DEDECMS5.7后台getshell

dedecms5.7爆出了一个新的漏洞，可以后台getshell。
(https://mp.weixin.qq.com/s/IZn_xnO2tyUWmx9dronYUQ) 还挺有意思的，大家可以跟一下。

漏洞的点主要是在开发者在调用了通用的文件处理函数之后，又对文件进行了进一步的处理。

```
if(!empty(${$_key.'_name'}) && (preg_match("#\.
(".$cfg_not_allowall.")$#i",${$_key.'_name'}) || !preg_match("#\.#",
${$_key.'_name'})) ){
        if(!defined('DEDEADMIN'))
        {
            exit('Not Admin Upload filetype not allow !');
        }
} //uploadsafe.inc.php
```

```
$imgfile_name = trim(preg_replace("#[ \r\n\t\*\%\\\/\?><\|\":]{1,}#", '',
$imgfile_name));
if(!preg_match("#\.(".$cfg_imgtype.")#i", $imgfile_name))
{
    ShowMsg("你所上传的图片类型不在许可列表，请更改系统对扩展名限定的配置！", "-1");
    exit();
}
$nowtme = time(); //select_images_post.php
```

其实公共函数处理没多大问题，但是因为后面将一些特殊字符替换成空格，所以这个时候思路就有了。在上传时将文件名改成夹杂特殊字符的文件名。如`1.jpg?p*h%p`。

但是稍微坑爹的地方来了，在`select_images_post.php`的最后调用了`getimagesize()`，如果为false则直接`die()`。但是其实getimagesize是可以绕过的，原因在于它判断imagetype时只是判断前三个字节，因此只要前三个字节符合jpg等图片格式即可。有文章提到这部分的底层细节，值得看看。

http://0x1.im/blog/php/php-function-getimagesize.html

因此，最后成功实现了后台传shell。

所以说，开发的在涉及一些敏感操作前，最好看看底层实现。我相信类似的漏洞在其他CMS也是存在的，可以审一波。