# EtherDrive® SATA/RAID (SR) Appliance Storage
# Software User Manual
# Update 1/30/07

## 1. Introduction

EtherDrive SR Appliance Storage is a block storage RAID appliance with front loading, hot-swappable SATA disk drives. Disks can be used individually or in RAID sets and are exported as logical storage devices accessible using the ATA-over-Ethernet (AoE) protocol. The appliance is built upon an embedded operating system similar to UNIX.

This manual explains the concepts behind and the commands used to manage the appliance. The output of commands listed in this document may differ from that output by your software. **Software functionality documented in this manual may not be available in firmware revisions dated older than this manual.** This document does not cover the hardware used in constructing the appliance (please refer to the companion Installation Guide).

For quick start examples, please see **Appendix B**.

A low-traffic mailing list is available to notify customers of documentation and firmware updates. Please email `support@coraid.com` and ask to be placed on the SR announcement mailing list. SR firmware and documentation can be found at the SR support page: `http://coraid.com/support/sr/`.

### 1.1. Concepts

#### 1.1.1. Network Connections

The Coraid SR Appliance is equipped with two Gigabit Ethernet ports and can be connected directly to a server using Ethernet or to multiple servers using standard Ethernet switching equipment. The SR Appliance will present all logical AoE devices to each segment. It is recommended that administrators use caution when connecting both ports to the same LAN segment as the exported logical devices will be addressible from two network locations causing potential performance problems. Please see the paper titled *SR Redundancy and Throughput in Linux* at the SR support page for details. For CEC configuration, /net/ether0 is the left interface and /net/ether1 is the right interface, when viewed from the rear.

#### 1.1.2. Coraid EtherDrive Storage Blades

Coraid's flagship product, the EtherDrive storage blade, is a small blade nanoserver with its own CPU, RAM, and interfaces for ATA and Ethernet. Its sole job is to perform the ATA over Ethernet (AoE) protocol to the attached disk, effectively giving the ATA disk an Ethernet port.

The AoE protocol rides directly on top of Ethernet and does not use IP. In order to simplify management of large installations of EtherDrive blades, the AoE protocol permits addressing devices based on an EtherDrive shelf and slot address. Using this method, administrators can manage an EtherDrive blade based on its physical location instead of its MAC address.

Inside a client system, EtherDrive blade devices are named by a system specific shelf and slot naming scheme. On the SR Appliance, local disks are addressed as <*shelf*>.<*slot*>, where the shelf address is assigned by the administrator. For example, if the shelf address were set to 0 the first ten disks would be addressed as *0.0, 0.1, ... 0.9.*

### 1.1.3. Logical Blades

Each SR Appliance must be given a shelf address with the `shelf` command. The appliance exports logical blade (lblade) slots within this shelf address to the network. Lblades are created containing one RAID and may be grown with additional RAIDs. Each RAID in an lblade may be of any size and type and is designated by a part number within the lblade. Lblades are grown by appending new component RAID parts. The creation of the lblade and its initial RAID is accomplished with the `make` command. Additional RAIDs may be added with the `grow` command. From the client system on the network the SR Appliance looks like a shelf of blades.

One or more disks in the appliance can be exported individually as a typical JBOD configuration. To conform with the software architecture, a single disk element lblade is created by initialising a linear RAID over a single disk. The `jbod` command is provided to simplify this task. `Jbod` assumes the external lblade slot is the same as the internal disk slot being exported. If this is not desired, the administrator may create the single disk linear RAID lblade using the `make` command.

Specifying lblade RAID component devices is accomplished with a three tiered naming scheme: *lblade.part.drive.* Lblade is the lblade number, part is the RAID component in the lblade, and drive is the drive component in the RAID. As an example, `0.1.2` would be the third drive (`2`) in the second RAID part (`1`) of lblade `0`.

An administrator may choose to force an lblade to stop responding to requests. This is accomplished by changing the state of the lblade to offline using the `offline` command. To bring an lblade back online, an administrator can use the `online` command.

### 1.1.4. RAID Spare Allocation

Spare allocation for redundant RAID levels is accomplished by two means. The administrator can choose to manage RAIDs manually by replacing failed drives using the `unfail` or `replace` commands. Alternately, the administrator can declare a pool of spares to be automatically allocated when a failure occurs using the `spare` command.

