

**NAME**

efax – send/receive faxes with Class 1, 2 or 2.0 fax modem

(Please read the **fax** man page first.)

**SYNOPSIS**

**efax** [ *options* ] [ **-t num** [ *file...* ] ]

**OPTIONS**

Where *options* are:

- a cmd** use the command **ATcmd** when answering the phone. The default is "A".
- c caps** set the local modem capabilities. See the section on capabilities below for the format and meaning of *caps*. For Class 1 the default is 1,n,0,2,0,0,0,0 where n is the highest speed supported by the modem. For Class 2 the default is determined by the modem.
- d dev** use the fax modem connected to device *dev*. The default is **/dev/modem**.
- f fnt** use font file *fnt* for generating the header. The default is a built-in 8x16 font. See the `efix(1) -f` option for the font file format.
- g cmd** if a **CONNECT** (or **DATA**) response indicates a data call, the shell **/bin/sh** is `exec(2)`'ed with *cmd* as its command. *cmd* is a `printf(3)` format that may contain up to 6 %d escapes which are replaced by the baud rate following the most recent **CONNECT** message. *cmd* typically `exec's` `getty(8)`.
- h hdr** put string 'hdr' at the top of each page. The first %d in 'hdr' is replaced by the page number and the second, if any, is replaced by the number of pages being sent.
- i str**
- j str**
- k str** send the command **ATstr** to the modem to initialize it. -i commands are sent before the modem is put into fax mode, -j commands after the modem is in fax mode, and -k commands just before `efax` exits. The only default is a hang-up (ATH) command that is sent before exiting only if no other -k options are given. Multiple options may be used.
- l id** set the local identification string to *id*. *id* should be the local telephone number in international format (for example "+1 800 555 1212"). This is passed to the remote fax machine. Some fax machines may not accept characters other than numbers, space, and '+'.
- o opt** use option *opt* to accommodate a non-standard fax modem protocol. See the **MODEM REQUIREMENTS** section below for more details. The *options* are:
  - 0** Force use of Class 2.0 fax modem commands. The modem must support Class 2.0.
  - 2** Force use of Class 2 fax modem commands. The modem must support Class 2.

- 1** Force use of Class 1 fax modem commands. The modem must support Class 1. By default efax queries the modem and uses the first of the three above classes which is supported by the modem.
- a** use software adaptive answer method. If the first attempt to answer the call does not result in a data connection within 8 seconds the phone is hung up temporarily and answered again in fax mode (see "Accepting both fax and data calls" below).
- e** ignore errors in modem initialization commands.
- f** use "virtual flow control". efax tries to estimate the number of bytes in the modem's transmit buffer and pauses as necessary to avoid filling it. The modem's buffer is assumed to hold at least 96 bytes. This feature does not work properly with Class 2 modems that add redundant padding to scan lines. Use this option only if you have problems configuring flow control.
- h** use hardware (RTS/CTS) in addition to software (XON/XOFF) flow control. Many modems will stop responding if this option is used. See the section 'Resolving Problems' before using this option.
- l** halve the time between testing lock files when waiting for other programs to complete. By default this is 8 seconds. For example -olll sets the interval to 1 second.
- n** ignore requests for pages to be retransmitted. Use this option if you don't care about the quality of the received fax or if the receiving machine is too fussy. Otherwise each page may be retransmitted up to 3 times.
- r** do not reverse bit order during data reception for Class 2 modems. Only Multitech modems require this option. Not normally required since efax detects these modems.
- x** send XON (DC1) instead of DC2 to start data reception. Applies to a very few Class 2 modems only.
- z** delay an additional 100 milliseconds before each modem initialization or reset command. The initial delay is 100 ms. For example, -ozzz produces a 400 ms delay. Use with modems that get confused when commands arrive too quickly.
- q n** ask for retransmission of pages received with more than *n* errors. Default is 10.
- r pat** each received fax page is stored in a separate file. The file name is created using *pat* as a strftime(3) format string. A page number of the form .001, .002, ... is appended to the file name. If *pat* is blank ("") or no -r option is given a default string of "%m%d%H%M%S" is used.
- s** remove lock file(s) after initializing the modem. This allows outgoing calls to proceed when efax is waiting for an incoming call. If efax detects modem activity it will attempt to re-lock the device. If the modem has been locked by the other program efax will exit and return 1 ("busy"). Normally a new efax process is then started by init(8). The new efax process will then check periodically until the lock file disappears and then re-initialize the modem.

