-75
The amount of time, in seconds, to wait between sending TCP keep-alive probes.

[*TCP retransmit options*]
Options that control retransmission of TCP packets, including:

rto_min=30-1000 (milliseconds)
The lower bound or threshold for the TCP retransmission timeout (RTO), in milliseconds. The default is 1000 milliseconds.

TCP uses progressively larger retransmit values, starting at a minimum value that is calculated from a sliding window of ACK response round-trip times that is bounded at the bottom by “rto_min.” So, essentially, “rto_min” is not necessarily the timeout that will be used as the starting retransmit timeout, but it is the smallest such value that could be used.

This affects latency, because lowering “rto_min” ensures that retransmits take place in less time if they occur. By occurring sooner, the network is able to recover the lost data in less time at the expense of possibly retransmitting data that is still in-flight or successfully received by the other side, but unacknowledged due to a “delayed ACK” mechanism or something similar. Choosing a value lower than the default of 1000 milliseconds may help achieve improved latency performance when retransmissions occur.

rto_max=1-240 (seconds)
The upper bound or threshold for the TCP retransmission timeout (RTO), in seconds. When one side of a TCP connection sends a packet and does not receive an acknowledgment from the other side within the timeout period, the sending station retransmits the packet and sets an exponential backoff timeout. This is done for each successive retransmit until the maximum retransmission timeout is reached. Then, the TCP connection resets.

[*ARP options*]

Are options that control Address Resolution Protocol (ARP) requests.

arp_ttl=1-20 (minutes)

The initial value of the ARP time-to-live variable, which is the amount of time that an ARP entry remains in the network ARP cache. When an ARP cache entry first populated, the ARP time-to-live variable is set to this value. When the entry has existed in the table for this long without being updated, another ARP cache request is performed to make sure that there is not a new a new device at that IP.

garp=30-3600 (seconds)

The frequency of Gratuitous ARP (GARP) announcements. A Gratuitous ARP is a broadcast announcement to the network of a device's MAC address and the IP address being used for it. This allows the network to update its ARP cache tables without performing an ARP request on the network.

Gratuitous ARP announcements can affect latency in a limited way, because some systems stall or dispose of data that is transmitted during an ARP cache refresh. If this happens, setting the Gratuitous ARP frequency to be more often than the problem system's time-to-live variable can cause it to refresh the cache without needing to perform a request.

Examples

Manually set the device IP address

```
#> set network ip=10.0.0.1 gateway=255.255.255.0 submask=255.255.255.0
dhcp=off static=on autoip=off
```

Use DHCP to find an IP address, gateway address, and submask

```
#> set network static=off dhcp=on
```

Use DHCP or the Auto IP protocol to automatically configure network settings

```
#> set network static=off dhcp=on autoip=on
```

See also

- "revert" on page 61.
- "set autoconnect" on page 81.
- "set dhcpserver" on page 96.
- "set service" on page 191.
- "set wlan" on page 241.
- "show" on page 249.

set passthrough

Devices supported

This command is supported in Digi Cellular Family products only.

Purpose

Configures the IP pass-through feature. IP pass-through allows a Digi Cellular Family device to provide bridging functionality similar to a cable or DSL modem, where the Digi Cellular Family device becomes “transparent” to the router or connected device. In this case, the router’s WAN interface believes it is connected directly to the mobile network, and has no knowledge that the Digi Cellular Family device is the mechanism providing that connectivity.

A Digi Cellular Family device configured for IP pass-through, such as a ConnectPort WAN or Digi Connect WAN, passes its mobile IP address directly through and to the Ethernet device (router or PC) to which it is connected through the Ethernet port. From the perspective of the connected device, the Digi Cellular Family device essentially becomes transparent (similar to the behavior of a cable or DSL modem) to provide a bridge from the mobile network directly to the end device attached to the Digi Cellular Family device.

Since the mobile network address is effectively “passed-through” to the local device connected to the Ethernet port of the Digi Cellular Family device, all network access to it is bypassed, with some specific exceptions.

Services disabled when IP pass-through is enabled

When IP pass-through is enabled, the Digi Cellular Family device effectively disables all router and IP service functionality. Services that are disabled are:

- NAT
- Port Forwarding
- VPN
- DDNS updates
- Socket Tunnel
- Network Services configuration

Services available when IP pass-through is enabled

The Digi Cellular Family device is effectively transparent to all IP activity and network access by other devices, with these exceptions:

- It can be accessed via the serial port for configuration using the command line interface.
- It accepts TCP/IP connections for purposes of configuration by means of a “pinhole” on the mobile interface.
- It can be accessed by other devices on the local Ethernet segment via the default IP address of 192.168.1.1.
- Clients such as SureLink, and client/server services such as Connectware Manager are operational and enabled by default.

Using Pinholes to Manage the Digi Cellular Family Device

IP pass-through uses a concept called *pinholes*. You can configure the Digi Cellular Family device to listen on specific TCP ports, and terminate those connections at the Digi Cellular Family device for purposes of managing it. Those ports are called pinholes, and they are not passed on to the device connected to the Ethernet port of the Digi Cellular Family device. Each pinhole command option specifies whether the network service and port are passed on to the device connected to the Ethernet port of the Digi Cellular Family device, or terminate at the Digi Cellular Family device. Network services or applications and ports that can be configured as pinholes include:

- Telnet network service: for accessing the device through a Telnet login and the command-line.
- SSH network service: for accessing to the device through a Secure Shell (SSH) login and the command-line.
- HTTP network service: for accessing the device through HTTP and the Web user interface.
- HTTPS network service: for accessing to the device through HTTPS and the Web user interface
- SNMP network service: for monitoring and managing the device through SNMP.
- Connectware Manager application (client-initiated connection)
- Digi SureLink application

For more information on the network services, see "set service" on page 191.

Connectware Manager and Digi SureLink applications are automatically set up as pinholes so that they continue to work with the Digi Cellular Family device.

In addition, the Digi Cellular Family device uses a private address on the Ethernet interface strictly for use in configuration or local access. This allows a user on the local network to gain access to the Web user interface or a Telnet session to make configuration changes.

Remote Device Management and IP Pass-through

The Digi Cellular Family device allows you to enable pinholes for specific ports to allow remote users to manage the Digi Cellular Family device from the mobile network or open Internet. The Digi Cellular Family device retains its remote management capabilities using Connectware Manager. The necessary pinholes are automatically defined when the Digi Cellular Family device is configured for IP Pass-through.This provides administrators with the same remote-management capabilities that exist in Digi remote devices.

Using the “set service” command with IP Pass-through

You can use the “set service” command to have a network service terminate both at a port on the Digi Cellular Family device and a different port on the connected device. For example, you could have the Digi Cellular Family device terminate the SSH service on port 2222, and the connected device terminate SSH at port 22. To do so, you would issue a “set service” command to move the SSH server from listening on port 22 to listening on port 222. With such a configuration, both the Digi Cellular Family device an the connected box could respond to SSH.

Syntax

Configure IP pass-through mode

```
set passthrough [state={enabled|disabled}]
   [http={pass|terminate}]
   [https={pass|terminate}]
   [telnet={pass|terminate}]
   [ssh={pass|terminate}]
   [snmp={pass|terminate}]
   [connectware={pass|terminate}]
   [surelink={pass|terminate}]
   [ping={pass|terminate}]
```

Display current IP pass-through mode settings

```
set passthrough
```

Options

state={enabled|disabled}
Enables or disables IP Pass-through.

http={pass|terminate}
Specifies whether the HTTP network service is configured to pass to the connected device or terminate at the Digi Cellular Family device for purposes of managing it, known as a pinhole.

https={pass|terminate}
Specifies whether the HTTPS network service is configured to pass to the connected device (“pass”) or terminate at the Digi Cellular Family device for purposes of managing it, known as a pinhole (“terminate”).

telnet={pass|terminate}
Specifies whether the Telnet network service is configured to pass to the connected device (“pass”) or terminate at the Digi Cellular Family device for purposes of managing it, known as a pinhole (“terminate”).

ssh={pass|terminate}
Specifies whether the SSH network service is configured to pass to the connected device (“pass”) or terminate at the Digi Cellular Family device for purposes of managing it, known as a pinhole (“terminate”).

snmp={pass|terminate}
Specifies whether the SNMP network service is configured to pass to the connected device (“pass”) or terminate at the Digi Cellular Family device for purposes of managing it, known as a pinhole (“terminate”).

connectware={pass|terminate}
Specifies whether the Connectware Manager application is configured to pass to the connected device (“pass”) or terminate at the Digi Cellular Family device for purposes of managing it, known as a pinhole (“terminate”). The default is "terminate."

surelink={pass|terminate}
Specifies whether the SureLink application is configured to pass to the connected device (“pass”) or terminate at the Digi Cellular Family device for purposes of managing it, known as a pinhole (“terminate”). The default is "terminate."

ping={pass|terminate}
Specifies whether ICMP echo (ping) requests pass to the connected device (“pass”) or terminate at the Digi Cellular Family device (“terminate”). The default is “pass.”

See also

- "revert" on page 61.
- "show" on page 249.
- The *Digi Cellular Family User’s Guide’s* section on IP Pass-through settings.
- For descriptions of network services and their default network port numbers, see "set service" on page 191.
- For descriptions of the Connectware Manager application and related settings, see "set mgmtconnection" on page 137, "set mgmtglobal" on page 140, and "set mgmtnetwork" on page 143.

set permissions

Devices supported

This command is supported in all Digi Connect Family devices. However, the extent of its use varies according to the user model implemented in the Digi Connect product. For Digi Connect products with one or two users, this command does not apply. It does apply to Digi products with two or more users. See "User Models and User Permissions in Digi Connect Products" on page 14 for more information.

Use of command options also depends on the features implemented in Digi devices. For example, “s-ethernet” is only supported in the wired devices.

Purpose

Used to set user permissions associated with various services and command-line interface (CLI) commands, or display current permission settings.

Commands without permissions

There are no permissions associated with the following commands:

- close
- exit
- help
- info
- quit

Permissions for the “revert” command

For the “revert” command, the permissions associated with the various “set” commands are used, except for the “revert all” command variant, which uses a different mechanism that bypasses the individual “set” commands.

Permission descriptions

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions s-permissions=read” to display permissions, and “set permissions s-permissions=rw” to display and change permissions. When permissions are set to “set permissions s-permissions=rw,” a user cannot set another user’s permission level higher than their own level, nor can they raise their own permission level.

Syntax

Set permissions

```
set permissions [type={user|group}]
   {id=range|name=string}
   [backup={none|execute}]
   [boot={none|execute}]
   [connect={none|execute}]
   [display={none|execute}]
   [dhcpserver={none|execute}]
   [buffers={none|r-self|read|rw-self|w-self-r|rw}]
   [filesys={none|read|rw}]
   [kill={none|execute}]
   [newpass={none|rw-self|rw}]
   [ping={none|execute}]
   [provision={none|execute}]
   [reconnect={none|execute}]
   [revert-all={none|execute}]
   [rlogin={none|execute}]
   [s-accesscontrol={none|read|rw}]
   [s-alarm={none|read|rw}]
   [s-autoconnect={none|r-self|read|rw-self|w-self-r|rw}]
   [s-bsc={none|read|rw}]
   [s-ddnsupdater=={none|read|rw}]
   [s-dhcpserver={none|read|rw}]
   [s-ekahau={none|read|rw}]
   [s-ethernet={none|read|rw}]
   [s-gpio={none|read|rw}]
   [s-group={none|read|rw}]
   [s-host={none|read|rw}]
   [s-ia={none|read|rw}]
   [s-login={none|read|rw}]
   [s-menu={none|read|rw}]
   [s-mgmtconnection={none|read|rw}]
   [s-mgmtglobal={none|read|rw}]
   [s-mgmtnetwork={none|read|rw}]
   [s-network={none|read|rw}]
   [s-permissions={none|read|rw}]
   [s-pmodem={none|r-self|read|rw-self|w-self-r|rw}]
   [s-ppp={none|read|rw}]
   [s-profile={none|r-self|read|rw-self|w-self-r|rw}]
   [s-rciserial={none|r-self|read|rw-self|w-self-r|rw}]
   [s-router={none|read|rw}]
   [s-rtstoggle={none|r-self|read|rw-self|w-self-r|rw}]
   [s-serial={none|r-self|read|rw-self|w-self-r|rw}]
   [s-service={none|read|rw}]
   [s-snmp={none|read|rw}]
   [s-socket-tunnel={none|read|rw}]
   [s-system={none|read|rw}]
   [s-tcpserial={none|r-self|read|rw-self|w-self-r|rw}]
   [s-term={none|read|rw}]
   [s-udpserial={none|r-self|read|rw-self|w-self-r|rw}]
   [s-user={none|read|rw}]
   [s-vpn={none|read|rw}]
   [s-wlan={none|read|rw}]
   [status={none|read|rw}]
   [telnet={none|execute}]
   [vpn={none|execute}]
```

```
[who={none|execute}]
[webui={none|execute}
```

Display current permission settings

```
set permissions
```

Options

Permission descriptions

Here are the user permissions and their effects on commands.

none
The command cannot be executed.

execute
The command can be executed.

r-self
The user can execute the "display" portions for both commands if the user is logged in on the specified line.

read
The user can execute the "display" portions for both commands for any line.

rw-self
The user can execute the "display" and "set" portions for both commands if the user is logged in on the specified line.

w-self-r
The user can execute the "display" portions for both commands for any line and the "set" portions for both commands if the user is logged in on the specified line.

rw
The user can execute the "display" and "set" portions for both commands for any line.

type={user|group}
Specifies whether the command applies to users or groups. This option defaults to “user.”

id=*range*
Specifies the ID or the range of IDs of the users or groups to be acted on. If omitted, the “name” option must be specified.

name=*string*
Specifies the name of the user or group to be acted on. If omitted, the “id” option must be specified.

backup={none|execute}
Permissions for the “backup” command. (See "backup" on page 18.)

boot={none|execute}
Permissions for the “boot” command. (See "boot" on page 19.)

buffers={none|r-self|read|rw-self|w-self-r|rw}
Permissions for the “display buffers” and “set buffer” commands. (See "display buffers" on page 37 and "set buffer" on page 90.)

connect={none|execute}
Permissions for the “connect” command. (See "connect" on page 22.)

dhcpserver={none|execute}
Permissions for the "dhcpserver" command. (See "dhcpserver" on page 23.)

display={none|execute}
Permissions for the “display” command. (See "display" on page 27.)

filesys={none|read|rw}
Permissions for user access to the Digi Connect product’s file system.

none
The user cannot access the file system.

read
The user can read the file system.

rw
The user can read and write the file system.

kill={none|execute}
Permissions for the “kill” command. (See "kill" on page 49.)

newpass={none|rw-self|rw}
Permissions for the “newpass” command. (See "newpass" on page 51.)

none
The command cannot be executed.

rw-self
The user can set their own password.

rw
The user can set any user’s password.

ping={none|execute}
Permissions for the “ping” command. (See "ping" on page 52.)

provision={none|execute}
Permissions for the "provision" command. (See "provision" on page 53.)

reconnect={none|execute}
Permissions for the “reconnect” command. (See "reconnect" on page 60.)

revert-all={none|execute}
Permissions for the “revert all” command. (See "revert" on page 61.)

Individual “revert” commands are governed by the permissions for that particular command, but “revert all” uses a different mechanism that bypasses the individual commands.

rlogin={none|execute}
Permissions for the “rlogin” command. (See "rlogin" on page 66.)

s-accesscontrol={none|read|rw}
Permissions for the “set accesscontrol” command. (See "set accesscontrol" on page 68.)

s-alarm={none|read|rw}
Permissions for the “set alarm” command. (See "set alarm" on page 70.)

s-autoconnect={none|r-self|read|rw-self|w-self-r|rw}
Permissions for the “set autoconnect” command. (See "set autoconnect" on page 81.)

s-bsc={none|read|rw}
Permissions for the "set bsc" command. (See "set bsc" on page 85.)

s-ddnsupdater={none|read|rw}
Permissions for the "set ddns" command. (See "set ddns" on page 92.)

s-dhcpserver={none|read|rw}
Permissions for the "set dhcpserver" command. (See "set dhcpserver" on page 96.)

s-ekahau={none|read|rw}
Permissions for the “set ekahau” command. (See "set ekahau" on page 104.)

s-ethernet={none|read|rw}
Permissions for the “set ethernet” command. (See "set ethernet" on page 107.)

s-gpio={none|read|rw}
Permissions for the “set gpio” command. (See "set gpio" on page 113.)

s-group={none|read|rw}
Permissions for the “set group” command. (See "set group" on page 115.)

s-host={none|read|rw}
Permissions for the “set host” command. (See "set host" on page 118.)

s-ia={none|read|rw}
Permissions for the "set ia" command. (See "set ia" on page 119.)

s-login={none|read|rw}
Permissions for the "set login" command. (See "set login" on page 131.)

s-menu={none|read|rw}
Permissions for the “set menu” command. (See "set menu" on page 132.)

s-mgmtconnection={none|read|rw}
Permissions for the “set mgmtconnection” command. (See "set mgmtconnection" on page 137.)

s-mgmtglobal={none|read|rw}
Permissions for the “set mgmtglobal” command. (See "set mgmtglobal" on page 140.)

s-mgmtnetwork={none|read|rw}
Permissions for the “set mgmtnetwork” command. (See "set mgmtnetwork" on page 143.)

s-network={none|read|rw}
Permissions for the “set network” command. (See "set network" on page 149.)

s-permissions={none|read|rw}
Permissions for the “set permissions” command itself.

s-pmodem={none|r-self|read|rw-self|w-self-r|rw}
Permissions for the “set pmodem” command. (See "set pmodem" on page 164.)

s-ppp={none|read|rw}
Permissions for the “set pppoutbound” command. (See "set pppoutbound" on page 166.)

s-profile={none|r-self|read|rw-self|w-self-r|rw}
Permissions for the “set profile” command. (See "set profile" on page 172.)

s-rciserial={none|read|rw}
Permissions for the “set rciserial” command. (See "set rciserial" on page 184.)

s-router={none|read|rw}
Permissions for the “set forwarding” and “set nat” commands. (See "set forwarding" on page 109 and "set nat" on page 146.)

s-rtstoggle={none|r-self|read|rw-self|w-self-r|rw}
Permissions for the “set rtstoggle” command. (See "set rtstoggle" on page 187.)

s-serial={none|r-self|read|rw-self|w-self-r|rw}
Permissions for the “set serial” and "set switches" commands. (See "set serial" on page 189 and "set switches" on page 208.)

s-service={none|read|rw}
Permissions for the “set service” command. (See "set service" on page 191.)

s-snmp={none|read|rw}
Permissions for the “set snmp” command. (See "set snmp" on page 198.)

s-socket-tunnel={none|read|rw}
Permissions for the "set socket_tunnel" command. (See "set socket_tunnel" on page 200.)

s-system={none|read|rw}
Permissions for the “set system” command. (See "set system" on page 211.)

s-tcpserial={none|r-self|read|rw-self|w-self-r|rw}
Permissions for the “set tcpserial” command. (See "set tcpserial" on page 212.)

s-term=s-term={none|read|rw}
Permissions for the “set term” command. (See "set term" on page 215.)

s-udpserial={none|r-self|read|rw-self|w-self-r|rw}
Permissions for the “set udpserial” command. (See "set udpserial" on page 216.)

s-user={none|read|rw}
Permissions for the “set user” command. (See "set user" on page 220.)

s-vpn={none|read|rw}
Permissions for the “set vpn” command. (See "set vpn" on page 228.)

s-wlan={none|read|rw}
Permissions for the “set wlan” command. (See "set wlan" on page 241.)

status={none|read|rw}
Permissions for the “status” command. (See "status" on page 254.)

telnet={none|execute}
Permissions for the “telnet,” “mode,” and “send” commands. (See "telnet" on page 255, "mode" on page 50, and "send" on page 67.)

vpn={none|execute}
Permissions for the "vpn" command. (See "vpn" on page 256.)

who={none|execute}
Permissions for the “who” command. (See "who" on page 258.)

webui={none|execute}
Permissions for access to the Web user interface for a Digi Connect device.

none
The user cannot use the Web user interface.

execute
The user can access the Web user interface.