Spare allocation from the spare pool is accomplished with a best-fit algorithm. When a failure occurs, the spare pool is checked for the smallest drive that will satisfy the RAID's need. As an example, if the spare pool contained 500GB and 400GB drives and a failure occurred on a RAID needing a 300GB drive, a 400GB drive would be elected as the replacement.

When a RAID sustains a failure, a background process periodically checks for a spare to become available. If any redundant RAID is operating with a failure, adding a spare to an empty spare group will cause that spare to be allocated to the RAID with the failure. If the appliance sustains a disk failure to a RAID part that is currently having its redundancy reconstructed -- either raid5 parity rebuild or mirror rebuild -- a double failure will occur causing the appliance to mark the RAID and its associated lblade as unusable. The lblade must be removed and remade to be reinstated.

There is a single spare pool for all RAIDs managed within the appliance.

### 1.1.5. Limiting Access

Administrators may limit client access to lblades by giving an lblade a MAC mask list. By default, an lblade's mask list is empty permitting access to anyone on the network. MAC addresses can be added to an lblade's mask list using the `mask` command. If an lblade's mask list is not empty, only those MAC addresses in the mask list will be permitted access. The MAC mask list **must be set per lblade**.

### 1.1.6. Coraid Ethernet Console (CEC)

CEC (pronounced "kick") is a method for obtaining a console connection to the appliance using standard Ethernet frames. CEC is very lightweight and does not include any security or encryption mechanisms. The commands `cecon` and `cecoff` are used to manage which interfaces accept CEC connections.

For ease of first configuration, CEC is enabled on both interfaces on new shipments. CEC first appeared in SR release 20060717; users updating to obtain CEC functionality must manually enable CEC. **Be aware that enabling CEC and failing to logout may leave your system insecure depending on your network configuration.** As with AoE, the appliance is only as secure as your network.

The CEC configuration will persist across reboots. To obtain a CEC client, please visit the SR support page at coraid.com.

### 1.1.7. Shutting Down the System

In order to cleanly shut down the system administrators should use the `shutdown` command which flushes out dirty buffers, marks all RAIDs as cleanly shut down, and reboots the system. As a matter of policy, when the appliance starts it will validate all raid5 RAIDs that are not marked as clean, ensuring the parity is correct.

### 1.1.8. RAID Configuration

Each RAID's configuration is saved on the component drives used in the RAID. When shutting down, the system must be able to communicate with all of a RAID's components in order to cleanly release the RAID.

### 1.1.9. Specifying Devices

Many commands require the specification of one or more lblades or disk drives. To ease the specification of these devices a special range character (`-`) may be used. As an example, `0.2-5` is a valid substitution for `0.2 0.3 0.4 0.5`, indicating disks 2 through 5 on shelf 0. Analagously for lblades, `0-3` is a valid substitution for `0 1 2 3`, indicating lblades 0 through 3.

### 1.1.10. Console Port Settings

In addition to using keyboard and video display the Coraid SR Appliance can be configured using the serial port. Before connecting a terminal to the serial port, configure the terminal to match the serial port: **9600 8-N-1** ( 9600 baud, 8 data bits, no parity, 1 stop bit ).

### 1.1.11. Keyboard shortcuts

In order to make the system more friendly to users accustomed to the GNU readline interface, the following control sequences are provided. A history buffer of 32 lines is maintained.

| | |
|---|---|
| backspace | erase the character before the cursor |
| ^d | delete character under cursor |
| ^w | delete previous word |
| ^u | delete entire line |
| ^b | cursor back one char |
| ^f | cursor forward one char |
| ^a | cursor beginning of line |
| ^e | cursor end of line |
| ^k | kill text from cursor to end of line |
| ^y | yank killed text into line after cursor |
| ^p | go back one line in the history, cursor at end of line |
| ^n | go forward one line in the history, cursor at end of line |

In addition, the control sequences emitted by most Linux systems for the left, right, up, and down arrows are recognized. These are equivalent to ^b, ^f, ^p and ^n, respectively.

### 1.1.12. RAIDShield™ Protection

Each redundant RAID is protected by a mechanism called the RAIDShield. When a disk in a well-functioning -- not degraded or failed -- redundant RAID exhibits a block read error, the RAIDShield is enacted to calculate the block contents from the remaining disks in the array. RAIDShield then writes the block to the "failed" disk and then rereads it to permit the disk to internally remap the bad sector(s). Whether or not this is successful, the user will be notified by a syslog diagnostic of the attempt.

