



## Web

### Cosmos的博客后台

#### 后台登陆

用户名
密码
Submit

看到登录界面，一开始想到的就是sql注入，但是一直写不出，直到后来学长给了hint，才知道这题是本地文件包含漏洞.....百度了一堆资料，知道了php://filter可以通过post来执行指定的php代码，一开始一直以为要从username和password传，后来发现原来可以直接通过action来传递(QAQ

#### 二、php://filter

php://filter可以获取指定文件源码。当它与包含函数结合时，php://filter流会被当作php文件执行。所以我们一般对其进行编码，让其不执行。从而导致 任意文件读取。

POC1:

```
?file=php://filter/resource=xxx.php
```

POC2:

```
?file=php://filter/read=convert.base64-encode/resource=xxx.php
```

POC1直接读取xxx.php文件，但大多数时候很多信息无法直接显示在浏览器页面上，所以需要采取POC2中方法将文件内容进行base64编码后显示在浏览器上，再自行解码。

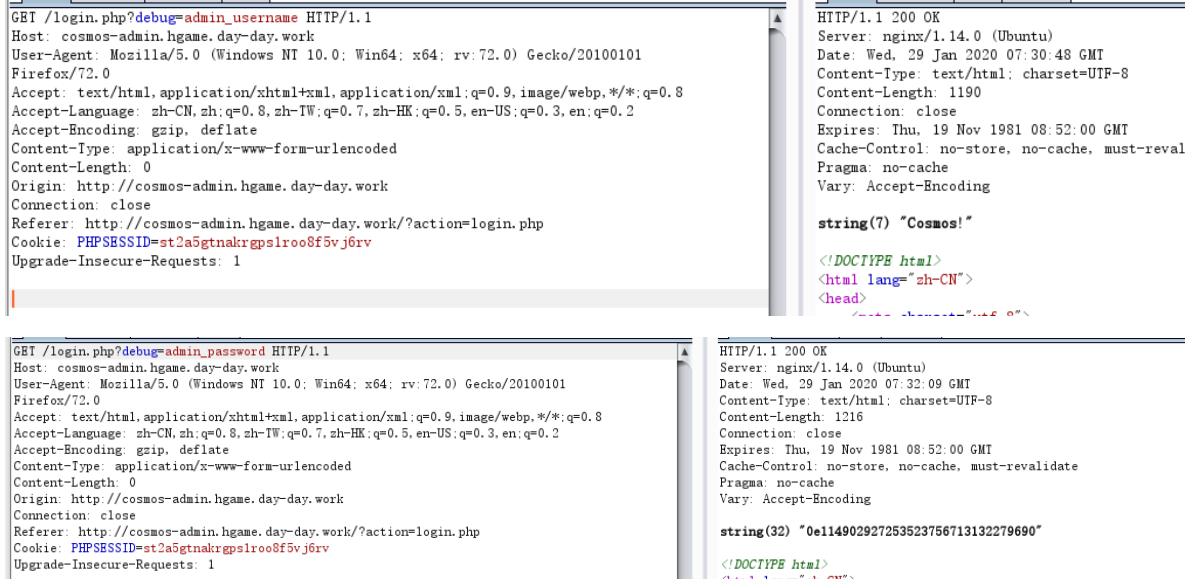
第一步，构造php语句，读取login.php和admin.php的内容

InLnBocCI7CnNlc3Npb25fc3RhcnQoKTsKCj8vT25seSBmb3lgZGVidWcKaWYgKERFQlVHX01PREUpewogICAgYWYoaXNzZXQoJF9l

base64解码后得到login.php的内容，考察代码审计

分析：我们需要达成以下条件登录，而admin\_password和admin\_username可以通过GET传入  
DEBUG\_MODE中的debug变量，匹配了正则表达式后，由eval("var\_dump(\$debug);输出

```
if ($admin_password == md5($_POST['password']) && $_POST['username'] === $admin_username)
```



