

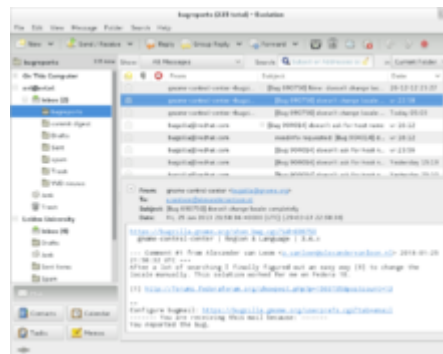
## WIKIPEDIA

# Email

**Electronic mail** (**email** or **e-mail**) is a method of exchanging messages ("mail") between people using electronic devices. Email first entered limited use in the 1960s and by the mid-1970s had taken the form now recognized as email. Email operates across computer networks, which today is primarily the Internet. Some early email systems required the author and the recipient to both be online at the same time, in common with instant messaging. Today's email systems are based on a store-and-forward model. Email servers accept, forward, deliver, and store messages. Neither the users nor their computers are required to be online simultaneously; they need to connect only briefly, typically to a mail server or a webmail interface, for as long as it takes to send or receive messages.

Originally an ASCII text-only communications medium, Internet email was extended by Multipurpose Internet Mail Extensions (MIME) to carry text in other character sets and multimedia content attachments. International email, with internationalized email addresses using UTF-8, has been standardized, but as of 2017 it has not been widely adopted.<sup>[2]</sup>

The history of modern Internet email services reaches back to the early ARPANET, with standards for encoding email messages published as early as 1973 (RFC 561). An email message sent in the early 1970s looks very similar to a basic email sent today. Email had an important role in creating the Internet,<sup>[3]</sup> and the conversion from ARPANET to the Internet in the early 1980s produced the core of the current services.



This screenshot shows the "Inbox" page of an email client, where users can see new emails and take actions, such as reading, deleting, saving, or responding to these messages



The at sign, a part of every SMTP email address<sup>[1]</sup>

## Contents

### Terminology

### Origin

### Operation

### Message format

- Message header

- Header fields

- Message body

- Content encoding

- Plain text and HTML

### Servers and client applications

- Filename extensions

- URI scheme mailto

### Types

Google 1017

POP3 email services  
IMAP email servers  
MAPI email servers

### Uses

Business and organizational use  
  Email marketing  
Personal use  
  Personal computer  
  Mobile  
  Declining use among young people

### Issues

Attachment size limitation  
Information overload  
Spam  
Malware  
Email spoofing  
Email bombing  
Privacy concerns  
Flaming  
Email bankruptcy  
Internationalization  
Tracking of sent mail

### See also

### References

### Further reading

### External links

## Terminology

---

Historically, the term *electronic mail* was used generically for any electronic document transmission. For example, several writers in the early 1970s used the term to describe fax document transmission.<sup>[4][5]</sup> As a result, it is difficult to find the first citation for the use of the term with the more specific meaning it has today.

Electronic mail has been most commonly called *email* or *e-mail* since around 1993,<sup>[6]</sup> but variations of the spelling have been used:

- *email* is the most common form used online, and is required by IETF Requests for Comments (RFC) and working groups<sup>[7]</sup> and increasingly by style guides.<sup>[8][9]</sup> This spelling also appears in most dictionaries.<sup>[10][11][12][13][14][15][16]</sup>
- *e-mail* is the format that sometimes appears in edited, published American English and British English writing as reflected in the Corpus of Contemporary American English data,<sup>[17]</sup> but is falling out of favor in some style guides.<sup>[9][18]</sup>
- *mail* was the form used in the original protocol standard, RFC 524.<sup>[19]</sup> The service is referred to as *mail*, and a single piece of electronic mail is called a *message*.<sup>[20][21]</sup>
- *EMail* is a traditional form that has been used in RFCs for the "Author's Address"<sup>[20][21]</sup> and is expressly required "for historical reasons".<sup>[22]</sup>
- *E-mail* is sometimes used, capitalizing the initial *E* as in similar abbreviations like *E-piano*, *E-guitar*, *A-bomb*, and *H-bomb*.<sup>[23]</sup>

## Origin

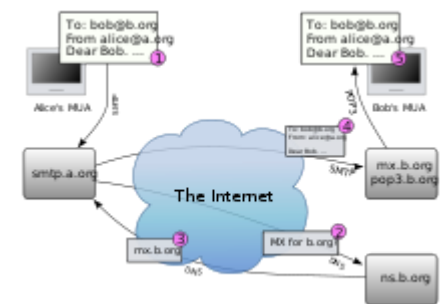
Computer-based mail and messaging became possible with the advent of time-sharing computers in the early 1960s, and informal methods of using shared files to pass messages were soon expanded into the first mail systems. Most developers of early mainframes and minicomputers developed similar, but generally incompatible, mail applications. Over time, a complex web of gateways and routing systems linked many of them. Many US universities were part of the ARPANET (created in the late-1960s), which aimed at software portability between its systems. That portability helped make the Simple Mail Transfer Protocol (SMTP) increasingly influential.

