

NAME

rx, rb, rz – XMODEM, YMODEM, ZMODEM (Batch) file receive

SYNOPSIS

rz [- +8abeOpqRtTuUvy]
rb [- +abqRtuUvy]
rx [- abceqRtuUv] *file*
[-][v]**rzCOMMAND**

DESCRIPTION

This program uses error correcting protocols to receive files over a dial-in serial port from a variety of programs running under PC-DOS, CP/M, Unix, and other operating systems. It is invoked from a shell prompt manually, or automatically as a result of an "sz file ..." command given to the calling program.

While *rz* is smart enough to be called from *cu(1)*, very few versions of *cu(1)* are smart enough to allow *rz* to work properly. Unix flavors of Professional-YAM are available for such dial-out application.

Rz (Receive ZMODEM) receives files with the ZMODEM batch protocol. Pathnames are supplied by the sending program, and directories are made if necessary (and possible). Normally, the "rz" command is automatically issued by the calling ZMODEM program, but some defective ZMODEM implementations may require starting *rz* the old fashioned way.

Rb receives file(s) with YMODEM, accepting either standard 128 byte sectors or 1024 byte sectors (YAM sb -k option). The user should determine when the 1024 byte block length actually improves throughput without causing lost data or even system crashes.

If True YMODEM (Omen Technology trademark) file information (file length, etc.) is received, the file length controls the number of bytes written to the output dataset, and the modify time and file mode (iff non zero) are set accordingly.

If no True YMODEM file information is received, slashes in the pathname are changed to underscore, and any trailing period in the pathname is eliminated. This conversion is useful for files received from CP/M systems. With YMODEM, each file name is converted to lower case unless it contains one or more lower case letters.

Rx receives a single *file* with XMODEM or XMODEM-1k protocol. The user should determine when the 1024 byte block length actually improves throughput without causing problems. The user must supply the file name to both sending and receiving programs. Up to 1023 garbage characters may be added to the received file.

Rz may be invoked as **rzCOMMAND** (with an optional leading - as generated by login(1)). For each received file, *rz* will pipe the file to "COMMAND filename" where filename is the name of the transmitted file with the file contents as standard input.

Each file transfer is acknowledged when COMMAND exits with 0 status. A non zero exit status terminates transfers.

A typical use for this form is *rzrmail* which calls *rmail(1)* to post mail to the user specified by the transmitted file name. For example, sending the file "caf" from a PC-DOS system to *rzrmail* on a Unix system would result in the contents of the DOS file "caf" being mailed to user "caf".

On some Unix systems, the login directory must contain a link to COMMAND as login sets SHELL=rsh

which disallows absolute pathnames. If invoked with a leading “v”, *rz* will be verbose (see **v** option). The following entry works for Unix SYS III/V:

rzrmail::5:1::bin:/usr/local/rzrmail

If the SHELL environment variable includes *rsh*, *rbash* or *rksh* (restricted shell), *rz* will not accept absolute pathnames or references to a parent directory, will not modify an existing file, and removes any files received in error.

If **rz** is invoked with stdout and stderr to different datasets, Verbose is set to 2, causing frame by frame progress reports to stderr. This may be disabled with the **q** option.

OPTIONS

The meanings of the available options are:

-+, --append

append received data to an existing file (ZMODEM, ASCII only).

-a, --ascii

Convert files to Unix conventions by stripping carriage returns and all characters beginning with the first Control Z (CP/M end of file).

-b, --binary

Binary (tell it like it is) file transfer override.

-B NUMBER, --bufsize NUMBER

Buffer **NUMBER** bytes before writing to disk. Default is 32768, which should be enough for most situations. If you have a slow machine or a bad disk interface or suffer from other hardware problems you might want to increase the buffersize. **-1** or **auto** use a buffer large enough to buffer the whole file. Be careful with this options - things normally get worse, not better, if the machine starts to swap.

-c, --with-crc

XMODEM only. Use 16 bit CRC (normally a one byte checksum is used).

