

NAME

`rsh` - restricted shell (command interpreter)

SYNOPSIS

`rsh` [flags] [name [arg1 ...]]

DESCRIPTION

Rsh is a restricted version of the standard command interpreter *sh*(1). It is used to set up login names and execution environments whose capabilities are more controlled than those of the standard shell. The actions of *rsh* are identical to those of *sh*, except that the following are disallowed:

- cd*
- command names containing /
- > and >>

When invoked with the name `-rsh`, *rsh* reads the user's `.profile` (from `$HOME/.profile`). It acts as the standard *sh* while doing this, except that an interrupt causes an immediate exit, instead of causing a return to command level. The restrictions above are enforced after `.profile` is interpreted.

When a command to be executed is found to be a shell procedure, *rsh* invokes *sh* to execute it. Thus, it is possible to provide to the end user shell procedures that have access to the full power of the standard shell, while restricting him to a limited menu of commands; this scheme assumes that the end user does not have write and execute permissions in the same directory.

The net effect of these rules is that the writer of the `.profile` has complete control over user actions, by performing guaranteed setup actions, then leaving the user in an appropriate directory (probably *not* the login directory). The `.profile` should make `$PATH readonly` to keep the user from changing it.

Rsh is actually just a link to *sh* and any *flags* arguments are the same as for *sh*(1).

The system administrator often sets up a directory of commands that can be safely invoked by *rsh*. Some systems also provide a restricted editor *red*.

SEE ALSO

`sh`(1), `profile`(5).