

Appendix A. Network Control

This appendix discusses the control of ethernet on client systems when the client interacts directly with Enstore via `encp`, as opposed to interacting with the dCache as a front-end.

A.1 Default Routing for Encp

By default, `encp` uses the DNS name obtained by the `hostname` command for control messages. `Encp` uses whatever routing the client system or network administrator sets between the client system and the Enstore system for data transfers.

Typically, the default configuration suffices for machines having a single network interface, or having a single network interface dedicated to data movement.

A.2 Routing via the `enstore.conf` File

For large client machines with multiple network connections (interfaces), each network interface is attached to a different (virtual) router. A one-to-one mapping is made between the IP address of each interface and the IP of the router it is attached to. The Computing Division's networking group must perform the configuration for this.

Administrators can create an `enstore.conf` file to configure the routing to allow for multiple network interface cards dedicated to Enstore, and/or to allow for a different IP address to be used for the `encp` control socket. The default location for the file is `/etc/enstore.conf`. The location of the file can be overridden with the environment variable `ENSTORE_CONF`.

The file format supports comments, a host ip line, and zero or more interface lines (all these line types are optional).

Comment Lines

Comment lines begin with a "pound" sign, "#", e.g.,:

```
# this is s comment line
```

Hostip Line

The `hostip` line gives the host IP address used to override the DNS name that `encp` uses to `bind` with. This is used when doing a (passive) open on a socket used to listen for a call back from the mover. One and only one hostname line is required per `enstore.conf` file. For example:

```
hostip=131.225.42.42
```

Interface Lines

Lines starting with `interface` are used in this file to specify more than one network interface for data transfers. An `enstore.conf` file would typically contain either zero or at least two of these lines, since a single interface can be controlled more conventionally with static routes.

This functionality is known to work on IRIX machines. It has not been tested on SunOS or OSF1, although it is expected to work; the Linux kernel doesn't support it.

The underlying implementation mechanisms for multiple interfaces are portable, and the scheme can be extended on demand. When more than two interfaces are used, it is necessary to have each interface on its own subnet. (The system or network administrator needs to configure the subnets.) The `enroute2` executable (part of the `encp` product) must be installed and have `setuid root` on the machine in order to enable this feature.

In order for the interface lines in the `/etc/enstore.conf` to be used; an executable named `enroute2` with the `setuid` bit turned on needs to be in `encp`'s path. This executable is included with the `encp` product from **UPS/UPD**, but the `setuid` bit is not set by default. The search path for this executable with the `setuid` bit set is:

- 1) `$ENROUTE2`
- 2) `$ENCP_DIR/enroute2`
- 3) `$ENSTORE_DIR/enroute2`
- 4) `/usr/local/bin/enroute2`
- 5) `/etc/enroute2`

An `interface` line must specify four keywords (with an optional fifth for IRIX):

- `interface` specifies the network device.
- `weight` specifies the relative capacity of each interface. For example, if 1 Gb/s and 100 Mb/s interfaces are used, they might be assigned weights of 30 and 10, respectively¹.
- `ip` specifies the ip address corresponding to the device given by `interface`.

- `gw` specifies the ip address corresponding to the gateway to the Enstore movers for the device given by `interface`. (Get this information from the networking admins.)
- `cpu` The `cpu` keyword is used on IRIX systems only. Its use is desirable for minimizing the amount of CPU used per transfer, though it is technically not required. The performance enhancements will take effect if two conditions are met: If the CPU used by `encl` has hardware affinity with the slot holding the network card, and if the same CPU performs interrupt service for the network card.

For example, a file may contain two `interface` lines as follows:

```
interface=eg2 weight=30 cpu=2 gw=131.225.32.32 ip=131.225.32.31
interface=eg3 weight=10 cpu=3 gw=131.225.32.36 ip=131.225.32.35
```

1. Despite the 1Gb name, these cards typically get 30 MB/s. A 100Mb/s network card typically gets a 10/MB/s rate. Hence, the weights 30 and 10.

