360doc
个人图书馆

首 页　　　阅览室　　　馆友　　　我的图书馆

搜文章 找馆友　　　搜 索

来自：champion_xu > cgi

配色：　字号：大 中 小

CGI上传文件

2012-04-24 | 阅：88 转：3 | 分享▼　转藏到我的图书馆

upload.html

```html
<HTML>
  <HEAD>
    <TITLE>上传数据文件</TITLE>
    <META HTTP-EQUIV="CONTENT-TYPE" CONTENT="TEXT/HTML;CHARSER=UTF-8">
    <link rel="stylesheet" type="text/css" href="css/main.css">
<script type="text/javascript">
  function upload_file(){
      if( document.getElementById('FILE1').value == '' ){
          alert("请选择文件！");
          return false;
      }

      document.getElementById('uploadInfo').innerHTML = '' ;
      document.getElementById('uploadInfo').innerHTML = '<img src="images/wait.gif"><br>数
据上传中，请稍候……';

      document.getElementById('form1').submit();
  }
</script>
  </HEAD>
  <BODY BGCOLOR="#FFFFFF">
    <FORM METHOD="POST" id="form1" name="form1" ENCTYPE="multipart/form-data"
ACTION="cgi-bin/uploadcgi.cgi">
      选择文件：<INPUT TYPE="FILE" NAME="FILE1" id="FILE1" class="btn"><INPUT
TYPE="button" onclick="upload_file()" VALUE="上传" class="btn"><br>
      <center>
        <div id="uploadInfo"></div>
      </center>
    </FORM>
  </BODY>
</HTML>


###########################
uploadcgi.c


/***********************************************************************
    2007-1-5 11:42 establish by lzh.A cgi program.
    get a file from user's explorer.
***********************************************************************/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define DEAL_BUF_LEN  1024
#define SIGN_CODE_LEN  100
```

```
#define SIGN_CODE_LEN 100
#define FILE_NAME_LEN 64
#define FILE_SAVE_DIR "/var/landog/db_upload/"

enum
{
  STATE_START,
  STATE_GET_SIGN_CODE,
  STATE_GET_FILE_NAME,
  STATE_GET_FILE_START,
  STATE_GET_FILE_CONTENT,
  STATE_CHECK_END,
  STATE_END
};
/*****************************************************************************
ShowErrorInfo
*****************************************************************************/
static void ShowErrorInfo(char * error)
{

  printf("Content-Type:text/html;charset=UTF-8\n\n");

  printf("<center><font color='red'>%s</font></center>" , error );
}

/* 主体从这里开始 */

int main(void)
{
  FILE *fp; /* 文件指针，保存我们要获得的文件 */
  int getState = STATE_START;
  int contentLength;/*标准输入内容长度*/
  int nowReadLen;
  int signCodeLen;
  int tmpLen;
  char *nowReadP;
  char *nowWriteP;
  char dealBuf[DEAL_BUF_LEN];
  char signCode[SIGN_CODE_LEN]; /*存储本次的特征码*/
  char tmpSignCode[SIGN_CODE_LEN];
  char fileName[FILE_NAME_LEN];
  memset(dealBuf,0,DEAL_BUF_LEN);
  memset(signCode,0,SIGN_CODE_LEN);
  memset(fileName,0,FILE_NAME_LEN);
  nowReadLen = 0;
  if((char *)getenv("CONTENT_LENGTH")!=NULL)
  {
    contentLength = atoi((char *)getenv("CONTENT_LENGTH"));
  }
  else
  {
    ShowErrorInfo("没有恢复数据!");
    exit(1);
  }

  while(contentLength > 0)
  {
    if(contentLength >= DEAL_BUF_LEN)
    {
      nowReadLen = DEAL_BUF_LEN;
    }
```

```c
        else

        {
            nowReadLen = contentLength;
        }
        contentLength -= nowReadLen;
        if(fread(dealBuf,sizeof(char),nowReadLen,stdin) != nowReadLen)
        {
            ShowErrorInfo("读取恢复数据失败，请重试！");
            exit(1);
        }
        nowReadP = dealBuf;
        while(nowReadLen > 0)
        {
            switch (getState)
            {
              case STATE_START:
                nowWriteP = signCode;
                getState = STATE_GET_SIGN_CODE;
              case STATE_GET_SIGN_CODE:
                if(strncmp(nowReadP,"\r\n",2) == 0)
                {
                    signCodeLen = nowWriteP - signCode;
                    nowReadP++;
                    nowReadLen--;
                    *nowWriteP = 0;
                    getState = STATE_GET_FILE_NAME;
//ShowErrorInfo(signCode);
                }
                else
                {
                    *nowWriteP = *nowReadP;
                    nowWriteP++;
                }
                break;
              case STATE_GET_FILE_NAME:
                if(strncmp(nowReadP,"filename=",strlen("filename=")) == 0)
                {
                    nowReadP += strlen("filename=");
                    nowReadLen -= strlen("filename=");
                    nowWriteP = fileName + strlen(FILE_SAVE_DIR);
                    while(*nowReadP != '\r')

                    {
                        if(*nowReadP == '\\' || *nowReadP == '/')
                        {
