NAME

mailatt − send e−mail with attachments

SYNOPSIS

mailatt [*message options* │ *encoding options* │ *header options* │ *debug options*]
[*encoding options* │ −− │ *file1* [*file2...*]]

mailatt { −h │ −v }

Message options:
[−FIai]

Header options:
[−r *recipients*] [−c *cc-addresses*] [−b *bcc-addresses*] [−f *from*] [−R *reply-to-addresses*] [−s *subject*] [−H *headerline*]

Encoding options:
[−C *charset*] [−M *mime-type*] [−mqu]

Debug options:
[−DVd]

Please note that the order of the commandline options is significant.

DESCRIPTION

mailatt is used to send multipart MIME e−mail messages with file attachments.

The message can be composed in either of two possible formats:

− *multipart/mixed*

This is the default. It signifies that the message contains a collection of files which are not necessarily related.

One of the message parts may be displayed inline, as the inline message body; see the **−i** option below.

− *multipart/alternative*

This message type can be used to provide alternative representations of the same file (*e.g.* plain text and HTML), in which case the mail client can show the one it supports. See also the **−a** option below.

Files can be of any MIME type and can be attached using several encoding types:

− Files of type *text/** and those that are known to be (mostly) ASCII-only, like *e.g.* many script types, will be attached using quoted-printable encoding by default (but see also the **−m** and **−i** options below). Note however, that this is only done for readability, and that **mailatt**'s implementation of this encoding is binary-safe.

− Binary files (*i.e.* files having MIME types other than *text/**) will be attached using base64 encoding by default (but see the **−q** option below).

− Files of type *message/** or *multipart/** will always be attached using 8bit−encoding, because of requirements imposed by RFC 2045.

Is is usually not necessary to specify the encoding, as **mailatt** has got sensible defaults. Specifying the character set however, is often necessary if a text is not clean 7−bit ASCII (see **−C** below).

If the host system has an **md5** (1) or **md5sum** (1) program, a `Content-MD5` header will be added for all ascii and binary attachments to ensure data integrity to some extent. Attachments of types *message/** or *multipart/** will never have any `Content-MD5` header, since RFC 1864 disallows it. See also the **−D** option.

OPTIONS

−C *charset*

This option may be used multiple times, and in any position on the commandline. It specifies the character set to be used for all subsequent attachments and header lines. If unspecified, US-ASCII is used. See the EXAMPLES below.

Some often used character sets:

```
US-ASCII       ISO-8859-6     GB18030        Windows-1251
UTF-8          ISO-8859-7     GB2312         Windows-1252
UTF-16         ISO-8859-8     Shift_JIS      Windows-1253
ISO-8859-1     ISO-8859-9     KOI8-R         Windows-1254
ISO-8859-2     ISO-8859-15    KOI8-U         Windows-1255
ISO-8859-3     EUC-JP         Big5           Windows-1256
ISO-8859-4     EUC-KR         Windows-874    Windows-1257
ISO-8859-5     EUC-CN         Windows-1250   Windows-1258
```

−D Suppress sending a `Content-MD5` header line. This prevents mail clients from checking the file checksums, which may be useful in case problems are encountered.

−F Specifies that the first attachment must be attached with the **format=flowed** parameter. This signals to e−mail clients that the text may be flowed by paragraph and quotations may be rendered in a more legible way, e.g. with quote bars. This also prevents incorrect parsing of **>** characters at the start of lines.

This is probably only useful in combination with **−i** and a *text/plain* attachment.

Use this option for best legibility, and if preservation of the formatting is of less concern. Not using this option will preserve formatting, possibly at the expense of legibility.

See also **−i**.

−H *headerline*
Specifies a line to add to the mail header. This option can be used multiple times. Do not end the line with a newline character.

−I This option is nearly identical to **−i**, but uses the "old" behavior, which is to include a filename for all MIME parts, even those that are attached using `Content-Disposition: inline`.

Warning: not all e−mail clients support this. This option is provided only for backward-compatibility. The use of this option is therefore discouraged; use **−i** instead.

−M *mime-type*
Specifies the MIME type to be used for the next file on the commandline. If not specified, the MIME type is inferred from the filename extension.