Examples

Set group permissions

```
#> set permissions type=group name=gurus newpass=rw-self s-user=read
```

Set user permissions

```
#> set permissions id=1 newpass=rw s-user=rw s-group=rw
```

See also

- "User Models and User Permissions in Digi Connect Products" on page 14.
- "set user" on page 220.
- "set group" on page 115.
- "show" on page 249.

set pmodem

Devices supported

This command is supported in the following products:

- Connect Family: all products
- Digi Cellular Family: All products except Digi Connect WAN.

Not supported in ConnectPort Display.

Purpose

Used to configure various options for modem emulation over TCP/IP, and display current modem-emulation settings.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display the modem emulation settings for the line on which they are logged in: “set permissions s-pmodem=r-self”
- For a user to display the modem emulation settings for any line: “set permissions s-pmodem=read”
- For a user to display and set the modem emulation settings for the line on which they are logged in: “set permissions s-pmodem=rw-self”
- For a user to display the modem emulation settings for any line, and set modem emulation settings for the line on which the user is logged in: “set permissions s-pmodem=w-self-r”
- For a user to display and set the modem emulation settings on any line: “set permissions s-pmodem=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Configure modem emulation

The connection-type option, "telnet," applies to both incoming and outgoing calls via the pmodem feature.

```
set pmodem port=range
   [state={on|off}]
   [telnet={on|off}]
```

Display modem-emulation settings

```
set pmodem [port=range]
```

Options

port=*range*

Used to specify the serial port. Optional on a single-port device.

state={on|off}

Used to enable or disable modem emulation on a given serial port.

on

Enables modem emulation.

off

Disables modem emulation.

The default is “off.”

telnet

Enables or disables Telnet processing on incoming and outgoing modem-emulation connections.

on

Enables Telnet processing.

off

Disables Telnet processing.

The default is “off.”

Example

```
#> set pmodem port=1 state=on
```

See also

- "revert" on page 61.
- "show" on page 249.
- Chapter 3, "Modem Emulation Commands" for descriptions of Digi-specific commands for modem-emulation configurations.

set pppoutbound

Devices supported

This command is supported in Digi Cellular Family products only.

Purpose

Configures Point-to-Point Protocol (PPP) outbound connections, or displays current PPP outbound settings.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display the pppoutbound settings: "set permissions s-ppp=read"
- For a user to display and set pppoutbound settings: "set permissions s-ppp=rw"

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Configure PPP outbound connections

```
set pppoutbound port=range
   [state={enabled|disabled}]
   [auth_method={none|PAP|CHAP|both}]
   [passive={on|off}]
   [remote_address={negotiated|ip address}]
   [local_address={negotiated|ip address}]
   [address_mask=ip address mask]
   [default_gateway={yes|no}]
   [protocol_compression={on|off}]
   [address_compression={on|off}]
   [header_compression={on|off}]
   [lcp_keepalive={on|off}]
   [lcp_ka_quiet_time=(10-86400 seconds)]
   [lcp_ka_max_missed_replies={(2-255|0=ignore missed replies)}]
   [asyncmap=hex string]
   [chap_id=chap id]
   [chap_key=chap key]
   [pap_id=pap_id]
   [pap_password=pap password]
   [mru=256-1500]
   [mtu=256-1500]
   [n{1-4}=phone_number]
   [redial_attempts=attempts]
   [redial_delay=delay]
   [rx_idle_timeout=timeout]
   [tx_idle_timeout=timeout]
   [init_script=chat script]
   [dial_script=chat script]
   [login_script=chat script]
   [ipcp_dns_enabled={on|off}]
```

Display PPP outbound settings

```
set pppoutbound
```

Options

port=*range*
The physical interface to which the PPP outbound configuration applies. Required.

state={enabled|disabled}
The state of the interface. The default is “disabled.”

auth_method={none|PAP|CHAP|both}
Determines whether authentication is required for outbound PPP connections and, if so, what kind.

none
The remote user does not require PPP authentication.

pap
Password Authentication Protocol (PAP) authentication is required.

chap
Challenge Handshake Authentication Protocol (CHAP) authentication is required.

both
Both CHAP and PAP authentication are required.

The default is “none.” CHAP authentication works between two Digi Connect products. CHAP will be negotiated to PAP for all other connections.

passive={on|off}
Specifies whether the device server waits for the remote system to begin PPP negotiations, or can initiate PPP negotiations on its own.

on
The device server waits for the remote system to begin PPP negotiations.

off
The device server may initiate PPP negotiations.

The default is “off.”

Do not set both sides of a PPP connection to “passive=on.”

remote_address={negotiated|*ip address*}
The address of the peer at the other end of the outbound PPP connection. Either a specific address or the keyword “negotiated” can be specified; “negotiated” means that the address will be accepted from the peer. An IP address of all zeroes is equivalent to specifying the keyword “negotiated.”

local_address={negotiated|*ip address*}
The IP address of the local end of the PPP outbound connection. Either a specific address or the keyword “negotiated” can be specified; “negotiated” means that the address will be accepted from the peer. An IP address of all zeroes is equivalent to specifying the keyword “negotiated.”

address_mask=*ip address mask*
The IP mask to apply to the address specified on the “remote address” and “local address” options. When you specify a specific IP address on the “remote address” and “local address” options, this option modifies the meaning of the IP address for routing purposes. The default is 255.255.255.255.

default_gateway={yes|no}
Selects whether to use the PPP interface as the default route. The default is “no.”

protocol_compression={on|off}
Specifies whether the device server attempts to negotiate protocol compression on PPP connections.

on
The device server attempts to negotiate protocol compression on PPP connections.

off
The device server will **not** negotiate protocol compression.

The default is “on.”

address_compression={on|off}
Specifies whether the device server attempts to negotiate address compression on PPP connections.

on
The device server attempts to negotiate address compression.

off
The device server does **not** attempt to negotiate address compression.

The default is “on.”

header_compression={on|off}
Specifies whether the device server attempts to negotiate IP protocol header compression on PPP connections. This is commonly referred to as Van Jacobsen (VJ) header compression.

on
The device server attempts to negotiate IP protocol header compression.

off
The device server does **not** attempt to negotiate IP protocol header compression.

The default is “on.”

lcp_keepalive={on|off}
Specifies whether the device server sends Link Control Protocol (LCP) echo requests after a “quiet” interval, in order to test the PPP link and/or keep it alive. “Quiet” means not having received any bytes over the PPP link for a specified time interval, which is set by the “lcp_ka_quiet_time” option. In PPP networks that support LCP echoes, an LCP echo reply is returned by the remote end of the PPP connection.

Even if LCP keepalives are disabled in this device (by “lcp_keepalive=off”), the device will still reply to LCP echo request messages it may receive from the remote side of the PPP connection by sending an LCP echo reply message. But the device itself will not originate any LCP echo request messages.

The options are:

on
The device server sends LCP echo requests after a configurable “quiet” interval, set by the “lcp_ka_quiet_time” option.

off
The device server does not send LCP echo requests.

lcp_ka_quiet_time=10-86400 seconds
Specifies the “quiet” interval, in seconds, after which the device server sends an LCP echo request. “Quiet” means not having received any bytes over the PPP link for the interval specified by this option.

lcp_ka_max_missed_replies={2-255|0=ignore missed replies}
Specifies how many consecutive echo replies may be missed before the device server disconnects the PPP link. A value of 0 (zero) specifies that the device server should not act on missed LCP echo replies by disconnecting the PPP link. Note that if bytes of any kind, LCP echo reply or otherwise, are received, the PPP link is deemed to be active, and the “missed LCP echo replies” count is reset to zero.

asyncmap=*hex string*
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 masks most likely used:

FFFFFFFF
Transpose all control characters.

00000000
Transpose none.

000A0000
Transpose Ctrl-Q and Ctrl-S.

chap_id=*chap id*
A character string that identifies the outbound PPP user using CHAP authentication. This is equivalent to a user or login name. The string must be 32 or fewer characters and must be recognized by the peer.

chap_key=*chap key*
A character string that authenticates the outbound 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.

pap_id=*pap id*
A character string that identifies the outbound PPP user using PAP authentication. This is equivalent to a user (or login) name. The string must be 32 or fewer characters and must be recognized by the peer.

pap_password=*pap password*
A character string that authenticates the outbound 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.

mru=256-1500
The maximum received unit (MRU), or frame size, in bytes, to be received from the other end of the PPP connection. This is a negotiated value. The default is 1500 bytes.

mtu=256-1500
The maximum transmission unit (MTU), or frame size, in bytes, to use for this PPP outbound connection. For PPP connections, the MTU is negotiated, so enter 1500, the largest size device server will permit the remote host to send. For PPP users, the range is 128 to 1500 bytes, and the default is 1500 bytes.

n*{1-4}=*phone_number
Up to four phone numbers to dial to request a PPP outbound connection. The phone numbers are dialed sequentially.

redial_attempts=*attempts*
The number of times the firmware will attempt to redial before giving up.

redial_delay=*delay*
The time to wait after an unsuccessful dial attempt.

rx_idle_timeout=*timeout*
The time, in seconds, after which if no data has been received over the link, the PPP connection is disconnected.

tx_idle_timeout=*timeout*
The time, in seconds, after which if no data has been transmitted over the link, the PPP connection is disconnected.

init_script=*chat script*
An initialization script, run once at interface startup. For example:

```
init_script="'' ATZ OK \c"
```

dial_script=*chat script*
A dialing script, used any time a number is dialed for the interface. For example:

```
dial_script="'' ATDT\T CONNECT \c"
```

login_script=*chat script*
A login script, used to log in to the remote system on the other end of the outbound PPP connection. For example:

```
login_script=”ogin: <username> assword: <password>”
```

ipcp_dns_enabled={on|off}
Enables or disables the IPCP (PPP Internet Protocol Control Protocol) acquisition of DNS IP addresses. This option is enabled by default to preserve prior behavior.

See also

- The “display pppstats” command displays the current status of PPP connections. See "Information returned by “display pppstats”" on page 32 for descriptions of the status information.
- "revert" on page 61
- "show" on page 249.

set profile

Devices supported

This command is supported in all Digi Connect products. However, some port profiles are not supported in particular devices, as noted in the “profile” option’s description.

Purpose

Associates a particular port with one of several port configuration profiles, or displays the current port-profile settings.

Port profiles are a defined set of port configuration settings for a particular use. A port profile reconfigures serial-port settings to the necessary default values in order for the profile to operate correctly.

Port-profile configuration is most often performed through the Web user interface for a device. It is not often specified from the command line, but is available if needed.

Digi Connect devices support several port profiles. Following is the complete set of port profiles. The profiles supported on your Digi Connect product may vary.

- Console Management profile: Allows you to access a device’s console port over a network connection.
- Local Configuration profile allows you to connect standard terminals or terminal emulation programs to the serial port in order to use the serial port as a console to access the command line interface.”
- Modem Emulation profile: Allows you to configure the serial port to act as a modem. (Not supported in Digi Connect WAN.)
- RealPort profile: Allows you to map a COM or TTY port to the serial port. (Not supported in Digi Connect WAN)
- TCP Sockets profile: Allows a serial device to communicate over a TCP network.
- Tunneling profile, also known as the Serial Bridge profile: Configures one side of a serial bridge. A bridge connects two serial devices over the network, as if they were connected with a serial cable.
- UDP Sockets profile: Allows a serial device to communicate using UDP. (Not supported in Digi Connect WAN)
- Custom profile: An advanced option to allow full configuration of the serial port. This profile allows you to view all settings associated with the serial port.
- IA profile: Configures the serial port for use in Industrial Automation (ia).

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display the profile settings for the line on which they are logged in: “set permissions s-profile=r-self”
- For a user to display the profile settings for any line: “set permissions s-profile=read”
- For a user to display and set the profile settings for the line on which they are logged in: “set permissions s-profile=rw-self”
- For a user to display the profile settings for any line, and set modem emulation settings for the line on which the user is logged in: “set permissions s-profile=w-self-r”
- For a user to display and set the profile settings on any line: “set permissions s-profile=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Configure port profile settings

```
set profile port=port profile=profile
```

Display current port profile settings for all available serial ports

```
set profile
```

Display current port profile settings for a particular serial port

```
set profile port=port
```

Options

port=*port*

The serial port number or range of serial ports associated with the port profile. Required when configuring port profiles.

profile=*profile*

The port profile to use for the serial port. Required when configuring port profiles. Choosing a particular port profile causes the serial port’s configuration to be reset to defaults, and then for the default settings for that port profile to take effect.

Depending on the port-profile choices available for the device, the value of “profile” can be one of the following:

console_management

Associates the Console Management port profile with the port. Not supported in Digi Connect WAN.

local_config

Associates the Local Configuration port profile with the port.

modem_emulation

Associates the Modem Emulation port profile with the port. Not supported in Digi Connect WAN.

realport

Associates the RealPort port profile with the port.

tcp_sockets
Associates the TCP Sockets port profile with the port.

tunneling
Associates the Serial Bridge port profile with the port.

udp_sockets
Associates the UDP Sockets port profile with the port. Not supported in Digi Connect WAN.

custom
Associates the Custom port profile with the port.

ia
Associates the ia (Industrial Automation) port profile with the port.

The default configuration settings assume Modbus/RTU slaves with addresses 1 to 32 are attached to the serial port. Default port characteristics are 9600:8,N,1. Unit ID zero (0) is auto-mapped to Modbus/RTU slave address 1. The electrical interface is set as EIA-232, 422, or 485 by the four DIP switches on the bottom of the unit.

Example

```
#> set profile port=1 profile=realport
```

See also

- "revert" on page 61.
- "show" on page 249.
- "set ia" on page 119 for a description of the default settings for Industrial Automation.
- For more information on port profiles, see the topic “About Port Profiles" in the *Digi Connect Family User’s Guide*.

set putty

Devices supported

This command is supported in ConnectPort Display devices only.

Purpose

Configures terminal-emulation settings for ConnectPort Display, and displays current terminal-emulation settings.

ConnectPort Display can emulate a terminal connected to a host/server over a serial line or the network. When connected over the network RealPort must be installed on the server. RealPort ports appear to applications on the server as serial ports, but the data is redirected over the network to the terminal. For more information on RealPort, see the *RealPort Installation Guide*.

A ConnectPort Display device can emulate a terminal connected to a host/server. Data sent from the host application is processed and displayed on the terminal screen. A keyboard can also be used. If a keyboard is connected to the terminal, the terminal data is sent to the host application for it to process.

A reboot is required for the terminal-emulation settings to take effect.

Syntax

Set general terminal emulator options

```
set putty [state={on|off}]
   [width={80|132}]
   [height=10-60]
   [hostport={/com/0|/com/1|/vcom/0]
   [keyboardport={/com/0|/com/1}]
   [cursortype={none|block|underline|vertical}]
   [blinkcursor={on|off}]
   [blinktext={on|off}]
   [backspaceisdelete={on|off}]
   [lfimpliescr={on|off}]
   [characterset=host charset]
```

Set key mappings - range required

```
set putty
   [deletekeymaprange=1-32]
   [keymaprange=1-32]
   [inseq=00-FF]
   [outseq=00-FF]
```

Display terminal emulation settings

```
set putty
```

set putty

Options

General terminal emulation options

state={on|off}
Enables or disables the terminal emulator.

width={80|132}
The default width of the terminal, specified as the number of columns of text to display on the terminal emulator. The default width is 80.

height=10-60
The default height of the terminal, specified as the number of rows of text to display on the terminal emulator. The default height is 24.

hostport={/com/0|/com/1|/vcom/0}
Specifies how the terminal emulator connects to a host application, and how it reads input from the host. The terminal emulator reads input from a host application and displays it on the screen. Input can be read over one of the serial ports on the ConnectPort Display, or over the network. Network connections are achieved using Realport.

Valid values are “/com/0” and “/com/1” (serial ports 1 and 2) and “/vcom/0” (network via RealPort). The default is “/com/0.”

When using a network connection, you must install the RealPort driver on the host. This will create a virtual COM port for each serial port on your ConnectPort Display (these are the traditional RealPort COM ports) as well as one additional virtual COM port that can be used for the terminal emulator connection. The host application must be configured to use this additional virtual COM port.

keyboardport={/com/0|/com/1|No Keyboard}
Specifies how a keyboard, if used, is connected to the terminal emulator. Connecting a keyboard is optional. The terminal emulator can read keyboard input from one of the serial ports. Keyboard data is then passed back up to the host application over the host connection.

Valid values are “/com/0” and “/com/1” (serial ports 1 and 2) and No Keyboard. The default is “/com/1.”

In some environments, the keyboard data should not be passed back up to the host application over the host connection. In this case, you can still connect a keyboard to a serial port, and simply treat it like any other serially connected device. To do so, you would configure the terminal emulator to use “No Keyboard” for the Keyboard Connection, and then configure the serial port for the keyboard to use the RealPort port profile. Keyboard data would then be sent to the host system over the standard RealPort COM port. In this case, the host application reads keyboard data from one COM port and writes host data to a different COM port.

cursortype={none|block|underline|vertical}
Specifies how the cursor appears on the terminal emulator display: as a block, an underline, a vertical line, or no cursor.

none
The cursor has no visible display characteristics.

block
The cursor is displayed as a block.

underline
The cursor is displayed as an underline (underscore) character.

vertical
The cursor is displayed as a vertical bar.

The default is “underline.”

blinkcursor={on|off}
Enables or disables blinking of the cursor. The default is “on.”

blinktext={on|off}
Enables or disables the use of blinking text.The terminal emulator can display text that blinks on and off. This setting allows you to turn off blinking text. When blinking text is disabled and the terminal emulator attempts to make some text blink, the text will instead be displayed with a bold background color. The default is “on.”

backspaceisdelete={on|off}
This option allows you to choose which code, ASCII code 8 or 127, is generated and sent to the host when the Backspace key is pressed. On some terminals, pressing the Backspace key sends the same code as Ctrl-H (ASCII code 8). On other terminals, pressing the Backspace key sends ASCII code 127 (usually known as Ctrl-? or Delete), so that the action can be distinguished from Ctrl-H. The default is “on.”

lfimpliescr={on|off}
Specifies whether an LF (Line Feed) character includes an implicit CR (Carriage Return) character.

Most servers send two control characters, CR and LF, to start a new line of the screen. The CR character makes the cursor return to the beginning of the current line of text. The LF character makes the cursor move one line down. Some servers only send LF, and expect the terminal to move the cursor over to the left automatically. If your server does this, you will see a stepped effect on the screen. If this happens, try enabling this setting.The default is “off.”

characterset=*host charset*

The character set for data received from the host. During a session, the terminal emulator receives a stream of 8-bit bytes from the server, and in order to display them on the screen it needs to know the character set in which to interpret these streams of bytes.

There are several character sets from which to choose. A few notable character sets are:

- The ISO-8859 series are all standard character sets that include various accented characters appropriate for different sets of languages.
- The Win125x series are defined by Microsoft for similar purposes. Win1252 is almost equivalent to ISO-8859-1, but contains a few extra characters such as matched quotes and the Euro symbol.
- CP437 contains the old IBM PC character set with block graphics and line-drawing characters. This is also used on MS-DOS systems.
- UTF-8 contains unicode data interpreted as being in the UTF-8 encoding. Not all server applications will support UTF-8.