A special case occurs if the RAIDShield is enacted to recover from a failure on a disk in a raid5 when the parity is not yet initialized. This can occur when the raid5 is first created, but is more common when the appliance goes down without cleanly releasing the raid5. In the latter case the parity for the row containing the failure is likely to be correct and can be relied upon to reconstruct the block. Since this cannot be guaranteed, however, RAIDShield will take the lblade offline to protect the user from potential data corruption. The user must acknowledge the potential error by running the online command twice to bring the lblade back online. As expected, syslog messages are generated at every step.

Also covered under the RAIDShield is a background scrubber process that continually reads redundant RAID arrays to detect and correct read failures using the same mechanism described above. The RAIDShield scrubber operates in a loop reading a predetermined amount of data and sleeping for one second. The amount of data read is determined based on the access of the RAID array and is calculated to avoid conflicting with user access I/O. The scrubber’s access pattern will cause the disk access LEDs on scrubbed disks to flash every second.

The scrubber will not run if the array is initing, degraded, or failed. As a result of the scrubber process, disks in redundant arrays are rarely idle. As disk access does not contribute to shortened disk life this should not be a concern.

## 1.2. Commands

### 1.2.1. passwd

```
usage: passwd
```

The `passwd` command is used to set a console login password. If the administrator should lose the password given to the appliance, entering the reserved password `ivelostit` will begin a challenge/response dialog which the administrator must contact Coraid support to complete. After successfully passing the challenge/response dialog, the password will be cleared.

### 1.2.2. shelf

```
usage: shelf [ shelfno ]
```

The `shelf` command is used to set and show the shelf address of the appliance. Without an argument, the shelf address is printed. The argument to shelf should be a value between 0 and 65534, inclusive. It is recommended that the administrator ensure each shelf address is unique among all AoE storage devices attached to the network.

The `shelf` command will reconfigure all live lblades to be owned by the new shelf address. As a result of this process, lblades are taken down and restarted. Issuing the `shelf` command during disk reconstruction or parity initialisation will cause the reconstruction or initialisation to start over.

*Example:*

```
SR shelf 1> shelf 2
Reading config information from drives ... done.
SR shelf 2> shelf
2
SR shelf 2> shelf 1
Reading config information from drives ... done.
SR shelf 1> shelf
1
SR shelf 1>
```

### 1.2.3. show

```
usage: show [ -lc ]
```

The `show` command can be used to display the disks in the appliance. The `-l` (the letter "l") flag shows extended information about each disk, including its size and state. The `-c` flag shows raid configuration information sometimes needed by coraid support.

*Example:*

```
SR shelf 1> show
   1.0   1.1   1.2   1.3
SR shelf 1> show -l
   1.0     82.35GB up
   1.1     82.35GB up
   1.2     82.35GB up
   1.3     82.35GB up
SR shelf 1>
```

### 1.2.4. list

```
usage: list [ -l ] [ lblade ... ]
```

The `list` command shows the logical blades currently exported by the appliance. If an lblade is not specified, all current logical blades are listed. Each line output lists the logical blade slot number, size, and state. An lblade's state is either online or offline, indicating whether or not it is accepting requests.

The `-l` (the letter "l") flag gives detailed information about the logical blade components. The first line in each record displays the slot number, size, and state of the lblade. The next indention level displays component RAID parts and their size, type, state, and if appropriate, reconstruction block number. The state of the component RAID is one or more of:

initing - RAID is initializing parity.
recovering - RAID is rebuilding replaced component.
degraded - RAID is operating with failed or missing components.
failed - RAID has sustained too many component failures and is unusable.
normal - RAID is operating normally.

The reconstruction block number is an offset in units of 64kB.

Following the RAID component at the next indention level are the drive components in the RAID, one per line. Each line displays the drive's RAID component address, state, size, and shelf.slot location. The state of the drive component is one of:

failed - drive has failed.
replaced - drive is being used as a replacement for a failed drive.
missing - a placeholder for a missing drive; possible if all components are not available at startup time.
normal - drive is operating normally.

