RealPort
on
SCO OpenServer

Setup Guide

92000317A

Contents

chapter 1

About This Guide

In This Chapter

Purpose

Use this guide for the following:

- An introduction to RealPort, how to install, configure and use RealPort and, how to troubleshoot should you encounter a problem
- Configuration examples
- Administration procedures

Audience

This manual is intended for the person responsible for configuring and administering RealPort. This person should have experience configuring network devices and be familiar with networking concepts.

Conventions

Following are the conventions used in this document:

Special Fonts

A special font is used for any input you need to enter. For example,

```
set config
```

Square Brackets

Optional parameters are displayed within square brackets. For example,

```
set config [dhcp=on]
```

Note: The square brackets themselves are not actually part of the command, and should not be entered.

Italics

Variables are displayed in italics. For example,

```
set config ip=ip-address
```

Note: Substitute an appropriate IP address for *ip-address* in the above command.

Vertical Bar

A vertical bar character (|) is used to denote a choice (logical "or"). For example,

```
set flow=on|off
```

Note: The above command would be entered as either:
`set flow=on` or `set flow=off`

chapter 2

Planning and Requirements

In This Chapter

Requirements and Considerations

Read this section before beginning the RealPort driver installation.

- The SCO OpenServer RealPort driver is compatible with SCO OpenServer System V Release 3.2 and SCO OpenServer. The SCO OpenServer RealPort driver is not compatible with SCO UnixWare. SCO UnixWare is supported by the SVR4 driver.
- Installing the RealPort software on a SCO OpenServer host where RealPort has already been installed will overwrite the existing RealPort configuration on that host. If a PortServer is added to an existing RealPort configuration, the software must be reinstalled and all the Digi devices configured again. It is best to configure extra Digi devices during the initial RealPort configuration. By doing this, additional Digi devices can be connected later without reinstalling the driver.
- The Digi RealPort driver for SCO OpenServer is an installable device driver. The software development system is not required to create a new kernel.

Information to Gather

Before you install RealPort, you need to determine:

- The hostname or IP address to assign each unconfigured Digi device.
- The number of ports for each Digi device. Include any Xem PORTS Module ports that are attached to the Digi device.
- Determine the link speed between the host machine and the Digi device if a slow WAN link (for example, a 56K leased line) connects them.

Configuring a Digi One RealPort, Digi One IA RealPort, or PortServer for RealPort

Use this procedure to configure a Digi One RealPort, Digi One IA RealPort, or PortServer for use with Digi RealPort drivers.

Note: See the appropriate Command Reference or Configuration and Administration Guide for information on how to perform the steps below.

Procedure

1. Access a root prompt on the Digi device.
2. Set the Digi device's IP address.
3. Verify that the RealPort TCP port number is set to 771. Change if necessary.
4. For all ports that will be using RealPort, set the device type to rp. The following example configures ports 2 through 16 of a PortServer for RealPort:

```
set ports range=2-16 dev=rp
```

Configuring EtherLite for RealPort

Use this procedure to install and configure an EtherLite Terminal Server for use with Digi RealPort drivers.

Procedure

1. Attach the EtherLite module to the network using an appropriate cable.
2. Assign an IP address to the EtherLite module (see the following topic).

Assigning an IP Address

EtherLite IP addresses may be set by either of two methods:

- dgipserv, a UNIX utility for setting EtherLite IP addresses (see below)
- EtherLite Boot Console (see the EtherLite Administration Card)

Using dgipserv to Assign an IP Address

dgipserv is a Digi utility used to set the IP addresses for Digi devices. It will also set the gateway and subnet mask addresses when needed to upgrade the Digi device's firmware. For more information on dgipserv consult the man pages.

Note: dgipserv will only work with firmware version 7.9 or later.

