LINUX DRIVER
INSTALLATION AND USER GUIDE

DC0128-03

ZNYX Networks
48421 Milmont Drive
Fremont, CA 94538 USA

Telephone: (510) 249-0800 or (800) 724-0911
Fax: (510) 656-2460
Email: sales@znyx.com
support@znyx.com
Website: www.znyx.com

Linux Installation and User Guide
Document # dc0128-03
May 17, 2002

Table of Contents

Chapter 1 Getting Started

The ZNYX Networks Linux driver is compatible with many of the ZNYX adapter models. This manual contains instructions for installing the driver into your system.

The driver includes RAINlink technology (Redundant Array of Independent Netports) to provide critical link services such as link aggregation, link failover, and dynamic load balancing over multiple network ports. These ports can be a group of single-channel or multi-channel ZNYX Networks adapter ports, up to a maximum of 32 ports.

Platforms

- Motorola MCP750, MCPN750, or MCPN765 cPCI single board computers
- Motorola CPN5365 high-performance cPCI single board computer
- X86-based computer with standard PCI, PMC or cPCI slots.

Operating Systems

- Linux 2.2.12 and up, 2.4.x (tested up through 2.4.18) for x86
- Linux 2.2.17 for MCP750, MCPN750, MCPN765
- Linux 2.4.10 OA Node for Motorola CPN5365

ZNYX Adapter Models

Model	Bus	Number of Ports
ZX212	PMC	2
ZX214	PMC	4
ZX222	PMC	2
ZX244	PMC	4
ZX345Q	PCI	1
ZX346Q	PCI	4
ZX348Q	PCI	2
ZX372	PCI	2

Model	Bus	Number of Ports
ZX374	PCI	4
ZX412	cPCI	2
ZX414	cPCI	4
ZX422	cPCI	2
ZX424	cPCI	4
ZX442 HotSwap	cPCI	2
ZX444 HotSwap	cPCI	4
ZX474 Rear I/O & HotSwap	cPCI	4
ZX478 Rear I/O & HotSwap	cPCI	8

The Linux driver for ZNYX Network products is distributed as a compressed tar file.

`ZNYXnb.linux.tgz` includes the ZNYX Networks Linux driver source code necessary to build a loadable module for all supported versions of Linux. The package includes RAINlink.

`ZNYX559.linux.tgz` includes the ZNYX Networks OA Node source code designed to build a loadable module for Linux kernel 2.4.10 on the Motorola CPN5365. This package also includes RAINlink.

Chapter 2 Installing the Linux Drivers

Use the following procedure to add the ZNYX Networks Linux driver into your system. The driver is supported only as a loadable module.

Downloading the standard Linux Driver:

1. To install the driver `ZNYXnb.linux.tgz`, download from the ZNYX Networks web site Driver Support page, http://www.znyx.com.
2. Go to the directory in your Linux system that you want to download the driver files.
3. In your Linux window, enter the following command to uncompress and untar the file:

```
tar xvzf ZNYXnb.linux.tgz
```

Downloading the Motorola CPN5365 Linux Driver:

4. To install the driver `ZNYX559.linux.tgz`, download from the ZNYX Networks web site Driver Support page, http://www.znyx.com.
5. Go to the directory in your Linux system that you want to download the driver files.
6. In your Linux window, enter the following command to uncompress and untar the file:

```
tar xvzf ZNYX559.linux.tgz
```

Adding the standard Linux Driver:

The driver source files will be extracted and placed into a subdirectory called ZNYXnb. A `makefile` is included which compiles the source, links in the RAINlink object, and builds a module.

`cd ZNYXnb`	Change directory to the location of the makefile.
`make module`	Builds the driver object file `znb.o` as a loadable module object. The module can then be loaded directly using *insmod*.

Adding the Motorola CPN5365 Linux Driver:

The driver source files will be extracted and placed into a subdirectory called ZNYXnb. A `makefile` is included which compiles the source, links in the RAINlink object, and builds a module.

