

基于 HTTP 协议的文件上传的实现

彭作民

(南京师范大学计算中心,南京 210097)

[摘要] 分析了 HTTP 协议实现文件上传协议包的构成,在 DELPHI 应用程序中通过对 NMHTTP 控件编程,直接生成并发送上传文件的请求报文,从而实现了将任意类型的文件上传到 WEB 服务器中。

[关键词] HTTP;Web;Internet;文件上传

[中图分类号] TP393.4; [文献标识码] A; [文章编号] 1001-4616(2001)03-0036-05

目前,Internet 对许多企业和个人来说已不陌生,人们越来越多地利用互联网查询大量的信息。随着 Internet 的深入发展,大家已不仅仅满足于信息的获取,更需要信息的双向交换,特别是通过互联网传递文件,实现文件传输的网络通信协议有很多,如 FTP(文件传输协议),HTTP(超文本传输协议),SMTP(简单邮件传输协议),UDP(用户数据报协议)。本文通过分析 HTTP 协议实现文件上传的报文的构成,在 DELPHI 应用程序中直接编程生成上传的 HTTP 报文,并向 WEB 服务器发送报文,从而实现了文件的上传。

1 HTTP 请求与响应的报文结构

超文本传输协议 HTTP(Hyper-Text Transfer Protocol)是专门为传输 HTML 页面设计的一种网络通讯协议,它属于应用层协议,位于 TCP/IP 协议之上,在设计和使用时需要以 TCP/IP 协议中的其他协议为基础。HTTP 也是客户/服务器体系结构的典型代表,这里的客户是浏览器或其它应用程序,服务器则是 Web 服务器。

HTTP 事务处理的主要内容是客户端的请求与服务器端的响应。客户端向服务器发送连接请求报文,服务器响应客户的请求并向客户端发送响应报文,从而完成一次 HTTP 事务。

1.1 HTTP 请求报文

在 HTTP 报文中,请求报文的格式如下:

方法 URI HTTP 版本号

一般头标
请求头标
实体头标

} HTTP 头标

实体主体 实体内容

在上述格式中,实体主体与 HTTP 头标之间有一个空行,常用的 HTTP 请求的方法有 GET、HEAD、PUT、POST、DELETE、LINK、UNLINK,其中 GET、HEAD、POST 方法被大多数服务器支持,

其余的方法大多数服务器不支持。

1.2 HTTP 响应报文

在 HTTP 报文中,响应报文的格式如下:

HTTP 版本号 状态码 理由短语

一般头标
响应头标
实体头标

} HTTP 头标

实体主体 实体内容

在响应报文中,HTTP 头标与实体主体之间有一个空行,实体主体的长度由实体头标决定;

HTTP 响应的状态码为三位数字编码,其含义如下:

1xx 提供信息

2xx 成功,动作被接受、理解或访问。

3xx 更改,为完成请求所要求采取的动作。

4xx 客户端错,请求中有语法错或请求不能被执行。

5xx 服务器错,服务器不能执行看来有效的请求。

注:其中 xx 表示不同的响应状态编码。

2 文件上载的 HTTP 请求报文的特点

2.1 普通 HTTP 请求报文 通常有五个部分组成

(1)请求行:它包含方法、URI 和协议版本号。

(2)一般头标:一般包括 Date 字段、MIME-Version 字段、Pragma 字段。

(3)请求头标:包括 6 个字段,Accept 字段、Authorization 字段、From 字段、If-Modifide-since 字段、Referer 字段 User-Agent 字段。

(4)实体头标:包括 7 个字段,Allow 字段、Content-Encoding 字段、Content-length 字段、Content-Type 字段、Expires 字段、Extension-Header 字段、Last-Modified 字段。

(5)实体部分:请求的实体内容,实体与实体头标间有一个空行。

2.2 文件上载请求的 HTTP 报文与普通 HTTP 请求报文有三个地方是不同的

(1)请求的方法一定是 POST 方法。

(2)实体头标中的 Content-Type 字段值使用多成份内容-类型值。

在 HTTP 规范中,它是采用 MIME 来支持内容-类型标识(Content-Type),而 MIME 允许通过 HTTP 传递任意数据格式。MIME 内容-类型的语法如下:

类型/子类型;参数;...参数;

其中常用的类型有 Application、audio、image、message、multipart、text、video;子类型指定改类型数据的特定格式,对某些特定类型才有参数。

在文件上载的报文中,应使用多成份内容-类型(multipart),子类型可用 form-data,multipart 类型带有一个参数 boundary,它的值是一个不超过 70 个字节的字符串,以非空格结束。(使用来界定试题内容各成份的分隔字符串。)其内容-类型字段的格式如下:

Content-Type:multipart/form-data;boundary = 分隔字符串

(3)实体主体使用分隔字符串。

万方数据

普通请求报文是在实体头标后空一行,然后就是实体的内容,对实体内容未加任何标记或分隔符.

而多成份-类型的实体使用特殊的分隔字符串,其结构如下:

分隔字符串

Content-Disposition :form-data ;name =“ 上载的文件名 ”;filename =“ 路径文件名 ”

Content-Type ;application/octet-stream

空一行

上载的文件内容(字节流)

结束分隔字符串

其中:

(1)分隔字符串由实体头标中的 boundary 参数定义.

(2)Content-Type 值根据不同类型的文件给出不同的值,可用 :application/octet-stream 通配任何类型的文件.