得到管理员的用户名和密码，密码是0e开头的md5值，刚好查资料的时候看到过(XD,考察的是Hase漏洞

## 漏洞描述

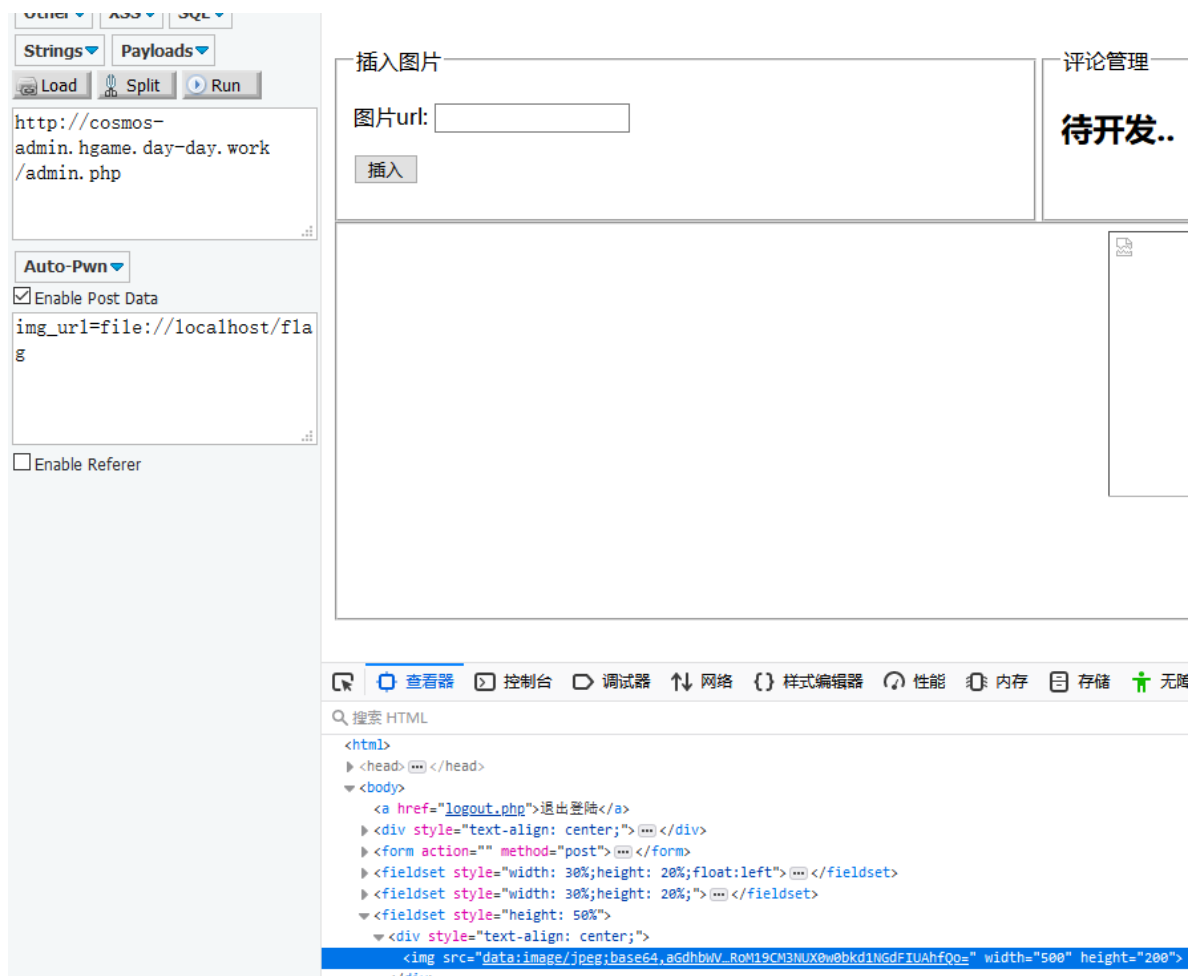
PHP在处理哈希字符串时，会利用“!=”或“==”来对哈希值进行比较，它把每一个以“0E”开头的哈希值都解释为0，所以如果两个不同的密码经过哈希以后，其哈希值都是以“0E”开头的，那么PHP将会认为他们相同，都是0。

攻击者可以利用这一漏洞，通过输入一个经过哈希后以“0E”开头的字符串，即会被PHP解释为0，如果数据库中存在这种哈希值以“0E”开头的密码的话，他就可以以这个用户的身份登录进去，尽管并没有真正的密码。