The default is ISO-8859-1.

The complete list of allowed character sets is:

Character Set name	Description
ISO-8859-1	ISO-8859-1:1998 (Latin-1, West Europe)
ISO-8859-2	ISO-8859-2:1999 (Latin-2, East Europe)
ISO-8859-3	ISO-8859-3:1999 (Latin-3, South Europe)
ISO-8859-4	ISO-8859-4:1998 (Latin-4, North Europe)
ISO-8859-5	ISO-8859-5:1999 (Latin/Cyrillic)
ISO-8859-6	ISO-8859-6:1999 (Latin/Arabic)
ISO-8859-7	ISO-8859-7:1987 (Latin/Greek)
ISO-8859-8	ISO-8859-8:1999 (Latin/Hebrew)
ISO-8859-9	ISO-8859-9:1999 (Latin-5, Turkish)
ISO-8859-10	ISO-8859-10:1998 (Latin-6, Nordic)
ISO-8859-11	ISO-8859-11:2001 (Latin/Thai)
ISO-8859-13	ISO-8859-13:1998 (Latin-7, Baltic)
ISO-8859-14	ISO-8859-14:1998 (Latin-8, Celtic)
ISO-8859-15	ISO-8859-15:1999 (Latin-9, “euro”)
ISO-8859-16	ISO-8859-16:2001 (Latin-10, Balkan)
CP437	CP437 (IBM-437/MS-DOS Latin, United States)
CP850	CP850 (IBM-850/MS-DOS Latin 1, West Europe)
CP1250	Win1250 (Central European)
CP1251	Win1251 (Cyrillic)
CP1252	Win1252 (Western)
CP1253	Win1253 (Greek)
CP1254	Win1254 (Turkish)
CP1255	Win1255 (Hebrew)
CP1256	Win1256 (Arabic)
CP1257	Win1257 (Baltic)
CP1258	Win1258 (Vietnamese)
KOI8-R	
KOI8-U	
Mac Roman	
Mac Turkish	
Mac Croatian	

Character Set name	Description
Mac Iceland	
Mac Romanian	
Mac Greek	
Mac Cyrillic	
Mac Thai	
Mac Centeuro	
Mac Symbol	
Mac Dingbats	
Mac Ukraine	
Mac VT100	
VISCII	
HP ROMAN8	
DEC MCS	
UTF-8	

Key mapping terminal emulation options

Character codes received from a keyboard can be converted to different character codes before being sent to the host. This conversion, known as key mapping, can be useful when you have different types of keyboards that need to be mapped to the same set of character codes.

A key mapping consists of an input sequence of character codes and the output sequence of codes to which they will be converted. Generally, you would specify both the input and output sequences as single character codes, although you can define up to 5 character codes for each. A character code is entered as two hexadecimal digits. For example:

- To convert the ASCII character A to B, you would define the input and output sequences as '41' and '42' respectively, which are the hexadecimal representations of the ASCII characters.
- To convert a code of decimal 10 to 0, you would define the input and output sequences as '0A' and '00', respectively.

Note that character codes are always two hexadecimal digits, which means that leading zeroes must be provided.

A key mapping entry requires a range, specified by “keymaprange,” and at least an input sequence, specified by “inseq.” The output sequence (“outseq”) is optional. When removing a key mapping entry, only “deletekeymaprange” is required.

The keymap entries are held in a table, as are other device settings such as UDP, serial destinations, alarms, etc. When adding a new entry (an “inseq”/”outseq” pair), you specify at what index in the table to add it using “keymaprange.” To delete an entry (or range of multiple entries) you specify the index/range with “deletekeymaprange.”

Note that the Terminal Emulation settings in the Web user interface manages the indexes for you. If you do not want to deal with the key mappings at an index level, you can configure the key mapping through that interface.

Options specified for key mapping include:

deletekeymaprange=1-32
Removes the key mapping entry at the specified index or range of indexes.

keymaprange=1-32
The index/range used when adding new key mapping entries or replacing existing ones.

inseq=00-FF
The input key sequence, specified as two hexadecimal digits.

outseq=00-FF
The output sequence, specified as two hexadecimal digits.

Example

Display current terminal emulation settings

```
#> show putty
Terminal Configuration :

state = on
width = 80
height = 24
hostport = /com/0
keyboardport = /com/1
cursortype = underline
blinkcursor = on
blinktext = on
backspaceisdelete = on
lfimpliescr = off
characterset = ISO-8859-1
Key Map: range              inseq      outseq
           1                A1         F1
           2                A2         F2
           3                A3         F3
```

Configure general terminal emulation settings

Given the above settings, to adjust the screen height and cursor type, you would enter:

```
# set putty height=30 cursortype=vertical
```

Add, replace, and delete entries in the key mapping table

To add/replace the first 3 entries in the table you would use the following commands:

```
#> set putty keymaprange=1 indeq=A1 outseq=F1
#> set putty keymaprange=2 indeq=A2 outseq=F2
#> set putty keymaprange=3 indeq=A3 outseq=F3
```

Now, to delete the first 2 entries:

```
#> set putty deletekeymaprange=1-2
```

You are left with one keymap entry, and it is at index 3, so to delete this last one enter:

```
#> set putty deletekeymaprange=3
```

See also

- "revert" on page 61.
- "set serial" on page 189.
- "set video" on page 225.
- "show" on page 249.
- The *ConnectPort Display User's Guide's* section on configuring terminal emulation settings.

set python

Devices supported

This command is supported in ConnectPort X products only.

Purpose

Configures Python programs to execute when the Digi device boots.

Syntax

```
set python [range=1-4]
   state={on|off}
   command=filename
```

Options

range=1 – 4
The index or indices to view or modify with the command.

state={on|off}
When the state is set to on, the command specified will be run when the device boots.

command=*filename*
The program filename to execute, including any arguments to pass with the program, similar to the arguments for the "python" command. While this option allows for programs to be run from a TFTP server, this use is not recommended. If there are spaces to provide arguments, make sure to wrap the entire command in quotation marks.

See also

- "python" on page 58
- The *Digi Python Programming Guide*.

set rciserial

Devices supported

This command is supported in Connect Family and Digi Cellular Family products. Not supported in ConnectPort Display.

Purpose

Used to:

- Turn on/off RCI serial mode on the first serial port. The RCI serial mode is a mode that allows a configuration file to be loaded over a serial port when the DSR input signal is high.
- Display current RCI serial-mode settings.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display the RCI serial settings for any line: “set permissions s-rciserial=read”
- For a user to display and set the RCI serial settings on any line: “set permissions s-rciserial=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Turn on off RCI serial mode

```
set rciserial [state={on|off}]
```

Display current RCI serial-mode settings

```
set rciserial
```

Options

state

Enables (on) or disables (off) RCI serial mode on the port. The default is “off.”

Example

```
set rciserial state=on
```

See also

- "backup" on page 18.
- "revert" on page 61.
- "show" on page 249.

set realport

Devices supported

This command is supported in all Digi Connect products.

Purpose

Configures and displays RealPort-related settings.

Required permissions

For Digi Connect products with root and non-root (normal) users, the root user can configure RealPort settings. The normal user can display RealPort settings.

Syntax

Configure RealPort settings

```
set realport [keepalive={on|off}]
   [exclusive={on|off}]
```

Display current RealPort settings

```
set realport
```

Options

keepalive={on|off}

Enables or disables sending of RealPort keepalives. RealPort keepalives are messages inside the RealPort protocol, sent approximately every 10 seconds, to tell whoever is connected that the connection is still alive. RealPort keepalives are different from TCP keepalives, which are done at the TCP layer, and configurable. The default is “on.”

As RealPort keepalives generate additional traffic--several bytes every 10 seconds--this option allows you to turn them off. RealPort keepalives may cause issues in environments that are metered for traffic, or that do not require this type of mechanism. In situations such as cellular/mobile wireless communications, when you are paying by the byte, such additional traffic is undesirable when a TCP keepalive can do the same job, and only when the connection is idle.

If you want to have the RealPort keepalive set to “off;” consider using a TCP keepalive instead. This is because if the link is not closed properly, you could end up with your port being “locked up” with a dead TCP session, which is why RealPort keepalives were implemented in the first place.

exclusive={on|off}

Enables or disables exclusive mode for RealPort connections. Exclusive mode allows the Digi Connect device to close an existing RealPort connection and establish a new one immediately upon a new connection request from the same IP address. This mode is useful when using RealPort over wide area networks, which can be unstable and where you are charged by the byte (such as cellular or satellite), and you do not wish to incur costs for keep-alive traffic. Exclusive mode allows your application to retain continuity when temporary, unexpected interruptions in network connectivity occur.

Example

```
#> set realport keepalive=on
```

See also

- "set network" on page 149. The “set network” keepalive options (“idle,” “probe_count,” “probe_interval,” “garbage_byte,” and “override_dhcp”) should be configured for various services that are configured by “set service keepalive={on|off},” or clients such as autoconnect (“set autoconnect keepalive={on|off}”).
- "set service" on page 191.
- "set autoconnect" on page 81.

set rtstoggle

Devices supported

This command is supported in the following products:

- Connect Family: All products.
- Digi Cellular Family: All products except Digi Connect WAN.

Not supported in ConnectPort Display.

Purpose

Used to:

- Enable or disable RTS toggle on a given serial port. RTS toggle is used to raise RTS when sending data.
- Display current RTS toggle settings.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display the RTS toggle settings for the line on which they are logged in: “set permissions s-rtstoggle=r-self”
- For a user to display the RTS toggle settings for any line: “set permissions s-rtstoggle=read”
- For a user to display and set the RTS toggle settings for the line on which they are logged in: “set permissions s-rtstoggle=rw-self”
- For a user to display the RTS toggle settings for any line, and set RCI serial settings for the line on which the user is logged in: “set permissions s-rtstoggle=w-self-r”
- For a user to display and set the RTS toggle settings on any line: “set permissions s-rtstoggle=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Enable or disable RTS toggle

```
set rtstoggle port=range [state={on|off}]
   [predelay=delay] [postdelay=delay]
```

Display current RTS toggle settings

```
set rtstoggle [port=range]
```

Options

port=*range*

Used to specify the serial port. Optional on a single-port device.

state={on|off}

Used to enable or disable the RTS toggle feature.

on

Enables the RTS toggle feature.

off

Disables the RTS toggle feature.

The default is “off.”

predelay=*delay*
Specifies the time in milliseconds to wait after the RTS signal is turned on before sending data. The range is 0 to 5000 milliseconds. The default is 0.

postdelay=*delay*
Specifies the time in milliseconds to wait after sending data before turning off the RTS signal. The range is 0 to 5000 milliseconds. The default is 0.

Examples

```
#> set rtstoggle state=on predelay=10
```

See also

- "revert" on page 61.
- "show" on page 249.

set serial

Devices supported

This command is supported in all Digi Connect products.

Purpose

Sets and displays general serial configuration options, such as baud rate, character size, parity, stop bits, and flow control.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display the serial settings for the line on which they are logged in: “set permissions s-serial=r-self”
- For a user to display the serial settings for any line: “set permissions s-serial=read”
- For a user to display and set the serial settings for the line on which they are logged in: “set permissions s-serial=rw-self”
- For a user to display the serial settings for any line, and set serial settings for the line on which the user is logged in: “set permissions s-serial=w-self-r”
- For a user to display and set the serial settings on any line: “set permissions s-serial=rw”

See "set permissions" on page 157 for details on setting user permissions for commands. Permissions for "set serial" also apply to the "set switches" command. See "set switches" on page 208.

Syntax

Set general serial options

```
set serial port=range
   [altpin={on|off}]
   [baudrate=bps]
   [csize={5|6|7|8}]
   [parity={none|even|odd|mark|space}]
   [stopb={1|2}]
   [flowcontrol={hardware|software|none}]
```

Display current serial options

```
set serial [port=range]
```

Options

port=*range*

Used to specify the serial port. Optional on a single-port device.

altpin={on|off}

Determines whether the altpin option, which swaps DCD with DSR so that eight-wire RJ-45 cables can be used with modems, is used:

on

The altpin option is used.

off

The altpin option is **not** used.

The default is “off.”

baudrate=*bps*
The baud rate in bits per second. The default is 9600.

csize={5|6|7|8}
The character size, which can be 5, 6, 7, or 8 bits. The default is 8.

flowcontrol={hardware|software|none}
Specifies which kind of flow control is used on the line.

hardware
Hardware flow control (RTS/CTS).

software
Software flow control (Xon/Xoff).

none
No flow control.

The default is “software.”

parity={none|even|odd|mark|space}
The parity used for the line.

none
No parity.

even
Even parity.

odd
Odd parity.

mark
Mark parity.

space
Space parity.

The default is “none.”

stopb={1|2}
The number of stop bits per character to use on this line. The value used here must match the setting on the device connected to this port. Use 1 or 2 stop bits.

The default is 1 stop bit.

Example

```
#> set serial baudrate=9600 flowcontrol=hardware
```

See also

- "revert" on page 61.
- "show" on page 249.

set service

Devices supported

This command is supported in all Digi Connect products.

Purpose

Used to:

- Enable and disable network services.
- Change the network port on which a given service listens.
- Display the entire service table, or an entry in the service table.

Caution on enabling and disabling services

Exercise caution in enabling and disabling network services, particularly disabling them. Changing certain settings can render your Digi Connect product inaccessible. For example, if you disable Advanced Digi Discovery Protocol (ADDP), the device will not be discovered on a network, even if it is actually connected. If you disable HTTP and HTTPS, the Web interface can be disabled. Disabling basic services such as Telnet, Rlogin, etc. can make the Command-Line interface inaccessible.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions s-service=read” to display network service settings, and “set permissions s-services=rw” to display and change network service settings. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Enable/disable network services or change network port for service

```
set service [range=range]
   [state={on|off}]
   [ipport=network_port]
   [keepalive={on|off}]
   [nodelay={on|off}]
   [delayed_ack=0-1000]
```

Display service table or entries in the table

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

Options

range=*range*

Used to specify the index of the network service to which the rest of the command’s options apply. For more information on using this option, see "Index numbers and changing default port numbers" on page 196.

state={on|off}

Used to enable or disable a given network service.

ipport=*network port*

Used to change the network port on which a given network service listens. See "Supported network services and their default network port numbers" on page 193 for more information on the network services available.

keepalive={on|off}

Indicates whether or not TCP keepalives will be sent for specified range of network services. If set to on, keepalives will be sent, if it is off, keepalives will not be sent.

Configurable TCP keepalive parameters, for example, how many keepalives to send and when to send them are configured globally via the “set network” command (see "set network" on page 149).

nodelay={on|off}

Used to allow unacknowledged or smaller-than-maximum-segment-sized data to be sent for the specified range of network services.

The “nodelay” option disables Nagle’s algorithm, which is on by default, for some TCP services. The purpose of Nagle's algorithm is to reduce the number of small packets sent. The algorithm establishes not sending outgoing data when there is either unacknowledged sent data, or there is less-than-maximum segment size (typically around 1500 bytes for Ethernet) worth of data to be sent. While this algorithm allows for efficient data transmission, there are times when it is desirable to disable it.

delayed_ack=0-1000

The time, in milliseconds, to delay sending ACK packets in response to received data for the specified range of network services. The default is 200 milliseconds.

Setting this option to 0 (zero) sends an ACK packet back acknowledge the received data immediately. Setting it to any other value means that the ACK packet will be sent after the specified time. If the network services generate new data during that time, the ACK packet will be sent along with the data packet.

You can use this setting to avoid congestion and reduce network traffic, However, do not change this option from its default setting unless you have a solid understanding of network services and data transmission, or have been instructed to make the change.

Supported network services and their default network port numbers

The following table shows the network services controlled by the "set services" command, the services provided, and the default network port number for each service.

In Digi Connect products that have multiple serial ports, the network port number defaults for various services are set based on the following formula:

base network port number + serial port number

For example, the Telnet Passthrough service is set to network port 2001 for serial port 1, 2002 for serial port 2, 2003 for serial port 3, etc.

If you change a network port for a particular service, that is the only network port number that changes. That change does not carry over to the other network ports. For example, if you change the network port number Telnet Passthrough from 2001 to 3001, that does not mean that the other network ports will change to 3002, 3003, etc.

There are two types of network services available:

- Basic services, which are accessed by connecting to a particular well-known network port.
- Passthrough services, in which a particular serial port is set up for a particular type of service. To use the service, users must both use the correct protocol and specify the correct network port. For example, assuming default service ports and using a Linux host, here is how a user would access the SSH and Telnet passthrough services:

```
#> ssh -l fred digi16 -p 2501
#> telnet digi16 2101
```

Service	Services Provided	Default Network Port Number
Advanced Digi Discovery Protocol (ADDP), also known as Device Discovery	Discovery of Digi Connect products on a network.	2362
Encrypted (Secure) RealPort	Secure Ethernet connections between COM or TTY ports and device servers or terminal servers.	1027
Hypertext Transfer Protocol (HTTP), also known as Web Server	Access to web pages for configuration that can be secured by requiring a user login.	80
Hypertext Transfer Protocol over Secure Socket Layer (HTTPS), also known as Secure Web Server	Access to web pages for configuration that can be secured by requiring a user login, with encryption for greater security.	443
Line Printer Daemon (LPD)	Allows network printing over a serial port.	515
Modem Emulation Pool (pmodem)	Allows the Digi Connect product to emulate a modem. Modem emulation sends and receives modem responses to the serial device over the Ethernet instead of Public Switched Telephone Network (PSTN). Telnet processing can be enabled or disabled on the incoming and outgoing modem-emulation connections. The pmodem service is for connecting to whatever serial port will answer.	5000
Modem Emulation Passthrough	Allows the Digi Connect product to emulate a modem. This service is for dialing in to a particular serial port that has been set up for modem emulation.	5001
RealPort	A virtual connection to serial devices, no matter where they reside on the network.	771
Remote login (Rlogin)	Allows users to log in to the Digi Connect device and access the command-line interface via Rlogin.	513
Remote shell (Rsh)	Allows users to log in to the Digi Connect device and access the command-line interface via Rsh.	514
Secure Shell (SSH)	Allows users secure access to log in to the Digi Connect device and access the command-line interface.	22
Secure Shell (SSH) Passthrough	Accessing a specific serial port set up for SSH.	2501
Secure Socket Service	Authentication and encryption for Digi Connect devices.	2601
Simple Network Management Protocol (SNMP)	Managing and monitoring the Digi Connect device. If you want to run SNMP, but in a more secure manner, note that SNMP allows for “sets” to be disabled.This securing is done in SNMP itself, not through this command.	161
Transmission Control Protocol (TCP) Echo	Used for testing the ability to send and receive over a TCP connection, similar to a ping.	7
Telnet	Allows users an interactive Telnet session to the Digi Connect product’s command-line interface.	23

Service	Services Provided	Default Network Port Number
Telnet Passthrough	Allows a Telnet connection directly to the serial port, often referred to as reverse Telnet.	2001
Transmission Control Protocol (TCP) Passthrough	Allows a raw socket connection directly to the serial port, often referred to as reverse sockets.	2101
User Datagram Protocol (UDP) Passthrough	Used for testing the ability to send and receive over a UDP connection, similar to a ping.	7
User Datagram Protocol (UDP) Passthrough	Allows raw data to be passed between the serial port and UDP datagrams on the network.	2101
VNC Client Listen Daemon	Remote access to a computer on the network or internet using the VNC (Virtual Network Computing) protocol. VNC server software must be installed on the remote computer.	5500
VNC Server	Allows users to remotely view what is currently displayed on the screen using a standard VNC client (viewer).	5900

Index numbers and changing default port numbers

An index number is assigned to each of these services. The index numbers assigned can vary over time. If you want to change the network port number for a service, enter a “set service” or “show service” command to display the current index number assigned to all services. Locate the service for which you want to change the network port number, and note the index number for the service. Enter a “set service” command, specify that index number for the “range” option, and the new network port number for the “ipport” option.

