## NAME

tip – connect to remote system

## SYNOPSIS

**tip** [ **–v** ] [ *–speed-entry* ] *hostname* | *phone-number*

## AVAILABILITY

SUNWcsu

## DESCRIPTION

**tip** establishes a full-duplex terminal connection to a remote host. Once the connection is established, a remote session using **tip** behaves like an interactive session on a local terminal.

The *remote* file contains entries describing remote systems and line speeds used by **tip**.

Each host has a default baud rate for the connection, or you can specify a speed with the *–speed-entry* command line argument.

When *phone-number* is specified, **tip** looks for an entry in the *remote* file of the form:

**tip** *–speed-entry*

When it finds such an entry, it sets the connection speed accordingly. If it finds no such entry, **tip** interprets *–speed-entry* as if it were a system name, resulting in an error message.

If you omit *–speed-entry*, **tip** uses the **tip0** entry to set a speed for the connection.

When establishing the connection **tip** sends a connection message to the remote system. The default value for this message can be found in the *remote* file.

When **tip** attempts to connect to a remote system, it opens the associated device with an exclusive-open **ioctl**(2) call. Thus only one user at a time may access a device. This is to prevent multiple processes from sampling the terminal line. In addition, **tip** honors the locking protocol used by **uucp**(1C).

When **tip** starts up it reads commands from the file **.tiprc** in your home directory.

## OPTIONS

**–v** Display commands from the **.tiprc** file as they are executed.

## USAGE

Typed characters are normally transmitted directly to the remote machine (which does the echoing as well).

At any time that **tip** prompts for an argument (for example, during setup of a file transfer) the line typed may be edited with the standard erase and kill characters. A null line in response to a prompt, or an interrupt, aborts the dialogue and returns you to the remote machine.

### Commands

A tilde ( **˜** ) appearing as the first character of a line is an escape signal which directs **tip** to perform some special action. **tip** recognizes the following escape sequences:

**˜^D**
**˜.** Drop the connection and exit (you may still be logged in on the remote machine).

**˜c** [*name*]
Change directory to *name* (no argument implies change to your home directory).

**~!** Escape to an interactive shell on the local machine (exiting the shell returns you to **tip**).

**~>** Copy file from local to remote.

**~<** Copy file from remote to local.

**~p** *from* [ *to* ]
Send a file to a remote host running the UNIX system. When you use the put command, the remote system runs the command string

**cat >** *to*

while **tip** sends it the *from* file. If the *to* file is not specified, the *from* file name is used. This command is actually a UNIX-system-specific version of the '**~>**' command.

**~t** *from* [ *to* ]
Take a file from a remote host running the UNIX system. As in the put command the *to* file defaults to the *from* file name if it is not specified. The remote host executes the command string

**cat** *from* **; echo ^A**

to send the file to **tip**.

**~|** Pipe the output from a remote command to a local process. The command string sent to the local system is processed by the shell.

**~C** Connect a program to the remote machine. The command string sent to the program is processed by the shell. The program inherits file descriptors 0 as remote line input, 1 as remote line output, and 2 as tty standard error.

**~$** Pipe the output from a local process to the remote host. The command string sent to the local system is processed by the shell.

**~#** Send a BREAK to the remote system.

**~s** Set a variable (see the discussion below).

**~^Z** Stop **tip** (only available when run under a shell that supports job control, such as the C shell).

**~^Y** Stop only the "local side" of **tip** (only available when run under a shell that supports job control, such as the C shell); the "remote side" of **tip**, the side that displays output from the remote host, is left running.

**~?** Get a summary of the tilde escapes.

Copying files requires some cooperation on the part of the remote host. When a **~>** or **~<** escape is used to send a file, **tip** prompts for a file name (to be transmitted or received) and a command to be sent to the remote system, in case the file is being transferred from the remote system. While **tip** is transferring a file the number of lines transferred will be continuously displayed on the screen. A file transfer may be aborted with an interrupt.

**Auto-call Units**

**tip** may be used to dial up remote systems using a number of auto-call unit's (ACU's). When the remote system description contains the **du** capability, **tip** uses the call-unit (**cu**), ACU type (**at**), and phone numbers (**pn**) supplied. Normally **tip** displays verbose messages as it dials.