*Example:*

```
SR shelf 1> list
 0   82.348GB online
 1   82.348GB online
 2   82.348GB online
 3   82.348GB online
SR shelf 1> list -l
 0   82.348GB online
  0.0     82.348GB raidL
    0.0.0  normal       82.348GB 1.0
 1   82.348GB online
  1.0     82.348GB raidL
    1.0.0  normal       82.348GB 1.1
 2   82.348GB online
  2.0     82.348GB raidL
    2.0.0  normal       82.348GB 1.2
 3   82.348GB online
  3.0     82.348GB raidL
    3.0.0  normal       82.348GB 1.3
SR shelf 1>
```

### 1.2.5. make

`usage: make lblade raidtype [ shelf.slot ... ]`

The `make` command is used to create RAIDs and initialize them to a logical blade slot.

The `lblade` field must be a value between 0 and 63, inclusive. If set sufficiently high the lblade slot may not be addressible from the client system. Coraid recommends administrators verify slot addressibility with host system drivers prior to allocation.

The `raidtype` field may be one of:

raidL - A linear raid device.
raid0 - A striped raid device.
raid1 - A mirrored raid device.
raid5 - A round-robin parity raid device.
raid10 - A stripe of mirrors raid device.
update - A ram based device for appliance update.

A full description of the various RAID levels is beyond the scope of this document. The update raidtype is a pseudotype for a raidL over a RAM based disk. No component drives should be specified when declaring the update lblade. Please see **Appendix C** for an explanation of the software update procedure.

The `shelf.slot` fields following `raidtype` specify the drives to be used as components of the RAID.

The make command initializes lblades in the offline state, permitting administrators to set a mac mask list prior to making the lblade available for access. Generally, newly created lblades must be brought online with the `online` command before they will be visible to client systems. There are two exceptions to this rule; the update lblade and lblades created with the jbod command are automatically brought online as a convenience.

For raid10, the drive list must contain an even number of elements. The list is split into two equal ordered sets at the center of the list. Mirrors are constructed across the pairs of drives in the same position in each set. Once the mirrors are chosen, a stripe is placed across all mirrored elements.

In the example below, the mirrored elements on lblade 1 are {1.4, 1.7}, {1.5, 1.8}, and {1.6, 1.9}.

*Example:*

```
SR shelf 1> make 1 raid10 1.4-9
SR shelf 1> list
 1  247.045GB offline
SR shelf 1> online 1
SR shelf 1> list -l
 1  247.045GB online
  1.0    247.045GB raid10
    1.0.0  normal       82.348GB 1.4
    1.0.1  normal       82.348GB 1.5
    1.0.2  normal       82.348GB 1.6
    1.0.3  normal       82.348GB 1.7
    1.0.4  normal       82.348GB 1.8
    1.0.5  normal       82.348GB 1.9
SR shelf 1>
```

#### 1.2.6. remove

`usage: remove lblade ...`

The `remove` command is used to remove one or more lblades. All drives used in component RAIDs will be released for reuse.

*Example:*

```
SR shelf 1> list -l
 0  247.045GB online
  0.0    247.045GB raid5 initing 93188
    0.0.0  normal       82.348GB 1.0
    0.0.1  normal       82.348GB 1.1
    0.0.2  normal       82.348GB 1.2
    0.0.3  normal       82.348GB 1.3
 1  247.045GB online
  1.0    247.045GB raid10
    1.0.0  normal       82.348GB 1.4
    1.0.1  normal       82.348GB 1.5
    1.0.2  normal       82.348GB 1.6
    1.0.3  normal       82.348GB 1.7
    1.0.4  normal       82.348GB 1.8
    1.0.5  normal       82.348GB 1.9
SR shelf 1> remove 0-1
Removing lblades:
0 1
Data is about to be lost! Continue? [n] y
SR shelf 1>
```

#### 1.2.7. when

`usage: when`

The `when` command lists RAID devices in the initing or recovering state and shows their I/O rate and time to completion. Time is formatted as hours:minutes:seconds.

*Example:*

```
SR shelf 1> when
0.0 90701 KBps 0:55:18 left
SR shelf 1>
```

#### 1.2.8. shutdown

`usage: shutdown`

The `shutdown` command cleanly shuts down all lblades and their component RAIDs and reboots the system.

#### 1.2.9. reboot

`usage: reboot`

The `reboot` command is an alias to the shutdown command. It is provided for user convenience.

#### 1.2.10. halt

`usage: halt`

The `halt` command cleanly shuts down all lblades and their component RAIDs. After the system is halted pressing enter will reboot the unit.

