



Electronic Mail (SMTP, MIME, POP3, IMAP)

Bởi:

unknown

Electronic Mail (SMTP, MIME, POP3, IMAP)

Email là một trong những ứng dụng mạng lâu đời nhất nhưng lại phổ dụng nhất. Thử nghĩ khi bạn muốn gửi thông điệp đến một người bạn ở đầu kia của thế giới, bạn muốn mang thư chạy bộ qua đó hay chỉ đơn giản lên máy tính gõ ít hàng và nhấn nút **Send**? Thật ra, những bậc tiền bối của mạng ARPANET đã không tiên đoán được email sẽ là ứng dụng then chốt chạy trên mạng này, mục tiêu chính của họ là thiết kế hệ thống cho phép truy cập tài nguyên từ xa. Hệ thống email ra đời không mấy nổi bật, để bây giờ lại được sử dụng hàng ngày bởi hàng triệu người trên thế giới.

Mục tiêu của phần này là chỉ ra những nhân vật hoạt động trong hệ thống email, vai trò của họ, giao thức mà họ sử dụng và khuôn dạng thông điệp mà họ trao đổi với nhau.

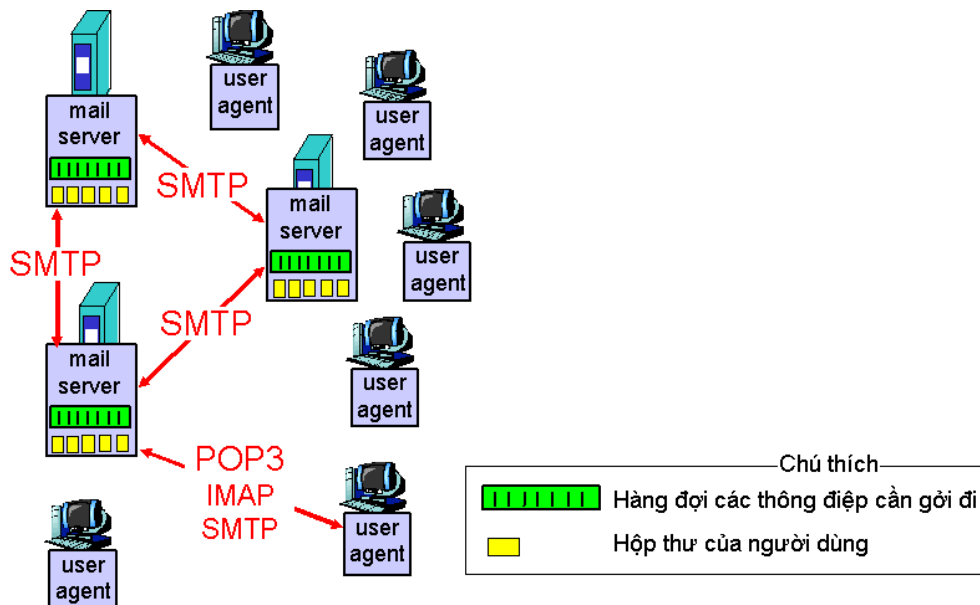
Các thành phần của hệ thống email

Một hệ thống email thường có 3 thành phần chính: Bộ phận trợ giúp người dùng (User Agent), Mail Server và các giao thức mà các thành phần này dùng để giao tiếp với nhau.

Người ta phân loại các giao thức như sau:

- Giao thức giữa các mail servers bao gồm:
 - SMTP (Simple Mail Transfer Protocol): được các server dùng để chuyển thư qua lại với nhau. Ví dụ nôm na, nó giống như cách thức mà các trạm bưu điện dùng để chuyển các thùng thư của khách hàng cho nhau. Thông tin chi tiết về giao thức này được mô tả trong tài liệu RFC 822.
- Giao thức giữa mail server và user agent bao gồm:
 - POP3 (Post Office Protocol version 3 [RFC 1939]): được user agent sử dụng để lấy thư về từ hộp thư của nó trên server.
 - SMTP: được user agent sử dụng để gửi thư ra server.

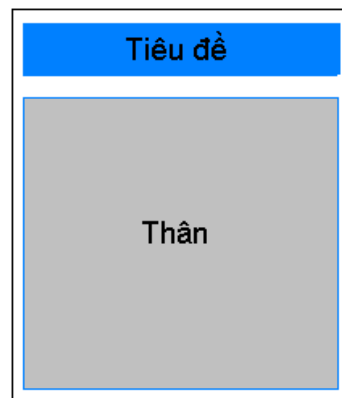
- IMAP: (Internet Mail Access Protocol [RFC 1730]): Có nhiều tính năng vượt trội hơn POP3. Ngoài ra IMAP còn cho phép gửi mail.