Procedure

1. Enter the command:

   ```
   dgipserv
   ```

2. To store an IP addresses in your Digi device enter:

   ```
   dgipserv -store MAC_address(IP_address)
   ```

   where the MAC address and the IP address are the addresses of the Digi device. An example of this command is:

   ```
   dgipserv -store 10:e0:f7:15:20:8g 143.182.5.63
   ```

Related Documentation

PortServer Documentation

- PortServer Configuration and Reference Guide
- Digi One/PortServer Cable Guide

PortServer II Documentation

- PortServer II Hardware Installation Guide
- PortServer II Command Reference
- PortServer II Configuration and Administration Guide
- Digi One/PortServer Cable Guide
- Digi Port Authority--Remote Device Monitor Setup Guide

PortServer TS 8/16 Documentation

- PortServer TS 8/16 Command Reference
- PortServer TS 8/16 Configuration and Administration Guide
- Digi One/PortServer Cable Guide
- Digi Port Authority--Remote Device Monitor Setup Guide

Digi One/PortServer TS 2/4 Documentation

- Digi One/PortServer TS 2/4 Quick Reference Card
- Digi One/PortServer TS 2/4 Command Reference
- Digi One/PortServer TS 2/4 Configuration and Administration Guide
- Digi One/PortServer Cable Guide
- Digi Port Authority--Remote Device Monitor Setup Guide

Digi One IA RealPort Documentation

- Digi One/PortServer TS 2/4 Command Reference
- Digi One/PortServer TS 2/4 Configuration and Administration Guide
- Digi One/PortServer Cable Guide
- Digi Port Authority--Remote Device Monitor Setup Guide

EtherLite Documentation

- EtherLite Hardware Information Guide
- EtherLite Administration Card
- EtherLite Cable Guide

chapter 3

RealPort Setup

In This Chapter

Master Setup Process

Use this process to install and configure the SCO OpenServer RealPort driver.

1. Read any release notes that may be included with the installation media.
2. Gather information necessary for installation. See Information to Gather, on page 2-2.
3. Install the RealPort driver. See Installing the Driver: CD, on page 3-2, or Installing the Driver: Downloaded, on page 3-2.
4. Configure the PortServer for RealPort. See Configuring a Digi One RealPort, Digi One IA RealPort, or PortServer for RealPort, on page 2-2, or Configuring EtherLite for RealPort, on page 2-3.

Installing the Driver: Downloaded

Use this procedure to install a downloaded SCO OpenServer driver.

Procedure

1. Download the appropriate driver package file.
2. Copy the downloaded package file to a directory on the SCO OpenServer system.
3. Log in as root on the SCO OpenServer system.
4. Change to the directory into which you copied the package file.
5. Enter the following command:

   ```
   custom -i -p misc:ncxa -F `pwd`/file_name
   ```

   where *file_name* is the name of the downloaded driver package.
6. Follow the prompts.
7. Reboot the system when the installation is complete.

Uninstalling RealPort

The Digi device driver software can be removed from the kernel by using custom. The Development Kit is not required, although the link kit must be installed.

1. Log onto the console as super-user (root), and enter this command: custom
2. Choose Software > Remove from the menu that appears.
3. Choose the DigiWARE for RealPort Driver.

   When prompted to remove Digi device entries from /etc/ttytype, enter <y> to remove the entries. Enter <n> to preserve the entries.

   When prompted to boot the new kernel by default, enter <y> for the new kernel to boot by default. Enter <n> if there is a specific reason for this new kernel not to boot by default.
4. When prompted to rebuild the kernel environment, enter <y>.
5. Exit custom.
6. Enter the following command to shut the system down and boot the new kernel:

```
shutdown -y -g0
```

RealPort Devices

The RealPort installation creates four different devices for each port in the /dev directory.

- Standard device
- Modem device
- Standard transparent print device
- Modem transparent print device.

Device Name Format

The devices are named according to the following conventions:

a-z	The Digi device letter ID.
01-64	The port number on the individual Digi device.
lowercase letter	At the end of a device name indicates a standard device.
uppercase letter	At the end of a device name indicates a modem device.

Here are examples of the devices for the first port of a Digi device with the identification letter "a":

Device	Path/Name Example
Standard Device	/dev/ttya01
Modem Device	/dev/ttyA01
Standard Transparent Print Device	/dev/pra01
Modem Transparent Print Device	/dev/prA01

Standard Device

Example: /dev/ttya01

A standard Digi device uses a lowercase identification letter in their names. The default handshake method is XON/XOFF. Data Carrier Detect (DCD) need not be present to open the device.