#### 1.2.11. grow

```
usage: grow lblade raidtype shelf.slot ...
```

The `grow` command is used to add a RAID to an existing lblade. The command syntax is identical to that of the `make` command. Once added to an lblade, a RAID may not be removed. The only method for reallocating the storage associated with an lblade is to remove the entire lblade with the `remove` command.

*Example:*

```
SR shelf 1> list -l
 0  247.045GB online
  0.0    247.045GB raid5 initing 127661
    0.0.0  normal       82.348GB 1.0
    0.0.1  normal       82.348GB 1.1
    0.0.2  normal       82.348GB 1.2
    0.0.3  normal       82.348GB 1.3
SR shelf 1> grow 0 raid5 1.4-9
SR shelf 1> list -l
 0  658.786GB online
  0.0    247.045GB raid5 initing 133506
    0.0.0  normal       82.348GB 1.0
    0.0.1  normal       82.348GB 1.1
    0.0.2  normal       82.348GB 1.2
    0.0.3  normal       82.348GB 1.3
  0.1    411.741GB raid5 initing 1042
    0.1.0  normal       82.348GB 1.4
    0.1.1  normal       82.348GB 1.5
    0.1.2  normal       82.348GB 1.6
    0.1.3  normal       82.348GB 1.7
    0.1.4  normal       82.348GB 1.8
    0.1.5  normal       82.348GB 1.9
SR shelf 1>
```

#### 1.2.12. spare

```
usage: spare [ shelf.slot ... ]
```

The `spare` command is used to list and manage the spare device pool. Without arguments, `spare` will list all devices in the spare pool. For each device listed, `spare` will try to recruit the device as a spare. If the device is already in use, an error will occur.

*Example:*

```
SR shelf 1> spare
SR shelf 1> spare 1.10-11
SR shelf 1> spare
1.10   82.348GB
1.11   82.348GB
SR shelf 1>
```

#### 1.2.13. rmspare

`usage: rmspare shelf.slot ...`

The `rmspare` command is used to remove devices from the spare pool. One or more devices may be specified as arguments.

*Example:*

```
SR shelf 1> spare
1.10   82.348GB
1.11   82.348GB
SR shelf 1> rmspare 1.10
spare 1.10 removed
SR shelf 1>
```

#### 1.2.14. mask

`usage: mask [ lblade ... [ +mac ... ] [ -mac ... ] ]`

The `mask` command is used to manage client access to exported lblades. Without any arguments, `mask` will list all lblades performing mac masking and their mask lists. Given only lblade arguments, `mask` will list the mac mask list for all specified lblades. Further arguments are directives to add `+` or remove `-` mac addresses from the mask list.

*Example:*

```
SR shelf 1> mask
0 000E0C65BAA3
SR shelf 1> mask 0 +000E0C65BAA2 +000E0C65BAA4 +000E0C65BAA5
SR shelf 1> mask
0  000E0C65BAA5 000E0C65BAA4 000E0C65BAA2 000E0C65BAA3
SR shelf 1> mask 0 -000E0C65BAA4
SR shelf 1> mask
0  000E0C65BAA5 000E0C65BAA2 000E0C65BAA3
SR shelf 1>
```

#### 1.2.15. fail

`usage: fail lblade.part.drive`

The `fail` command changes the state of a drive to failed. It is a convenient method for testing appliance behaviour in failure conditions.

*Example:*

```
SR shelf 1> list -l
 0    0.105GB online
  0.0      0.105GB raid1
    0.0.0  normal        0.105GB 1.0
    0.0.1  normal        0.105GB 1.1
SR shelf 1> spare
 1.2    0.105GB
SR shelf 1> fail 0.0.1
SR shelf 1> list -l
 0    0.105GB online
  0.0      0.105GB raid1 needrecover recovering degraded 739
    0.0.0  normal        0.105GB 1.0
    0.0.1  replaced      0.105GB 1.2
SR shelf 1>
```

#### 1.2.16. unfail

`usage: unfail lblade.part.drive`

The `unfail` command changes the state of a failed drive to replaced, essentially replacing it with itself. After replacement, the drive will be reconstructed.