`cd ZNYX559`	Change directory to the location of the makefile.
`make module`	Builds the driver object file `znb.o` as a loadable module object. The module can then be loaded directly using *insmod*.

The target also builds RAINlink tools and creates a directory named, */etc/rain* that contains the resulting tools necessary to support RAINlink. These tools are only needed if you intend to use ZNYX Network's RAINlink features.

For all Linux systems:

1. To use the driver as an installable module, change directories into the driver directory. If your kernel configuration has module version dependencies turned off, edit the file `cdefs.mak` and comment out the current CFLAGS and INCLUDES entries and uncomment the CFLAGS and INCLUDES entries for builds with module version turned off.

   For standard Linux systems execute "make" with the module target,

   ```
   cd ZNYXnb
   make module
   ```

   For Motorola CPN5365 Linux systems execute "make" with the module target,

   ```
   cd ZNYX559
   make module
   ```

   Both of these options will build a module file `znb.o` that can be installed/loaded with the Linux *insmod* command,

```
insmod znb.o
```

Forcing Line Speed Settings

By default, all ZNYX drivers and adapters will autonegotiate. If you need to force the line speed settings for a particular port, options are passed to the driver using *insmod* at the time the module is loaded. The possible media settings are as follows:

Line Speed	Media Setting
Autonegotiate	auto
100 Full Duplex	txfd
100 Half Duplex	tx
10 Full Duplex	tpfd
10 Half Duplex	tp

For example, if you wanted to force the line speed of port0 (znb0) to 100 Full Duplex, use:

```
insmod znb zxe_media=txfd
```

The parameters are passed as a comma-delineated list in order of the ports, so if you wanted to force the line speed of port 2 (znb2) to 100 Full Duplex, you would load the module with the following parameters:

```
insmod znb zxe_media=auto,auto,txfd
```

Changing Driver Debug Levels

If you need to change the debug level of the driver for debugging purposes, a debug level from one to four (four provides the most information) can be passed to the driver using *insmod* at the time the module is loaded:

```
insmod znb zxe_debug_arg=3
```

The default debug level is one.

Configuring Linux to automatically load the ZNYX Driver:

Network interfaces can be configured to brought up at boot time with either the command line based *netconfig* tool or through the X-based *netconfig* tool. First, copy the module to the required location for *insmod* and the other module load utilities.

```
cp znb.o /lib/modules/linux_version/net/znb.o
```

Since the ZNYX driver has been built as a module, the correct "alias" commands must be added to /etc/modules.conf (or /etc/conf.modules on some versions of Linux) so that *modprobe* will know to load the znb driver at boot time. The alias commands are of the form:

```
alias znb0 znb
alias znb1 znb
alias znb2 znb
        ...
alias zrl0 znb
alias zrl1 znb
        ...
alias zrm znb
```

Single port devices are named "znb". RAINlink virtual devices are named "zrl". If you need to pass options to the module on load, the options are also included in the /etc/modules.conf. For instance, if you wished to force both ports of a two port ZNYX adapter to 100 Full-Duplex, you would add the following entries,

```
alias znb0 znb
alias znb1 znb
options znb zxe_media="txfd,txfd"
```

Consult your Linux documentation for more information on how to build and configure network drivers, as well as the Linux "how-to" for module support.

Chapter 3 Configuring RAINLINK for Linux

RAINlink for Linux includes utilities and example files for standard configurations to get you up and running quickly.

1. Change directories to */etc/rain* to begin the installation:

   ```
   cd/etc/rain
   ```

   An installation script is included called *rainlink*. The *rainlink* script can be used to activate, deactivate, or display the current RAINlink for Linux configuration. The first time the *rainlink* script is run, the script will give you a list of default configurations to choose from suitable for most basic configurations of two or four port trunks or failover groups.

