World Wide Web (HTTP)

Bởi:

unknown

World Wide Web (HTTP)

Ứng dụng Web đã rất thành công, giúp cho nhiều người có thể truy cập Internet đến nỗi Web được hiểu đồng nghĩa với Internet! Có thể hiểu Web như là một tập các client và server hợp tác với nhau và cùng nói chung một ngôn ngữ: HTTP (Hyper Text Transfer Protocol). Đa phần người dùng tiếp xúc với Web thông qua chương trình client có giao diện đồ họa, hay còn gọi là trình duyệt Web (Web browser). Các trình duyệt Web thường được sử dụng nhất là Netscape Navigator (của Netscape) và Internet Explorer (của Microsoft). Hình H8.8 thể hiện trình duyệt Explorer đang trình bày trang chủ của Khoa Công Nghệ Thông Tin – Đại Học Cần Thơ:

Trình duyệt Web Internet Explorer (H8.8)

Bất kỳ trình duyệt Web nào cũng có chức năng cho phép người dùng "mở một URL". Các URL (Uniform Resource Locators) cung cấp thông tin về vị trí của các đối tượng trên Internet; chúng thường trông giống như sau:

http://www.cit.ctu.edu.vn/index.html

Nếu người dùng mở URL trên, trình duyệt Web sẽ thiết lập một kết nối TCP đến Web Server tại địa chỉ **www.cit.ctu.edu.vn** và ngay lập tức tải tập tin **index.html** về và thể hiện nó. Hầu hết các tập tin trên Web chứa văn bản và hình ảnh, một số còn chứa audio và video clips. Chúng còn có thể chứa các liên kết đến các tập tin khác – được gọi là các liên kết siêu văn bản (hypertext links). Khi người dùng yêu cầu trình duyệt Web mở ra một liên kết siêu văn bản (bằng cách trỏ chuột và click lên liên kết đó), trình duyệt sẽ mở một nối kết mới, tải về và hiển thị một tập tin mới. Vì thế, rất dễ để duyệt từ server này đến server khác trên khắp thế giới để có được hết những thông tin mà người dùng cần.

Khi người dùng chọn xem một trang Web, trình duyệt Web sẽ nạp trang Web đó từ Web server về sử dụng giao thức HTTP chạy trên TCP. Giống như SMTP, HTTP là giao thức hướng ký tự. Về cốt lõi, một thông điệp HTTP có khuôn dạng tổng quát sau:

START_LINE <CRLF>

MESSAGE_HEADER <CRLF>

<CRLF>

MESSAGE_BODY <CRLF>

Hàng đầu tiên chỉ ra đấy là thông điệp yêu cầu hay trả lời. Nó sẽ chỉ ra "thủ tục cần được thực hiện từ xa" (trong tình huống là thông điệp yêu cầu) hoặc là "trạng thái trả về" (trong tình huống là thông điệp trả lời). Tập hợp các hàng kế tiếp chỉ ra các tùy chọn hoặc tham số nhằm xác định cụ thể tính chất của yêu cầu hoặc trả lời. Phần MESSAGE_HEADER có thể không có hoặc có một vài hàng tham số và được kết thúc bằng một hàng trống. HTTP định nghĩa nhiều kiểu header, một số liên quan đến các thông điệp yêu cầu, một số liên quan đến các thông điệp trả lời và một số lại liên quan đến phần dữ liệu trong thông điệp. Ở đây chỉ giới thiệu một số kiểu thường dùng. Cuối cùng, sau hàng trống là phần nội dung của thông điệp trả lời (MESSAGE_BODY), phần này thường là rỗng trong thông điệp yêu cầu.

Các thông điệp yêu cầu

Hàng đầu tiên của một thông điệp yêu cầu HTTP sẽ chỉ ra 3 thứ: thao tác cần được thực thi, trang Web mà thao tác đó sẽ áp lên và phiên bản HTTP được sử dụng. Bảng sau sẽ giới thiệu một số thao tác phổ biến.

Hành động	Mô tả
OPTIONS	Yêu cầu thông tin về các tùy chọn hiện có.
GET	Lấy về tài liệu được xác định trong URL
HEAD	Lấy về thông tin thô về tài liệu được xác định trong URL
POST	Cung cấp thông tin cho server
PUT	Tải tài liệu lên server và đặt ở vị trí được xác định trong URL
DELETE	Xóa tài liệu nằm ở vị trí URL trên server
TRACE	Phản hồi lại thông điệp yêu cầu
CONNECT	Được sử dụng bởi các proxy

Hai thao tác thường được sử dụng nhiều nhất là GET (lấy một trang Web về) và HEAD (lấy về thông tin của một trang Web). GET thường được sử dụng khi trình duyệt muốn tải một trang Web về và hiển thị nó cho người dùng. HEAD thường được sử dụng để kiểm tra tính hợp lệ của một liên kết siêu văn bản hoặc để xem một trang nào đó có bị thay đổi gì không kể từ lần tải về trước đó.

