

**NAME**

**rmt** – remote magnetic tape server

**SYNOPSIS**

**rmt**

**DESCRIPTION**

**Rmt** provides remote access to files and devices for **tar**(1), **cpio**(1), and similar backup utilities. It is normally called by running **rsh**(1) or **ssh**(1) to the remote machine, optionally using a different login name if one is supplied.

The calling program communicates with **rmt** by sending requests on its standard input and reading replies from the standard output. A request consists of a request letter followed by an argument (if required) and a newline character. Additional data, if any, are sent after the newline. On success, **rmt** returns

**A***number*\n

where *number* is an ASCII representation of a decimal return code. Additional data are returned after this line. On error, the following response is returned:

**E***errno*\n*error-message*\n

where *errno* is one of the system error codes, as described in **errno**(3), and *error-message* is a one-line human-readable description of the error, as printed by **perror**(3).

Available commands and possible responses are discussed in detail in the subsequent section.

**COMMANDS**

**O***device*\n*flags*\n

Opens the *device* with given *flags*. If a device had already been opened, it is closed before opening the new one.

**Arguments**

*device* The name of the device to open.

*flags* Flags for **open**(2): a decimal number, or any valid **O\_\*** constant from **fcntl.h** (the initial **O\_** may be omitted), or a bitwise or (using **|**) of any number of these, e.g.:

```
576
64 | 512
CREAT | TRUNC
```

In addition, a combined form is also allowed, i.e. a decimal mode followed by its symbolic representation. In this case the symbolic representation is given preference.

**Reply**

**A0**\n on success.

**Extensions**

BSD version allows only decimal number as *flags*.

**C**[*device*]\n

Close the currently open device.

**Arguments**

Any arguments are silently ignored.

**Reply**

**A0**\n on success.

**L***whence*\n*offset*\n

Performs an **lseek**(2) on the currently open device with the specified parameters.

**Arguments**

*whence* Where to measure offset from. Valid values are:

0, SET, SEEK_SET	seek from the file beginning
1, CUR, SEEK_CUR	seek from the current location
2, END, SEEK_END	seek from the file end

**Reply**

**Aoffset\n** on success. The *offset* is the new offset in file.

**Extensions**

BSD version allows only 0,1,2 as *whence*.

**Rcount\n**

Read *count* bytes of data from the current device.

**Arguments**

*count* number of bytes to read.

**Reply**

On success:

**Ardcount\n**

followed by *rdcount* bytes of data read from the device.

**Wcount\n**

Writes data onto the current device. The command is followed by *count* bytes of input data.

**Arguments**

*count* Number of bytes to write.

**Reply**

On success: **Awrcount\n**, where *wrcount* is the number of bytes actually written.

**Iopcode\ncount\n**

Perform a **MTIOCOP ioctl(2)** command with the specified parameters.

**Arguments**

*opcode* **MTIOCOP** operation code.

*count* mt\_count.

**Reply**

On success: **A0\n**.

**S\n**

Returns the status of the currently open device, as obtained from a **MTIOCGET ioctl(2)** call.

**Arguments**

None

**Reply**

On success: **Acoun\n** followed by *count* bytes of data.

**SEE ALSO**

**tar(1)**.

**BUGS**

Using this utility as a general-purpose remote file access tool is discouraged.

**BUG REPORTS**

Report bugs to <bug-tar@gnu.org>.

**HISTORY**

The **rmt** command appeared in 4.2BSD. The GNU **rmt** is written from scratch, using the BSD specification.

**COPYRIGHT**

Copyright © 2013 Free Software Foundation, Inc.

License GPLv3+: GNU GPL version 3 or later <<http://gnu.org/licenses/gpl.html>>

This is free software: you are free to change and redistribute it. There is NO WARRANTY, to the extent permitted by law.