(3)最后一个分隔字符串后接二个“-”字符,表示实体主体的结束标志,与表示开始的分隔字符串标志区分.

3 文件上载的程序实现

根据前述文件上载的报文构成,我们可以在应用程序中通过对 HTTP 协议的编程,生成上载文件所需的报文,然后向 WEB 服务器发送上载的请求报文,从而实现文件的上载.以下是 DELPHI 5 编写的生成并上载请求报文的一段程序,它的功能是将 c:\demo.bmp 文件上载到服务器中,与此无关的程序在此省略:

(1)将原文件按多成份-类型实体主体结构打包

首先要将原文件按 HTTP 协议的格式打包,生成实体主体内容,程序如下:

```
1 begin
2   assignfile( destfile , 'c:\demotemp.bmp' ); ...打包后的文件为 c:\demotemp.bmp
3   rewrite( destfile ,1 );
4   sss := '12345' ; ...定义分隔字符串
5   stringlength := length( sss );
6   strcopy( buf , sss );
7   seek( destfile , filesize( destfile ) );
8   blockwrite( destfile , buf , stringlength , NumRecsWritten );
9   ..... 加入回车换行符的语句在此省略
10  sss := 'Content-Disposition :form-data ;name = "file 1" ;filename = "c:\demotemp.bmp"' ;
11  stringlength := length( sss );
12  strcopy( buf , sss );
13  blockwrite( destfile , buf , stringlength , NumRecsWritten );
14  ..... 加入回车换行符的语句在此省略
15  sss := 'Content-Type :application/octet-stream' ;
16  stringlength := length( sss );
```

```

17      strcpy( buf ,sss );
18      blockwrite( destfile ,buf ,stringlength ,NumRecsWritten );
19      .....      加入回车换行符的语句在此省略
20      .....      加入回车换行符的语句在此省略
21      assignfile( srcfile ,'c : \ demo.bmp' );      ....指定要上传的源文件
22      reset( srcfile ,1 );
23      filesize := filesize( srcfile );
24      repeat
25          blockread( srcfile ,buffer ,size of( buffer ) ,bytesread );
26          if bytesread > 0 then
27              begin
28                  blockwrite( destfile ,buffer ,bytesread ,byteswritten );
29              end ;
30      until bytesread = 0 ;
31      .....      加入回车换行符的语句在此省略
32      sss := '12345--' ;      ....定义结束分隔字符串
33      stringlength := length( sss );
34      strcpy( buf ,sss );
35      blockwrite( destfile ,buf ,stringlength ,NumRecsWritten );
36      .....      加入回车换行符的语句在此省略
37      end

```

注 程序段 21 30 是将源文件 c : \ demo.bmp 按字节写入 demotemp.bmp 文件的末尾. 程序中的变量的声明在此省略.

(2) 生成并发送文件上传的请求报文

生成实体主体后, 还要生成上传文件的 HTTP 报文头标, 然后向 WEB 发送, 下面是相关的程序:

```

1      NMHTTP1.SendHeader.clear ;
2      NMHTTP1.SendHeader.Append( 'Post//uploadfile.exe HTTP1.1' );
3      NMHTTP1.SendHeader.Append( 'Accept :image/gif application/msword ,* / * ' );
4      NMHTTP1.SendHeader.Append( 'Accept-Language :zh-cn' );
5      NMHTTP1.SendHeader.Append( 'Host :192.168.1.1' );
6      NMHTTP1.SendHeader.Append( 'Connection :Keep-Alive' );
7      NMHTTP1.SendHeader.Append( 'Accept-Encoding :gzip deflate' );
8      NMHTTP1.SendHeader.Append( 'Content-Type :multipart/form-data ;boundary = 12345' );
9      NMHTTP1.OutputFileMode := true ;
10     NMHTTP1.Pos( 'http ://192.168.1.1/uploadfile.exe' ,'c : \ demotemp.bmp' );

```

注 其中 NMHTTP1 是 DELPHI 5 中的 NMHTTP 控件的一个实例. 本程序是将本地 C 盘根目录中的文件 DEMOTEMP.BMP 上传到服务器中, 上传是通过调用 NMHTTP 控件的 POST 方法 (第 10 行) 在调用前, 1 ~ 8 行程序段生成 HTTP 请求头标, 第 9 行是指定 POST 方法中的第二万方数据

项参数为文件名,而文件 DEMOTEMP.BMP 中已通过编程将分隔字符串以及相关的描述信息加入到其头尾中。

4 结束语

本文介绍了 HTTP 协议实现文件上载的报文结构及其详细的头标,利用上述方法和原理,作者成功地在一个网站应用系统中实现了文件上载下载的功能,效率很高。

[参考文献]

- [1] 乔林 杨志刚 等. DELPHI 5 编程——Internet 应用实务篇 [M]. 北京: 中国铁道出版社, 2000.
- [2] DEVRA HALL. 建立 WEB 站点的方法 [M]. 北京: 科学出版社, 1997.

Implementation of Uploading Files Based on HTTP Protocol

Peng Zuomin

(Computer Center of Nanjing Normal University ,Nanjing 210097 ,PRC)

Abstract :The paper analyzes the structure of HTTP packet for uploading files and describes how to directly generate and send the request packet for uploading files by programming with the NMHTTP Controls in DELPHI programs in order to implement uploading files from client to WEB server.

Key words :HTTP ;Web ;Internet ;upload

[责任编辑 陆炳新]