- t** *num [file...]* dial telephone number *num* and send the fax image files *file...*. If used, this must be the last argument on the command line. The telephone number *num* is a string that may contain any dial modifiers that the modem supports such as a T prefix for tone dialing or commas for delays. If no file names are given the remote fax machine will be polled. If no -t argument is given efax will answer the phone and attempt to receive a fax.
- v** *strng* select types of messages to be printed. Each *lower-case* letter in *strng* enables one type of message:
- e** - errors
  - w** - warnings
  - i** - session progress information
  - n** - capability negotiation information
  - c** - modem (AT) commands and responses
  - h** - HDLC frame data (Class 1 only)
  - m** - modem output
  - a** - program arguments
  - r** - reception error details
  - t** - transmission details
  - f** - image file details
  - x** - lock file processing
- Up to two -v options may be used. The first is for messages printed to the standard error and the second is for messages to the standard output. The default is "ewin" to the standard error only.
- w** wait for an OK or CONNECT prompt instead of issuing an answer (**ATA**) command to receive a fax. Use this option when the modem is set to auto-answer (using *S0=n*) or if another program has already answered the call.
- x** *lkf* use a UUCP-style lock file *lkf* to lock the modem device before opening it. If the device is locked, efax checks every 15 seconds until it is free. Up to 16 -x options may be used if there are several names for the same device. A '#' prefix on the file name creates a binary rather than text (HDB-style) lock file. This is the reverse of what was used by previous efax versions.

## FAX FILE FORMATS

efax can read the same types of files as **efix(1)** including text, T.4 (Group 3), PBM, single- and multi-page TIFF (G3 and uncompressed). efax automatically determines the type of file from its contents. TIFF files are recommended as they contain information about the image size and resolution.

Each page to be sent should be converted to a separate TIFF format file with Group 3 (G3) compression. Received files are also stored in this format. The EXAMPLES section below shows how efix and other programs can be used to create, view and print these files.

## OPERATING SYSTEM REQUIREMENTS

The operating system must provide short response times to avoid protocol timeouts. For Class 2 and 2.0 modems the delay should not exceed 1 or 2 seconds.

When using Class 1 modems the program must respond to certain events within 55 milliseconds. Longer delays may cause the fax protocol to fail in certain places (between DCS and TCF or between RTC and MPS). Class 1 modems should therefore not be used on systems that cannot guarantee that the program will respond to incoming data in less than 55 milliseconds. In particular, some intelligent serial cards and

terminal servers may introduce enough delay to cause problems with Class 1 operation.

The operating system must also provide sufficient low-level buffering to allow uninterrupted transfer of data between the modem and a disk file at the selected baud rate, typically 9600 bps. Since the fax protocol does not provide end-to-end flow control the effectiveness of flow control while receiving is limited by the size of the modem's buffer. This can be less than 100 bytes. Efax does not use flow control during reception.

## MODEM REQUIREMENTS

The "Group" is the protocol used to send faxes between fax machines. Efax supports the Group 3 protocol used over the public telephone network.

The "Class" is the protocol used by computers to control fax modems. Efax supports Class 1, 2 and 2.0 fax modems.

Most fax modems use XON/XOFF flow control when in fax mode. This type of flow control adds very little overhead for fax use. Many modems have unreliable hardware (RTS/CTS) flow control in fax mode. By default efax enables only XON/XOFF flow control and the `-oh` option must be used to add hardware flow control.

While some modems have serial buffers of about 1k bytes, many inexpensive modems have buffers of about one hundred bytes and are thus more likely to suffer overruns when sending faxes.

A few older modems may need a delay between commands of more than the default value used by efax (100 milliseconds). If the delay is too short, commands may not echo properly, may time out, or may give inconsistent responses. Use one or more `-oz` options to increase the delay between modem initialization commands and use the E0 modem initialization command to disable echoing of modem commands.

