

Chapter 5: Copying Files with Encp

Encp is an end-user command used to copy data files from disk to storage media and vice-versa. Its use is being discouraged in favor of the dCache, however we document it here for completeness.

Encp is maintained in KITS and in AFS product space as a separate product from Enstore, and is designed to be used in conjunction with it. Encp does not support recursive copies of data to and from Enstore; ensync is provided as a wrapper to encp for that purpose when writing to Enstore, see section Chapter 6: *Copying Directory Structures with Ensynchron*. Encp can copy multiple files to a single directory only. Encp can be used only from on-site machines in the fnal.gov domain. For off-site use, see section Chapter 4: *Using the dCache to Copy Files to/from Enstore*.

In this chapter, we assume you have UPS/UPD running on your local machine.

5.1 Setup encp

To setup **encp**, run the command:

```
% setup -q <qualifier> encp
```

where **<qualifier>** stands for one of the Enstore system hosts. Currently, these include:

| | |
|--------------|--------------------------------------|
| stken | for general Fermilab (and CMS) users |
| d0en | for D0 users |
| cdfen | for CDF users |

For example, a CDF experimenter would type:

```
$ setup -q cdfen encp
```

If you don't specify the qualifier, the environment variable ENSTORE_CONFIG_HOST may get set to the wrong value. Check that ENSTORE_CONFIG_HOST specifies the correct server.

5.2 Encp Command Syntax and Usage

Encp plays the same role in the Enstore system that `cp` plays in UNIX. The syntax is:

```
% encp [<options>] <source_file> <destination_file>
```

With the exception of the option `--help`, we defer the list and definitions of options to section 5.5 *Encp Command Options*, and instead proceed with usage information.

Use the `--help` option to request the option listing for `encp` (we give the option listing in section 5.5 *Encp Command Options*), or the `--usage` option for syntax information:

```
% encp --usage

encp [OPTIONS]... <source file> <destination file>
encp [OPTIONS]... <source file> [source file [...]]\
<destination>
```

5.3 Copy Files to and from Enstore Media

5.3.1 Run encp

First, setup `encp` (using the `-q` flag). You can use filename expansion (wildcard characters to specify a group of files). We recommend, however, that you copy one file at a time. Run the command as follows to copy a file to Enstore:

```
% encp [<options>] /<path-to>/.../<localfilename> \
  /pnfs/<storage-group>/.../<targetdir>/<remotefilename>
```

The presence of `/pnfs/` in the destination path indicates that this is a copy to the Enstore system (see section 1.2 *PNFS Namespace*). To copy from Enstore, change the source and destination file specifications, e.g.:

```
% encp [<options>] \
  /pnfs/<storage-group>/.../<targetdir>/<remotefilename> \
  /<path-to>/.../<localfilename>
```

5.3.2 Examples

- 1) Standard copy to Enstore; no options. Copy `myfile` to the directory

```
/pnfs/expt1/subdir/:
```

```
% encp /path/to/myfile /pnfs/expt1/subdir/
```

- 2) Standard copy; no options. Download `/pnfs/expt1/subdir/myfile` to a different local directory from the cwd, and change the filename:

```
% encp /pnfs/expt1/subdir/myfile \  
/other/local/dir/newfilename
```

- 3) Request the process to output some information to screen (`--verbose`). Again, copy `myfile` to the directory `/pnfs/expt1/subdir/`:

```
% encp --verbose 3 /path/to/myfile \  
/pnfs/expt1/subdir/
```

- 4) Copy all the files in the cwd starting with the string `trigger1` to the directory `/pnfs/expt1/subdir/`:

```
% encp ./trigger1* /pnfs/expt1/subdir/
```

- 5) Copy all the files in `/pnfs/expt1/subdir/` starting with the string `trigger1` to the cwd:

```
% encp /pnfs/expt1/subdir/trigger1* .
```

5.4 More about Encp

5.4.1 Preventing Unwanted Overwriting

When an `encp` job starts, it first creates a zero length output file for every input file. In this way it reserves the necessary filenames and thus prevents another party from starting a competing `encp` process which would clobber the first.

5.4.2 Killing an `encp` Job

There are four traditional ways to abort a process:

- Ctrl-C (SIGINT)
- Ctrl-\ (SIGQUIT)
- kill (SIGTERM)
- kill -9 (SIGKILL)

The first three result in encp removing any remaining zero length files (as discussed directly above). With a “kill -9”, no cleanup occurs. For multi-file transfers, files successfully transferred before the signal is caught will be left alone.