For example, to change the network port number for the Telnet basic service from its default port number, 23, you would enter the following “set service” command:

```
#> set service
```

which displays the services defined in and their current network port number assignments:

```
Service Configuration :

 index state ipport keepalive nodelay                          service
     1    on     23       off     off                  Telnet Service
    12    on     80        na      na                    HTTP Service
    13    on    161        na      na                    SNMP Service
     2    on    443        na      na                   HTTPS Service
    15    on    513       off     off                  Rlogin Service
    16    on    514       off     off                     Rsh Service
     9    on    515       off     off             Line Printer Daemon
     8    on    771       off      na                RealPort Service
     3    on   1027       off      na      Encrypted RealPort Service
     4    on   2001       off     off          Telnet Server (Port 1)
     5    on   2101       off     off             TCP Server (Port 1)
     6   off   2101        na      na      Serial/UDP Server (Port 1)
    14    on   2362        na      na                     ADDP Service
     7    on   2601       off     off  Secure Socket Service (Port 1)
    10    on  50000        na      na           Modem Emulation (Pool)
    11    on  50001       off     off        Modem Emulation (Port 1)
```

Note that the index number assigned to the Telnet basic service is 1. You would then specify that index number for the “range” option, and the new network port number for the “ipport” option:

```
#> set service range=1 ipport=100
```

Examples

Disable service

```
#> set service range=1 state=off
```

Change the network port (ipport) of a service

```
#> set service range=1 ipport=500
```

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
```

Allow outgoing data that is unacknowledged or less than maximum segment size

```
#> set service ra=5 nodelay=on
```

See also

- "revert" on page 61.
- "set network" on page 149.
- "set passthrough" on page 153 for information on network services and applications that are enabled and disabled by default when a Digi device is configured for IP passthrough.
- "show" on page 249.
- For descriptions of the Connectware Manager application and related settings, see "set mgmtconnection" on page 137, "set mgmtglobal" on page 140, and "set mgmtnetwork" on page 143.
- For more information on SureLink Link Integrity Monitoring tests, see "set surelink" on page 202.

set snmp

Devices supported

This command is supported in all Connect Family and Digi Cellular Family products. Not supported in ConnectPort Display.

Purpose

Configures the Simple Network Management Protocol (SNMP) agent, or displays current SNMP settings.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions s-snmp=read” to display network service settings, and “set permissions s-snmp=rw” to display and change network service settings. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Set SNMP settings

```
set snmp [trapdestip=ipaddress]
   [publiccommunity=string]
   [privatecommunity=string]
   [setsenabled={on|off}]
   [authfailtrap={on|off}]
   [coldstarttrap={on|off}]
   [linkuptrap={on|off}]
   [logintrap={on|off}]
```

Display SNMP settings

```
set snmp
```

Options

trapdestip=*ipaddress*

Used to configure the IP address of the system to which the agent should send traps. To enable any of the traps, a non-zero value for trapdestip must be specified.

The ”trapdestip” option is required in order for alarms to be sent in the form of SNMP traps. See "send" on page 67.

publiccommunity=*string*

The password required to “get” SNMP-managed objects. The default is “public”.

privatecommunity=*string*

The password required to “set” SNMP-managed objects. The default is “private”.

setsenabled={on|off}

Enables or disables “sets” of SNMP-managed objects.

on

Enables “sets” if the provided private community matches the current private community.

off

Disables “sets” even if the provided private community matches the current private community.

The default is “off.”

authfailtrap={on|off}
Enables or disables the sending of authentication failure traps.

on
Enables the sending of authentication failure traps.

off
Disables the sending of authentication failure traps.

The default is “off.”

coldstarttrap={on|off}
Enables or disables the sending of cold start traps.

on
Enables the sending of cold start traps.

off
Disables the sending of cold start traps.

The default is “off.”

linkuptrap={on|off}
Enables or disables the sending of link up traps.

on
Enables the sending of link up traps.

off
Disables the sending of link up traps.

The default is “off.”

logintrap={on|off}
Enables or disables the sending of login traps.

on
Enables the sending of login traps.

off
Disables the sending of login traps.

The default is “off.”

Examples

Enable authentication failure traps

```
#> set snmp trapdestip=10.0.0.1 authfailtrap=on
```

Specify a new private community string

```
#> set snmp privatecommunity=”StLucia72!”
```

See also

- "revert" on page 61.
- To disable and enable SNMP, use the “set service” command. See "set service" on page 191.
- To disable and enable SNMP alarm traps, see "set alarm" on page 70.

set socket_tunnel

Devices supported

This command is supported in all Digi Connect Family and Digi Cellular Family products. It is not supported in ConnectPort Display products.

Purpose

Configures a socket tunnel. A socket tunnel can be used to connect two network devices: one on the Digi device server's local network and the other on the remote network. This is especially useful for providing SSL data protection when the local devices do not support the SSL protocol.

One of the endpoint devices is configured to initiate the socket tunnel. The tunnel is initiated when that device opens a TCP socket to the Digi device server on the configured port number. The Digi device server then opens a separate connection to the specified destination host. Once the tunnel is established, the Digi device server acts as a proxy for the data between the remote network socket and the local network socket, regardless of which end initiated the tunnel.

The socket tunnel feature is most useful for devices with two interfaces. It could also be used as a connection proxy on a single-interface device, such as the Digi Connect ME. One way the socket tunnel feature would be very useful in a single interface device is when the device has the capability to use specified keys, and other devices connected to it do not have that capability. Using the socket tunnel feature, the device with the key capability basically becomes a security gatekeeper for simple devices that cannot use PKI certificates.

Required Permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions s-socket-tunnel=read” to display socket tunnel settings, and “set permissions s-socket-tunnel=rw” to display and change socket tunnel settings, settings. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Configure a socket tunnel

```
set socket_tunnel [state={disabled|enabled}]
   [timeout={0|seconds}] {0 is no timeout}
   [from_hostname={name|ip address}
   [from_port=port number]
   [from_protocol={tcp|ssl}]
   [to_hostname={name|ip address}]
   [to_port=port number]
   [to_protocol={tcp|ssl}]
```

Display current socket tunnel settings

```
set socket_tunnel
```

Options

state={disabled|enabled
Enables or disables the configured socket tunnel.

timeout={0|*seconds*}] {0 is no timeout}
The timeout (specified in seconds) controls how long the tunnel will remain connected when there is no tunnel traffic. If the timeout value is zero, then no timeout is in effect and the socket tunnel will stay up until some other event causes it to close.

from_hostname={*name*|*ip address*}
The initiating host: the hostname or IP address of the network device that initiates the socket tunnel.

from_port=*port number*
The initiating port: the port number that the Digi device uses to listen for the initial socket tunnel connection.

from_protocol={tcp|ssl}
The initiating protocol: the protocol used between the device that initiates the socket tunnel and the Digi device. Currently, TCP and SSL are the two supported protocols.

to_hostname={*name*|*ip address*}
The destination host: The hostname or IP address of the destination network device.

to_port=*port number*
The destination port: the port number that the Digi device uses to make a connection to the destination device.

to_protocol={tcp|ssl}
The destination protocol: the protocol used between Digi device and the destination device. Currently, TCP and SSL are the two supported protocols. This protocol does not need to be the same for both connections.

See also

- "revert" on page 61.
- "show" on page 249.
- The section on socket tunnel settings in your Digi product's *User's Guide*.

set surelink

Devices supported

This command is supported in Digi Cellular Family products only.

Purpose

Configures Digi SureLink™ settings for a Digi Cellular Family device. Digi SureLink™ provides an “always-on” mobile network connection to ensure that a Digi Cellular Family device is in a state where it can connect to the network. It does this through hardware reset thresholds and periodic tests of the connection.

There are several groups of command options for “set surelink:”

- Hardware reset thresholds: these settings can be configured to clear any error states that were resident in the device's cellular module, so the device can once again connect to the network, if the connection is lost. SureLink does this by first resetting the cellular module after a default or specified number of consecutive failed connection attempts, and then resetting the Digi Cellular device after a default or specified number of failed consecutive connection attempts. Each of these connection-failure settings can be disabled as well.
- Link Integrity Monitoring Tests: Digi SureLink can be configured to run tests, known as Link Integrity Monitoring tests, that are run over the established PPP connection to ensure that end-to-end communication is possible. There are command options that apply to all link testing, and options for the three available Link Integrity Monitoring tests:
  - Ping Test
  - TCP Connection Test
  - DNS Lookup Test

Syntax

Set Hardware reset thresholds

```
set surelink [module_reset_connect_failures={1-255|0=off}]
   [system_reset_connect_failures={1-255|0=off}]
```

Configure Link Integrity Monitoring Tests:

Set general link test options

```
set surelink [state={on|off}]
   [test={ping|tcp|dns}]
   [trigger={interval|idle}]
   [max_consecutive_failures={1-255|0=off}] (probe failures before
   link reset)
   [interval=10-65535]
```

Set ICMP ping link test options

```
set surelink [pingaddr1={ipv4 address|fqdn}]
   [pingaddr2={ipv4 address|fqdn}]
```

Set DNS lookup link test parameters:

```
set surelink [dnsfqdn1=dns fqdn]
   [dnsfqdn2=dns fqdn]
```

Set TCP connection link test parameters:

```
set surelink [ipaddr1={ipv4 address|fqdn}]
   [ipaddr2={ipv4 address|fqdn}]
   [ipport=1-65535]
```

Display current SureLink settings

```
set surelink
```

Options

Options for Hardware reset thresholds

module_reset_connect_failures={1-255|0=off}

The number of failed connection attempts that occur before the cellular modem module is reset. This value can be a number between 1 and 255, or 0, which turns off the cellular modem module-reset feature. The default is 3.

system_reset_connect_failures={1-255|0=off}

The number of failed connection attempts that occur before the Digi Cellular Family device is reset. This value can be a number between 1 and 255, or 0, which turns off the system-reset feature. The default is 0, or off.

Options for Link Integrity Monitoring Tests

General link test options:

state={on|off}

Enables or disables link integrity monitoring tests. If “on,” the other Link Integrity Monitoring settings may be configured and are used to verify the functional integrity of the mobile connection. The default is “off.”

test={ping|tcp|dns}

The Link Integrity Monitoring test to be run.

Each test can be used to demonstrate that two-way communication is working over the mobile connection. This variety of tests is provided because different mobile networks or firewalls may allow or block Internet packets for various services. The appropriate test may be selected according to mobile network constraints and user preference.

The link integrity tests are performed only while the mobile connection is established. If the mobile connection is disconnected, the link integrity tests are suspended until the connection is established again.

For the link integrity tests to provide meaningful results, the remote or target hosts must be accessible over the mobile connection and not through the LAN interface of the device (if it has one). That is, the settings should be configured to guarantee that the mobile connection is actually being tested.

The link integrity test settings may be modified at any time. The changes are used at the start of the next test interval.

ping

Ping test. This test uses “ping” (ICMP) as a test to verify the integrity of the mobile connection. The test is successful if a valid ping reply is received in response to the ping request sent. The ping test actually sends up to three ping requests, at three second intervals, to test the link. When a valid reply is received, the test completes successfully and immediately. If a reply is received for the first request sent, there is no need to send the other two requests.

Two destination hosts may be configured for this test. If the first host fails to reply to all three ping requests, the same test is attempted to the second host. If neither host replies to any of the ping requests sent, the test fails. The primary and secondary addresses may be either IP addresses or fully qualified domain names.

tcp

TCP Connection Test. This test creates a new TCP connection as a test to verify the integrity of the mobile connection. The test is successful if a TCP connection is established to a specified remote host and port number. If the remote host actively refuses the connection request, the test is also considered to be successful, since that demonstrates successful two-way communication over the mobile connection. The TCP connection test waits up to 30 seconds for the connection to be established or refused. When the TCP connection is established, the test completes successfully, and the TCP connection is closed immediately.

Two destination hosts may be configured for this test. If the first host fails to establish (or refuse) the TCP connection, the same test is attempted to the second host. If neither host successfully establishes (or refuses) the TCP connection, the test fails. The primary and secondary addresses may be either IP addresses or fully qualified domain names.

dns

DNS Lookup Test. This test uses a Domain Name Server (DNS) lookup as a test to verify the integrity of the mobile connection. The test is successful if a valid reply is received from a DNS server. Typically, this means the hostname is successfully “resolved” to an IP address by a DNS server. But even a reply such as “not found” or “name does not exist” is acceptable as a successful test result, since that demonstrates successful two-way communication over the mobile connection. When a valid reply is received, the test completes successfully and immediately.

The DNS servers used in this test for the hostname lookup, are the primary and secondary DNS servers obtained from the mobile network when the mobile PPP connection is first established. These addresses may be viewed in your web browser on the Administration | System Information | Mobile page.

Note that this DNS test is independent of the normal DNS client configuration and lookup cache, which is used for other hostname lookups. This test has been specifically designed to require communication over the mobile connection for each lookup, and to avoid being “short-circuited” by previously cached information. Also, this test does not interfere in any way with the normal DNS client configuration of this device.

Two hostnames may be configured for this test. If the first hostname fails to get a reply, the same test is attempted for the second hostname. If no reply is received for either hostname, the test fails. The primary and secondary DNS names should be fully qualified domain names. Note that the reverse lookup of an IP address is possible, but that is usually unlikely to succeed in returning a name. Still, such a reverse lookup can be used to demonstrate the integrity of the mobile connection.

trigger={interval|idle}

The conditions under which link integrity monitoring tests are performed.

interval

Link integrity monitoring tests are repeated at the interval specified by the “interval” option.

idle

Link integrity monitoring tests are performed only when idle; that is, if no data is received for the period of time specified by the “interval” option.

This value changes the behavior of the test, in that the test interval varies according to the presence of other data received from the mobile connection.

Although using "trigger=idle" may result in less data being exchanged over the mobile connection, it also prevents the link integrity tests from running as often to verify the true bi-directional state of that connection.

max_consecutive_failures={1-255, 0=off} (probe failures before link reset)]

The maximum number of consecutive Link Integrity Monitoring tests. After this number is reached, the mobile connection should be disconnected and reestablished.

This value must be between 1 and 255. The default is 3. A value of 0 turns off this feature. When the mobile connection is reestablished, the “consecutive failures” counter is reset to zero.

Note: if the mobile connection is disconnected for any reason (including not as a result of a link integrity test failure), the consecutive failures count is reset to zero when the mobile connection is reestablished.

interval=10-65535

Specifies the interval, in seconds, at which the selected Link Integrity Monitoring test is initiated or repeated. A new test will be started every specified number of seconds while the mobile connection is established. This value must be between 10 and 65535. The default is 240.

If the configured interval is less time than it takes a test to complete, the next test will not be initiated until the previous (current) test has completed.

ICMP ping link test options

pingaddr1={*ipv4 address*|*fqdn*}

The first host to test.

pingaddr2={*ipv4 address*|*fqdn*}

The second host to test, if the first host fails.

TCP connection link test parameters:

ipaddr1={*ipv4 address*|*fqdn*}

The first host to test.

ipaddr2={*ipv4 address*|*fqdn*}

The second host to test, if the first host fails.

ipport=1-65535

The TCP port number to connect to on the remote host. The default is 80.

DNS lookup link test parameters

dnsfqdn1=*dns fqdn*

The first hostname to look up.

dnsfqdn2=*dns fqdn*

The second hostname to look up, if the first hostname fails.

Example

```
set surelink system_reset_connect_failures=30
```

See also

- "revert" on page 61.
- "show" on page 249.
- The "display pppstats" command displays connection and activity information for PPP links, including SureLink statistics. See "SureLink Statistics" on page 35 for descriptions of these statistics.
- Digi SureLink™ “Always-On” Connection White Paper, available from the Documentation page for Digi Cellular Family products at digi.com.

set switches

Device support

This command is supported in ConnectPort TS MEI products only.

Purpose

Configures Multiple Electrical Interface (MEI) settings on a per-port basis, and displays current MEI settings. MEI settings include the type of electrical interface (EIA-232 or EIA-485), the number of differential wires used for communication, and whether termination and biasing resistors are used.

Required permissions

The serial permissions associated with the "set serial" command also apply to this command. For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following.

- For a user to display the serial settings for the line on which they are logged in: “set permissions s-serial=r-self”
- For a user to display the serial settings for any line: “set permissions s-serial=read”
- For a user to display and set the serial settings for the line on which they are logged in: “set permissions s-serial=rw-self”
- For a user to display the serial settings for any line, and set serial settings for the line on which the user is logged in: “set permissions s-serial=w-self-r”
- For a user to display and set the serial settings on any line: “set permissions s-serial=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Configure MEI settings

```
set switches [port=range]
   [mode={232|485}]
   [wires={two|four}]
   [termination={on|off}]
```

Display current MEI settings

```
set switches
```

Options

range=*range*
The port or range of ports to which this command applies.

mode={232|485}
Selects the electrical interface of the serial port. The selected value determines whether the "wires" and "termination" options are meaningful.

232
The serial port uses electrical interface EIA-232. This interface uses independent wires to transmit and receive data, which allows data to be sent and received between devices simultaneously.

485
The serial port uses electrical interface EIA-485. This mode can also be used for EIA-422 connections. This interface uses two wires to both transmit and receive data. This interface also allows for multiple transmitters and receivers to be easily connected together.

The "wires" and "termination" command options specifically apply to serial ports in EIA-485 mode.

The default is "232."

wires={two|four}
Applies when the serial port is running in EIA-485 mode only. Selects the number of differential wires used for communication and implicitly determines the duplex of the connection.

two
The serial port operates in two-wire mode. This mode is a half-duplex connection with *shared* transmit and receive wires.

four
The serial port operates in four-wire mode. This mode is a full-duplex connection with *independent* transmit and receive pairs.

The default is "four."

termination={on|off}
Applies when the serial port is running in EIA-485 mode only. Determines whether termination and biasing resistors are used across the lines.

on
Termination and biasing resistors are enabled across the lines.

Termination should be set to "on" if this node is an endpoint of the 485 network. Biasing should be used in at least one unit in a two-wire environment.

off
Termination and biasing resistors are disabled across the lines.

The default is "off."

Examples

Configure standard EIA-232 communication

```
#> set switches port=1 mode=232
```

Configure a half-duplex EIA-485 endpoint

```
#> set switches port=1 mode=485 wires=two termination=on
```

Configure a full-duplex 422 interior node

```
#> set switches port=1 mode=485 wires=four termination=off
```

See also

- "display" on page 27. The “display switches” command displays the current switch settings.
- "revert" on page 61. The “revert switches” command reverts the “set switches” configuration.

set system

Devices supported

This command is supported in all Digi Connect products.

Purpose

Configures and displays system-identifying information, such as a description of the device, its location, and a contact person.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions s-service=read” to display network service settings, and “set permissions s-services=rw” to display and change network service settings.

Syntax

Change system-identifying information

```
set system [description=string|location=string|contact=string]
```

Display system-identifying information

```
set system
```

Options

description=*string*
A description of this device. The maximum length is 64 characters. The default is “”.

location=*string*
The location of this device. The maximum length is 64 characters. The default is “”.

contact=*string*
The contact for this device. The maximum length is 64 characters. The default is “”.

Examples

Set description, contact, and location