**即：**如果md的值是以0e开头的，那么就与其他的0e开头的Md5值是相等的。



关于解决这个漏洞可参考 <http://www.freebuf.com/news/67007.html>

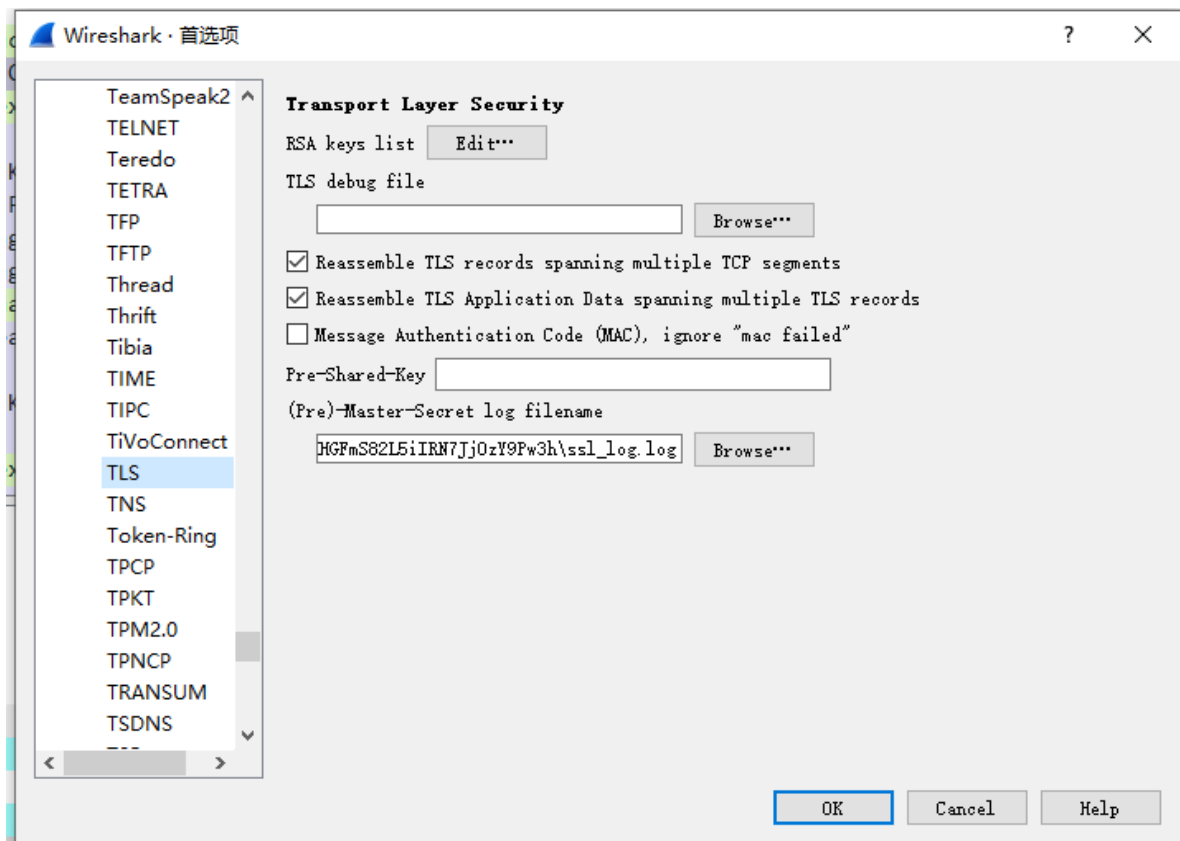
登录后发现可以插入图片url，考察了SSRF，题目提示flag在根目录，通过未过滤的url，利用file：//得到flag，这一步一开始我还以为是错的，因为图片无法显示.....结果后面发现源码里给了一串base64，解密后就是flag