2. Start the *rainlink* script by typing *rainlink -a* at the command prompt:

   ```
   rainlink -a
   ```

   The script dialogue enables you to choose from a set of default configurations. Select one to continue.

```
RAINlink

This script initializes or reinitializes the RAINlink layer
within the ZNYX
driver.  An input script located at /etc/rain/rainlink.conf
is used
as input to ZNYX's rlconfig application.  Either choose one
of the following
prepared input scripts which will be copied to
/etc/rain/rainlink.conf,
or exit and create your own custom script.

        1 - System-to-System trunking between 4 ports.
        2 - Fast EtherChannel trunking between 4 ports.
        3 - Fast Failover between 4 ports.
        4 - System-to-System trunking between 2 ports.
        5 - Fast EtherChannel trunking between 2 ports.
        6 - Fast Failover between 2 ports.
        7 - Exit and create custom script.

      Choose one of the above 7 choices(1-7):
```

A note is displayed informing you that the *rainlink* script will create a configuration file named, */etc/rain/rainlink.conf*, which will be the default configuration file used from this point forward. Details of the contents of the configuration file are provided later.

3. Select Enter to continue:

```
NOTE:   /etc/rain/rainlink.conf has been created.  If a
rainlink.conf file
        exists rainlink will not prompt for one of the
previous 7 choices
        To modify the RAINlink configuration in the future,
edit
        /etc/rain/rainlink.conf, and either run rainlink
again, or reboot the
        system.

Hit any key to continue:
Bringing down all Net Blaster interfaces now....
Running rlconfig ...
#####TRUNK0 Successfully Created
TRUNK0 mode set
```

The next series of questions relate to bringing up the interfaces and starting the RAINlink daemon, rld. Starting the RAINlink daemon allows you access to the statistics and configuration of RAINlink interfaces from a remote location. Bring up the RAINlink interfaces, then answer whether or not you will use the RAINlink daemon, and then bring up the remaining non-RAIN interfaces if there are any. You will need to supply IP names or addresses for each configured interface you bring up.

```
Do you want to configure and bring up the RAINlink interfaces [y,n,?] y
Enter hostname for interface zrl0 [barbrady-zrl0]: server1

The following host names must be in the hosts database.
Check with your Network Administrator for more information.

server1

Do you want to start the RAINlink daemon to allow for network
monitoring[ y,n,?] y

Do you want to configure and bring up any non-RAINlink interfaces [y,n,?] y

Bring znb4 up [y,n,?] y
Enter hostname for interface znb4 [barbrady-znb4]: access1

The following host names must be in the hosts database.
Check with your Network Administrator for more information.

access1

Bringing up available boards now....
```

The script then exits. Your RAINlink and non-RAINlink interfaces should now be configured.

4. You can use *rainlink –s* to verify the status of your RAINlink configuration:

```
rainlink –s
```

```
trunk0<LAYER_3_MODE> = znb0<UP>, znb1<UP>, znb2<UP>,
    znb3<UP>, znb4<UP>  ==>  zrl0
```

Advanced Trunking

You can create link aggregation groups (trunks), assign ports to these trunks and select a scheme of dynamic load balancing. The most straightforward method for configuring RAINlink for Linux is to edit the */etc/rain/rainlink.conf* file and run *rainlink -a.* The script de-installs the current ZNYX interfaces, runs a configuration application *rlconfig* with the *rainlink.conf* input script, and brings up the RAIN links. The *rainlink -a* script can also bring up any regular non-RAIN interfaces, which were not configured into the RAIN links. All these steps can be done individually. Details of using the *rlconfig* application are included in a later section.