By default efax sends DC2 to start the data flow from the modem when receiving faxes from Class 2 modems. A few older modems require XON instead. Use of DC2 would cause the modem to give an error message and/or the program to time out. The `-ox` option should be used in this case.

A few older Class 2 modems (e.g. some Intel models) don't send DC2 or XON to start the data flow to the modem when sending faxes. After waiting 2 seconds efax will print a warning and start sending anyways.

A very few Class 2 modems do not reverse the bit order (MSB to LSB) by default on receive. This might cause errors when trying to display or print the received files. The `-or` option can be used in this case.

Some inexpensive "9600 bps" fax modems only *transmit* at 9600 bps and reception is limited to 4800 bps.

The following Class 1 modems have been reported to work with efax: AT&T DataPort, Cardinal Digital Fax Modem (14400), Digicom Scout+, Motorola Lifestyle 28.8, Motorola Power 28.8, QuickComm Spirit II, Smartlink 9614AV-Modem, Supra Faxmodem 144LC, USR Courier V.32bis Terbo, USR Sportster (V.32 and V.34), Zoom AFC 2.400, Zoom VFX14.4V.

The following Class 2 modems have been reported to work with efax: 14k4 Amigo Communion fax/modem, Adtech Micro Systems 14.4 Fax/modem, askey modem type 1414VQE, AT&T DataPort, ATT/Paradyne, AT&T Paradyne PCMCIA, Boca modem, BOCA M1440E, Crosslink 9614FH faxmodem, FuryCard DNE 5005, GVC 14.4k internal, Intel 14.4 fax modem, Megahertz 14.4, , Microcom DeskPorte FAST ES 28.8, Motorola UDS FasTalk II, MultiTech 1432MU, Practical Peripherals PM14400FXMT, Supra V32bis, Telebit Worldblazer, TKR DM-24VF+, Twincom 144/DFi, ViVa 14.4/Fax modem, Vobis Fax-Modem (BZT-approved), Zoom VFX14.4V, ZyXEL U-1496E[+], ZyXEL Elite 2864I.

## MODEM INITIALIZATION OPTIONS

The required modem initialization commands are generated by `efax`. Additional commands may be supplied as command-line arguments. The modem must be set up to issue `verbose(text)` result codes. The following command does this and is sent by `efax` before trying to initialize the modem.

**Q0V1** respond to commands with verbose result codes

The following commands may be useful for special purposes:

**X3** don't wait for dial tone before dialing. This may be used to send a fax when the call has already been dialed manually. In this case use an empty string ("") as the first argument to the `-t` command. Use **X4** (usual default) to enable all result codes.

**M2** leave the monitor speaker turned on for the duration of the call (use **M0** to leave it off).

**L0** turn monitor speaker volume to minimum (use **L3** for maximum).

**E0** disable echoing of modem commands. See the Resolving Problems section below.

**&D2** returns the modem to command mode when DTR is dropped. The program drops DTR at the start and end of the call if it can't get a response to a modem command. You can use **&D3** to reset the modem when DTR is dropped.

**S7=120** wait up to two minutes (120 seconds) for carrier. This may be useful if the answering fax machine takes a long time to start the handshaking operation (e.g. a combined fax/answering machine with a long announcement).

## CAPABILITIES

The capabilities of the local hardware and software can be set using a string of 8 digits separated by commas:

*vr,br,wd,ln,df,ec,bf,st*

where:

*vr* (vertical resolution) =  
 0 for 98 lines per inch  
 1 for 196 lpi

*br* (bit rate) =  
 0 for 2400 bps  
 1 for 4800  
 2 for 7200  
 3 for 9600  
 4 for 12000 (V.17)  
 5 for 14400 (V.17)

*wd* (width) =  
 0 for 8.5" (21.5 cm) page width  
 1 for 10" (25.5 cm)

2 for 12" (30.3 cm)

*ln* (length) =

0 for 11" (A4: 29.7 cm) page length

1 for 14" (B4: 36.4 cm)

2 for unlimited page length

*df* (data format) =

0 for 1-D coding

1 for 2-D coding (not supported)

*ec* (error correction) =

0 for no error correction

*bf* (binary fi le) =

0 for no binary fi le transfer

*st* (minimum scan time) =

0 for zero delay per line

1 for 5 ms per line

3 for 10 ms per line

5 for 20 ms per line

7 for 40 ms per line

When *receiving* a fax the *vr*, *wd*, and *ln* fi elds of the capability string should be set to the maximum values that your display software supports. The default is 196 lpi, standard (8.5"/21.5cm) width and unlimited length.