Depending on the type of auto-dialer being used to establish a connection the remote host may have garbage characters sent to it upon connection. The user should never assume that the first characters typed to the foreign host are the first ones presented to it. The recommended practice is to immediately type a **kill** character upon establishing a connection (most UNIX systems either support @ or CTRL-U as the initial kill character).

**tip** currently supports the Ventel MD-212+ modem and DC Hayes-compatible modems.

When **tip** initializes a Hayes-compatible modem for dialing, it sets up the modem to auto-answer. Normally, after the conversation is complete, **tip** drops DTR, which causes the modem to "hang up."

Most modems can be configured such that when DTR drops, they re-initialize themselves to a preprogrammed state. This can be used to reset the modem and disable auto-answer, if desired.

Additionally, it is possible to start the phone number with a Hayes **S** command so that you can configure the modem before dialing. For example, to disable auto-answer, set up all the phone numbers in **/etc/remote** using something like **pn=S0=0DT5551212**. The **S0=0** disables auto-answer.

**Remote Host Description**

Descriptions of remote hosts are normally located in the system-wide file **/etc/remote**. However, a user may maintain personal description files (and phone numbers) by defining and exporting the REMOTE shell variable. The *remote* file must be readable by **tip**, but a secondary file describing phone numbers may be maintained readable only by the user. This secondary phone number file is **/etc/phones**, unless the shell variable PHONES is defined and exported. The phone number file contains lines of the form:

*system-name phone-number*

Each phone number found for a system is tried until either a connection is established, or an end of file is reached. Phone numbers are constructed from '**0123456789−=***', where the '**=**' and '*' are used to indicate a second dial tone should be waited for (ACU dependent).

**tip Internal Variables**

**tip** maintains a set of variables which are used in normal operation. Some of these variables are read-only to normal users (root is allowed to change anything of interest). Variables may be displayed and set through the **˜s** escape. The syntax for variables is patterned after **vi**(1) and **mail**(1). Supplying **all** as an argument to the **˜s** escape displays all variables that the user can read. Alternatively, the user may request display of a particular variable by attaching a **?** to the end. For example '**˜s escape?**' displays the current escape character.

Variables are numeric (num), string (str), character (char), or Boolean (bool) values. Boolean variables are set merely by specifying their name. They may be reset by prepending a **!** to the name. Other variable types are set by appending an **=** and the value. The entire assignment must not have any blanks in it. A single set command may be used to interrogate as well as set a number of variables.

Variables may be initialized at run time by placing set commands (without the **˜s** prefix) in a **.tiprc** file in one's home directory. The **–v** option makes **tip** display the sets as they are made. Comments preceded by a **#** sign can appear in the **.tiprc** file.

Finally, the variable names must either be completely specified or an abbreviation may be given. The following list details those variables known to **tip**.

**beautify**
: (bool) Discard unprintable characters when a session is being scripted; abbreviated **be**. If the **nb** capability is present, **beautify** is initially set to **off**; otherwise, **beautify** is initially set to **on**.

**baudrate**
: (num) The baud rate at which the connection was established; abbreviated **ba**. If a baud rate was specified on the command line, **baudrate** is initially set to the specified value; otherwise, if the **br** capability is present, **baudrate** is initially set to the value of that capability; otherwise, **baudrate** is set to 300 baud. Once **tip** has been started, **baudrate** can only changed by the super-user.

**dialtimeout**
: (num) When dialing a phone number, the time (in seconds) to wait for a connection to be established; abbreviated **dial**. **dialtimeout** is initially set to 60 seconds, and can only changed by the super-user.

**disconnect**
: (str) The string to send to the remote host to disconnect from it; abbreviated **di**. If the **di** capability is present, **disconnect** is initially set to the value of that capability; otherwise, **disconnect** is set to a null string (**""**).

**echocheck**
: (bool) Synchronize with the remote host during file transfer by waiting for the echo of the last character transmitted; abbreviated **ec**. If the **ec** capability is present, **echocheck** is initially set to **on**; otherwise, **echocheck** is initially set to **off**.