Các thành phần của hệ thống email (H8.6)

Khuôn dạng của một email

RFC 822 định nghĩa một email gồm có hai phần: phần tiêu đề (header) và phần thân (body).



Khuôn dạng của email (H8.7)

Cả hai phần đều được thể hiện dưới dạng ký tự ASCII. Lúc đầu, phần thân được qui định có khuôn dạng văn bản đơn giản. Sau này người ta đề nghị một chuẩn mới gọi là MIME, có thể cho phép phần thân của email chứa bất kỳ loại dữ liệu nào.

Phần tiêu đề bao gồm nhiều dòng thông tin, mỗi dòng kết thúc bằng hai ký tự <CRLF>. Phần tiêu đề được chia khỏi phần thân bởi một hàng rỗng. Mỗi một hàng tiêu đề chứa

một cặp “tên” và “giá trị”, cách nhau bởi dấu hai chấm (:). Người dùng có thể rất quen với nhiều hàng tiêu đề vì họ thường phải điền thông tin vào đây. Ví dụ

Tên	Giá trị
From:	Địa chỉ người gửi
To:	Địa chỉ của người nhận
Subject:	Chủ đề thư
Date:	Ngày gửi

RFC 822 được mở rộng năm 1993 (và được cập nhật lại năm 1996) để cho phép email mang được nhiều loại dữ liệu: audio, video, hình ảnh, tài liệu Word, ... MIME (Multipurpose Internet Mail Extensions) về cơ bản có ba phần. Phần đầu tiên là tập các dòng header dùng để bổ túc cho phần header cũ của RFC 822. Theo nhiều cách, những dòng header này mô tả dữ liệu chứa trong phần thân. Cụ thể như sau:

Tên	Giá trị
MIME-Version:	Phiên bản MIME đang sử dụng
Content-Description:	Mô tả trong thư đang có dữ liệu gì
Content-Type:	Mô tả kiểu dữ liệu đang nằm trong thư
Content-Transfer-Encoding:	Mô tả cách thức mã hóa dữ liệu trong thư

Phần thứ hai là các định nghĩa cho một tập các kiểu nội dung (và kiểu con nếu có). Ví dụ một số kiểu mà MIME định nghĩa:

Kiểu	Ý nghĩa
image/gif	Ảnh dạng gif
image/ jpeg	Ảnh dạng jpeg
text/plain	Văn bản đơn giản
text/ richtext	Văn bản mở rộng (có đặt font chữ, được định dạng đậm, nghiêng hoặc gạch dưới ...)
application	Dữ liệu trong thư được xuất ra từ một ứng dụng nào đó. Chẳng hạn: <i>application/postscript</i> : tài liệu Postscript (.ps) <i>application/msword</i> : tài liệu Microsoft Word (.doc)

MIME cũng định nghĩa kiểu **multipart** để chỉ ra cách mà phần thân của thư mang nhiều loại dữ liệu khác nhau như thế nào. Chỉ có một kiểu con của **multipart** là **mixed** với ý nói rằng trong phần thân của thư có nhiều mảnh dữ liệu khác nhau, độc lập với nhau và được sắp xếp theo một trình tự cụ thể. Mỗi mảnh dữ liệu sẽ có phần tiêu đề riêng để mô tả kiểu dữ liệu của mảnh đó.

Phần thứ ba mô tả cách thức mã hóa các kiểu dữ liệu nói trên để có thể truyền chúng dưới dạng ASCII. Lý do để mọi bức thư phải chứa các ký tự ASCII là vì để đi được đến đích, bức thư đó có thể phải trung chuyển qua nhiều gateway, mà các gateway này đều coi mọi bức thư dưới dạng ASCII. Nếu trong thư chứa bất kỳ ký tự nào khác ASCII thì thư sẽ bị đứt gãy nội dung. MIME sử dụng phương pháp mã hóa trực tiếp dữ liệu nhị phân thành các ký tự nhị phân, gọi là **base64**. Ý tưởng của **base64** là ánh xạ 3 bytes dữ liệu nhị phân nguyên thủy thành 4 ký tự ASCII. Giải thuật đơn giản như sau: tập hợp 3 bytes dữ liệu nhị phân lại thành 24 bits, sau đó chia 24 bits này thành 4 cụm, một cụm 6 bits. Một cụm 6 bits được ánh xạ vào một trong 64 ký tự ASCII hợp lệ; ví dụ 0 ánh xạ thành A, 1 ánh xạ thành B... Nếu nhìn vào bức thư đã được mã hóa dạng **base64**, người dùng sẽ thấy chỉ có 52 chữ cái cả hoa lẫn thường, 10 chữ số từ 0 đến 9 và các ký tự đặc biệt + và /.