To add a trunk, edit the file */etc/rain/rainlink.conf*. The file is a plain text file that is extensively commented. Commented lines begin with a pound sign (#). Add a line for each trunk containing the list of ports to be included in that trunk. End each line with a semicolon. For example, to include ports 0 through 3 in trunk0, add an entry:

```
trunk0=znb0, znb1, znb2, znb3;
```

The order of entries is not important. You can use any available ports in any order. Port entries should not be duplicated in trunks. A trunk must consist of at least two ports, and cannot contain other trunks. The following is an acceptable way to configure two trunks:

```
trunk0=znb0, znb3;
trunk1=znb1, znb2;
```

For system-to-system trunking, enable IP Trunking mode. Both systems should be setup identically. In IP Trunking mode, packets are sent to the driver as large datagrams and fragmented across the active links. To enable IP Trunking mode, uncomment/add the following line for each configured trunk:

```
trunk0=ip_trunking_mode;
```

For system-to-switch trunking, the default method of load balancing scheme uses Layer 3 Protocols. To use only the “Layer 2 Protocols” for dynamic load balancing, comment out the “layer_3_mode” option, and uncomment the “layer_2_mode” option. The options are mutually exclusive.

```
# trunk0=layer_3_mode;
trunk0=layer_2_mode;
```

To explicitly choose Layer 3, uncomment (or add) the “Layer 3 Protocols” option, and comment out the “Layer 2 Protocols” option:

```
trunk0=layer_3_mode;
# trunk0=layer_2_mode;
```

For system-to-switch trunking, you can disable Balance mode. Balance mode is on by default, and is normally left enabled. Balance mode dynamically distributes the load across the ports equally. It can be disabled by adding a “not equals” to the option:

```
trunk0 != balance_mode;
```

An “equals” sign explicitly enables balance mode:

```
trunk0 = balance_mode;
```

Once you have assembled your *rainlink.conf* file, run *rainlink -a* to bring up RAINlink for Linux with the new configuration.

```
rainlink -a
```

Advanced Failover

You can create link failover groups, assign ports and trunks to these groups and select a mode of failover. Failover groups are configured in the same manner as trunks: Edit the */etc/rain/rainlink.conf* file and run *rainlink -a. rainlink -a* de-installs the current ZNYX interfaces, runs a configuration application *rlconfig* with the *rainlink.conf* input script, and brings up the RAIN links. The *rainlink -a* script can also bring up any regular non-RAIN interfaces, which were not configured into the RAIN links. All these steps can be done individually. Details of using the *rlconfig* application are included in a later section.

To add a failover group, edit the file */etc/rain/rainlink.conf*. The file is a plain text file that is extensively commented. Commented lines begin with a pound sign (#). Failover groups can contain individual ports, trunks, or both ports and trunks. Always configure trunks before failover groups. Add a line for each failover group containing the list of ports and trunks to be included in that group. End each line with a semicolon. For example, to build a failover group of ports 0 and 1:

```
failover0 = znb0, znb1;
```

To build a failover group of two trunks, first build the trunks, then the failover group:

```
trunk0 = znb0, znb1;
trunk1 = znb2, znb3;
failover0 = trunk0, trunk1;
```

The following order will not work. You must build the trunks first:

```
failover0 = trunk0, trunk1;
trunk0 = znb0, znb1;
trunk1 = znb2, znb3;
```

You can also mix trunks and ports in failover groups, as long as the trunks are built first:

```
trunk0 = znb0, znb1;
failover0 = trunk0, znb2, znb3;
```

To remove a failover group, simply remove it, or comment it out, and run *rainlink -a*.

The default mode of failover is fast failover. In this mode, RAINlink for Linux moves the traffic over to a redundant stand-by link in case of a link failure in as little as milliseconds. In addition to fast failover, Timeout mode can be enabled for a failover group. By enabling Timeout Mode, if no traffic is received in the specified time interval, the active port is automatically switched to another available link. You can specify the timeout interval in milliseconds. To set Timeout Mode on a failover group with a timeout of 30 seconds, add a line for the failover group:

```
failover0 = timeout_mode 30000;
```

Two additional modes exist for failover groups hub_mode and snap_bcast_mode.