**eofread**
: (str) The set of characters which signify an end-of-transmission during a **˜<** file transfer command; abbreviated **eofr**. If the **ie** capability is present, **eofread** is initially set to the value of that capability; otherwise, **eofread** is set to a null string (**""**).

**eofwrite**
: (str) The string sent to indicate end-of-transmission during a **˜>** file transfer command; abbreviated **eofw**. If the **oe** capability is present, **eofread** is initially set to the value of that capability; otherwise, **eofread** is set to a null string (**""**).

**eol** (str) The set of characters which indicate an end-of-line. **tip** will recognize escape characters only after an end-of-line. If the **el** capability is present, **eol** is initially set to the value of that capability; otherwise, **eol** is set to a null string (**""**).

**escape** (char) The command prefix (escape) character; abbreviated **es**. If the **es** capability is present, **escape** is initially set to the value of that capability; otherwise, **escape** is set to '**~**'.

**etimeout**
(num) The amount of time, in seconds, that **tip** should wait for the echo-check response when **echocheck** is set; abbreviated **et**. If the **et** capability is present, **etimeout** is initially set to the value of that capability; otherwise, **etimeout** is set to 10 seconds.

**exceptions**
(str) The set of characters which should not be discarded due to the beautification switch; abbreviated **ex**. If the **ex** capability is present, **exceptions** is initially set to the value of that capability; otherwise, **exceptions** is set to '**\t\n\f\b**'.

**force** (char) The character used to force literal data transmission; abbreviated **fo**. If the **fo** capability is present, **force** is initially set to the value of that capability; otherwise, **force** is set to **\377** (which disables it).

**framesize**
(num) The amount of data (in bytes) to buffer between file system writes when receiving files; abbreviated **fr**. If the **fs** capability is present, **framesize** is initially set to the value of that capability; otherwise, **framesize** is set to 1024.

**halfduplex**
(bool) Do local echoing because the host is half-duplex; abbreviated **hdx**. If the **hd** capability is present, **halfduplex** is initially set to **on**; otherwise, **halfduplex** is initially set to **off**.

**hardwareflow**
(bool) Do hardware flow control; abbreviated **hf**. If the **hf** capability is present, **hardwareflow** is initially set to **on**; otherwise, **hardwareflowcontrol** is initially set to **off**.

**host** (str) The name of the host to which you are connected; abbreviated **ho**. **host** is permanently set to the name given on the command line or in the HOST environment variable.

**localecho**
(bool) A synonym for **halfduplex**; abbreviated **le**.

**log** (str) The name of the file to which to log information about outgoing phone calls. **log** is initially set to **/var/adm/aculog**, and can only be inspected or changed by the super-user.

**parity** (str) The parity to be generated and checked when talking to the remote host; abbreviated **par**. The possible values are:

- **none**
- **zero** Parity is not checked on input, and the parity bit is set to zero on output.
- **one** Parity is not checked on input, and the parity bit is set to one on output.
- **even** Even parity is checked for on input and generated on output.
- **odd** Odd parity is checked for on input and generated on output.

If the **pa** capability is present, **parity** is initially set to the value of that capability; otherwise, **parity** is set to **none**.

**phones**
The file in which to find hidden phone numbers. If the environment variable PHONES is set, **phones** is set to the value of PHONES; otherwise, **phones** is set to **/etc/phones**. The value of **phones** cannot be changed from within **tip**.

**prompt**
(char) The character which indicates an end-of-line on the remote host; abbreviated **pr**. This value is used to synchronize during data transfers. The count of lines transferred during a file transfer command is based on receipt of this character. If the **pr** capability is present, **prompt** is initially set to the value of that capability; otherwise, **prompt** is set to **\n**.

**raise** (bool) Upper case mapping mode; abbreviated **ra**. When this mode is enabled, all lower case letters will be mapped to upper case by **tip** for transmission to the remote machine. If the **ra** capability is present, **raise** is initially set to **on**; otherwise, **raise** is initially set to **off**.

**raisechar**
(char) The input character used to toggle upper case mapping mode; abbreviated **rc**. If the **rc** capability is present, **raisechar** is initially set to the value of that capability; otherwise, **raisechar** is set to **\377** (which disables it).