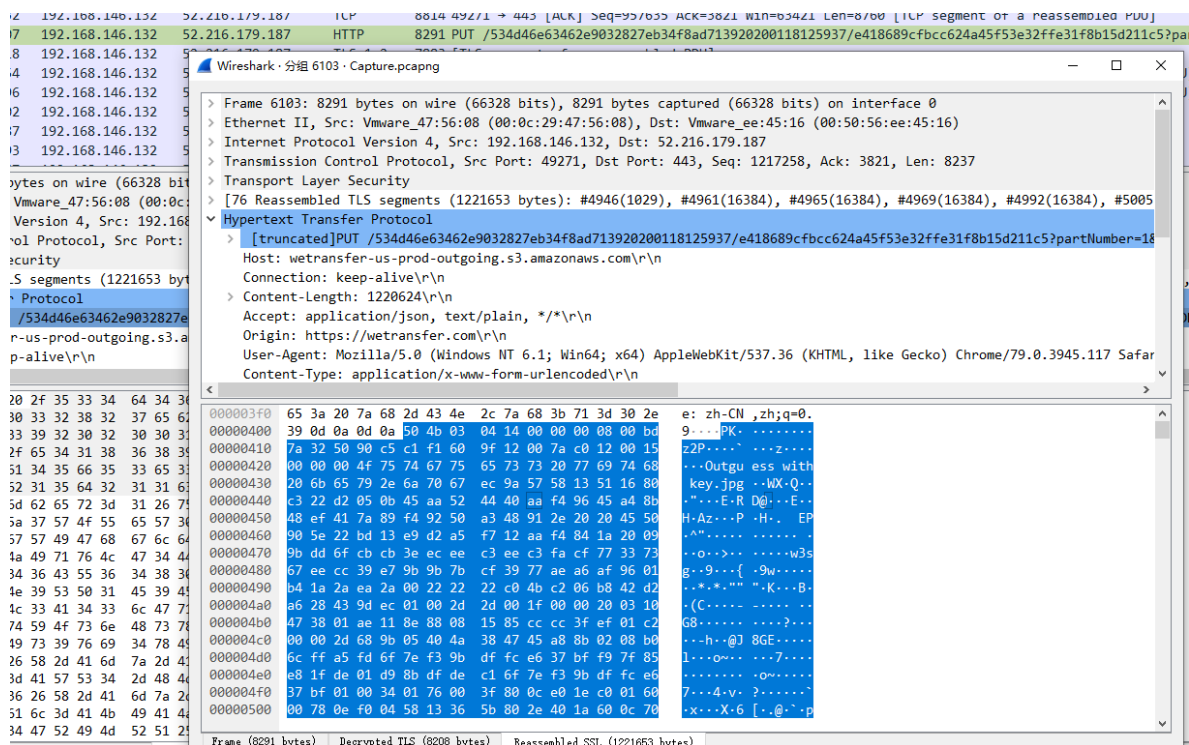
## misc

### Cosmos的午餐


 Capture.pcapng	2020/1/18 21:01	Wireshark captu...	6,858 KB
 ssl_log.log	2020/1/18 20:59	文本文档	20 KB




Firefox和Chrome浏览器都支持用日记文件的方式记录下用来加密TLS数据包对称会话密钥,导入日志文件



发现一个带有压缩包和jpg的流量包, 导出用foremost分离



00000000.zip



Outguess with key.jpg

分级 ☆☆☆☆☆

标记

备注 Key: gUNrbbdR9XhRBDGpzz

来源

作者

拍摄日期

程序名称

得到一个压缩包解压出来一张图片，这里一开始我还以为是从key中找flag，后来发现outguess其实是一种图片隐写的方法，用outguess解密后得到一个txt，里面是一个地址，打开后下载了一个压缩包

```

root@ubuntu:/home/r4inyini9ht/outguess# outguess -k "gUNrbbdR9XhRBDGpzz" -r '/home/r4inyini9ht/Desktop/Outguess with key.jpg' hidden.txt
Reading /home/r4inyini9ht/Desktop/Outguess with key.jpg....
Extracting usable bits: 1161827 bits
Steg retrieve: seed: 3, len: 24
  
```

打开是一个二维码



扫码得到flag，（我还以为还要再来一个二维码的考点XD

## 所见即为假

output	2020/1/23 21:38	文件夹	
AllFake_XK2ipRXBI3Usi17r57EmawrO...	2020/1/23 20:07	ZIP 压缩文件	1,040 KB
FLAG_IN_PICTURE.jpg	2020/1/20 15:28	JPG 文件	1,043 KB
foremost.exe	2019/11/24 11:06	应用程序	88 KB



```
<?xml version="1.0" encoding="UTF-8" standalone="true"?>  
<flag>hgame{mkLbn8hP2g!p9ezPHqHuBu66SeDA13u1}</flag>
```