```
#> set system description=”Engineering printer” location=”Room 1347”
contact=”John Doe at x-3749”
```

See also

- "revert" on page 61.
- "show" on page 249.

set tcpserial

Devices supported

This command is supported in all Digi Connect products.

Purpose

Used to set behaviors of TCP serial connections, or display current TCP serial settings.

This command affects the following TCP serial connections:

- Connections made using the autoconnect feature.
- Incoming network connections made to the following:
  - The TCP server (raw socket, IP port 2101)
  - The Telnet server (telnet socket, IP port 2001)
  - Secure Sockets Layer (ssl socket, IP port 2601)

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display the TCP serial settings for the line on which they are logged in: “set permissions s-tcpserial=r-self”
- For a user to display the TCP serial settings for any line: “set permissions s-tcpserial=read”
- For a user to display and set the TCP serial settings for the line on which they are logged in: “set permissions s-tcpserial=rw-self”
- For a user to display the TCP serial settings for any line, and set TCP serial settings for the line on which the user is logged in: “set permissions s-tcpserial=w-self-r”
- For a user to display and set the TCP serial settings on any line: “set permissions s-tcpserial=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Set behaviors of TCP serial connections

```
set tcpserial port=range
   [hangupdcd={on|off}]
   [hangupdsr={on|off}]
   [idletime={0|n}]
   [sid={on|off}]
   [sidstring=socketid string]
   [buffered={on|off}]
   [sendcount=1-65535 bytes]
   [sendtime={0|1-65535ms}]
   [endpattern=string]
   [strippattern={on|off}
```

Display TCP serial settings

```
set tcpserial [port=range]
```

Options

port=*range*
Used to specify the serial port. Optional on a single-port device.

hangupdcd={on|off}
Indicates whether an established network connection should be terminated when the serial port's DCD signal drops. The default is "off."

hangupdsr={on|off|}
Indicates whether an established network connection should be terminated when the serial port's DSR signal drops. The default is "off."

idletime=idletime={0|*n*}
Indicates that established network connection should be terminated if the serial port is idle for the specified amount of time in seconds. A value of 0 (zero) disables this option. The default is 0.

sid={on|off}
Determines how the socket ID (SID) string in the "sidstring" option is handled.

on
The value for the "sidstring" option is sent to the network destination right before the first data bytes are sent to the network.

off
The value for the "sidstring" option is not sent to the network destination.

The default is "off."

sidstring=*socketid string*
When the "sid" option is set to on, this string is sent to the network destination right before the first data bytes are sent to the network. The maximum length of this string is 256 characters, including escape sequences for special characters. The maximum parsed length of this string is 256 characters. That is, this string must reduce down to a 256-character string when the escape sequences are processed. For more details on the escape sequences, see "Entering Special Characters in String Values" on page 13.

buffered={on|off}
Turning on this feature on allows controlling how serial data is sent out to the network. The "sendcount," "sendtime," "endpattern," and "strippattern" options are used to control how data is sent out once the "buffered" option is set to "on." The default is "off."

sendcount=1 - 65535 bytes
Indicates that data from the serial port should be sent out to the network after buffering the given number of bytes. This option only is valid when the "buffered" option is "on." The default is 1024 bytes.

sendtime={0|1-65535ms}
Indicates that data from the serial port should be sent out to the network after the given amount of time has past where no new data has arrived from the serial port. This option only is valid when the “buffered” option is “on.” A value of 0 (zero) disables this option. The default is 0.

endpattern=*string*
Indicates that data from the serial port should be sent out to the network after the given endpattern string has been found in the data from the serial port. This option only is valid when the “buffered” option is “on.” An empty string disables this option.

The maximum length of this string is 16 characters, including escape sequences for special characters. For more details on the escape sequences, see "Entering Special Characters in String Values" on page 13. The maximum parsed length of this string is 4 characters. That is, this string must reduce down to a 4-character string when the escape sequences are processed.

strippattern={on|off}
This option corresponds with the “endpattern” option. When a valid “endpattern” string is found, this option indicates whether the matching string is stripped or kept in the data stream. The default is “off.”

Examples

```
#> set tcpserial hangupdcd=off idletime=20
#> set tcpserial port=1 sid=on sidstring=”abc”
#> set tcpserial port=1 buffered=on sendtime=50 sendcount=512
#> set tcpserial
```

See also

- "revert" on page 61.
- "show" on page 249.

set term

Devices supported

This command is supported in all Digi Connect products except ConnectPort Display.

Purpose

Allows for connecting a terminal to a device's serial port and accessing the command line of the device.

In the cases where the default access to the terminal and the command line is "on," this command is important if users want to use the serial port for purposes other than having a command line. That is, they must change the state of the serial port access from "on" to "off" in order to use the serial port for another purpose.

Required permissions

For Digi Connect products with two or more users, permissions must be set to "set permissions s-term=read" to display terminal settings, and "set permissions s-term=rw" to display and change terminal settings.

Syntax

Configure terminal settings

```
set term [state={on|off}]
```

Display current terminal settings

```
set term
```

Options

state={on|off}

Specifies whether terminal access is enabled for the serial port. The default is "on" for Digi Connect WAN and Digi Connect RG devices.

Examples

```
#> set term

 Serial Terminal Configuration :

 port# state
     1    on
     2    on
```

See also

- "revert" on page 61.
- "show" on page 249.

set udpserial

Devices supported

This command is supported in all Digi Connect products except Digi Connect WAN and ConnectPort Display.

Purpose

Use this command to set up the UDP serial feature, or display current UDP serial settings.

The UDP serial feature allows data to be sent between the serial port and one or more remote network destinations using the UDP protocol. When this feature is enabled for a given serial port, data sent to the serial port will be sent out to the configured destinations. Also any time data is sent to the UDP serial service (IP port 2101) and the serial port is not being used by another service, the data will be sent to the serial port.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display the UDP serial settings for the line on which they are logged in: “set permissions s-udpserial=r-self”
- For a user to display the UDP serial settings for any line: “set permissions s-udpserial=read”
- For a user to display and set the UDP serial settings for the line on which they are logged in: “set permissions s-udpserial=rw-self”
- For a user to display the UDP serial settings for any line, and set UDP serial settings for the line on which the user is logged in: “set permissions s-udpserial=w-self-r”
- For a user to display and set the UDP serial settings on any line: “set permissions s-udpserial=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Set general UDP serial forwarding characteristics for a serial port

```
set udpserial port=range [state={on|off}]
   [sendcount=bytes] [sendtime={0|time}]
   [endpattern=string] [strippattern={on|off}]
   [sid={on|off}] [sidstring=string}
   [closetime=time]
```

Set UDP destinations for a given serial port

```
set udpserial port=range range=1-64 [description=string]
   [active={on|off} [ipaddress=ip address}
   [ipport=ip port]
```

Display current UDP serial settings

```
set udpserial [port=range [range=range]]
```

Options

Options for setting general UDP serial forwarding characteristics

port=*range*
Used to specify the serial port. Optional on a single-port device.

state={on|off}
Used to enable or disable sending data from the serial port to remote network destinations. The default is “off.”

sendcount=*bytes*
The number of bytes received from the serial port that will cause the data to be sent on to the network destinations. This trigger cannot be disabled. The default is 1024 bytes.

sendtime={0|*time*}
The amount of idle time, in milliseconds, allowed before sending data to the network. If no data is received on the serial port for the time specified by this option, any buffered data will be sent on to the network destinations. A value of 0 (zero) disables this trigger.

endpattern=*string*
If this string is set, any pattern match of data received from the serial port will cause the data to be sent on to the network destinations. The maximum length of this string is 16 characters, including escape sequences for special characters. For more details on the escape sequences, see "Entering Special Characters in String Values" on page 13. The maximum parsed length of this string is 4 characters. That is, this string must reduce down to a 4-character string when the escape sequences are processed.

strippattern={on|off}
Determines how the data specified by the “endpattern” option is handled.

on
The endpattern that is found is stripped from the stream before any data is to be sent on to the network destinations.

off
The endpattern is not stripped from the stream before data is sent on to network destinations.

The default is “off.”

sid={on|off}
Determines how the socket ID (SID) string in the “sidstring” option is handled; that is, whether the string specified by the “sidstring” option is sent at the beginning of each UDP packet.

on
The value of “sidstring” is sent at the beginning of each UDP packet.

off
The value of “sidstring” is not sent at the beginning of each UDP packet.

The default is “off.”

sidstring=*string*
The string sent at the beginning of each UDP packet if the “sid” option is set to on. The maximum length of this string is 256 characters, including escape sequences for special characters. For more details on the escape sequences, see "Entering Special Characters in String Values" on page 13. The maximum parsed length of this string is 256 characters. That is, this string must reduce down to a 256-character string when the escape sequences are processed.

clostime=*time*
The amount of idle time before closing the serial port, in milliseconds. If no data is sent or received on the serial port for the specified amount of time, the serial port is closed. This allows the serial port to be used by other things such as TCP socket or RealPort. If a value of 0 milliseconds is set, the “closetime” option will internally be recalculated to be 1 millisecond or twice the send time, whichever is greater. The default is 0 milliseconds.

Options for setting UDP destinations for a given serial port

The following options require a specific range to be specified by the “range” option.

port=*range*
Specifies the serial port. Optional on a single-port device.

range={1-64}
Specifies the UDP destination to be configured.

description=*string*
A string for descriptive purposes only.

active={on|off}
Specifies whether data from the serial port is sent to this destination.

on
Data from the serial port is sent to this destination.

off
This destination is not sent any data.

The default is “off.”

ipaddress=*ip address*
The IP address of the network destination to which data is sent.

ipport=*ip port*
The UDP port of the destination to which data is sent.

Options for displaying current UDP serial settings

port=*range*
Used to specify the serial port. Optional on a single-port device.

range=range
Identifies the range of UDP destinations to be displayed.

Examples

Set general UDP serial forwarding based on bytes received

In this example, the amount of bytes received from the serial port will cause the data to be sent on to the network destination.

```
#> set udpserial port=1 state=on sendcount=2
```

Set UDP destinations for a given serial port

In this example, data will be sent to the destination identified.

```
#> set udpserial port=1 range=1 ipaddress=10.0.0.1 ipport=2101 active=on
```

Display current UDP serial settings

The following are all valid ways of using set udpserial to display current UDP serial settings:

```
#> set udpserial
#> set udpserial port=1
#> set udpserial port=1 range=1-12
```

See also

- "revert" on page 61.
- "show" on page 249.

set user

Devices supported

This command is supported in all Digi Connect products.

Purpose

Used to:

- Add users for access to a Digi Connect device. The number of users that can be defined varies by Digi Connect device. To determine the number of users allowed for your Digi Connect device, enter “set user” or “show user”.
- Associate a user with a group. A user can be associated with up to two groups.
- Disassociate a user from a group.
- Remove users.
- Change user configuration attributes.
- Display user configuration attributes.
- Load an SSH public key, and, for single-user model products, unload a public key.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display user configuration attributes: “set permissions s-user=read”
- For a user to display and set user configuration attributes: “set permissions s-user=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Default permissions for a new user

When a new user is created, it is given a set of default permissions. Once a user is created, an administrator can adjust permissions up or down as needed. Default permissions for a new user are as follows. For more information on user permissions, see "set permissions" on page 157.

- none: for backup, boot, connect, display, buffers, kill, s-alarm, s-gpio, s-permissions, s-pmodem, s-rciserial, s-snmp, status, webui, filesys, s-idle, s-panic, revert-all, s-trace, s-wlan, s-menu, s-profile
- execute: For reconnect, rlogin, telnet, who, ping.
- read: access, s-ethernet, s-group, s-network, s-serial, s-service, s-system, s-trane, s-user
- r-self: autoconnect, rtstoggle, tcpserial, udpserial
- rw-self: newpass

Syntax

Add a user

```
set user add id=number newname=string
```

Remove a user

```
set user remove {id=range|name=string}
```

Associate a user with a group

```
set user associate {id=number|name=string}
  {gid=number|gname=string}
```

Disassociate a user from a group

```
set user disassociate {id=number|name=string}
  {gid=number|gname=string}
```

Change user configuration attributes

```
set user [id=range|name=string]
   [newname=string]
   [commandline={on|off}]
   [groupaccess={on|off}]
   [menu={none|index|name}]
   [defaultaccess={none|commandline|group|menu}]
   [defaultgroup={none|gid|gname}]
```

Display user configuration attributes

```
set user {id=range|name=string}
```

Display user configuration attributes for all users

```
set user
```

Load an SSH public key

```
set user public_key=tftphost:filename
```

Remove an SSH public key

```
set user public_key=clear
```

Options

add
Add a user. New users are created with the default permissions (see "Default permissions for a new user" earlier in this description). A maximum of 32 users can be defined.

remove
Remove users.

associate
Associate a user with a group. A user can be associated with a maximum of two groups.

disassociate
Disassociate a user from a group.

id=*range*
Specifies the ID or range of IDs of the users to be acted on.

name= *string*
Specifies the name of the user to be acted on.

newname=*string*
Specifies a new user name.

gid=*number*
Specifies the identifier for the group being associated with a user. If omitted, the “gname” option must be specified.

gname=*string*
Specifies the name of the group being associated with a user. If omitted, the “gid” option must be specified.

commandline={on|off}
Specifies whether the user is allowed to access the command line of the device.

on
User can access the command line interface.

off
User can not access the command line interface.

The default is “on.”

groupaccess={on|off}
Specifies whether the user is allowed to use the access rights for any associated groups. This allows a group to define the access rights for users. For instance, if the user has “commandline=off” and an associated group has “commandline=on,” then the user will have command line access if “groupaccess=on.”

on
The user can use group access rights.

off
The user cannot use group access rights.

The default is “off.”

menu={none|index|name}
Specifies whether the user is allowed to access the custom menu interface of the device and defines the custom menu that the user will have displayed.

none
The user is not allowed to access the custom menu interface.

index
The user is allowed to access the custom menu interface and will be displayed the custom menu at the specified index.

name
User is allowed to access the custom menu interface and will be displayed the custom menu using the specified name.

The default is “none.”

defaultaccess={none|commandline|menu|group}

Specifies the default access method and interface that a user will be given upon logging into the device. Note that the specified interface must be enabled for the user and have a valid menu and/or group if specified.

none

The user has no default access to the device and must explicitly specify the access type. If the user and/or associated group has no access rights then the user is not allowed to access either the command line interface or the custom menu interface.

commandline

The user will be displayed and given access to the command line interface assuming the user and/or associated groups have command line access rights enabled.

menu

The user will be displayed and given access to the custom menu interface and be displayed the custom menu as specified by the “menu” option.

group

The user will be displayed the default access interface as specified by the “defaultgroup” option, assuming the specified group is valid and associated to this user. This allows the default access for a user to be controlled by the associated group.

The default is “commandline.”

defaultgroup={none|gid|gname}

Specifies the default group to use when checking the default access rights when the “defaultaccess” option is set to group. The specified group must be valid and associated to the user.

none

The user will not be given any default access.

gid

The user will be given the default access method according to the default access of the group with the specified gid.

gname

The user will be given the default access method according to the default access of the group with the specified name.

The default is “none.”

public_key={*tftphost*:*filename*|clear}

Loads or clears an SSH public key used for authentication of this user. The key must be an RSA public key, in either OpenSSH or the IETF draft format.

tftphost*:*filename

Loads an SSH2 public key for use with this user, where:

tftphost

The IP address or DNS name of a host from which the SSH public key will be downloaded to the Digi Connect device using TFTP.

filename

The name of a file on the host that contains the SSH public key. If your host's implementation requires a complete path to this file, specify the path here as well.

clear

Unloads an SSH public key.

Examples

Add a new user

```
#> set user add newname=jsmith id=4
```

Remove user 7

```
#> set user remove id=7
```

Associate user "johndoe" with the root group

```
#> set user associate name=johndoe gname=root
```

Disassociate user 15 from group 2

```
#> set user disassociate id=15 gid=2
```

Set a new user name to be entered at login

```
#> set user id=4 newname=jdoe
```

Set a user to have default command line interface access

```
#> set user id=4 commandline=on defaultaccess=commandline
```

Set a user to use group access rights

```
#> set user name=johndoe groupaccess=on defaultaccess=group defaultgroup=root
```

See also

- "User Models and User Permissions in Digi Connect Products" on page 14.
- "newpass" on page 51.
- "revert" on page 61
- "set group" on page 115.
- "set menu" on page 132.
- "set permissions" on page 157.
- "show" on page 249.

set video

Devices supported

This command is supported in ConnectPort Display only.

Purpose

Configures or displays video settings for ConnectPort Display.

Syntax

Configure video settings

```
set video mode={640x480@60-16|800x600@56-16|800x600@60-16|
  1024x768@70-8}
  splash_time=0-30 seconds
```

Display current video settings

```
set video
```

Options

mode={640x480@60-16|800x600@56-16|800x600@60-16|1024x768@70-8}

The resolution, refresh rate, and color depth of the display screen.

splash_time=0-30 seconds

The amount of time, in seconds, to show the splash screen. Valid values are 0 through 30. A value of 0 disables the splash screen.

Example

```
#> set video mode=800x600@60-16 splash_time=5
```

See also

- "revert" on page 61.
- "set putty" on page 175.
- "set vncclient" on page 226.
- "show" on page 249.

set vncclient

Devices supported

This command is supported in ConnectPort Display only.

Purpose

Configures or displays remote-access settings. Your ConnectPort Display can provide remote access to a computer on the network or Internet using the VNC (Virtual Network Computing) protocol.

VNC server software must be installed on the remote computer. A VNC server is provided on your ConnectPort Display Software and Documentation CD.

You can interact with the remote computer using a keyboard and mouse connected to the USB ports on your ConnectPort Display.

Syntax

Configure remote-access settings

```
set vncclient [state={on|off}]
   [server=vnc server ipaddr|dns name]
   [port=vnc server network port]
   [password=vnc server password]
   [reconnect=0-2000000 seconds]
   [shared={on|off}]
   [localcursor={on|off}]
   [keepalive={on|off}]
```

Display current remote-access settings

```
set vncclient
```

Options

state={on|off}
Enables or disables the connection to a remote computer's VNC server.

server={*vnc server ipaddr*|*dns name*}
The VNC server's IP address or DNS name.

port=*vnc server network port*
The network port number to connect to on the VNC server. The default port number for VNC servers is 5900.

password=*vnc server password*
The password for logging on to the VNC server.

reconnect=0-2000000 seconds
The maximum amount of time to wait before attempting to reconnect to the VNC server if the connection cannot be established or lost.

shared={on|off}
Specifies whether the VNC server desktop can be shared with other clients. If "shared=on," Other VNC clients can connect to the VNC server while your ConnectPort Display is connected.

localcursor={on|off}
Enables or disables local mouse cursor handling. Tracking the mouse cursor locally can improve mouse performance, especially with a slow VNC server or slow network.

keepalive={on|off}
Indicates whether or not TCP keep-alives will be sent while connected to the VNC server. Keep-alives help to detect when a connection has been lost. TCP keep-alive parameters (such as how often to send them) are configured globally.

Example