**rawftp** (bool) Send all characters during file transfers; do not filter non-printable characters, and do not do translations like **\n** to **\r**. Abbreviated **raw**. If the **rw** capability is present, **rawftp** is initially set to **on**; otherwise, **rawftp** is initially set to **off**.

**record** (str) The name of the file in which a session script is recorded; abbreviated **rec**. If the **re** capability is present, **record** is initially set to the value of that capability; otherwise, **record** is set to **tip.record**.

**remote** The file in which to find descriptions of remote systems. If the environment variable REMOTE is set, **remote** is set to the value of REMOTE; otherwise, **remote** is set to **/etc/remote**. The value of **remote** cannot be changed from within **tip**.

**script** (bool) Session scripting mode; abbreviated **sc**. When **script** is **on**, **tip** will record everything transmitted by the remote machine in the script record file specified in **record**. If the **beautify** switch is on, only printable ASCII characters will be included in the script file (those characters between 040 and 0177). The variable **exceptions** is used to indicate characters which are an exception to the normal beautification rules. If the **sc** capability is present, **script** is initially set to **on**; otherwise, **script** is initially set to **off**.

**tabexpand**
(bool) Expand TAB characters to SPACE characters during file transfers; abbreviated **tab**. When **tabexpand** is **on**, each tab is expanded to 8 SPACE characters. If the **tb** capability is present, **tabexpand** is initially set to **on**; otherwise, **tabexpand** is initially set to **off**.

**tandem**
(bool) Use XON/XOFF flow control to limit the rate that data is sent by the remote host; abbreviated **ta**. If the **nt** capability is present, **tandem** is initially set to **off**; otherwise, **tandem** is initially set to **on**.

**verbose**
(bool) Verbose mode; abbreviated **verb**; When verbose mode is enabled, **tip** prints messages while dialing, shows the current number of lines transferred during a file transfer operations, and more. If the **nv** capability is present, **verbose** is initially set to **off**; otherwise, **verbose** is initially set to **on**.

**SHELL** (str) The name of the shell to use for the **~!** command; default value is **/bin/sh**, or taken from the environment.

**HOME** (str) The home directory to use for the **~c** command; default value is taken from the environment.

## EXAMPLES

An example of the dialogue used to transfer files is given below.

**arpa% tip monet**
**[connected]**
*. . .(assume we are talking to a UNIX system). . .*
**ucbmonet login: sam**
**Password:**
**monet% cat > sylvester.c**
**~> Filename: sylvester.c**
**32 lines transferred in 1 minute 3 seconds**
**monet%**
**monet% ~< Filename: reply.c**
**List command for remote host: cat reply.c**
**65 lines transferred in 2 minutes**
**monet%**
*. . .(or, equivalently). . .*
**monet% ~p sylvester.c**
*. . .(actually echoes as ~[put] sylvester.c). . .*
**32 lines transferred in 1 minute 3 seconds**
**monet%**
**monet% ~t reply.c**
*. . .(actually echoes as ~[take] reply.c). . .*

**65 lines transferred in 2 minutes**
**monet%**
. . .*(to print a file locally)*. . .
**monet% ˜|Local command: pr –h sylvester.c | lpr**
**List command for remote host: cat sylvester.c**
**monet% ˜^D**
**[EOT]**
. . .*(back on the local system)*. . .

## ENVIRONMENT

The following environment variables are read by **tip**.

**REMOTE** The location of the *remote* file.

**PHONES** The location of the file containing private phone numbers.

**HOST** A default host to connect to.

**HOME** One's log-in directory (for chdirs).

**SHELL** The shell to fork on a '**˜!**' escape.

## FILES

| | |
|---|---|
| **/etc/phones** | |
| **/etc/remote** | |
| **/var/spool/locks/LCK ..*** | lock file to avoid conflicts with UUCP |
| **/var/adm/aculog** | file in which outgoing calls are logged |
| **˜/.tiprc** | initialization file |

## SEE ALSO

**cu**(1C), **mail**(1), **uucp**(1C), **vi**(1), **ioctl**(2)

## BUGS

There are two additional variables **chardelay** and **linedelay** that are currently not implemented.