



# FreeBSD 4.x Driver

*with Embedded RAINlink Technology (v2.3.10)*

## Installation and User's Guide

### Overview

The FreeBSD driver for the ZNYX NetBlaster™ PCI, CompactPCI and PMC 10/100 adapters includes embedded RAINlink technology. RAINlink provides critical link services such as link aggregation (trunking), link failover and dynamic load balancing over multiple network ports. These ports can be a group of multiple single-channel or multi-channel ZNYX adapter ports, up to a maximum of 16 ports.

The ZNYX drivers provide support for FreeBSD 4.x for x86 systems. The driver can be used either built-in to the kernel, or as a kernel loadable module. The appropriate driver can be obtained from the ZNYX web site: <http://www.znyx.com>. The distribution is available as a compressed, tar file (tar.gz).

The following steps will take you through the additions necessary to build a new kernel that includes the ZNYX driver. Refer to your FreeBSD documentation for detailed information about installing drivers and rebuilding kernels.

### Systems Requirements

- Platforms:** x86-based platforms with standard PCI, PMC, or CompactPCI slots.
- Operating Systems:** FreeBSD 4.0, 4.1, 4.1.1, 4.2 for x86 platforms
- NOTE: For 4.0, the ZNYX driver only exists as a loadable module

### Supported ZNYX NetBlaster Adapters:

Model	Bus	Number of Ports
212	PMC	2
214	PMC	4
222	PMC	2
244	PMC	4
345, 345Q	PCI	1
346, 346Q	PCI	4
348, 348Q	PCI	2
372	PCI	2
374	PCI	4

412	cPCI	2
414	cPCI	4
422	cPCI	2
424	cPCI	4
442 hot-swap	cPCI	2
444 hot-swap	cPCI	4
474 rear i/o & hot-swap	cPCI	4
478 rear i/o & hot-swap	cPCI	8

### **Retrieving the ZNYX Drivers**

1. Go to the ZNYX web site at <http://www.znyx.com> and navigate under the driver downloads area to get the FreeBSD driver for your adapter.
2. The driver is available as a compressed tar file. Retrieve the file to the /usr/src directory of your FreeBSD system. Uncompress and untar the file. You must have super-user privileges to complete this installation.

```
cd /usr/src
tar xvzf ZNYXnb.i386.tar.gz
```

### **Installing the ZNYX Driver for FreeBSD**

Building a kernel with the ZNYX driver is relatively simple. It involves removing the standard DEC driver (if\_de.o and/or if\_dc.o) and replacing it with the ZNYX driver. The driver can be used as either built-in to the kernel or as a kernel loadable module.

To install the ZNYX driver do the following:

1. In /usr/src/sys/i386/conf, make a copy of the GENERIC kernel configuration file (lets call it ZNYX). Edit the ZNYX kernel configuration file as follows:

Delete the lines:

```
device de          # DEC/Intel DC21x4x (``Tulip'')
device dc          # 21143 and clones
device ida        # DEC/Compaq RAID controller (claims 21554)
```

The ZNYX driver requires Berkeley Packet Filter support in the kernel. Versions of FreeBSD prior to 3.4 did not include support by default. Make sure that the following line is uncommented in your config file:

```
pseudo-device bpf      # Berkeley Packet Filter
```

2. Run config on the ZNYX kernel configuration file:

```
/usr/sbin/config ZNYX
```

3. Change directories to /usr/src/sys/compile/ZNYX.

- a) To build the driver into the kernel (Not supported for 4.0. For 4.0, you must use a loadable module):

```

Edit the Makefile and add if_zxe.o to the end of the list of OBJS,
OBJS=device_if.o bus_if.o cam.o cam_xpt.o cam_extend.o cam_queue.o \
    cam_periph.o cam_sim.o scsi_all.o scsi_da.o scsi_sa.o scsi_cd.o \
    .
    .
    syscons_isa.o vga_isa.o subr_diskmbr.o divdi3.o moddi3.o \
    qdivrem.o udivdi3.o umoddi3.o if_zxe.o

```

Copy if\_zxe.o to /usr/src/sys/compile/ZNYX/if\_zxe.o

- b) To use the driver as a kernel loadable module:

Copy if\_zxe.ko into /modules.