```
#> set vncclient state=on server=10.20.1.107 port=5900 password=dnf10
reconnect=10 shared=on localcursor=on
```

See also

- "revert" on page 61.
- "set service" on page 191. The VNC Client Listen Daemon and VNC Server services are enabled and disabled by the “set service” command.
- "set video" on page 225.
- "show" on page 249.
- The *ConnectPort Display User’s Guide’s* section titled “Configure Remote Access (VNC Client) Settings.”

set vpn

Devices supported

This command is supported in Digi Cellular Family products only, except Digi Connect WAN.

Purpose

Configures Virtual Private Network (VPN) settings. Virtual Private Networks (VPN) are used to securely connect two private networks together so that devices may connect from one network to the other network using secure channels. VPN uses IP Security (IPSec) technology to protect the transferring of data over the Internet Protocol (IP).

The Digi Cellular Family device is responsible for handling the routing between networks. Devices within the private network of the Digi Cellular Family device can connect directly to devices on the other private network to which the VPN tunnel is established to. The VPN tunnels are configured using various security settings and methods to ensure the networks are secured.

Connect WAN products support up to two VPN tunnels. ConnectPort WAN products support up to five VPN tunnels.

It is generally easier to configure VPN tunnel settings through the Web user interface. VPN settings are configured on the **Network** > **Virtual Private Network (VPN)** configuration pages named **VPN Settings** and **VPN Tunnel Settings**.

There are several uses of the “set vpn” command:

- Configure global VPN options, including:
  - The connection mode method used to negotiate Internet Key Exchange (IKE) Phase One using Internet Security Association and Key Management Protocol (ISAKMP).
  - How the VPN client is identified to the remote VPN endpoint.
  - The Diffie-Hellman group used within IKE to establish the session keys used to create a secure channel. The method and security factor used to control the key exchange is specified by the Diffie-Hellman group.
  - Use of Perfect Forward Secrecy (PFS).
  - Use of antireplay.
- Configure and modify VPN tunnel options: VPN Tunnels define the actual tunnels that exist between two private networks. The tunnels specify the information required to establish the secure channel, the routing between the networks, and the security policies used to encrypt and authorize the data. A maximum of two tunnels may be created. Configuring a VPN tunnel requires the remote VPN endpoint and the method by which to establish the VPN tunnel. These settings are typically specified by the remote VPN server and should correspond accordingly. Both manually keyed and ISAKMP tunnels can be configured.

- Configure IKE/ISAKMP SA Phase 1 and Phase 2 options, which create an authenticated secure channel and specify how IKE negotiates security associations (SAs).
- Display current VPN settings.

Required permissions

For Digi Connect products with two or more users, to use this command, permissions must be set to one of the following:

- For a user to display VPN settings: “set permissions s-vpn=read”
- For a user to display and set VPN settings: “set permissions s-vpn=rw”

See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

Set global VPN options

```
set vpn global
   [mode={main|aggressive}]
   [identity={fqdn|user fqdn|ip address}]
   [dh_group={1|2|5}]
   [pfs={on|off}]
   [antireplay={on|off}]
```

Set VPN tunnel options

```
syntax: set vpn tunnel [tunnel options]
   [manually-keyed options]
   [isakmp options]
```

Where:

```
[tunnel options]:
   [index={1-2 (for Connect WAN products)|1-5 (for ConnectPort
   products)}]
   [name=tunnel name]
   [newname=tunnel name]
   [mode={disabled|manually-keyed|isakmp}]
   [remote_vpn_endpoint=(fqdn|ip address}]
   [remote_tunnel_addr=ip address]
   [remote_tunnel_mask=subnet mask]
   [remote_tunnel_range=ip address-ip address]
   [local_tunnel_addr=ip address]
   [local_tunnel_mask=subnet mask]
   [local_tunnel_range=ip address-ip address]
[manually-keyed options]:
   mode=manually-keyed
   [inbound_spi=256 - 2^32) (Please see option details below)
   [inbound_authentication={none|md5|sha1}]
   [inbound_auth_key={ascii key|hex key}]
   [inbound_encryption={none|des|3des|aes}]
   [inbound_enc_key={ascii key|hex key}]
   [outbound_spi=256-2^32] (Please see option details below)
   [outbound_authentication={none|md5|sha1}]
   [outbound_auth_key={ascii key|hex key}]
   [outbound_encryption={none|des|3des|aes}]
   [outbound_enc_key={ascii key|hex key}]
```

```
[isakmp options]:
  mode=isakmp
  [shared_key={ascii key|hex key}]
```

To specify proposals: see syntax and options for “Set IKE/ISAKMP SA Phase 2 Options.”

Set IKE/ISAKMP SA Phase 1 Options

```
syntax: set vpn phase1
  [index=1-2]
  [state={enabled|disabled}]
  [auth_method={shared_key|dss|rsa}]
  [authentication={md5|sha1}]
  [encryption={des|3des|aes}]
  [encryption_size={0|128|192|256}]
  [sa_lifetime=10-2^32] (Please see option details below)
  [sa_lifetime_data=0-2^32] (Please see option details below)
```

Set IKE/ISAKMP SA Phase 2 Options

```
set vpn phase2
  [tunnel=1-2]
  [name=tunnel name]
  [proposal=(1-8)]
  [state={enabled|disabled}]
  [authentication={none|md5|sha1}]
  [encryption={none|des|3des|aes}]
  [sa_lifetime=60-2^32] (Please see option details below)
  [sa_lifetime_data=0-2^32] (Please see option details below)
```

Display current VPN settings

```
set vpn global

set vpn tunnel

set vpn phase1

set vpn phase2
```

Options

Global VPN options

set vpn global

Specifies that the “set vpn” command is for setting global VPN options.

mode={main|aggressive}

The method used to negotiate Internet Key Exchange (IKE) Phase One using Internet Security Association and Key Management Protocol (ISAKMP). Negotiations establish security settings and a secure channel for subsequent messages. For the negotiations to progress, both sides must be configured identically.

main

Main mode processes Phase One negotiations using three two-way exchanges between the VPN client and remote VPN endpoint. The exchanges are meant to match IKE Security Associations (SA) between peers to provide a protected pipe for subsequent protected ISAKMP exchanges between the peers. The first exchange negotiates and agrees upon algorithms and hashes/keys used to secure the IKE communications. The second exchange uses a Diffie-Hellman exchange, per the specified Diffie-Hellman group, to generate nonces and shared secret keys to sign and prove identities. The third exchange verifies the identity per the specified Identity.

aggressive

Aggressive mode processes Phase One negotiations using fewer exchanges than Main Mode processing. In the first exchange, almost everything is sent in the proposed IKE values, including the Diffie-Hellman key, nonce to sign and verify, and the identity. The weakness of using Aggressive Mode compared to Main Mode is that negotiations exchange information before the secure channel is created. However, because fewer exchanges are used, aggressive mode is faster than main mode. Aggressive mode may be required when a peer gateway IP address is dynamic.

The default is “main.”

identity={*fqdn*|*user fqdn*|*ip address*}

Specifies how the VPN client is identified to the remote VPN endpoint. The identity must match the value provided by the remote VPN endpoint to properly identify this client and its respective security settings. This option assumes the use of pre-shared key and is used to identify the pre-shared key. This option can be specified in three ways:

identity=*fqdn*

Identity is specified as a Fully Qualified Domain Name (FQDN), usually the FQDN of the Digi Connect device in the form of an Internet hostname, for example www.myhost.com or remote3.digi.com.

identity=*user fqdn*

Identity is specified as a User Fully Qualified Name (UFQN, or User FQDN). A User FQDN is similar to standard FQDN, but with a user name. The format is the same as an email address, for example, user@myhost.com or remote3@digi.com. This is the default representation used by Digi devices, because it can easily be added to authentication systems.

identity=*ip-address*

Identity is specified as the Digi device's IP address. Using this method, you can specify either of the following:

The Network Address (IPv4): A standard IP address (version 4). that uses the standard IPv4 dotted format (four numeric values between 0 and 255 separated by periods). For example: 10.0.0.1

The Mobile IP address as the identity: This means that the IP address of your mobile network interface will automatically be used as the VPN identity.

The IP-address method is the easiest for system administrators to use, because it is both familiar and should be unique. However, it is not always the best choice. The IP address may be for the device, unless special arrangements are made with the cellular carrier. This presents a difficult configuration issue, unless a large subnet of addresses are defined to use a single pre-shared key.

The default identify form is "*macaddress*@digi.com."

dh_group={1|2|5}

The Diffie-Hellman (DH) prime modulus group. Diffie-Hellman is a public-key cryptography protocol for establishing a shared secret over an insecure communications channel. Diffie-Hellman is used with IKE to establish the session keys that create a secure channel. This setting is used if Perfect Forward Secrecy is also enabled ("pfs=on.")

Digi Cellular Family products support the following Diffie-Hellman prime modulus groups:

dh_group=1
Group 1 (768-bit).

dh_group=2
Group 2 (1024-bit).

dh_group=5
Group 5 (1536-bit).

The default is 2 (Group 2).

pfs={on|off}]

Specifies whether the Perfect Forward Secrecy (PFS) method is on or off. PFS is a method of deriving session keys from known keying material. PFS establishes greater resistance to cryptographic attacks by ensuring that a given key of an IKE SA is not derived from any other secret, and that no other key can be derived from this key.

For negotiations to succeed, both the local and remote sides of the connection must have the "pfs" and "dh_group" options set to the same values.

The default is "on."

antireplay={on|off}
Specifies whether the antireplay feature is on or off. Antireplay allows the IPsec tunnel receiver to detect and reject packets that have been replayed. It does this by adding information to the packets exchanged between VPN endpoints, to ensure that a third party cannot replay the same information to one of the VPN endpoints at a later time to recreate the secure channel again.

Important: If using manually-keyed tunnels, disable this option.

For negotiations to succeed, both the local and remote sides of the connection must be set to the same value. Set this field to match that at the remote VPN gateway. The default is “on.”

VPN tunnel options

VPN tunnel options are specified in this format:

```
set vpn tunnel [tunnel options] [manually-keyed options]
   [isakmp options]
```

Where:

set vpn tunnel
Specifies that the “set vpn” command is for configuring a VPN tunnel.

[*tunnel options*]
The VPN tunnel configuration options. The set of options specified depends on whether the method of establishing the VPN tunnel is manually-keyed or ISAKMP.

index={1-2}
The index number for an existing VPN tunnel.

name=*tunnel name*
A name that describes the VPN tunnel. This may be used to help identify each tunnel with a descriptive and unique name.

newname=*tunnel name*
The new name for the VPN tunnel.

mode={disabled|manually-keyed|isakmp}
The method of establishing the VPN tunnel.

disabled
The VPN tunnel is enabled or disabled. Use this option when creating several tunnels, where only one would be used initially. In that case, you would add a disabled tunnel for future use and enable it on a subsequent “set vpn” command.

manually-keyed
The VPN tunnel is established by manually keying in VPN tunnel and security settings. These settings must match the settings of the remote VPN endpoint. Manually-keyed VPNs do not use IKE/ISAKMP. Manually-keyed VPN keys never expire.

isakmp
The VPN tunnel is established by specifying a list of security policies to negotiate a set of security settings from the remote VPN endpoint.

remote_vpn_endpoint=(fqdn|*ip address*}
The IP address or hostname of the peer with which the VPN connection is established.

remote_tunnel_addr=*ip address*
remote_tunnel_mask=*subnet mask*
remote_tunnel_range=*ip address-ip address*
These options specify the routes required to access clients on the remote network. They also specify the remote peers that local clients are allowed to connect to. The remote network specifies the private network to which the remote VPN endpoint is connected.

local_tunnel_addr=*ip address*
local_tunnel_mask=*subnet mask*
local_tunnel_range=*ip address-ip address*
These options specify the routes required to access clients on the local network. They also specify the clients that are allowed to access the remote clients through the VPN tunnel. Typically, the local network specifies the same network and subnet connected to the Digi Cellular device's Ethernet port. Thus, any client on the same network will be able to communicate over the VPN tunnel.

[*manually-keyed options*]
These options are for VPN manually-keyed VPN tunnels. To properly configure a manual-keyed tunnel, the following settings are required to be set as specified by the remote VPN server. This includes the local and remote network settings that handle the routing between the local and remote peers. It also includes the security settings for both incoming and outgoing traffic, which may be different from each other, depending on the implementation of the remote VPN server. Incoming or inbound traffic is defined as any traffic sent from a remote peer on the remote network of the remote VPN endpoint to a local peer on the local network. Outgoing or outbound traffic is defined as any traffic sent from a local peer to a remote peer.

mode=manually-keyed
Indicates that the settings are for a manually-keyed VPN tunnel. Manually-keyed tunnels specify the tunnel and security settings manually. These settings must match the settings of the remote VPN endpoint.

inbound_spi=256 - 2^32
The Security Parameter Index (SPI) for inbound traffic.The SPI defines the unique index for a tunnel used to identify the security settings for IPSec. The SPI is a 32-bit unsigned value that must not be less than 256.

inbound_authentication={none|md5|sha1}
The optional authentication algorithm, used with the associated authentication key specified by the “inbound_auth_key” option, to authorize access on the VPN tunnel for inbound traffic.

none
No authentication algorithm is used.

md5
MD5 authentication algorithm, which uses 128-bit keys.

sha1
SHA1 authentication algorithm, which uses 160-bit keys.

inbound_auth_key={*ascii key*|*hex key*}
The authentication key for inbound traffic, according to the authentication algorithm specified by the “inbound_authentication” option. The authentication key may be specified as an ASCII value using alpha-numeric characters or may be specified as a hexadecimal value prefixed by “0x”. The following table lists the associated lengths of the authentication keys based on the authentication algorithm.

Algorithm	Size	Key Length	
		ASCII	Hexadecimal
MD5	128-bit	16	32
SHA1	160-bit	20	40

inbound_encryption={none|des|3des|aes}
The optional encryption algorithm used with the associated encryption key specified by the “inbound_enc_key” option to encrypt data on the VPN tunnel for inbound traffic.

none
No encryption algorithm is used.

des
DES encryption algorithm, which uses 64-bit keys.

3des
3DES encryption algorithm, which uses 192-bit keys.

aes
AES encryption algorithm, which uses 128-bit keys.

inbound_enc_key={*ascii key*|*hex key*}
The encryption key for inbound traffic, according to the authentication algorithm specified by the “inbound_encryption” option. The encryption key may be specified as an ASCII value using alpha-numeric characters or may be specified as a hexadecimal value prefixed by “0x”. The following table lists the associated lengths of the encryption keys based on the encryption algorithm.

Algorithm	Size	Key Length	
		ASCII	Hexadecimal
DES	64-bit	8	16
3 DES	192-bit	24	48
AES	128-bit	16	32

outbound_spi=256 - 2^32
The SPI for outbound traffic.The SPI defines the unique index for a tunnel used to identify the security settings for IPSec. The SPI is a 32-bit unsigned value that must not be less than 256.

outbound_authentication={none|md5|sha1}
The optional authentication algorithm used with the associated authentication key specified by the “outbound_auth_key” option to authorize access on the VPN tunnel for outbound traffic.

none
No authentication algorithm is used.

md5
MD5 authentication algorithm, which uses 128-bit keys.

sha1
SHA1 authentication algorithm, which uses 160-bit keys.

outbound_auth_key={*ascii key*|*hex key*}
The authentication key for outbound traffic, according to the authentication algorithm specified by the “outbound_authentication” option. The authentication key may be specified as an ASCII value using alpha-numeric characters or may be specified as a hexadecimal value prefixed by “0x”. For the allowed lengths for this key, see “inbound_auth_key.”

outbound_encryption={none|des|3des|aes}
The optional encryption algorithm used with the associated encryption key specified by the “outbound_enc_key” option to encrypt data on the VPN tunnel for outbound traffic. For the allowed values, see “inbound_encryption.”

outbound_enc_key={*ascii key*|*hex key*}
The encryption key for outbound traffic, according to the authentication algorithm specified by the “outbound_encryption” option. For the allowed values and key length, see “inbound_enc_key.”

[*isakmp options*]

To configure an ISAKMP tunnel, you must configure the settings to match those on the remote VPN server.

mode=isakmp
Indicates that the settings are for a VPN ISAKMP tunnel. ISAKMP tunnels specify a list of proposals, or security policies, in order to negotiate a set of security settings from the remote VPN endpoint.

shared_key={*ascii key*|*hex key*}
A key that secures the VPN tunnel. The key can be either an ASCII value using alphanumeric characters or a hexadecimal value prefixed by 0x.

To specify security proposals for VPN ISAKAMP tunnels, see "IKE/ISAKMP SA Phase 2 options" on page 239.

IKE/ISAKMP SA Phase 1 and Phase 2 options

Internet Key Exchange (IKE) negotiates the IPSec security associations (SA). This process requires that the IPSec systems first authenticate themselves to each other and establish ISAKMP (IKE) shared keys. The SAs are relationships between two or more entities or peers that describe how the entities or peers will use security services to communicate securely.

IKE negotiations are handled using two different phases.

- Phase 1 is responsible for creating an authenticated and secure channel between the two peers. Typically, phase one is completed using a Diffie-Hellman exchange using cryptography.
- Phase 2 is then responsible for negotiating the final SAs and generating the required keys and key material for IPSec. This is completed by negotiating one or more sets of security policies, or proposals, between the two peers until a given set is agreed upon by both peers.

Default Security Policies

The security policies that are negotiated and used in securing the SAs include the encryption algorithm, authentication algorithm, and the SA lifetime in seconds. By default, the Digi Cellular Family device includes the following set of defaults. If these settings do not match the VPN and IKE SA configuration of the remote peers or if further policies are required, select **Use the following policies to negotiate Internet Key Exchange (IKE) security settings** and add one or more security policies.

Encryption	Authentication	SA Lifetime
3-DES (192-bit)	SHA1	86400 seconds

IKE/ISAKMP SA Phase 1 options

set vpn phase1

Specifies that the “set vpn” command is for configuring a VPN Phase 1 options.

index=1-2

The index number for an existing VPN tunnel.

state={enabled|disabled}

Whether the VPN tunnel is enabled or disabled. You can use this option when creating several tunnels, where only one would be used initially. In that case, you would add a disabled tunnel for future use and enable it on a subsequent “set vpn” command.

auth_method={shared_key|dss|rsa}

The authentication method used by the VPN tunnel.

shared_key

Authentication is performed by using a key that secures the VPN tunnel, where the key is either an ASCII alphanumeric value or a hexadecimal value.

dss

Authentication is performed using Digital Signature Standard (DSS).

rsa

Authentication is performed using RSA, which uses a combination of sender’s and receiver’s public and private keys.

authentication={md5|sha1}

The authentication algorithm used in IKE negotiations to authenticate the IKE peers and Security Associations (SAs).

md5

MD5 authentication algorithm, which uses 128-bit keys.

sha1

SHA1 authentication algorithm, which uses 160-bit keys.

encryption={des|3des|aes}

The encryption algorithm used in IKE negotiations for encrypting data.

des

DES encryption algorithm, which uses 64-bit keys.

3des

3DES encryption algorithm, which uses 192-bit keys.

aes

AES encryption algorithm, which uses 128-bit keys.

encryption_size={0|128|192|256}

The encryption key length, in bits, used in IKE negotiations for encrypting data. The key length is based on the encryption algorithm and is used to calculate and create the shared key.

sa_lifetime=10-2^32
Determines how long an Security Association (SA) policy is active, in seconds. After the IKE SA has been negotiated, the SA lifetime begins. Once the lifetime has completed, a new set of SA policies are negotiated using IKE phase 2 negotiation.

sa_lifetime_data=0-2^32
The amount of data, in bytes or kilobytes, that is sent and received until the SA is renegotiated. This value is analogous to the SA lifetime. Also known as SA life size.

IKE/ISAKMP SA Phase 2 options

Security policies define the set of security settings for incoming and outgoing traffic used to encrypt and authorize data. One or more sets of settings may be specified. The actual set of negotiated settings depends on the available policies specified by the remote VPN endpoint.

The VPN Phase 2 options are used to configure a set of security policies for ISAKMP tunnels. The settings define the set of encryption and authentication algorithms used for incoming and outgoing traffic over the VPN tunnel.

A security policy can have multiple proposals. For example, a policy can have two proposals so to allow older VPN devices to connect using less-secure methods, while allowing the same policy to have a second (or more) proposal to allow newer, more powerful end-points to use more secure methods.

set vpn phase2
Specifies that the “set vpn” command is for configuring a VPN Phase 2options.

tunnel=1-2
The index number assigned to the VPN tunnel.

name=*tunnel name*
The name of the VPN tunnel.

proposal=(1- 8)
The index number assigned to the security proposal.

state={enabled|disabled}
Whether the VPN tunnel is enabled or disabled. You can use this option when creating several tunnels where only one would be used initially. In that case, you would add a disabled tunnel for future use and enable it on a subsequent “set vpn” command.

authentication={none|md5|sha1}
The authentication algorithm used in authenticating clients.

none
No authentication. No authentication can be used to save time and CPU cycles. It is not as secure, but the peers were authenticated in phase 1.

md5
MD5 authentication, which uses 128-bit keys.

sha1
SHA1 authentication, which uses 160-bit keys.

encryption={none|des|3des|aes}
The encryption algorithm used for encrypting data.

none
No encryption is used. One use of IPsec is to tie to private networks together. If security is not a major concern, encryption can be disabled to save on processing and overhead.

des
DES encryption, which uses 64-bit keys.

3des
3-DES encryption, which uses 192-bit keys.

aes
AES encryption, which uses either 128-bit, 192-bit, or 256-bit keys depending on the negotiated security settings.

sa_lifetime=60-2^32
Determines how long a Security Association (SA) policy is active, in seconds. After the SA has been negotiated, the SA lifetime begins. Once the lifetime has completed, a new set of SA policies are negotiated with the remote VPN endpoint.

sa_lifetime_data=(0 - 2^32) (kilobytes)
The amount of data, in bytes or kilobytes, that is sent and received until the SA is renegotiated. This value is analogous to the SA lifetime. Also known as SA life size.

See also

- "display" on page 27. The “display sadb,” “display sp,” and “display vpn” commands display VPN-related connection and status information.
- "revert" on page 61. The “revert vpn” options revert groups of VPN settings, or all VPN settings.
- "show" on page 249.
- "vpn" on page 256. The “vpn” command is used to manage and display the status of VPN tunnels.
- The VPN settings in the Web user interface (**Network > Virtual Private Network (VPN) Settings**) and the online help for these settings.
- The *Digi Cellular Family User’s Guide* section titled “Virtual Private Network (VPN) Settings.”

set wlan

Devices supported

This command is supported in Digi Connect Wi-ME, Digi Connect Wi-EM, and Digi Connect Wi-SP devices.

Purpose

Configures wireless devices, or displays the status of wireless devices.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions s-wlan=read” to display wireless settings, and “set permissions s-wlan=rw” to display and change wireless settings. See "set permissions" on page 157 for details on setting user permissions for commands.

Configuring wireless settings

Following is information on how configuration choices for wireless devices, such as the authentication method, affect other configuration choices, such as encryption types and other “set wlan” command options.

Authentication methods and available encryption types

The following table shows the authentication methods available for wireless devices, and the encryption types that apply to each method. The Xs show the encryption types that can be used with each authentication method. At least one encryption type must be selected if a particular authentication method is selected.

Encryption Type:	Authentication Method:					
	Open	Shared Key	WEP authentication	WPA-PSK authentication	WPA	LEAP
Open	X	X				
WEP	X	X	X	X	X	X
TKIP				X	X	
CCMP				X	X	

Using "show wlan" to display authentication encryption methods

The "show wlan" command displays evaluation information about wireless LAN settings, including ineffective settings and a list of valid combinations. It displays whether encryption methods are specified and in use or not used by authentication methods, and whether setup of certain options appear to be complete. See the Examples section for "show" on page 249; for the results of "show wlan."

Authentication methods and associated data fields

The following table shows the authentication methods available for wireless devices, and the associated data fields, or command options, that apply to each method. All data fields with that have an X in a particular authentication method's column are required, except for trusted certificates, which is optional.

Data Fields:	Authentication Method:					
	Open	**Shared Key**	**WEP authentication**	**WPA-PSK**	**WPA authentication**	**LEAP**
WEP keys	X If WEP encryption is selected.	X				
Passphrase				X		
Authentication methods			X		X	
Username, password			X		X	X
Client certificate			X If TLS is selected.		X If TLS is selected.	
Trusted certificates			X		X	

Inner and outer protocols

The following table shows relationships between outer protocols and inner protocols specified on the “set wlan” command. Outer protocols are the types of Extensible Authentication Protocols (EAP) that are allowed to establish the initial connection with an authentication server or access point. The outer protocols are specified by the “outer_eap” option. Inner protocols are the types of protocols that are allowed to authenticate the device. These protocols are used within the encrypted connection established by PEAP or TTLs. The inner protocols are specified by the “inner eap” option.

Inner Protocols:	Outer Protocols:		
	PEAP	TLS	TTLS
GTC	X		
MD5	X		X
MSCHAPv2	X		X
OTP	X		X
TLS	X		X
CHAP			X
MSCHAP			X
MSCHAPv2			X
PAP			X

Syntax

Configure wireless settings