5.4.3 Isolating Source of Bottlenecks

Encp (as of v3_1) supports isolating the rate transfers in the tape, disk and network via the option `--threaded` used in conjunction with the option `--verbose` with a value of 1 or higher. If `--threaded` is not specified, then the network and disk rates are calculated the same way as before, and display the same value as one another.

Here is an example without `--threaded` (with off-topic output removed for brevity):

```
% encp --verbose 1 /pnfs/xyz/10MB_002 /tmp/myfile
```

with output:

```
...
Transfer /pnfs/xyz/10MB_002 -> /tmp/myfile: 10485760 bytes
copied from 'TEST01' at 1.57 MB/S
(1.67 MB/S network) (2.87 MB/S drive) (1.67 MB/S disk)
...
Completed transferring 10485760 bytes in 1 files in
14.2875500917 sec.
Overall rate = 0.7 MB/sec. Drive rate = 2.87 MB/sec.
Network rate = 1.67 MB/sec. Exit status = 0.
```

Note in the above output, the network and disk rates are the same.

Here is an example with `--threaded` and `verbose 1` (abbreviated output); note that the rates are separated, so that you can see where the bottleneck is (the disk, in this case):

```
% encp --verbose 1 --threaded /pnfs/xyz/10MB_002 /tmp/myfile
```

It produces output:

```
...
Transfer /pnfs/xyz/10MB_002 -> /tmp/myfile:
10485760 bytes copied from 'TEST01' at 2.41 MB/S
(8.09 MB/S network) (9.36 MB/S drive) (2.71 MB/S disk)
...
Completed transferring 10485760 bytes in 1 files in
14.9129179716 sec.
Overall rate = 0.671 MB/sec. Drive rate = 9.36 MB/sec.
Network rate = 8.09 MB/sec. Exit status = 0.
```

The network and drive each have rates above 8 MB/s, and the disk rate is only 2.71 MB/s.

5.4.4 Encp Error Handling

Encp has functionality to retry and resubmit requests, where we distinguish between these two terms. Encp will *retry* (i.e., resend) a request after an error occurs. Encp will *resubmit* a request if it has been waiting for a mover for over 15 minutes; this is not due to an error condition but rather to keep queues current regardless of the server condition.

There are two general classifications of errors in encp: those that can be retried and those that can't. Three "retriable" errors can occur before the error "TOO_MANY_RETRIES" occurs.

The most common nonretriable errors include:

| | |
|------------|--|
| NOACCESS | the system has marked the volume as "potentially" bad |
| NOTALLOWED | an enstore administrator has marked a tape as unavailable for user access |
| USERERROR | usually is a file accessibility problem (doesn't exist, has wrong permissions, etc.) |

Among the less common ones, there are:

| | |
|------------------|--|
| VERSION_MISMATCH | the encp version is no longer compatible with the running Enstore system |
| CRC_MISMATCH | indicates a corruption error |

Ask your Enstore administrator if you see others.

5.4.5 Finding files in different Enstore systems

| | |
|-------------|--|
| File reads: | When reading from Enstore, encp can determine whether the current value of \$ENSTORE_CONFIG_HOST (see section 2.2 <i>Important Environment Variables</i>) is pointing to the Enstore system that contains the requested file. If it points to the wrong one, encp will try the other Enstore installations to find the requested file. If the file is found, encp will retrieve the file; if the file is not found on any Enstore system, an error is returned to the user. |
|-------------|--|

| | |
|--------------|---|
| File writes: | When writing to Enstore; the value of \$ENSTORE_CONFIG_HOST is always used. |
|--------------|---|

5.4.6 Order of Processing Queued Requests

For reads, files are sorted out by volume. When all files from a single volume are complete, the next volume's files are requested.

For writes, one file at a time is submitted to the library manager. The order is that in which the files are specified on the command line. The tape is kept mounted during file writes on a best-effort basis. See Chapter 10: *Job Priority and Queue Management* for more information.

5.4.7 Calculating the CRC of a Local File

There is a small program called `ecrc` that calculates the CRC of a local file. It is used in this way:

```
% ecrc <filename>
```

For example,

```
$ ecrc ~/test_files/10MB_002
      size 10485760 buf_size 1048576 blocks 10 rest 0
      CRC 1294565006
```

To see what CRC information Enstore knows, see section 8.4 *enstore pnfs*, in particular the `--xref` option of the `enstore pnfs` command.