For a time in the late 1980s and early 1990s, it seemed likely that either a proprietary commercial system or the X.400 email system, part of the Government Open Systems Interconnection Profile (GOSIP), would predominate. However, once the final restrictions on carrying commercial traffic over the Internet ended in 1995,<sup>[26][27]</sup> a combination of factors made the current Internet suite of SMTP, POP3 and IMAP email protocols the standard.

## Operation

The diagram to the right shows a typical sequence of events<sup>[28]</sup> that takes place when sender Alice transmits a message using a mail user agent (MUA) addressed to the email address of the recipient.

1. The MUA formats the message in email format and uses the submission protocol, a profile of the Simple Mail Transfer Protocol (SMTP), to send the message content to the local mail submission agent (MSA), in this case *smtp.a.org*.
2. The MSA determines the destination address provided in the SMTP protocol (not from the message header), in this case *bob@b.org*. The part before the @ sign is the *local part* of the address, often the username of the recipient, and the part after the @ sign is a domain name. The MSA resolves a domain name to determine the fully qualified domain name of the mail server in the Domain Name System (DNS).
3. The DNS server for the domain *b.org* (*ns.b.org*) responds with any MX records listing the mail exchange servers for that domain, in this case *mx.b.org*, a message transfer agent (MTA) server run by the recipient's ISP.<sup>[29]</sup>
4. smtp.a.org sends the message to mx.b.org using SMTP. This server may need to forward the message to other MTAs before the message reaches the final message delivery agent (MDA).
5. The MDA delivers it to the mailbox of user *bob*.
6. Bob's MUA picks up the message using either the Post Office Protocol (POP3) or the Internet Message Access Protocol (IMAP).



Email operation

In addition to this example, alternatives and complications exist in the email system:

- Alice or Bob may use a client connected to a corporate email system, such as IBM Lotus Notes or Microsoft Exchange. These systems often have their own internal email format and their clients typically communicate with the email server using a vendor-specific, proprietary protocol. The server sends or receives email via the Internet through the product's Internet mail gateway which also does any necessary reformatting. If Alice and Bob work for the same company, the entire transaction may happen completely within a single corporate email system.
- Alice may not have a MUA on her computer but instead may connect to a webmail service.
- Alice's computer may run its own MTA, so avoiding the transfer at step 1.
- Bob may pick up his email in many ways, for example logging into mx.b.org and reading it directly, or by using a webmail service.
- Domains usually have several mail exchange servers so that they can continue to accept mail even if the primary is not available.

Many MTAs used to accept messages for any recipient on the Internet and do their best to deliver them. Such MTAs are called *open mail relays*. This was very important in the early days of the Internet when network connections were unreliable.<sup>[30][31]</sup> However, this mechanism proved to be exploitable by originators of unsolicited bulk email and as a consequence open mail relays have become rare,<sup>[32]</sup> and many MTAs do not accept messages from open mail relays.

## Message format

---

The Internet email message format is now defined by RFC 5322, with encoding of non-ASCII data and multimedia content attachments being defined in RFC 2045 through RFC 2049, collectively called *Multipurpose Internet Mail Extensions* or *MIME*. RFC 5322 replaced the earlier RFC 2822 in 2008, and in turn RFC 2822 in 2001 replaced RFC 822 – which had been the standard for Internet email for nearly 20 years. Published in 1982, RFC 822 was based on the earlier RFC 733 for the ARPANET.<sup>[33]</sup>

Internet email messages consist of two major sections, the message header and the message body, collectively known as content.<sup>[34][35]</sup> The header is structured into fields such as From, To, CC, Subject, Date, and other information about the email. In the process of transporting email messages between systems, SMTP communicates delivery parameters and information using message header fields. The body contains the message, as unstructured text, sometimes containing a signature block at the end. The header is separated from the body by a blank line.

### Message header

Each message has exactly one header, which is structured into fields. Each field has a name and a value. RFC 5322 specifies the precise syntax.

Informally, each line of text in the header that begins with a printable character begins a separate field. The field name starts in the first character of the line and ends before the separator character ":". The separator is then followed by the field value (the "body" of the field). The value is continued onto subsequent lines if those lines have a space or tab as their first character. Field names and values are restricted to 7-bit ASCII characters. Some non-ASCII values may be represented using MIME encoded words.