If you set hub_mode on a failover group, the group only receives on the ACTIVE port. Use when connecting failover ports in the same collision domain, like to a hub. Do not use when connecting system-to-system, or system-to-switch. To set hub_mode on a failover group,

```
failover0 = hub_mode;
```

If you set snap_bcast_mode, one SNAP broadcast packet is sent out the ACTIVE port upon failover. Use this option to improve updates to Layer 2 table entries in switches and routers.

```
failover0 = snap_bcast_mode;
```

Using Rlconfig

The *rlconfig* application views or changes the RAINlink for Linux configuration. *Rlconfig* is installed into */etc/rain* during the Linux driver installation. You can view the current RAINlink for Linux configuration at any time, but the interfaces must be "down" in order to change the configuration See *ifconfig(1M)* for more explanation of how to bring an interface down.

Options for *rlconfig* include:

Option:	**Use:**
-s	Displays the current RAIN configuration. Commands are not read from standard input with this option.
-t	Tear down all previously configured RAIN ports. Commands are not read from standard input with this option.
-p ppa	Displays statistics for the specified ppa device. Use 0 for ppa0, etc.
-l [#] file_name	Creates a file containing a list of the configured ZRL devices and the available ZNB devices. The "#" is used for version identification. This option is not intended for use by the user; it is used by the rainlink script to configure RAIN

To view the current configuration at any time, enter:

```
rlconfig  –s
```

The resulting display shows you the status of the RAIN configuration.

```
trunk0<IP_TRUNKING> = znb0<UP>, znb1<UP>, znb2<UP>,
        znb3<UP>, znb4<UP>  ==>  zrl0
```

Lines can be input directly to *rlconfig* from standard input, or from a script like */etc/rain/rainlink.conf*. To start *rlconfig*, enter:

```
rlconfig
```

Enter commands one line at a time and conclude with Ctrl-d.

To use a previously prepared file of commands, redirect the file into rlconfig. For example:

```
rlconfig < /etc/rain/rainlink.conf
```

Typical usage would be to redirect a script of commands into *rlconfig* which instruct *rlconfig* to build and set modes on trunks and failover groups. Semicolons delimit commands. Spaces and new lines are ignored. Commands take the following form:

```
trunk<number> = znb<number> [ ,  znb<number> … ] ;
failover<number> = znb | trunk<number> [ , znb|trunk<number> …]

trunk<number> = mode ;
trunk<number> != mode ;

failover<number> = mode ;
failover<number> != mode ;
```

where <number> is a value between 0 and the maximum number of ports in the system. Trunks can consist of ports denoted by a comma-delimited list of znb<number>. A failover group can consist of ports or trunks. You must build a trunk prior to its use in a failover group.

Acceptable modes for trunks are:

Mode:	**Appropriate for:**
ip_trunking_mode	System-to-system trunking
layer_2_mode	System-to-switch trunking
layer_3_mode	System-to-switch trunking
balance_mode	System-to-switch load balancing enabled/disabled
timeout_mode <time>	Failover mode; time parameter in milliseconds
hub_mode	Failover mode; Only receives on the ACTIVE port. Use when connecting failover ports in the same collision domain, like to a hub. Do not use when connecting system-to-system, or system-to-switch.
snap_bcast_mode	Failover mode; Upon failover, sends a SNAP Broadcast packet out the ACTIVE link.

To tear down the existing RAIN configuration, the interface must first be down. See ifconfig(1M):

```
ifconfig zrl0 down
rlconfig –t
```

Chapter 4 RAINLINK Reference Pages

RAINlink includes the following files and utilities to aid in configuration and monitoring:

- rainlink
- rlconfig
- rlstats
- rld

This section includes Unix-style man pages to be used as reference.

rainlink(1M)

NAME

rainlink—Activate/Deactivate RAINlink

SYNOPSIS

```
/etc/rain/rainlink [ -a | -d | -s | -u ]
```

DESCRIPTION

The *rainlink* command script initializes and reinitializes the RAINlink layer within the ZNYX driver. An input script located in */etc/rain/rainlink.conf* is used as input to ZNYX's rlconfig application. A list of default choices is presented if the */etc/rain/rainlink.conf* input script does not exist. The script can be used to bring up the RAINlink and non-RAINlink interfaces. It can also be used to display the current RAINlink configuration.