5.5 Encp Command Options

In this section, we've placed a bomb in front of any option that should be used with utmost care; these options, if misused, can adversely affect not only your jobs, but those of others, as well. We've placed a pointing finger in front of options that, if misused, may adversely affect your own job, but not others' jobs.



`--age-time <AGE_TIME>` Specifies the time period, in minutes, after which the priority is eligible to change from the initial job priority. We recommend that you don't set this, just use the default (which is "never").



`--array-size <ARRAY_SIZE>` Sets the number of buffers in the array. If `--threaded` is specified but this option is not, `array-size` defaults to 3. If this is used without `--threaded`, this value

becomes 1 and is ignored. Changing this value for multi-threaded transfers may increase transfer speed.



`--buffer-size <BUFFER_SIZE>`

Sets the number of bytes of data to transfer at one time (default is 256k). Increasing this value may increase transfer speed. This value must remain lower than the available memory.



`--bypass-filesystem-max-filesize-check`

Disables the check to protect against the user reading from Enstore a file larger than the maximum size file the local filesystem supports. Use this switch with care.

`--data-access-layer`

Turns on special status printing; output has standardized format whether error occurred or not.



`--delayed-dismout <DELAY>` Specifies time period in minutes to delay dismount of volume. Use this to tell Enstore: “More work is coming for the volume, don’t dismount the volume too quickly once the current transfer is completed.”



`--delpri <DELPRI>`

Changes the initial job priority by specified value after a period given by the age-time switch. We recommend that you don’t set this, just use the default (1).



`--direct-io`

Uses direct I/O for disk access on supporting file systems¹. Generally, direct I/O makes disk access slower. But when the size of the read/write buffer is made large enough, say, 64Mb or larger, direct I/O is faster because of the skipped memory-to-memory copy.

1. Direct I/O is not universally supported; some filesystems, versions of filesystems, kernels, etc. do not support it. If this doesn’t work for you, contact an enstore admin, and communicate your kernel, library versions, filesystem and filesystem version.

| | | |
|---|-----------------------------|--|
| | --ecrc | (stands for Enstore crc) This can be used when reading from Enstore. After a file is written to disk, this causes Enstore to reread the disk copy of the file and recalculate the checksum on it. |
|  | --ephemeral ¹ | This option creates a temporary file family of name “ephemeral”, and copies files to this ephemeral file family on storage media in the order specified. Overrides file family tag in /pnfs destination directory. |
|  | --file-family <FILE_FAMILY> | This is used to write data on volumes assigned to specified file family. Overrides file family tag in /pnfs destination directory. (Footnote for --ephemeral applies here, too.) |
| | --help | Displays the list of options for encp. |
|  | --map-size <MMAP_SIZE> | The amount of data to map from the file to local memory at one time in bytes (default is 96Mb); use with --mmap-io. |
|  | --mmap-io | Uses memory-mapped I/O for disk access on supporting file systems (see the <i>Enstore Glossary</i> for an explanation). Make sure you have read and write permissions on the file. |
| | --no-crc | Tells encp to bypass the crc ² on the local file. (For the minor performance enhancement that this affords, you lose both the encp crc and the one performed by the mover; we discourage use of this option.) |
|  | --priority <PRIORITY> | Sets the initial job priority to the specified integer value. We recommend that you don’t set this, just use the default. |
| | --threaded | Multithreads the actual data transfer. |

1. The options --ephemeral and --file-family require care when used so that tapes do not get mounted in a way that causes improper and/or inefficient tape usage. Beware of runaway scripts!

2. CRC stands for Cyclic Redundancy Check, a type of checksum.

| | |
|--------------------------------------|---|
| <code>--usage</code> | Displays information about the encp options. |
| <code>--verbose <LEVEL></code> | Changes the amount of information printed about the transfer; provide an integer value. Default is 0. Larger integer numbers provide more “verbosity”. Largest meaningful number may change as development continues. |
| <code>--version</code> | Displays encp version information. |
| <code>--pnfs-is-automounted</code> | Typically, users should not automount pnfs. If you do, you can specify this option. It alerts encp to retry errors due to known OS automounting problems. Do not use this in non-automounted cases; it can slow the setup of the transfer. |



If you feel compelled to set `--priority`, `--delpri` or `--age-time`, please email enstore-admin@fnal.gov first with an explanation, as the defaults should work in almost all cases and changing them may affect other users. Priority goes in strict number sequence, where a higher number means higher priority. Note that Enstore’s selection of which file to transfer at a given time uses a much more complicated algorithm than simple priority, however. See Chapter 10: *Job Priority and Queue Management*.