Once a connection is established and DCD becomes active, standard devices behave in the same way as modem devices--subsequent loss of the Data Carrier Detect signal will cause active processes on the port to be killed and the user will be automatically logged off.

Modem Device

Example: /dev/ttyA01

Digi modem devices use an uppercase identification letter in their names. They are sometimes referred to as "dial-in" devices.

Modem devices are traditional Unix ports with modem control. They require Data Carrier Detect to be high before they will operate. RTS/CTS handshaking is enabled by default. When used with a modem, the ports will wait for carrier before sending out the login: prompt

When used with a terminal or other device, it is usually wise to wire the Digi DCD signal to the terminal's Data Terminal Ready DTR) line. When the terminal is turned on, the system outputs a login prompt. When the terminal is turned off, any associated jobs are killed, and the user is logged out.

Transparent Print Devices

Examples: /dev/pra01 and /dev/prA01

The transparent print devices can be used with auxiliary printer ports on terminals. Output directed to a pr device goes out the auxiliary port of a terminal while you continue to use the terminal normally.

Use pra01 if the terminal device is ttya01 (standard device); use prA01 if the terminal device is ttyA01 (modem device).

Configuring a Device for a Terminal

Use this procedure to configure a Digi serial device for a terminal. See your operating system documentation for more information on configuring a serial device for a terminal.

Procedure

1. Connect a serial cable between the port and terminal.
2. Edit the /etc/inittab file on the SCO OpenServer system. Change the device to the proper speed, data bits, stop bits and parity bit. Do this by editing the letter or number at the end of the line for the port (the default is m, which corresponds with 9600 8,1,N). See /etc/gettydefs for /etc/inittab letter and number definitions.
3. Enable the port for login by entering this command at a prompt:

   ```
   enable device
   ```

where *device* is the name of the Digi device.

Configuring a Device with scoadmin

Use this procedure to configure Digi serial ports with the SCO OpenServer system administration utility.

Procedure

1. Access the system administration utility either by accessing the system administration window, or by entering this command at a root prompt:

   ```
   scoadmin
   ```

   Note: For older versions of SCO OpenServer, use the command sysadmsh.
2. Configure the ports as necessary. Consult your operating system documentation for detailed information on scoadmin.

Configuring a Device for a Printer

Use this procedure to configure a Digi serial device for a printer. Consult your operating system documentation for more information on configuring a serial device for a printer.

Procedure

1. Connect a proper cable between the port and printer.
2. *If you are configuring EtherLite ports, skip this step.*

   If you are using 8-pin RJ-45 cabling from the Digi serial port to the modem, enable altpin by inserting a ditty command or altpin option to an existing ditty command for the port in the file /dev/rc.d/8/userdef.

   Example

   ```
   ditty altpin device_name
   ```

   where *device_name* is the name of the Digi device.

   Note: By inserting the command in the startup file, it will execute every time the system is booted. For the altpin option to take effect immediately, enter the ditty altpin command at a UNIX prompt.
3. Insert the following command at the end of the file /etc/rc.d/8/userdef:

   ```
   ditty printer device_name
   ```

   This command forces the specified port to stay open, thus preserving any "non-sticky" options. Substitute your device name in the command.
4. Insert any other necessary ditty commands at the end of the file /etc/rc.d/8/userdef. The command ditty can be used to change the settings of a port. For additional information on ditty, see Setting TTY Options, on page 3-9.

   For example, the following command will set the port ttya01 to a speed of 38400, disable software flow control, and enable hardware flow control on the port:

   ```
   ditty 38400 -ixon -ixany -ixoff ctspace device_name
   ```

   By placing the ditty command in /etc/rc.d/8/userdef, it will be executed whenever the system is booted.
5. If you wish to set up a print spooler, enter the following command at a root prompt:

   ```
   scoadmin printer
   ```

   See your SCO OpenServer documentation for information on scoadmin.
6. Reboot the system or enter the ditty command manually for the new settings to take effect.