Ví dụ, dòng START_LINE

GET http://www.cit.ctu.edu.vn/index.html HTTP/1.1

nói rằng: người dùng muốn tải về trên server www.cit.ctu.edu.vn trang Web có tên index.html và hiển thị nó. Ví dụ trên dùng URL tuyệt đối. Ta cũng có thể sử dụng URL tương đối như sau:

GET /index.html HTTP/1.1Host: www.cit.ctu.edu.vn

Ở đây, **Host** là một trong các trường trong MESSAGE_HEADER.

Các thông điệp trả lời

Giống như các thông điệp yêu cầu, các thông điệp trả lời bắt đầu bằng một hàng START_LINE. Trong trường hợp này, dòng START_LINE sẽ chỉ ra phiên bản HTTP đang được sử dụng, một mã 3 ký số xác định yêu cầu là thành công hay thất bại và một chuỗi ký tự chỉ ra lý do của câu trả lời này.

Ví dụ, dòng START_LINE

HTTP/1.1 202 Accepted

chỉ ra server đã có thể thõa mãn yêu cầu của người dùng.

Còn dòng

HTTP/1.1 404 Not Found

chỉ ra rằng server đã không thể tìm thấy tài liệu như được yêu cầu.

Có năm loại mã trả lời tổng quát với ký số đầu tiên xác định loại mã.

Mã	Loại	Lý do
1xx	Thông tin	Đã nhận được yêu cầu, đang tiếp tục xử lý
2xx	Thành công	Thao tác đã được tiếp nhận, hiểu được và chấp nhận được

3xx	Chuyển hướng	Cần thực hiện thêm thao tác để hoàn tất yêu cầu được đặt ra
4xx	Lỗi client	Yêu cầu có cú pháp sai hoặc không thể được đáp ứng
5xx	Lỗi server	Server thất bại trong việc đáp ứng một yêu cầu hợp lệ

Cũng giống như các thông điệp yêu cầu, các thông điệp trả lời có thể chứa một hoặc nhiều dòng trong phần MESSAGE_HEADER. Những dòng này cung cấp thêm thông tin cho client. Ví dụ, dòng header **Location** chỉ ra rằng URL được yêu cầu đang có ở vị trí khác. Vì thế, nếu trang Web của Khoa Công Nghệ Thông Tin được di chuyển từ địa chỉ **http://www.cit.ctu.edu.vn/index.html** sang địa chỉ **http://www.ctu.edu.vn/cit/index.html** mà người dùng lại truy cập vào URL cũ, thì Web server sẽ trả lời như sau

HTTP/1.1 301 Moved Permanently Location: http://www.ctu.edu.vn/cit/index.html

Trong tình huống chung nhất, thông điệp trả lời cũng sẽ mang theo nội dung trang Web được yêu cầu. Trang này là một tài liệu HTML, nhưng vì nó có thể chứa dữ liệu không phải dạng văn bản (ví dụ như ảnh GIF), dữ liệu này có thể được mã hóa theo dạng MIME. Một số hàng trong phần MESSAGE_HEADER cung cấp thêm thông tin về nội dung của trang Web, bao gồm **Content-Length** (số bytes trong phần nội dung), **Expires** (thời điểm mà nội dung trang Web được xem như lỗi thời), và **Last-Modified** (thời điểm được sửa đổi lần cuối cùng).

Các kết nối TCP

Nguyên tắc chung của giao thức HTTP là client nối kết đến cổng TCP số 80 tại server, server luôn lắng nghe trên cổng này để sẵn sàng phục vụ client. Phiên bản đầu tiên (HTTP/1.0) sẽ thiết lập một nối kết riêng cho mỗi hạng mục dữ liệu cần tải về từ server. Không khó để thấy rằng đây là cơ chế không mấy hiệu quả: Các thông điệp dùng để thiết lập và giải phóng nối kết sẽ phải được trao đổi qua lại giữa client và server và khi mà tất cả client muốn lấy thông tin mới nhất của một trang Web, server sẽ bị quá tải.