*Example:*

```
SR shelf 1> list -l
 0    1.235GB online
  0.0      1.235GB raid5 degraded
    0.0.0  normal        0.412GB 1.0
    0.0.1  normal        0.412GB 1.1
    0.0.2  normal        0.412GB 1.2
    0.0.3  failed        0.412GB 1.4
SR shelf 1> unfail 0.0.3
SR shelf 1> list -l
 0    1.235GB online
  0.0      1.235GB raid5 needrecover recovering degraded 2939
    0.0.0  normal        0.412GB 1.0
    0.0.1  normal        0.412GB 1.1
    0.0.2  normal        0.412GB 1.2
    0.0.3  replaced      0.412GB 1.4
SR shelf 1>
```

#### 1.2.17. replace

`usage: replace lblade.part.drive shelf.slot`

The `replace` command replaces a failed drive with a new drive. After replacement, the drive will be reconstructed.

*Example:*

```
SR shelf 1> list -l
 0    1.235GB online
  0.0      1.235GB raid5 degraded
    0.0.0  normal        0.412GB 1.0
    0.0.1  normal        0.412GB 1.1
    0.0.2  normal        0.412GB 1.2
    0.0.3  failed        0.412GB 1.4
SR shelf 1> replace 0.0.3 1.3
SR shelf 1> list -l
 0    1.235GB online
  0.0      1.235GB raid5 needrecover recovering degraded 1426
    0.0.0  normal        0.412GB 1.0
    0.0.1  normal        0.412GB 1.1
    0.0.2  normal        0.412GB 1.2
    0.0.3  replaced      0.412GB 1.3
SR shelf 1>
```

#### 1.2.18. online

```
usage: online [ lblade ... ]
```

The `online` command is used to move one or more lblades into the online state, enabling them for network access. Without an argument, `online` lists all lblades currently online.

*Example:*

```
SR shelf 1> online 0
SR shelf 1> online
 0    82.35GB online
SR shelf 1>
```

#### 1.2.19. offline

```
usage: offline [ lblade ... ]
```

The `offline` command is used to move one or more lblades into the offline state. While offline, lblades will not be accessible from the network. Without an argument, `offline` lists all lblades currently offline.

*Example:*

```
SR shelf 1> offline 0
SR shelf 1> offline
 0    82.35GB offline
SR shelf 1>
```

#### 1.2.20. sync

```
usage: sync
```

The `sync` command writes all dirty (unwritten) data buffers to disk. This command should be called before the appliance is halted in an unusual manner (e.g. before disconnecting power) to avoid data loss. To ensure no new dirty buffers are created after the sync call, all lblades exported by the appliance should be in an `offline` state before issuing the sync command.

#### 1.2.21. update

```
usage: update [ -l ]
```

The `update` command is used to process the **tarc** file on the update lblade. With the `-l` (the letter "l") flag `update` will display the lblade being used as the update target without performing any action. For an explanation of the software update procedure, please see **Appendix C**.

*Example:*

```
SR shelf 1> update -l
0
SR shelf 1> update
Updating kernel ... done.
Updating root fs ... done.
Update successful. Please shutdown and reboot to use the new software.
SR shelf 1>
```

#### 1.2.22. syslog

```
usage: syslog [ -cp ] [ -s severity ] 'message'
```

The `syslog` command is used to send syslog messages to a remote syslog host at UDP port 514. Given the `-c` flag, `syslog` will enter a dialogue for the administrator to specify the source and destination IP used in the syslog datagram. The destination IP should be set to the desired syslog host. The administrator need only set this configuration once; future calls to syslog will use the stored information. Given the `-p` flag, `syslog` will print the stored source and destination ip addresses used in the syslog datagram. The `-s` flag requires an integer argument and sets the syslog severity for this message only.

If unset, the source IP address used in the syslog UDP datagram is 205.185.197.30. The syslog facility used is *local0 (16)* and if unspecified, the severity is *informational (6)*. Syslog messages are only sent out the first interface to avoid duplication in certain network configurations.

Configuration of the syslog host is beyond the scope of this document.

`Syslog` always prints its message to the console.

*Example:*

```
SR shelf 1> syslog -c
Configuring syslog destination.  Enter addresses in dotted notation.
IPv4 destination address [205.185.197.1]: 192.168.0.1
IPv4 source address []: 192.168.0.30
Configuration successful.
SR shelf 1> syslog -p
destination: 192.168.0.1
     source: 192.168.0.30
SR shelf 1>
```

#### 1.2.23. release

```
usage: release
```

The `release` command prints the release date of the currently running firmware.

*Example:*

```
SR shelf 1> release
Fri May 19 13:34:33 EDT 2006
SR shelf 1>
```

#### 1.2.24. exit

```
usage: exit
```

The `exit` command exits the command line interface. If a password has been set with the `passwd` command, the login prompt is issued.