-C, --allow-remote-commands

allow remote command execution (**insecure**). This allows the sender to execute an arbitrary command through **system** () or **execl** (). Default is to disable this feature (?). This option is ignored if running in restricted mode.

-D, --null

Output file data to /dev/null; for testing. (Unix only)

--delay-startup N

Wait **N** seconds before doing anything.

-e, --escape

Force sender to escape all control characters; normally XON, XOFF, DLE, CR-@-CR, and Ctrl-X are escaped.

-E, --rename

Rename incoming file if target filename already exists. The new file name will have a dot and a number (0..999) appended.

-h, --help

give help screen.

-m N, --min-bps N

Stop transmission if BPS-Rate (Bytes Per Second) falls below **N** for a certain time (see **--min-bps-time** option).

-M N, --min-bps-time

Used together with **--min-bps**. Default is 120 (seconds).

-O, --disable-timeouts

Disable read timeout handling code. This makes *lrz* hang if the sender does not send any more, but increases performance (a bit) and decreases system load (through reducing the number of system calls by about 50 percent).

Use this option with care.

--o-sync

Open output files in synchronous write mode. This may be useful if you experience errors due to lost interrupts if update (or bdf flush or whoever this daemon is called on your system) writes the buffers to the disk.

This option is ignored and a warning is printed if your systems doesn't support O_SYNC.

-p, --protect

(ZMODEM) Protect: skip file if destination file exists.

-q, --quiet

Quiet suppresses verbosity.

-r, --resume

Crash recovery mode. lrz tries to resume interrupted file transfers.

-R, --restricted

Enter more restricted mode. lrz will not create directories or files with a leading dot if this option is given twice.

See **SECURITY** for mode information about restricted mode.

-s HH:MM, --stop-at HH:MM

Stop transmission at **HH** hours, **MM** minutes. Another variant, using **+N** instead of **HH:MM**, stops transmission in **N** seconds.

-S, --timesync

Request timesync packet from the sender. The sender sends its system time, causing lrz to complain about more then 60 seconds difference.

Lrz tries to set the local system time to the remote time if this option is given twice (this fails if lrz is not run by root).

This option makes lrz incompatible with certain other ZModems. Don't use it unless you know what you are doing.

--syslog[=off]

turn syslogging on or off. the default is set at configure time. This option is ignored if no syslog support is compiled in.

-t TIM, --timeout TIM

Change timeout to *TIM* tenths of seconds. This is ignored if timeout handling is turned of through the **O** option.

--tcp-client ADDRESS:PORT

Act as a tcp/ip client: Connect to the given port.

See **--tcp-server** for more information.

--tcp-server

Act as a server: Open a socket, print out what to do, wait for connection.

You will normally not want to use this option as lrzsz is the only zmodem which understands what to do (private extension). You might want to use this if you have to use zmodem (for which reason whatever), and cannot use the **--tcp** option of *lsz* (perhaps because your telnet doesn't allow to spawn a local program with stdin/stdout connected to the remote side).

If you use this option you have to start *lsz* with the **--tcp-client ADDRESS:PORT** option. *lrz* will print the address and port on startup.

Use of this option imposes a security risk, somebody else could connect to the port in between. See **SECURITY** for details.

-U, --unrestrict
 turn off restricted mode (this is not possible if running under a restricted shell).

--version
 prints out version number.

-v, --verbose
 Verbose causes a list of file names to be appended to stderr. More v's generate more output.

-wN, --window size N
 Set window size to N.

-X, --xmodem
 use XMODEM protocol.

-y, --overwrite
 Yes, clobber any existing files with the same name.

--ymodem
 use YMODEM protocol.

-Z, --zmodem
 use ZMODEM protocol.

SECURITY

Contrary to the original ZMODEM lrz defaults to restricted mode. In restricted mode lrz will not accept absolute pathnames or references to a parent directory, will not modify an existing file, and removes any files received in error. Remote command execution is disabled.

