#### Pass-16

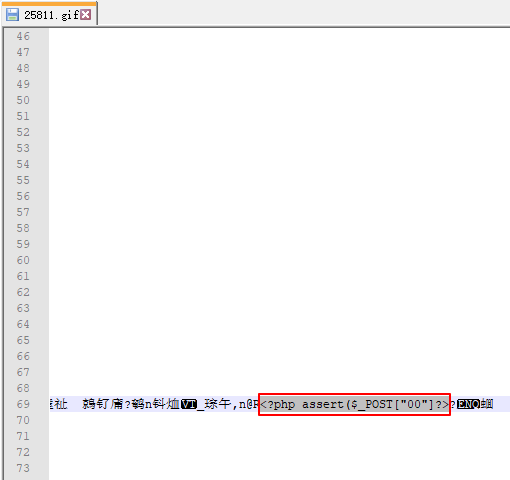
Imagecreatefromgif函数：这个函数会把我们图片的信息和非图片的信息分离开来，如果我们在图片里加入一句话木马，他会把木马摘出来过滤。

原理：将一个正常显示的图片，上传到服务器。寻找图片被渲染后与原始图片部分对比仍然相同的数据块部分，将Webshell代码插在该部分，然后上传。具体实现需要自己编写Python程序，人工尝试基本是不可能构造出能绕过渲染函数的图片webshell的。

这里提供一个包含一句话webshell代码并可以绕过PHP的imagecreatefromgif函数的GIF图片[示例](https://github.com/LandGrey/upload-labs-writeup/blob/master/webshell/bypass-imagecreatefromgif-pass-00.gif)。

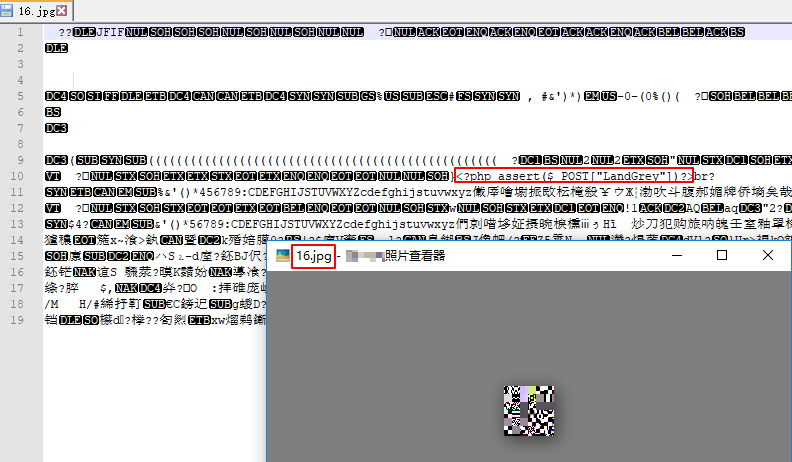
[](https://github.com/LandGrey/upload-labs-writeup/blob/master/image/16-1.png)

打开被渲染后的图片，Webshell代码仍然存在

[](https://github.com/LandGrey/upload-labs-writeup/blob/master/image/16-2.png)

提供一个jpg格式图片绕过imagecreatefromjpeg函数渲染的一个[示例文件](https://github.com/LandGrey/upload-labs-writeup/blob/master/webshell/bypass-imagecreatefromjpeg-pass-LandGrey.jpg)。 直接上传示例文件会触发Warning警告，并提示文件不是jpg格式的图片。但是实际上已经上传成功，而且示例文件名没有改变。

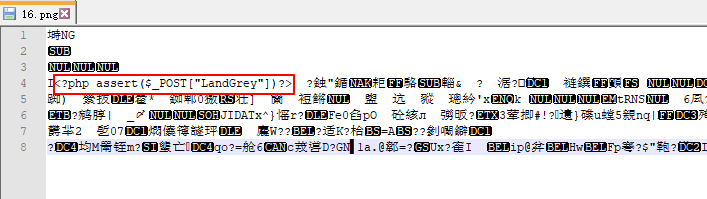
[](https://github.com/LandGrey/upload-labs-writeup/blob/master/image/16-3.png)

[](https://github.com/LandGrey/upload-labs-writeup/blob/master/image/16-4.png)

从上面上传jpg图片可以看到我们想复杂了，程序没有对渲染异常进行处理，直接在正常png图片内插入webshell代码，然后上传[示例文件](https://github.com/LandGrey/upload-labs-writeup/blob/master/webshell/bypass-imagecreatefrompng-pass-LandGrey.png)即可，并不需要图片是正常的图片。

[](https://github.com/LandGrey/upload-labs-writeup/blob/master/image/16-5.png)

程序依然没有对文件重命名，携带webshell的无效损坏png图片直接被上传成功。

[](https://github.com/LandGrey/upload-labs-writeup/blob/master/image/16-6.png)