```
set wlan
   [protmode={bss|ibss_create|ibss_join|any}]
   [channel={0|1-14}]
   [ssid=string]
   [authentication={[open],[sharedkey],[wep_auth],[wpa_psk],
   [wpa_auth],[leap],[any]}
   [encryption={[open],[wep],[tkip],[ccmp],[any]}]
   [outer_eap={[peap],[tls],[ttls],[any]}]
   [inner_eap={[gtc],[md5],[mschapv2],[otp],[chap],[mschap],
   [ttls_mschapv2],[pap],[any]}]
   [options={[diversity],[short_preamble],[verify_cert]}]
   [username=string]
   [password=string]
   [psk=string]
   [wepmode={64bit|128bit}]
   [wepindex=1-4]
   [wepkeyN=hex string]]
   [country=string]
   [maxtxrate={1|2|5.5|11}] (Mbps)
   [txpower={6|8|10|12|14|16}] (dBm)
```

Display wireless settings

```
set wlan
```

Or:

```
show wlan
```

Options

Regarding command options “authentication encryption,” “outer_eap,” “inner_eap,” and “options:” These options have multiple values. More than one value may be specified for each option to indicate the set of allowed values. The actual value used will be determined by the capabilities of the wireless network.

protmode={bss|ibss_create|ibss_join|any}

Used to change the operation mode in which the device will work.

bss

Indicates that the device should join an access point.

ibss_create

Indicates the device will attempt to first join an Independent Basic Service Set (IBSS), and create one if it is unable to find one.

ibss_join

Indicates the device should attempt to join an IBSS or self-contained wireless network.

any

Enables all operation modes.

Typically, the operation mode is “bss.” The default is “bss.”

channel={0|1-14}

Sets the frequency channel that the wireless Ethernet radio will use. A value of 0 indicates that the device will scan all frequencies until it finds one with an available access point or wireless network it can join. The default value is 10.

ssid=*string*

Used to specify the identifier of the wireless network that the device should be joined to. The default is an empty string, which indicates that the first wireless network that the device finds will be joined to.

authentication=
{[open],[sharekey],[wep_auth],[wpa_psk],[wpa_auth],[leap],[any]}

The types of authentication that are allowed to establish a connection with the access point.

open
IEEE 802.11 open system authentication is used to establish a connection with the access point.

sharedkey
IEEE 802.11 shared key authentication is used to establish a connection with the access point. At least one WEP key must be specified to use shared key authentication.

wep_auth
IEEE 802.1x authentication (EAP) is used to establish a connection with an authentication server or access point. Wired Equivalent Privacy (WEP) keys are dynamically generated to encrypt data over the wireless link.

wpa_psk
The Wi-Fi Protected Access (WPA) protocol is used with a pre-shared key (PSK) that you specify to establish a connection with the access point and encrypt the wireless link.

wpa_auth
The WPA protocol and IEEE 802.1x authentication (EAP) is used to establish a connection with an authentication server or access point. Encryption keys are dynamically generated to encrypt data over the wireless link.

leap
Lightweight Extensible Authentication Protocol (LEAP) is used to establish a connection with an authentication server or access point. Wired Equivalent Privacy (WEP) keys are dynamically generated to encrypt the wireless link. A username and password must be specified to use leap.

any
Sets all authentication types.

encryption={[open],[wep],[tkip],[ccmp],[any]}
The types of encryption that are allowed to encrypt data transferred over the wireless link.

open
No encryption is used over the wireless link. Can be used with open and sharedkey authentication.

wep
Wired Equivalent Privacy (WEP) encryption is used over the wireless link. Can be used with open, sharedkey, wep_auth, wpa_psk, wpa_auth, and leap authentication.

tkip
Temporal Key Integrity Protocol (TKIP) encryption is used over the wireless link. Can be used with wpa_psk and wpa_auth authentication.

ccmp
CCMP (AES) encryption is used over the wireless link. Can be used with wpa_psk and wpa_auth authentication.

any
Sets all encryption types.

outer_eap={[peap],[tls],[ttls],[any]}
The types of Extensible Authentication Protocols (EAP) that are allowed to establish the initial connection with an authentication server or access point. These are used with wep_auth and wpa_auth authentication.

peap
Protected Extensible Authentication Protocol (PEAP). A username and password must be specified to use peap.

tls
Transport Layer Security (TLS). A client certificate and private key must be installed on the device to use tls.

ttls
Tunneled Transport Layer Security (TTLS). A username and password must be specified to use ttls.

any
Sets all outer and inner Extensible Authentication Protocols.

inner_eap={[gtc],[md5],[mschapv2],[otp],[chap],[mschap],[ttls_mschapv2],[pap,[any]]}

The types of protocols that are allowed to authenticate the device. These are used within the encrypted connection established by PEAP or TTLS.

The following are Extensible Access Protocols (EAP) that can be used with PEAP or TTLS:

gtc
Generic token card.

md5
Message Digest Algorithm (MD5).

mschapv2
Microsoft Challenge response Protocol version 2.

otp
One Time Password.

The following are non-EAP protocols that can be used with TTLS:

chap
Challenge response Protocol.

mschap
Microsoft Challenge response Protocol.

ttls_mschapv2
Microsoft Challenge response Protocol version 2.

pap
Password Authentication Protocol.

any
Sets all inner Extensible Authentication Protocols.

options={[diversity],[short_preamble],[verify_cert]}

diversity
Enable reception on multiple antennas on devices with this capability.

short_preamble
Enable transmission of wireless frames using short preambles, if allowed by the access point.

verify_cert
Verify that certificates received from an authentication server or access point are signed by a trusted certificate authority (CA). Standard CAs are built in, and additional trusted certificates may be added.

username=*string*
Used when the “security” option is set to “wpa_auth.” This option specifies the user name to be used during authentication.

password=*string*
Used when the “security” option is set to “wpa_auth.” This option specifies the password to be used during authentication.

psk=*string*
Used when the “security” option is set to “wpa_psk.” This option specifies a string that is converted into a pre-shared key (PSK) that is used for encryption.

wepmode={64bit|128bit}
Specifies the key size used when WEP encryption is enabled. The default is 64bit.

wepindex=1-4
Specifies which of the 4 possible keys will be used. The default is 1.

wepkeyN=*hex string*
A hexadecimal string that serves as the key if WEP encryption is enabled. The key consists of 26, 10, or 0 (zero) hexadecimal digit characters. If “wepmode=64bit”, the wepkey is 10 digits. If “wepmode=128bit”, the wepkey is 26 digits. A wepkey value of 0 length clears the value.

country=*string*
The country in which the device will be used. By selecting a country, the channel settings will be restricted to the legal set for that country.

maxtxrate={1|2|5.5|11} (Mbps)
The maximum transmission rate that the device will use, in megabits per second.

txpower={6|8|10|12|14|16} (dBm)
The wireless transmit power, in decibels relative to one milliwatt (dBm).

Example

```
#> set wlan wepkey1=ab12cd34ef567ab12cd34ef567 wepindex=1
#> set wlan wepmode=128bit
#> set wlan ssid="access point 1"
```

See also

- "revert" on page 61. The “revert wireless” option reverts the settings configured by this command.
- "show" on page 249. "show wlan" displays

show

Devices supported

This command is supported in all Digi Connect products.

Purpose

Displays the current settings in a device, including current configuration settings, boot code loaded in the device, and the effects of commands issued to the device.

Required permissions

For Digi Connect products with two or more users, for this command to display current device settings, the various “set” commands must have be set to either “read” or “r-self,” depending on the available permissions for the commands. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

`show` *`option`* `[port=`*`range`*`] [range=`*`range`*`]`

Options

option

Specifies which settings in the device to show. The following options can be specified. The use of the “port” and “range” options on the show command depends on whether the command that was used to configure the settings uses the “port” and “range” options as well. If you are viewing the PDF file of this document, click the “set” commands in the “Displays settings...” column to go to the command descriptions.

Option	Displays settings configured by	Works w/ port option	Works w/ range option
accesscontrol	set accesscontrol	No	No
alarm	set alarm	No	Yes
arp	The arp table. This option is not associated with a “set” command.	No	No
autoconnect	set autoconnect	Yes	No
bsc	set bsc	No	No
buffer	set buffer	Yes	No
ddns	set ddns	No	No
dhcpserver	set dhcpserver	No	Yes
ekahau	set ekahau	Yes	No
ethernet	set ethernet	No	No
forwarding	set forwarding	No	No
gpio	set gpio	No	Yes

Option	Displays settings configured by	Works w/ port option	Works w/ range option
group	set group	No	No
host	set host	No	No
ia master	set ia command: settings for an IA master; see "Configure network-based masters (set ia master)" on page 120.		No
ia serial	set ia command: settings for IA serial; see "Configure serial-port connected devices (set ia serial)" on page 120.		Yes
ia table	set ia command: settings for IA destination tables and route entries; see "Configure destination tables and route entries (set ia table)" on page 121.		No
login	set login	No	No
menu	set menu	No	Yes
mesh	set mesh		
mgmtconnection	set mgmtconnection	No	Yes
mgmtglobal	set mgmtglobal	No	No
mgmtnetwork	set mgmtnetwork	No	Yes
nat	set nat	No	No
network	set network	No	No
passthrough	set passthrough	No	No
permissions	set permissions	No	No
pmodem	set pmodem	Yes	No
pppoutbound	set pppoutbound	Yes	No
profile	set profile	Yes	No
putty	set putty	No	No
python	set python		
rciserial	set rciserial	No	No
realport	set realport	No	No

Option	Displays settings configured by	Works w/ port option	Works w/ range option
route	The IP routing table. This command is not associated with a “set” command.	No	No
rtstoggle	set rtstoggle	Yes	No
serial	set serial	Yes	No
service	set service	No	Yes
snmp	set snmp	No	No
socket_tunnel	set socket_tunnel	No	No
surelink	set surelink		
system	set system	No	No
tcpserial	set tcpserial	Yes	No
term	set term		
udpserial	set udpserial	Yes	Yes (when specifying UDP serial destinations)
user	set user	No	Yes
versions	This command shows firmware version information. It is not associated with a “set” command.	No	No
video	set video	No	No
vncclient	set vncclient	No	No
vpn	set vpn	No	No
wlan	set wlan This option displays an evaluation of saved wireless settings, including ineffective settings and a list of valid combinations. It displays whether encryption methods are specified and in use or not used by authentication methods, and whether setup of certain options appear to be complete (see Examples).	No	No

port=*range*
Identifies a particular serial port. Optional on a single-port device.

range=*range*
A configuration table entry or range of entries.

Examples

Display network configuration settings

```
#> show network

Network configuration:

MAC Address          : 00:40:9D:24:8B:B3

                       Currently in use by
                       the network stack      Stored configuration
                       --------------------   -------------------
ipaddress            : 10.8.16.8              192.168.4.25
submask              : 255.255.0.0            255.255.0.0
gateway              : 10.8.1.1               0.0.0.0
static               : off                    off
dhcp                 : supplied IP address    on
autoip               : on                     on
keepalive idle       : 7200                   7200
probe count          : 9                      9
probe interval       : 75                     75
garbage byte         : on                     on
override dhcp        : off                    off
dns1                 : 10.10.8.62             0.0.0.0
dns2                 : 10.10.8.64             0.0.0.0
rto_min              : 1000                   1000
rto_max              : 10                     10
arp_ttl              : 15                     15
garp                 : 3600                   3600
```

Display current alarm settings

```
#> show alarm
```

Display settings for a particular user

```
#> show user range=3
```

Display wireless settings

In addition to showing the current wireless settings, "show wlan" displays evaluation notes and warning messages about the effect and interaction of wireless settings. As the example shows, warning messages note encryption methods that have been defined but not used by any authentication methods, and notes identify whether configuration of certain features appears to be complete.

```
#> show wlan

Wireless LAN Configuration:


                   Active settings
                  (Stored settings)
                  --------------------
country          :  United States
protocol mode    :  any
channel          :  scan
ssid             :
maxtxrate(Mbps)  :  11
authentication   :  open
encryption       :  open,wep,tkip,ccmp
eap outer        :  peap,tls,ttls
eap inner        :  gtc,md5,mschapv2,otp,chap,mschap,ttls_mschapv2,pap
options          :
txpower(dBm)     :  14dbm
wepmode          :  64bit
wepindex         :  1
username         :

Evaluation of your saved wireless settings:

Warning: TKIP encryption specified but unused by any of the specified
   Authentication methods
Warning: CCMP encryption specified but unused by any of the specified
   Authentication methods
Note: Settings for Protocol Modes IBSS Create or IBSS Join appear
   to be complete
Note: Settings for Open Authentication appear to be complete
```

See also

- "revert" on page 61.
- The “set” commands (“set user,” “set network,” “set serial,” etc.). Entering a set command without any options displays the same information as that displayed by the “show” command.
- "set wlan" on page 241 for the encryption and authentication methods and protocol modes referenced in the "show wlan" output.

status

Devices supported

This command is supported in all Digi Connect products.

Purpose

Displays the current list of sessions. The “status” command displays the status of outgoing connections (connections made by “connect,” “rlogin,” or “telnet” commands). In contrast, the “display” command displays real-time information about a device, while the “info” command displays statistical information about a device over time. Typically, the “status” command is used to determine which sessions to close.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions status=read” or “set permissions status=rw” to use this command. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

```
status [range] [session number]
```

Options

range

The range of sessions to view.

session number

An index number identifying the session number to view.

Examples

```
#> status

  Connection: 3          From: 10.8.109.8

    Connection not associated with any sessions.
```

See also

- "connect" on page 22
- "close" on page 21, for information on ending a connection.
- "display" on page 27
- "info" on page 41.
- "rlogin" on page 66
- "telnet" on page 255
- "who" on page 258

telnet

Devices supported

This command is supported in all Digi Connect Family devices.

Purpose

Used to make an outgoing Telnet connection, also known as a session.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions telnet=execute” to use this command. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

```
telnet [options] ip addr [tcp port]
```

Options

options

The Telnet options for the command, which may be as follows:

binary={on|off}

Turns on or off Telnet binary mode.

crmod={on|off}

Turns on or off the replacement of the carriage-return character sequence (\r) with the new-line character sequence (\n) on incoming network data.

ip addr

The IP address of the host to which you want make a Telnet connection.

tcp port

The TCP port assigned the Telnet application on the remote system. The default is 23, the port typically used for Telnet.

Examples

Establish a Telnet session 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
```

Establish a Telnet session to a device server port from the LAN

In this example, a user on the LAN initiates a Telnet connection to port 4 on a device server.

```
#> telnet 192.192.150.28 2004
```

See also

- "rlogin" on page 66
- "connect" on page 22
- "close" on page 21
- "status" on page 254

vpn

Devices supported

This command is supported in Digi Cellular Family products only, except Digi Connect WAN.