To use a more restricted mode set the environment variable **ZMODEM_RESTRICTED** or give the **R** option. This disables creation of subdirectories and invisible files.

Restricted mode may be turned off with the **U** option, unless lrz runs under a restricted shell.

Use of the

--tcp-client or **--tcp-server** options imposes a security risk, as somebody else could connect to the port before you do it, and grab your data. If there's strong demand for a more secure mode i might introduce some sort of password challenge.

ENVIRONMENT

lrz uses the following environment variables:

SHELL

lrz recognizes a restricted shell if this variable includes *rsh* or *rksh*

ZMODEM_RESTRICTED

lrz enters the more restricted mode if the variable is set.

EXAMPLES

(Pro-YAM command)

<ALT-2>

Pro-YAM Command: *sz *.h *.c*

(This automatically invokes *rz* on the connected system.)

SEE ALSO

ZMODEM.DOC, YMODEM.DOC, Professional-YAM, *crc(omen)*, *sz(omen)*, *usq(omen)*, *undos(omen)*

Compile time options required for various operating systems are described in the source file.

NOTES

Sending serial data to timesharing minicomputers at sustained high speeds has been known to cause lock-ups, system halts, kernel panics, and occasional antisocial behaviour. When experimenting with high speed input to a system, consider rebooting the system if the file transfers are not successful, especially if the personality of the system appears altered.

The Unix "ulimit" parameter must be set high enough to permit large file transfers.

The TTY input buffering on some systems may not allow long blocks or streaming input at high speed. You should suspect this problem when you can't send data to the Unix system at high speeds using ZMODEM, YMODEM-1k or XMODEM-1k, when YMODEM with 128 byte blocks works properly. If the system's tty line handling is really broken, the serial port or the entire system may not survive the onslaught of long bursts of high speed data.

The DSZ or Pro-YAM **zmodem l** numeric parameter may be set to a value between 64 and 1024 to limit the burst length ("zmodem pl128").

32 bit CRC code courtesy Gary S. Brown. Directory creation code from John Gilmore's PD TAR program.

BUGS

Calling *rz* from most versions of *cu(1)* doesn't work because *cu*'s receive process fights *rz* for characters from the modem.

Programs that do not properly implement the specified file transfer protocol may cause *sz* to "hang" the port for a minute or two. Every reported instance of this problem has been corrected by using ZCOMM, Pro-YAM, or other program with a correct implementation of the specified protocol.

Many programs claiming to support YMODEM only support XMODEM with 1k blocks, and they often don't get that quite right.

Pathnames are restricted to 127 characters. In XMODEM single file mode, the pathname given on the command line is still processed as described above. The ASCII option's CR/LF to NL translation merely deletes CR's; *undos(omen)* performs a more intelligent translation.

VMS VERSION

The VMS version does not set the file time.

VMS C Standard I/O and RMS may interact to modify file contents unexpectedly.

The VMS version does not support invocation as **rzCOMMAND**. The current VMS version does not support XMODEM, XMODEM-1k, or YMODEM.

According to the VMS documentation, the buffered input routine used on the VMS version of *rz* introduces a delay of up to one second for each protocol transaction. This delay may be significant for very short files. Removing the "#define BUFREAD" line from *rz.c* will eliminate this delay at the expense of increased CPU utilization.

The VMS version causes DCL to generate a random off the wall error message under some error conditions; this is a result of the incompatibility of the VMS "exit" function with the Unix/MSDOS standard.

ZMODEM CAPABILITIES

Rz supports incoming ZMODEM binary (-b), ASCII (-a), protect (-p), clobber (-y), and append (-+) requests. The default is protect (-p) and binary (-b).

The Unix versions support ZMODEM command execution.

FILES

rz.c, *crctab.c*, *rbsb.c*, *zm.c*, *zmodem.h* Unix source files.

rz.c, *crctab.c*, *vrzsz.c*, *zm.c*, *zmodem.h*, *vmodem.h*, *vvmodem.c*, VMS source files.