Configuring a Device for a Modem

Use these procedures to configure a Digi serial device for a Dial-in/Dial-out modem connection. Configuring a device for a modem requires familiarity with both the operating system and the modem being used. While the following procedure is sufficient for most cases, it may be necessary to take additional steps to properly configure your modem or to set up the operating system for a specific application.

Procedure: Configuring the UNIX Device

1. Connect a serial cable between the port and modem.
2. Power the modem on.
3. *If you are configuring EtherLite ports, skip this step.*

   If you are using 8-pin RJ-45 cabling from the Digi serial port to the modem, enable altpin by inserting a ditty command (or adding the altpin option to an existing ditty command) in the file /dev/rc.d/8/userdef.

   Example

   ```
   ditty altpin device
   ```

where *device* is the name of the Digi device.

Note: By inserting the command in the startup file, it will execute every time the system is booted. For the altpin option to take effect immediately, enter the ditty altpin command at a SCO OpenServer prompt.

4. Edit the /etc/inittab file on the SCO OpenServer server. Change the upper-case device to the letter o by editing the letter or number at the end of the line for the device. See /etc/gettydefs for inittab letter and number definitions.

5. Disable the modem control and non modem control devices for the port by entering:

```
disable modem-control-device
disable non-modem-control-device
```

where *modem-control-device* is the name of the modem control device for the port and *non-modem-control-device* is the name of the non modem control device for the port.

6. Edit the file /usr/lib/uucp/Devices. Add a Direct line for the device like this:

```
Direct ttyh1a - 38400 direct device
```

The "D" in Direct must be the first character on the line.

Procedure: Configuring Modems for Dial-in

1. Comment out any ACU lines relating to this device until dial-in is working properly.

Some applications may require an ACU entry in the Devices file. If an application requires an entry, refer to the modem manual or to the modem manufacturer for the correct modem initialization string, and to the SCO OpenServer Administration handbook for guidance on setting up an appropriate ACU entry that uses this initialization string.

2. Connect to the modem. At a root prompt, enter this command

```
cu -l /dev/device -s 38400
```

where *device* is the name of the non modem control device for the port.

3. Set the modem to answer after the first ring with this command:

```
ats0=1
```

4. Enter the following command to train the modem to the port speed:

```
at&w
```

5. Enter any other desired modem commands.

6. Disconnect from the modem by entering a tilde and a period:

```
~.
```

7. Configure flow control on the port. Hardware flow control is typical and can be configured by inserting this command at the end of the file /etc/rc.d/8/userdef:

```
ditty rtspace ctspace -ixon device
```

Note: The ditty command is not supported for EtherLite devices. Use standard operating system tools (scoadmin or stty) to set device options for EtherLite devices.

8. To add dial-in capability, enter this command at a root prompt:

```
enable device
```

In SCO OpenServer, the uppercase device name is for dial-in, the lower-case name is for dial-out and modem configuration.

Transparent Printer Setup

About Transparent Printing

Most terminals have an auxiliary port that can be connected to a serial printer. When this port is configured as a transparent printer port, print jobs may be run simultaneously with normal terminal operation.

Data bound for the printer is preceded by a terminal escape sequence which turns on transparent printing, and followed by a sequence which turns transparent printing off.

Set up a transparent printer in the same way you would set up a printer wired directly to a serial port. Data sent to a transparent printer device is automatically "wrapped" in the transparent print on/off command strings for the specified printer.

Configuring Transparent Printers

Use this procedure to set up transparent printers on terminals.

Note: This procedure sets up the communication characteristics for transparent printers. Once a transparent printer has been set up, you can use it as you would a printer connected directly to a serial port. See your UNIX documentation for information on setting up print queues.