                            nowWriteP = fileName + strlen(FILE_SAVE_DIR);
                        }
                        else if(*nowReadP != '\"')
                        {
                            *nowWriteP = *nowReadP;
                            nowWriteP++;
                        }
                        nowReadP++;
                        nowReadLen--;
                    }
                    *nowWriteP = 0;
                    nowReadP++;
                    nowReadLen--;
                    getState = STATE_GET_FILE_START;
                    memcpy(fileName,FILE_SAVE_DIR,strlen(FILE_SAVE_DIR));
```

```
                if((fp=fopen(fileName,"w"))==NULL)
                {
                    fprintf(stderr,"open file error\n");
                    exit(1);
                }
//ShowErrorInfo(fileName);
            }
            break;
        case STATE_GET_FILE_START:
            if(strncmp(nowReadP,"\r\n\r\n",4) == 0)
            {
                nowReadP += 3;
                nowReadLen -= 3;
                getState = STATE_GET_FILE_CONTENT;
//ShowErrorInfo("get");
            }
            break;
        case STATE_GET_FILE_CONTENT:
            if(*nowReadP != '\r')
            {
                fputc(*nowReadP,fp);
            }

            else
            {
              if(nowReadLen >= (signCodeLen + 2))
              {
                if(strncmp(nowReadP + 2,signCode,signCodeLen) == 0)
                {
                    getState = STATE_END;
                    nowReadLen = 1;
                    ShowErrorInfo("数据上传成功");
                    /*   if( system( "/var/landog/app/sniff/db_recover.sh" ) == 0 ){
                            ShowErrorInfo( "数据库恢复完成，请重新启动landog" );
                          }else{
                            ShowErrorInfo( "数据库恢复过程中出现错误，错误原因：恢复的文件已
经损坏" );
                        }
                    */
                }
                else
                {
                    fputc(*nowReadP,fp);
                }
              }
              else
              {
                getState = STATE_CHECK_END;
                nowWriteP = tmpSignCode;
                *nowWriteP = *nowReadP;
                nowWriteP++;
                tmpLen = 1;
              }
            }
            break;
        case STATE_CHECK_END:
            if(*nowReadP != '\r')
            {
              if(tmpLen < signCodeLen + 2)
              {
                *nowWriteP = *nowReadP;
```

```
                nowWriteP++;
                tmpLen++;

                if(tmpLen == signCodeLen + 2)
                {
                   *nowWriteP = 0;
                   if((tmpSignCode[1] == '\n')&&(strncmp(tmpSignCode +
2,signCode,signCodeLen) == 0))
                     {
                        getState = STATE_END;
                        nowReadLen = 1;
                        ShowErrorInfo("数据上传成功");
                        /*   if( system( "/var/landog/app/sniff/db_recover.sh" ) == 0 ){
                            ShowErrorInfo( "数据库恢复完成，请重新启动landog" );
                          }else{
                            ShowErrorInfo( "数据库恢复过程中出现错误，错误原因：恢复的文件
已经损坏" );
                          }
                        */
                     }
                   else
                     {
     //fprintf(fp,tmpSignCode);
                        fwrite(tmpSignCode,sizeof(char),tmpLen,fp);
                        getState = STATE_GET_FILE_CONTENT;
                     }
                }
              }
            }
          }
          else
          {
            *nowWriteP = 0;
     //fprintf(fp,tmpSignCode);
            fwrite(tmpSignCode,sizeof(char),tmpLen,fp);
            nowWriteP = tmpSignCode;
            *nowWriteP = *nowReadP;
            nowWriteP++;
            tmpLen = 1;
          }
          break;
        case STATE_END:
          nowReadLen = 1;
          break;

        default:break;
      }
      nowReadLen--;
      nowReadP++;
    }
  }
  if(fp != NULL)
  {
    fclose(fp);
  }
  return 0;
}
```

上一篇：CGIC简明教程

类似文章

unsigned char 与 char

C语言面试题大汇总之华为面试题(3)(

编程实现字符串长度测量

char与unsigned char的综合比较及实例

几个常见的C语言面试题分析

Re: 请问有谁用过zlib库的? - China Lin...

struts 1.2 文件上传与下载

提升权限终极技巧

秀出自我 10分钟个性QQ表情自己做

更多

您可能会喜欢

CGI

CGI之C语言篇

用C语言进行CGI程序设计

PHP-FPM与spawn-CGI

CGI原理及其性能

无觅

发表评论：

您好，请 登录 或者 注册 后再进行评论

使用合作网站登录： 新浪微博   QQ   人人