OPTIONS

-a	Activate RAINlink features. Creates an /etc/rain/rainlink.conf file if it doesn't exist from a list of default choices, brings down the existing ZNYX adapter interfaces, tears down the existing RAINlink configuration, builds the new RAINlink configuration, brings up the RAINlink interfaces, and finally any other ZNYX adapter interfaces.
-d	Deactivate RAINlink features. Tears down the existing RAINlink interfaces, and brings up ZNYX adapter interfaces.
-s	Show RAINlink configuration (same *as /etc/rain/rlconfig –s*)
-u	Display usage (same as no parameters)

FILES

```
/etc/rain/rainlink.conf
```

SEE ALSO

rlconfig(1M)

rlconfig(1M)

NAME

```
rlconfig—Configure ZNYX Redundant Array of Independent Netports
(RAIN).
```

SYNOPSIS

```
/etc/rain/rlconfig [ -s ] [ -t ] [ -p ppa ]
                [ -l # list_file ] < input_file
```

DESCRIPTION

The *rlconfig* application is used to configure multiple ZNYX adapter ports into trunks or failover groups. With exception of the -s option, all interfaces affected by *rlconfig* must be in the down states. See *ifconfig(1M)* for explanation on bringing down interfaces. The application reads standard input for commands.

A trunk is a grouping of two or more ports that can do port aggregation. Port aggregation can be put in a mode where multiple clients connecting to a server can utilize the bandwidth of multiple ports seamlessly, or in a mode where a server connecting to a server can seamlessly utilize the higher bandwidth of multiple ports.

A failover group is a grouping of ports where only one port is active at any one time. The remaining ports in the failover group are in standby in case the active port goes down. Typically, a failover group would only switch from one link to another if the physical link were lost. By enabling timeout mode, if no traffic is received in the specified time interval, the active port is automatically switched to another available link. The timeout mode time interval is settable by the user in increments of milliseconds.

OPTIONS

-s	Displays the current RAIN configuration. Commands are not read from standard input with this option.
-t	Tear down all previously configured RAIN ports. Commands are not read from standard input with this option.
-l [#] file_name	Creates a file containing a list of the configured ZRL devices and the available ZNB devices. The “#” is used for version identification. This option is not intended for use by the user; it is used by the *rainlink* script to configure RAIN
-p ppa	Displays hardware level statistics for the specified ppa.

USAGE

Typically, a script of commands is directed into *rlconfig*, which instruct *rlconfig* to build and set modes on trunks and failover groups. Semicolons delimit commands. Spaces and new lines are ignored. Commands take the following form:

```
trunk<number> = znb<number> [ ,  znb<number> … ] ;
failover<number> = znb<number>  | trunk<number> [ , znb<number> |
                   trunk<number> …];

trunk<number> = mode ;
trunk<number> != mode ;

failover<number> = mode ;
failover<number> != mode ;
```

where <number> is a value between 0 and the maximum number of ports in the system. Trunks can consist of ports denoted by a comma-delimited list of znb<number>.A failover group can consist of ports or trunks. You must build a trunk prior to its use in a failover group.

Acceptable modes for trunk and failover configurations:

Mode:	Appropriate for:
ip_trunking_mode	System-to-system trunking
layer_2_mode	System-to-switch trunking
layer_3_mode	System-to-switch trunking
balance_mode	System-to-switch load balancing enabled/disabled
timeout_mode <time>	Failover mode; time parameter in milliseconds

DISPLAYS

The "-s" option of the *rlconfig* displays the current status of the RAINlink interfaces. The display includes any configured modes for trunks or failover groups, and the status of the interfaces in the form:

```
trunk<number> <mode> = znb<number>, znb<number> …  ==>  zrl<number>
```

Where number is the trunk, group, or interface identifier, and mode is one of the acceptable modes. For example:

```
trunk0<IP_TRUNKING> = znb0<UP>, znb1<UP>  ==> zrl0
```

The interface status can be:

Name:	Meaning:
UP	Interface is UP
DOWN	Interface is DOWN
ACTIVE	The interface is UP, and is the ACTIVE member of a failover group
STANDBY	The interface is UP, but is not the ACTIVE member of a failover group.