Procedure

At the command prompt, enter:

```
ditty -n ttyname [options]
```

where *ttyname* is the name of the terminal device and options are selected from the list below..

maxcps *n*	Limits the maximum printer port character-per-second data rate. *n* should be set to the minimum character rate the printer can sustain in typical use.
maxchar *n*	Limits the number of characters queued to the printer ahead of terminal output. Lower numbers increase system overhead, higher numbers result in keystroke echo delays. A value of 50 is generally a good compromise at 9600 baud.
bufsize *n*	This parameter should be set to a value just below the printer's buffer size. After a period of inactivity, the driver will burst up to this many characters to the printer to fill the print buffer before slowing to the maxcps rate.
onstr "*s*"	Defines the terminal escape sequence to direct subsequent data to the transparent printer. *s* is a string of ASCII characters, enclosed in quotes, that command the terminal to enter transparent printing mode. An arbitrary octal character xxx may be given as \xxx. For example, the sequence "<Esc>[5i" would be entered as: "\033[5i".
offstr "*s*"	Defines the terminal escape sequence to stop directing data to the printer. *s* is a string of ASCII characters, enclosed in quotes, that command the terminal to enter transparent printing mode. An arbitrary octal character xxx may be given as \xxx. For example, the sequence "<Esc>[4i" would be entered as: "\033[4i".
term *t*	Sets the transparent printer on/off strings to values found in the internal default table. Internal defaults are used for the following terminal types: adm31, ansi, dg200, dg210, hz1500, mc5, vt100, vt220, vt320, vt420, wang2x36, wyse30, wyse50, wyse60 or wyse75. If the terminal type is not found in the internal default table, then ditty reads the terminfo entry for the terminal type and sets the transparent print on/off strings to the values given by the mc5/mc4 attributes found there.

Setup Examples for Transparent Printing

Use the ditty program to set up your terminal for transparent printing.

Example 1:

The following command configures the transparent print options for a DEC VT100 terminal connected to /dev/ttya01 (note that the printer will be called /dev/pra01). maxcps, maxchar and bufsize are left to defaults. Enter:

```
ditty term vt100 /dev/ttya01
```

Example 2:

The following example uses onstr and offstr arguments (this sets the terminal to use ANSI Standard). Again maxcps, maxchar, and bufsize are defaults. Enter (on a single command line):

```
ditty onstr \033[5i offstr \033[4i /dev/ttya01
```

Example 3:

This example command sets the transparent print option for a WYSE30 terminal, with maxcps of 75, a maxchar of 100, and a printer buffer size, bufsize, of 1000. Enter (on a single command line):

```
ditty term wyse30 maxcps 75 maxchar 100 bufsize 1000 /dev/ttya01
```

Setting TTY Options

RealPort UNIX device driver packages include a command, ditty, which is a superset of stty, and may be used to set and display the device options for Digi RealPort devices.

The general command format is:

ditty [-a] [-n *ttyname*] [*option(s)*] [*ttyname*]

With no options, ditty displays all Digi special driver settings, modem signals, and all standard parameters displayed by stty(1) for the TTY device referenced by standard input.

Command options are provided to change flow control settings, set transparent print options, force modem control lines, and display all TTY settings. Any unrecognized options are passed to stty(1) for interpretation.

ditty commands may be executed from the command line, or placed in a startup script to be run whenever the system is booted.

The options are:

-a	Display all of the unique Digi option settings, as well as all of the standard TTY settings reported by stty -a.
-n *ttyname*	Set and display options for the given TTY device, instead of standard input. This option may be specified multiple times to perform the same operation on multiple TTYs.
ttyname	Set and display options for the specified TTY device. Replace *ttyname* with the TTY pathname (such as /dev/ttya01s, /dev/term/a01 or /dev/dty/a001s, depending on your operating system). This option may be used on a modem control line when no carrier is present.

The following options specify transient actions to be performed immediately:

break	Send a 250 MS break signal out on the TTY line.
clear	Clear the tbusy flag in the channel structure to free the port.
flush	Immediately flush (discard) TTY input and output.
flushin	Flush TTY input only.
flushout	Flush TTY output only.

The following options specify actions which are not sticky, meaning that the changes are cancelled when the device is closed, and that the device will use the default values the next time it is opened.

stopout	Stop output exactly as if an XOFF character were received.
startout	Restart stopped output exactly as if an XON character were received.
stopin	Activate flow control to stop input.
startin	Release flow control to resume stopped input.
[-]dtr	Raise [drop] the DTR modem control line, unless DTR hardware flow control is selected.
[-]rts	Raise [drop] the RTS modem control line, unless RTS hardware flow control is selected.