Đối với những người dùng chỉ sử dụng trình đọc thư hỗ trợ duy nhất kiểu ký tự thì việc đọc những bức thư có kiểu **base64** sẽ rất là đau khổ. Vì lý do nhân đạo, MIME còn hỗ trợ kiểu mã hóa ký tự thường được gọi là **7-bit**. **7-bit** sẽ giữ nguyên dạng ký tự mà người ta nhập vào.

Tổng hợp lại, ví dụ một bức thư có 2 loại dữ liệu: văn bản thường, một ảnh JPEG, sẽ có hình dáng như sau:

```
From: ptpphi@cit.ctu.edu.vn
To: TH27@cit.ctu.edu.vn
Subject: Picture of students.
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="--98766789"

--98766789
Content-Transfer-Encoding: 7bit
Content-Type: text/plain

Hi,

Please find a picture of you.

--98766789
Content-Transfer-Encoding: base64
```

Chuyển thư

Kể đến, chúng ta sẽ xem xét giao thức SMTP – giao thức được dùng để chuyển thư từ máy này đến máy kia. Để đặt SMTP vào đúng ngữ cảnh, chúng ta nên nhắc lại các nhân vật then chốt trong hệ thống email. Đầu tiên, người dùng tương tác với trình đọc thư (hay còn gọi là user agent) để soạn, lưu, tìm kiếm và đọc thư của họ. Hiện trên thị trường có nhiều phần mềm đọc thư, cũng giống như hiện cũng đang có nhiều loại trình duyệt Web vậy. Thứ hai, có trình xử lý thư (hay còn gọi là mail server) chạy trên một máy nào đó trong mạng nội bộ của người dùng. Có thể xem mail server như một bưu điện: Người dùng trao cho mail server các bức thư mà họ muốn gửi cho người dùng khác, mail server sử dụng giao thức SMTP trên TCP để chuyển bức các thư này đến mail server bên đích. Mail server bên đích nhận các thư đến và đặt chúng vào hộp thư của người dùng bên đích. Do SMTP là giao thức mà rất nhiều người có thể tự cài đặt, vì thế sẽ có rất nhiều sản phẩm mail server hiện có trên thị trường. Sản phẩm mail server thường được sử dụng nhất là **sendmail**, ban đầu được cài đặt trong hệ điều hành Berkeley Unix.

Tất nhiên mail server bên máy gửi có thể kết nối SMTP/TCP trực tiếp tới mail server bên máy nhận, nhưng trong thực tế, một bức thư có thể đi ngang qua vài mail gateways trước khi đến đích. Cũng giống như máy đích, mỗi mail gateway cũng chạy một mail server. Không phải ngẫu nhiên mà các nút chuyển thư trung gian được gọi là mail gateway. Công việc của chúng cũng giống như các IP gateway là lưu tạm và chuyển phát tiếp các bức thư của người dùng. Điểm khác nhau duy nhất giữa chúng là, mail gateway trữ tạm các bức thư trong đĩa, trong khi các IP gateway trữ tạm các gói tin IP trong bộ nhớ.

Bạn có thể đặt câu hỏi: tại sao lại cần đến các mail gateways? Tại sao không dùng phương pháp nối kết SMTP/TCP trực tiếp từ bên gửi sang bên nhận? Lý do thứ nhất, người gửi không muốn kèm trong thư địa chỉ của máy đích. Ví dụ, riêng việc nhập vào trong thư địa chỉ đích **ptphi@cit.ctu.edu.vn** đã mất công rồi, không ai thấy thoải mái khi phải nhập thêm địa chỉ máy đích là **machine-of-phi.cit.ctu.edu.vn**. Thứ hai, không chắc lúc bên gửi thiết lập nối kết đến bên nhận, người dùng bên nhận đã bật sẵn máy! Thành thử chỉ cần địa chỉ thư bên nhận là đủ. Khi bức thư đến được mail gateway của Khoa Công Nghệ Thông Tin – Đại học Cần Thơ, nếu người dùng **ptphi** đang mở máy, mail gateway sẽ chuyển thư cho anh ta ngay, nếu không mail gateway sẽ trữ tạm thư trên đĩa của nó đến khi **ptphi** bật máy lên và kiểm tra thư.

Dù có bao nhiêu mail gateways trung gian trên đường đến đích vẫn không đáng lo lắng, bởi vì mỗi mail gateway trung gian sẽ nỗ lực sử dụng một kết nối SMTP độc lập đến gateway kế tiếp trên đường đi nhằm chuyển thư càng ngày càng đến gần người nhận.

