【代码审计】某开源商城前台Getshell

水泡泡 / 2018-02-05 16:07:00 / 浏览数 4261 安全技术 漏洞分析 顶(0) 踩(0)

0x00 前言
常见的文件上传，逻辑处理问题，发出来让大家看看。

0x01 审计入口

看到inc\function\global.php 文件

这套系统用了360的防护代码

对get，post,cookie都进行了过滤，但有一点挺有趣的。

在p=admin的时候，post数据是不会过滤的，目测后台是有sql执行这样的功能。（后台还没看）

在下面发现，对get，post数据进行了全局变量注册。可以考虑有没有全局变量覆盖的问题

在前台用户中心，头像上传的地方发现了个。

0x02 漏洞分析

看到文件inc\module\upload_img.php

这个文件用来处理用户头像上传的功能。

在一开始就用白名单写死了后缀名。

接着往下看，看到127行这里

仔细分析代码，这几行代码的意思是获取上传文件的后缀名，并判断是否在允许的后缀名白名单内。

看看这个判断，这是一个与判断，需要前后条件同时成立，才会进入if语块内。

看到后面的条件，判断是否有设置is_h5这个变量？？

仔细看看前面的代码，发现并没有获取这个变量的代码，没有初始化，也就是我们可以通过全局变量注册的方式，覆盖整个变量的值，从而绕过白名单判断。

继续往下看，当设置了is_h5变量的时候，也就意味着要使用h5上传的方式来上传头像。

但这里又出现了问题。

分析这里的if判断，同样是与判断，需要同时成立。变量is_h5 的值要等于1的时候才会进入h5上传代码内。

我们可以设置is_h5变量的值不为1即可。

代码继续往下走。

清除之前的图片

写入到临时文件中

从临时文件中读取内容并写入

注意到

这里的$uploadPath在上面有设置：

后缀名是我们传入的。

一个文件上传至getshell漏洞也就达成了。

总结一下，漏洞利用，设置一下is_h5的值，修改上传的文件后缀名为php即可。

至于文件名，程序在最后有输出路径

0x03 一些坑

1，shell的文件路径问题

因为程序在后面，会判断上传的路径是不是在avatar目录下，如果是的话就会把缩略图的路径覆盖uploadPath。

而这个smalltargetFile 是这样复制的。

比原路径中间多了一个_s。

比如我们获取到的路径是这样的。

这个是缩略图的路径：upload\img\avatar\20180125\a1d3bce4bf71c368eb687d89b231f136_s.php

原来的路径只要把_s去掉就好了：

upload\img\avatar\20180125\a1d3bce4bf71c368eb687d89b231f136.php

当然，我们可以手动修改avatar为其他的，只要在程序的白名单内就好。

2，缩略图问题

其实缩略图也是php文件，但是这里是经过php-gd库的，我用jpg_payload.php生成的图片马过了n次都是没成功。望大佬们能指点一下。

0x04 漏洞复现

注册一个会员，来到个人信息设置里，点击上传头像，抓到这样的数据包：

设置一下is_h5的值，不为1就行。

下面的文件名也要改一下后缀。

返回路径：

去掉_s ，请求就是shell地址了。

0x05 总结

感觉php的弱类型（之前的dede前台用户密码修改），全局变量注册问题，虽然这里没有弱类型比较问题，但依旧是php类语言源码的审计重点。

之前一直没发现前面还生成了一个图像原文件，一直想着过gd库耗费了不少时间。

上一篇：Apache ActiveMQ A... 下一篇：渗透测试 -- VulnHub –...

1. 1 条回复

www.xss.tv 2018-03-07 21:51:28

厉害了，我的哥

0 回复Ta

目录