When *sending* a fax *efax* will determine *vr* and *ln* from the image fi le and set *wd* to the default.

If the receiving fax machine does not support high resolution (*vr*=1) mode, *efax* will reduce the resolution by combining pairs of scan lines. If the receiving fax machine does not support the image's width then *efax* will truncate or pad as required. Most fax machines can receive *ln* up to 2. Few machines support values of *wd* other than 0.

## HEADERS

*efax* adds blank scan lines at the top of each image when it is sent. This allows room for the page header but increases the length of the image (by default about 0.1" or 2.5mm of blank space is added).

The header placed in this area typically includes the date and time, identi fi es the, and shows the page number and total pages. Headers cannot be disabled but the header string can be set to a blank line.

The default font for generating the headers is the built-in 8x16 pixel font scaled to 12x24 pixels (about 9 point size).

Note that both *efax* and *efix* have *-f* options to specify the font. *efix* uses the font to generate text when doing text-to-fax conversions (during "fax make") while *efAx* uses the font to generate the header (during "fax send").

## SESSION LOG

A session log is written to the standard error stream. This log gives status and error messages from the program as selected by the **-v** option. A time stamp showing the full time or just minutes and seconds is printed before each message. Times printed along with modem responses also show milliseconds.

## RETURN VALUES

The program returns an error code as follows:

- |   |   |
|---|---|
| 0 | The fax was successfully sent or received.  |
| 1 | The dialed number was busy or the modem device was in use. Try again later.   |
| 2 | Something failed (e.g. file not found or disk full). Don't retry. Check the session log for more details.   |
| 3 | Modem protocol error. The program did not receive the expected response from the modem. The modem may not have been properly initialized, the correct <b>-o</b> options were not used, or a bug report may be in order. Check the session log for more details. |
| 4 | The modem is not responding. Operator attention is required. Check that the modem is turned on and connected to the correct port.   |
| 5 | The program was terminated by a signal.   |

## EXAMPLES

### Creating fax (G3) files

The `efix` program can be used to convert text files to TIFF-G3 format. For example, the following command will convert the text file **letter** to the files **letter.001**, **letter.002**, etc.:

```
efix -nletter.%03d letter
```

Ghostscript's **tiffg3** driver can generate fax files in TIFF-G3 format from postscript files. For example, the command:

```
gs -q -sDEVICE=tiffg3 -dNOPAUSE \  
-sOutputFile=letter.%03d letter.ps </dev/null
```

will convert the Postscript file **letter.ps** into high-resolution ( $vr=1$ ) G3 fax image files **letter.001**, **letter.002**, ...

The images should have margins of at least 1/2 inch (1 cm) since the fax standard only requires that fax machines print a central portion of the image 196.6mm (7.7 inches) wide by 281.5mm (11.1 inches) high.

The `efix` program can also insert bitmaps in images to create letterhead, signatures, etc.

### Printing fax files

You can use the `efix` program to print faxes on Postscript or HP-PCL (LaserJet) printers. For example, to print the received fax file **reply.001** on a Postscript printer use the command:

```
efix -ops reply.001 | lpr
```

### **Sending fax files**

The following command will dial the number 222-2222 using tone dialing and send a two-page fax from the TIFF-G3 files `letter.001` and `letter.002` using the fax modem connected to device `/dev/cua1`.

```
efax -d /dev/cua1 \  
-t T222-2222 letter.001 letter.002
```

### **Manual answer**

You can use `efax` to answer the phone immediately and start fax reception. Use this mode if you need to answer calls manually to see if they are fax or voice.