−R *addr1*[,*addr2...*]
Specify Reply-To: addresses for the e−mail. See **−r** for allowed formats.

−V For debugging purposes only. The name of each file will be reported on *stderr* when it is attached.

−a Use *multipart/alternative* for the mail message instead of the default *multipart/mixed*. All message parts will automatically be included with `Content-Disposition: inline`, regardless of any **−i** option. Note that the simplest representation of the mail (probably *text/plain*) should be attached first.

−b *addr1*[,*addr2...*]
Specify Bcc: addresses for the e−mail. See **−r** for allowed formats.

−c *addr1*[,*addr2...*]
Specify Cc: addresses for the e−mail. See **−r** for allowed formats.

−d For debugging purposes only. The composed mail will be printed on *stdout* instead of actually being sent. This option is overruled by the MAILATT_DEBUG variable (see below).

−f *addr1*
Specify From: address for the e−mail.

−h Print help (usage information) and exit.

−i   Specifies that the first file on the commandline must be included with a `Content-Disposition:` `inline` header, and without a filename. If this option is not specified, all parts of the message will be included with `Content-Disposition: attachment` and a specified filename.

If the first attachment is of type *text/** and the characterset is US-ASCII, then it will be attached using 7bit−encoding.

Note that it is likely that e−mail clients will support this format only for files of type *text/**.

The option **−a** takes precedence over **−i**.

−m This option may be used multiple times, and in any position on the commandline. It indicates that **base64** should be used as Content-Transfer-Encoding for all subsequent attachments and header lines. This is the default if unspecified.

This option is called **−m** as a reference to **mmencode** (1). However, **mailatt** does not require any external **mmencode** (1) program; instead, it handles the encoding itself.

See also **−q**.

−q   This option may be used multiple times, and in any position on the commandline. It indicates that **quoted-printable** should be used as Content-Transfer-Encoding for all subsequent attachments and header lines.

Although quoted-printable is normally only used for ASCII attachments, **mailatt** also supports this encoding for binaries.

See also **−m**.

−r *addr1*[,*addr2...*]
     Specify recipients (To: addresses) for the e−mail.

Each one of the addresses may be specified in one of the following formats:

   user
   <user>
   <user@domain>
   name <user>
   name <user@domain>

The first two forms will be expanded (using the system *passwd* file or (if configured) the NIS *passwd* map) to the form "*name <user>*"‘, which the mailer system may further expand to "*name <user@domain>*".

The **−r** option is strictly speaking not mandatory for **mailatt**, but it is for programs like **sendmail** (1).

−s *subject*
     Specify subject for the e−mail.

−u   This option may be used multiple times, and in any position on the commandline. It indicates that **uuencode** should be used as Content-Transfer-Encoding for all subsequent attachments (header lines will be base64−encoded).

mailatt does not require any external **uuencode** (1) program; instead, it handles the encoding itself.

Warning: not all e−mail clients support this. The use of this option is therefore discouraged; use **−m** instead.

See also **−m** and **−q**.

−v   Display version information and exit.

−−   Indicates that no more options follow; all the remaining commandline arguments are to be interpreted as filenames.

file1 [*file2...*]

> Indicate which files should be attached to the mail. Depending on the "extension" (*i.e.* the filename part following a `.` in the filename), a corresponding MIME type will be selected from a built-in list.
>
> If the file type is not known or the extension is empty, the file will be sent as type *application/octet−stream*, unless the **−a** or **−i** option requires it to be included inline. In that case, the MIME type will be reported as *text/plain*.
>
> At most one of the filenames may be specified as a single − (minus), which will cause **mailatt** to read a document from stdin. If entered from the terminal, such a document should be ended with the *eof−*character (usually CTRL-D). **mailatt** will assume that this document is of type *text/plain*, unless indicated otherwise with **−M**.