The following options are sticky—the effects continue until the system is rebooted or until the options are changed.

[-]fastbaud	Alter the baud rate tables to permit the use of data rates that are beyond the range supported by the operating system. See fastbaud Data Rate Mapping, on page 3-11.
[-]rtspace	Enable [disable] RTS hardware input flow control, so RTS drops to pause remote transmission.
[-]ctspace	Enable [disable] CTS hardware output flow control, so local transmission pauses when CTS drops.
[-]dsrpace	Enable [disable] DSR hardware output flow control, so local transmission pauses when DSR drops.
[-]dcdpace	Enable [disable] DCD hardware output flow control, so local transmission pauses when DCD drops.
[-]dtrpace	Enable [disable] DTR hardware input flow control, so DTR drops to pause remote transmission.
[-]forcedcd	Disable [re-enable] carrier sense, so the TTY may be opened and used even when carrier is not present.
startc *c*	Sets the XON flow control character. The character may be given as a decimal, octal or hexadecimal number. Octal numbers are recognized by the presence of a leading zero, and hexadecimal numbers are denoted by a leading "0x". For example, the standard XON character, <CTRL-Q>, can be entered as "17" (decimal), "021" (octal) or "0x11" (hexadecimal).
stopc *c*	Sets the XOFF flow control character. The character may be given as a decimal, octal, or hexadecimal number (see startc, above, for format of octal and hexadecimal numbers).
astartc *c*	Sets auxiliary XON flow control character. The character may be given as a decimal, octal, or hexadecimal number (see startc, above, for format of octal and hexadecimal numbers).
astopc *c*	Sets auxiliary XOFF flow control character. The character may be given as a decimal, octal, or hexadecimal number (see startc, above, for format of octal and hexadecimal numbers).
[-]aixon	Enables auxiliary flow control, so that two unique characters are used for XON and XOFF. If both XOFF characters are received, transmission will not resume until both XON characters are received.
maxcps *n*	Sets the maximum Characters Per Second (CPS) rate at which characters are output to the transparent print device. The rate chosen should be just below the average print speed. If the number is too low, printer speed will be reduced. If the number is too high, the printer will resort to flow control, and user entry on the terminal will be correspondingly impaired. Default is 100 CPS.
maxchar *n*	Sets the maximum number of transparent print characters the driver will place in the output queue. Reducing this number increases system overhead; increasing this number delays operator keystroke echo times when the transparent printer is in use. Default is 50 characters.
bufsize *n*	Sets the driver's estimate of the size of the transparent printer's input buffer. After a period of inactivity, the driver bursts this many characters to the transparent printer before reducing to the maxcps rate selected above. Default is 100 characters.

onstr "*s*" — Defines the terminal escape sequence to direct subsequent data to the transparent printer.

s is a string of ASCII characters, enclosed in quotes, that command the terminal to enter transparent printing mode. An arbitrary octal character xxx may be given as \xxx.

For example, the sequence <Esc>[5i would be entered as: `"\033[5i"`.

offstr "*s*" — Defines the terminal escape sequence to stop directing data to the printer.

s is a string of ASCII characters, enclosed in quotes, that command the terminal to enter transparent printing mode. An arbitrary octal character xxx may be given as \xxx.

For example, the sequence <Esc>[5i would be entered as: `"\033[5i"`.

term *t* — Sets the transparent printer on/off strings to values found in the internal default table. Internal defaults are used for the following terminals: adm31, ansi, dg200, dg210, hz1500, mc5, microterm, multiterm, pcterm, tvi, vp-a2, vp-60, vt52, vt100, vt220, wyse30, wyse50, wyse60, or wyse75. If the terminal type is not found in the internal default table, then ditty reads the terminfo entry for the terminal type and sets transparent print on/off strings to values given by the mc5/mc4 attributes found there.

fastbaud Data Rate Mapping

Use the table below to see how setting fastbaud affects RealPort data rates.

Specified Data Rate:	Data Rate Mapped to:
50	57600
75	76800
110	115200
134	131657
150	153600
200	230400
300	460800