For example, the following command will make the fax modem on device `/dev/ttyS1` answer the phone and attempt to receive a fax. The received fax will be stored in the files `reply.001`, `reply.002`, and so on. The modem will identify itself as "555 1212" and receive faxes at high or low resolution (`vr=1`), at up to 14.4 kbps (`br=5`).

```
efax -d /dev/ttyS1 -l "555 1212" \  
-c 1,5 -r reply
```

### **Automatic answer**

The `-w` option makes `efax` wait for characters to become available from the modem (indicating an incoming call) before starting fax reception. Use the `-w` option and a `-iS0=n` option to answer the phone after  $n$  rings. The example below will make the modem answer incoming calls in fax mode on the fourth ring and save the received faxes using file names corresponding to the reception date and time.

```
efax -d /dev/ttyb -w -iS0=4 2>&1 >> fax.log
```

### **Sharing the modem with outgoing calls**

The modem device can be shared by programs that use the UUCP device locking protocol. This includes `pppd`, `chat`, `minicom`, `kermit`, `uucico`, `efax`, `cu`, and many others. However, locking will only work if all programs use the same lock file.

`efax` will lock the modem device before opening it if one or more UUCP lock file names are given with `-x` options. Most programs place their lock files in the `/usr/spool/uucp` or `/var/lock` directories and use the name `LCK..dev` where `dev` is the name of the device file in the `/dev` directory that is to be locked.

If the `-s` (share) option is used, the lock file is removed while waiting for incoming calls so other programs can use the same device.

If `efax` detects another program using the modem while it is waiting to receive a fax, `efax` exits with a termination code of 1. A subsequent `efax` process using this device will wait until the other program is finished before re-initializing the modem and starting to wait for incoming calls again.

Programs that try to lock the modem device by using device locking facilities other than UUCP lock files not be able to use this arbitration mechanism because the device will still be open to the `efax` process. In this case you will need to kill the `efax` process (e.g. "fax stop") before starting the other program.

When efax is waiting for a fax it leaves the modem ready to receive in fax mode but removes the lock file. When a slip or PPP program takes over the modem port by setting up its own lock file efax cannot send any more commands to the modem -- not even to reset it. Therefore the other program has to set the modem back to data mode when it starts up. To do this add a modem reset command (send ATZ expect OK) to the beginning of your slip or PPP chat script.

### Accepting both fax and data calls

Many modems have an adaptive data/fax answer mode that can be enabled using the **-j+FAE=1** (for Class 1) or **-jFAA=1** (for Class 2[.0]) initialization string. The type of call (data or fax) can then be deduced from the modem's responses.

Some modems have limited adaptive answer features (e.g. only working properly at certain baud rates or only in Class 2) or none at all. In this case use the initialization string **-i+FCLASS=0** to answer in data mode first and the **-oa** option to then hang up and try again in fax mode if the first answer attempt was not successful. This method only works if your telephone system waits a few seconds after you hang up before disconnecting incoming calls.

If the **-g** option is used then the option's argument will be run as a shell command when an incoming data call is detected. Typically this command will exec **getty(8)**. This program should expect to find the modem already off-hook and a lock file present so it should not try to hang up the line or create a lock file. Note that the modem should be set up to report the DCE-DTE (modem-computer, e.g. CONNECT 38400) speed, not the DCE-DCE (modem-modem, e.g. CONNECT 14400) speed. For many modems the initialization option **-iW0** will set this.

The following command will make efax answer incoming calls on **/dev/cua1** on the second ring. This device will be locked using two different lock files but these lock files will be removed while waiting for incoming calls (**-s**). If a data call is detected, the **getty** program will be run to initialize the terminal driver and start a **login(1)** process. Received fax files will be stored using names like **Dec02-12.32.33.001**, in the **/usr/spool/fax/incoming** directory and the log file will be appended to **/usr/spool/fax/faxlog.cua1**.

```
efax -d /dev/cua1 -j '+FAA=1' \
-x /usr/spool/uucp/LCK..cua1 \
-x /usr/spool/uucp/LCK..ttyS1 \
-g "exec /sbin/getty -h /dev/cua1 %d" \
-iS0=2 -w -s \
-r "/usr/spool/fax/incoming/%b%d-%H.%I.%S" \
>> /usr/spool/fax/faxlog.cua1 2>&1
```