EXAMPLES

Send two **.pdf* files to two recipients (local users):

```
mailatt −s 'Revised speeches' −r alexander,maxima
        throne-speech.pdf christmas-speech.pdf
```

Send a file fragment, carbon copy to the sysadmin:

```
mailatt −s'Fragment 2/10' −r admin −c bofh dumps.tar.gz.xab
```

Send a formatted file, preceded by an introductory message, typed interactively:

```
mailatt −s manual −r beatrix@drakensteyn.nl −i − sendmail.pdf
Good morning Your Royal Highness,
Attached you will find the sendmail manual I promised.
^D
```

Send a formatted mail in HTML format, specifying a From: address:

```
mailatt −f 'Pieter <pvvollenhoven@hotmail.com>' −r margriet
        −i birthdaycard.html
```

Send a message in both plain text and HTML format:

```
mailatt −a −r constantijn,friso −s 'Greetings!' msg.txt msg.html
```

Attach a file of unknown MIME type using quoted-printable encoding (*e.g.* to increase readability). This overrules the default encoding (base64) that is used for attachments of unknown MIME type coming *before* the **−q** option.

```
mailatt −r christina −s "config files" −i body.txt
        /etc/sendmail.cf −q /etc/hosts
```

Send a mail using the Cyrillic character set to multiple addresses:

```
mailatt −r 'Mr. Putin <putin@kremlin.ru>,root'
        −s 'Important letter' −i −C ISO−8859−5 letter.txt
```

Add custom lines to the mail header:

```
mailatt −r 'Benedictus_XVI@vatican.va' −s 'Oldest translation'
        −i −H "X-Year-Finished: 1637" message.txt SV.pdf
```

Add a Sender: line to the mail header:

```
mailatt −r all@company.com −s 'Important announcement' −i
        −H 'Sender: secretary@company.com' −f ceo@company.com
        message.html
```

Send a mail with mixed character sets:

```
mailatt −C ISO−8859−15 −r 'François <francois@nimporte.ou>'
        −s 'Allô' −C UTF−8 −i message.html −C windows−1253 m.txt
```

Send a mail with specified MIME types:

```
      mailatt −r amalia −i greeting.txt
              −M application/x-rpm mailcap-2.1.31.noarch.rpm
              −M audio/x-pn-realaudio recording.rpm
```

Specify the MIME type of an attachment fed via stdin:

```
   mailatt −r alexia −s 'Holiday pictures' −i −C ISO-Latin-15
           −M text/html − skiing-1.jpg skiing-2.jpg < greeting.html
```

ENVIRONMENT

MAILATT_DEBUG

For debugging purposes only. If set and not **0**, this causes the composed mail to be printed on *stdout* instead of actually being sent. This behavior is identical to adding the **−d** option.

Setting this variable to **0** disables the debugging behavior. This takes precedence over adding **−d**.

BUGS and WARNINGS

This script aims to run out of the box in **ksh** (1) and **bash** (1) on a wide variety of Un*x−like platforms (such as AIX, HP-UX, SunOS, Solaris, Darwin, FreeBSD and Linux (SuSE, Debian, CentOS)), but has not yet been tested on all of them.

RFC 5322 warns that lines of characters in the body MUST be limited to 998 characters. If a file of type *message/** or *multipart/** is attached, this is not enforced, since the file is attached as-is.

SEE ALSO

mail (1), **mailx** (1), **md5** (1), **md5sum** (1), **sendmail** (8).

RFC 822:   Format of Internet Text Messages

RFC 1049:   The Content-Type Header Field for Internet Messages

RFC 1864:   The Content−MD5 Header Field

RFC 2045:   MIME: Format of Internet Message Bodies

RFC 2046:   MIME: Media Types

RFC 2047:   MIME: Message Header Extensions for Non-ASCII Text

RFC 2183:   The Content-Disposition Header Field

RFC 2387:   The multipart/related Content-Type

RFC 2392:   Content-ID and Message-ID URLs

RFC 4648:   The Base16, Base32, and Base64 Data Encodings

RFC 5322:   Internet Message Format

VERSION

This manual pertains to **mailatt** version 1.27.4.

AUTHOR and COPYRIGHT

Copyright © 2011-2022, René Uittenbogaard (ruittenb@users.sourceforge.net) This program is free software; you can redistribute it and/or modify it under the terms described by the GNU General Public License version 3.

mailatt is distributed without any warranty, even without the implied warranties of merchantability or fitness for a particular purpose.