4. Execute the following to build and install the new kernel:

```

make depend
make
make install

```

Reboot to run the new kernel.

5. If you are using the driver as a kernel loadable module, edit /etc/rc.conf, and add,

```
kldload if_zxe.ko
```

after the first echo statement in network\_pass1().

6. Edit /etc/rc.conf (or your equivalent) and configure the ZNYX devices (znb0, znb1,...)

```
ifconfig znb0 10.1.1.2.3 (the IP address is just an example)
```

7. By default the ZNYX Ethernet adapters will auto negotiate with hubs, switches, routers, or other NICs. If you need to set the media speed or full duplex options you can do so with the following commands:

<b>Command:</b>	<b>Line Speed:</b>
ifconfig znb0 media 10baseT/UTP	10 Mbps Half-duplex
ifconfig znb0 media 10baseT/UTP mediaopt full-duplex	10 Mbps Full-duplex
ifconfig znb0 media 100baseTX	100 Mbps Half-duplex
ifconfig znb0 media 100baseTX mediaopt full-duplex	100 Mbps Full-duplex
ifconfig znb0 media auto	Autonegotiation

	(default)
--	-----------

If you need information to do troubleshooting you can set debug levels as follows:

Command:	Debug Level:
<code>ifconfig znb0 debug</code>	Sets to 2
<code>ifconfig znb0 debug</code>	Sets to 3 after executing the above
<code>ifconfig znb0 debug</code>	Sets to 4 after executing the above
<code>ifconfig znb0 -debug</code>	Resets to 1 (the default)

8. Reboot the system to run the new kernel.

## Configuring RAINlink for FreeBSD

Typically you need to configure either trunks or failover groups for a given network. With RAINlink for FreeBSD, you can also include trunks in failover groups. It is strongly recommended that you use trunking whenever your network adapters connect directly system-to-system, or to Cisco FEC compatible switches, routers or other adapters. This is because trunking provides both increased bandwidth and improved fault-tolerance, whereas the failover service only provides improved fault-tolerance.

Configuring a trunk means to create (add) a trunk between two end stations and assign the member ports (links) to a trunk from EACH end station. A trunk should have at least two member ports (links.) Each port in a trunk can be a half or full duplex.

Configuring a failover group means to create (add) a failover group between two end stations and assign the member ports (links) to a group from each station. Either ports or trunks, or both can be members of a failover group. A failover group should have at least two members. Each port in a failover group can be a half or full duplex.

### Quick Start

RAINlink allows you to configure the ZNYX adapters for failover, Cisco Fast EtherChannel compatible trunking, or system-to-system trunking.

1. From the target directory where you untarred the ZNYX driver package enter:

```
make
```

This will run a makefile that will create the `/etc/rain` directory and copy the necessary files into `/etc/rain`.

2. Go to the `/etc/rain` directory:

```
cd /etc/rain
```

3. An installation script is included called `rainlink`. The `rainlink` script can be used to activate, deactivate or display the current RAINlink for FreeBSD 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. Start the *rainlink* script by typing *rainlink -a* at the command prompt:

```
rainlink -a
```

4. The script dialogue will allow you to choose from a set of default configurations.

```
RAINlink

This script initializes or reinitializes the RAINlink layer within the ZNYX
NetBlaster driver. An input script located at /etc/rain/rainlink.conf is used
as input to ZNYX's rconfig 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):
```

5. Make your selection and continue. A note will be 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. Hit 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 rconfig ...
#####TRUNK0 Successfully Created
TRUNK0 mode set
```

6. The next series of questions relate to bringing up the interfaces. Next, bring up the RAINlink interfaces, and then 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 zr10 [barbrady-zr10]: server1

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

server1

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.

7. Edit `/etc/rc.conf` (or `/etc/rc.network`, or your local equivalent) and add the following line:

```
/etc/rain/rainrc
```

The following link may have already been added when `rainrc` was run, if not add it to `/etc/rc.conf`:

```
ifconfig zr10 10.1.1.2.3 (the IP address is just an example)
```

8. 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> ==> zr10
```

For most configurations, you will only need to pick the correct default configuration profile and run `rainlink -a`, as described previously.