#### 1.2.25. cecon

```
usage: cecon [ interface ]
```

The `cecon` command is used to turn on CEC for a specified interface. Without an argument, `cecon` lists all interfaces for which CEC is served. The interface specification should be either /net/ether0 or /net/ether1.

*Example:*

```
SR shelf 1> cecon /net/ether1
SR shelf 1> cecon
/net/ether1
SR shelf 1>
```

#### 1.2.26. cecoff

```
usage: cecoff interface
```

The `cecoff` command is used to turn off CEC on a specified interface. The interface specification should be either /net/ether0 or /net/ether1.

*Example:*

```
SR shelf 1> cecon
/net/ether1
SR shelf 1> cecoff /net/ether1
SR shelf 1> cecon
SR shelf 1>
```

#### 1.2.27. jbod

```
usage: jbod shelf.slot ...
```

The `jbod` command is used to export one or more drive slots as lblades. It is provided as a convenience for administrators wishing to easily obtain a JBOD configuration. All lblades created by `jbod` are automatically brought online.

`Jbod` simply makes a linear raid lblade over a single shelf.slot, setting the lblade number to the drive slot. Administrators may choose to export the drives as other lblade numbers by creating the raidL lblade themselves. As an example, the following two sets of commands are identical in their effect:

```
SR shelf 1> jbod 1.0
```

```
SR shelf 1> make 0 raidL 1.0
SR shelf 1> online 0
```

For an example of configuring the appliance as a JBOD, please see **Appendix B**.

#### 1.2.28. sos

```
usage: sos
```

The `sos` command is used to print information to send to Coraid support. The output contains the SR release, model, sata card rev, disk information (model/serial/firmware), and lblade/raid configuration.

Providing the output of this command when first contacting support will often result in a faster solution to a problem.

#### 1.2.29. help

```
usage: help [ cmd ]
```

The `help` command will print the list of user commands to the console. Specifying a `cmd` argument will print the usage for the named command.

**Appendix A -- Command Index**

**Appendix B -- Quick Start Examples**

The following example initializes a raid5 over 14 disks in the appliance. The remaining disk is allocated as a spare.

```
SR shelf 1> show -l
   1.0     82.35GB up
   1.1     82.35GB up
   1.2     82.35GB up
   1.3     82.35GB up
   1.4     82.35GB up
   1.5     82.35GB up
   1.6     82.35GB up
   1.7     82.35GB up
   1.8     82.35GB up
   1.9     82.35GB up
   1.10    82.35GB up
   1.11    82.35GB up
   1.12    82.35GB up
   1.13    82.35GB up
   1.14    82.35GB up
SR shelf 1> list
SR shelf 1> make 0 raid5 1.0-13
SR shelf 1> spare 1.14
SR shelf 1> online 0
SR shelf 1> list -l
 0 1070.527GB online
  0.0   1070.527GB raid5 initing 1975
    0.0.0  normal       82.348GB 1.0
    0.0.1  normal       82.348GB 1.1
    0.0.2  normal       82.348GB 1.2
    0.0.3  normal       82.348GB 1.3
    0.0.4  normal       82.348GB 1.4
    0.0.5  normal       82.348GB 1.5
    0.0.6  normal       82.348GB 1.6
    0.0.7  normal       82.348GB 1.7
    0.0.8  normal       82.348GB 1.8
    0.0.9  normal       82.348GB 1.9
    0.0.10 normal       82.348GB 1.10
    0.0.11 normal       82.348GB 1.11
    0.0.12 normal       82.348GB 1.12
    0.0.13 normal       82.348GB 1.13
SR shelf 1> spare
1.14   82.348GB
SR shelf 1>
```