SMTP là một giao thức đơn giản dùng các ký tự ASCII. Sau khi thiết lập nối kết TCP đến cổng 25 của máy đích (được coi là server), máy nguồn (được coi là client) chờ nhận kết quả trả về từ server. Server khởi đầu cuộc đối thoại bằng cách gửi một dòng văn bản đến client thông báo danh tính của nó và khả năng tiếp nhận thư. Nếu server không có khả năng nhận thư tại thời điểm hiện tại, client sẽ hủy bỏ nối kết và thử thiết lập lại nối kết sau.

Nếu server sẵn sàng nhận thư, client sẽ thông báo lá thư đó từ đâu đến và ai sẽ là người nhận. Nếu người nhận đó tồn tại, server sẽ thông báo cho client tiếp tục gửi thư. Sau đó client gửi thư và server báo nhận cho thư đó. Sau khi cả hai bên hoàn tất phiên truyền nhận, kết nối sẽ được đóng lại.

Ví dụ một phiên truyền nhận được cho ngay dưới đây. Những dòng bắt đầu bằng **C:** là của phía client gửi đi; bằng **S:** là các câu trả lời của server.

```

S: 220 ctu.edu.vn
C: HELO cit.ctu.edu.vn
S: 250 ctu.edu.vn says hello to cit.ctu.edu.vn
C: MAIL FROM: <ptphi@cit.ctu.edu.vn>
S: 250 Sender ok
C: RCPT TO: <lhly@yale.edu>
S: 250 Recipient ok
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: Subject: It's Xmas!
C: So I hope you a merry Xmas and a happy new year!
C: .
    
```

Như đã thấy trong ví dụ, client gửi đi một lệnh (**HELO, MAIL FROM, RCPT TO, DATA, QUIT**) và server trả lời bằng một mã số (250, 354, 221) có kèm theo lời chú thích có thể đọc được. Client kết thúc thư bằng <CRLF>.<CRLF>. Sau đây là bảng giải thích một số lệnh của client và mã số trả lời của server.

LỆNH CỦA CLIENT	
Lệnh	Ý nghĩa
HELO	Câu chào và xưng danh của client
MAIL FROM	Địa chỉ email của người gửi
RCPT TO	Địa chỉ email của người nhận
DATA	Bắt đầu truyền nội dung của thư
QUIT	Hủy nối kết
TRẢ LỜI CỦA SERVER	
Trả lời	Ý nghĩa
250	Yêu cầu hợp lệ
550	Yêu cầu không hợp lệ, không tồn tại hộp thư như client đã chỉ ra.
354	Cho phép bắt đầu nhập thư vào. Kết thúc thư bằng <CRLF>.<CRLF>

221	Server đang đóng kết nối TCP
-----	------------------------------

Vẫn còn nhiều lệnh và mã trả lời chưa được trình bày, xin tham khảo tài liệu RFC 822 để có được đầy đủ thông tin.

Phân phát thư

Như đã trình bày, khi đứng về góc độ người dùng thư, họ sẽ dùng user agent để gửi và nhận thư cho họ. User agent dùng giao thức SMTP để gửi thư đi, dùng giao thức POP3 hoặc IMAP để nhận thư về.

POP3

Một phiên làm việc theo giao thức POP3 bắt đầu tại user agent. User agent khởi động một nối kết TCP đến cổng 110 của mail server. Khi kết nối thực hiện xong, phiên làm việc POP3 sẽ trải qua theo thứ tự ba kỳ:

1. Chứng thực.
2. Giao dịch dữ liệu.
3. Cập nhật.

Kỳ chứng thực buộc người dùng thực hiện thủ tục đăng nhập bằng cách nhập vào hai lệnh sau:

Lệnh	Ý nghĩa
USER <tên người dùng>	Khai báo tên người dùng.
PASS <mật khẩu>	Khai báo mật khẩu.

Báo trả của mail server sẽ là một trong hai câu sau:

Trả lời	Ý nghĩa
+OK <chú thích>	Khai báo của người dùng là đúng.
+ERR <chú thích>	Khai báo của người dùng là sai và lời giải thích.