Purpose

Manages and displays the status of a Virtual Private Network (VPN) tunnel.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions vpn=execute” to use this command. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

```
vpn [status|connect|disconnect]
   [index=1-2)]
   [name=tunnel name]
```

Options

status|connect|disconnect
Specifies the action performed by the command. Note that a data connection must be attempted before the VPN tunnel will actually be brought up.

status
Display the status of the VPN tunnel.

connect
Enable the VPN tunnel.

disconnect
Disconnect the VPN tunnel.

index=1-2
Identifies the VPN tunnel.

name=*tunnel name*
The name of the tunnel.

Example

Display VPN tunnel status

```
#> vpn status

 VPN Tunnel #1 Status :

    name           : Tunnel 1
    mode           : isakmp

    status         : down
    remote address: : 65.214.122.53
    mobile address: : not connected

 VPN Tunnel #2 Status :

    name           : Tunnel 2
    mode           : disabled
```

Enable a VPN tunnel

This command enables the tunnel at index 1 to be used.

```
#> vpn connect index=1
```

See also

- "display" on page 27. The “display sadb,” “display sp,” and “display vpn” commands display VPN-related connection and status information.
- "revert" on page 61. The “revert vpn” options revert groups of VPN settings, or all VPN settings.
- "set vpn" on page 228. The “set vpn” command configures VPN settings.
- The VPN settings in the Web user interface (**Network > Virtual Private Network (VPN) Settings**) and the online help for these settings.
- The **Connections Management** page in the Web user interface (from the Home page, click the **Connections link**) is the Web equivalent of this command.
- The Digi Cellular Family User’s Guide section titled “Virtual Private Network (VPN) Settings.”

who

Devices supported

This command is supported in all Digi Connect products.

Purpose

Displays active connections to and from the device.

Required permissions

For Digi Connect products with two or more users, permissions must be set to “set permissions who=execute” to use this command. See "set permissions" on page 157 for details on setting user permissions for commands.

Syntax

```
who
```

Options

None at this time.

Examples

Display a list of all current connections

```
#> who
```

See also

"kill" on page 49. The “kill” command is used to kill a connection.

Chapter 3 Modem Emulation Commands

This chapter describes the commands that can be issued when Digi Connect products are configured in modem emulation mode.

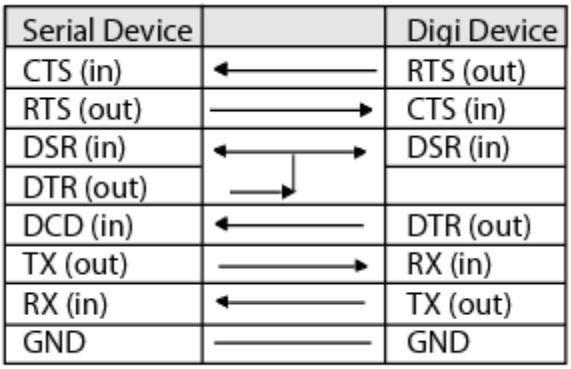
What Is Modem Emulation?

Modem emulation enables a system administrator to configure a networked Digi device to act as a modem. The Digi device emulates modem responses to a serial device and seamlessly sends and receives data over an Ethernet network instead of a PSTN (Public Switched Telephone Network). The advantage for a user is the ability to retain legacy software applications without modification and use a less expensive Ethernet network in place of public telephone lines.

As an aid in configuring modem emulation, the Digi Device Setup Wizard and the default web interface have a serial port profile for modem emulation.

Modem Emulation Cable Signals

Use the following signal assignments to make a cable connecting the Digi device to a serial device.



DSR and DTR on the serial device side are connected to the DSR signal of the Digi device.

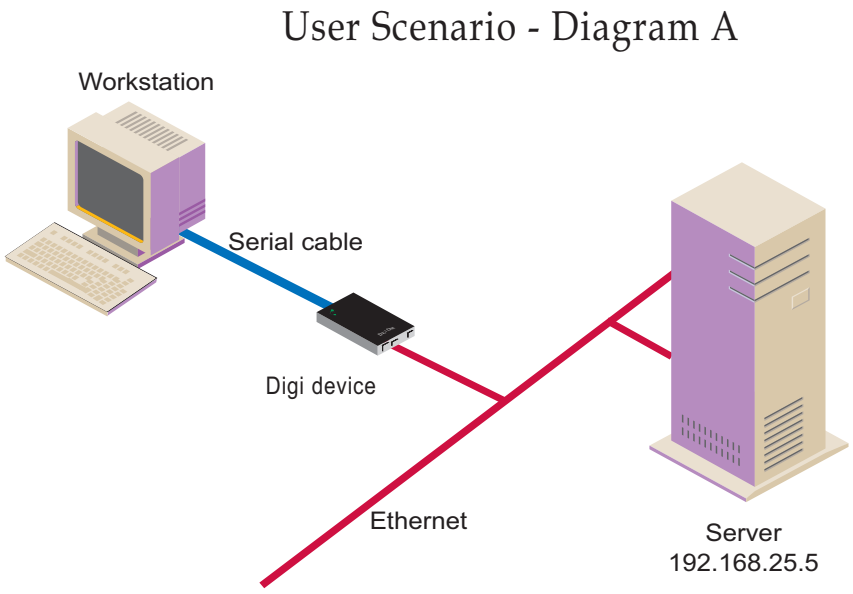
Modes of Operation

There are two modes of operation in modem emulation:

- Command mode: Issuing AT commands to a Digi device.
- Data mode: After a network connection is established, the device switches to data mode.

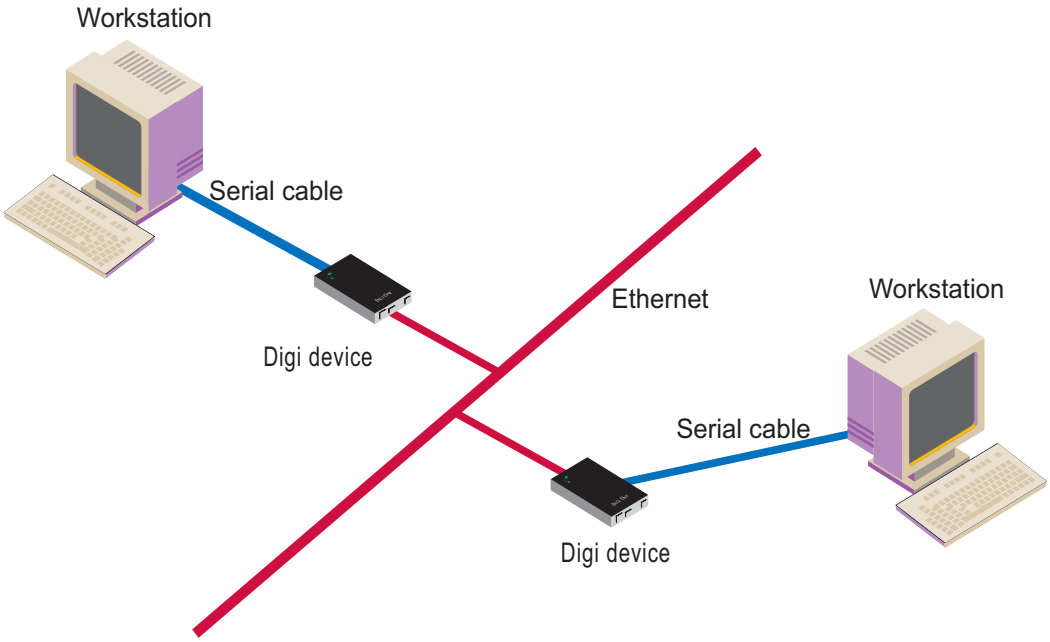
Common User Scenarios for Modem Emulation

The Digi device in modem emulation mode allows for the easy replacement of modems in almost any environment where there is a LAN or WAN.



In Diagram A, the Digi Connect device replaces a modem connected to a workstation running an application. The Digi Connect device allows for the use of software applications without modification by responding to all the AT commands configured in the workstation application. The Digi Connect device connects to the IP Address of the server when an `ATDT ipaddress:port` (`ATDT 192.168.25.5:50001`) command is issued. Once the remote device establishes the TCP connection, a CONNECT message is sent to the serial port and only then does the Digi device switch from AT command mode to data mode. Using the modem escape sequence or dropping DTR on either side terminates the connection. A DISCONNECT message will be sent to the application if the remote side closes the TCP connection.

User Scenario - Diagram B



In Diagram B, two Digi devices will replace modems on both sides of the connection. The initiation of the connection occurs with either of the Digi devices. If both ends are Digi devices, the TCP listening port number is 50001 for port 1. An example of the connection command is `ATDT 192.168.25.30:50001`. Upon establishing a successful TCP connection, a CONNECT message is sent to the serial port and only then does the Digi device switch from AT command mode to data mode. After the CONNECT is received, the transmission of data begins. Using the modem escape sequence or dropping DTR on either side terminates the connection.

Connection Scenarios for Modem Emulation

Modem emulation can involve the following types of connection scenarios:

Outgoing Modem Emulation Connection

In an outgoing modem emulation connection, a serial device sends an ATD*x.x.x.x:y* command, which triggers the Digi device to establish a connection to destination IP=*x.x.x.x*, port=*y*.

Incoming Modem Emulation Connection

In an incoming modem emulation connection, a device on the network connects to port 50001 (50000+1 = 1st serial port). This incoming connection triggers the Digi device to generate a RING on the serial port. The device attached to the serial port will answer the RING and the connection is established.

Modem Emulation Pooling

Modem emulation pooling is a combination of Incoming Modem Emulation Connection and a hunt group. A device on the network connects to port 50000. The Digi device checks if a serial port configured for modem emulation is available. If so, it connects to the port, otherwise returns an error.

Modem Emulation Bridge

A modem emulation bridge is combination of Outgoing and Incoming Modem Emulation Connections, in which both serial devices require to talk to a modem. The first serial device connects to the second device using ATDx.x.x.x:y, the second device gets a RING and accepts the incoming connection.

About the Commands in this Chapter

This chapter describes the Digi-specific modem emulation commands that have been implemented for Digi Connect devices. It is divided into several sections:

- The AT command set. These are commands to perform actions in a modem-emulation connection.
- Modem S-Register definitions.
- A description of the result codes for the commands.

Accepted But Ignored AT Commands

Any other commands not described in this chapter but in the standard AT command set are accepted but ignored and therefore have no effect. Such commands are pertinent to actual modems, but not to modem emulation.

Modem Emulation AT Command Set

The following commands can be issued to perform actions in a modem-emulation configuration scenario.

AT Command	Function	Result Code
n+++*n*	When in data mode, this command causes the modem to switch to command mode. The value of *n* corresponds to the required delay before and after the escape sequence is entered. The delay can be changed by modifying S-register 12. The escape character can be changed by modifying S-register 2.	
A/	Repeats the last command string.	
AT?	Prints the value of the last-accessed S-register.	
ATA	Answer command: Answers an incoming TCP connection and switches to data mode.	
ATD (*ipaddress*): (*ipport*)	Used to connect to a remote network device. This command directs the Digi device to go on-line, dial according to the IP address entered as follows, and attempt to establish a TCP connection. Dial Modifiers. The valid dial string parameters are described below. Punctuation characters may be used for clarity with parentheses, hyphen, and spaces being ignored. • 0-9: DTMF digits 0 through 9. • . (period): Dot notation used for IP addresses. IP addresses are written as four numbers separated by periods, where the first number is between 1 and 255, and the other three numbers are between 0 and 255. Enter the IP address in the format *xxx.xxx.xxx.xxx* • : (colon): Colon notation used for the TCP port. • L: Redial the last number. The modem will reconnect to the last IP address accessed. The L must immediately follow the D, and any following characters are ignored. • P: This command is accepted but not acted on. • T: This command is accepted but not acted on. • R: This command is accepted but not acted on. • , (comma): This command is accepted but not acted on.	
ATE*n*	Command echo. The Digi device enables or disables the echo of characters to the DTE according to the parameter supplied. The parameter value, if valid, is written to S14 bit 1. • E0: Disables command echo. • E1: Enables command echo.	OK n=0 or 1 ERROR Otherwise
ATH	Disconnect (Hang up) command. H0, H1: Hangs up the TCP connection if a connection is active.	OK n=0 or 1 ERROR Otherwise
ATI*n*	Identification command. • I0, I1: Reports product name. • I3: Reports product name, firmware revision. • I4: Reports product configuration. • I6: Reports network connection information.	OK n=0 or 9 ERROR Otherwise

AT Command	Function	Result Code
ATO	Return to on-line data mode. If the modem is in the on-line command mode, the modem enters the on-line data mode. If the modem is in the off-line command mode (no connection), ERROR is reported. • O0, O1: If there is an active connection, switches the modem to data mode.	OKn = 0 or 1 and a connection exists. ERROR Otherwise or if not connected.
ATQ*n*	Quiet results codes control command. The command enables or disables the sending of the result codes to the DTE according to the parameter supplied. The parameter value, if valid, is written to S14 bit 2. • Q0: Enables result code to the DTE (Default). • Q1: Disables result code to the DTE. • Q2: Disables “CONNECT” result codes. • Q3: Disables “CONNECT” result codes on incoming connections.	OK n=0 or 1 ERROR Otherwise
ATS*n*	Read/Write to the specified S-Register. • *n* Establishes S-register *n* as the last register accessed. • *n*=*v* Sets S-Register *n* to the value *v*. • *n*? Reports the value of S-Register *n*. See "S-Register Definitions" on page 267 for definitions of S-Registers.	OK n=0 or 1 ERROR Otherwise
ATV*n*	The verbose setting for result codes. This command selects the sending of short-form or long-form codes to the DTE. The parameter, if valid, is written to S14 bit 3. • V0: Result codes are issued in numeric or short form. Line feeds are not issued before a short-form result. • V1: Result codes are issued in text or long form. This is the default.	OK n=0 or 1 ERROR Otherwise
ATZ	Load configuration. Reloads the S-register configuration from flash memory. See "S-Register Definitions" on page 267 for definitions of S registers.	OK n=0 or 1 ERROR Otherwise
AT&C*n*	DCD option. The Digi device controls the DCD output in accordance with the parameter supplied. The parameter value, if valid is written to S21 bit 5. • &C0: DCD remains ON at all times. • &C1: DCD follows the state of the connection.	OK n=0 or 1 ERROR Otherwise
AT&D*n*	DTR option. This command interprets the ON to OFF transition of the DTR signal from the DTE in accordance with the parameter supplied. The parameter value, if valid, is written to S21 bits 3 and 4. Also see S25. • &D0: DTR drop is ignored (assumed ON). • &D1: DTR drop is interpreted by the modem as if the asynchronous escape sequence had been entered. The modem returns to command mode without disconnecting. • &D2: DTR drop causes the modem to hang up. (Default.) • &D3: DTR drop causes the modem to do a soft reset, as if the ATZ command was executed.	OK n=0 to 3 ERROR Otherwise
AT&F	Restore factory configuration. The device reloads the factory default S-register configuration from flash memory. The factory defaults are identified for each command and in the S-Register descriptions. A configuration consists of a subset of S-Registers.	OK n=0 or 1 ERROR Otherwise

AT Command	Function	Result Code
AT&V	Displays current values and settings. • AT&V0- AT&V5: Displays S-Register/command values for the current and stored configuration. • AT&V6: Displays current network settings.	OK n=0 to 5 ERROR Otherwise
AT&W*n*	Store configuration. Stores the specified S-registers in flash memory.	OK n=0 or 1 ERROR Otherwise

S-Register Definitions

Following is a description of the S-registers that can be set.

Register	Function	Range	Units	Default
S0	Rings to Auto-Answer. Sets the number of rings required before the Digi device automatically answers a call. Setting this register to Zero disables auto-answer mode.	0-255	Rings	0
S1	Ring Counter. Specifies the current number of rings. S1 is incremented each time the modem detects a ring signal on the telephone line. S1 is cleared when the existing connection is established or dropped.	0-255	Rings	0
S2	Escape Character. S2 holds the value of the ASCII character used as the escape character. The default value corresponds to an ASCII '+'. A value over 127 disables the escape process. That is, no escape character will be recognized.	0-255	ASCII	43
S3	Carriage Return Character. Sets the value of the carriage return character used when displaying commands or results.	0-127	ASCII	13
S4	Line Feed Character. Sets the character recognized as a line feed when displaying commands or results. If verbose result codes are used, the Line Feed control character is output after the Carriage Return control character.	0-127	ASCII	10
S5	Backspace Character. Sets the character recognized as a backspace, used to erase the last character typed on the command line.	0-32	ASCII	8
S12	Escape Prompt Delay. The amount of time required before and after an escape sequence (+++) is entered in order for the modem to transition from data mode to command mode.	0-255	0.02 second, 20 ms	50 1 second
S14	General Options Status. Indicates the status of command options. • Default: 138 (8Ah) (10001010b) • Bit 0: Ignored. • Bit 1: Command echo (E*n*): 0 = Disabled (E0). 1 = Enabled (E1). (Default.) • Bits 2 and 4: Quiet mode (Q*n*): 0 = Display result codes (Q0). (Default.) 1 = Do not display result codes (Q1). 2 = Disables “CONNECT” result codes (Q2). 3 = Disables “CONNECT” result codes on incoming connections (Q3). • Bit 3: Result codes (V*n*): 0 = Display numeric result codes (V0). 1 = Display verbose result codes (V1). (Default.) • Bits 5-7: Ignored.			138 (8Ah)

Register	Function	Range	Units	Default
S21	General Options Status. Indicates the status of command options. • Default: 52 (34h) (00110100b) • Bits 0 - 2: Ignored. • Bits 3-4: DTE's DTR behavior (&D*n*): 0 = DTR drop is ignored (&D0). 1 = DTR drop causes a transition from data to command mode without hanging up an existing connection (&D1). 2 = DTR drop hangs up the existing connection (&D2) (Default.) 3 = DTR drop causes the modem to do a soft reset if the ATZ command was executed (&D3). • Bit 5: Modem's DTR behavior: 0 = The modem's DTR remains on at all times (&C0). 1 = The modem's DTR follows the state of the TCP connection (&C1). (Default.) • Bits 6-7: Ignored.	-	-	52 (34h)
S25	Delay to DTR Off. The amount of time that the modem will delay before taking the action specified by the AT&D*n* command.	0-255	s or 0.01 s	5

Result Codes

Following is a description of the return codes returned by modem emulation commands.

Short	Long Form		Short	Long Form		Short	Long Form
0	OK		13	CONNECT 7200		84	CONNECT 33600
1	CONNECT		14	CONNECT 12000		91	CONNECT 31200
2	RING		15	CONNECT 14400		165	CONNECT 32000
3	NO CARRIER		16	CONNECT 19200		166	CONNECT 34000
4	ERROR		17	CONNECT 38400		167	CONNECT 36000
5	CONNECT 1200		18	CONNECT 57600		168	CONNECT 38000
6	NO DIALTONE		19	CONNECT 115200		169	CONNECT 40000
7	BUSY		20	CONNECT 230400		170	CONNECT 42000
8	NO ANSWER		59	CONNECT 16800		171	CONNECT 44000
9	CONNECT 0600		61	CONNECT 21600		172	CONNECT 46000
10	CONNECT 2400		62	CONNECT 24000		173	CONNECT 48000
11	CONNECT 4800		63	CONNECT 26400		174	CONNECT 50000
12	CONNECT 9600		64	CONNECT 28800			

Index

S

T

U

V

W

X