NAME

callback – call a user back, presenting a login prompt

SYNOPSIS

callback [**-x**<debug**level>**] [**-V**] [**-l**<modem**lines>**] [**-m**<initstring>**] [**-s**<speed>**] [**-d**] [**-S**] [*phone-number*]

DESCRIPTION

Call the given phone number (if none is given on the command line, ask user for one), and if a CON-NECT is established, hand over control to mgetty(8) to present user with a login name prompt.

callback is used for various purposes:

* security: make sure your users are who they pretend to be by calling a well-known phone number.

* cost savings: make your company call you back.

callback can be called directly from the command line (but you must be "root" to do this, otherwise callback can't signal mgetty), or from mgetty's "login.config". See the login.config file shipped with mgetty for an example.

OPTIONS

-x <debug level>
> Use the given level of verbosity for logging - 0 means no logging, 5 is really noisy.

-V Print version number and quit.

-d Do not go into the background. This is helpful for debugging.

-l <modem lines>
> Use the given modem lines. Multiple lines can be separated by ":", as with send-fax(8). Example: callback -l tty1a:tty2a

-m <init sequence>
> Set the modem initialization sequence (as usual: expect send expect ...). This can do nearly everything, as long as it leaves the modem command responses on (that is, no ATQ1 here!) and switches the modem to data mode (AT+FCLASS=0) if it is used in data/fax mode.

-s <speed>
> This is the bit rate that should be used for the machine-modem connection. Usually you'll set this via the "speed <nnnn>" option in "callback.config".

-S Use the line where callback is started from for dialing out. Callback can make use of multiple modem lines, and with this options, you can force it to use just one modem, the one where a call comes in.

CONFIG FILE

callback will read all its configuration at run-time from a file, usually called /etc/mgetty+send-fax/callback.config. See the documentation in the mgetty.info manual for details.

DIAGNOSTICS

In most cases, *callback* can't print any error messages to the console, because it must detach itself immediately from the terminal, in case someone wants to be called back on the modem line he called in. So, nothing to print messages to...

Because of this, all callback errors are logged to a protocol file (the extent of the data written is controlled by the "-x" option), especially including the reason why a call was not made, or what exactly failed.

Just two messages are printed on stdout, and those are self-explaining, a call from a non-root user, and an invalid option.

INTERNALS

How does it work?

This is a bit tricky, because of the way init(8) handles the utmp(5) file. You can't just have any program ask the user for a login name, and then start a "login shell", it won't work (this is for the same reason *mgetty(8)* has to be started from /etc/inittab).

So, mgetty has to do the "asking for login name". But I do not want to have all that dialout code in mgetty, bloating it even more.

The way it works is this: *callback* dials out on a modem device. It will only take a modem device that has a mgetty watching over it (!). When the connection is established (CONNECT), *callback* will send a signal SIGUSR1 to mgetty, which, in turn, will send the same signal back to signal "I got your signal". *callback* then exits, and *mgetty* takes over the existing connection, prompts the user for a login name, and forks off /bin/login.

Conclusion: this will not work with mgetty versions before February 04, 1996 (no support for this signalling), and if it doesn't work for you, please send me **BOTH** the mgetty and the callback log file, otherwise it's very hard to find the bugs.

BUGS

callback is "alpha" code, not very stable right now.

callback is fairly dumb concerning retries.

callback must be run as root.

Most of the documentation consists of "reading the source".

SEE ALSO

mgetty(8), ct(1)

AUTHOR

callback is Copyright (C) 1993-1996 by Gert Doering, <gert AT greenie DOT muc DOT de>.