The following example configures the appliance as a JBOD.

```
SR shelf 1> jbod 1.0-14
SR shelf 1> list -l
 0   82.348GB online
  0.0     82.348GB raidL
    0.0.0  normal       82.348GB 1.0
 1   82.348GB online
  1.0     82.348GB raidL
    1.0.0  normal       82.348GB 1.1
 2   82.348GB online
  2.0     82.348GB raidL
    2.0.0  normal       82.348GB 1.2
 3   82.348GB online
  3.0     82.348GB raidL
    3.0.0  normal       82.348GB 1.3
 4   82.348GB online
  4.0     82.348GB raidL
    4.0.0  normal       82.348GB 1.4
 5   82.348GB online
  5.0     82.348GB raidL
    5.0.0  normal       82.348GB 1.5
 6   82.348GB online
  6.0     82.348GB raidL
    6.0.0  normal       82.348GB 1.6
 7   82.348GB online
  7.0     82.348GB raidL
    7.0.0  normal       82.348GB 1.7
 8   82.348GB online
  8.0     82.348GB raidL
    8.0.0  normal       82.348GB 1.8
 9   82.348GB online
  9.0     82.348GB raidL
    9.0.0  normal       82.348GB 1.9
10   82.348GB online
  10.0    82.348GB raidL
    10.0.0 normal       82.348GB 1.10
11   82.348GB online
  11.0    82.348GB raidL
    11.0.0 normal       82.348GB 1.11
12   82.348GB online
  12.0    82.348GB raidL
    12.0.0 normal       82.348GB 1.12
13   82.348GB online
  13.0    82.348GB raidL
    13.0.0 normal       82.348GB 1.13
14   82.348GB online
  14.0    82.348GB raidL
    14.0.0 normal       82.348GB 1.14
SR shelf 1>
```

**Appendix C -- Updating**

Updating the SR Appliance is accomplished by a three step process:

- Make a RAM based update lblade using `make`.
- Copy the appropriate **tarc** update file to the update lblade from a client system.
- Run the update command on the appliance to process the **tarc** update.

The **tarc** update file contains a kernel update and a root filesystem update. Command updates are performed as a process of updating the flash root filesystem.

The following example performs the above three steps to update an SR Appliance. The client system used is Linux with a 2.6 kernel. Updating from a different client system (solaris, freebsd, etc) is analagous; simply write the update tarc file to the raw block device represented by the update lblade.

*Example:*

```
SR shelf 1> make 0 update
SR shelf 1> update -l
0
SR shelf 1> list -l
 0    0.034GB online
  0.0      0.034GB raidL
    0.0.0  normal        0.034GB update
SR shelf 1>
```

*On the client system:*

```
# modprobe aoe
# aoe-stat
     e1.0            eth1              up
# ls
SR20060927.tarc
# dd if=SR20060927.tarc of=/dev/etherd/e1.0
3400+0 records in
3400+0 records out
#
```

*Back on the SR Appliance:*

```
SR shelf 1> update
Updating kernel ... done.
Updating root fs ... done.
Update successful. Please shutdown and reboot to use the new software.
SR shelf 1> shutdown
cpu0: exiting
someone's exiting
cpu1: exiting
calling arch->reset() 80157e48
apshutdown: active = 0x00
calling arch->reset() 80157e48
```

**Appendix D -- Support**

Coraid support for the SR appliance is accomplished via email through `support@coraid.com`. Since our line of appliances is primarily command line based, text-based support enables us to isolate problems and clearly communicate solutions. This is uncomfortable for users at first as we've all been subject to poor email support and autogenerated responses. Coraid takes great pride in our level of customer support, providing top level engineers as first responders. In order to make this interaction most effective it is important to provide detailed information about the problem at hand. In many cases, our engineers can provide a solution in the first response.

For general questions about the SR, please check the SR FAQ at the SR support page before contacting Coraid support to see if the question has already been answered.

For SR problems, the best information to start a support communication with is the output from the `sos` command. This information gives us a snapshot of your SR and in many cases will dictate the problem at hand. If the SR firmware is older and does not include the `sos` command, the output of the following commands is a good start:

```
release
list -l
show -l
show -c
grep . /raiddev/*/stat
```

Use the serial console or CEC (the Coraid Ethernet Console) to get real text to send us instead of summarizing what's going on. For example, the output of "list -l" will be much more helpful than saying something like, "list -l shows the same thing." The last item in the command list above is "grep <space> <dot> <space>".

Include a description of how the networking is configured. For simplicity in eliminating potential problems, it's best to start with a single cable between the SR and the client system. The SR has automatic MDI/MDI-X capability and will auto-negotiate crossover.

Our high quality email support routinely converts customers who are accustomed to tiered phone support. Email support doesn't require a complete halt to all other activities -- a must in today's multitasking work environment. Customers enjoy being able to have past solutions easily referenced from their inbox. It can take a little getting used to, but in our experience it's a big win for us and for our customers. We sincerely hope you'll agree.