### Header fields

Email header fields can be multi-line, and each line should be at most 78 characters long and in no event more than 998 characters long.<sup>[36]</sup> Header fields defined by RFC 5322 can only contain US-ASCII characters; for encoding characters in other sets, a syntax specified in RFC 2047 can be used.<sup>[37]</sup> Recently the IETF EAI working group has defined some standards track extensions,<sup>[38][39]</sup> replacing previous experimental extensions, to allow UTF-8 encoded Unicode characters to be used within the header. In particular, this allows email addresses to use non-ASCII characters. Such addresses are supported by Google and Microsoft products, and promoted by some governments.<sup>[40]</sup>

The message header must include at least the following fields:<sup>[41][42]</sup>

- *From*: The email address, and optionally the name of the author(s). In many email clients not changeable except through changing account settings.
- *Date*: The local time and date when the message was written. Like the *From*: field, many email clients fill this in automatically when sending. The recipient's client may then display the time in the format and time zone local to them.

RFC 3864 describes registration procedures for message header fields at the IANA; it provides for permanent (<http://www.iana.org/assignments/message-headers/perm-headers.html>) and provisional (<http://www.iana.org/assignments/mess>

relevant RFCs. Common header fields for email include:<sup>[43]</sup>

- **To:** The email address(es), and optionally name(s) of the message's recipient(s). Indicates primary recipients (multiple allowed), for secondary recipients see Cc: and Bcc: below.
- **Subject:** A brief summary of the topic of the message. Certain abbreviations are commonly used in the subject, including "RE:" and "FW:".
- **Cc:** Carbon copy; Many email clients will mark email in one's inbox differently depending on whether they are in the To: or Cc: list. (Bcc: Blind carbon copy; addresses are usually only specified during SMTP delivery, and not usually listed in the message header.)
- **Content-Type:** Information about how the message is to be displayed, usually a MIME type.
- **Precedence:** commonly with values "bulk", "junk", or "list"; used to indicate that automated "vacation" or "out of office" responses should not be returned for this mail, e.g. to prevent vacation notices from being sent to all other subscribers of a mailing list. Sendmail uses this field to affect prioritization of queued email, with "Precedence: special-delivery" messages delivered sooner. With modern high-bandwidth networks, delivery priority is less of an issue than it once was. Microsoft Exchange respects a fine-grained automatic response suppression mechanism, the *X-Auto-Response-Suppress* field.<sup>[44]</sup>
- **Message-ID:** Also an automatically generated field; used to prevent multiple delivery and for reference in In-Reply-To: (see below).
- **In-Reply-To:** Message-ID of the message that this is a reply to. Used to link related messages together. This field only applies for reply messages.
- **References:** Message-ID of the message that this is a reply to, and the message-id of the message the previous reply was a reply to, etc.
- **Reply-To:** Address that should be used to reply to the message.
- **Sender:** Address of the actual sender acting on behalf of the author listed in the From: field (secretary, list manager, etc.).
- **Archived-At:** A direct link to the archived form of an individual email message.

Note that the *To:* field is not necessarily related to the addresses to which the message is delivered. The actual delivery list is supplied separately to the transport protocol, SMTP, which may or may not originally have been extracted from the header content. The "To:" field is similar to the addressing at the top of a conventional letter which is delivered according to the address on the outer envelope. In the same way, the "From:" field does not have to be the real sender of the email message. Some mail servers apply email authentication systems to messages being relayed. Data pertaining to server's activity is also part of the header, as defined below.

SMTP defines the *trace information* of a message, which is also saved in the header using the following two fields:<sup>[45]</sup>

- **Received:** when an SMTP server accepts a message it inserts this trace record at the top of the header (last to first).
- **Return-Path:** when the delivery SMTP server makes the *final delivery* of a message, it inserts this field at the top of the header.

Other fields that are added on top of the header by the receiving server may be called *trace fields*, in a broader sense.<sup>[46]</sup>

- **Authentication-Results:** when a server carries out authentication checks, it can save the results in this field for consumption by downstream agents.<sup>[47]</sup>
- **Received-SPF:** stores results of SPF checks in more detail than Authentication-Results.<sup>[48]</sup>
- **Auto-Submitted:** is used to mark automatically generated messages.<sup>[49]</sup>
- **VBR-Info:** claims VBR whitelisting<sup>[50]</sup>

## Message body

### Content encoding

# Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

## Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

## Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

## Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

## API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

## LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

## FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

## E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.