Note that adaptive answer of either type will not work for all callers. For some data calls the duration of the initial data-mode answer may be too short for data handshaking to complete. In other cases this duration may be so long that incoming fax calls will time out before efax switches to fax mode. In addition, some calling fax modems mistake data-mode answering tones for fax signaling tones and initiate fax negotiation too soon. If you use software adaptive answer you can reduce the value of the initial data-mode answer (set by **TO\_DATAF** in **efax.c**) to get more reliable fax handshaking or increase it for more reliable data handshaking. However, if you need to provide reliable fax and data service to all callers you should use separate phone numbers for the two types of calls.

When a call is answered the modem goes on-line with the computer-to-modem baud rate fixed at the speed used for the most recent AT command. When efax is waiting for a fax or data call it sets the interface speed to 19200 bps since this is the speed required for fax operation. This prevents full use of 28.8kbps modem capabilities.

## USING INIT TO RUN EFAX

efax can answer all incoming calls if you place an entry for efax in **/etc/inittab** (for SysV-like systems) or **/etc/ttytab** (for BSD-like systems). The **init(8)** process will run a new copy of efax when the system boots up and whenever the previous efax process terminates. The inittab or ttytab entry should invoke efax by running the **fax** script with an **answer** argument.

For example, placing the following line in **/etc/inittab** (and running "kill -1 1") will make init run the **fax** script with the argument **answer** every time previous process terminates and **init** is in runlevel 4 or 5.

```
s1:45:respawn:/bin/sh /usr/bin/fax answer
```

For BSD-like systems (e.g. SunOS), a line such as the following in **/etc/ttytab** will have the same effect:

```
ttya "/usr/local/bin/fax answer" unknown on
```

You should protect the fax script and configuration files against tampering since init will execute them as a privileged (root) process. If you will be allowing data calls via getty and login you should ensure that your system is reasonably secure (e.g. that all user id's have secure passwords).

If efax exec()'s getty properly but you get a garbled login prompt then there is probably a baud rate mismatch between the modem and the computer. First, check the efax log file to make sure the modem's CONNECT response reported the serial port speed (e.g. 19200), **not** the modem-modem speed (e.g. 14400). Next, check the getty options and/or configuration files (e.g. /etc/gettydefs) for that particular baud rate. Then run getty manually with the same arguments and verify the port settings using "stty </dev/XXX". Note that you'll probably want to enable hardware flow control for data connections (-h for agetty, CRTSCTS for getty\_ps).

A few programs won't work properly when efax is set up to answer calls because they don't create lock files. You can put the shell script "wrapper" below around such programs to make them work properly. Change BIN and LOCKF to suit.

```
#!/bin/sh
BIN=/bin/badprogram
LOCKF=/var/spool/uucp/LCK..cua1
if [ -f $LOCKF ]
then
    echo lockfile $LOCKF exists
    exit 1
else
    printf "%10d0 $$ >$LOCKF
    $BIN $*
    rm $LOCKF
fi
```

## DELIVERING RECEIVED FAXES BY E-MAIL

The "fax answer" script described above can be configured to e-mail the fax files received by the previous fax answer process to a "fax manager" who can then forward the fax to the correct recipient. The received fax files are sent as MIME attachments, one file per page, using the "base64" text encoding and the "image/tiff" file format.

To view the fax images directly from your e-mail reader you will have to configure it with an application

that can display files of type image/tiff. Typically this is specified in a "mailcap" file. For example, placing the following line in /etc/mailcap will cause the fax file attachments to be displayed using the "fax view" command.

```
image/tiff; fax view %s
```

## SENDING FAXES USING THE PRINT SPOOLER

You can configure a "fax" printer into the lpr print spooler that will fax a document out using efax instead of printing it. This allows a network server running efax to send faxes on behalf of other machines, including non-Unix clients. In the following steps use the directories specified in the fax script if they are different than /usr/bin and /var/spool/fax (FAXDIR). To set up a fax printer do the following as root:

(1) Create a link to the fax script called "faxlpr" so the fax script can determine when it is being invoked from the print spooler:

```
ln -s /usr/bin/fax /usr/bin/faxlpr
```

(2) Edit /etc/printcap and add an entry such as:

```
fax:lp=/dev/null:sd=/var/spool/fax:if=/usr/bin/faxlpr:
```

to define a printer called "fax". Print files will be spooled to the /var/spool/fax (sd=) directory and then piped to the /usr/bin/faxlpr filter (if=). Error messages will appear on /dev/console.