Trong kỳ giao dịch, người dùng có thể xem danh sách thư chưa nhận về, nhận thư về và xóa thư trong hộp thư của mình khi cần thiết. Các lệnh mà người dùng thường sử dụng để giao dịch với server là:

Lệnh	Ý nghĩa
------	---------

LIST[<số thứ tự thư>]	Nếu dùng LIST không tham số, server sẽ trả về toàn bộ danh sách các thư chưa nhận. Nếu có tham số là số thứ tự thư cụ thể, server sẽ trả về thông tin của chỉ bức thư đó thôi.
RETR <số thứ tự thư>	Tải lá thư có số thứ tự <số thứ tự thư> về.
DELE<số thứ tự thư>	Xóa lá thư số <số thứ tự thư> khỏi hộp thư.
QUIT	Hoàn tất giai đoạn giao dịch và hủy nối kết TCP

Các trả lời của server có thể là các số liệu mà client yêu cầu hoặc các thông báo +OK, -ERR như trong phần đăng nhập.

Sau đây là dàn cảnh một phiên làm việc ví dụ giữa người dùng **ptphi** khi anh ta đăng nhập và làm việc trên hộp thư của mình tại server có địa chỉ **mail.cit.ctu.edu.vn**.

Client	Server	Giải thích
	+OK POP3 server ready	Server sẵn sàng phục vụ client
USER ptpfi		
	+OK	Server xác nhận người dùng hợp lệ
PASS godblessus		
	+OK login successfully	Chứng thực thành công
LIST		ptphi kiểm tra hộp thư
	+OK1 10242 2550	Hộp thư của ptpfi còn hai thư chưa nhận về, thư thứ nhất có kích thước 1024 bytes, thư thứ hai có kích thước 2550 bytes
RETR 1		ptphi tải thư thứ nhất về
	+OK	server gửi thư thứ 1 cho ptpfi
DELE 1		ptphi xóa thư thứ nhất trong hộp thư

	+OK	server xoá thư thứ 1 thành công
QUIT		ptphi hủy nối kết
	+OK Bye-Bye	server hủy nối kết

IMAP

Với những người dùng có một tài khoản email trên một ISP và người dùng này thường truy cập email trên một PC thì giao thức POP3 hoạt động tốt. Tuy nhiên, một sự thật trong ngành công nghệ máy tính, khi một thứ gì đó đã hoạt động tốt, người ta lập tức đòi hỏi thêm nhiều tính năng mới (và tự chuốc lấy nhiều phiền nhiễu). Điều đó cũng xảy ra đối với hệ thống email. Ví dụ, người ta chỉ có một tài khoản email, nhưng họ lại muốn ngồi đâu cũng truy cập được nó. POP3 cũng làm được chuyện này bằng cách đơn giản tải hết các email xuống máy PC mà người dùng này đang ngồi làm việc. Và dĩ nhiên là thư từ của người dùng này nằm rải rác khắp nơi.

Sự bất tiện này khơi mào cho sự ra đời của giao thức phân phối thư mới, IMAP (Internet Message Access Protocol), được định nghĩa trong RFC 2060. Không giống như POP2, IMAP coi các thông điệp mặc nhiên nằm trên server vô hạn và trên nhiều hộp thư. IMAP còn đưa ra cơ chế cho phép đọc các thông điệp hoặc một phần của thông điệp, một tính năng hữu ích khi người dùng kết nối đến server bằng đường truyền tốc độ chậm như điện thoại nhưng lại đọc các email có âm thanh, hình ảnh... Với quan niệm cho rằng người dùng không cần tải thư về lưu trên PC, IMAP cung cấp các cơ chế cho phép tạo, xóa và sửa đổi nhiều hộp thư trên server.

Cung cách làm việc của IMAP cũng giống như POP3, ngoài trừ trong IMAP có rất nhiều lệnh. IMAP server sẽ lắng nghe trên cổng 143. Cũng nên chú ý rằng, không phải mọi ISP đều hỗ trợ cả hai giao thức POP3 và IMAP.

Bảng sau so sánh các tính năng của POP3 và IMAP

Tính năng	POP3	IMAP
Giao thức được định nghĩa ở đâu?	RFC 1939	RFC 2060
Cổng TCP được dùng	110	143
Email được lưu ở đâu	PC của người dùng	Server
Email được đọc ở đâu	Off-line	On-line
Thời gian nối kết	Ít	Nhiều

Sử dụng tài nguyên của server	Tối thiểu	Nhiều hơn
Nhiều hộp thư	Không	Đúng
Ai lưu phòng hờ các hộp thư	Người dùng	ISP
Tốt cho người dùng di động	Không	Có
Kiểm soát của người dùng đối với việc tải thư về	Ít	Tốt
Tải một phần thư	Không	Có
Quota đĩa có là vấn đề không?	Không	Thỉnh thoảng
Dễ cài đặt	Có	Không
Được hỗ trợ rộng rãi	Có	Đang phát triển