# 1、lanyui样式调整

layui-upload-self

# 2、修改效果显示

（点击文件名称可以查看文件）



# 3、列表显示



# 4、函数说明

a)：修改页面样式显示

/\*\*

\* 多文件上传显示源文件名称,点击文件名在新页面中打开

\* @param array $source\_data 已经有的数据

\* @param array $other\_config 其他配置项

\* @param string $input\_name 获取的input名称

\* @param string $upload\_name 前面的上传名称提示

\* @param string $div\_width 上传范围大小

\* @return string

\*/

$other\_config = array(

'ajax\_data' => '{}',//ajax 发送的数据

'show\_upload\_div' => 'demoList',//上传后展示的div id

'click\_button' => 'testList',//点击选择button的id

'upload\_button' => 'testListAction',//上传按钮button的id

'upload\_type' => 'file',//上传类型【images（图片）、file（所有文件）、video（视频）、audio（音频）】

'upload\_url' => '',//上传处理的url

'max\_size' => bcmul(floatval(ini\_get('post\_max\_size')),1024),//最大上传大小

'allow\_ext' => array('jpeg','jpg','png','pdf','doc','docx','xls','xlsx','txt','csv')//允许的类型

);

FileUploadUseSourceName($source\_data = array(),$other\_config = array(),$input\_name = '',$upload\_name = '文件选择',$div\_width = '800')

b): 上传好查看页面

/\*\*

\* 列表显示

\* @param array $source\_arr 处理好的数据

\* @param array $other\_config 一些配置项，需要添加

\* @return string

\*/

FileShowUseSourceName($source\_arr = array(),$other\_config = array())

c)：字符串解析文件字符串

/\*\*

\* 解析存好的数据结构 上传文件路劲|源文件名,....

\* @param string $source\_str 数据结构

\* @return array 处理好的数据

\*/

AnalysisUseSourceNameData($source\_str = '')

d)：文件上传

/\*\*

\* 上传文件

\* @param string $upload\_type 上传类型 $uploadPath中的key

\* @param string $object\_name 上传文件名称 file中的name值

\* @return array

\*/

UploadFileNoLimitType($upload\_type,$object\_name)

e)：通过后缀名获取mine【是个汇总集】

/\*\*

\* 全部的类型

\* @param array $allow\_ext 允许上传的类型

\* @return array|mixed 获取到的类型mine

\*/

FileUploadExtToMine($allow\_ext = array())

f)：基础的检查图片中是否包含木马【可执行语句】

**使用preg\_match匹配特殊字符串**

/\*\*

\* 基础的检查图片中是否包含木马

\* @param string $path 文件路劲

\* @return bool true是木马，false不是

\*/

FileUploadCheckPicVirus($path)

g）：字符串转换为16进制：先ASSCI后转，要不然转不了

/\*\*

\* 字符串转换为16进制

\* @param string $str 转换的字符串

\* @return string 转换好的16进制的字符串

\*/

function getStrHex($str)

k)：图片重绘【去除木马】

**说明：图片重绘也不能完全阻止文件木马操作**

**paper中说到了对于这种重新生成图片的防御方法，也有两种攻击方式**

**1. 利用数据二义性，构造符合图像数据格式的木马代码，在paper中只是个思路，不知道现在能不能正确构造出来这样的图像文件**

**2. 利用溢出,攻击生成新图片的函数，至于什么样的恶意图片会使得这个函数溢出**

/\*\*

\* 图片重绘

\* @param string $ext 图片扩展名

\* @param string $path 文件路劲，源文件名称替换

\*/

function imageRedraw($ext,$path)