If you wish to customize configuration for trunks or choose a different scheme for the dynamic load balancing, go to the "Advanced Trunking" section. If the link aggregation service is not suitable to your network configuration, then go to the "Advanced Failover" section to configure the failover groups instead of trunks. You can also configure failover groups of trunks themselves. Always configure trunks before failover groups.

## Advanced Trunking

You can create link aggregation groups (trunks), assign port to these trunks and select a scheme of dynamic load balancing. The most straightforward method for configuring RAINlink for FreeBSD 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 FreeBSD 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 FreeBSD moves the traffic over to a redundant stand-by link in case of a link failure in as little as 500 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 choose different failover modes for different groups. You can also specify the timeout interval. 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;
```

## Using *rlconfig*

The *rlconfig* application views or changes the RAINlink for FreeBSD configuration. *rlconfig* is installed into */etc/rain* by running `make` in the directory where you untarred the ZNYX driver. You can view the current RAINlink for FreeBSD 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.
-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 specified ppa device.

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 ==> 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:

<b>Mode:</b>	<b>Appropriate for:</b>
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.

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

```
ifconfig zrl0 down
rlconfig -t
```

## RAINlink for FreeBSD Man Page Reference

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

- rainlink
- rlconfig
- rlalarm
- rlstats

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

rainlink(1M)

### NAME

rainlink – Activate/Deactivate RAINlink for FreeBSD

### SYNOPSIS

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

### DESCRIPTION

The *rainlink* command script initializes and reinitializes the RAINlink layer within the ZNYX NetBlaster 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 <i>/etc/rain/rainlink.conf</i> file if it doesn't exist from a list of default choices, brings down the existing NetBlaster interfaces, tears down the existing RAINlink configuration, builds the new RAINlink configuration, brings up the RAINlink interfaces, and finally any other NetBlaster interfaces. |
| -d | Deactivate RAINlink features. Tears down the existing RAINlink interfaces, and brings up NetBlaster interfaces.   |
| -s | Show RAINlink configuration (same as <i>/etc/rain/rlconfig -s</i> )   |
| -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 NetBlaster 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 <i>rainlink</i> 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
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.

#### 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 rconfig 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:

<b>Name:</b>	<b>Possible states:</b>
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)*

rlalarm(1M)

#### **NAME**

rlalarm – RAINlink switch and event monitoring application

#### **SYNOPSIS**

```
/etc/rain/rlalarm [-d <debug_level> [-s <dev_id> <ppa>] [-r <dev_id>] [-m <ppa>]
[-p <dev_id> <ppa> <ppa> ...] [-h <host_name>
```

#### **DESCRIPTION**

*rlalarm* is a utility based on the RAINlink Management API (RMAPI) that allows manipulation of RAINlink devices as well as a method for event notifications of a specified device. The device can be either a RAINlink or MAC level device, and can be located locally or reachable remotely via a RAINlink server daemon (rld).

#### **OPTIONS**

- s *dev\_id* *ppa*            Switch ACTIVE *ppa* of device *dev\_id* to the specified *ppa*.
- r *dev\_id*                    Display RAINlink notifications from specified *dev\_id*.
- m *ppa*                        Display MAC level notifications from specified *ppa*.
- p *dev\_id* *ppa* *ppa* ...    Set precedence of device *dev\_id* on failover group to order of *ppa* list. First entry in list has highest precedence.
- d *debug\_level*               Sets debug level (1 through 4). Larger number increases verbosity.
- h *host\_name*                Connect to remote host.

## USAGE

Rlalarm allows manipulation and monitoring of RAINlink devices. To switch the active port in a failover group (zrl0) from port0 to port1, enter:

```
rlalarm -s 0 1
```

The 0 references the RAINlink (zrl) device, the 1 references the port (or *ppa*, *znb1*) to make ACTIVE in the failover group.

The "-r" and "-m" options allow you to receive notifications of event changes at either the RAINlink level or MAC level.

The "-p" option allows a method for setting a precedence within a failover group. This allows the user to configure a failover group to always return to a specified *ppa*, if it becomes available later. The default operation for a RAINlink failover group is to only switch when the ACTIVE port no longer is available. For example, once you've configured a failover group of two ports,

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

Use "rlalarm" to specify that whenever *znb0* is available, it should be the ACTIVE port:

```
rlalarm -p 0 0 1
```

You can also connect to a remote host running the RAINlink server daemon *rls*. Provide the remote hostname or ip address as an argument to rlalarm:

```
rlalarm -h 10.0.0.1 -m 0
```

This would attempt a TCP connection to remote host with IP address 10.0.0.1, and if successful retrieve the MAC level statistics for *znb0* (*ppa* 0).

## FILES

None

## rlstats(1M)

### NAME

rlstats – Displays statistics for NetBlaster 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 <i>ppa</i>	Displays MAC layer statistics for specified <i>ppa</i> .
-s <i>ppa</i>	Turns speedometer on for specified <i>ppa</i> .
-r <i>ppa</i>	Sets speedometer polling rate (in ms) for specified <i>ppa</i> . Use in conjunction with <i>-s ppa</i> .
-i <i>ppa</i>	Retrieve hardware information for specified <i>ppa</i>
-o <i>ppa</i>	Display speedometer statistics for the specified <i>ppa</i>
-d <i>debug_level</i>	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

## Release Notes

### Driver Version v2.3.10

#### New Features

- Added support for newer ZNYX adapters (ZX222, ZX244, ZX370 Series, ZX420 Series)
- Added support for RAINlink Management Application Programmer Interface (RMAPI)
- Added support for freeBSD 4.x, particularly kernel loadable module support

#### Discrepancies

- `if_zxe.ko` cannot be unloaded.
- The ZNYX driver cannot be built-in to the FreeBSD 4.0 kernel. You must use the loadable module.
- The call to `bpf_mtap` in receive routines has been removed; packets are sent to BPF twice.
- The readme included in the driver distribution does not include a reference to including the `mediaopt` parameter on the `ifconfig` command line to set full-duplex.
- Reconfiguring between RAINlink and non-RAINlink ports can result in errors. It is best to start from a clean reboot, once you have all interfaces configured properly.

## Where to go for Technical Support

The best way to obtain technical support is to email [support@znyx.com](mailto:support@znyx.com).

Please include:

1. Exact model number of ZNYX hardware.
2. Exact version of the FreeBSD operating system.
3. A very detailed description of your problem.
4. Attaching console output in the vicinity of the problem is a big plus.

Resource	Address
Telephone	(510) 249-0800
Toll-Free	(800) 724-0911 (USA Only)
FAX	(510) 656-2460
Website	<a href="http://www.znyx.com">www.znyx.com</a>
E-mail	<a href="mailto:support@znyx.com">support@znyx.com</a>



48421 Millmont Drive  
Fremont, CA 94538

**Document # DC0136-01**  
**Version # 23-January-2001**

© 2000-2001 ZNYX Networks. All rights reserved worldwide. All information in this document is subject to change without prior notice. ZNYX and NetBlaster are trademarks or registered trademarks of ZNYX Networks in the United States and/or other countries. All other marks, trademarks or service marks are the property of their respective owners.

ZNYX Networks may have patents, pending patent applications, trademarks, copyrights or other intellectual property rights covered in the subject matter of this document. By furnishing this document, ZNYX Networks does not license nor waive its license to those intellectual property rights except as expressly provided in a written license agreement from ZNYX Networks. Information in this document is subject to change without prior notice.