The “-p” option of rlconfig displays the hardware level statistics for the specifies ppa. The display includes current ppa state, fault state, link configuration and current link state. Transmit and receive statistics are also displayed.

The following are valid values for the above states:

Name:	**Possible states:**
State:	HS_STATE_NOT_INIT HS_STATE_STOPPED HS_STATE_RUNNING
Fault:	HS_FAULT_NONE HS_FAULT_INTERNAL HS_FAULT_EXTERNAL
Link configuration	HS_LINK_AUTO HS_LINK_TP HS_LINK_BNC HS_LINK_AUI HS_LINK_TPFD HS_LINK_TX HS_LINK_TXFD HS_LINK_T4
Current link state:	HS_LINK_DOWN HS_LINK_TP HS_LINK_BNC HS_LINK_AUI HS_LINK_TPFD HS_LINK_TX HS_LINK_TXFD HS_LINK_T4

FILES

/etc/rain/rainlink.conf

SEE ALSO

rainlink(1M)

rlstats(1M)

NAME

rlstats—Displays statistics for ZNYX adapter or RAINlink devices

SYNOPSIS

```
/etc/rain/rlstats [[-m|-s|-o|-i] <ppa>] [-d <level>] [-z]
```

DESCRIPTION

The *rlstats* application is used to display statistics and configuration information for RAINlink devices.

OPTIONS

-m ppa	Displays MAC layer statistics for specified *ppa*.
-s ppa	Turns speedometer on for specified *ppa*.
-r ppa	Sets speedometer polling rate (in ms) for specified *ppa*. Use in conjunction with –s *ppa*.
-i ppa	Retrieve hardware information for specified *ppa*
-o ppa	Display speedometer statistics for the specified *ppa*
-d debug_level	Sets debug level (1 through 4). Larger number increases verbosity.
-z	Displays RAINlink configuration

USAGE

Use *rlstats* to display statistics for a particular port or RAINlink configuration.

FILES

None

SEE ALSO

rld(1M)

rld(1M)

NAME

rld—RAINlink redirector daemon

SYNOPSIS

```
/etc/rain/rld [-d debug_level] [-p port] [-f] [-l log_level]
```

DESCRIPTION

The *rld* server is used for remote management of RAINlink facilities.

OPTIONS

-d debug_level	Sets debug level (1 through 4). Larger number increases verbosity
-p port	Sets TCP port to listen on. Default is 7000
-f	Run daemon in the foreground
-l log_level	Log debug messages at level (1 through 4) to log file /etc/rain/rld.log

FILES

None.

SEE ALSO

rlarm(1M), rlstats(1M)

Chapter 5 Release Notes

New Features and Bug Fixes

- Added support for newer versions of Linux up through 2.4.18 (x86)
- Added OA Node driver with RAINlink technology for Motorola MCP5365 Node card.
- Added snap_bcast_mode for failover groups. snap_bcast_mode sends one SNAP broadcast packet out the ACTIVE link upon failover. Use this option to improve updates to Layer 2 table entries in switches and routers. snap_bcast_mode is set on in the default RAINlink scripts.
- timeout_mode is no longer set in the default RAINlink scripts.
- Added support for newer revisions of the ZNYX adapters that will use the Intel 21152-BB, Intel 21154-BE, and Intel 21555-AA bridge chips.
- Version 2.3.16 Bugs fixed: 380, 465

Discrepancies

- Currently there are no “Open” bugs in 2.3.17