(3) Create and/or set the permissions to allow anyone to read and write in the fax spool directory. For example:

```
mkdir /var/spool/fax
chmod 777 /var/spool/fax
```

(4) Create a printer daemon lock file that is readable by anyone:

```
touch /var/spool/fax/lock
chmod 644 /var/spool/fax/lock
```

You should now be able to send a fax using the lpr interface by using a command such as:

```
lpr -P fax -J "555 1212" file.ps
```

where the -J option is used to specify the phone number or alias to be dialed.

Note that if more than one file is given on the command line they will be concatenated before being passed to "fax send". TIFF-G3, Postscript or PBM files must therefore be sent one file at a time although TIFF and Postscript files may contain multiple pages. Only multiple *text* files can be sent in one command. Page breaks in text files can be marked with form-feed characters. Files will be converted and sent at the default (high) resolution.

You can use lpq(1) to check the fax queue, lprm(1) to remove fax jobs and lpc(8) to control the spooler. In each case use the -Pfax option to specify the fax "printer." A log file will be mailed to the user when the fax is sent.

You should also be able to send a fax from any networked computer that has lpr-compatible remote printing software and that allows you to set the job name (-J option) to an arbitrary string. Such software is available for most computers.

See the lpd(8) and printcap(5) man pages for information on the print spooler and for restricting access by host name (/etc/host.lpd) or by user group (the 'rg' printcap entry).

## RESOLVING PROBLEMS

Double check the configuration setup in the first part of the fax script, particularly the modem device name and the lock file names.

If efax hangs when trying to open the modem device (typically /dev/ttyX), the device is either already in use by another process (e.g. pppd) or it requires the carrier detect line to be true before it can be opened. Many systems define an alternate device name for the same physical device (typically cuaX) that can be opened even if carrier is not present or other programs are already using it.

If responses to modem initialization commands are being lost or generated at random, another processes (e.g. getty or an efax auto-answer process) may be trying to use the modem at the same time. Try running efax while this other program is running. If efax does not report "/dev/ttyX locked or busy. waiting." then the lock file names are not specified correctly.

Attempt to send a fax. Check that the modem starts making the calling signal (CNG, a 0.5 second beep every 3 seconds) as soon as it's finished dialing. This shows the modem is in fax mode. You may need to set the SPKR variable to -iM2L3 to monitor the phone line to do this.

Listen for the answering fax machine and check that it sends the answer signal (CED, a 3 second beep) followed by "warbling" sounds (DIS frames) every 3 seconds. If you hear a continuous sound (tones or noise) instead, then you've connected to a data modem instead.

Your modem should send back its own warble (DCS frame) in response to DIS immediately followed by 1.5 seconds of noise (a channel check). If everything is OK, the receiving end will send another warble (CFR frame) and your modem will start to send data. If you have an external modem, check its LEDs. If flow control is working properly the modem's send data (SD) LED will turn off periodically while the fax data is sent.

Check the message showing the line count and the average bit rate when the page transmission is done. Low line counts (under 1000 for a letter size image) or the warning "fax output buffer overflow" while sending indicate that the image data format is incorrect. Check the file being sent using the "fax view" command.

If you get the error message "fbw control did not work" then fbw control was not active. This usually results in a garbled transmission and the receiving machine may reject the page, abort the call, print a distorted or blank image and/or hang up.

The warning "characters received while sending" or an <XOFF> character appearing after the transmission means that the operating system ignored the modem's XOFF flow control character. Ensure that you are not running other programs such as getty or pppd at the same time as efax since they will turn off xon/xoff flow control.