Cải tiến quan trọng nhất trong phiên bản HTTP/1.1 là nó cho phép các kết nối lâu dài – client và server sẽ trao đổi nhiều thông điệp yêu cầu/trả lời trên cùng một kết nối TCP. Kết nối lâu dài có hai cái lợi. Thứ nhất, nó làm giảm thiểu chi phí cho việc thiết lập/giải phóng nối kết. Thứ hai, do client gởi nhiều thông điệp yêu cầu qua một kết nối TCP, cơ chế điều khiển tắc nghẽn của TCP sẽ hoạt động hiệu quả hơn.

Tuy nhiên, kết nối lâu dài cũng có cái giá phải trả. Vấn đề phát sinh ở chỗ: không ai trong client và server biết được kết nối đó sẽ kéo dài bao lâu. Điều này thực sự gây khó khăn cho phía server bởi vì tại mỗi thời điểm, nó phải đảm bảo duy trì kết nối đến cả ngàn client. Giải pháp cho vấn đề này là: server sẽ mãn kỳ và cắt nối kết nếu nó không nhận được một yêu cầu cụ thể nào từ phía client trong một khoảng thời gian định trước. Ngoài ra, cả client và server phải theo dõi xem phía bên kia có chủ động cắt nối kết hay

không và lấy đó làm cơ sở để tự cắt nối kết của mình. (Nhắc lại rằng, cả hai bên phải cắt nối kết thì nối kết TCP mới thực sự kết thúc).

Trữ đệm

Một trong những lĩnh vực nghiên cứu tích cực nhất hiện nay về Internet là làm sao để trữ tạm các trang Web một cách hiệu quả. Việc trữ tạm mang lại nhiều lợi ích. Từ phía client, việc nạp và hiển thị một trang Web từ bộ đệm gần đấy là nhanh hơn rất nhiều so với từ một server nào đó ở nửa vòng trái đất. Đối với server, có thêm một bộ đệm để can thiệp vào và phục vụ giúp yêu cầu của người dùng sẽ giảm bớt tải trên server.

Việc trữ đệm có thể được cài đặt tại nhiều nơi khác khau. Ví dụ, trình duyệt Web có thể trữ tạm những trang Web mới được nạp về gần đây, để khi người dùng duyệt lại những trang Web đó, trình duyệt sẽ không phải nối kết ra Internet để lấy chúng về mà dùng bản trữ sẵn. Ví dụ khác, một khu vực làm việc (site) có thể đề cử một máy làm nhiệm vụ trữ tạm các trang Web, để những người dùng sau có thể sử dụng các bản trữ sẵn của những người dùng trước. Yêu cầu của hệ thống này là mọi người dùng trong site phải biết địa chỉ của máy tính làm nhiệm vụ bộ trữ tạm, và họ chỉ đơn giản là liên hệ với máy tính này để tải các trang Web về theo yêu cầu. Người ta thường gọi máy tính làm nhiệm vụ trữ tạm các trang Web cho một site là *proxy*. Vị trí trữ đệm có thể di chuyển gần hơn đến phần lõi của Internet là các ISP. Trong tình huống này, các site nối kết tới ISP thường không hay biết gì về việc trữ tạm ở đây. Khi các yêu cầu HTTP từ các site được chuyển phát đến router của ISP, router liền kiểm tra xem URL được yêu cầu có giống với các URL được trữ sẵn hay không. Nếu có, router sẽ trả lời ngay. Nếu không, router sẽ chuyển yêu cầu đến server thật sự và cũng không quên lưu vào bộ đệm của mình thông điệp trả lời từ phía server đó.

Việc trữ tạm là đơn giản. Tuy nhiên bộ đệm phải đảm bảo những thông tin trữ đệm trong đó không quá cũ. Để làm được việc này, các Web server phải gán "ngày hết hạn" (tức là trường **Expires** trong header) cho mọi trang Web mà nó phục vụ cho client. Nhân đó, các bộ đệm cũng lưu lại thông tin này. Và từ đó, các bộ đệm sẽ không cần phải kiểm tra tính cập nhật của trang Web đó cho đến khi ngày hết hạn đến. Tại thời điểm một trang Web hết hạn, bộ đệm sẽ dùng lệnh HEAD hoặc lệnh GET có điều kiện (GET với trường **If-Modified-Since** trong phần header được đặt) để kiểm tra rằng nó có một phiên bản mới nhất của trang Web kia. Tổng quát hơn, cần phải có "các chỉ thị hướng dẫn" cho việc trữ đệm và các chỉ thị này phải được tuân thủ tại mọi bộ đệm. Các chỉ thị sẽ chỉ ra có nên trữ đệm một tài liệu hay không, trữ nó bao lâu, một tài liệu phải tươi như thế nào và vân vân.