If you cannot get flow control to work properly then enable "virtual flow control" with the **-of** option or hardware flow control with the **-oh** option.

Check that the remote machine confirms reception with a +FPTS:1 response (Class 2) or an MCF frame (Class 1).

For Class 2 modems, the error message "abnormal call termination (code *m*)" indicates that the modem detected an error and hung up.

Many companies advertise services that will fax back information on their products. These can be useful for testing fax reception.

The message "run length buffer overfbw" when receiving indicates an error with the image data format. You may need to use the **-or** option with certain Class 2 modems.

If efax displays the message "can't happen (<details>)" please send a bug report to the author.

Finally, don't play "option bingo," if you can't resolve the problem send a verbose log of the failed session (the output from **fax -v ...**) to the address below.

## WEB PAGE

A Web Page with pointers to the latest version, known bugs and patches is available at:

<http://www.cce.com/efax/>

## RELATED SOFTWARE

For Linux Systems

Independent packages provide more user-friendly interfaces to efax (xfax, tefax) and provide an e-mail-to-fax (Qfax) gateway using efax. All are available by anonymous FTP from metalab.unc.edu in /pub/Linux/apps/serialcomm/fax/.

For Amiga Systems

A port of an early version of efax for the Amiga is available as a component of a shareware voice mail package, AVM, distributed by Al Villarica (rvillari@cat.syr.edu).

Other Ports

efax is relatively easy to port. All system-dependent code is in **efaxos.c**. An early version of efax was ported to VMS. Version 0.8a was ported to Win32 by Luigi Capriotti. Contact the author if you would like to integrate the Win32 code into the current version.

## AUTHOR

Efax was written by Ed Casas. Please send comments or bug reports to edc@cce.com.

## BUG REPORTS

Bug reports should include the operating system, the type of the modem and a copy of a verbose session log that demonstrates the problem. It's usually impossible to help without a verbose log. Please do **not** send fax image files.

## COPYRIGHT

efax is copyright 1993 -- 1999 Ed Casas. It may be used, copied and modified under the terms of the GNU Public License.

## DISCLAIMER

Although **efax** has been tested it may have errors that will prevent it from working correctly on your system. Some of these errors may cause serious problems including loss of data and interruptions to telephone

service.

## REFERENCES

CCITT Recommendation T.30, "Procedures for Document Facsimile Transmission in the General Switched Telephone Network". 1988

CCITT Recommendation T.4, "Standardization of Group 3 Facsimile Apparatus for Document Transmission". 1988.

For documentation on Class 1 and Class 2 fax commands as implemented by Connexant (formerly Rockwell) modems see <http://www.conexant.com/techinfo>.

For the TIFF specification see <http://partners.adobe.com/supportservice/devrelations/PDFS/TN/TIFF6.pdf> or RFC 2301 (<ftp://ftp.isi.edu/in-notes/rfc2301.txt>).

For information on Ghostscript see <http://www.cs.wisc.edu/~ghost/>.

The pbm utilities can be obtained by ftp from [wuarchive.wustl.edu](ftp://wuarchive.wustl.edu) in `/graphics/graphics/packages/NetPBM/netpbm-1mar1994.tar.gz`.

PCX and many other file formats are described in: Gunter Born, *The File Formats Handbook*, International Thomson Computer Press, 1995.

The "Fax Modem Source Book" by Andrew Margolis, published by John Wiley & Sons in 1994 (ISBN 0471950726), is a book on writing fax applications which includes source code.

Dennis Bodson et. al., "FAX: Digital Facsimile Technology and Applications", Second Edition. Artech House, Boston. 1992.

## SEE ALSO

**fax(1), efax(1), gs(1), init(8), inittab(5), ttytab(5), printcap(5), lpd(8), printf(3), strftime(3).**

## BUGS

Can't read TIFF files with more than 1 strip

Class 1 operation may fail if the program can't respond to certain data received from the modem within 55 milliseconds.

May fail if multitasking delays cause the received data to overflow the computer's serial device buffer or if an under-run of transmit data exceeds 5 seconds.

Polling does not work.

Does not support 2-D coding, ECM, or BFT.